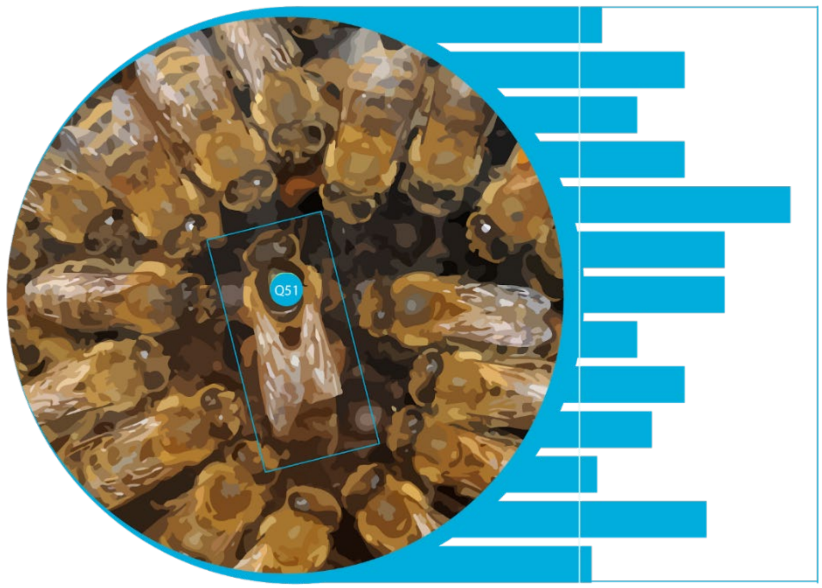
Project Documentation

Plan Bee

Commercial Beehive Health-Tracking Solution
Utilizing Novel Internet-of-Things Sub-1GHz Communications Template



University of Central Florida
Department of Electrical and Computer Engineering

Dr. Lei Wei
Senior Design II

Group 3

Tariq Ausaf – Electrical Engineering
Giovanny Reyes – Electrical Engineering
Yannick Roberts – Electrical Engineering
Katelyn Winters – Electrical Engineering

We would like to thank Steven Eisele at Pollination USA for fully sponsoring Plan Bee.

I. Table of Contents

II. Table of Figures

III. Executive Summary

The global market of connected devices is estimated to exceed $7.1 trillion by 2020. As such, the demand for a general-purpose asset tracking solution may be significant enough for an entirely new market segment to emerge. A cost-effective device capable of gathering data from a series of connected sensors and broadcasting this information to the cloud for further analysis. Plan Bee aims to accomplish this task.

Our project is focused on the creation of an Internet-of-Things Sub-1Ghz Communications Template and – because we are being fully funded by the commercial apiary Pollination USA – retrofitting this with sensors for monitoring the health and outputs of remote, commercial beehives. This project will cover MEMS and sensors technology to achieve tracking of metrics; this will include but not be limited to RF design and integration of a GPS locator, load cells, a thermistor, a humidity sensor, accelerometers and vibration sensors. The hardware technologies will be weatherproofed for placement upon the commercial beehives. IBM Watson will be discussed and used for intelligent algorithms, and Amazon AWS will be shown used for the storage and manipulation of the apiary's data. Other web technologies will be shown used to create an application "dashboard" for the commercial apiary.

Presented in this document is the implementation details required to develop a connected device capable of funneling sensory data across a secured internet connection to a cloud connected database for analysis and logistical planning.

[word count: 234]

IV. Project Description

This section is effectively a project overview, including: the motivation and application of our project, much of which was taken from interviewing Steven Eisele and other beekeepers and further market research; a technology-in-development overview to describe our technology-related goals; project illustrations and block diagrams including Power, Hardware and Software stack; requirements specifications including the House of Quality and a short description of our metrics, and how we plan to demo specific engineering specifications in front of committee; and related work, including research papers, prior art (US Patents), and on-the-market available products.

Note that this chapter is heavy with beekeeping terminology and other industry-related jargon, especially within the motivation. Because of this, we strongly recommend the reading of Appendix A3. Beekeeping Terminology, a short one-page summary on the usage of beekeeping jargon, before delving into this section.

I. Motivation

It is understood that bees are a vital component of the ecosystem. Many flowering, food-producing crops are reliant upon bees for reproduction through pollination. [1] Honeybees are responsible for pollinating greater than thirty-percent of food crops grown in the United States each year. [2] Honeybees are productive members of the agriculture ecosystem. As such, Beekeepers – Apiarists – are of vital importance to the agriculture economy, and thus the national security of the United States as well.

