Senior Design Research Document

Senior Design II

EEL 4915

Energy Sustainable Hydroponics with Automated Reporting and Monitoring

Group 15 - Members

David Mascenik (Electrical Engineer)

Jon Spychalsky (Electrical Engineer)

James Tooles (Electrical Engineer)

Table of Contents

1.0 Executive Summary

Creative potential can take many forms, through art, music, and writing. It has only been recent for technology to also be considered an artistic outlet because of one word, design. The word 'design' is used across all creative disciplines, to convey an idea 'intentionally' or with 'purposeful action' on behalf of the designer. They have an idea, a message that they would like to deliver to the world. As engineers, we are artists with an emphasis on designing functional systems, with goals such as high performance, efficiency, and durability. We are effected by our environment, its political and economic climates influencing our design choices, thus our group wished to partake in a project with the potential to better ourselves and skills as engineers while also making a contribution to an interesting topic where we could apply our abilities.

Hydroponics is a very new horticulture technique when comparing it to the timescale of several millennia that humans have typically cultivated crops using soil. It holds a modern promise for a future civilization where crops may be grown in urban centers, on rooftops, skyscrapers, underground bunkers; anywhere there is electricity and water. Hydroponic science has found methods with which to cultivate most crops either faster, higher yield, or both. Research and the free-market continue to develop the field and economic cost is projected to decrease as well, exciting further growth and opportunity for enthusiasts and businesses alike.

It is a rare opportunity to work on a project which has such a range of implications, we must hold our ethical standards high as our project is mutually influenced by factors such as the environment, politics, economics, and local social groups. Within these bounds, our project also had a plethora of design potential we could exploit: control systems, communication protocols, power distribution, and remote access; truly, our project had many aspects to define and deliberate. Our group diligently designed our project with the principles of effectiveness, efficiency, and cost-competitiveness right next to our computers and breadboards. Our document specifies the research and designs we gathered and devised, an action plan with which to manufacture our systems and controls, ultimately functioning to explore and contribute our ideas to the topic of hydroponics.

The majority of this document covers the automation systems and network used to organize important hydroponic data and appropriately respond. A brief overview of hydroponic science is given to familiarize the reader with the requirements of our project, the rest of the documentation detailing research and design of sensor networks, automation, power distribution, and information technology.

2.0 Project Description

The project aimed to automate some of the processes necessary in maintaining a hydroponics system. Our automation system, which we designed, built and tested, was attached to a hydroponics system and automate water quality, water circulation, and lighting while displaying data for analysis by the user by means of a LCD. The objectives of this senior design project was geared toward gaining experience with sensors, microcontrollers, power regulators, and solar power systems while meeting our requirements through carefully defined specifications.

2.1 Project Motivation and Goals

Our hydroponics project is an interesting subject which our group can maintain an active interest in the background material, hydroponic science, while applying our disciplined knowledge as electrical engineers to automate the system. The following is a brief overview of hydroponics and the important parameters necessary for an effective system.

2.1.1 Introduction to Hydroponic Science

The horticulture technique known as hydroponics has had many implications on agricultural science. With its only requisite being the use of any growing substrate other than soil, many variants of hydroponics have been implemented and researched. The plants typically founded in a substrate, while nutrient-rich water is delivered directly to the roots. This accelerates plant growth because as the roots develop, the increased surface area absorbs the nutrient rich water, allowing for faster growth with time. Indeed, plant biomass and water content is found to be greater in plants grown hydroponically than those cultured in soil.

Faster growth isn't the only advantage for hydroponics over using soil:

- Greater Resistance to Pests and Vermin
- More Biomass (higher crop yields)
- Control of Nutrients (Convenient for Studying Different Blends)
- Easier to Maintain
- Scalable/Modular

While there are many methods with which to culture plants with hydroponics, our group decided the Nutrient Film Technique (NFT) would be the most appropriate for our project. It offers the capacity for features we would like to implement in our design goals: outdoor use with solar charging and automation.

The NFT method of hydroponics is characterized by a structure which supports the plants as a stream of water with dissolved nutrients, shallow enough for the roots to be mostly exposed to the air, flows over the bottom of the plants' root system. Hobbyists and professionals typically pump a nutrient solution to the top of the structure and then use narrow channels to irrigate the water into a reservoir, where the solution is then pumped back up again to the top of the structure. Advantages of the Nutrient Film Technique is that the nutrient solution is turbulent, making the growth of fungi and molds more difficult, and the constant exposure to air, a necessary resource for plants which if grown in soil rely on small pockets, depressing growth. An example diagram of a NFT system is illustrated in Figure 2.1.

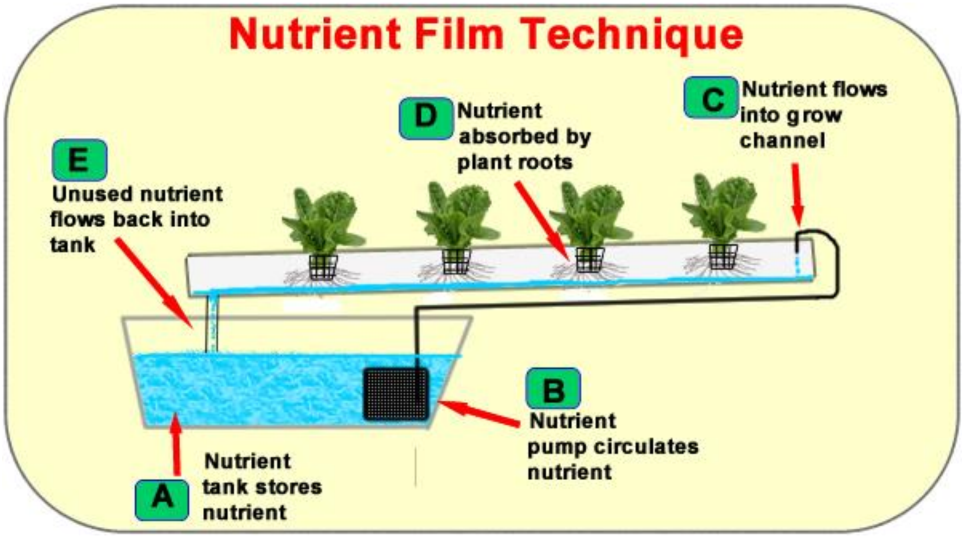


Figure 2.1: Operation of a NFT Hydroponic System (Permission Pending)

The disadvantage, referring to the diagram, of an NFT system is that although growth of fungi and mold is inhibited by the turbulence of the water, the channels, hose connections, and reservoir must be kept clean and enclosed from open air. If a critical mass of spores were to contaminate the system, colonies would spawn on the plant roots, inhibiting growth and ultimately exterminating the plants. To prevent this, the solution much be renewed about once a week to lower the risk of infection and rebalance the nutrient levels.

2.1.2 Important Parameters

Each species of plant has different tolerances of their environment and need to be cultured such that they experience the least amount of stress. Abrupt changes in light, temperature, and other variables can reduce a plant's growth or exterminate it all together if too stressful. Thus the following parameters were automated and data from sensor analysis displayed for the user within the project:

2.1.2.1 Light

A specie's requirement for light, factors including light's intensity and duration, required consideration while designing our system. Light is necessary for the photosynthetic processes with the plant, which converts light and carbon dioxide into water and simple sugars. In the event there isn't enough light, our system accommodates the plants with a light array matching the average light intensity the plant requires.

2.1.2.2 Temperature

Since our system was designed for outdoor use, temperature data is being displayed in real time for the user. This information is valuable for the gardener to interpret stress points for the plants, better planning schedules for planting for seasons. The system could well be redesigned for indoor use and the temperature data employed to control the temperature of the environment via feedback controllers.

2.1.2.3 Dissolved Oxygen

More advanced systems for hydroponics include dissolved oxygen generators to control the levels of dissolved oxygen in the environment. Since we evaluate the system outside and with dissolved oxygen generators being too expensive for our budget, our group came to the consensus on a control system. The design allows for expansion if we would like to implement a dissolved oxygen generator at a later time. The dissolved oxygen sensor is used to monitor the dissolved oxygen levels, and if needed the system water flush is activated.

2.1.2.4 pH

A second test for water quality, pH measurements measure alkali levels, dependent on species of plant used. The measurements also control the flushing mechanism, activated when levels are out of desired range. This is extremely useful for the latter half of the vegetative stage when the growth of the plant will greatest, thus needing nutrients replenished frequently.

2.1.3 Gain Knowledge and Skills

The project offered our group the chance to independently acquire skills and knowledge which interest us, an interdisciplinary set of problems ranging from mechanical and PCB manufacturing to software development. Our group divided the requirements according to our individual interests in the project, but the following were the necessary skills required to complete the hydroponics system.

2.1.3.1 Sensor Interfacing

There are several interfaces with which modern sensors communicate with MCUs, differing in bandwidth, pin count, and power usage. Reading sensor values is critical for most if not all industries and research fields, from automated processes to experimental data collection. Having no classwork on this material, it was to our group's benefit to acquire this skillset before going into the electrical engineering industry. It was our intention to use at minimum three sensors, evaluating one or more of the important hydroponic parameters each. After the system was completed, a total of eight sensors were used and their values displayed on a LCD screen.

2.1.3.2 Automation

For the sake of efficiency and control, automating routine processes is a valuable skill. It allows an electro-mechanical system to control a task, executing it more efficiently and reliability than a human operator. Designing such a system offers the human operators more time to perform other tasks that cannot be automated such systems.

2.1.3.3 System Integration

Often is the case where it makes more economic sense to purchase a module which performs a function rather than spending resources such as engineering salary, time and material costs, to design a module which performs the exact same function. Researching, selecting, and integrating such modules is an experience not found in coursework but rather through projects such as senior design. We incorporated a few of these modules ourselves, carefully sourcing modules and parts that are essential to making our project perform more efficiently.

2.1.3.4 Web Integration

Out of personal motivation and identifying a marketable need for web programmers, we wanted our project to include experience with web interfacing. The transition of data from our sensors to controller to web server would have implemented various communication protocols and technologies. The hydroponic system would have ideally included an RF communication to our webserver, as RF protocols are essential to most modern remote sensing designs and protects our webserver from outdoor elements. Our project presents an exciting opportunity to showcase our problem solving skills and practice our programming experience.

2.1.3.5 Linux Development

Embedded Linux environments are becoming more popular platforms as low footprint processors, such as ARM, continue to improve their technology, leveraging more computationally capable devices. This developing trend deserves attention, as it would be a very powerful and marketable skill to know for industry. Boards such as the Raspberry Pi or Beaglebone Black would be appropriate to use considering our experience and budget for the project. These modules have sufficient performance to be implemented as our webserver, since our functionality requires at most two to three clients connected at any time to view stored data.

2.2 Objectives

The objectives of the project met the following:

1. Gained knowledge and skills explicitly outlined in the project motivation and goals.

2. Obtained experiences with working in a professional engineering group.

3. Developed a system which showcases our combined experiences from coursework, academic research and internships with industry.

4. Became familiar with tools, software, and manufacturing methods common to developing products for the consumer market.

These objectives benefitted our project through having had a moral statement with which to abide by: pursued the best possible design with consideration to performance, cost, manufacturability and time. Efficient compromises were made in order for the “best design” to be realized. Each of our group members had an idea in mind for what our project was to accomplish, thus with these explicit objectives we agreed to a common set of standards with which we based our decisions.

2.3 Requirements Specifications

The following table is a series of requirements and specifications for our final project design. These parameters consist of constraints that were respected and specifications which are design goals independent of the project's successful operation. General and hydroponic attributes describe system level details, defined for the hydroponics setup, independent of automation details. Power, communication, and successive sections describe details for automation and information technology systems. Please reference Table 2.1. for information on featured requirements and specifications.

Attribute	Value
General	
# of Plants	4
Weight of System	30 lbs
Dimensions	4ft x 2ft
Reservoir Capacity	4 Gallons
Power	
Consumption (Automation Subsystems Off)	2W
Consumption (Automation Subsystems On)	50W
Solar Panel Output	40W
Logic operating Voltage	5V
Automation Voltage	12V
Battery Capacity	25Ah
Communication	
Signal Power	+10dBm
Protocol	Bluetooth
Data Rate	2Mbps
Automation Sub	
Water Pump Head Pressure	4ft
Water Pump Throughput	500 GPH
Tiller Feed Rate	50cc/Min
Light Array Luminance	35 Lumens
Light Array Power	8W
MCU	
Min. Sensor Interfaces	5
In-System Programmable	Yes
Hydroponic Science	
Dissolved Nutrient	10g/10L of Water
pH	7.6
Min. Water Change	1 per week
Test Plant	Tomatoes
Temperature	Min. 65° F
Min. Light Intensity	30 Lumens
Light Exposure	16 Hours per Day

Table 2.1 - Requirements and Specifications Table

3.0 Research Related to Project Definition

This section is a compilation of the research done for our project. The first section, Existing Similar Projects and Products, presents our research and analysis of projects that are related to ours, whether commercial, academic, or hobbyist. The second section, Relevant Technologies, shows the research on relevant technologies for our project. This gave us an idea of how exactly various different components and technologies could work together for our project, ultimately allowing us to design and prototype more efficiently. The third section, Strategic Components Comparisons and Decisions, illustrates the breakdown and decision making process of various components, software suites, and designs used in our project. Finally, the last section, Possible Architectures and Related Diagrams, showcases the architecture of the MCU technology that power our project.

3.1 Existing Similar Projects and Products

As with any project in the design phase, it is usually a good idea to research similar projects and their features and implementations in order to compare with your own design and requirements. This made it a bit easier to think of features or ways to improve on other projects' feature set.

3.1.1 Hyduino

The Hyduino is an automated hydroponics system built off of an Arduino Mega 2560 board. Similar to our project, the system is designed to control lights, a touchscreen LCD for controlling aspects of the system, a few sensors, pumps, pH solution and a few other things. Through the touchscreen LCD, you can set different values for pH, temperature, humidity, light, and tank fill level. The system uses relays to control the lights, pumps, a solenoid valve to reduce the water level, and a couple of fans to control temperature and humidity. While this project is similar to ours, we believe that we improved on this design because it uses AC power from the mains instead of a renewable energy resource such as solar energy. This project gave us a good starting point to begin researching for parts and to see how other people approached a similar goal using a microcontroller board.

3.1.2 Niwa

The Niwa is a product that was funded on the website Kickstarter, a crowd funding website, in mid-2014. The Niwa comes in 3 models, a smaller Niwa Mini, a regular sized Niwa Standard, and the Niwa Premium which uses a higher quality aluminum finish. The Niwa Standard was available for $199 as a very early bird special, however the regular size model is priced regularly at $319 for pre-order. The Niwa's primary goal is to make it as simple and easy as possible to get into hydroponics growing inside your house. Figure 3.1 shows a picture of the compact design of the Niwa.

Figure 3.1: An image showcasing the various features of the Niwa
(Reprinted with Permission from Founders of Niwa)

The goal is that even a child could set up and use the Niwa system. The Niwa system's core is powered by a board called the Spark board. The Spark board is controlled by an ARM Cortex M3 processor and the Wi-Fi module is a Texas Instruments CC3000 network processor. The Niwa is designed to be a compact size self-contained hydroponics system that you can put in your living room. The Niwa controls a heater, a light array, irrigation, ventilation, and climate control all through an interface online or on a mobile app. You can tell the app what the plant is inside that you want to grow is and it will control the rest of the automation based on presets for that plant. The Niwa is similar to our project in the sense that it is connected to the internet and it's designed to be easy to use. However, the Niwa is a compact indoor AC powered system versus our bigger outdoor solar-powered system.

3.1.3 HydroHomeSoft™

The HydroHomeSoft DIY Automated Hydroponics Control System is a project available online that is well documented. The documentation guides you on how to build your own HydroHomeSoft system as well as an understanding of how the underlying systems work. The HydroHomeSoft runs on a system using Phidgets, but is also being ported to the Arduino for more universal draw. The HHS supports a different hydroponics method from our own called ebb and flow. Similar to our project, the HHS system controls nutrient density, temperature and humidity with fan control, lights, and pumps with timers. The system also supports graphs of reports for nutrient consumption, concentration, and pH. It also supports email alerts for abnormal activity. The project lays the foundation from the ground up on how to program and build the hardware for your own HHS. This project is another similar project in the sense that it uses sensors and timers to make smart decisions about when to activate what subsystem (pump, light, etc).

3.1.4 LeafAlone Hydroponics System

The LeafAlone Hydroponics system was a University of Central Florida senior design project from the 2014 Spring-Summer semester that was sponsored by Duke Energy. The system was designed to be powered from either AC power or a battery powered by solar energy. The project cost $1146.69 to design and implement with the majority of the cost in the solar panel, battery, and charge controller. The project used an Arduino compatible Atmega328p microprocessor. This project is a good one to consider reading about since it was designed with a senior design mindset. The documentation is complete and the project is similar to ours. It uses a microprocessor and a webserver to control and monitor sensor readings. The sensors are pH, electrical conductivity, temperature, light, and a water sensor. The project also included a camera, a nutrient dispenser, and a Wi-Fi component.

3.2 Relevant Technologies

Throughout the following subsections the relevant technologies will be discussed in order to evoke a better understanding of the type of technologies to be implemented within the hydroponics system.

3.2.1 Solar Power

The way solar cells work on a solar panel is import for this system because this technology is the source of power for the entirety of the system. The solar panel we are using is a 50W solar panel from the senior design lab, utilizing silicon solar cells. This solar panel was selected because it met the system's power consumption requirements and was also available for free from the senior design lab. Solar cells resemble semiconductor transistors such as MOSFETs (Metal-Oxide-Semiconductor Field-Effect Transistor) and BJTs (Bipolar Junction Transistor) in their internal configuration and technology utilized, as shown in Figure 3.2.

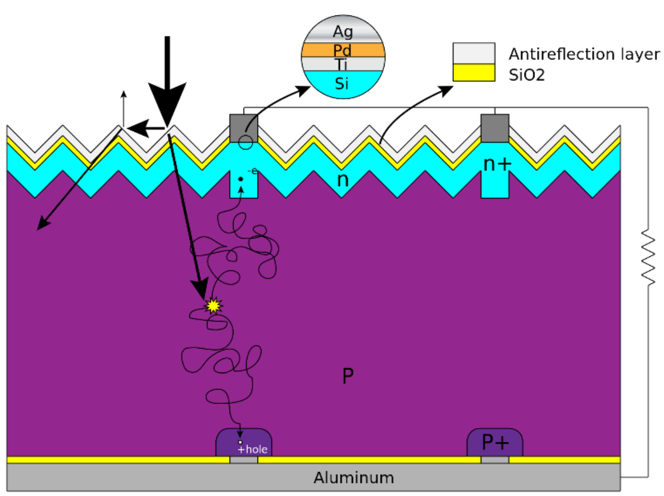


Figure 3.2: Solar Panel at Semiconductor Level (Permission Pending)

Solar cell operation consists of a silicon absorber layer, pn-junction, and a metal contact on the top and bottom of the cell. Light is emitted into the silicon absorber layer which is where the energy is absorbed and used to excite the charge carriers in the semiconductor pn-junction. The positively charged holes diffuse into the n-type silicon, while the negatively charged electrons diffuse into the p-type silicon. These oppositely charged energy particles are kept separated by the depletion region, in the same way transistors operate. The n-type silicon and p-type silicon regions are juxtaposed to the top and bottom metal contacts respectively. This separation of positive and negative chargers provides a voltage difference, and therefore current, resulting in power output of the solar cell.

Knowing how the solar panel operates allowed for the utilization of the solar cell as an energy source for the system. By providing the solar panel with optimal photons or sunlight, the solar panel outputs a linearly equivalent output of power. But, knowing how a solar cell works, the solar panel cannot store any meaningful amount of energy for a useful duration of time. The solar panel can also only output the amount of energy it is absorbing with respect to the solar panel's surface area. The system has relatively high power consumptive subsystems, which the solar panel may not be able to power in real time. Also, if a cloud rich day or night time is present, the solar panels are only able to output minuet or zero power. Therefore a storage system was used to save the energy generated by the solar panel for use within the system.

3.2.2 Battery Bank

A storage system for the solar panel power output was needed to retain and distribute power to the rest of the system. Currently the most widely used storage for solar panel energy is a battery or battery bank. There are other methods of storage, that can be more expensive or exotic but, for the practical purposes of the hydroponics system a battery bank was used. There are multiple battery chemistries on the market today but, almost all of them operate in the same way. There are at least two solutions or chemicals within the battery, separated by some physical barrier. There are also two metal nodes placed into the battery. The electrons stored in the chemicals produce a potential difference across the two metal terminals.

3.2.3 Battery Charging

Since there are a battery bank as the source for all power in the hydroponics system, a battery charging subsystem was needed to control the charging of the battery bank by the solar panel. A charging system has two main functions, first is to control and regulate charging of the battery bank, second is to control and regulate the flow of charge such that the power is not discharged in the wrong direction. The solar panels are connected to the battery charging subsystem, and the battery charging subsystem is connected to the battery bank, thus the battery bank is connected to the remaining hydroponics system and distributes power to the entire system. The battery charging subsystem regulates the electrical current rate to the battery bank and also regulates the voltage level of the charging to the battery bank. If the battery bank is fully charged, the battery charging subsystem is able to determine that battery bank is full and stop charging the battery bank, or at least periodically top off the battery bank with power. This is a core feature of the battery charging subsystem. If the battery bank were to over charge, depending on the chemistry of the batteries, the battery bank could explode, melt, discharge into the system, or become defective. All of these outcomes are undesirable to a stable sustainable system. The battery charging subsystem must also be able to prevent against discharging into the solar panel subsystem. While regulating the voltage level of the battery bank being charged, the charging system also has circuity that performs as a large diode to prevent the discharging of current into the solar panel but allows charging of the battery bank.

3.2.4 Battery Level Monitoring

A battery level monitoring subsystem is implemented in the hydroponics system. By monitoring the battery bank capacity level, the hydroponics system is able to report to the LCD screen. This is helpful for the user to record the amount of power discharged into the system, the power consumption of the hydroponics system is known, as well as the power output of the solar panels, so the system is compensated if needed due to lack of sunlight. This will be accomplished by the use of voltage sensors on both the output of the solar panels and the output of the battery bank. These voltage sensors deliver data to the MCU to be recorded and processed. The battery bank characteristic discharge curve was used to correlate the amount of charge remaining in the battery bank and calculated within the MCU processing.

3.2.5 RF Network

Since our system is intended for outdoor use, our project requires a flexible way of communicating sensor data to our local area network. Wireless Radio Frequency (RF) devices are most suitable and are tailored for a variety of power and data rate specifications, from low-energy communications (RF point-to-point modules) to high data-throughput devices (Wifi).

3.2.5.1 Baseline requirements: RF Networks

The project has baseline requirements for the RF Module, objectives that must be met independent of decided module. These requirements are the following:

1. Take sensory data and transmit to the Web Server
2. Have a range of at least 30ft
3. Meet FCC codes and regulations
4. Use no more than 500mW when transmitting

3.2.5.2 Important RF Network parameters

The RF network is affected by a variety of factors. Being wireless channels, they are exposed to a wide spectrum of background noise, interference from other channels, signal impedances from walls and other structures, to name a few. The most important factors when considering the environment the RF network were communicating and intended information throughput are carrier frequency, power usage, and data rate of the channel.

3.2.5.2.1 Carrier Frequency

The carrier frequency used for RF modules is typically between 100 MHz and 5 GHz. A property of the frequency used is the inherent range for transmission, lower frequencies tend to propagate further the higher frequency signals for wireless RF channels through air. Therefore, for open ranges lower frequencies are preferred to higher frequencies because they scatter less. Higher frequency carriers are preferred for enclosed spaces because they scatter more than lower frequency. With this information, it may be inferred that lower frequency modules require less power transmit the same distance than higher frequency carrier signals. To transmit a signal, oscillating current is applied to an antenna which induces carrier frequency. The carrier is modulated by other circuitry and the resulting signal is transmitted. The receiver takes this modulated signal and demodulates it to retrieve the encoded data.

3.2.5.2.2 Power Usage

Power usage is dependent on the range intended for transmission, the carrier frequency used, modulation scheme, and nominal state current usage. RF point-to-point modules are generally the most efficient, with a better range per electrical watt used. Bluetooth is comparable in power dissipation to RF point-to-point modules but with a more limited range. Wifi consumes power at least twice as great as RF point-to-point modules and with a more limited range.

3.2.5.2.3 Data rate

Data rate is most strongly dependent on the modulation scheme used, the greater the modulation frequency the more information transmitted in a period of time. RF point-to-point modules have the lowest performing data rate, followed by Bluetooth, and Wi-fi having the highest throughput. Signal to noise ratio (SNR) also affects the encoding used, with greater power usage the more information can be stored in an m-ary encoding, which effects the resulting data rate.

3.2.5.3 RF Module Constraints

Each RF module is limited in the amount of hardware that can be fit in a package, which affects the module's voltages tolerances, power requirements, pin multiplexing, and other factors. In addition, variables aside from the hardware weighed in our decisions: cost, communication protocol, and manufacturability. Table 3.1, RF Module Comparisons Table, was compiled to compare popular RF module choices for similar control system projects based on the important RF network parameters previously discussed:

Company	Nordic	Nordic	TI	Roving Networks
Type	Point-to-Point	Point-to-Point	Point-to-Point	Bluetooth
Part#	nRF905	nRF24L01	CC1101	RN-41
TX Power (Max)	+10dBm/30mA	0dBm/11.3mA	+12dBm/35mA	+15dBm/100mA
Frequencies	433/868/915 MHz	2.4GHz	315-915 MHz	2.4GHz
Data Rate	50kbps	2Mbps	500kbps	1.5Mbps
Total Power Dissipation	200 mW	60mW	250mW	360mW
Power Voltage	+1.9-3.6v	+1.9-3.6v	+1.8-3.6v	+3.0-3.6v
MCU Interface	SPI	SPI	SPI	UART
Package	32-pin QFN	20-pin QFN	20-pin QFN	35-pin Module
Cost	$4.24 | 1U	$1.78 | 1U	$1.92 | 1kU	$25.95 | 1U
Advantages	Greater range due to frequency, low power consumption	Excellent community support, cheap, low-power consumption	Greater flexibility, high data rate	High RF power
Disadvantages	No large community support	Discontinued chip, not relevant for future products	Consumes most power out of Pt-to-Pt modules	High cost

Company	TinySine	TI	Espressif
Type	BLE	Wifi	Wifi
Part#	TS BLE module	CC3000	ESP8266
TX Power (Max)	+7dBm/50mA	+14dBm/275mA	+19.5dBm/215mA
Frequencies	2.4GHz	802.11bg standard	802.11bgn standard
Datarate	48kbps	11 Mbps	11Mbps
Total Power_Diss	165mW	1000mW	700mW
Power Voltage	+3.3v	+1.8-3.6v	+3.3v
MCU Interface	UART	SPI	SPI,UART
Package	6-pin Module	46-pin LGA	32-pin QFN
Cost	$9.95 | 1U	$13.80 | 1kU	$6.95 | 1U
Advantages	Simple to interface, user manual is very explict	High-datarate, standardized communication	Great community support, cheap
Disadvantages	Half the typical RF Power	Power-hungry	Official documentation scarce

Table 3.1 - RF Module Comparisons Table

3.2.6 Automation

The automation subsystem is responsible for controlling the various devices which maintain the plants in the hydroponic system, such as the light array, tiller, water pump, and water flush system. This section is a summation of the research conducted in designing these sub-systems.

3.2.6.1 Light Array

The hydroponics system needed a light source that is consistent, one possible source was the sun. The sun is ideal because it has a high light intensity and consumes no electrical power on behalf of the hydroponic system. However, its duration changes with the seasons and making it unreliable for implementation. An alternative was use a light source which is controlled by the MCU. This light source could be an incandescent bulb, a fluorescent tube, or an LED array. Each of these has their advantages. For example, an incandescent bulb is the cheapest to integrate and many sizes and wattages exist but are inefficient and require frequent replacement. A fluorescent tube has a wide spectral range and is slightly more efficient than the incandescent bulb but it requires special equipment and is thus costly to integrate. An LED bulb which mounts in a standard incandescent bulb connector would be ideal because it is the most efficient and thus be cheaper to integrate, an example shown in Figure 3.3.

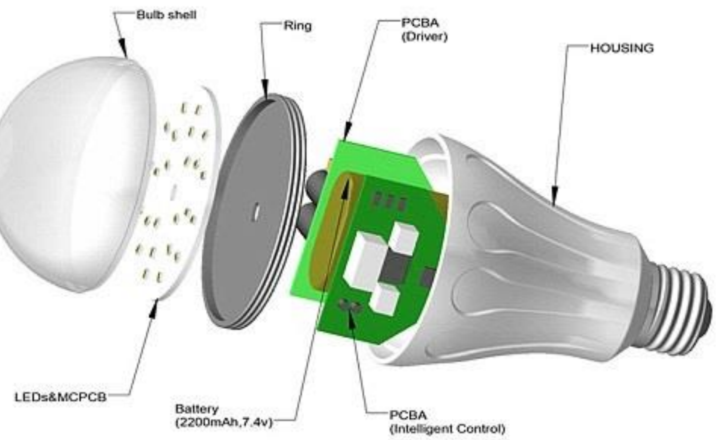


Figure 3.3: Example of LED bulb

3.2.6.2 Tiller

The amount of nutrient in the water can be automated via a belt-driven delivery or tilling column. A belt-drive would deposit nutrient solute by translating the solute from a container of solute to the water reservoir. An alternative to this approach is the use of a tilling column, which feeds nutrient into the reservoir by using a mechanism similar to a screw’s incline plane, the ramp slowly feeding nutrient into the reservoir’s water. Some initial estimations and sketches of implementing each type of delivery mechanism showed that the tilling method would be more reliable, at a hardware cost of occupying more space than the belt-driven delivery method.

Interfacing with the MCU, the tilling column would be affixed to a stepper motor, with its rotations translating the solute into the water reservoir. In collaborations on how to fabricate the tilling column, our group considered using a 3-D printer to prototype a unique screw or order an already manufactured tilling column. 3D printing a column would allow us to design the exact dimensions we'd want but could possibly be more fragile and costly. Therefore it was decided to use an auger bit as the tilling column, which could be obtained from a local hardware store at a reasonable price, and made of steel. A fair tradeoff for a part which doesn't meet the exact dimensions we'd like. We may choose to mount this system within the reservoir to avoid contamination that would result from an open reservoir or we may couple the two systems such that there is no air for possible mold or fungus to enter the system.

3.2.6.3 Water Pump

Being an NFT-based hydroponic system, it was critical to the design that we have a water pump with adequate water output and head pressure. In addition, being that we're designing the system to offset its power consumption via a solar charging system, it was ideal to have this water pump DC powered from the solar panels and modulated via an intelligent power switch. Some base requirements were that it have at least a 500 gallons per hour water throughput and 4 foot head pressure. The pump pressurizes the water to the top of the system where it flows through the channel back to the reservoir, where it repeats the cycle. A bilge pump for boats was the ideal solution as they are SLA battery-powered, submersible, and have adequate water throughput.