Commercial honeybees, which accomplish the bulk of the pollination of U.S. agricultural farms, are facing problems in today's changing climate. The hives are kept in yard-arrangements, and yards are kept distanced from each other. Five-thousand individual hives are usually spread across one-hundred-square-miles. [3] And these hives are all remotely located, and usually in extreme environments, making maintenance and upkeep of every individual hive unfeasible economically.

Like humans, the health of a beehive factors into account many different individual bees – like cells – that work best unanimously when healthy, and at the mercy of disease and pests. Over the past several years, on average, beehives have been dying at a rate of about thirty-percent-per-year. [4] This is largely due to several health concerns beehives face: pests including varroa mites and small hive beetles; diseases including Colony Collapse Disorder and bacterial, viral and fungal infections; and invaders including other aggressive, territorial beehives, and Africanized bees. [5]

Steven Eisele, our sponsor, has noted that the state governments keeps a census record of individual beehives and patterns, and reimburses commercial apiaries for individual hive losses each quarter. He noted that he would like to see a system for each individual beehive health monitoring, for reasons stated above, that keeps track of the:

- Weight,
- Feeding schedule,
- Medication schedule,
- Location of beehive (GPS),

- Humidity, and
- Temperature.

Hive weight and its rate-of-change (delta) is the single most important statistic to track for commercial apiaries. [6] Constant weight tracking gives the beekeeper information on when the bees are leaving (at dawn) and resting (at night) in the hive, how many bees are active within the hive, and the amassing of honey within the hive until the S-curve levels, when it is time to harvest.

Feeding and medication schedules need to be tracked, as well as a running tabulation of the last time each hive was last fed and medicated and by whom. Inefficiencies in hive maintenance that are currently present – like revisiting already-maintained hives – need to be eliminated, as it wastes company resources including time, food, and medication. The running tabulation keeps track of the last person to deal with each individual hive, eliminating any ambiguity in what work was done by whom and when.

During transportation to contract location and during contract duration, live location of the physical beehive (GPS) needs to be tracked by both the beekeeper and customer. This is important because of the high frequency with which these beehives are relocated, during which they are often lost or stolen. Precise location tracking is one proposed solution to this problem.

Honeybees optimally need to maintain a brood nesting temperature between 32 – 35 degrees Celsius for growth. [7] Temperature fluctuation of about 15 degrees in either direction from these temperatures is acceptable, because of mechanisms bees have developed to regulate hive temperature. Without brooding, it is acceptable for temperatures to range from about 0 – 50 degrees Celsius.

Similarly, relative humidity between 50 – 60% is optimal for brood nesting, while outside of brooding through 20% - 100% relative humidity is acceptable. [8] During brooding then, it is especially important to track temperature and humidity within the hive.

Stretch goals that we have discussed with Mr. Eisele also included:

- Location of the queen bee,
- Listening to the communication of the honeybees
- Pheromones present within the hive,
- Chemicals present within the honey, and a
- Feeding and Medication Control System that automates non-health maintenance of the individual beehives.

The queen bee is the heart of the hive; wherever she goes, the hive follows. And when swarming occurs, she is the orchestrator and lead. When bees swarm, if the queen bee is being properly tracked, the beehive can be found and relocated into a new apiary.

Honeybee colony vibrational measurements highlight the brood cycle. [9] Honeybees displaying dance communication behavior within the colony – the frequency of which generates a corresponding electrical impulse in the bee's antenna – gives information about quality of a food source, temperature, a need for water, the condition of the hive, and the condition of the queen bee. [10] Sensitive vibration sensors can detect the frequency at which certain bees are communicating, as well as the amplitude and deviation around each frequency. Trophallaxis – accomplished by exchange of pheromones – can be used to communicate the same information. It can also communicate that invaders and pests are present. [11] Pheromone sensors within the hive could thus give the apiarist information about the state of the bees within the beehive.

Honey texture is sensitive to the chemicals present within the honey. The color, flavor and aroma ("texture") of honey differs depending upon the nectar of flowers visited by the bees that made it. [12] For example, a hibiscus farm nearby to the colony would produce a light yellow-pink honey, with hibiscus-reminiscent aroma and a mild of the hibiscus flower - a honey varietal. It is important for the apiarist to keep track of honey varietals produced by a colony.

These beehive statistics would be presented in a browser through website featuring a login portal, or through a phone application available through the app store. He would additionally like to receive alerts when the status of his hives change. For example, when honey is ready to be harvested, when it is time for feeding or medication, when humidity or temperature could affect the beehive health, when the individual beehives are moved, and so forth.

II. Technology-in-Development Overview

The global market of connected devices is estimated to be valued at more than $7.1 trillion by 2020. [13] As such, the demand for a general-purpose asset tracking solution may be significant enough for an entirely new market segment to emerge. A cost-effective device capable of gathering data from a series of connected sensors and broadcasting this information to the cloud for further analysis. Our technology-in-development is aimed at accomplishing this task, generally appropriate for most Internet-of-Things application. Our technology is a general Commercial Sub-1Ghz Communications Template. We will focus on implementation details required to develop our technology, with the capability of funneling sensory data across a secured internet connection to a cloud connected database for analysis and logistical planning.

It is worth noting that we are developing this for a specific application because it fits our senior design project needs and allows us a budget, but this is not our final intent. Our final intent is a versatile Internet-of-Things (IoT) application communications template that can be applied directly to several applications for asset tracking, such as transcontinental transportation of goods, service applications such as agriculture farms, as well as many others. These are applications that we intend to market this device toward following senior design and intend to develop an end-product for. Implementation details required to create this sensory-communications device are presented here in an easy-to-follow manner aimed at a general engineering-experienced audience. However, those with non-technical background, with some thinking, should be able to follow our logic.

III. Project Illustration

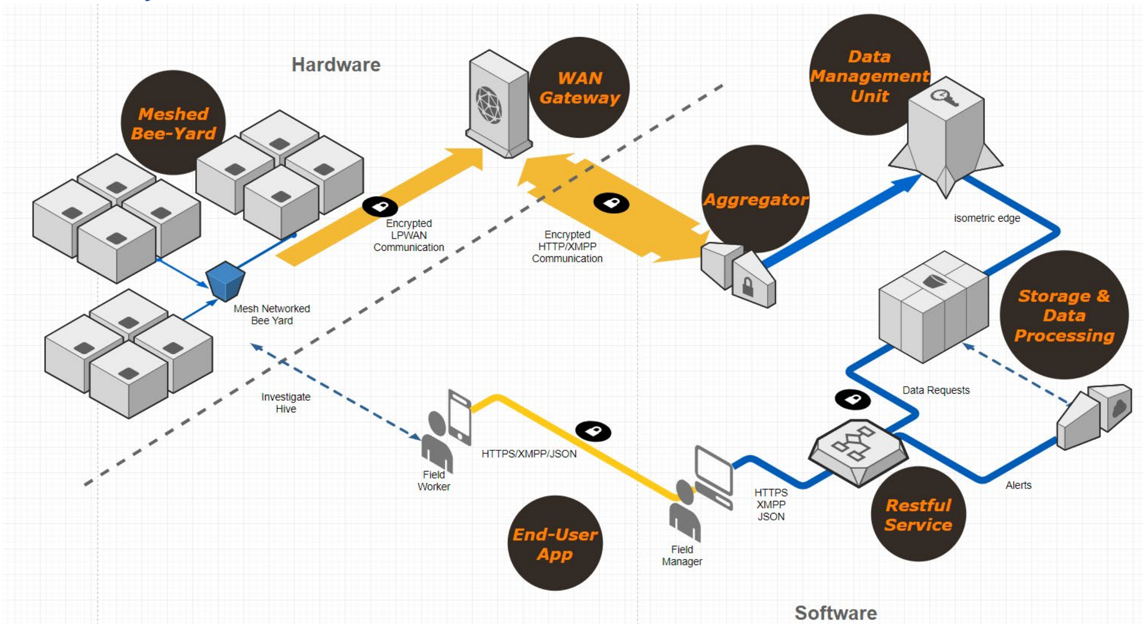


Figure IV-1: Technology in Development Overview

Project Bee will be focused on the hardware portion, as defined in the illustration above. Accordingly, a private server will be used to emulate the entire software portion. This includes the web application aggregator and data management unit. Various cloud-based tools will be researched for their feasibility, given the time constraint required to complete the design. While a PCB will be developed for the hardware tracking device, the WAN Gateway will be made using off the shelf modules which are compatible with some single-board-computer devices such as the Raspberry Pi. The gateway will also use wired ethernet to communicate with the internet. In an ideal scenario, a gateway may feature numerous ways of connecting to the internet, notably, cellular, Wi-Fi, satellite or one of the many IOT standards being deployed.