3.2.6.4 Water Flush System

Should the water be contaminated or its pH become out of balance, it was imperative that the system may emergency eject the water from the reservoir. This may be done by modulating the water pump’s output channel via a rotating coupling or having a separate water pump whose sole purpose is ejection. Modulating the output channel had a simpler hardware cost, in that it is a coupling actuated by a servo diverting the water to either the hydroponic channels or out of the system. This turned out not too difficult to fabricate and was simple to maintain. The second option we considered was integrating a second water pump whose output is solely out of the hydroponic channel. It would take more space in the reservoir but would be easier to maintain and possibly cost less.

3.2.6.5 Microcontroller Integration for Control Data

Each of these automation subsystems are controlled via the MCU through a control data protocol. From the MCU, control data is fed to a power switching board which modulates the state of these automated subsystems, alternating their states between active and hibernating. When active, the modulation element activates the system, in the form of a switch completing the circuit between the power supply and the automation subsystem unit (Pump, light array, etc.). Thus the control data is simple, one bit for on/off states or multiple bits for quantized states, such as a servo’s speed. The control data is sent over UART because these states are asynchronous, changing when a sensor value prompts the program to modulate the state of one of these automation subsystems.

3.2.7 Stepper Motors

Our nutrient feed mechanism requires precisely controlled movement, for which we choose to use a stepper motor. These devices are regulated by pulses through coils, varying the angle of the motor shaft depending on the frequency, number of coil phases, and several other factors. The result is a very precise motor which has continuous rotation and tremendous degree of control over the angle of the shaft. Modern stepper motors come with included driver circuits, modulating the pulse train within the coils when given a signal. Thus the designer need only worry about the power rails and the one signal wire, specified modulation typically being pulse width modulation because it requires only one wire and takes minimal resources on the part of the controller sending the signal.

Stepper motors have been used extensively for applications in machine automation, which is why we considered it above constant DC motors and servos. Having both the qualities of continuous rotation and angular control makes these motors the most attractive to work with and can be used for either a belt-driven or tilling feed systems, enabling our group even more design choices. Disadvantages that must be kept in mind are they're generally bulkier and slower than constant DC motors, a fair cost when considering our main priority is precision.

3.2.8 Sensors

Sensors are generally transducers which monitor and report information about an environment in an electrically useful output, either analog or digital voltage or current output which correlates to the environmental information. Though research the different types of sensor technologies available for each particular application required within the hydroponics. The different types of methods for transducing the environmental sensor data will be mentioned. As well as the different types of output of the sensor data, whether it be analog voltage, analog current, digital I^2C, SPI, UART, etc.

3.2.8.1 Light Sensor

A hydroponics system is used to grow plant life, and the majority of all plant life is dependent on light for photosynthesis which is their way of producing growth energy. Light being an essential part of plant growth, this automated hydroponics system will require light exposure monitoring. For that, a sensor determines the amount of energy a plant is exposed to by sun light or the light array subsystem required.

The light sensor selected for this system is the Intersil ISL29023, as shown in Figure 3.4, ambient and infrared light sensor. "The ISL29023 is an integrated ambient and infrared light to digital converter with I^2C (SMBus Compatible) Interface. Its advanced self-calibrated photodiode array emulates human eye response with excellent IR rejection." [20]

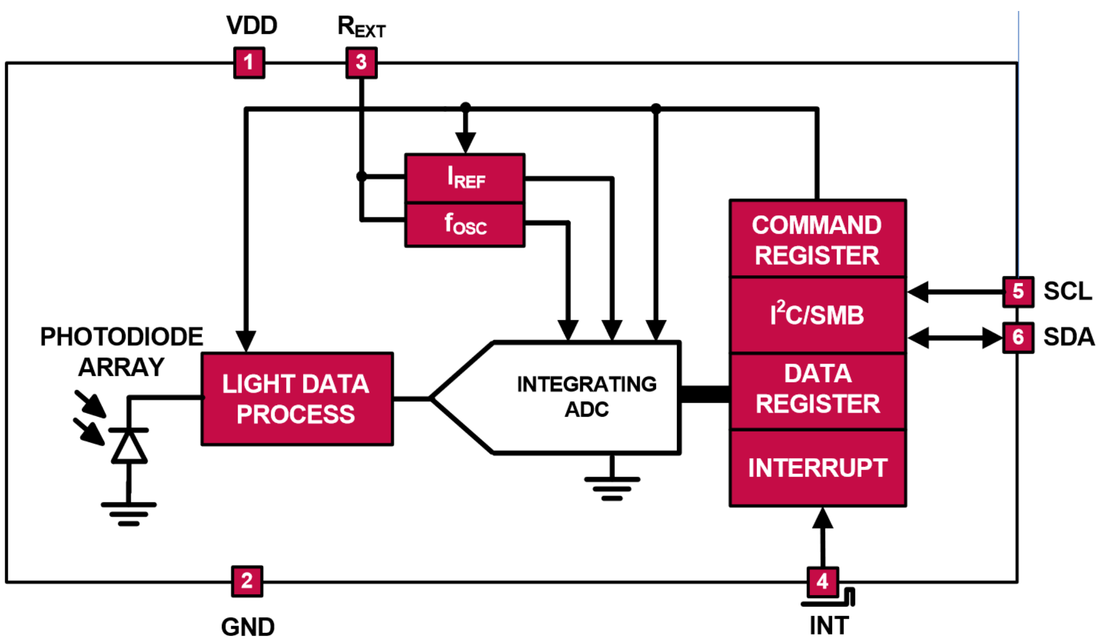


Figure 3.4: Intersil ISL 29023 Ambient and Infrared Light Sensor (Permission Pending)

In order to properly use the Intersil ISL29023 sensor, it is first necessary to understand how a photodiode array operates. A photodiode array is a collection of p-n junction diodes, just like a normal diode. The difference is that a photodiode maintains reverse bias, producing an electric field across the depletion zone. The photodiode is exposed to light, at which point the electromagnetic radiation from the light produces positive holes and negative electrons to move into the n-type and p-type respectively. This produces a current emitted by the diode, by analyzing this current, the intensity of light can be obtained. Different types of semiconductor material can be doped to monitor different frequency spectrum bands. "For example, silicon-based photodiodes are sensitive to radiation between 190-1100 nm, making them useful for UV, visible and limited near-IR spectroscopies, while lead sulfide photodiodes are sensitive in the range of 1000-3500nm, enabling detection in both near-IR and short-wave-IR." [21]

3.2.8.2 pH Sensor

For any agricultural system, the pH level has an important role in the growth and health of the plant within the agricultural system. The parameter pH measures the concentration of hydrogen ions within a liquid solution. This pH level is measured on a scale from 1 to 14, a pH measurement of 7 is said to be neutral, while a pH measurement greater than 7 is called basic, and a pH measurement lower than 7 is called acidic. For a hydroponics system, the pH levels of the water in which the plant grows acts as the environment in which the pH levels should be measured. To ensure proper growth and health of the plant the pH levels of the water should be maintained within a certain boundary area, which is dependent upon the plant type being grown.

In order to properly utilize a pH sensor, to monitor the pH levels within the hydroponics system, an understanding of how a pH sensor works is required. The pH of a liquid solution is also a function of temperature, therefore the temperature of the hydroponics system's water is taken into account using a water temperature sensor. Since the pH measure the concentration of hydrogen ions within a liquid solution, the higher the pH level, the more hydrogen ions within the liquid solution. The more hydrogen ions in the solution, the more positively charged the solution because hydrogen ions are positively charged. A pH sensor takes advantage of this electrical correlation of pH level and hydrogen ions or positive charge.

In the hydroponics system, there is water solution in which the plants' roots penetrate for all of their nutrients. This is the liquid solution in which the pH sensor will be placed to measure the levels of pH in the system. "A pH measurement loop is made up of three components, the pH sensor, which includes a measuring electrode, a reference electrode, and a temperature sensor; a preamplifier; and an analyzer or transmitter." [senorland.com] For this hydroponics system, the pH sensor consist of the above mentioned measuring electrode and reference electrode. By taking the potential different between the two probes, a voltage can be measured and this voltage will directly correlate to the pH level of the water subsystem.

3.2.8.3 Water Level Sensor

The hydroponics system required a subsystem to measure and monitor water levels within the system. If the water in the system were to increase or decrease due to rain or evaporation, then the change in water level is monitored to ensure that water is flushed out of the system or pumped into the system. To achieve this a component called liquid eTape is utilized in the water system.

The eTape Liquid Level Sensor is a sensor with a resistive output that varies with the level of the fluid. Liquid eTape is a resistive sensor, meaning that the greater the exposure to water or another liquid the eTape, the lower the resistance of the eTape. By interfacing this analog sensor into the hydroponic system's microcontroller, both the change in and the current water level can be calculated through software based on the resistive output given by the liquid eTape.

3.2.8.4 Temperature Sensor

The hydroponics system requires the monitoring of the ambient temperature for the agriculture to properly grow. By measuring the temperature of the surrounding environment, changes in the operation of the hydroponics system can be enacted. The reference diagram of the temperature sensor selected is displayed in Figure 3.5.

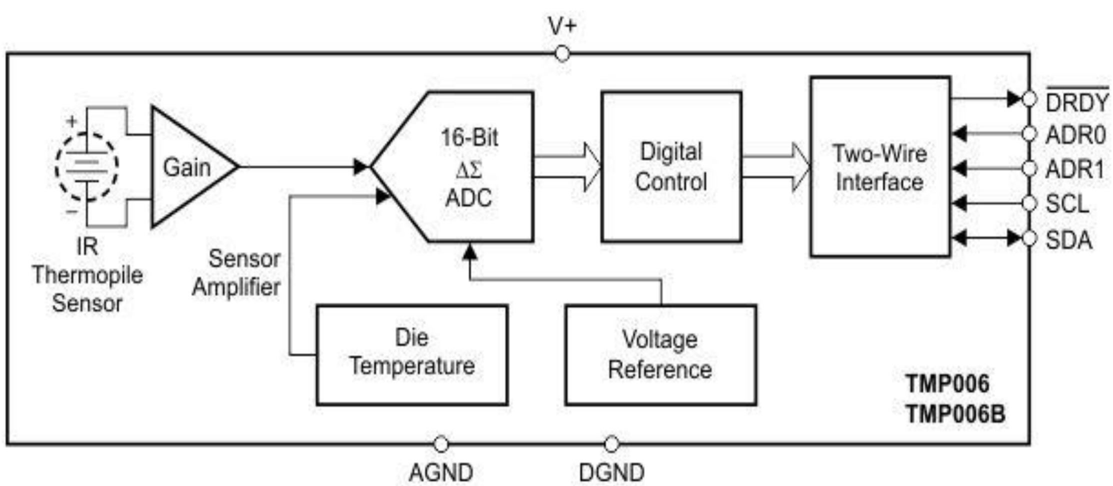


Figure 3.5: Thermopile Temperature Sensor

In order to properly utilize a temperature sensor within the hydroponics system, a basic understanding of how the temperature sensor operates is required. "Thermopile sensors use infrared (IR) radiation versus conduction for heat transfer, which provides unique solutions that allow for new levels of performance and reliability in many constrained applications" [22] By measuring the infrared radiation of the surrounds and using the die temperature as a reference, the temperature of the surroundings can be calculated and used for the hydroponics system.

3.2.8.5 Water Temperature Sensor

The hydroponics system requires the monitoring and reporting of the temperature levels as they change with time within the water of the hydroponics subsystem.

This is increasingly important for the temperature dependent pH sensor. For the pH subsystem to accurately calculate the dynamic pH level, a temperature reading at the same time must be performed within the water. The water temperature readings are also important for monitoring the health of the plants within the system.

3.2.8.6 Dissolved Oxygen Sensor

The dissolved oxygen sensor provides the hydroponics system with feedback of the quality and quantity of oxygen molecules in the water in which the roots of the plants are placed. Dissolved oxygen is important to the health and growth of the plant life, and is directly correlated with the plant production. The operational components of the dissolved oxygen sensor is shown in the following Figure 3.6.

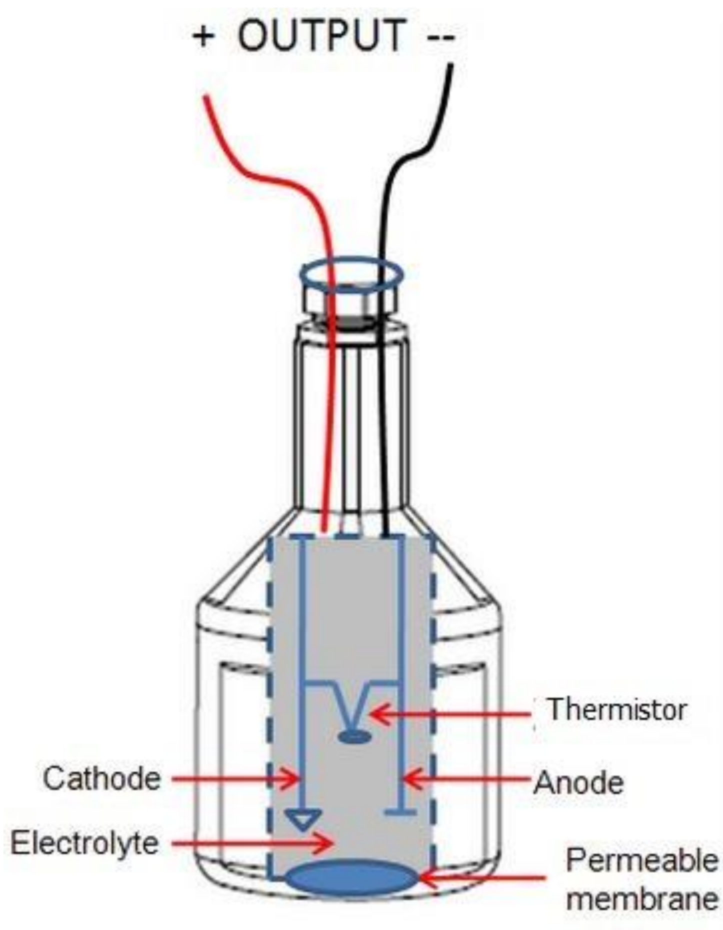


Figure 3.6: Dissolved Oxygen Sensor Probe

To properly utilize the dissolved oxygen sensor readings, an understanding of the operation and process of measuring the level of dissolved oxygen with the sensor is required. The dissolved oxygen sensor reads the voltage difference of the anode and the cathode within the water solution. By measuring this voltage difference and comparing it with the voltage difference from an air sensor probe, the amount of oxygen can be calculated. The dissolved oxygen sensor has to also take into account the temperature, pressure, and salinity of the solution. By incorporating these measurements a more precise reading of the level of dissolved oxygen can be determined.

3.2.9 Voltage Regulation

The solar panels are used to charge a battery pack that is a higher voltage than what is used for the MCU and sensors. This means that a DC-DC conversion circuit must be created for our board in order to properly power all of the components. There are multiple ways to convert DC voltages to other DC voltages. First, we have to establish what kind of voltage regulator we need to use. Since the battery voltage is most likely going to be higher than the voltage we need, a “buck” type voltage conversion should be used, which means that the output voltage is lower than the input voltage. Second, since our project uses renewable energy, we want to have a high amount of efficiency for our voltage conversion circuit. The following subsections compare multiple types of voltage conversion and analyze the advantages and disadvantages of each in context with our project goals.

3.2.9.1 Linear Voltage Regulators

The standard linear voltage regulator is one of the simplest methods of voltage conversion. Since a linear voltage regulator can only reduce voltage, and our project needs to reduce voltage from the battery pack, we considered using one as a possibility for our project. Linear voltage regulators are pretty common in the industry, so there are many reference designs available online to adapt to our project. Using a linear voltage regulator is also a cheap solution because there are not many components and the components are cheap. Typically a few transistors, resistors, and op-amps for a feedback loop are all that’s needed to implement the circuit. Figure 3.7 shows the basic design of a linear voltage regulator circuit.

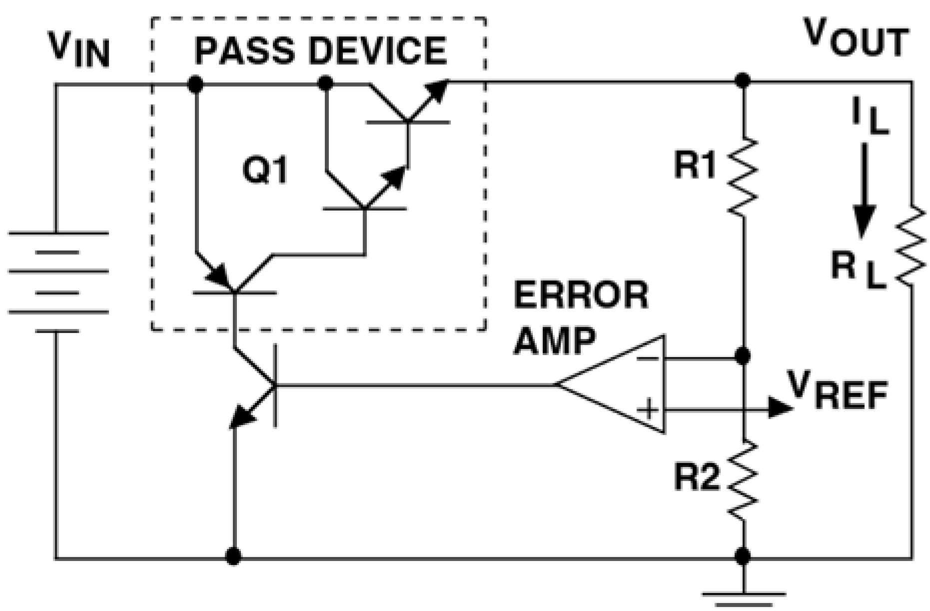


Figure 3.7: An example of a linear voltage regulator circuit
(Reprinted with Permission from Texas Instruments)

Although the circuit is cheap and relatively easy to implement, the major drawback of the linear voltage regulator is the low efficiency. Because of the way the circuit functions, there is a dropout voltage that is simply lost because of the transistor turn on voltages, typically in the range of 1.5 ~ 2V. The large voltage conversion also results in a large power waste. The wasted power is given by the following equation:

$$P_{waste} = I_{load} * (V_{in} - V_{out})$$

If, for example, the battery is 12V and the MCU and sensors require 3.3V at 1 A, the wasted power would be about 1 * (12 – 3.3) = 8.7 W. This not only doesn't give the efficiency that's good for this type of project, but also there's an added problem of heat dissipation because the wasted power directly translates into heat.

Advantages

- Simple to implement
- Cheap solution
- Low noise
- Small size

Disadvantages

- Low efficiency
- Heat dissipation may require a heat sink

3.2.9.2 Low Dropout Voltage Regulators

The primary difference between a low dropout voltage linear voltage regulator and a regular linear voltage regulator is that the dropout voltage is much lower. Typically the dropout voltage is somewhere in the 300mV or so area instead of the 1.5 – 2V area. While this would improve efficiency if the input voltage was close to the output voltage, it does not improve efficiency in our project.

Advantages

- Simple to implement
- Cheap solution
- Low noise
- Small size
- Good efficiency if input voltage is close to output voltage

Disadvantages

- Low efficiency if input voltage is not close to output voltage
- Heat dissipation if power wasted is too high

3.2.9.3 Switching Voltage Regulators

A switching voltage regulator is a regulator that uses a more complex design to regulate the output voltage. Because of its more complex design, it is capable of more than just a buck type conversion. It can also boost and invert voltages. This kind of regulator uses pulse width modulation (PWM) to output a voltage based on the pulse amplitude and duty cycle. In order to regulate, the duty cycle can be adjusted to raise or lower the voltage. Figure 3.8 shows the relationship between PWM and voltage in a switching voltage regulator circuit.

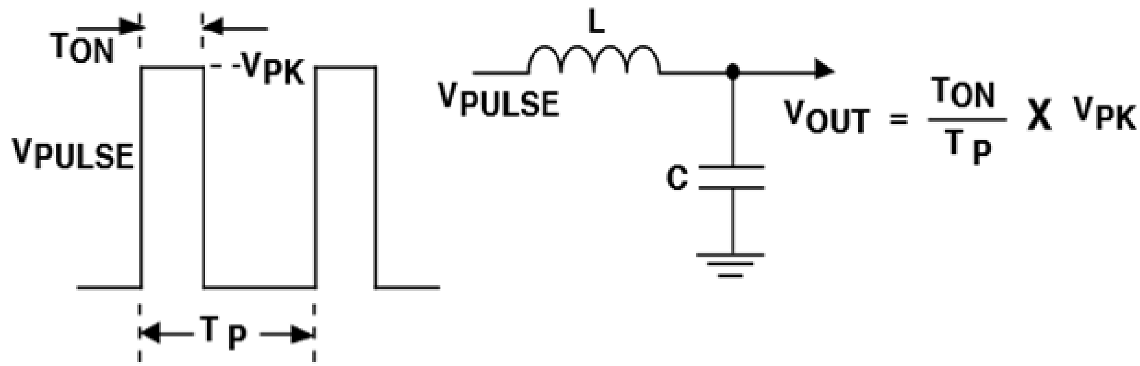


Figure 3.8: Pulse width modulation in voltage regulation
(Reprinted with Permission from Texas Instruments)

Since the pulses can be controlled, the input voltage can be cutoff at times, resulting in a higher efficiency than the simple linear voltage regulator. The drawback to higher efficiency is a harder to implement design and also the inclusion of an inductor. However, the high efficiency of the switching voltage regulator is attractive in order to take advantage of the battery's power as much as possible. The main advantages of this style of DC/DC voltage conversion are the high efficiency, the possibility of not having a heat sink due to the high efficiency and low wasted power of the switching circuit, and the ability to output multiple voltages. There are multiple disadvantages to this style of voltage conversion, however. The circuit is more complex to design and implement compared to the linear equivalent. Due to the pulse width modulation, it is also a possibility to add noise to the output voltage signal. This reference design also uses an inductor, which are usually quite large. Since the circuit design is more complex, it is also more expensive to implement, requiring more parts than the simple linear voltage converter.

3.2.10 Linux

Linux is an open source operating system that runs on millions of devices. The first Linux kernel was released by Linux Torvalds on October 5, 1991. Originally for an Intel x86 based PC, the strength of Linux is its portability to many different types of hardware configurations. Because of the popularity of Linux as an open source operating system, there is plenty of documentation and distributions of Linux. A distribution of Linux is a term that describes different configurations of the Linux kernel with different software and drivers. A distribution of Linux can be written by anyone, including foundations, companies, and hobbyists.

Linux can come in more user-friendly versions, such as Ubuntu and Mint Linux, with long term support for these versions similar to a paid operating system such as Microsoft Windows. Linux can also come in more minimalist distributions which can be completely personalized by the user to get a version of Linux that runs exactly the way they want it to such as Arch Linux. The advantage of these types of versions is that services that are unnecessary are not running in the background, so resources aren't being taken up by something that's running that isn't actively being used. The disadvantage of this type of Linux is that the install process is not automated or easy, with everything having to be done at the command line (partitioning your storage, installing a desktop interface, configuring drivers and settings). Linux, and open source software in general, has a large community that we can take advantage of for help on our project. Due to the ability to use Linux on low power applications, and the fact that an operating system is needed to run our web server, Linux is a technology that we will include in the Energy Sustainable Hydroponics with Automated Reporting and Monitoring system.

3.2.11 Web Servers

In order for the Energy Sustainable Hydroponics with Automated Reporting and Monitoring system to communicate with the outside world, our MCU will communicate to a single board computer device, such as a Beaglebone Black. This device will broadcast relevant data via a web server to be served to a web browser on the local network. A web server is a piece of software that enables a device to transmit pages from the local file system to a client accessing it via Hypertext Transfer Protocol (HTTP). HTTP is the same protocol that most websites on the internet use today and is a very well-known and documented protocol. Typically a web server serves an HTML (HyperText Markup Language) page, but it is also possible depending on the web server to serve pages that use additional scripts processed by the server such as PHP (Hypertext Preprocessor) or ASP (Active Server Pages). Additional technologies can be added to a web server, such as the ability to add a scripting language like JavaScript for dynamic web pages. These features are important as it allows you to build more robust websites with interactive content.

3.2.12 Single Board Computers

Single board computers are usually small computers that run on a single circuit board. They have all of the components of a real computer, a CPU, a GPU, random access memory, storage in the form of built-in flash storage and/or external memory card storage, USB ports, video, sound, and network ports. The advantage of these computers is that they are cheap, small, and take a relatively small amount of power to run continuously compared to a traditional laptop or desktop. Single board computers were made popular recently by the release of the Raspberry Pi in February 2012 for $25. The goal was to get cheap computers in the hands of everyone who wanted to learn how to develop software. The computers can also interface easily with outside circuits, such as breadboards with LEDs, sensor, and similar items through the use of GPIO (General Purpose Input/Output) pins. In order to run the web server for our Energy Sustainable Hydroponics with Automated Reporting and Monitoring System, a single board computer is a good choice to get access to the sensor data and send control data to the MCU. The single board computer can not only run the web server, but it can also interface a Wi-Fi adapter to connect to a Wi-Fi router in order to easily broadcast and receive the data from a user who wishes to see their hydroponic system status.

3.3 Strategic Components Comparisons and Decisions

Our project decisions need to encompass flexibility and efficiency in order to meet our requirements and specifications, in effect simplifying our design. We strategically chose components that would afford us the greatest number of options while considering our group's resources of development time, monetary cost, and complexity of the final design. Our careful deliberation, research and decisions are what follow in the continuing sections.

3.3.1 Solar Panel

There is a wide selection of solar panel technologies, voltages, and power outputs available on the market. Different types of solar panel technologies include monocrystalline silicon, polycrystalline silicon, building integrated photovoltaics, and solar thermal panels. All of these technologies have their benefits and trade-offs, shown in Table 3.2.

	Monocrystalline Silicon	**Polycrystalline Silicon**	**Building Integrated Photovoltaics**	**Solar Thermal Panels**
Pros	Most Energy Efficient	Inexpensive	Aesthetically Pleasing	Heats Water
Cons	Expensive	Low Energy Efficiency	Expensive	Non-Electric Power Producing

Table 3.2: Solar Panel Technology Trade-Offs

After taking the trade-offs of the solar panel technologies into consideration, solar thermal panels and building integrated panels are not an option for the hydroponics system. Which leaves monocrystalline silicon and polycrystalline silicon panels. Due to the University of Central Florida having different capacities of monocrystalline silicon solar panels available for use within the hydroponics system. The cost of the monocrystalline silicon solar panels are no longer a factor in the decision for selection of solar panel technology. The free monocrystalline silicon solar panel is used as the power generator for the hydroponics system.

3.3.2 Battery Bank

The selection of a battery bank for a solar powered system could possibly have been the most vital decision made within the entire system. When selecting a battery bank for the solar charging configuration of the hydroponics system, there are an innumerous amount of considerations to take into account. A small portion of those considerations are cost, voltage, capacity, easy of charging, weight, charge density, life cycle, environmental effects, operational temperature range and discharge characteristics. There are many other aspects to consider about battery selection, but these attributes apply most heavily on the battery selection within this application of a hydroponics system.

The two of the most popular battery chemistries in use today are sealed lead acid (SLA) and lithium iron blends, such as lithium iron phosphate (LiFePO4). For the application of a hydroponics system, in which a solar panel power source is used, these two battery chemistries were considered. After a more expansive list of chemistries were researched and considered, the possible candidates for selection within this application.

The respective benefits of each chemistry is as follows:

Sealed Lead Acid

- 20% the cost of LiFePO4
- Low charging and discharging complexity
- Greater operation temperature range -20C - 60C
- No battery memory
- Minimal self-discharge

Lithium Iron Phosphate

- 75% weight of SLA
- Higher charge density
- 65% of volume of SLA
- 7 times the life cycle of SLA
- 2000+ charge/discharge cycles
- No harmful elements
- Minimal voltage sag

3.3.3 Battery Charger

The battery charger subsystem is one of the most important subsystems in the hydroponics system. There are multiple options for implementations of the solar panel to battery bank battery charging subsystem. While there are numerous options on the market for prepackaged solutions for a battery charging subsystem, this hydroponics system project is aimed at educating it's engineers with the knowledge to design and construct practical solutions. Therefore a solar charge controller with 100W input at 12V and 7A current rating was donated by the UCF Senior Design Lab was used in the system.

3.3.4 MCU

The microcontroller unit (MCU) is a critical decision as it not only reads sensor values about the hydroponic system, but also communicates that data to the web server and controls the automation subsystems. This section outlines our requirements, design options, comparison of common MCUs and our choice of which MCU would be the most appropriate for our project.

3.3.4.1 Baseline requirements: Microcontroller Unit (MCU)

The project has baseline requirements for the MCU, objectives that must be met independent of decided processor. These requirements are the following:

1. Take 5 sensory scalars about the system (Light intensity, CO2 levels, Temperature, Water turbidity, and Water pH)

2. Hold and publish these results to a local LCD display and over RF to a web server.

3.3.4.2 Design options

With these set requisites, our group set out to design a system with an emphasis on modularity, should the project have further constraints unforeseen in the research phase. Our group's design goals and requirements are the factors setting the bounds for our project rather than physical limitations, our project focusing on control systems and monitoring with performance rated by duration of operation without errors.

Thus our design options when designing the MCU system were such:

1. Communication Protocols with Sensors
   a. I2C
   b. SPI
   c. UART
   d. Group-Designed Protocol
2. RF Implementation
   a. Point-to-Point Modules
   b. Wi-Fi Dongles
   c. Bluetooth Modules
   d. GSM Modules
3. Local Information Display
   a. LED Panel ("Check Engine Light")
   b. LCD Display
   c. Sound Notifications

3.3.4.3 Important MCU Parameters

Memory: Each processor has different sizes and variants of memory available. The popular types used in modern MCUs are SRAM and Flash memories. Flash type storage contains constants and instructions programmed by a user, this memory is safe from loss of power but is a relatively slower memory. SRAM stores data and instructions frequently used by the processor, this information loaded from Flash memory when the device receives power. SRAM is the fastest memory type, which is why it's designated for storing transitory program data as information is being processed. However the consequence of SRAM is the information is lost when power is removed from the device, lost to thermal equilibrium. Thus when deciding on a processor, larger storage for each memory type is better but limited based on footprint, cost, and current fabrication technology.

3.3.4.3.1 Timing

Clock frequency is a general metric that can be used to estimate the potential performance of a processor. A higher clock rate generally equates to a "faster" processor but is also dependent on the number of stages and buffers within the MCU. Clock rate is also linked to the power consumption of the device because the number of voltage transitions in a given period of time is proportional to the power usage. Thus a faster clock consumes more power than a device with a slower clock with equal voltage magnitude.