IV. Related Work

I. Research Papers

Communications Technology as Applied to Honeybees

While there is research available which attempts to track individual hives, some projects focus on tracking the individual bees themselves. This type of data collection contains drawbacks such as price, lack of reliability, and uninformative data. Tracking individual bees is mostly done with different types of radar. A common example of this is harmonic radar. Harmonic radar is only useful for tracking one specimen at a time. Considering bees do not all pollinate in the same area as each other at the same time, tracking one individual bee would be insufficient for attaining information such as pollination routes, etc. Considering that these methods are also extremely expensive to create and reproduce, Plan Bee elected to not follow that route for the current project. There is a focus on obtaining useful data, and an excess of information would be more confusing for apiarists.

Other areas to consider are separate ways to remotely access the data collected. A paper on System Architectures for Real-Time Bee Colony Temperature Monitoring (System Architecture for Real Time Bee Colony, n.d.) goes in depth into many possible ways of uploading data from a source. Plan Bee is focused on remote data extraction; because of this, one specific idea stood out in this article. This is the idea to upload the data attained from each individual hive to an interface device, such as a Raspberry Pi or Arduino. The data would then be transmitted from this device to a server, such as the cloud, using the internet. An example of this is given below.

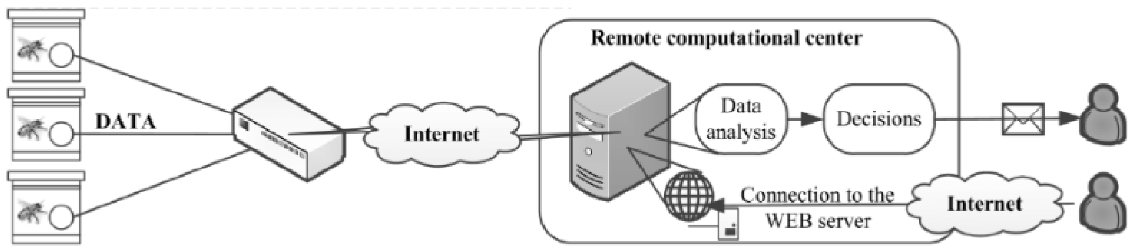


Figure IV-2 Implementation Detail

Plan Bee's design is very similar to this idea. Plan Bee, however, will be using the Texas Instrument CC1352R included on a prototyping board for the gateway. The Simple Link Texas Instrument board will be bought for testing; Plan Bee will then create a board based off this design. Using this technology will essentially be like creating a home-made Raspberry Pi. Some prototyping boards will include sensor arrays; these boards will communicate with the boards used for gateway connection via the CC1352R chip included on each board. Including these chips on all boards will create a mesh network. This will help with an autonomous set-up of yards; yards will not have to be manually designed. All these boards will also include GPS so that each individual hive is trackable no matter what happens.

The connections to the gateway are an important process, but the gateway will also connect to the server using cellular networks. The server database Plan Bee plans to use is Amazon AWS. This system is cheap to use and is a powerful data processor that even includes artificial intelligence tools. The data will then be processed on this server. A webpage or hosted application will then be used to help make the data easily accessible to the customer.

II. Prior Art

for Beehive Monitoring Systems

US Patent 20170079249 discusses how certain conditions inside and outside of a bee hive can be monitored and collected using a control system. This data can be compared to reference points of information to determine the meaning behind the data gathered. Measurement of the number of bees entering and leaving a hive is one of the variables monitored, as well as environmental conditions the hive faces. A variety of sensors can be attached to the control system for multitudes of data obtaining purposes, such as temperature measurements and humidity calculations.

US Patent 20070224914A1 discusses how the acoustics of bees and their interaction with the surrounding hive can be monitored and analyzed to determine if there are any airborne toxins affecting the bees. This idea has been patented since 2006; the patent uses specific statistical data to classify certain acoustic sounds that can be affected by air compound. These classifications can lead to different determinations about the conditions of bees in a hive and the air surrounding those bees. Speculations can be made, such as whether the bees are being affected by mites, or whether the bees are mating. Sounds are sampled for a period using the patented device; these sounds are then translated into frequencies. The obtained frequencies can then be compared to other frequency data statistics to determine the meaning behind the specific frequencies collected.

for Establishing Groups of Internet-of-Things Devices

US Patent 20140241354A1 discusses how any devices that interact with the Internet of Things (IoT) may be connected to form a larger hub of interconnected devices. Communication between these individual groups of IoT devices becomes possible, as well as contact between the more similar devices, therefore expanding the access and abilities of all devices connected on this plane. As an example, one could get information not only about temperature and relative humidity in their house from one IoT sensors group and retrieve information about their electricity usage through IoT-enabled breaker group, accessing all this data from the comfort of their iPad, anywhere in the world. Criteria is put in place to help make the devices that connect reasonably useful to the consumers using the IoT in this way.

It is worth noting that the use of this patent is almost unavoidable when constructing a recent technology. Having the inner workings of the device Plan Bee is building interconnected would be extremely useful to the finished product. Saving time and money, this patent would really help improve efficiency.

III. Market Products

Beehive GPS Trackers

There are several comparable products to our own communications template. FindMyHive offers a similar hive tracking product, with distinct differences involved in the technological build. FindMyHive only offers GPS tracking of hives, and uses a 3G network connection utilizing GSM standards, to cell towers to transmit data to the system connected over the internet. The distance that the trackers can be from any singular cell tower is thirty kilometers. We expect to have a cell-tower-independent data retrieval solution so that apiaries that are more off-grid can still collect data when needed, without having to go directly to the site. Also, FindMyHive only reports GPS tracking once per day. This is not a helpful preventative measure for things like theft, which Plan Bee anticipates addressing with the prototype design. Find My Hive does have a motion activated perimeter for triggering an alarm, but this is an expensive solution that can be improved upon. Any motion could trigger this sensor, causing many false alarms. Plan Bee expects its device to send a notification to the customer when it is moved too far from the originally specified location. FindMyHive is also not offered in the United States to date.

BuzzTech similarly utilizes GPS tracking to enable periodic location updates to the apiarist. The major problem with their solution, according to Steven Eisele, is the size and the cost. The size of the device is larger than absolute necessary and inhibits honeybee movement around the hive. This will cause swarming, at some point. The cost of implementation, according the BuzzTech, is $30 USD per hive per month, more than the cost of food for tracking of just one metric – location. BuzzTech is also only available in the New Zealand market due to several international standards for communications and GPS tracking that inhibit new market entry. We offer several improvements, firstly being available in the United States. We will also be implementing our beehive-aimed communications template in a much smaller package, that will be easily retrofitted on the side of commercial hives; our template will be much less costly per month and have a similar one-time fee for order. And we will offer many metrics rather than just GPS. As well, we are using different standards for communications, location services, and additional metrics for security monitoring.

Save Bees products are focused on preventing hive-theft. The anti-theft device is designed to conserve battery by staying off until it is moved in the slightest, using an accelerometer; the device will turn on and notify the customer of the movement. This is an interesting take on anti-theft because of how long battery life lasts under cited conditions, but there are a few potential problems worth considering. This technology is too sensitive to be practical; if the hive is jostled in the slightest it will falsely alert the consumer. One solution to this problem that we are integrating, is an algorithm that integrates the accelerometer with the GPS to determine if the hive is not only vibrated, but relocated more than a trivial distance, yet to be determined. Our devices would not be planted in every hive necessarily. If apiarists could not monitor every hive, the security is not foolproof.

Beehive Health Monitoring Solutions

There are Hive tracking systems in the industry that include much more than GPS and anti-theft precautions. Probably the most similar to Plan Bee's design is Solution Bee's product, the 'B-ware HiveHub' monitoring system. This American based product has similar data transfer tactics to Plan Bee's; the data can be collected with a solar powered device that is then uploaded to a type of gateway for access. This is very much like Plan Bee's design, except the distance the data can be accessed is much greater for Plan Bee. Solution Bee can only access the data up to 1000 ft., or a 30-meter range, using Wi-Fi. This is a much smaller distance than Plan Bee's proposed distance, which is around 10 miles. Solution Bee's 'HiveHub' is also very large and therefore bulky. Such a design is not very efficient to apply to every single hive; Plan Bee is focused on creating a more compact and versatile design that is more equipped for lengthy outdoor use. This Solution Bee product is also much more expensive than the marketable design Plan Bee is creating. Over-all, Plan Bee has a step-up on the technology and overall engineering of the hive tracking process.

Hive Tracks provides a user-friendly interface with an aim to simplify hive management. An interesting idea that Hive Tracks puts forward is for the data collected from each device to be uploaded to a communal database. This would allow the apiarists to see multitudes of data, helping not only themselves and their hives, but the apiary community as well. Collecting this much reliable data would help the research community understand why bees are struggling to survive. In the future, Plan Bee would like to utilize this basic concept as well; bee survival is a priority in a market that relies so heavily on these dying animals. This product is also more reasonably priced than other available hive tracking technologies on the market. Hive Tracks puts up a good front, but it is extremely difficult to access essential information about what the product really does. Data sheets are not available for consideration before purchasing the product; trial and error seems to be the only way to get a good feel for the system the company offers. Customers in the past have reported a glitchy mobile application system, as well as missing key features such as a to-do for apiarists.