Timers are yet another important consideration for MCUs. They are modules that count clock cycles, equating to some fraction of a second, and execute events and interrupts based on the loaded program. A timer's accuracy is dependent on the clock type used by the processor and precision is marked by the number of bits used. For example, a 32-bit processor more precisely describes, or can count more cycles, than a 16-bit timer. Frequently is the case that a 32-bit timer can also be split into two 16-bit timers.

3.3.4.3.2 Communication Protocols

There are three major communication protocols supported: I^2C, SPI, and UART. Each are used for data transmission across the same board and would be the most reasonable to implement for our project. These methods differ in their flexibility, ease of configuration, number of pins required, and baud-rate.

SPI, a reference diagram of SPI in figure 3.9, uses a minimum of three connections: MOSI (Master out, Slave in), MISO (Master in, Slave out), and a clock signal. Classically, SPI uses four connections; the fourth being a Chip Select line (CS) differentiating the master from the slaves. The advantages of SPI include separate data buses for MOSI and MISO, hardware defined chip relations, and full-duplex communication. The disadvantage is increased pin count and that master-slave relations typically can't be changed.

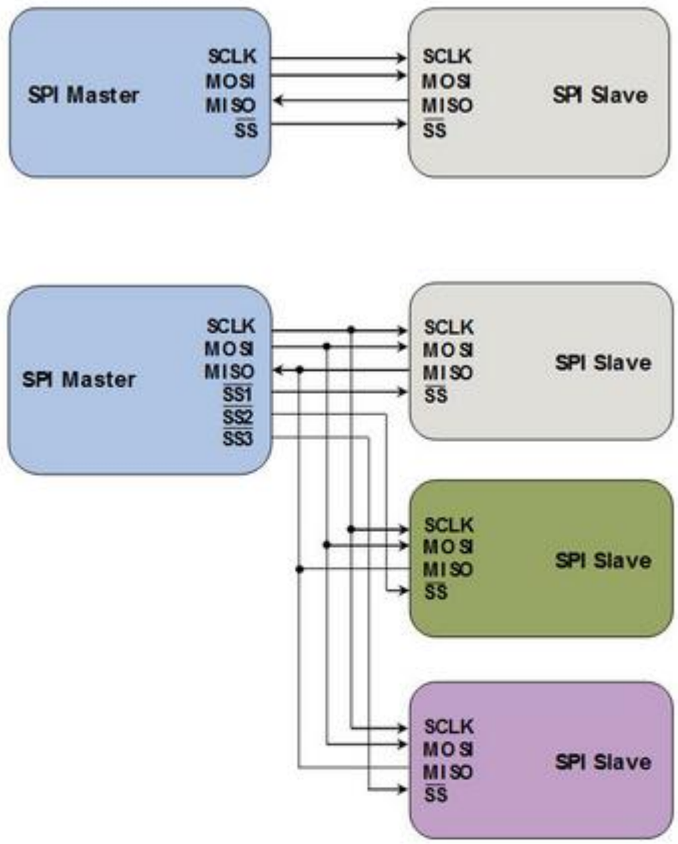


Figure 3.9: Sample Diagram of SPI
(Reprinted with Permission from byteparadigm.com)

I^2C uses two connections per device: for input and output communication. It behaves as a shift register, with data cycling through the devices connected by wire. The I^2C mode of communication is an improvement over SPI for cases where low footprint takes precedence over data-throughput of a system. Being connected by one wire though, the information throughput is less as each baud is transmitted serially versus in parallel. A sample diagram of I^2C is included in figure 3.10.

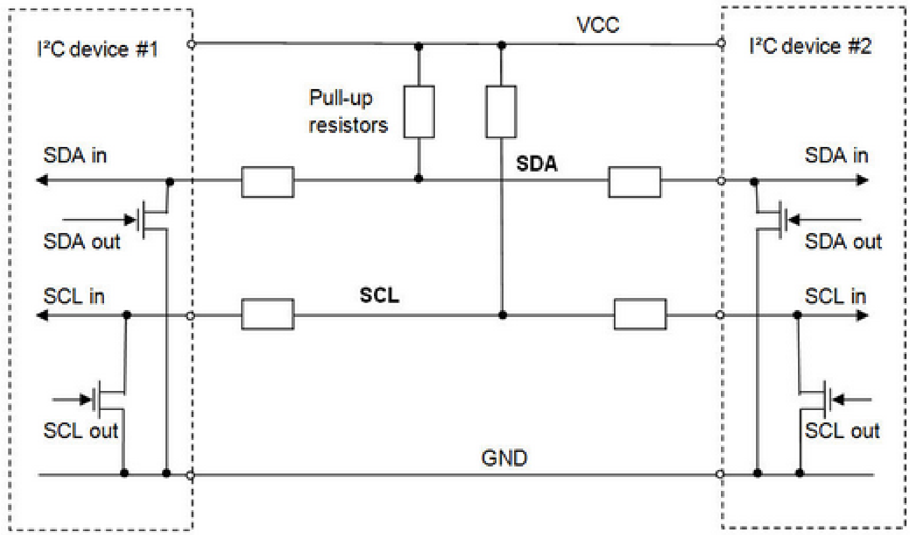


Figure 3.10: Sample diagram of I^{2}C
(Reprinted with Permission from byteparadigm.com)

UART uses three connections: TX (transmitting pin), RX (receiving pin), and a connection to ground. An asynchronous serial signal, UART doesn't require a clock, making it ideal for cases where power efficiency is greater than the need for a high data refresh.

3.3.4.3.3 Architecture

The MCUs are each designed to follow an architecture, of which there are two fundamental types: Von Neumann and Harvard. There are implications in programming with the decision of architecture type related the performance, size and cost of the processor.

Von Neumann machines use the same memory space for data and instructions, and are thus addressed equally by the CPU of the microcontroller from a single set of address/data buses. Harvard machines store data and instructions in separate memory spaces and are thus addressed by two sets of address/data buses. The hardware of Von Neumann architectures are simpler, owing to them having one memory space rather than two. The reduced memory footprint comes with a caveat in performance, which Harvard architectures are more suited for with their greater memory bandwidth. Where footprint and cost are of greater concern, Von Neumann architectures are used and for performance, Harvard based processors are more appropriate.

3.3.4.3.4 Programming

In addition to architectures, instruction sets are another important consideration for MCUs, the two being RISC (Reduced Instruction Set Computer) and CISC (Complex Instruction Set Computer). RISC , as the name implies, have fewer instructions available but this limitation allows for better compiler optimization, requires little to no intervention on behalf from the human programming the unit. CISC conversely has more instructions available allowing the machine's hardware to process more data with each iterative instruction, though requires action from the programmer to optimize the assembly. Between the two, CISC has greater code density than RISC because of the larger instruction set.

Ultimately, our group is programming in C language and will optimize the code only when all other aspects of the project have been handled first or if absolutely necessary for H.A.R.M. sub-systems to function properly. This shouldn’t suggest that knowledge of a machine being RISC or CISC based is arbitrary, rather it’s important for understanding the flow of information and which programming styles are most apt for a given task.

3.3.4.4 MCU Constraints

Each processor is limited in the amount of hardware that can be fit in a package, which affects the MCU's voltages tolerances, power requirements, pin multiplexing, and other factors. In addition, variables aside from the hardware weighed in our decisions: cost, programming interface, and manufacturability. Table 3.3, Comparison of Popular MCUs, was compiled to compare popular MCU choices from similar control system projects:

MCUs					
Company	TI	Atmel	TI	Atmel	STMicro
Nickname	MSP430	ATmega 328p	Tiva-C	SAM D20	STM32F3
Part#	MSP430G 2553		TM4C1294N CPDT	ATSAMD2 0J18	STM32F303 VCT6
SRAM	0.5 kB	2 kB	256 kB	32 kB	48 kB
Flash	16 kB	32 kB	1024 kB	256 kB	256 kB
# of Pins	24	32	128	64	48, 64, 100
Clock Freq	16 Mhz	20 Mhz	120 Mhz	48 Mhz	32 Mhz
Logic Voltage	+3.3V	3.3V, +5V	+3.3V	+3.3V	+3.3V
Input Voltage	+1.8-3.6V	+1.8-5.5V	+3.18-3.63V	+1.8-3.3V	+2-3.6V
Comm Protocols	I2C (1) SPI (1) UART (1)	I2C (1) SPI (2) UART (1)	I2C (10) SPI (4) UART (8)	I2C (6) SPI (6) UART (6)	I2C (2) SPI (3) UART (5)
Cost	$.90 | 1ku	$3.88 | 1u	$8.07 | 1ku	$2.92	$10.88 | 1u
Packages	QFN/PDIP	TQFP, PDIP	TQFP	TQFP	LQFP
Software Dev	C/Assembly	C	C/Assembly	C	C
Programming Int.	UART/JTAG	UART/ISP	UART/JTAG	UART	JTAG/SWD
Processing Unit	Register/ALU	AVR CPU	Cortex-M4F	Cortex-M0+	Cortex-M4F
Timers	16-bit (2)	8-bit (2), 16-bit (1)	16/32-bit (8)	8	32-bit(1), 16-bit(5)
Advantages	Low power	Very well documented, large tolerances	Best processing	Median Processing, cost	Easy digestible datasheet
Disadvantages	Less processing performance, lack of peripherals	Not enough peripherals for baseline	Cost, distribution of peripherals	Least amount of experience	Enigmatic* processor

Table 3.3: Comparison of Popular MCUs

Following this, we applied our baseline requirements begin the design process given the hardware available on the market. Elementary processors such as Texas Instrument's MSP430 and Atmel's ATMega328p are very well documented with datasheets and project build logs though lack the amount of COM ports necessary for our project with their standard hardware. Multiples of these processors could be arranged for parallel processing but would likely have increased the cost of development and difficulty to manufacture.

The ARM processors meet the baseline requirements for COM ports and some have the added benefit of a floating point unit as part of the hardware, making analogue measurements and images for an LCD display easier to realize.

3.3.5 Sensors

Sensors on the market come in all forms, specifications, efficiencies, accuracies, and other parameters. Selecting one can be vital to a system's operational performance depending on the sensor and the system. In the following section, sensors for a given subsystem are compared and selected for the implementation within the hydroponics system.

3.3.5.1 Light Sensor

An ambient light sensor is need to monitor the intensity of light to determine if additional light exposure to the hydroponics system is needed by use of the light array subsystem. The first component to take into account was the accuracy needed for the ambient light sensor. For the purposes of the hydroponics system, the ambient light sensor subsystem need only to be a general measurement, for this is a plant system and while plants require sunlight, they are not overly sensitive to a precise exposure to sunlight. The second component to take into account was the output protocol required for the sensor. For integration into the selected microcontroller unit for the hydroponics system, analog, UART, SPI, and I^2C are all viable methods of communication protocols. The third component was the power efficiency of the sensor, some sensors can consume multiple watts of power but, for the purposes of the hydroponic system, the light sensor subsystem is by default in the range of less than one watt. The last component to take into account was the price of the light sensor, the price for the light sensor was set to not exceed $50.00, and this also takes care of the more accurate light sensors which are not practical for this hydroponics system. After reviewing and researching multiple solutions for the light sensor subsystem, the Intersil ISL 29023 ambient and infrared light sensor were used in the light sensor subsystem. This sensor was mainly selected because it met all the practical requirements for the hydroponics sensor subsystem as well as, being incorporated into the Texas Instruments Sensor Hub BoosterPack as the light sensor. This booster pack is used for integration with the TivaC, the microcontroller unit selected for the hydroponics system. Therefore the integration was tested and indeed successful in the field. Allowing for the avoidance of integration problems when integrating the entire hydroponics system.

3.3.5.2 pH Sensor

A pH sensor is needed to monitor the pH levels of the water solution to determine if a water flush is needed of the hydroponics system by use of the water flush subsystem. This pH sensor subsystem must be water proof at the probe, since it will be used in a water based environment in the hydroponics system. The first component to take into account was the accuracy needed for the pH sensor. For the purposes of the hydroponics system, the pH sensor subsystem requires an accuracy of 0.1 pH, for this is the resolution that different types of plants are specified for when growing. The second component to take into account was the output protocol required for the sensor. For integration into the selected microcontroller unit for the hydroponics system, analog, UART, SPI, and I^2C are all viable methods of communication protocols. There are various pH sensors on the market that use USB or UART and I^2C protocol. While both are viable options for the interfacing of the pH sensor, the use of USB sensors has been selected to not be used, as to avoid complications of interfacing and PCB construction due to the existing complexity of the hydroponics system, additional complexity should and will be avoided. Therefore the I^2C protocol is used for the pH sensor subsystem. The third component was the power efficiency of the sensor, after research of various pH sensors, the power consumption of consumer grade pH sensors is negligible for the application within the hydroponics system. The last component to take into account was the price of the pH sensor, the price for pH sensors is relatively high, and so the cheapest sensor which meets all the other specification requirements will be used for the pH sensor subsystem. The only sensor found after researching sensor distributors that meet all the mentioned requirements is the Atlas Scientific EZO class embedded pH circuit.

3.3.5.3 Water Level Sensor

A water level sensor is needed to monitor the water level of the water solution to determine if a water flush or water addition is needed of the hydroponics system by the use of the water flush subsystem or water pump subsystem respectively. The first component to take into account was the accuracy needed for the water level sensor. For the purposes of the hydroponics system, the water level sensor an accuracy of at least 0.5 inches, for this is the resolution of root immersion within the water solution for different types of plants that are specified for when growing. The second component to take into account was the output protocol required for the sensor. For integration into the selected microcontroller unit for the hydroponics system, UART, SPI, and I^2C are all viable methods of communication protocols. The third component was the power efficiency of the sensor, which should be relatively minimal or less than 1 watt of power. After research of various water level sensors, the power consumption of consumer grade water level sensors is negligible for the application within the hydroponics system. The last component to take into account was the price of the water level sensor, the price of for the water sensor is relatively low, and so a water level sensor that meets the above mentioned requirements. The water level sensors found after researching sensor distributors that meet all the mentioned requirements is the "eTape" or continuous fluid level sensor PN-12110215TC-12. This type of sensor comes in various lengths or sizes, for the purposes of the hydroponics system, the 12 inch eTape will be used to accommodate a water depth of 12 inches.

3.3.5.4 Ambient Temperature Sensor

A temperature sensor is needed to monitor the air temperature levels of the surrounding air to determine the health of the plants within the hydroponics system. The first component to take into account was the accuracy needed for the ambient temperature sensor. For the purposes of the hydroponics system, the ambient temperature sensor subsystem requires an accuracy of 1 degree Fahrenheit. The second component to take into account was the output protocol required for the sensor. For integration into the selected microcontroller unit for the hydroponics system, analog, UART, SPI, and I^2C are all viable methods of communication protocols. There are various ambient temperature sensors on the market that use all of these protocols. While all of these methods are viable options for the interfacing of the ambient temperature sensor subsystem, the use of USB sensors has been selected to not be used, as to avoid complications of interfacing and PCB construction due to the existing complexity of the hydroponics system, additional complexity should and will be avoided. Therefore the I^2C protocol is used for the ambient temperature sensor subsystem. The third component was the power efficiency of the sensor, after research of various ambient temperature sensors, the power consumption of consumer grade ambient temperature sensors is negligible for the application within the hydroponics system. The last component to take into account was the price of the ambient temperature sensor, the price for the ambient sensors is relatively low, so the selection of ambient temperature sensors is not sensitive to price variance. There are numerous ambient temperature sensors that meet the above mentioned requirements. Therefore for the sake of simplicity the Sensirion SHT21 humidity and ambient temperature sensor will be used, because of the part's integration into the Texas Instruments Sensor Hub Booster Pack. This sensor hub is compatible with the TivaC, the selected microcontroller unit for the hydroponics system.

3.3.5.5 Water Temperature Sensor

A water temperature sensor is needed to monitor the water temperature levels of the water solution within the hydroponics system, in order to determine if a water flush is necessary due to the water temperature levels. This water temperature sensor must be water proof due to the immersion within the water solution of the hydroponics system. The first component to take into account was the accuracy needed for the water temperature sensor. For the purposes of the hydroponics system, the water temperature sensor subsystem requires an accuracy of 1 degree Fahrenheit. The second component to take into account was the output protocol required for the sensor. For integration into the selected microcontroller unit for the hydroponics system, analog, UART, SPI, and I^2C are all viable methods of communication protocols. There are various water temperature sensors on the market that use all of these protocols. While all of these methods are viable options for the interfacing of the water temperature sensor subsystem, the use of USB sensors has been selected to not be used, as to avoid complications of interfacing and PCB construction due to the existing complexity of the hydroponics system, additional complexity was avoided. Therefore the I^2C protocol was used for the water temperature sensor subsystem. The third component was the power efficiency of the sensor, after research of various water temperature sensors, the power consumption of consumer grade water temperature sensors is negligible for the application within the hydroponics system. The last component to take into account was the price of the water temperature sensor, the price for the water sensors is relatively low, and so the selection of water temperature sensors was not sensitive to price variance. There are numerous water temperature sensors that meet the above mentioned requirements. The sensor that meets all these requirements most closely was the Maxim DS18B20 Programmable Resolution 1-Wire Digital Thermometer, which has a variety of package that meets the water proof requirement is available on the market, and has been chosen for the hydroponics system.

3.3.5.6 Dissolved Oxygen Sensor

A dissolved oxygen sensor is needed to monitor the oxygen levels of the water solution within the hydroponics system, in order to determine if a water flush is necessary due to the oxygen levels. This dissolved oxygen sensor must be water proof due to the immersion within the water solution of the hydroponics system. The first component to take into account was the accuracy needed for the dissolved oxygen sensor. For the purposes of the hydroponics system, the dissolved oxygen sensor subsystem requires a general accuracy. The second component to take into account was the output protocol required for the sensor. For integration into the selected microcontroller unit for the hydroponics system, analog, UART, SPI, and I^2C are all viable methods of communication protocols. There are various dissolved oxygen sensors on the market that use all of these protocols. While all of these methods are viable options for the interfacing of the dissolved oxygen sensor subsystem, the use of USB sensors has been selected to not be used, as to avoid complications of interfacing and PCB construction due to the existing complexity of the hydroponics system, additional complexity should and will be avoided. Therefore the I^2C protocol is used for the water temperature sensor subsystem. The third component would be the power efficiency of the sensor, after research of various dissolved oxygen sensors, the power consumption of consumer grade dissolved oxygen sensors is negligible for the application within the hydroponics system. The last component to take into account was the price of the dissolved oxygen sensor, the price for the dissolved oxygen sensors was relatively high, so the selection of a dissolved oxygen sensor will be that of the cheapest available sensor on the market that meets the fore motioned requirements. There are extremely few dissolved oxygen sensors that meet the above mentioned requirements. The sensor that met all these requirements most closely is the Atlas Scientific EZO class embedded Dissolved Oxygen circuit, which has a variety of package that meets the water proof requirement is available on the market, and has been chosen for the hydroponics system.

3.3.6 Camera

A camera was considered to monitor the visual activity of the hydroponics system and report the live status of the system. There are multiple components of the camera specifications that must be taken into consideration when selecting an appropriate camera for use in the hydroponics system. First the camera would have to interface via a USB output to the video feed to integrate with the BeagalBone Black. Secondly, the camera must be weather proof or weather resistant in order to be sustained in an outdoor environment. The last requirement was for the camera to be within a reasonable price range, a price less than $100 was selected for the purposes of the hydroponic system. There are many cameras on the market for consumer use, which meet the requirements stated above. The Dbpower MINI HD 720P Waterproof Sport Action Helmet camera was selected as an appropriate camera for this subsystem in the hydroponics system.

3.3.7 Single Board Computer

One of the requirements of our hydroponics system is that the sensor data can be viewed through a normal web browser as well as the ability to send control data back through the interface to the MCU board through UART. This was proposed to be done using an easily accessible and programmable single board computer that runs a distribution of Linux. Among our research 3 boards stood out among the rest. The Raspberry Pi 2 Model B from the Raspberry Pi foundation, the BeagleBone Black from Texas Instruments, and the Intel Edison from Intel Corporation.

3.3.7.1 Raspberry Pi 2

The Raspberry Pi 2 Model B is the latest model of the Raspberry Pi series of small computers produced by the Raspberry Pi foundation with support from the Broadcom Corporation. It is the most popular series of single board computers by far compared to any other. It features more graphical power than other single board computers, able to produce a 1080p60 resolution over an HDMI connection. It also features a Broadcom BCM2836 900 MHz quad-core ARM Cortex-A7 CPU, 1GB LPDDR2 RAM, 4 USB Ports, a microSD card slot, and 40 GPIO pins. Figure 3.11 showcases the I/O pin configuration of the Raspberry Pi 2 Model B for reference.

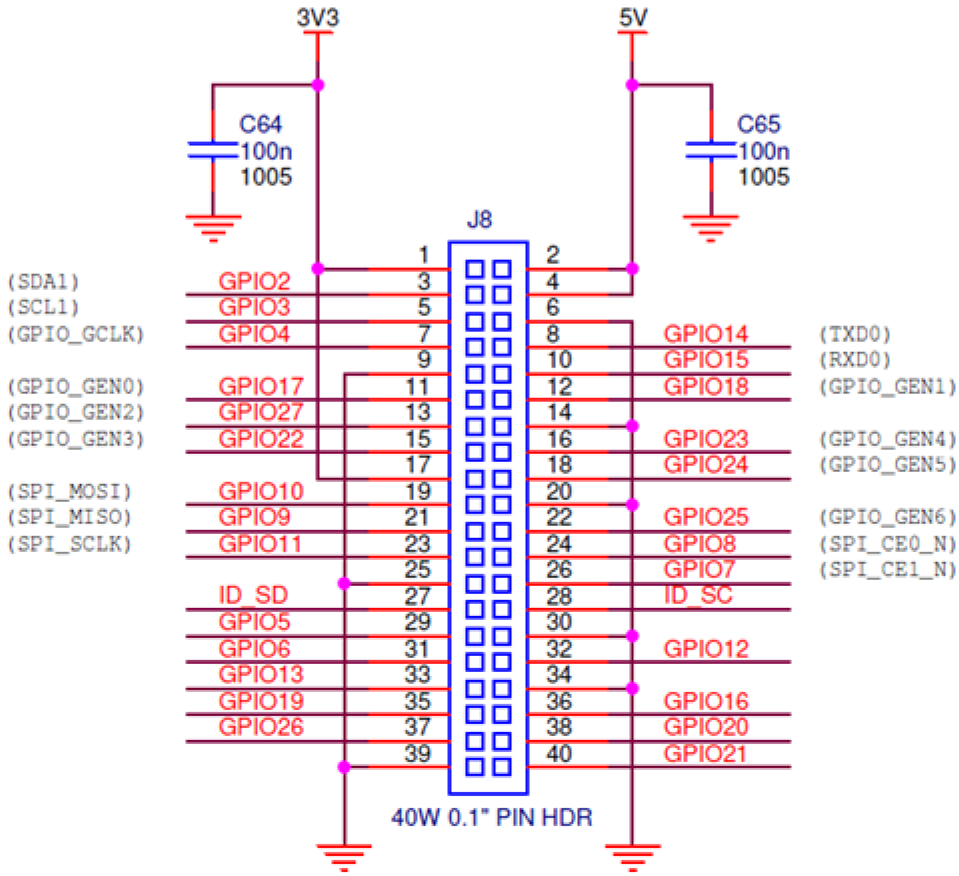


Figure 3.11: The Raspberry Pi 2 Model B GPIO Configuration
(Edited and Reprinted with Permission from the Raspberry Pi Foundation)

Since this device has a USB port, it would be easy to turn it into a Wi-Fi device, ensuring Wi-Fi connect ability to our webserver through compatible Linux drivers. The Raspberry Pi 2 Model B also supports UART natively on GPIO pins 14 & 15, which send and receive, respectively. It also features a 15-pin MIPI Camera Serial Interface (CSI-2), which makes it easy to add a video feed to our project. The average power consumption is around 2.6W idle and around 4W max according to an online benchmark [11]. The advantage of using this board is a large community to talk to in order to figure out how to interface with our microcontroller through UART as well as the relatively low cost to obtain one: $35.00 + shipping. However, the disadvantage of using this board is that the documentation isn't open due to Broadcom restrictions as well as a slightly higher power requirement compared to other boards.

3.3.7.2 BeagleBone Black

The BeagleBone Black is a product of the BeagleBone.org Foundation, which is made up of multiple engineers, including engineers from Texas Instruments to support an open-source software and hardware platform. It is a cheaper version of the original $89 BeagleBone. BeagleBone.org Foundation's dedication to having an open platform shows because the schematics and PCB files are openly available online to download and to reference for your own designs. The BeagleBone Black features a Texas Instruments AM335x 1-GHz Sitara ARM Cortex-A8 32- Bit RISC Processor, 4GB 8-bit eMMC on-board flash storage, a micro-HDMI port capable of outputting 1280x1024 @ 60 fps, 512 MB DDR3L RAM, 1 USB port, 69 digital GPIO pins and 7 analog inputs to a 12-bit ADC as well as 8 PWM outputs. The power usage, directly from the BeagleBone wiki page, is 210-460 mA @ 5V (1.05W – 2.3W). Since this device has a USB port, it would have been easy to turn it into a Wi-Fi device, ensuring Wi-Fi connect ability to our webserver through compatible Linux drivers. Figure 3.12 showcases the vast amount of I/O pins available to the BeagleBone Black as well as their locations for reference.

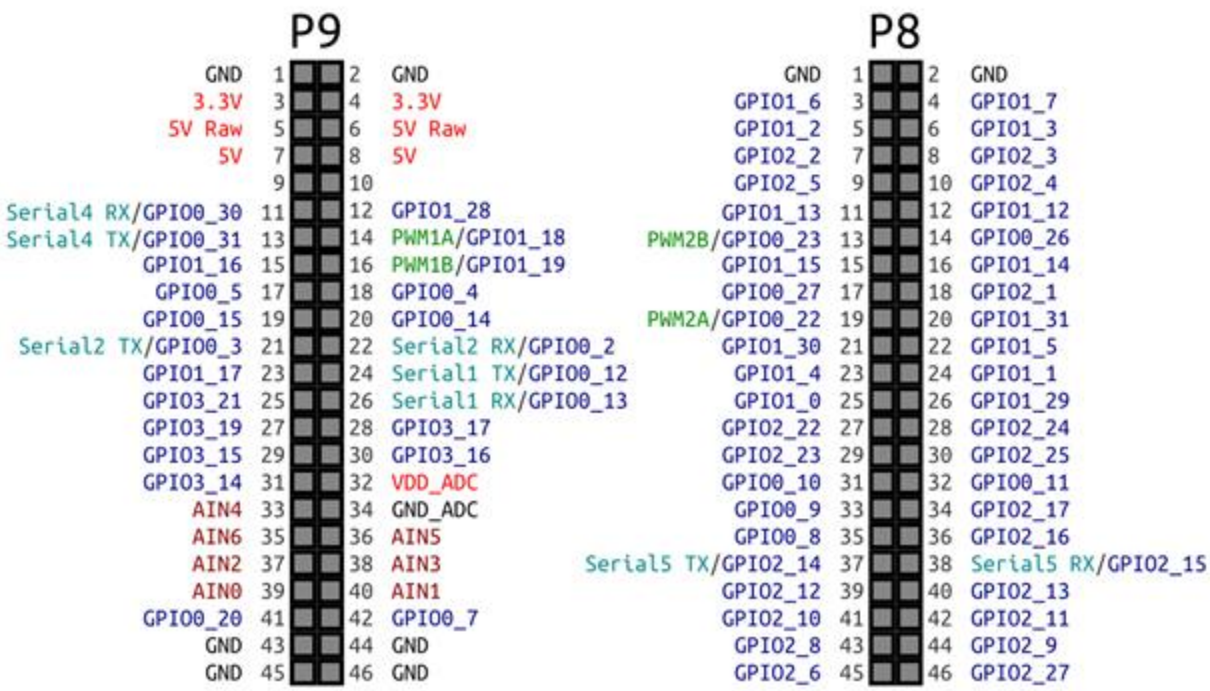


Figure 3.12: BeagleBone Black GPIO Configuration
(Reprinted with Permission from graycat.io)

The BeagleBone Black also has 5 UART pins and more connectability than any other single board computer around this price range, which is a plus to interface with our system. The advantage of using the BeagleBone Black is the lower power requirement, great open-source community, open-source hardware and schematics as well as datasheets available, and no need for a microSD card to function with a pre-configured eMMC flash storage. The disadvantage is a slightly higher cost, being around $55 plus shipping, only 1 USB port which may make it harder to interface a camera, and smaller community versus the Raspberry Pi.

3.3.7.3 Intel Edison

The Intel Edison is a product of the Intel Corporation that was created in order to quickly prototype and develop many different types of applications, including sensor and internet connected devices and projects. The Intel Edison features a

500 MHz dual core Intel Atom CPU with Intel Quark microcontroller clocked at 100 MHz, 4GB eMMC on-board flash storage, 1 GB LPDDR3 POP memory, 1 USB Port, 20 digital input/output pins, including 4 pins as PWM outputs as well as 6 analog inputs, onboard 802.11 a/b/g/n Wi-Fi and Bluetooth 4.0. The Intel Edison has the lowest power requirement which can range from 13 mW - less than 1W. It is also the only single board computer that has Wi-Fi built in without the need of an external adapter. It also supports UART but the documentation is scarce to get it working. The following figure, Figure 3.13, showcases the pinout for accessing the various I/Os of the Intel Edison.

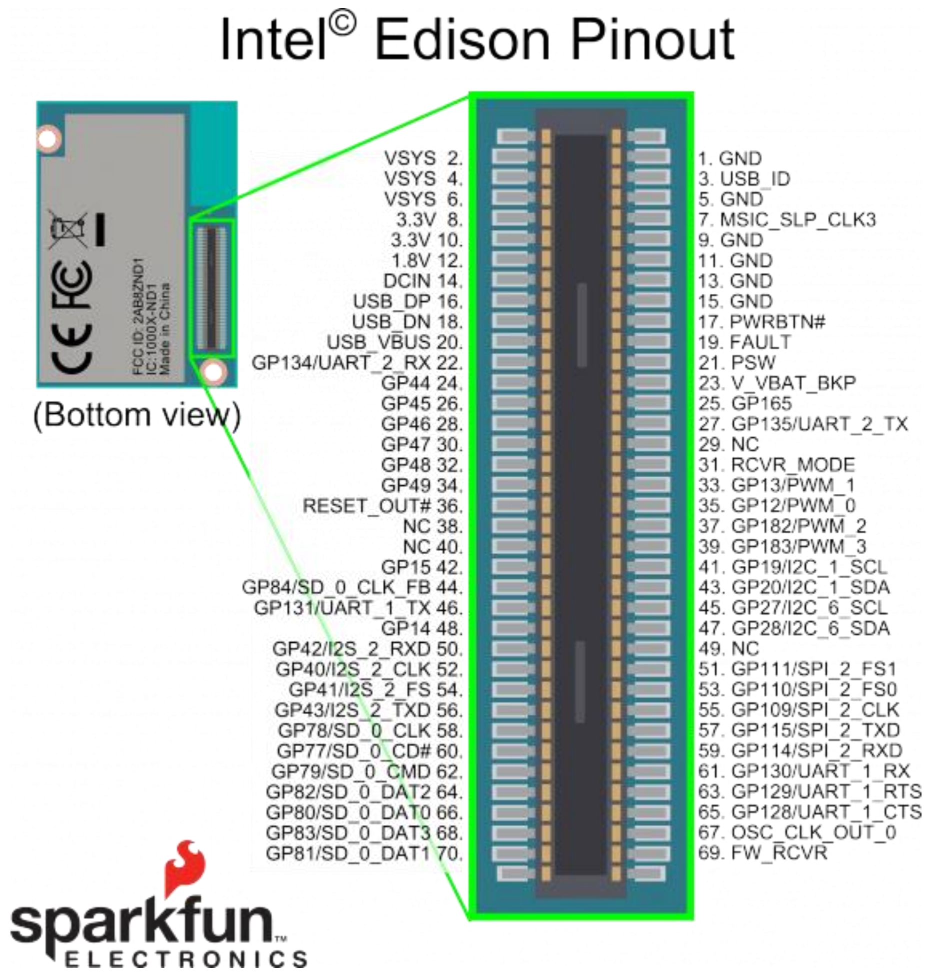


Figure 3.13: Intel Edison Pin Configuration
(Reprinted with Permission from sparkfun.com)

The advantages of using the Intel Edison were the small power requirement and built-in Wi-Fi module. However, the disadvantages greatly outweighed the advantages. Since this device is built by Intel, the hardware is a closed system. Also while researching, it was discovered that the community is not as big as the BeagleBone Black and the Raspberry Pi, which will make it harder to implement this in our project. The documentation is also rather scarce. In order to interface with the Intel Edison, a breakout board is required and the cost for both, which is $75 + shipping, is higher compared to the other boards that were researched.

3.3.7.4 Decision
The team made the decision that the BeagleBone Black would be the best to implement for our webserver hardware. The reason was that the BeagleBone Black had the most open documentation and still had a large community for support should any problems arise. It also had plenty of GPIO pins in case more than one UART was needed. The BeagleBone Black also had lower power consumption than the Raspberry Pi 2. Since graphical power isn't necessary, the more powerful graphics processor of the Raspberry Pi 2 did not justify the extra power consumption.

3.3.8 Voltage Regulator
One of the major components of our project is the voltage regulator, as it translates voltage from our battery power source into something usable by our microcontroller and sensors. Since efficiency is important with regards to our project, the design will be a switching based voltage regulator which is known for its high efficiency in combination with another switching regulator. The switching regulator circuit is used for the initial drop in voltage from 12V to 5V for use with some of the sensors as well as the LCD screen, and the second switching regulator circuit will be used to drop the 5V output to 3.3V for use with the MCU itself. Using the powerful WEBENCH tool on TI's web site, it was easy to compare and contrast different designs based on our requirements as well as efficiency, size, and cost.

Using this website, the TI IC model TPS563200 stood out as one of the better options for the first stage switching voltage regulator. It is capable of handling a wide range of input voltages and output voltages as well as having more than enough current for our application. It is also capable of efficiencies that are greater than 90% which is ideal (see Figure 3.14). The second stage voltage regulator IC that is appealing is the LM2854Y, which takes a relatively smaller input voltage range and outputs exactly 3.3 V for our MCU. It also can output more than enough current for our entire board. The efficiency of this chip is also above 90% (see Figure 3.14). The following figure, Figure 3.14, shows the graphs of efficiency, taken from datasheets available on TI's website for both of the ICs:

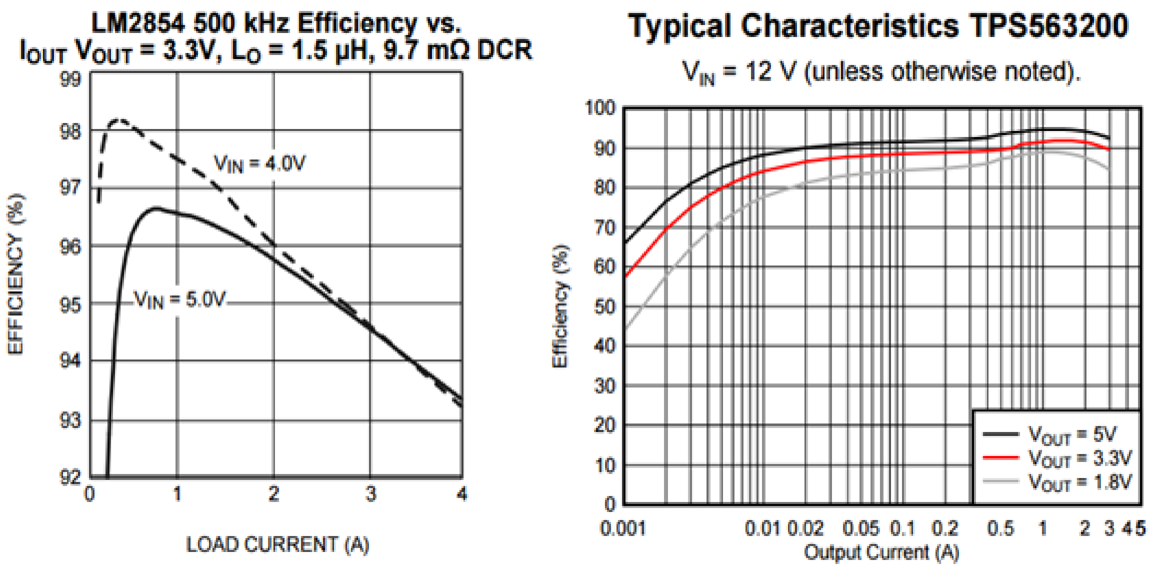


Figure 3.14: Efficiency Graphs
(Reprinted with Permission from Texas Instruments)

The LM2854's graph shows an efficiency of greater than 90% at all currents, whereas the TPS563200 needs current draw to be greater than 20mA for greater than 90% efficiency. The following table, Table 3.4, showcases the features of both the LM2854Y and the TPS563200:

Model	**LM2854Y**	**TPS563200**
Input Voltage Range	2.95 – 5.5 V	4.5 – 17 V
Output Voltage Range	0.8 – 5 V	0.76 – 7 V
Maximum Output Current	4 A	3 A
Maximum Switching Frequency	1 MHz	650 kHz
Max Duty Cycle	100%	80%
Pin/Package Type	16 pin Thermally Enhanced Thin Shrink Small-Outline Package (16HTSSOP)	6 pin Small Outline Transistor (6SOT)
Cost	$2.10	$0.52

Table 3.4: Specifications for Voltage Regulator ICs

3.3.9 LCD Screen

There were a few concerns when it comes to the LCD screen. Among those concerns are three major criteria:

- How easy will the screen be to implement into our project?
- Is the screen big enough to display all of the relevant information?
- Will a touchscreen add to the experience?

The first screen considered due to having experience with it is the Nokia 5110 LCD. This is a low cost screen. However, it is also a very small screen and is only available in black and white and no touch interface. This screen does not seem to meet the needs of the project so it was decided that we would not use it. The second screen to be considered was a color LCD BoosterPack by RobG. This is a 2.4” panel with the option to have a touch panel or not. While this was a very promising option, the panel was out of stock for an extended period of time. The last board to be considered was the Kentec 3.5” LCD with Touch BoosterPack for the Stellaris series microprocessors. This board seems to be the best fit because the screen would be able to display all of the information from the microprocessor at a visible size. It is also easy to implement as it is officially supported by Texas Instrument’s TivaWare libraries and grlib, a Texas Instruments graphics library designed to make it easy to implement your graphics on supported hardware. The final consideration was the touchscreen. Ultimately, it was decided that a touchscreen would make it more convenient for the user to navigate pages directly on the LCD screen, considering the wide use of tablets and smartphones that we see today. The following table, Table 3.5, showcases some of the comparisons between LCD screens that were considered for our project:

Company	**Nokia**	**Custom made by RobG**	**Kentec**
Model	5110	No specific model number	EB-LM4F120-L35
Size	1.5”	2.4”	3.5”
Color	No (Monochrome)	Yes	Yes
Touchscreen Capability	No	Optional	Yes
Resolution	84x48	320x240	320x240
Voltage Required	3.3 V	3.3 V	5 V
Data Bus	Serial	SPI	Parallel
Pins Required	8	Unknown exactly; around 20	22
Advantages	Supports 3.3 V natively.	Supports 3.3 V natively. Good resolution. Color display. Touchscreen Capability.	Good resolution. Good screen size Touchscreen Capability. Supported by Texas Instrument’s grlib graphics library.
Disadvantages	Too small. Monochrome. No touchscreen.	Not supported by Texas Instrument’s grlib graphics library. A little too small.	Does not support 3.3V natively (the voltage of our MCU)

Table 3.5: Comparison of LCD Screens

3.3.10 Software

When it comes to software, we had to make two major decisions for our project. The first decision was which PCB design software that we would use. This was an important consideration because ultimately the ability to efficiently and effectively build our board is influenced by the decision of which software suite to use. The second decision was which coding environment we would use. This was equally important because a large portion of our project relies on software that works reliably and correctly. Picking the correct coding environment ensures that we will have as little difficulty as possible programming our microcontroller while also ensuring that the compiler will effectively convert our C code to assembly for the processor.

3.3.10.1 PCB Design Software

Our group compiled the following table to compare different PCB CAD software environments available to use. They're all adequate tools for the requirements of this project: output files are of a standardized format, size limitations aren't a major factor, and most support hierarchical sheets. The two largest contributing factors for our decision are 1) support and 2) cost. Even though Altium Designer is the most expensive option, we were able to access it through work licenses for free. Considering the vast amount of libraries and resources available to Altium Designer, as well as the fact that it's something that many companies in the industry use, we chose to go with Altium for our PCB and schematic editing software. Altium also has the ability to import many other file formats (Eagle, KiCAD, OrCAD, etc.) for use with your own projects. Another useful feature is the ability to build a bill of materials and order based on your design, ensuring that we have all the parts for assembling our PCB. Our group compiled Table 3.6 to better compare the differences between CAD software.

PCB CAD Software				
Name	KiCAD	Eagle	Altium Designer	Orcad
Company	Open-source	Cadsoft Inc.	Altium Inc.	Cadence
License Cost	Free	Free (10x8cm) $169 (10x16cm)	$9,000 for perpetual license	Free (75 nets Max) $2,500 (No limit)
Size Limitations	No	Yes	No	No
Hierarchical Sheets	Yes	Not in free version	Yes	Yes
Community Support	CERN	Good Hobbyist Support	Through Altium, Poor Hobbyist Support	Fair
Platforms	Win, OSX, Linux	Win, OSX, Linux	Win	Win
Gerber File Type	274-X	274-X	274-X	274-X
Advantages	Best free software, supported by CERN foundation	Professional tool, mature software dev	Industry standard, huge part library	Mature software
Disadvantages	Not considered professional, documentation below average	Size limited free version	Very expensive	Limited net list in free version

Table 3.6: PCB CAD Software Comparison

3.3.10.2 Coding Environment

For coding environment our group had a few options. With all of the coding environments having meet the requirements for our project, we evaluated which studio would be the best choice on the basis of availability for the group. This immediately took out CrossWorks by Rowley Associates because only one of our group members had a license for the software. We could have gone with any of the remaining options: Code Composer Studio by Texas Instruments, IAR Embedded workbench by IAR systems, Keil uVision, or Atollic TrueStudio since we decided our MCU would be the TM4C1294 by TI, an M4 ARM processor. Ultimately, we decided Code Composer Studio would be the best environment because it seemed more likely we would encounter less error in setting up the coding studio than IAR or Atollic coding suites, which could take days to debug. Code Composer Studio also has the advantage of having many example libraries written in the form of TivaWare. For further information, please reference Table 3.7 which includes even more specifics concerning versions, available platforms, and licensing costs.

Coding Environment				
Name	Code Composer Studio	IAR Embedded Workbench	Atollic TrueStudio	CrossWorks
Company	Texas Instruments	IAR Systems	Atollic	Rowley Associates
License Cost	Free	Free (size limit) $6,000 (no limit)	Free (size limit) $3,000 (No limit)	$150
Version	6.1.0	7.4.0	5.3.0	3.3.0
Platforms	Win, Ubuntu	Win	Win	Win, OSX
Size limitations	None	Free Ver. - 32kB	Free Ver. - 32kB	None
Support	Good	Excellent	Good	Excellent
Advantages	Free and good quality	Industry standard, powerful, large # of supported devices	Industry standard, specifically for ARM proc.	Group member already familiar with soft., powerful
Disadvantages	Lacks advanced features, locked to TI devices	Free version size limited, pro is expensive	Free version size limited, pro is expensive	Only one member has it (code dev constrained)

Table 3.7: Comparison of Coding Environments

3.3.10.2.1 Coding Languages

Since Code Composer Studio is the coding suite that we've decided to use, there are two options for coding languages. The first option is assembly. The advantage of assembly is that we can write very efficient code since we are directly controlling registers and how they are used. It also helps to gain a greater understanding of the workings of the processor and to know exactly what is going on in terms of instructions. However, one of the major disadvantages of assembly is that it is different for every type of architecture and usually requires a fair amount of time to become familiar with. The second option is the C coding language. The advantage of the C coding language is that it can be used universally wherever C is an acceptable language. This greatly reduces the amount of time it will take to write, test, and debug the code. Syntax is the same between different architectures using C, however some functions may need to be written differently. The disadvantage of C is not being able to control exactly how the processor is going to function because the compiler translates your C code into assembly for you. Due to the limited time constraints of our project, it was decided that C would be the language of choice for the MCU code.

3.3.10.3 Linux

The BeagleBone Black comes with Debian Linux pre-installed and as also the recommended Linux distribution according to the BeagleBoard.org foundation. In order to maintain compatibility with as little room for error as possible, the latest version of the recommend Debian image would be used for the BeagleBone.

3.3.10.4 Web Servers

A web server is a program that takes information stored on a server or local computer and transmits the data via HTTP to other computers on a network. This is a good method to be able to retrieve the data from our MCU and sensors and also submit commands to the MCU to control various systems of the project. A user could access their hydroponics system through an easy to understand and use web based UI powered by a web server. Since we are using Linux for our web server's operating system, the goal was to find a well-documented and lightweight web server program in order to not tax our BeagleBone Black. The less CPU usage, the lower the power usage is in our web server, which is better for longer lasting charges in our battery pack.

3.3.10.4.1 Apache Web Server

Originally released in 1995, Apache web server is a very popular web server that's used in a lot of different web applications as well as home servers. It is an open source program that runs on multiple operating systems, including Linux. Apache has a lot of features and is designed for full load, highly diverse websites. The advantage of using Apache is the high amounts of documentation and a large community. Recent ratings show it as the number 1 used web server program on the internet today. However, multiple benchmark sites claim that the performance of Apache is lower and it is not as lightweight as others. Every connection to Apache requires a dedicated process thread, meaning the CPU is being taxed more than it should be. Since efficiency is a priority, other web server software was also researched and considered. To summarize, the advantages of apache web server are documentation, the largest community in comparison to other web servers, it is completely open source software, it is robust with a large feature set, and it's one of the easiest servers to get started with. However, because it is so robust, apache suffers from having too much overhead, which makes it a non-desirable choice in order let our BeagleBone Black idle as much as possible.

3.3.10.4.2 Nginx

Originally released in 2004, Nginx (pronounced "Engine-X") is another open source web server that is available for both Linux and Windows based operating systems. Nginx was a project started with a similar philosophy to Lighttpd: scalability in mind. Nginx is known for having high performance but using little RAM and CPU resources in the process. The documentation for nginx is extensive and the community is large. According to Netcraft.com's January 2015 Web Server Survery, nginx is the 3rd most used web server software, hosting 15% of all websites in the world. According to multiple benchmark websites, the difference between Lighttpd and nginx at small loads, which is what our project will have, is negligible in terms of efficiency. However, no apparently memory leaks have been reported with nginx in the research that was done. Nginx also has a strong case to be the software implemented for the web server. The largest advantage to nginx is the large community combined with being a more efficient web server than Apache. Although nginx would have been harder to set up than other web servers, the large community combined with the open source nature of the software would make it a non-issue.

3.3.10.4.3 Lighttpd

Originally released in 2003, Lighttpd (pronounced “Lighty”) is an open source web server that is available for both Linux and Windows based operating systems. Lighttpd was built with scalability in mind, which means that it was designed to use as little CPU load and memory resources as possible. Lighttpd also has extensive documentation and since it is used in some well-known websites (Youtube, etc), it has a decent sized community in case help is needed. According to Todd Hoff on highscalability.com regarding Youtube.com, “Servers use the lighttpd web server for video [because] Apache had too much overhead.” Some people have claimed there is a memory leak in the Lighttpd software. This makes a strong case to implement Lighttpd as our web server software, with some reservations. The advantages of using Lighttpd are good amounts of documentation, a decent sized community, being completely open source software, small footprint with less overhead than Apache, and it is easy to get started. The disadvantage is potentially having a memory leak problem, which would result in our web server becoming unstable or even crashing.

4.0 Standards and Specifications

Standards and specifications are tools which designers utilize to accelerate the development time of a project. These definitions are used as modules which can be treated as a block with a simple set of dependencies and parameters, a boon to designers in such they focus on the problem they're trying to solve rather than a subsystem to implement a function they need.

4.1 Related Standards and Specifications

The following are related to our project and impact our design choices when considering cost and the time to develop sub-systems for the project. This section is an exposition on some of the standards and specifications which influence the project.

4.1.1 802.11 Wireless

The 802 standard applies to local area networks (LANs), defining key specifications for the physical and data link networking layers. The two most well known standards, relevant to our project, are 802.3 (Ethernet) and 802.11 (Wireless LAN). Ethernet is included with the TM4C1294 in the FLASH memory, called upon by the user in the user-accessible FLASH portion and a Wi-fi dongle may be used with the single-board computer. It's therefore important that any possible physical or wireless connections follow the 802 standard, whether it be for Ethernet or wireless LANs, for consistency in the design and adaptations in the project.

4.1.2 C Programming Language

C is a programming language invented in 1972 by Dr. Ritchie of Bell Labs. The most current standard of C is defined as ISO/IEC 9899:2011, published in 2011 and also known as C11. We will be using the C language to program the Tiva-C ARM processor, being the highest level of abstraction available for ready implementation. Using a standardized programming language is useful as it makes finding a community or online forum for questions and inquiries easy to locate. It also assures our previous programming experience with C is relevant regardless of which platform each team member works on and communicate ideas for algorithm implementations.

4.1.3 JTAG Programming Interface

"Boundary scan" is a procedure for testing pin states and interconnects of a printed circuit board, it may also be used to program the circuit within the system. The Joint Test Action Group developed a standard which was then codified as IEEE1149.1-1990, the most current revision is IEEE1149.1-2013. The standard define 5 pins, one of them a optional reset, which communicate with all systems adhering to the boundary scan protocol. Various components such as sensors and MCUs can have their pin states read or FLASH memories programmed in-system, extremely useful for interfacing with manufactured PCBs for updating programs or debugging. Our group will likely use JTAG (defined by the IEEE1149.1 standard) or Serial Wire Debug interface, both implementations of boundary scan.

4.1.4 Chip Package Types

Industry has standardized different package types so PCB designers wouldn't need to make a new CAD footprint for every device they used in the circuit. Common packages are as follows:

- PDIP (Plastic Dual-Inline Package) - two rows of pins along a rectangular package, typical pin count is less then 30. Used for low-resolution devices, convenient for switching out chips if mounted in a socket. Popular for prototyping stages of product development.

- SOP/SOIC (Small Outline IC) - Same outline of PDIP but with dimension more appropriate for SMD implementation. Typical pin count is between 8 and 64.

- QFP (Quad Flat Package) - A square package type with rows of pins on each side, pins may have leads or pads on the underside. Pin count anywhere from 32 to 188. Moderate to hard difficulty to manufacture using hand soldering tools.

- BGA (Ball Grid Array) - Package with many pads on the underside of the device, each an interconnect to a pin within the device. These require a reflow over or advanced PCB mounting techniques capable only with expensive equipment. Not recommended for prototyping.

4.1.5 Connector Types

Connectors have multiple characteristics to ensure proper connectivity: keying, color coding, raised markings, environment proofing. It is of great import that our group choose components which use standardized connectors, for it's common for companies to use specialized connectors to restrict access across devices and results in expense of resources and time to work around. Easy to overlook but with the possibility of great consequences if so. Connectors such as MOLEX and 2.54mm crimp housings are fairly common and parts with such interconnects would be preferred over those without.

4.1.6 PCB File Types

Gerber files are the standard CAD file used to describe a PCB layout: the cut layer, trace layers, vias, silkscreen, and other parameters. Gerbers are defined by two standards: RS-274D which separates these layers into multiple files, and RS-274X which concatenates the information of all the layers into one file. Therefore PCB fabrication houses prefer RS-274X since data management is easier but both standards are equally relevant. It is up to the PCB facility's design rules as to what is and isn’t acceptable.

4.2 Design Impact of Relevant Standards and Specifications

Adopting a standard or specification is useful for design, they define protocols, mechanicals and electrical characteristics. Overall, a tool for a system designer to focus more on addressing the problem using a well-defined regulation where limitations, physical or otherwise, are easy to calculate. However, the standard could have less potential through restriction in the definition, examples in the following subsections.

4.2.1 Communication Bandwidth

Adopting a standard such a 802 or Bluetooth specification, simplifies the work of designing a communication protocol to account for transferring data, error checking, and other complications that arise when implementing a wireless communication network. These protocols do inherently influence power consumption, thus if we were concerned about power dissipation for a system that transmits at seldom frequencies, we would need to develop our own protocol. We believe that features about 802 and Bluetooth such as Digital Spread Spectrum Communication, which limits the amount of crosstalk between channels, would be more beneficial than developing our own protocol.

4.2.2 Component Layout

The sizes and layout of pins/pads affects the layout of components on a PCB, ultimately influencing the cut size, number of board layers and ounces of copper necessary for traces. Components sensitive to Electromagnetic Interference (EMI) need adequate separation from inductive sources, which include traces if current is switched sufficiently high frequencies, which can block high speed signals. Designing boards on the computer though are much easier in part because of the vast libraries of schematic blocks and footprints available for standardized chip and passive component packages. The Gerber PCB file standard also makes it convenient to export designs to PCB fabrication house, assuring the layout is sent in a common language both designer and manufacturer can read.

4.2.3 Processing Capability Given Language

One of the factors that influence the performance of code is the efficiency with which the compiler optimizes from our language of choice (C) to assembly. The ability to optimize our code would not have been feasible if our group had gone with a non-standardized language to write our programs. To say the least of developing linkers or run-time environments, it would have gone far beyond the scope of this project and miss our ultimate goals if we were to develop our own optimizing compiler of a non-standardized language for this project. The C language defined by the ISO/IEC 9899:2011 standard, is a very efficient language that optimizes well with most free compilers. Thus the C language had the best processing performance given its cost, time to develop programs, and library availability.

5.0 Realistic Design Constraints

Our design was constrained by more than pure physical limitations on the system. There was a range of factors: economic, social, environmental, safety and manufacturability were but a few of the topics covered. These extra-physical system constraints affected the design the more than physical type constraints and influenced our design choices that make sense only when considering these realistic design constraints.

5.1 Economic and Time Constraints

Our group, which was generously sponsored by Duke Energy, had a finite budget and time frame with which to design, build and test our hydroponics automation system. Our project group considered these constraints when establishing our system specifications and selecting parts and manufacturing methods for our design. The following is a more detailed exposition of the consequences these design constraints had with our project during our design phase.

5.1.1 Compromise of Cost Versus Performance for Parts

Throughout this project, the "cost versus performance" argument was perhaps the first to come to mind when researching parts and manufacturing methods. For example, our group could have attempted to design a PCB utilizing BGA components on a six layer board, but it would have cost a fair amount to fabricate the board and then would have to pay to have a company mount the components as we don't possess the tools to mount BGA components ourselves. There were many electronic components, parts for our automation subsystems and power regulators, we needed to consider what features we really needed from the part. Could we go with a similar part made out of plastic instead of metal? Is there an off-brand that has roughly the same characteristics as the more established brand? Could we modify an existing product to save not only money but design time which can be spent on other aspects of the project.

These questions, asked rhetorically here, greatly improved our design because it forced us to consider the most necessary qualities each part of our system needed. Another example, rather than purchasing a liquid fluid pump, our group found a submersible bilge pump intended for use on boats, a part which had the same rough characteristics but for a fraction of the cost. Yet another case, our group considered making our own LED light array for the lightening subsystem because it would need to be power from +12 volts. Instead we thought the better course of action would be to modify an existing LED bulb intended for ~120VAC for our system, taking only the time to modify the bulb for our system rather than designing, building, and integrating our own iteration. A huge savings when considering our learning focus is mostly sensors, MCU control and web server hosting.

5.1.2 Refinement of Design with Deadlines

After "cost versus performance", came the issue of time. Realistically we had three talented people with 7 months to design, build, and show our hydroponics system. We managed our goals and definition of project success by taking into account how much time we could actually contribute. There is a finite amount of time we could each contribute to the project, as we also had jobs, families, and other responsibilities to attend to. Thus our group needed to be as effective as possible with the time we had to work with.

Our deadlines this summer semester we're planned around design, prototyping, and building to do final design implementations. In January and February, our group did rough research of sub-systems and components to implement in our hydroponics automation system, forming initial software and hardware diagrams while starting a parts list. In March we started designing the MCU control board schematics and analyzing solar charge controller designs, also beginning this senior design 1 documentation. The month of April we finished documentation and had most of our designs for the automation sub-systems finished, ready for testing and refinement in the summer semester.

5.2 Environmental, Social, and Political Constraints

Our project has a range of implications on a local community level and on the global market level. Indeed, our group must practice and uphold ethical standards as environmentalists and consumers as we design and build our system. The following sections are a more detailed exposition.

5.2.1 Environmental Codes and Ethics

Environmental responsibility directly affects the operation procedures of the hydroponics system as the nutrient rich solution has implications which if improperly disposed, can lead to the eutrophication of lakes and rivers, harmful algae blooms in lagoons, and a loss of biodiversity in local aquatic environments. A case study for excess nutrients affecting the local environment is that of farmlands and the now regulated use of fertilizer. When excess fertilizer is washed into irrigation channels, the nitrogen, phosphates and trace elements is directly absorbed by algae, which has the potential of a short intensive effect such as an algae bloom or a long term effect of eutrophication, where the aquatic environment experiences hypoxia, a deficiency of oxygen which kills aquatic animals.

Since our system is intended for outdoor use, it was our responsibility to ensure the system was made with environmentally stable materials. Bare copper in the system needs protection not only for electrical hazards but also to prevent copper salts from forming an entering the environment. Numerous studies have found the impact copper salts have on poisoning aquatic animals, to include fish and crustacean, and its potential for bioaccumulation in land-based predators of these aquatic animals (ex. birds).

5.2.2 Project Effect on the Community

Our project has implications in our community from local, environmental and social, to a more global level, environmental again and political. We considered each aspect of our project to make sure our project was of minimal detriment to our community, from where each stage of our system is fabricated, to who we purchased our parts and electronic components from.

5.2.2.1 Manufacturing Site

Our manufacturing site needed to be an indoor space for initial construction of the automation system to meet our goal of environmental responsibility. We had many sites to choose from and the location changed during different stages of the build. For example, for modifications to the hydroponic system, we needed to use our industrial complex space because the noise of battery drills, linear reciprocating saws, and other cutting tools alike generated too much noise for our apartment residences. Electronic component mounting and testing though were done in a variety of places so long as proper ventilation was maintained while soldering components to boards.

5.2.2.2 Noise and Light Population Considerations

While the system was being tested, it had the potential to be a noise disturbance. The water pump is source of most noise from the system so when testing in the proximity of high-density urban living spaces, we needed to dampen the sound using foam or another material for layering the reservoir in which the pump is submerged. In addition, when the LED array subsystem is operating it needs to be in a space where it does not disturb anyone that may be nearby as it will operate when the sun has set or the system is indoors. We made a curtain for times when the LED array is programmed to operate.

5.2.3 Responsibly Sourced Materials

More of a global scale concern, our group is aware of the increased violence induced by the need for precious materials such as tantalum, tin and tungsten; and gold (collectively, “3TG”), specifically referred to as "conflict minerals". These minerals are used in electronics manufacturing and are found in areas where mining and sale of said minerals is used to support armed conflicts. Thus it was important ethically for our group to purchase components from companies and suppliers which publicly disclosed information about their material supply chain. For example, any company which publically trades stock on the US stock exchange must submit a supply chain review report to the U.S. Securities and Exchange Commission (SEC), making it simple for our group to check the supply chain of public companies to verify they are taking appropriate measures in reducing the amount of conflict minerals in their electronic components.

5.3 Ethical, Health, and Safety Constraints

Safety ethics and procedure will be upheld for the duration of project construction and systems have been designed with safety as a top priority. This is to ensure all group members designed and built the system and with users of the hydroponics system were kept safe throughout operation.

5.3.1 Electrical Safety

Our system had to be safe not only for the designers, who already had an understanding of the dangers and risks of the electrical systems, but also for anyone who should come in contact with, or try to operate, the hydroponic automation system. A safety mechanism such project boxes prevent the system from have electrical contacts shorted from outside influences (ie. stray metal, water, etc.). Fuses provide a means secondary means of protection in that if a short or fault were to occur, excess current will overheat a conductive filament, terminate the connection. This protects not only the equipment and system, but all any persons who may be exposed to the short.

Fuses were installed in areas of our system where high current is expected (in excess of 2 Amperes), frequent connects and disconnects, and on the input power stage of our designed PCBs. These safety measures were held throughout the build process, not just in the final system, as a measure of protection for our system and ourselves.

5.3.2 Non-Carcinogenic Construction Materials

Leaded solder is a common material used in electronics because of its low cost and relatively low melting point, however lead is a known carcinogen and neurotoxin. Lead-free solder has become more commonly more available as government regulations have limited the amount of leaded solder allowed in commercial products. Polyvinyl Chloride (PVC) is a mostly stable building material, used in plumbing for its cost and durability. When exposed to fire or combusted though, it produces dioxin, reported as the most dangerous carcinogen. Thus our group will use lead-free solder for all fabrication of our sub-systems and keep all

PVC away or protected from sources of fire to ensure no one's health is affected during project construction.

5.3.3 Safe Working Environment

Use of power tools and solder rework stations were necessary for this project, which have the potential of harm or hazard to the health of our group members. Caution was exercised when using these tools, no broken, defective or greasy tools were handled. Tools were used for their intended purposes only, gloves and safety glasses were worn at all times when using power tools. Machine guards were engaged whenever power tools are not in use to prevent accidental activation. A fire extinguisher was always be with 15 yards of work station.