Beehive Metrics Tracking Software

Hivetrack – not Hive Tracks, which was discussed earlier – offers its user a free platform to record inspection data. Data entries are available for queen color, type and existing hives, for hive type and queen relationships, for queen family tree and inspection history of the hive, including feed and medication, weight and honey production. This is the most basic way to utilize technology to keep track of apiaries, none of the processes have been streamlined. We aim to streamline these data-recording processes and use a much more advanced and modular (customizable) data dashboard, setting us apart from Hivetrack.

Bee Cloud pride themselves on their easy to use software. Allowing users to change around yards with ease and manage many several aspects of data organization is extremely useful. Plan Bee also expects to have a very efficient and user-friendly interface for organizing electronically obtained data. Combining software design aspects like Bee Cloud has with more physical properties, such as Buzz Tech's hive trackers, is the most well-rounded technique for helping customers manage their apiaries with ease. Improving on both techniques is also a priority for Plan Bee. For example, Bee Cloud does not yet connect to mobile devices or tablets. Plan Bee's data collecting technique should allow mobile devices to connect to the data infrastructure directly.

V. Requirement Specifications

I. House of Quality

The illustration below contrasts the project's engineering requirements against a set of marketing requirements. The notable engineering requirement were limited to a set of 7 items which were considered crucial in accomplishing the design priorities set forth by this document. In contrast, the marketing requirements were obtained by inquiring about the most regarded features from actual beekeepers seeking such a product. Given this information, the house of quality was developed to underscore the relationship between engineering development capabilities and consumer need. Understanding this diagram will further align the project goals with that of the end-user.

The qualities we have decided to focus on for marketing requirement specifications (customer demanded quality) include data latency, personalized dashboard, event reporting alerts, ease of assembly (installation), long-range, high power efficiency, long-term durability, weatherproofing, and offering a low-cost solution. For engineering requirement specifications (voice of the company engineering development capabilities), we have chosen to focus on range of communications, peripherals scalability, rate of communications, product dimensions, power consumption, weather resistance, reproducible unit cost as given through itemized quote, and data latency for updates. Toward the end of this section we also explain these qualities, both marketing and engineering requirement specifications, and how we plan to demonstrate these end-product deliverables on presentation day to a closed audience.