5.4 Manufacturability and Sustainability Constraints

Our design was limited to the technology commercially available to us as we do not possess advanced prototyping tools. This property of manufacturability is of equal, if not greater, value as cost, for if some of our parts were not viably synthesized on our current budget, we could not integrate it within our project. Generally, the difficulty or time to manufacture is the basis for the cost of a component. Sustainability of materials used was also a cost factor, where more recyclable or easily produced materials are cheaper than those with longer production cycles or more refinement processes. It's the difference between making a product out of plastic, generally recyclable and can be produced from crop bi-products, and steel, whose mineral must be extracted, refined, and transported on heavy machinery. Thus our project contains sustainable construction materials, such as wood and plastic, over metals unless necessary for a function sustainable materials cannot fulfill.

6.0 Project Hardware and Software Design Details

The following section outlines the design of our project. The design details will start with a high level abstraction, using block diagrams to explain as a visual guide. From there, each subsystem's design will be detailed to gain a clearer understanding of our vision for the project's workings. Individual subsections will contain a write-up of the design philosophy as well as supporting figures, such as a more in-depth block diagram than the overall hardware and software block diagrams, schematics used in making the design, or flowcharts for the behavior of a system and its logic.

6.1 Initial Design Architectures and Related Diagrams

This section contains a high level look at our project in the form of block diagrams. These diagrams illustrate the various systems and how they are connected and how they interact with each other, indicating which systems can read or write from each other or which systems gain power from others.

6.1.1 Hardware Block Diagram and Description

Figure 6.1 represents the overall block diagram for the hardware portion of our project. Ultimately, the solar panels provide power for the entire system. They are connected to a charge controller which will control the rate of charging for our battery bank, and will stop charging when the appropriate charge level is reached. The next block is the battery bank, which allows our system to store charge for periods where there is not enough sunlight. The battery bank is then connected to both a relay power switch and a DC/DC voltage converter board. The relay power switch is controlled by the Tiva C MCU in order to control when various subsystems are powered. The first subsystem controlled by the relay power switch is the LED light array. This array makes sure that the plants have adequate lighting during either the night, indoors, or on a cloudy day. The second subsystem controlled by the relay power switch is the water pump. The water pump makes sure that fresh water gets pumped into the system as needed, using a combination of solenoids that activate and deactivate to control the direction of the water flow. The DC/DC voltage converter board converts the voltage from the battery bank into usable voltage for both the LCD touchscreen and the Tiva C microcontroller board.

The main component of the system is the Tiva C microcontroller. This interfaces with the majority of the hardware on our system. It takes and records meaningful readings from the various sensors in our system. It controls systems based on these readings and user input such as the LED light array, the water pump, and the solenoids. The software inside the microcontroller controls the LCD touchscreen and takes inputs and controls from a user through the interface. The MCU also uses the LCD interface to display sensor readings and thresholds.

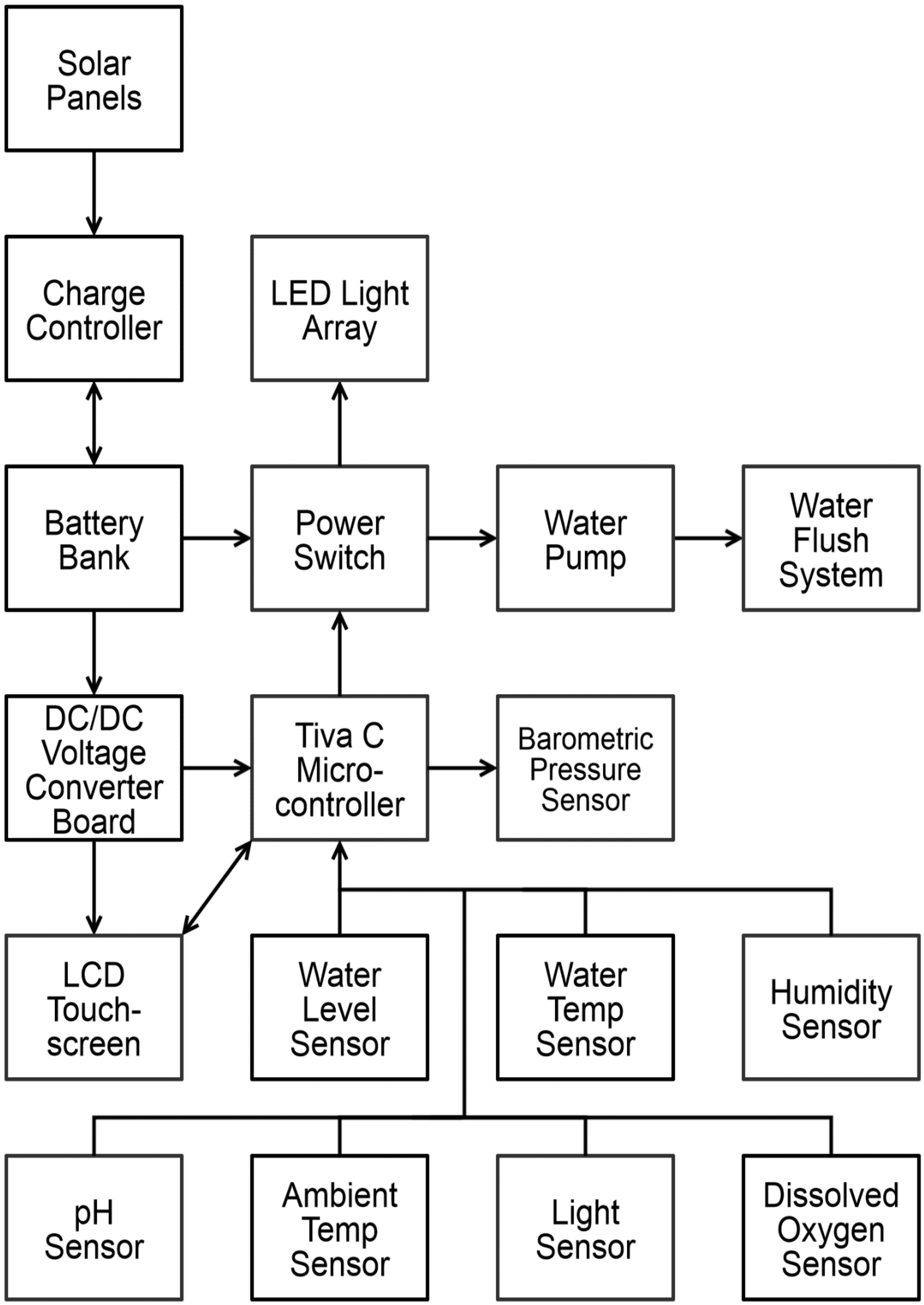


Figure 6.1: Overall Hardware Block Diagram

6.1.2 Software Block Diagram and Description

Figure 6.2 represents the overall software block diagram of the hydroponics system. The MCU main program is the primary block that controls the majority of the system. The relay power switch algorithm allows the MCU to directly control the power relay for the light array, water pump, and solenoids to control the direction of water flow. The sensor reading algorithm reads the sensor data through their respective communication protocols (UART, I^2C, etc) using interrupts for periodic readings or user initiated readings. A sensor threshold alert, powered by interrupts, alerts the user via the LCD UI when a particular measurement has gone above or below a maximum or minimum threshold, respectively. The MCU, in combination with an LCD driver and the TivaWare grlib graphics library, interfaces with a graphical interface powered by user driven interrupts.

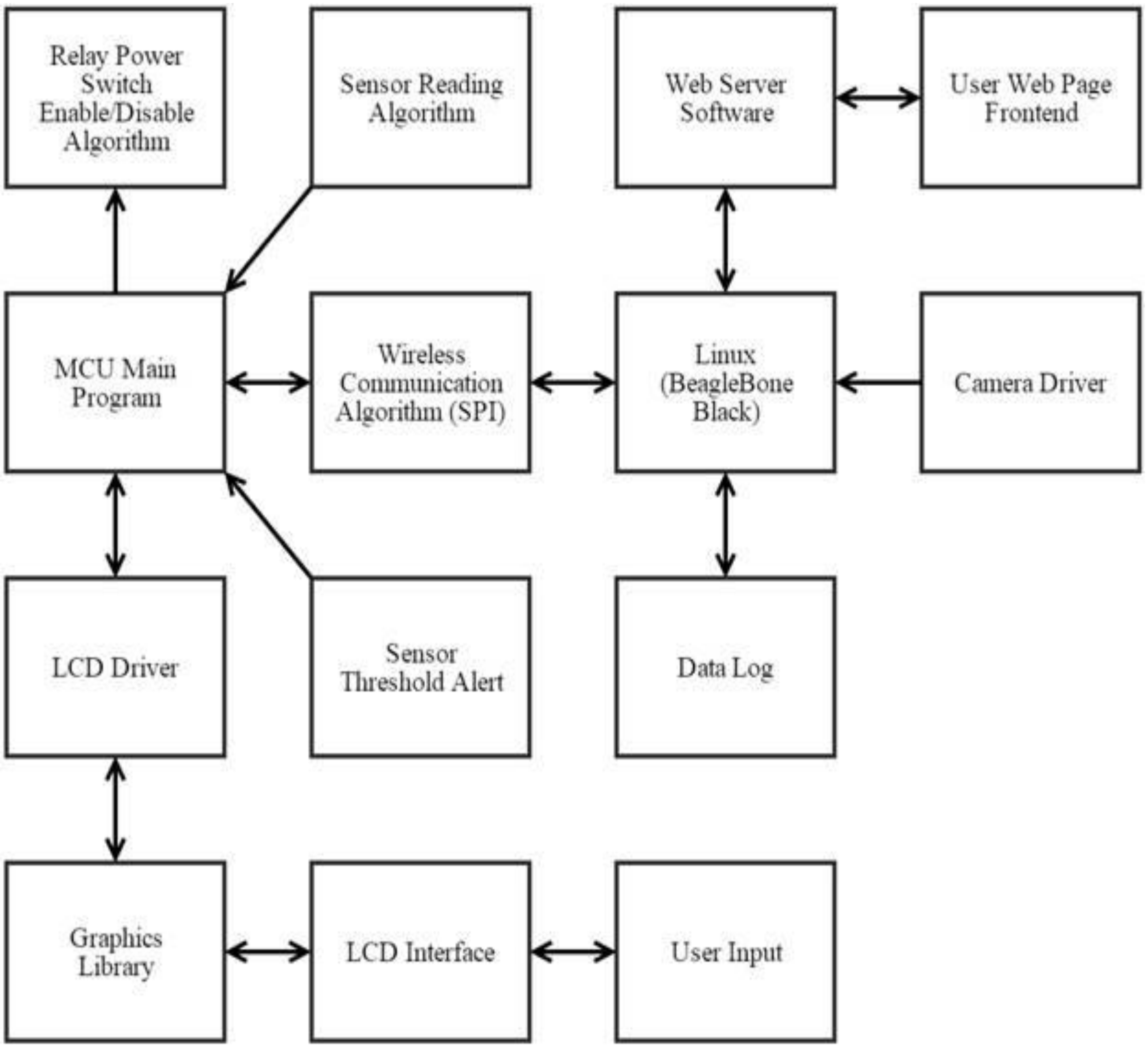


Figure 6.2: Overall Software Block Diagram

6.2 Hydroponics System

Our group is using an NFT hydroponics system prototype which utilizes black drainage pipe for the water channels and a pine bookcase for the structure. The water reservoir is a 5 gallon bucket which the water pump submerges in. As water is pumped to the top of the system, it flows through the water channels over plant roots, and ultimately drains to the reservoir where the cycle repeats. A picture of the prototyped system is included in Figure 6.3.

Figure 6.3: Hydroponic System Prototype

One such material is aluminum extrusion coated in a chemically inert layer. Regular aluminum extrusion can be found in home hardware stores for rain gutters and chemically inert paint for coating the aluminum can be found online or a cheaper alternative can be utilized from a local hardware store. If we were to go with an alternative, it's crucial that the paint be non-water soluble or reactive with nitrates, as they compose most of nutrient solute. Some testing would be necessary to determine the performance and longevity of these paints, going beyond the scope of our project.

6.3 Power Subsystems

The following subsections documents the power subsystems of the hydroponics system. The design details will be described, as well as the interfacing and implementation of the subsystems that are integrated into the hydroponics system. Design requirements and specifications are outlined and documented within the following subsections for each power subsystem.

6.3.1 Solar Panel Integration

The hydroponics system is utilizing a 50W solar panel subsystem loaned from the University of Central Florida Senior Design Lab. In order to properly integrate the solar panel subsystem, the solar panels are mounted on a wooden platform, and orientated in the best direction for maximum sun exposure. After the solar panel was mounted, proper gauge wires of at least 16 AWG were run directly to the charge controller which will be the shortest possible distance away from the solar panel subsystem. A solar panel charge controller was used to regulate and maintain safe charging of the battery from the solar panel. This charge controller is the interfacing medium between the solar panel and the battery.

6.3.2 Battery Power

The hydroponics system requires a battery for the storage of solar power charge to be used both during solar panel discharge (charging of battery), as well as night time or overcast daytime when the solar panels are not outputting power to the battery bank. For the purposes of the hydroponic system's power consumption requirements, a 12V 89Ah battery is used to power the entire system. This capacity of battery is more than sufficient to power the entire hydroponics system for full operation for a minimum of 2 days without charging from the solar panels subsystem. The battery is connected to the charge controller on the input side for balanced and regulated charging of the battery. On the output side of the battery, certain components, such as the water pump and LED array are directly connected. The DC/DC voltage regulator is also connected, and used to distribute power to the rest of the system.

6.3.3 Voltage Regulation

The project requires a couple of voltage drops in order to operate properly. First, a 12V to 5V drop and a further drop from 5V to 3.3V. In order to design the complete circuit, two individual circuits were designed so problems can be isolated if any are encountered. The first circuit, shown in Figure 6.4, is powered by the Texas Instruments TPS563200 IC. The circuit design is based on TI's reference design is setup using TI's recommended components and component values. The GND, VIN, and EN pins are the ground, input voltage, and enable pins respectively. The VFB represents the feedback voltage, which controls V_{OUT} with the following equation:

$$V_{OUT} = 0.765 * \left(1 + \frac{R2}{R3}\right)$$

Where R2 and R3 represent the top and bottom resistor, respectively, of the resistor divider network attached to VFB (Rfbt and Rfbb in Figure 6.4). VBST powers an internal NFET drive circuit. The SW pin is to connect the internal switching circuitry, which is powered by PWM, to the circuit's inductor, L1. According to the datasheet, a capacitor, Cbst, should connect the SW and Vbst pins and have a 100 nF capacitor in between. Cin as well as Cout are used to prevent sudden changes in voltage. L1 stores the charge necessary for switching voltage regulators (explained in section 3.2.9.3). Rfbt and Rfbb control V_{OUT}. The values were picked in accordance with the datasheet. Using the above equation, V_{OUT} = 5.0643V which is very close to the 5V necessary. This circuit is also capable of providing up to 3A, which leaves plenty of room for our project's current needs.

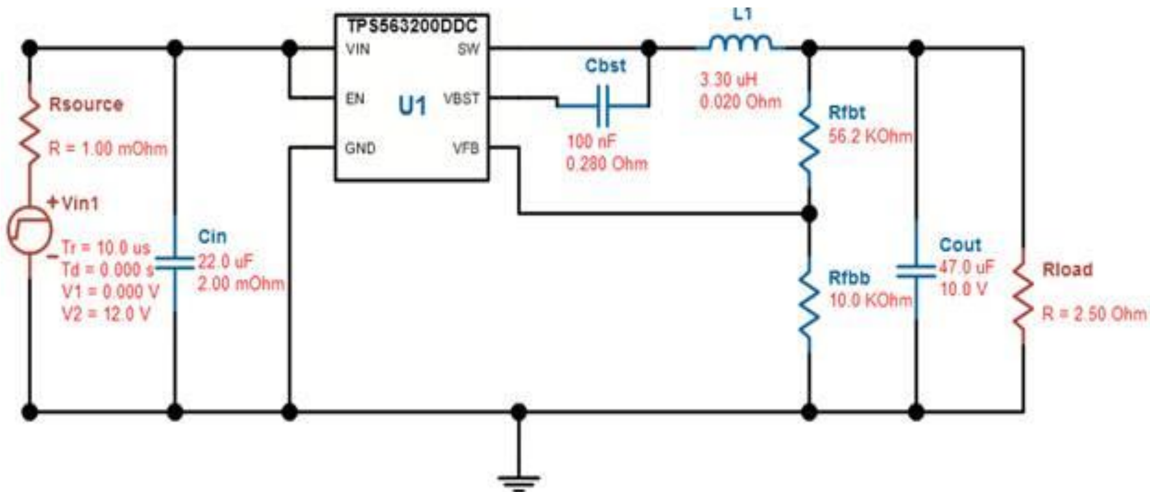


Figure 6.4: 12V-to-5V Switching Regulator Circuit

The second circuit takes the 5V output of the first regulator circuit as an input and then outputs 3.3V for use with the MCU. It is controlled by the LM2854 IC, also by Texas Instruments. The circuit design is based on TI's reference design is setup using TI's recommended components and component values. The PVIN and AVIN pins take the input voltage. AVIN is the analog voltage input and PVIN is used to power the internal power switches of the IC. PGND and SGND are ground for the power switches and signal ground, respectively. EN is the enable pin. SW is the switching output of the IC. SS is a soft-start pin, which the capacitor, Css, affects at a rate of 2.5 nF/ms. In the circuit below, the startup time should be around 1.08 ms. Finally, FB is the feedback pin which controls the output voltage of the circuit with the following equation:

$$V_{OUT} = 0.8 * \left(\frac{Rfb1 + Rfb2}{Rfb1}\right)$$

Using the values in the circuit below (Rfb1 = 57.6k, Rfb2 = 180k), the Vout comes to exactly 3.3V. Figure 6.5 represents the 5V-to-3.3V circuit:

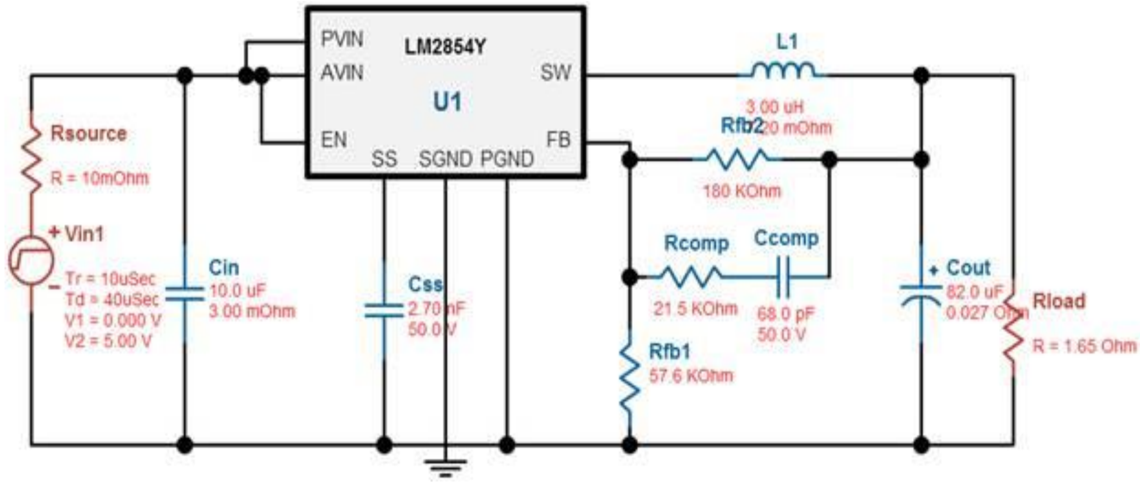


Figure 6.5: 5V-to-3.3V Switching Regulator Circuit

From here, the circuit can be connected to the MCU's VDD and VDDA lines, which require 3.3V for operation, as well as any other components that require 3.3V input. Simulations for this design are presented in Section 8.2.1.4.

6.3.4 Light, Water Pump, and Tiller Power Control

The automation subsystems is modulated by an array of relay elements. We had the choice of physical contact relays, which operate by having an electromagnet close and open a micro switch, and using a solid state relay, operating much like an optical isolator but with two states (on/off). Taking cost, reliability, performance and footprint into consideration, our group decided to implement solid state relays because of their longer lifetimes and lower voltage/current requirements to activate the device. The relay array circuit will take the control signal from the MCU board, interpret the signal using onboard logic, subsequently activating/deactivating the proper automation subsystems. This is depicted in Figure 6.6 below. Our group chose to use an 8-bit MCU as a decoder because it would make the design simpler for adding more automation subsystems after the project.

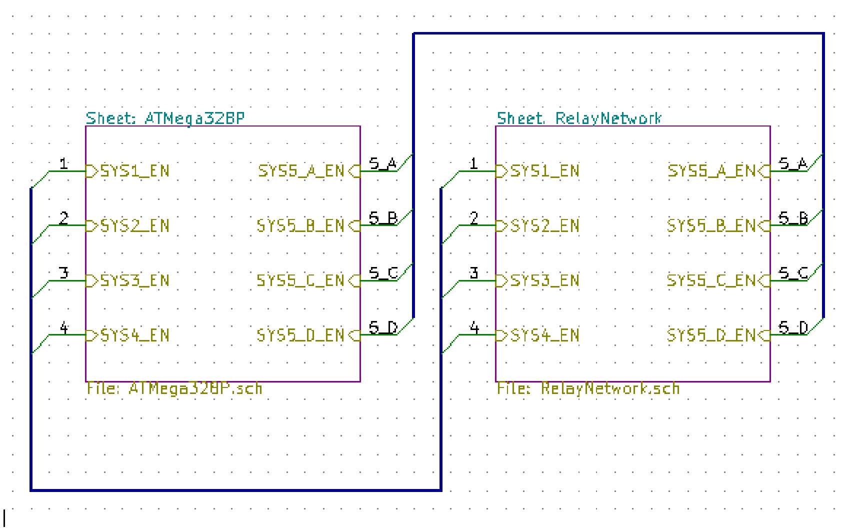


Figure 6.6: Schematic for Automation Sub-systems controller

With the decoder and solid state relays, there are two levels of electrical isolation from the higher voltages and currents of the automation subsystems. These systems are treated as low impedance resistors in the schematic because they are expected to draw ~2A of current at battery voltage (~12VDC). Our group is confident in designing such a circuit with emphasis on isolation and control because of the protection, control, and scalability it affords our project while using a minimal amount of hardware to function.

6.4 Monitoring Subsystem

The following monitoring subsystems encompasses the components being used in the hydroponics system, and the process of interfacing with the remaining system or subsystems, as well as the type of data provided to the system for integration and reporting.

6.4.1 Sensors

In the following subsections the sensors will be discussed involving analysis of integration, hardware interfacing, and other hardware constraints relating to the sensors within the hydroponics system. The types of connectors, wire, and mechanical positioning of the sensors being used will be mentioned.

6.4.1.1 Light Sensor

The ambient light sensor, Intersil ISL29023, being used in the hydroponics system will be implemented directly onto the Tiva C Sensor Hub. The ambient temperature sensor will be connected on board, directly into the I^2C input pins on the microcontroller unit board through the traces laid out on the board. By doing this, there is no need for wires to interface the ambient temperature sensor to the microcontroller board. This ambient light sensor will detect the level of ambient light relative to the microcontroller unit board. Therefore the microcontroller unit board is housed and displayed in direct sunlight to properly measure the levels of sunlight that the hydroponics system is exposed to during the operational time frame.

6.4.1.2 pH Sensor

The pH sensor, Atlas Scientific EZO class embedded pH circuit, being used in the hydroponics system is connected to the microcontroller board by interfacing with the I^2C pins on the microcontroller unit. The pH sensor probe is submerged into the water solution of the hydroponics system, orientated in the middle of the water solution in order to get an average measurement of pH throughout the hydroponics system. The data and power wires for the pH probe are connected to the interfacing board provided by the manufacturers of the pH sensor, Atlas Scientific, as shown in Figure 6.7. The interfacing board is connected directly to the microcontroller unit by 24 AWG wires, for both power and data.

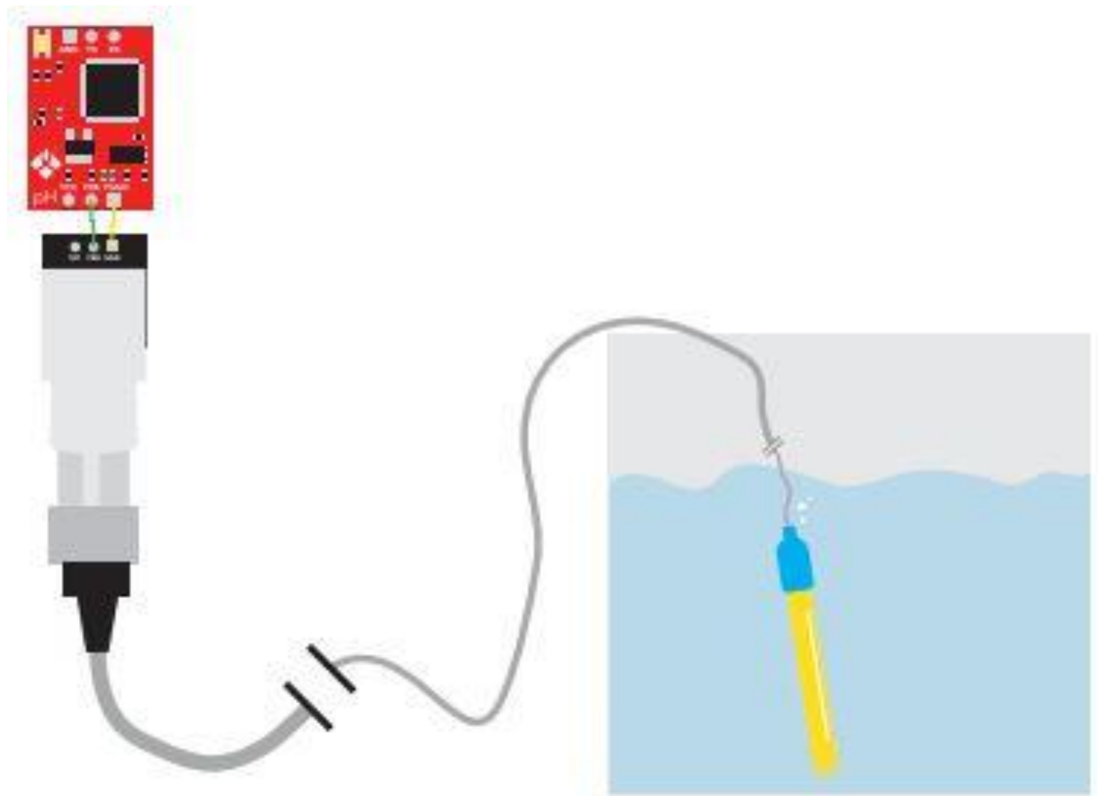


Figure 6.7: Spatial Layout of pH Sesnor

(Permission Pending)

6.4.1.3 Water Level Sensor

The water level sensor, eTape PN-12110215TC-12, being used in the hydroponics system is connected to the microcontroller board by interfacing with the analog pins on the microcontroller unit. The water level sensor is submerged into the water solution of the hydroponics system, orientated in the middle of the water solution in order to get an average measurement of the water level throughout the hydroponics system. The data wires is connected to the microcontroller board by using 24 AWG wire, connected to pin headers, which will connect directly to the microcontroller board.

6.4.1.4 Ambient Humidity Sensor

The ambient humidity sensor, Sensirion SHT21, being used in the hydroponics system is implemented directly onto the Tiva C Sensor Hub. The ambient humidity sensor is connected on board, directly into the I^2C input pins on the microcontroller unit board through the traces laid out on the board. By doing this, there is no need for wires to interface the ambient temperature sensor to the microcontroller board. This ambient humidity sensor detects the level of ambient humidity relative to the microcontroller unit board. Therefore the microcontroller unit board is housed and displayed in direct sunlight to properly measure the temperature levels that the hydroponics system is exposed to during the operational time frame.

6.4.1.5 Ambient Temperature Sensor

The ambient temperature sensor, Texas Instruments TMP006, being used in the hydroponics system is implemented directly onto the Tiva C Sensor Hub. The ambient temperature sensor is connected on board, directly into the I^2C input pins on the microcontroller unit board through the traces laid out on the board. By doing this, there is no need for wires to interface the ambient temperature sensor to the microcontroller board. This ambient temperature sensor detects the level of ambient temperature relative to the microcontroller unit board. Therefore the microcontroller unit board is housed and displayed in direct sunlight to properly measure the temperature levels that the hydroponics system is exposed to during the operational time frame.

6.4.1.6 Water Temperature Sensor

The water temperature sensor, Maxim DS18B20, being used in the hydroponics system is connected to the microcontroller board by interfacing with the I^2C pins on the microcontroller unit. The water temperature sensor is submerged into the water solution of the hydroponics system, orientated in the middle of the water solution in order to get an average measurement of the water temperature throughout the hydroponics system. The data wires are connected to the microcontroller board by using the stock wire connected to the water temperature probe, connected to pin headers, which are connected directly to the microcontroller board.

6.4.1.7 Dissolved Oxygen Sensor

The dissolved oxygen sensor, Atlas Scientific EZO class embedded Dissolved Oxygen circuit, being used in the hydroponics system is connected to the microcontroller board by interfacing with the I^2C pins on the microcontroller unit. The dissolved oxygen sensor probe is submerged into the water solution of the hydroponics system, orientated in the middle of the water solution in order to get an average measurement of dissolved oxygen throughout the hydroponics system. The data and power wires for the dissolved oxygen probe are connected to the interfacing board provided by the manufacturers of the dissolved oxygen sensor, as shown in Figure 6.7. The interfacing board is connected directly to the microcontroller unit by 24 AWG wires, for both power and data.

6.4.2 Microcontroller Integration for Sensor Data

For the integration of microcontroller sensor data, the sensors are all connected to the microcontroller unit by their respective pins. The majority of sensors used in the hydroponics system utilize I^2C. This allows the sensors to share the same I^2C bus, so that the sensors can share the same pin on the board, and the data will be separated by address for interpretation by the microcontroller unit. The data is parsed for use by the microcontroller and output to the appropriate subsystems. The analog sensors are interfaced and connected to the analog pin headers on the microcontroller board. The analog output is analyzed by the microcontroller and output to the appropriate subsystems. All of the sensor data is read at the appropriate intervals and if necessary averaged over time to get a more stable reading of the sensor data.

6.5 Automation Subsystems

The automation subsystems actuate the processes necessary to maintain the hydroponic setup: water circulation, lighting, and emergency water flush. These are addressed via the light array, water pump, and solenoids comprising the water flush system; modulated by a control signal from the MCU to activate solid-state relays to power these subsystems.

6.5.1 Light Array

Our group discussed amongst ourselves and with previous groups on the subject of whether it would be better to build the LED array ourselves or buy and modify an existing product for our needs. On the basis of cost, the options are of roughly equal standing if made with the same quality of materials. On the subject of design competence however, our group feels it would be more appreciable and worthwhile to reverse-engineer an existing product. If we were to build an LED array ourselves, it would be designed to operate directly from the DC power stored in the battery bank and such a design would only include parallel circuits consisting of a power rail, diodes in series, a bias resistor, and a ground rail. It would be more beneficial on the basis of time and education to modify an existing LED array, observing their internal AC power regulators for future projects.

6.5.2 Water Pump

For our power specifications, we sourced a water pump intended for use as a boat bilge pump. This product is ideal for our project because bilge pumps are submersible, have adequate water output and head pressure, operate at 12V DC, and are affordable for our budget. We have the water pump submerged in our water reservoir, circulating water through our hydroponics system via the head pressure of the pump and returning by gravity.

6.5.3 Microcontroller Integration for Control Data

The MCU controls these systems via a decoder with connections to solid state relays, this setup is more thoroughly described in section 6.3.4. The purpose of the control data is with *n-many* pins control 2^n-outputs. This improves our design by making it more scalable because we feasibly only need three pins to control four automation systems. This allows us to add more features such as adding more lights or a CO_2 generator. The system would only need one more control pin and four relays to control four additional outputs. For the board design, a minimum of four pins (three for the decode lines, one for the enable line) are necessary to properly operate the decoder, the extra decode line to future proof the design in case of one of the outputs failing or additional automation sub-systems are added.

6.5.4 Water Flush System

Two solenoid valves are installed between the output of the water pump and the top of the hydroponic system: one leads to the top of the system and the second leads out of the system. Normal operation consists of the pump->hydroponics solenoid engaged and the pump->out solenoid disengaged. Both valves are nominally closed. In the event the water becomes contaminated or the level of nutrient is high enough to burn the roots (large pH imbalance), the pump-

>hydroponics solenoid disengages and the pump->out solenoid engages, diverting all water in the reservoir out of the system. Similarly, a third solenoid valve may be added as a modular feature to employ a water refilling feature from a clean water source.

6.6 User Controlled Subsystem

The user controlled subsystem consists of an LCD touchscreen device attached to the MCU in order to access the hydroponics system remotely for convenience. The LCD touchscreen is directly connected to the MCU through the PCB traces and allows complete view of the sensor data as well as control over the system components.

6.6.1 LCD Screen

The LCD screen is a user controllable screen. It not only outputs relevant sensor data, but also take inputs for sensor thresholds and control actions from a touchscreen interface. The sensor data updates around every 5 minutes or so. However, the user can choose to update the data immediately if necessary via a refresh button on the screen. The user can also control various components of the system, such as the water pump, light array, and flush system. The LCD screen is directly connected to the PCB of the MCU and gets power from the same PCB. The following table, Table 6.1, lists the pins that the LCD screen interfaces to on our TM4C1294NCPDT microcontroller:

LCD Pin Name	LCD Pin Description	TM4C1294NCPDT Pin Description	TM4C1294NCPDT Pin Number
3.3 V	3.3 V Power Supply	N/A	N/A
LCD_D5	LCD Data bit 5	GPIO Port D Bit 2	PD2 (3)
LCD_D0	LCD Data bit 0	GPIO Port P Bit 0	PP0 (118)
LCD_D1	LCD Data bit 1	GPIO Port P Bit 1	PP1 (119)
TOUCH_XP	Resistor Touch (Left)	GPIO Port D Bit 4	PD4 (125)
TOUCH_YP	Resistor Touch (Top)	GPIO Port D Bit 5	PD5 (126)
LCD_D4	LCD Data bit 4	GPIO Port Q Bit 0	PQ0 (5)
LCD_WR	LCD Write Control	GPIO Port P Bit 4	PP4 (105)
LCD_RS	LCD Reg/Data Select	GPIO Port N Bit 5	PN5 (112)
LCD_CS	LCD Chip Select	GPIO Port N Bit 4	PN4 (111)
GND	Ground	N/A	N/A
LCD_D2	LCD Data bit 2	GPIO Port M Bit 7	PM7 (71)
RESET	RESET	RESET	N/A
LCD_D7	LCD Data bit 7	GPIO Port Q Bit 2	PQ2 (11)
LCD_D6	LCD Data bit 6	GPIO Port Q Bit 3	PQ3 (27)
LCD_RD	LCD Read control	GPIO Port P Bit 3	PP3 (104)
TOUCH_XN	Resistor Touch (Right)	GPIO Port Q Bit 1	PQ1 (6)
TOUCH_YN	Resistor Touch (Bottom)	GPIO Port M Bit 6	PM6 (72)
5V0	5 V Power Supply	N/A	N/A
GND	Ground	N/A	N/A
LCD_BL	LCD Backlight On/Off	GPIO Port G Bit 1	PG1 (50)
LCD_D3	LCD Data bit 3	GPIO Port K Bit 5	PK5 (62)

Table 6.1: LCD Touchscreen Pin Configuration

Table 6.1 represents the hardware configuration for how the LCD screen interfaces with the MCU. The LCD screen operates using a parallel data stream. 8 bits (LCD data bit 0-7) are written at the same time for faster screen refreshing. The software allows the user to directly interact with the MCU via the touchscreen. The flowchart in Figure 6.9 explains how the software works:

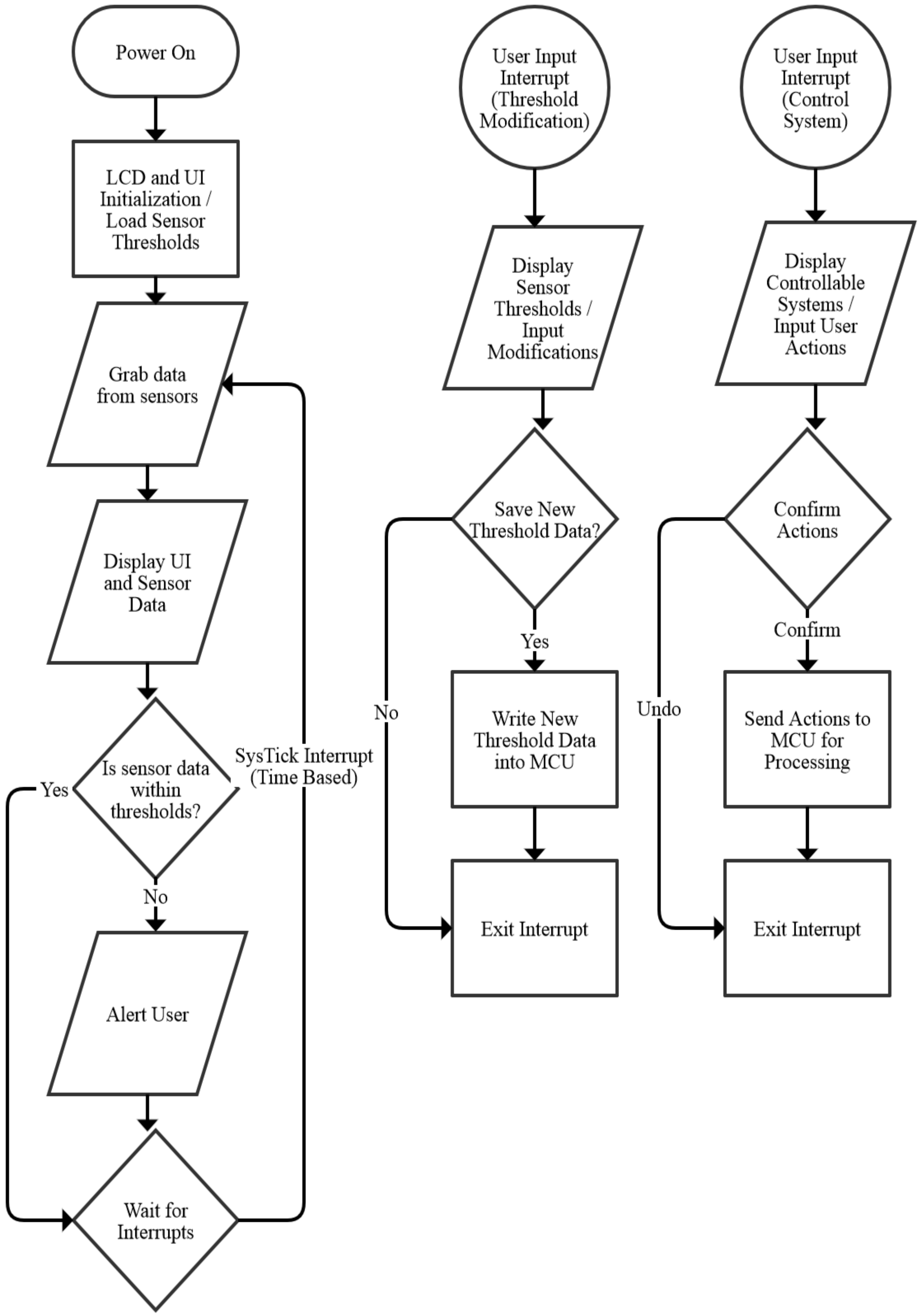


Figure 6.9: LCD Touchscreen Software Flowchart

The software is mainly controlled by interrupts. This allows the processor to operate in a low power mode until interrupted with either user input or periodic sensor readings. It also allows the processor to respond immediately when a user interacts with the touchscreen.

6.7 Software Design Methods

The following section describes the general plan for the software powering our project. Section 6.7.1 will discuss the planned implementation for the overall MCU code. Section 6.7.2 and Section 6.7.3 will go into detail about the web server interface and LCD interface, which are vital functions in the system to ensure user operability as well as meaningful data display and recordkeeping.

6.7.1 MCU Code Design

The code for the MCU program is written in C and compiled using TI's Code Composer Studio version 6.1. Since the MCU is a Texas Instruments product, Code Composer Studio ensures maximum compatibility. The code is designed using a modular format with many sub-functions that can be added or deleted with minimal effect to the main program. This allows our program to add or delete sensors or functionality without having to rewrite a majority of the code. The first step is to load relevant header files. This is important because Texas Instruments provides necessary header files to define various names to port addresses via the TivaWare Peripheral Library. Many of the TM4C1294 pins have many alternate functions, so the first function to run from main should be a function to initialize all of the ports to the proper function. There is a specific procedure to do this, which involves setting various registers in the MCU. The procedure is outlined below:

1. Activate the clock for the port in the SYSCTL (system control) register.
2. Unlock the port using the LOCK and CR (commit register) registers.
3. Set the AMSEL (analog mode select) register to enable or disable analog.
4. Clearing or settings bits in the PCTL (port control) register.
5. Specifying input or output for the port with the DIR (direction) register.
6. Setting or clearing bits in the AFSEL (alternate function select) register to determine whether the port is using a regular or alternate function
7. Setting the PUR (pull-up resistor) or PDR (pull-down resistor) registers depending on if an internal pull-up or pull-down resistor is required.
8. Finally, setting bits in the DEN (digital enable) register to enable digital I/O.

The next set of functions in main initialize the interrupt system. The interrupt system is important for our system to function properly. It allows the program to efficiently read sensors when necessary, either based on a timer or user interaction. Using an interrupt system allows the MCU to wait for an interrupt flag rather than using an infinite loop constantly checking variables with if statements or something similar. Using the SysTick timer, an interrupt can be time based. The BASEPRI register can also be used to set the order of importance for interrupts so that certain functions cannot be interrupted. Following that, various other systems will be initialized. The sensor interfaces (I^2C, UART, etc.), the relay power switch interface which uses GPIO high and low voltage levels, as well as the LCD display driver. Finally, the system loads the pre-defined sensor thresholds and begin to read sensor data. After sensor data has been read, the system remains in a perpetual “WaitForInterrupt” mode which is interrupted by a custom sensor read timer or user interaction with the LCD or webpage.

6.7.2 LCD Interface

The LCD interface was designed using the TivaWare Graphics Library (grlib). The software is written in C using the program Code Composer Studio to compile and LM Flash Programmer to flash to the MCU. The LCD interface is a user friendly color GUI designed to directly interact with the MCU and hydroponics system. Upon power up of the hydroponics system, the LCD program starts as a part of the main hydroponics program built into the MCU itself. If the LCD is unable to communicate with the MCU, it may not be able to display an error message, since communication with the MCU would be required to display it. If there is no functioning LCD interface, then there is either a communication or power problem between the LCD display and the MCU. Since the LCD hardware is a resistive touchscreen, the interface has intuitive buttons that can be directly pressed on the screen in order to access various functions of the UI. By default, and similarly to the web page, the LCD interface starts out with a display of the latest sensor readings as well as any alerts for out of threshold sensor readings. The main display page has on-screen buttons to lead to other UI sections for adjusting sensor threshold values and controlling systems through the MCU. The sensor readings only update periodically to conserve processor resources, however they can be manually refreshed.

6.8 Main Board Design

The following sections outline the final design of the main board, with included schematics and the PCB layout for our final design.

6.8.1 Main Board Schematics

The following figures, Figure 6.10, Figure 6.11, and Figure 6.12 show the circuit layout of the main board that interfaces with the sensors, voltage regulator board, and control board. Figure 6.10 gives the layout of the Tiva C TM4C1294NCPDT chip and the various pin assignments the chip has. Figure 6.11 shows the power distribution of the chip with the board, using various decoupling resistors, crystals for clock cycles, voltage headers, and a reset button for debugging. Finally, Figure 6.12 shows the connections between the pins and the various sensors. Certain sensors require separate pull-up resistors to power the I^2C interface. Also present are headers for the JTAG programming debug pins and a breakout header for extra GPIO pins that were unused but made available just in case.

Figure 6.10: Layout of the majority of the TM4C1294NCPDT pins

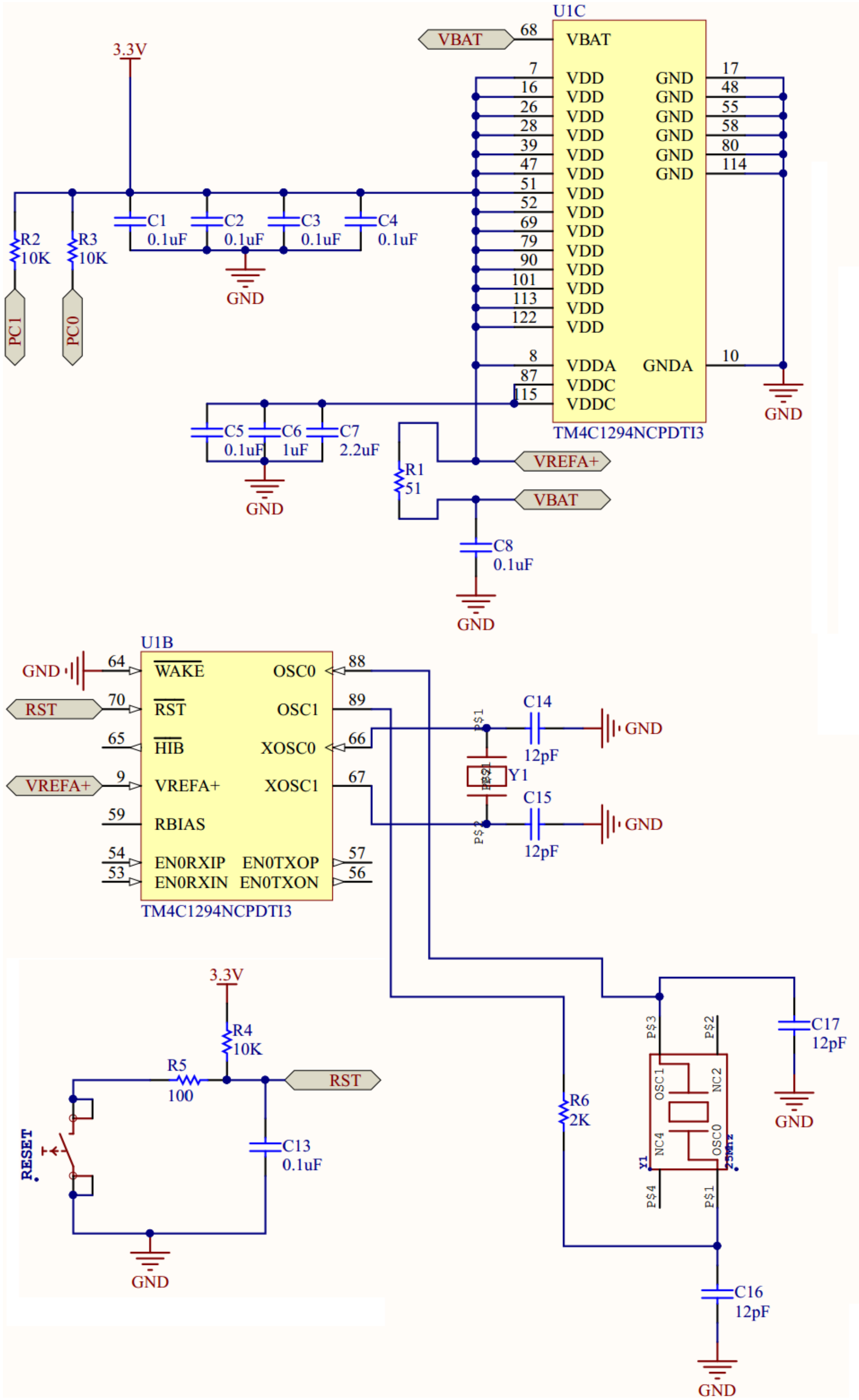


Figure 6.11: Schematic for Power, Crystals, and Reset Button

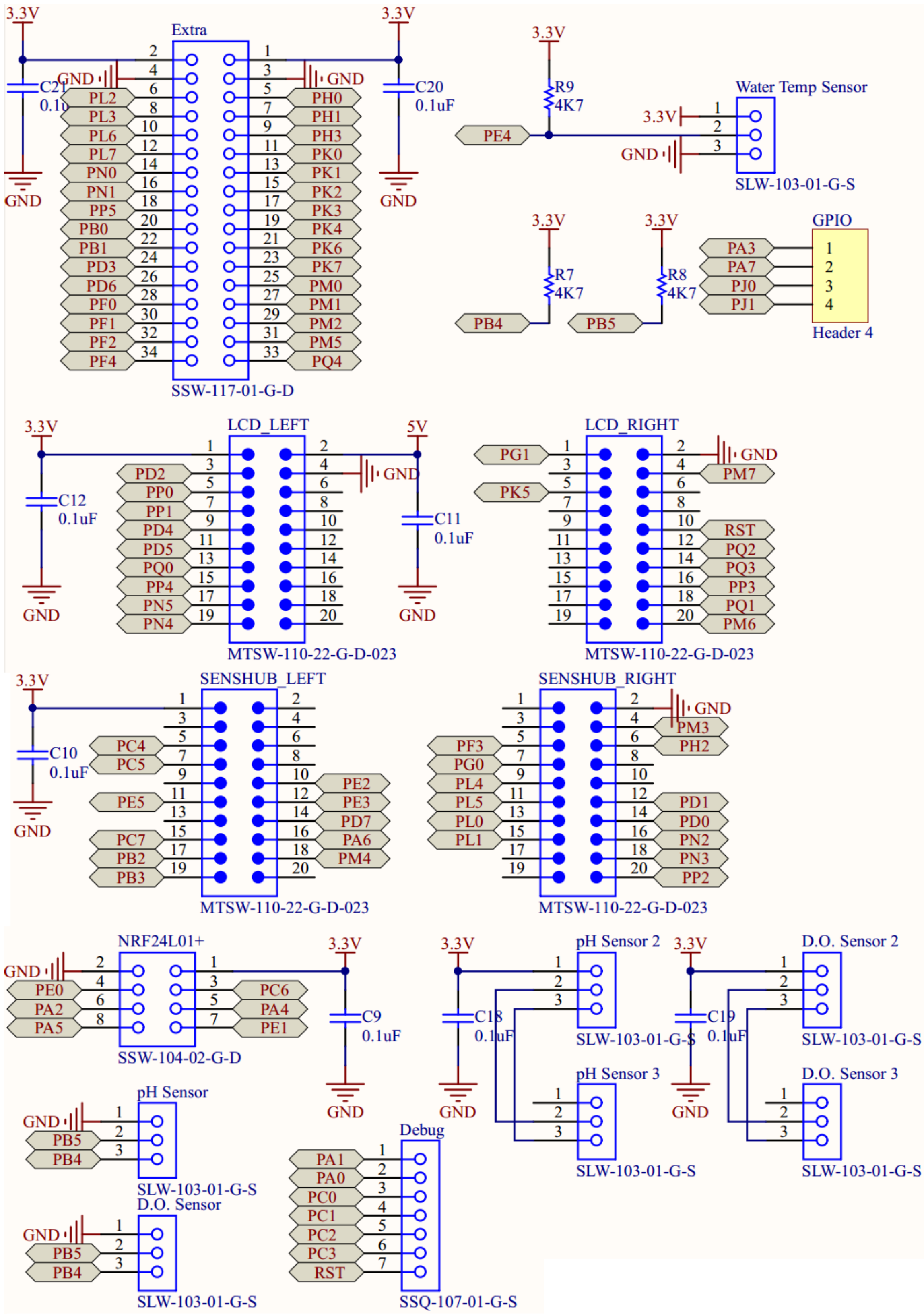


Figure 6.12: Schematic for Sensor, Debug, Screen, and Extra Connections

6.8.2 Main Board PCB Layout

The following figure, Figure 6.13, shows the layout of the main board's PCB. It was designed as a four layer board, with the two center layers being a ground plane and a 3.3 V power plane, and the outer two layers being signal layers that allow the microcontroller to communicate with all of the different peripherals. The left side contains terminal blocks for the sensor hub and LCD screen to interface with, and the right side contains headers for power, external sensors, and the GPIO pins to interface with the control board to activate the relays.

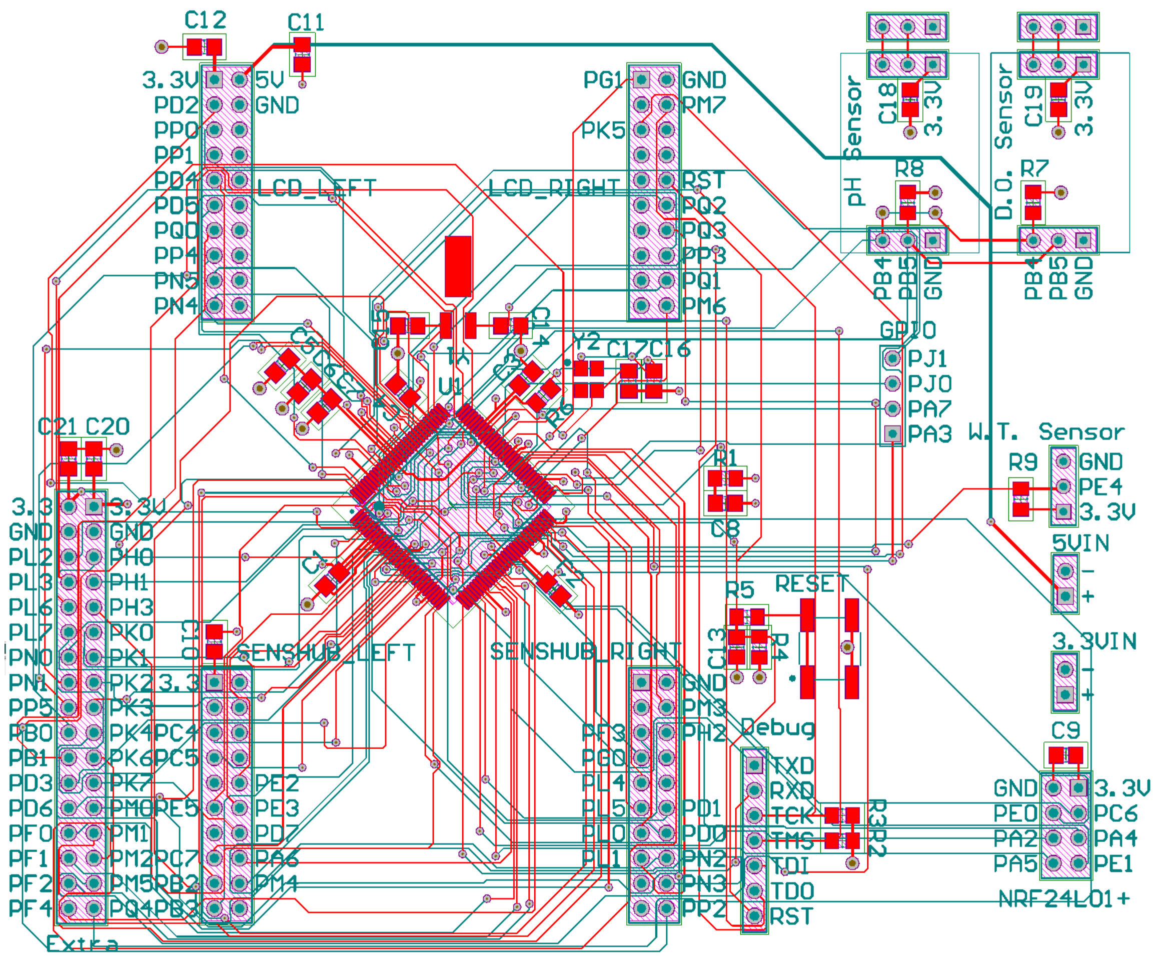


Figure 6.13: PCB Layout of the Main Board

7.0 Project Prototype Construction and Coding

The following sections and subsections will encompass the hydroponics system prototyping construction and coding, procedure and development environments for prototyping the hydroponics system and the relevant subsystems. The construction, trouble shooting, coding, and execution of prototype testing methods for each of the subsystems will be documented in the following subsections.

7.1 Parts Acquisition and BOM

The bill of materials for the critical parts utilized in the hydroponics system are laid out in Table 7.1 with the most up-to-date prices for each part. The minor parts such as, wire, connectors, etc. will not be included in the bill of materials due to lack of certainty of quantity of such minor parts. The bill of materials will however cover all critical parts for the hydroponics system to operate as designed throughout this document.

Device	Device Type	Number of Units	Price per Unit	Description/Purpose
Atlas Scientific - pH Sensor	Sensor	1	$135.54	Monitors the pH level of the water solution.
Atlas Scientific - Dissolved Oxygen Sensor	Sensor	1	$231.99	Monitors the dissolved oxygen level of the water solution.
Milone Technologies - eTape (12 inchs)	Sensor	1	$47.91	Monitors the water level of the water solution.
Maxim DS18B20 - Water Temperature Sensor	Sensor	1	$9.95	Monitors the temperature of the water solution.
Sensirion SHT21 - Ambient Temperature Sensor	Sensor	1	Free Sample	Monitors the ambient temperature of the surrounding air.
Intersil ISL29023 - Ambient Light Sensor	Sensor	1	Free Sample	Monitors the ambient light of the surroundings.
Trojan 27-AGM Battery	Battery	1	Free	Battery storage for power distribution.
Texas Instruments LM2854 - Voltage Buck Regulator	Voltage Regulator	1	Free Sample	Converts voltage from 5V to 3.3 V
Texas Instruments TPS563200 - Voltage Buck Regulator	Voltage Regulator	1	Free Sample	Converts voltage from 12V to 5V
Texas Instruments TM4C1294NCPDT	Micro-Controller	1	Free Sample	Handles the automation, sensor readings,

Device	Device Type	Number of Units	Price per Unit	Description/Purpose
Kentec EB-LM4F120-L35 - 3.5" LCD Touchscreen	LCD Touch-screen	1	$41.19	Handles the local user interaction. Displays sensor readings and alerts.
Various Inductors, Capacitors, and Resistors	Passive circuit devices	Multiple	$52.22	Voltage regulator circuit parts.
Veewon Wires	Wires	120	$9.99	Wires for connecting various parts.
Bilge Pump	Water Pump	1	Free	Water pump for automation subsystem
Solid-State Relays	Active Devices	4	$4.99	Automation controller
LED Bulbs	Light	4	$8.00	Light Array
Solenoids	Active Devices	2	$12.99	Water Flush sub-system
3/4 in Hosing	Hardware	4	$3.95	Pump and solenoid connections
Main Board PCB	PCB	3	64.60	Main Board PCB
VReg PCB	PCB	3	$3.27	VReg PCB
Battery Connectors	Terminals	Set of 2	$24.29	Connects the battery
Fuses	Fuses	Multiple	$9.20	Prevents shorts
Heat Shrink	Hardware	1	$8.69	Makes wiring better
Exterior 3M Tape	Hardware	1	$8.88	Mounting
		Total Price	**$861.40**	

Table 7.1: Bill of Materials

7.2 PCB Vendor and Assembly

Our group compiled a table comparing the four most attractive or popular PCB vendors we heard of, with focus on capabilities, cost, and processing time. Macrofab has become popular recently as a "one stop" for PCB fabrication, part mounting, and even firmware flashing of onboard chips. Its weakness though is the lack of information on pricing or manufacturing capabilities. The next company, 4PCB has an elaborate range of manufacturing capabilities, the shortest processing time, part mounting services and even an educational discount for students... at a significant cost. If we were to design a larger PCB it would be a more attractive option but will our board sizes projected to use half the max allowable area, there are better alternatives for the price. Table 7.2 tabulates important metrics of each PCB vendor is a convenient format for the reader.

PCB Vendors				
Company	Macrofab	4PCB	OSH Park	TinySine
Location	Houston, TX	USA	Portland, OR	China
Max Layers	Not Specified	4	4	4
Board Size	Not Specified	30 sqin Max.	Variable	10cm x 10cm
Min. Track Width	Not Specified	6mil(0.15mm)	6mil(0.15mm)	6mil(0.15mm)
Min. Spacing	Not Specified	6mil(0.15mm)	6mil(0.15mm)	6mil(0.15mm)
Cost Comparison	1 board/ $30	1 board/ $66	$10/ sqin (3 boards)	10 boards/ $85
Processing Time	3 weeks	2 weeks	3 weeks	4 weeks
Advantages	Parts mounting available	Education Discount, most professional	Great for small runs	Many boards for cheap
Disadvantages	Pricing hidden for most options	Most expensive for service	Purple silkscreen default	Processing time is longer

Table 7.2: Comparison of PCB Vendors

OSH Park has been very popular in the maker movement for its reasonable prices and quality, a viable option for us. TinySine, an operation based out of China, sells boards at very competitive prices and have extremely favorable customer reviews. For average manufacturing ability this is the best option if processing time isn't critical, taking an extra week to week and half to ship with standard shipping.

For assembly of our project boards, more complex designs such as our MCU processor board had QFN packages while we hand soldered all of the components. Devices such as resistors and capacitors were SMD packages large enough to where we used a hand soldering or hot air station. Duplicates of the designed boards and components were ordered for in the event a component is dead or board fails to function due to an identifiable hardware fault.

For the fabrication of our PCB, we decided to both go with OSH Park and to fabricate them on our own.

8.0 Project Prototype Testing
The following sections outline the methods that were used to prototype and test various functions, hardware components, circuit designs, and software components of our system. Depending on the component, it contains testing methods, expected results, and simulations.