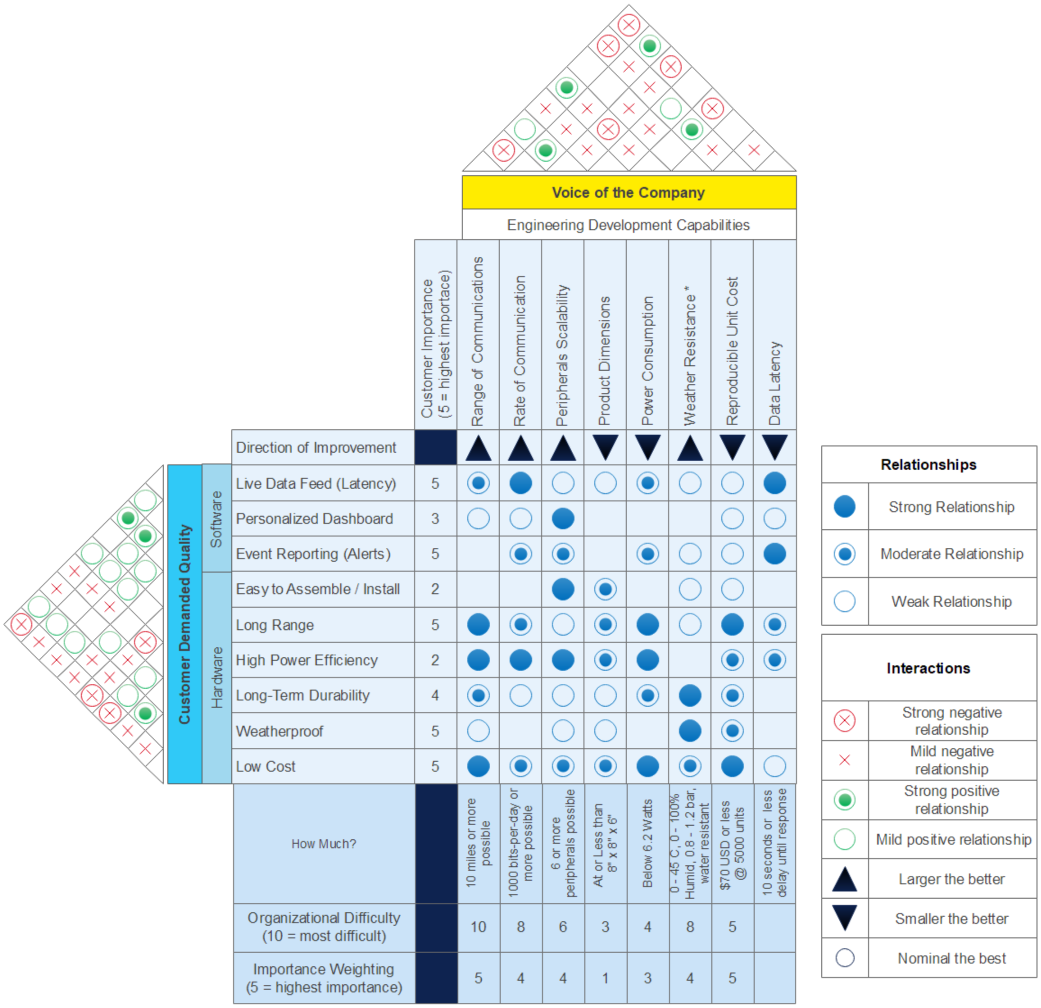


Figure IV-3: House of Quality

II. Engineering Specifications

Range of Communications

This defines a set distance by which a data-collecting device will be able to communicate with a network gateway. Given that bee-yards are usually placed within a mile of each other, 10 miles should be enough to maintain communication across yards. This allows the placement of network gateways at centralized locations to collect and deliver endpoint data to a remote server. The communication distance of 10 miles will be achieved by using a wireless transceiver capable of communicating using the Sub-1 GHz SIM bands. Current devices capable of operating across said bands can maintain a robust enough connection which is able to pierce walls and other forms of obstructions, even in the most urban environments. In contrast, hives are usual kept in rural environments with little obstructions other than trees, which has negligible impact on the range of a SUB-1 GHz signal. This should reinforce the possible range and robustness of the endpoint device. Additionally, while the range of communication is inversely correlated with response-time and cost, the data-rate may have the most significant impact on the range. As a result, a clever balancing act will be considered in selecting the data-rate.

Rate of Communication

Because data-rate and range are inversely correlated, a larger communication range dictates that the signal could produce a significantly lower data throughput. As a result, a set data-rate must be defined to accommodate the expected traffic which may be transmitted. Both digital and analog sensor data are expected to be sent across the wireless network. In addition, endpoints incapable of communicating with a gateway may seek to delegate the transmission of their specific data to another endpoint which does have a connection. Correspondingly, the Implementations and Standards Section analyzes the required data that is to be recorded and estimates the rate of communication as presented in the illustration above.

Peripheral Scalability

This declares the maximum number of external sensors which can be attached to a node. The node will be able to make such connections using a RS485 hardwired interface or wirelessly through Bluetooth. A maximum of 4 devices will be attachable over the hardwired interface while a maximum of 8 devices will be attachable over Bluetooth. While the limitation for the Bluetooth interface can be considered arbitrary, the limitation for the hardwire interface provides ease of communication and cable management when working with a hive. This will ease installation ease and hive access.

Product Dimension

The project dimensions are dictated by the allowable area between a hive. It should also be discrete and attached enough to not be displaced if the hives are moved, by weather or by trespassers. We want the product dimensions to be close to nominal, but no larger than 8” x 8” x 6”. This is also inversely related – a limiting factor – to the peripherals scalability, antenna size, weather resistance and power housing.

Power Consumption

Although we are uncertain of the maximum power requirements at this time, we are limited by the cost and quality of power being supplied through solar panels which will be mounted to the top of the hive. The most power we can plausibly afford per unit, considering the cost of load cells and other included components per device, is 6.2 Watts of power supplied. Less power consumption is preferential because this will reduce the cost of per unit, reduce product dimensions, and allow more room for weather resistance measures, communications components. Reducing power consumption however, may also limit us in final peripherals scalability.

Weather Resistance

The nature of the placement of the device on commercial beehives will require weather resistance. More weather resistance is better; increasing the weather resistance will also limit our area for communications components, power components, and of course peripheral scalability. The bare minimum of weather resistance should be able to protect the end-product from tropical summer to northern winter temperatures (0 – 45 degrees Celsius), low-to-high humidity 0 – 100%), heavy rain and snow, and weather depressions (0.9 to 1.1 bar of pressure). These should not be hard to accomplish. There are many national and international standards for our weather enclosures. Weather resistance enclosure standards are discussed further in the Standards section.