8.1 Hardware Test Environment
The following sections describe the methods for testing our hardware prototype. Since we are using a TM4C1294NCPDT MCU, we used the TI EK-TM4C1294XL Connected LaunchPad. The LaunchPad uses the exact same processor, so any hardware or software was programmed directly using the LaunchPad as a prototype platform. The LaunchPad has a built-in in-circuit debug interface (ICDI) that, in conjunction with Code Composer Studio, allowed problems to be found relatively easily. It also has the ability to, with some soldering and headers, be attached to a breadboard for easy access to certain pins on the MCU. The LaunchPad enabled us to rapidly prototype and design our custom PCB layout by knowing in advance which pins used and are using the LaunchPad design as a reference for our PCB.

8.2 Hardware Specific Testing
The following sections and subsections will encompass the hardware testing procedure and development environments for prototyping the hydroponics system and the relevant subsystems. The construction, trouble shooting, and execution of prototype testing methods for each of the subsystems were documented in the following subsections.

8.2.1 Power System Testing
The following sections and subsections will encompass the power system testing procedure and development environments for prototyping the hydroponics system and the relevant subsystems. The systems include the solar panel, which is connected to the battery charger, which is connected to the battery, which is connected to both the power switch relays and DC/DC voltage regulator. In the conclusion of the testing parameters of the following subsections, the subsystems were integrated into the hydroponics system to ensure that all design parameters are consistent in an operational environment.

8.2.1.1 Solar Panel
The solar panel subsystem was tested in the following format using the methods included in within this subsection. The solar panel was mounted on a wooden structure orientated in the optimal direction for maximum exposure to sunlight. After the mounting of this subsystem, a voltage and current monitoring was performed over the duration of at least 24 hours. This enabled the measurement of power output of the solar panel subsystem under normal conditions. The power output of the solar panel subsystem should reasonably equate to the 50W specified by the solar panel subsystem. The solar panel subsystem was then connected to the battery charger subsystem, which was connected to a drained

battery to ensure that the solar panel could properly perform battery charging, as are the design requirements.

8.2.1.2 Battery Charger

The battery charger subsystem was tested in the following format using the methods included within this subsection. The battery charger was connected to a DC power supply, with the same power output characteristics as the solar panel subsystem. The output of the battery charger was measured to ensure that the battery charger is operational as designed and specified. After the battery charger was tested on a DC power supply bench, the battery charger was connected to both the solar panel subsystem and the battery subsystem, to ensure that the operational parameters are consistent. The power input and output was measured with a multimeter, to conclude the operational specifications are met, when used in the battery charger's intended operational environment.

8.2.1.2.1 Battery Test

The battery subsystem was tested in the following format using the methods included within this subsection. The battery was measured with a multimeter to ensure that the voltage and current characteristics are within the specified limits. After the battery is tested, the battery was discharged to ensure that the battery is capable of discharging. Following the discharge procedure, the battery was again tested with a multimeter, to ensure that the battery remains consistent with discharge design specifications. The battery subsystem was then be connected to a commercial battery charging station to ensure that the battery could be charged in optimal conditions. Once the battery was charged, measurements of the battery voltage were again taken, and compared with initial measurements. The battery was then discharged once again, and after this discharged the battery was integrated into the battery charger, solar panel, power switch, and the voltage regulation subsystems. The performance of the battery under these conditions was monitored and recorded to ensure that the battery met all design specifications. When the battery operational measurements were consistent with the entirety of the fore mentioned testing procedures, the battery was confirmed as being able to operate as the intended design requirements.

8.2.1.3 Power Switch

The power switch, or relay switching board, is composed a decoder and a minimum of 4 solid state relays to control the automation sub-systems. Testing begins with reading the output lines of the decoder when voltage is applied to the inputs (S_1, S_2, S_3). If all outputs correspond to the expected values given controlled inputs, the test passes. Iteratively, the relays are evaluated by testing for continuity when a voltage is applied to the internal LED, inducing the photodiode to conduct. If there's continuity, then the internal operational-isolation circuit of the solid state relay is properly functioning. The power switch can then be prepared for a more finalized installation, ensuring the system is safer and more reliable to operate.

8.2.1.4 Voltage Regulation for PCB

Since the voltage regulator parts are surface-mount devices, it is unrealistic to breadboard it. Like any circuit, it is more time effective to begin testing with a software simulation. TI's WEBENCH contains built-in simulation software and can track waveforms for relevant measurements. The most important measurements for the voltage regulator design are Vin, Vout, and Iout. The circuits must be able to properly regulate the output voltage within normal operating conditions. This includes simulation for circuit startup, voltage variations, and steady state. The following figures, Figure 8.1, Figure 8.2, and Figure 8.3, show the result of a startup simulation, a steady-state simulation, and an input voltage transient simulation on the 12V-to-5V circuit (Figure 6.4):

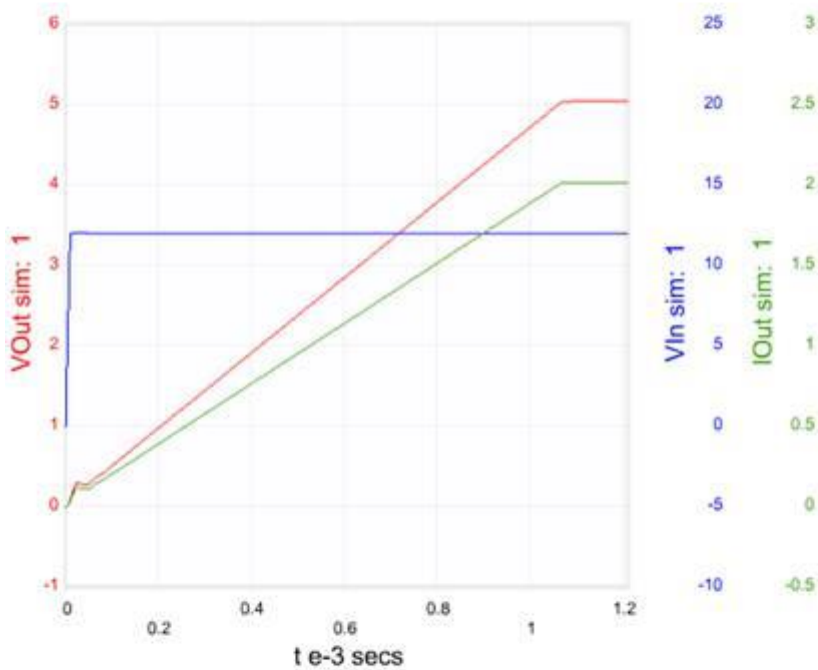


Figure 8.1: 12V-to-5V Regulator Startup Simulation Waveforms

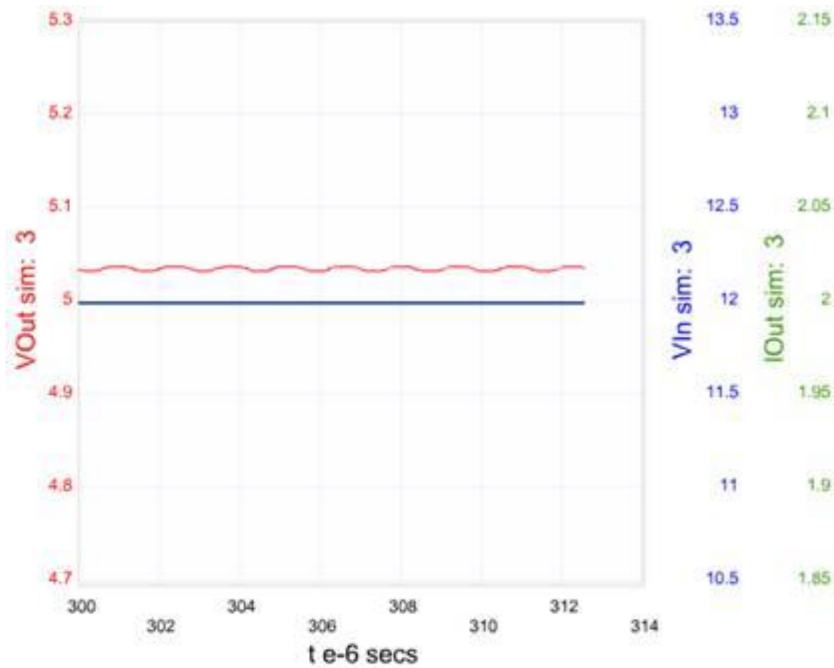


Figure 8.2: 12V-to-5V Regulator Steady-State Simulation Waveforms

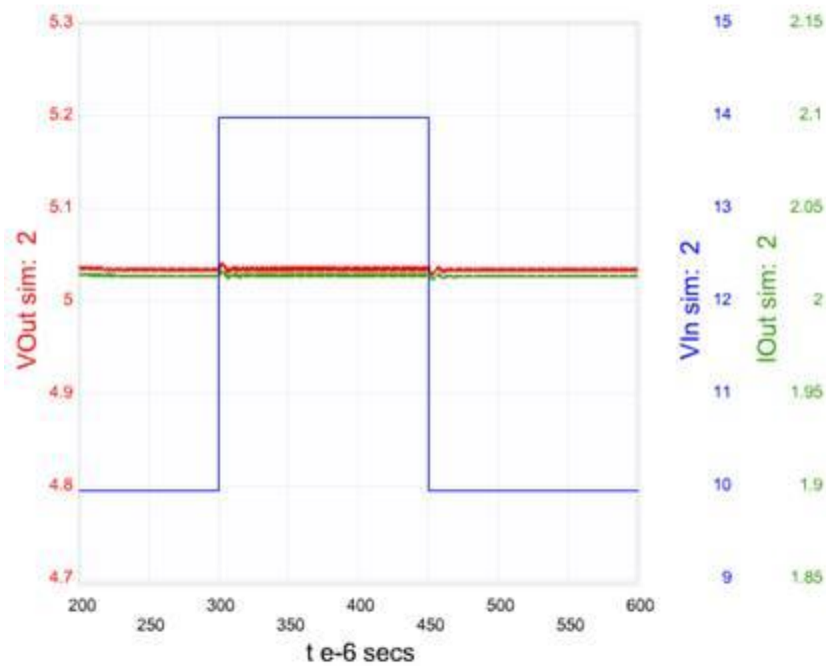


Figure 8.3: 12V-to-5V Regulator Transient Simulation Waveforms

Figure 8.1 shows that the startup simulation passed, taking about 1.05 ms for the output voltage to get to 5V and stay with an input of 12V. The current also stops rising slightly above 2A. The steady-state simulation waveforms in Figure 8.2 show that at a constant 12V input, the output voltage has a very small ripple of about 5mV, which is tolerable. The output current also stays fixed at about 2A. The transient waveforms in Figure 8.3 show the input voltage start at 10V, jump to 14V, and drop back down to 10V. During the jump, the output voltage jumps from about 5.034V to 5.041V which is only about a 7mV difference. During the drop, the output voltage drops from around 5.039V to 5.029V, which is a 10mV drop before steadying out at ~5.034V. These voltage differences are very minor and are not expected to have an impact on the functionality of the circuit.

The next step was to simulate the 5V-to-3.3V circuit to ensure proper functionality. Similar to the previous circuit, the measurements of interest are Vin, Vout, and Iout. Vin was kept at a constant 5V. Since the previous circuit proved to be stable in simulation during a transient input, a transient simulation was not necessary for this circuit. Figures 8.4 and 8.5 show the results of a startup simulation and a steady-state simulation for the 5V-to-3.3V regulator circuit.

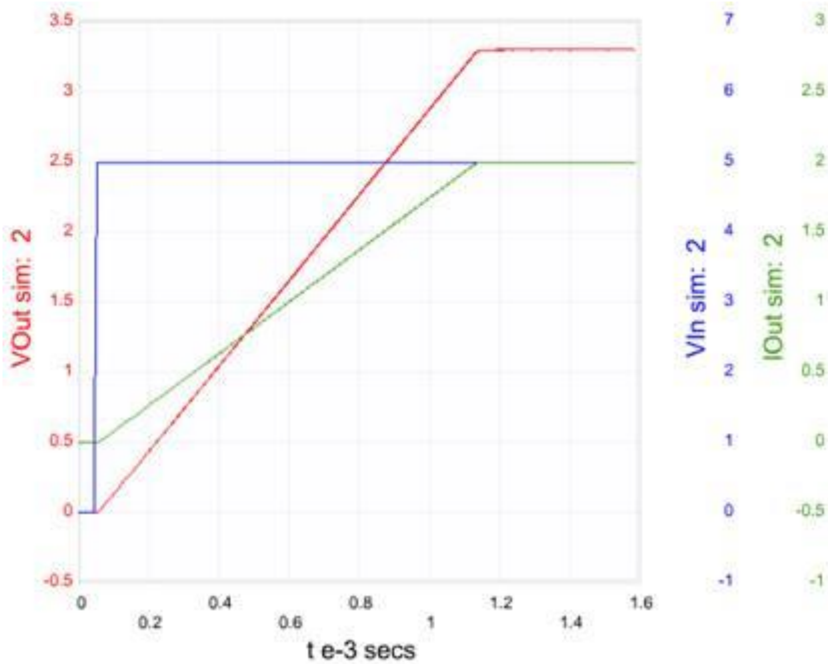


Figure 8.4: 5V-to-3.3V Regulator Startup Simulation Waveforms

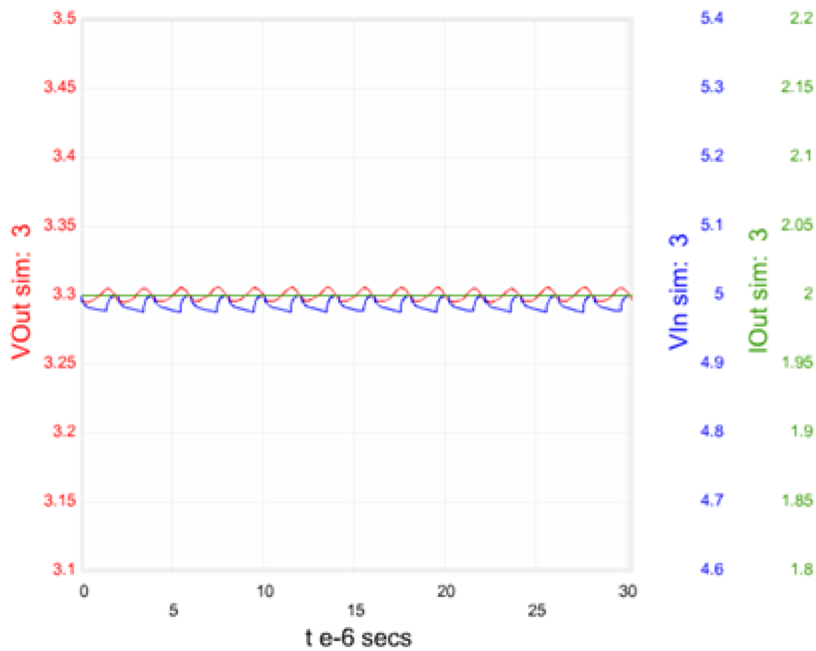


Figure 8.5: 5V-to-3.3V Regulator Steady-State Simulation Waveforms

Figure 8.4 shows that the startup simulation matches the expected output. Since the steady-state capacitor, Css, is 2.70 nF, the startup time should be 1.08 ms, which matches the startup time on the graph. The 5V step input results in a 3.3V output with a current of 2A, which matches the current of the previous 12V-to-5V circuit. The steady-state simulation waveforms shown in Figure 8.5 show a slight variation in the input and output voltage but a steady current output. The output voltage varies from ~3.296V to ~3.304V which is only an 8mV peak-to-peak variation.

The next step in the testing process was to design an isolated voltage regulator PCB board based on this design and to test using a DC power supply that varies between 8V – 17V (17V is the input limit of the TPS563200 IC). Based on the simulation results, however, the circuit was able to handle varying voltages with stability.

8.2.2 Sensor Testing

The following sections and subsections will encompass the sensor testing procedure and development environments for prototyping the hydroponics system and the relevant subsystems. The construction, trouble shooting, and execution of prototype testing methods for each of the subsystems will be documented in the following subsections.

8.2.2.1 Ambient Light Sensor

The ambient light sensor subsystem was tested in the following format using the methods included within this subsection. This subsystem was integrated into the TivaC Sensor Hub. The ambient light sensor was tested by interfacing with the microcontroller unit, to display the output of the of the ambient light sensor data coming from the I^2C data line. The ambient light sensor was covered and the data from the sensor was output to a computer terminal to ensure that the readings are minimal. Once the readings were output correctly and the measurements were confirmed to be minimal relative to the ambient light sensor datasheet. The sensor was then be exposed to bright sunlight, and the data readings were again measured to confirm that the sensor data outputs change accordingly. Once both of these tests are done, the light sensor is confirmed to perform as the design requirements specify.

8.2.2.2 pH Sensor

The pH sensor subsystem was tested in the following format using the methods included within this subsection. This subsystem package included calibration solutions to ensure that the data output readings of the pH sensor are accurate. The pH sensor was calibrated, from the sensor's datasheet specifications. Once the pH sensor was calibrated, it was connected to the interfacing board included in the pH sensor package. The interfacing board was then connected to the I^2C header pins on the microcontroller unit. The pH sensor probe, was submerged into the various control solutions included in the pH sensor package. The output of the sensor data was output to a computer terminal to ensure that the sensor readings were within the datasheet specifications. Once these readings were properly output. The pH sensor was integrated with the entire system, that is, it was submerged into the water solution of the hydroponics system, and once again the data output readings were measured and compared with the expected readings. When the readings are confirmed the pH sensor testing was concluded, and the sensor behaved as the design requirements specify.

8.2.2.3 Water Level Sensor

The water level sensor subsystem was tested in the following format using the methods included within this subsection. The water level sensor was connected to the analog inputs on the microcontroller unit by connecting 24 AWG wire to the analog header pins. The water level sensor was then submerged in measured water quantities. The data output of the sensor was input to the microcontroller unit, the microcontroller was then be set to output to a computer terminal. The measurements of the water level sensor on the computer terminal was compared with the measured water quantities. Once the measurements were consistent with the expected measurements. The water level was varied, and once again the data output will be confirmed. After all the measurements are consistent, the water level sensor was integrated into the hydroponics system. Finally, the measurements of the water level sensor was examined in operational environment, the water level sensor was submerged in the hydroponics system's water solution. The measurements of the water level sensor will be confirmed to be consistent with the previous measurements. Once the water level sensor was confirmed to work within the hydroponics system, the testing of the water level sensor was confirmed to be within the operational parameters of the design specifications.

8.2.2.4 Ambient Temperature Sensor

The ambient temperature sensor subsystem was tested in the following format using the methods included within this subsection. This subsystem was integrated into the Tiva C Sensor Hub. The ambient temperature sensor was tested by interfacing with the microcontroller unit, to display the output of the of the ambient temperature sensor data coming from the I^2C data line. The ambient temperature sensor was surrounded by an ice enclosure and the data from the sensor was output to a computer terminal to ensure that the readings were minimal. Once the readings are output correctly and the measurements were confirmed to be minimal relative to the ambient temperature sensor datasheet. The sensor was then exposed to a low level heat gun, and the data readings were measured again to confirm that the sensor data outputs change accordingly. Once both of these tests were done, the temperature sensor was confirmed to perform as the design requirements specify.

8.2.2.5 Water Temperature Sensor

The water temperature sensor subsystem was tested in the following format using the methods included within this subsection. The water temperature sensor will be connected to the digital inputs on the microcontroller unit by connecting the sensor package wire to the digital header pins. The water temperature sensor was then submerged in an ice water bath. The data output of the sensor was input to the microcontroller unit, the microcontroller was then be set to output to a computer terminal. The measurements of the water temperature sensor on the computer terminal was compared with the physics temperature of 0 degrees Celsius. Once the measurements were consistent with the expected measurements. The water temperature was varied, the water was brought to a low boil. After all the measurements were consistent, the water temperature sensor was integrated into the hydroponics system. Finally, the measurements of the water temperature sensor was examined in operational environment, the water temperature sensor was submerged in the hydroponics system's water solution. The measurements of the water temperature sensor was confirmed to be consistent with the previous measurements. Once the water temperature sensor was confirmed to work within the hydroponics system, the testing of the water temperature sensor was confirmed to be within the operational parameters of the design specifications.

8.2.2.6 Dissolved Oxygen Sensor

The dissolved oxygen sensor subsystem was tested in the following format using the methods included within this subsection. This subsystem package included calibration solutions to ensure that the data output readings of the dissolved oxygen sensor are accurate. The dissolved oxygen sensor was calibrated, from the sensor's datasheet specifications. Once the dissolved oxygen sensor was calibrated, it was connected to the interfacing board included in the pH sensor package. The interfacing board was then be connected to the I^2C header pins on the microcontroller unit. The dissolved oxygen sensor probe, was submerged into the various control solutions included in the sensor package. The output of the sensor data was output to a computer terminal to ensure that the sensor readings were within the datasheet specifications. Once these readings were properly output. The dissolved oxygen sensor was integrated with the entire system, that is, it was submerged into the water solution of the hydroponics system, and once again the data output readings was measured and compared with the expected readings. When the readings were confirmed the dissolved oxygen sensor testing is concluded, and the sensor was behave as the design requirements specify.

8.2.3 Automation Testing

All automation systems underwent a series of tests to validate that aspects of their operation perform properly before testing the system integrally. The purpose not only diagnoses for expected problems, but also discovered new challenges or ideas we did not consider during the research and design phases of the project.

8.2.3.1 Water Pump

The water pump, being submersible, resides in the nutrient reservoir during testing, its electrical connections made outside the reservoir. Testing started by verifying the pump operates when battery voltage is applied to the electrical leads of the pump. If the test would have failed, the pump would have been looked over for electrical shorts being made, taking precautions from electrical shocks while handling wet equipment. Afterwards, the pump was tested in the automation power relay system. The supplied battery power is modulated between two states (on/off), monitoring water head pressure and gallons per hour (GPH) meet the system specifications. In the event head pressure or GPH were to be significantly below our specifications, terminal voltage and current draw was measured while still maintaining our safety procedure, and from the measurements diagnosing the issue. All tests are passed, final integration was performed and tested again.

8.2.3.2 Light Array

The light array was housed directly on the housing of the plant enclosure. The first test verified light array operation by directly connecting the light array with the battery source. In the event the bulb didn't power on, the bulb would have been opened and inspected by testing the continuity of the internal fuse. If the fuse was found to be proper, the circuit board would have been visually inspected for evidence of burns or other suspicious markings. If still the problem had not been identified, the LEDs would have been first visually inspected for "blackened" enclosures, indicating the LED popped or burned. To save sanity on the part of the reader, various troubleshooting methods were performed et cetera or a new bulb will be purchase. Secondly, the bulb was connected to the automation power relay system and modulated between two states (on/off), verifying the LED bulb properly modulates within the system.

8.2.3.3 Water Flush System

The water flush system went through a series of verification steps. After installation, testing begins by ensuring all hoses did not leak nutrient solution. All leaks were remedied by adding Teflon tape to the bards where the hose mates with the barbed component. Once the system is found to be absent of leaks, the water reservoir was filled and the pump activated. This verified that the nominal flow for the water flush system was correctly oriented. If the test were to have failed, then the solenoids would have been adjusted by either switching them or altering the control code to correct the nominal flow. The states of the solenoids were then toggled to verify the two states were leak-free and route water to the proper destinations. The control code and testing procedure is outlined in section 8.4.

8.2.4 Hardware for User Control Testing

The following sections outline the hardware prototype testing devices and methods for the user control portion of our project. This is important to ensure compatibility between our chosen parts and the MCU itself as well as the web server.

8.2.4.1 LCD Screen

The LCD screen that was chosen for the user interface is the Kentec 3.5" TFT LCD Touch Screen BoosterPack. Because it is a BoosterPack, it interfaces easily with the EK-TM4C1294XL LaunchPad that we chose to prototype the project. The LCD has female header ports built-in to easily plug into the BoosterPack XL pins. In order to test the LCD screen functions, TI provides sample code in the form of their grlib_demo program. This program can quickly be compiled in Code Composer Studio and run after flashing to the LaunchPad device. After flashing and resetting the MCU, the LCD should power on and display sample graphics and recognize touch input to the screen to change pages and various markers and sliders. Once basic testing of the screen was completed, another BoosterPack, the Sensor Hub BoosterPack, was connected into the second set of BoosterPack XL pins on the LaunchPad board. The Sensor Hub Booster Pack contains 5 sensors that use the I^2C protocol. Testing both together ensured that the MCU can handle running the LCD screen and powering/reading sensor data simultaneously.

8.3 Software Test Environment

Since most of our software was written for the MCU, a majority of the software testing was done in Code Composer Studio. Since Code Composer Studio has a comprehensive debugging system built into it, we analyzed in real-time what variables and registers are being modified in our MCU and adjusted our code accordingly. Code Composer Studio also highlights what lines of code are modifying which variables and registers to easily be able to tell where the problem may lied, if there was one.

8.4 Software Specific Testing

The following sections and subsections will encompass the software testing procedure and development environments for prototyping the hydroponics system and the relevant subsystems. The software coding, trouble shooting, and execution of prototype testing methods for each of the subsystems will be documented in the following subsections.

8.4.1 MCU Sensor Data

The microcontroller sensor data has various inputs, I^2C, digital, UART, and analog; all of these protocols have their own unique specifications. The interfacing of the sensors to the microcontroller unit, include reading the data from the appropriate input pins. The I^2C sensors are byte addressable, as such, all of the I^2C data required parsing during the integration process. Once the data is parsed properly, the output of the data read correctly to the terminal, after proper formatting to the respective input data. The digital and UART sensors are simple binary serial, and each have their own respective pin inputs, so parsing was not required. Thought both of these methods needed to be formatted within the software code, to ensure useful output to the computer terminal. The analog sensor, were read from the analog input pin on the microcontroller, and the data was correlated to the datasheet of the respective sensor. All of the sensors included in the hydroponics system design, have significantly documentation, to verify that the sensor software interfacing is done effectively. All of the software prototyping testing was done in series with the hardware prototyping and interfacing with each of the sensors by use of the microcontroller unit software coding methods.

8.4.2 MCU Control Data

The MCU changes the states of the automation subsystems via a decoder with connections to solid state relays. Control data programed into the MCU memory eases the implementation of automation since conditional statements and cases was written to change the control data for given sensor values.

8.4.2.1 Light Array Control

With testing complete of the light array, modulation of the device was evaluated to ensure it functions properly when a control signal from the MCU was applied. The bit-banded signal controls which solid state relay will close, forming a circuit for current to conduct through the light array. With this test, the light array either activated with the expected control signal, or it did not activate. In the event the circuit didn't close, the control signal would have been analyzed with an oscilloscope with a byte read feature or a serial terminal program such as HyperTerminal. If the control signal were proper, the decoder would have evaluated next, testing for which output pin is held high. Once corrected, the control code for the light array was verified and treated as fully functional.

8.4.2.2 Water Flush Control

When testing of the water flush system is complete, toggling of the solenoids is evaluated to verify its functionality when a control signal from the MCU is applied. The signal controls which solid state relay closed via a process known a bit-banding, the result being the circuit closed, forming a path for current to conduct through the water flush system. With this test, the solenoids either activated with the expected control signal or an error resulted, requiring diagnosis and repair of the control signal. As previously discussed, equipment or a program capable of analyzing the serial control signal would have been used to check that the control signal is sending data as expected to the solenoids to activate.

8.4.2.3 Sensor Threshold Modifications

The sensor thresholds for all of the sensors will need to be modified relative to the type of plant life being cultivated in the hydroponics system. The sensor data readings will be averaged with themselves over time, to ensure the smoothing out of in spikes or dips in the sensor readings, especially during the hydroponic system state transitions such as, a water flush or a automatic tilling.

8.4.2.5 Water Pump Control

The control signals tests for the water pump are most similar to the light array control procedure. The purpose is to verify that the decoder still works and the solid state relay was installed correctly. Once these checks were performed, the electrical connections can be fitted and prepared in a more final arrangement, improving the longevity and safety of the system. With the final automation subsystem in place, the current draw from the battery was measured and using simple DC power calculations (P = I*V), we then compare our final power dissipation with our expected values. If power dissipation would have been too great, we would have used a current limiter on the water pump to reduce power dissipation or revise our specifications if limiting is not feasible while maintaining adequate head pressure and flow rate.

8.4.3 LCD Interface

The LCD interface was tested with the LCD screen being directly connected to the board. A basic test of whether or not the LCD displays the UI and receives touch commands were performed. If the screen is able to display graphics and receive inputs via touch, then the system was pass the initial test and further testing can begin. Testing of the functionality of the system is similar to that of the web page.

8.4.3.1 Data Display

On the main LCD interface screen, the sensor data is displayed prominently. The first test was to see if any data shows up at all on the main page. If data is not showing up, then checks to the connections between the LCD screen and the MCU was done as well as debugging to make sure variables are getting properly updated. If data is showing up, it was cross checked with the debug information from the MCU to make sure the values agree. If values don't agree, then once again the code was examined to determine why. Data can also be refreshed manually by the user, so this function will be checked as well to ensure reliability. It also measured how long it takes for the data to refresh both on the MCU itself and on the LCD screen. If the data was displayed correctly and the manual refresh function works in a timely manner (~2-3 seconds), then the system passed the test.

8.4.3.2 Sensor Thresholds

The main page should be able to display the current minimum and maximum thresholds before triggering sensor alerts. The first test was to determine if the alerts stay off if the sensor readings are within threshold. The second test was to change some aspects of the hydroponics system (add more water, dump water, increase temperature, etc.) to make sure the alerts for both minimum and maximum threshold were activated. The LCD screen should also have the ability to adjust minimum and maximum thresholds. After adjusting on the LCD screen using touch buttons, the LCD screen should have been able to display the new thresholds and trigger an alert if necessary at the time of the change. Several threshold changes was performed and compared to the MCU debug data to see how long it takes for threshold changes to occur in the MCU. The ability to refresh thresholds after adjusting and alerting was tested. If the system is able to adjust and display thresholds as well as generate alerts accurately, the system test passed.

8.4.3.3 Sending Control Data to MCU

One of the functions of the LCD screen was able to send control data to the MCU and to control the water pump, solenoids, and light array. It is also be able to tell if the pump is currently running as well as of the light array is on. The first test was to check if the status of the pump, light array, and solenoids are accurate. Then, an attempt to activate and deactivate each of the systems were made. If the system accurately reflects the status of the water pump, light array, and solenoids as well as having a reasonable time for the MCU to act on a command (around 2-3 seconds), then the system passed.

9.0 Project Operation

This section describes the operation of the project, including instructions on how to connect and power up the project, how to operate the LCD screen, and how to troubleshoot potential issues.

9.1 How to Power Up the Project

Before operation, the solar panel and battery must first be connected. Connect the solar panel to the charge controller, and then connect the battery to the charge controller. After connecting both, connect the “load” cables from the charge controller to the bus bar attached to the hydroponics system. This will ensure that the solar panel will be able to charge the battery, as well as let the battery power the respective components of the system. There is a switch connected between the battery and the bus bar (the component that allows the correct voltages to be distributed across the project). Flipping the switch to the ON position will light up an LED on the switch to indicate power is successfully flowing. Once power is flowing, the system will activate and the voltage regulator board will make sure that the main board, control board, sensors, and LCD touchscreen all get either 3.3V or 5V power. The pump, LED light array, and solenoids run at 12V and do not need to be connected to the voltage regulator bus bar voltages.

9.2 How to Operate the User Interface

Upon powering up the system, the LCD screen will display a logo for Duke Energy and UCF’s College of Engineering. After that, the menu in Figure 9.1 will be displayed:

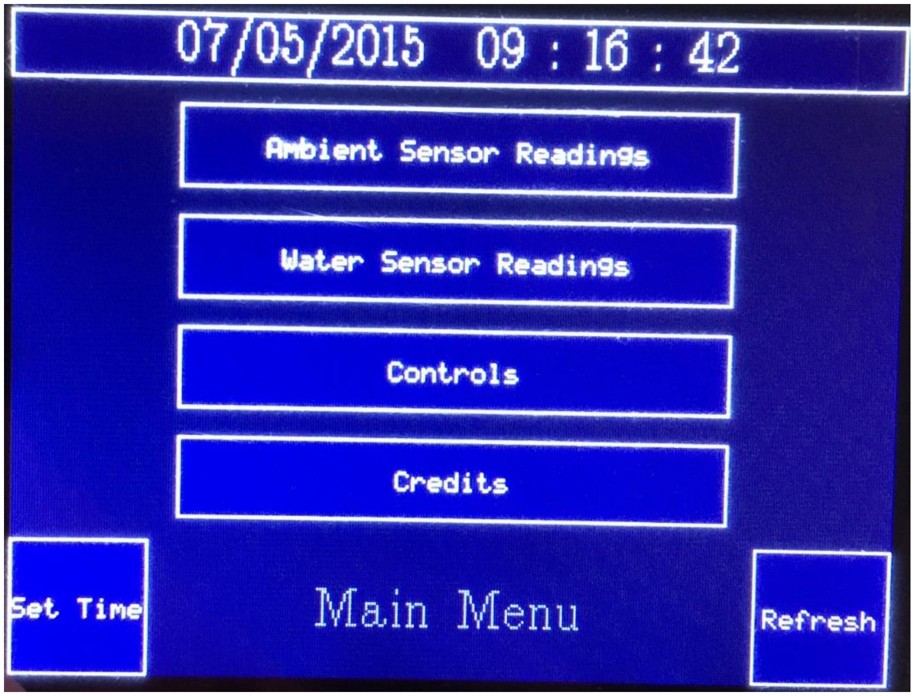


Figure 9.1: The Main Menu

The main menu screen consists of the current time at the top, several options to choose from, a Refresh button to refresh your current screen, and a Set Time button to change the time. Selecting either of the “Sensor Readings” options will bring you to a screen similar to Figure 9.2. Selecting the “Controls” button will bring you to a screen similar to Figure 9.4.

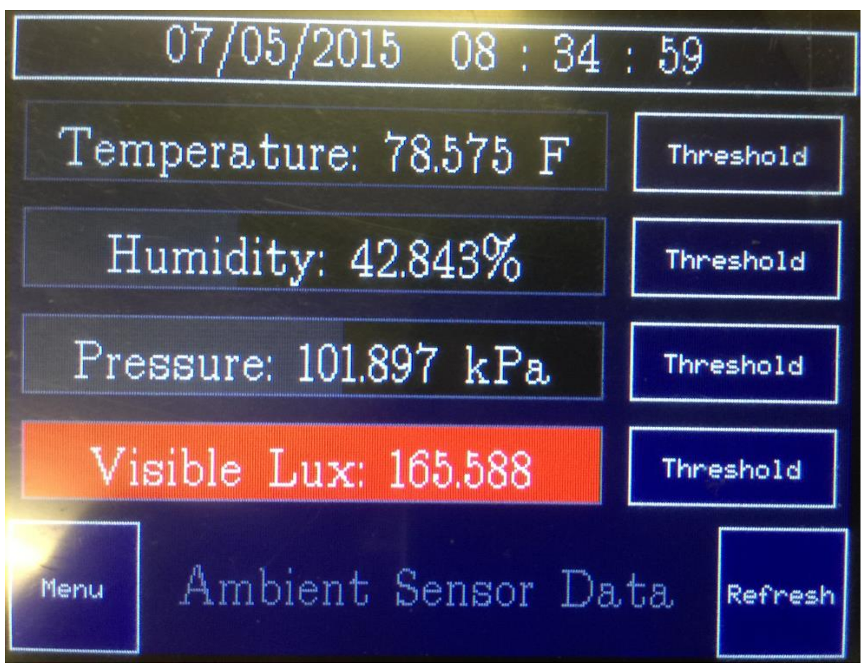


Figure 9.2: Ambient Sensor Readings Screen

The ambient sensor readings screen shows you the current time as well as the most current readings from the ambient sensors. Readings include ambient temperature, humidity, pressure, and visible light. Readings will be highlighted in red if they are below the minimum or above the maximum threshold set by the user. Pressing the “Threshold” button will bring you to the Threshold Adjustment screen shown in Figure 9.3. Pressing the “Refresh” button will refresh the readings and update the display with the current readings. Pressing the “Menu” button will return you to the main menu screen on Figure 9.1.

Figure 9.3: Threshold Adjustment Screen

In the Threshold Adjustment screen, both the minimum and maximum threshold of each individual sensor reading can be set and saved. Hitting the Done button will return you to the Sensor Readings screen in Figure 9.2. Hitting the Menu button will return you to the Main Menu screen in Figure 9.1.

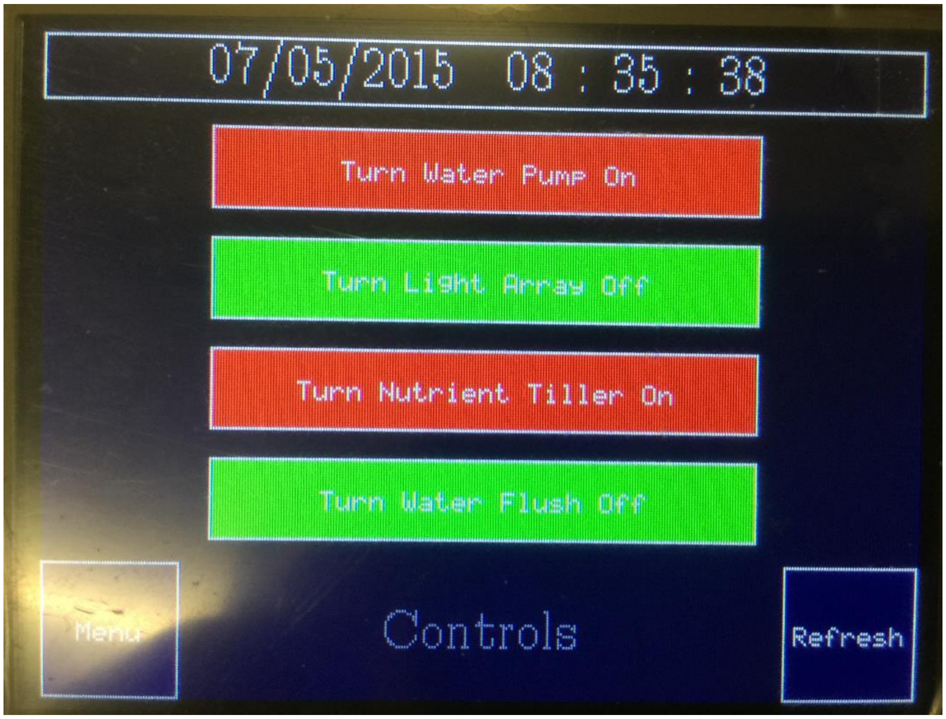


Figure 9.4: Controls Screen

In the Controls screen, individual subsystems can be enabled and disabled just by touching the button. Hitting the Menu button will return you to the Main Menu.

9.3 Troubleshooting Issues

If any of the sensor readings show 0.0 on any of the readings screen, or the sensors screen freezes while loading, make sure all of the sensors are powered on and connected to the main board. The software may freeze trying to read from the I^2C bus when there are no sensors connected where they are expected to be.

If the system does not power on, make sure the battery is connected to the charge controller and is sufficiently charged. Make sure that the load cables are connected from the charge controller to the system's 12V input. Make sure that the main board is connected to the 3.3V out of the voltage regulator and that the 5V of the LCD screen is connected to the 5V out of the voltage regulator.

If the controls do not activate the subsystems, make sure the control board is plugged in and the relays are wired into the system's bus bar, giving the ability to power each of the subsystems.

If the system freezes, it can be restarted by hitting the reset button on the main board or by switching the power switch off and on.

10.0 Administrative Content

The administration content pertains to objective performance of the project, defined by timelines and a finance discussion.

10.1 Milestone Discussion

Milestones for the hydroponics project were essential for evaluating group performance and deadlines for the course. Below is a summary of the Senior Design I timeline and the timeline for Senior Design II.

10.1.1 Senior Design I

During Senior Design I, the research phase took slightly longer because our group wanted to design the system to be more flexible since the schedule for Senior Design II is much shorter over the summer. Otherwise, the agreed upon timeline was mostly upheld and is shown in figure 10.1.

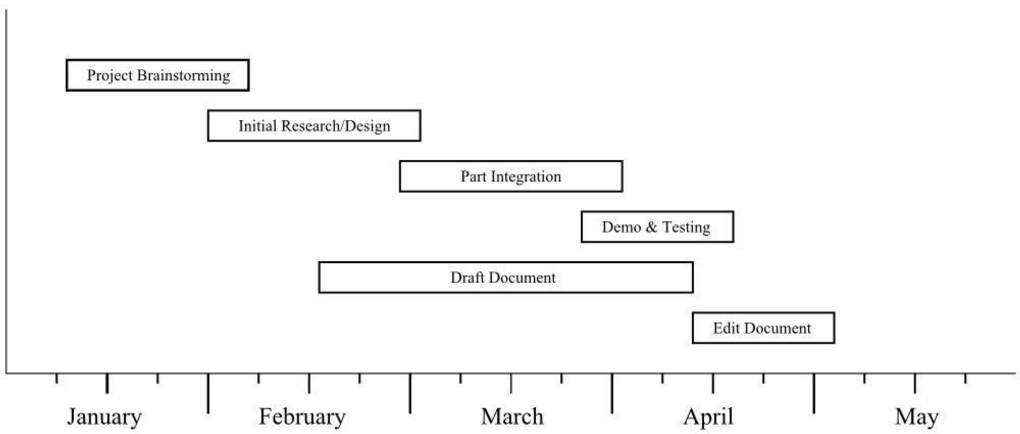


Figure 10.1: Senior Design I Timeline

10.1.2 Senior Design II

In Senior Design II, all focus was directed toward final design revisions and testing during Senior Design II. We made an aggressive schedule to compensate for any contingencies unforeseen in the design phase or part failures. Figure 10.2 displays the timeline for our project prototyping and testing.

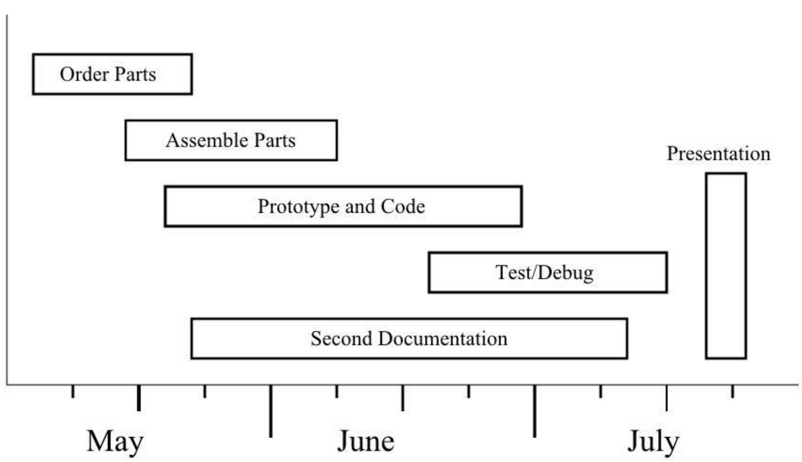


Figure 10.2: Senior Design II Timeline

10.2 Budget and Finance Discussion

This project was designed to make an energy sustainable hydroponics easy and accessible to people. People who may be unfamiliar with the solar power and botanical knowledge required to successfully deploy and maintain a hydroponics garden will benefit from this project. The goals and objectives of this project were to accomplish the following tasks. Create a low powered solar energy dependent system for energy sustainability. Understand and implement sensor interfacing, automation of electromechanical systems, system integration, and electrical power systems. This hydroponics system was designed in such a way that the entire system is green energy dependent, automated, low power, user friendly, low cost and enables an easy to use user interface. The function of this project was to fully execute, automate, monitor, and report the status of the system.

Appendices

Appendix A – Copyright Permissions

Permission to use SPI and I2C figures college x

Jon Spy <spychalskyjc@gmail.com> Mar 25
to info

Hello, my name is Jon Spychalsky and I was wondering if my senior design group may use the SPI and I2C figures from your SPI/I2C intro webpage (link) for our paper. We would be very grateful for your permission and naturally we would credit byteparadigm as the source.

Thank you,
Jon Spychalsky

Frédéric Leens <frederic.leens@byteparadigm.co Mar 26
to me

Hi Jon,

Yes, you can reuse the figures.
Which school / university are you attending?
Can I receive a link to download the paper if you publish it?

Thanks,

Best regards,

Frederic

Frédéric Leens
CEO

byteparadigm
Avenue Molière, 18
B-1300 Wavre
Belgium

Permission for Figure 3.1:

Re: Permission to use image

DELETE REPLY REPLY ALL FORWARD ...

agnieszkanazaruk@gmail.com on behalf of Aga and Javier <founders@getni Mark as unread
Mon 3/30/2015 9:40 PM
Inbox

To: David Mascenik;

Hey David! Thats all good! :)

Agnieszka Nazaruk (Aga) - Founder
Javier Morillas - Founder
http://getniwa.com/

UK: +44 07546 141 192 | CN: +86 195 7664 5276
Like | Follow | Skype ID: agnieszka.nazaruk

> *Now, Everyone Can Grow*

On Mon, Mar 30, 2015 at 4:55 PM, David Mascenik <Dmascenik@knights.ucf.edu> wrote:

> Hello, I am a senior EE undergraduate student currently attending UCF. As part of my Senior Design paper, I wanted to use an image from your website that I've edited to be smaller. This part of the paper is just a small section of comparisons of automation based hydroponics systems. Please let me know if I can use this image (http://getniwa.com/images/niwa.png) in my paper and what kind of credit that I need to give if I can. Thanks!

Permission for Figures 3.4 3.5, 3.7, 3.8, and 3.14:

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Permission for Figure 3.6 and 6.7:

Request for Permission for Educational Use of Pictures and Diagrams

James Tooles <jtoolesiii@gmail.com> 11:13 PM (9 hours ago)
to info

Hello,

I am contacting Atlas Scientific to request permission to use pictures, wiring diagrams, and data sheets for the purposes of educational research documentation. None of the images or information used from your company will be used for a profit, or for commercial purposes.

Name: James C. Tooles
Project: Autonomous Hydroponics System
School: University of Central Florida
Major: Electrical Engineering
Class: Senior Design
Products: EZO pH Sensor
EZO Dissolved Oxygen Sensor

Thank You,
James Tooles

Click here to Reply or Forward

Permission for Figure 3.11:

Our learning resources and documentation are released under a Creative Commons licence.

> A Creative Commons licence is used when an author wants to give people the right to share, use and build upon a work that they have created.
>
> – Wikipedia

The licence we use is CC BY-SA (attribution and share-alike), which is the licence used by Wikipedia.

You are free:

- **to Share** – to copy, distribute and transmit the material, and
- **to Remix** – to adapt the material

Under the following conditions:

- **Attribution** – You must attribute the materials to the Raspberry Pi Foundation (but not in any way that suggests that they endorse you or your use of the materials), as well as including a link to the original resource. You must also indicate if changes were made.
- **Share Alike** – If you alter, transform, or build upon these materials, you must distribute the resulting materials under the same licence.

Link to license: https://www.raspberrypi.org/creative-commons/

Link to the original resource: https://www.raspberrypi.org/wp-content/uploads/2014/04/bplus-gpio.png

The image was altered to edit out the information about the ID_SD and ID_SC pins.

Permission for Figure 3.12:

Except where otherwise noted, content on this site is licensed under a
Creative Commons Attribution-ShareAlike 4.0 International License.
Copyright 2014 - Gray Cat Labs

You are free to:

Share — copy and redistribute the material in any medium or format

Adapt — remix, transform, and build upon the material

for any purpose, even commercially.

The licensor cannot revoke these freedoms as long as you follow the license terms.

Under the following terms:

Attribution — You must give **appropriate credit**, provide a link to the license, and **indicate if changes were made**. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

ShareAlike — If you remix, transform, or build upon the material, you must distribute your contributions under the **same license** as the original.

No additional restrictions — You may not apply legal terms or **technological measures** that legally restrict others from doing anything the license permits.

Image created by Alexander Hiam.

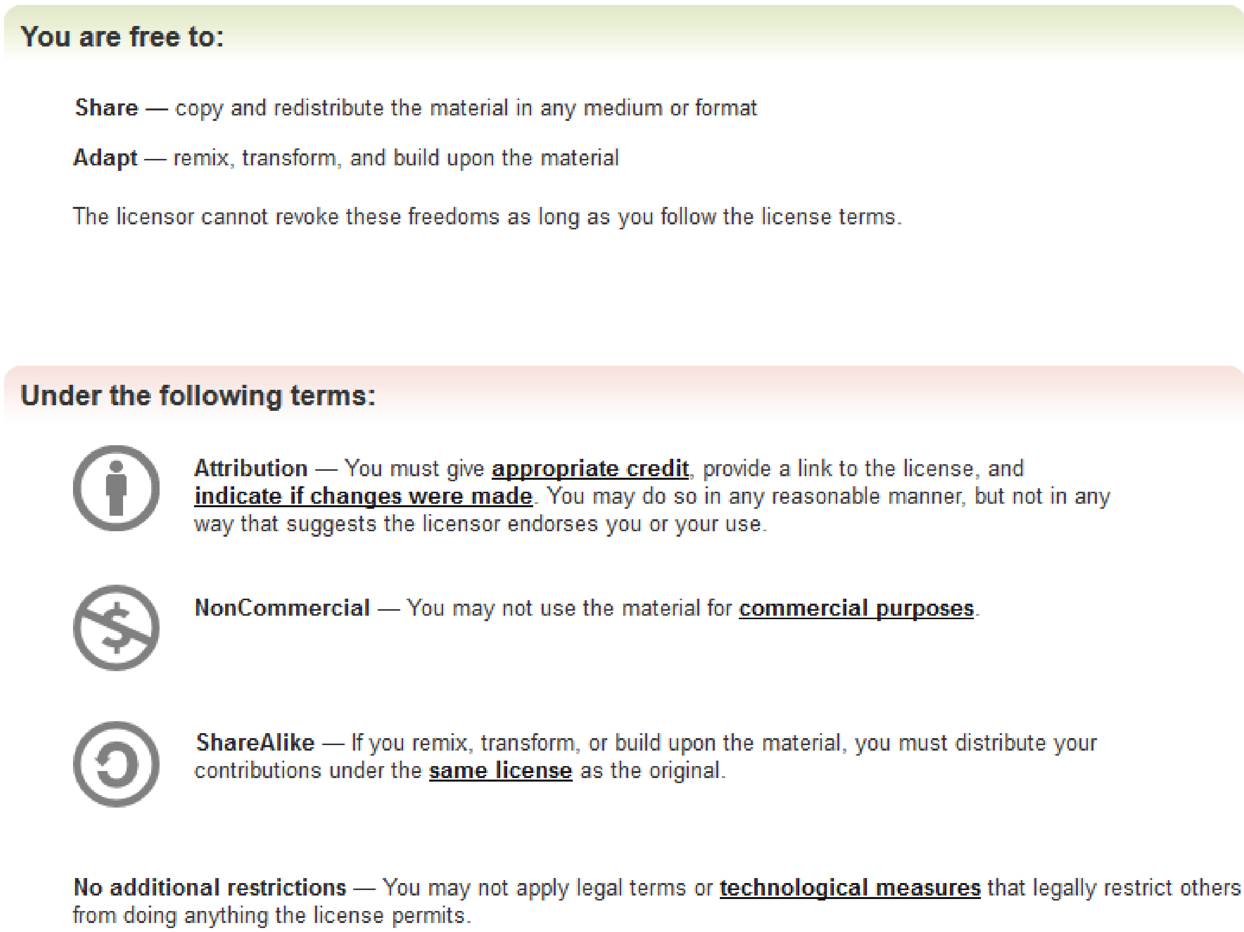
Link to license: http://creativecommons.org/licenses/by-sa/4.0/

Link to the original resource: http://graycat.io/wp-content/uploads/2013/01/beaglebone_pinout1.png

Permission for Figure 3.13:

License

tutorials are CC BY-NC-SA 3.0


You are free to:

Share — copy and redistribute the material in any medium or format

Adapt — remix, transform, and build upon the material

The licensor cannot revoke these freedoms as long as you follow the license terms.

Under the following terms:

Attribution — You must give **appropriate credit**, provide a link to the license, and **indicate if changes were made**. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

NonCommercial — You may not use the material for **commercial purposes**.

ShareAlike — If you remix, transform, or build upon the material, you must distribute your contributions under the **same license** as the original.

No additional restrictions — You may not apply legal terms or **technological measures** that legally restrict others from doing anything the license permits.

Image created by CaseyTheRobot.

Link to license: http://creativecommons.org/licenses/by-nc-sa/3.0/

Link to the original resource: https://cdn.sparkfun.com/assets/learn_tutorials/3/2/2/edison-pinout_1.png

Appendix B – Datasheets

LM2854 Buck Regulator Datasheet: http://www.ti.com/lit/ds/symlink/lm2854.pdf

TPS56x200 Step-Down Regulator Datasheet: http://www.ti.com/lit/ds/symlink/tps563200.pdf

EK-TM4C1294XL LaunchPad Datasheet: http://www.ti.com/lit/ug/spmu365a/spmu365a.pdf

TM4C1294NCPDT MCU Datasheet: http://www.ti.com/lit/ds/symlink/tm4c1294ncpdt.pdf

nRF24L01+ 2.4 GHz Transceiver Datasheet: https://www.sparkfun.com/datasheets/Components/SMD/nRF24L01Pluss_Preliminary_Product_Specification_v1_0.pdf

Kentec EB-LM4F120-L35 Datasheet:
http://www.kentecdisplay.com/uploads/soft/Products_spec/EB-LM4F120-L35_UserGuide_04.pdf

BOOSTXL-SENSHUB Sensor Hub BoosterPack Datasheet:
http://www.ti.com/lit/ug/spmu290/spmu290.pdf

Atlas Scientific EZO Class Embedded pH Circuit Datasheet:

https://cdn.sparkfun.com/datasheets/Sensors/Biometric/pH_EZO_datasheet_v13.pdf

Atlas Scientific EZO Class Embedded Dissolved Oxygen Circuit Datasheet:

http://www.atlas-scientific.com/_files/_datasheets/_circuit/DO_EZO_Datasheet.pdf?

Milone Technologies, Inc. eTape Continous Fluild Level Sensor PN-12110215TC-12 Datasheet:

http://www.adafruit.com/datasheets/eTape%20Datasheet%2012110215TC-12_040213.pdf

Maxim DS18B20 Programmable Resolution 1-Wire Digital Thermometer Datasheet:

http://www.adafruit.com/datasheets/DS18B20.pdf

Intersil Integrated Digital Light Sensor with Interrupt Datasheet:

http://www.intersil.com/content/dam/Intersil/documents/isl2/isl29023.pdf

Bibliography

[1] Simonson, Austin. (2014, November 11). Hyduino - Automated Hydroponics with an Arduino [Online]. Available: http://www.instructables.com/id/Hyduino-Automated-Hydroponics-with-an-Arduino/

[2] Niwa. (2015). Niwa [Online]. Available: http://getniwa.com/

[3] Peterphrastus. (2012, Febuary 20). DIY Automated Hydroponics and Irrigation Control Systems [Online]. Available: http://peterphrastus.com/drupal/

[4] DiLeonardo, Matthew, et al. (2014, April 29) LeafAlone Hydroponics System [Online]. Available: http://www.eecs.ucf.edu/seniordesign/sp2014su2014/g09/documentation.html

[5] Simpson, Chester. (2011). Linear and Switching Voltage Regulator Fundamentals Part 1 [Online]. Available: http://www.ti.com/lit/an/snva558/snva558.pdf

[6] Simpson, Chester. (2011). Linear and Switching Voltage Regulator Fundamentals Part 2 [Online]. Available: http://www.ti.com/lit/an/snva559/snva559.pdf

[7] Hearst Electronic Products. (2012, May 08) Understanding the Advantages and Disadvantages of Linear Regulators [Online]. Available: http://www.digikey.com/en/articles/techzone/2012/may/understanding-the-advantages-and-disadvantages-of-linear-regulators

[8] Stewart, William. (2000, January 7) Linux History [Online]. Available: http://www.livinginternet.com/i/iw_unix_gnulinux.htm

[9] Trebisky, Tom. (2013, June 13) Beaglebone and Hardware interfacing [Online]. Available: http://cholla.mmto.org/computers/beagle/hardware/interfacing.html

[10] Upton, Eben. (2014, July 14) New product launch! Introducing Raspberry Pi Model B+ [Online]. Available: https://www.raspberrypi.org/introducing-raspberry-pi-model-b-plus/

[11] Martin, Ben. (2015, February 27) Performance Testing the New $35 Raspberry Pi 2 [Online]. Available: http://www.linux.com/news/embedded-mobile/mobile-linux/813223-performance-testing-the-new-35-raspberry-pi-2-

[12] Kuhns, Casey. (2014, November 22) General Guide to SparkFun Blocks for Intel® Edison [Online]. Available: https://learn.sparkfun.com/tutorials/general-guide-to-sparkfun-blocks-for-intel-edison

[13] Vilo, Rei. (2013, July 13) Kentec 3.5" LCD With Touch BoosterPack for Stellaris [Online]. Available: http://embeddedcomputing.weebly.com/kentec-35-lcd-with-touch-boosterpack-for-stellaris.html

[14] Dmitrij, S. (2011, April 1) Nginx vs Cherokee vs Apache vs Lighttpd [Online]. Available: http://www.whisperdale.net/11-nginx-vs-cherokee-vs-apache-vs-lighttpd.html

[15] Hoff, Todd. (2008, March 12) YouTube Architecture [Online]. Available: http://highscalability.com/youtube-architecture
[16] Hiam, Alexander. (2012, December 8) BeagleBone pinout [Online]. Available: http://graycat.io/beaglebone/beaglebone-pinout/
[17] Kridner, Jason. (2015, March 30) Getting Started with BeagleBone & BeagleBone Black [Online]. Available: http://beagleboard.org/getting-started
[18] Valvano, Jonathan, "Interrupt Programming and Real-time Systems," in *Embedded Systems: Introduction to Arm® Cortex(TM)-M Microcontrollers (Volume 1)*, 5th ed. Austin: Self-Published, 2014, ch. 9, sec. 1-9, pp. 343-375.
[19] "How do solar cells work?" [Online]. Available: http://www.physics.org/article-questions.asp?id=51
[20] Intersil (2014, May 1) "Integrated Digital Light Sensor with Interrupt" [Online]. Available: http://www.intersil.com/content/dam/Intersil/documents/isl2/isl29023.pdf
[21] Josh Greenfield. "Detectors" [Online]. Available: http://chemwiki.ucdavis.edu/Analytical_Chemistry/Instrumental_Analysis/Spectrometer/Detectors/Detectors#Photodiodes_and_Photodiode_Arrays
[22] Vasco Polyzoev and Habib Karaki. (2014, February 27) [Online]. Avaiable:http://www.sensorsmag.com/sensors-mag/demystifying-thermopile-ir-temp-sensors-13157
[23] "NFT system" [Online]. Available: http://hydroponie.fr/mendota-hydroponie-production-local-solution-global/
[24] Byte Paradigm. "Introduction to SPI and I2C protocols" [Online]. Available:http://www.byteparadigm.com/applications/introduction-to-i2c-and-spi-protocols/?/article/AA-00255/22/Introduction-to-SPI-and-IC-protocols.html)
[25] "Society of Robotics. Microcontroller UART Tutorial" [Online]. Available:http://www.societyofrobots.com/microcontroller_uart.shtml
[26] Chenyang Lu. "Von Neumann and Harvard Architectures" [Online]. Available:http://www.cse.wustl.edu/~lu/cse467s/slides/dsp.pdf
[27] "Physical and Data Link layer" [Online]. Available: http://standards.ieee.org/getieee802/download/802.11-2012.pdf
[28] Keld Simonsen. "C11 Standard" [Online]. Available: http://www.open-std.org/jtc1/sc22/wg14/www/standards.html#9899
[29] C Clark. "IEEE 1149.1-2013" [Online]. Available: http://standards.ieee.org/findstds/standard/1149.1-2013.html
[30] "Chip Packages" [Online]. Available: https://learn.sparkfun.com/tutorials/integrated-circuits/ic-packages
[31] "Bluetooth Core Version 4.0" [Online]. Available: https://www.bluetooth.org/en-us/specification/adopted-specifications
[32] "PCB Design Software" [Online]. Available: http://www.diyspaceexploration.com/best-pcb-design-software/

[33] “Eutrophication” [Online]. Available: http://toxics.usgs.gov/definitions/eutrophication.html
[34] M Ruse. “Copper Salts” [Online]. Available: http://www.inchem.org/documents/pims/chemical/pimg002.htm#SectionTitle:3.4 Hazardous characteristics
[35] “Bioaccumulation and Biomagnification” [Online]. Available: https://www.marietta.edu/~biol/102/2bioma95.html
[36] “Conflict Materials” [Online]. Available: http://www.sony.net/SonyInfo/csr_report/sourcing/materials/
[37] “Example SEC Filing” [Online]. Available: http://www.sony.net/SonyInfo/IR/library/ConflictMineralsReport2013.pdf
[38] “JST connectors” [Online]. Available: http://www.rcgroups.com/forums/showthread.php?t=1493712
[39] “Leaded and Unleaded Solder” [Online]. Available: https://www.paceworldwide.com/pacenter/soldering/lead-free-vs.-leaded-solder
[40] Sarah Griffiths. “LED bulb guide” [Online]. Available: http://www.dailymail.co.uk/sciencetech/article-2546665/At-A-bulb-WONT-power-cut.html