Transmission Latency

The response time ("Latency") of our online application, which will be accomplished via a website or app-store application, will need to be low. The lower the response time, the faster apiarists can see updates in their dashboard. We can feasibly accomplish a data latency of 10 seconds or less between updates to make the data appear as though it is "live". Device-level, this will depend upon a high rate of communication, and upon communications distance because this will need to be high enough that the server will pull the data from any yard. Latency is also negatively correlated with decreasing power consumption.

III. Marketing Specifications

Live Data Feed

The latency of the data that we will be sending from our device to the user (apiarists) live must be low. As specified, it must be able to transmit every 10 seconds to maintain a live update status. This latency can be affected by the power and rate of communication, but also by proximity as it needs to work within the desired 10 miles along with a preferred, optimal data-rate.

Personalized Dashboard

A dashboard interface that will carry all important sub-fields that the user might want to have available. This will be a personalized tool that will not only list out the preferred and most important hive characteristics of the time, but it will also be updating to give a live feedback of every yard connected. This will also be where apiarists can receive and view push notifications that will be based from real time alerts gathered by the sensors of the device.

Event Reporting

These alerts are linked to and demonstrated in the personalized dashboard. Our device should have the capability of detecting any abnormalities that may be present using one of its many sensors. The alerts should be sent with as low a latency as possible and must be able to keep updating and informing the user of what the sensors are picking up. We expect to create a product that will warn things such as change in weight, detection of a certain unwanted chemical, disturbance from outside forces such as wild animals, etc.

Ease of Assembly & Installation

Our product will be one that will be easily accessible and usable by any user that has zero experience with electronic technology. Our model will consist of an easy to install main control unit, as well as easily attachable units to the sides of the hives that will only need to put around two cables with small sensors at the ends inside of the hives. Since the communication will be running through sub-1 GHz communication, then this will also make the device easily accessible when trying to add other sensors to collect more data.

Long-Range

Our device is expected to cater to the needs of users that are trying to monitor their hives from far away distances. The distance of 10 miles is considered as appropriate proximity to be getting full information on the state of the hives at the fastest time possible. There is also the fact that these long distances are also remote areas where most signals do not reach, and the user must be able to get all information……

High Power-Efficiency

Certain factors such as data rate/latency, distance, and maybe even sunny days all could affect power efficiency. The device will use the most optimal panels that will power a battery which will efficiently transmit power into the board. Since all power will be solar provided, the main expense for the user will be that of a battery which tend to last long for these models. To make the battery last longer, the power consumption can be reduced.

Weatherproof

The user will need the device to work in regions where it will be exposed to the regular elements as well as extremities of the Florida region. The device will have to deal with cases such as hurricane winds, extreme heat, and wild animals such as bears. The casing it will come in should provide enough safety.

Low Cost

This product should be as low cost to the consumer as possible and this will be made possible due to the easy integration and low-maintenance, expanding nature of the product, as well as its long-lasting power life-time through the combination of solar panels and low power batteries. The costs will be further reduced by having the base communications template and adding only necessary sensors suited to monitor each individual beehive.

IV. Proposed Verification Techniques

We are going to record video of the working functionality of the following engineering constraints: 10+ miles Range of Communication, 1000+ bits/day Rate of Communications, 6.2 Watts or less Power Consumption. Range of Communication will be demoed by showing GPS location of the hive transmitter and the cellular phone receiver and showing live data tracking being pushed at this range. Similarly, we can push as much data as possible at this range and record how much is pushed over a 24-hour period. We will measure the power consumption using a voltmeter at the input and ground of the communications device.

Weather resistance against the elements will be verified by submerging the weatherproofed device under 3 meters of water for 10 minutes, by leaving the weatherproofed device in the freezer for 10 minutes, by 10 minutes of 45 degrees Celsius in a scientific oven and then testing the working functionality of the electrical components of the communications device inside with a multimeter.

Within the Financials section we are also going to include a running tabulation of incurred costs, with receipts, and include at least three price quotes for 5,000 units of our communications device, to verify $70 USD or less unit cost at the quoted number of units.

We are going to verify, the day of the demonstration, the automated Mesh-Networking in close-proximity using ZigBee, Range of Communications and Data Acquisition, Data Latency, and Product Dimensions. Automated Mesh-Networking will be shown by bringing two bare-communications devices close together (within 10 feet) and showing within the application that they have "meshed" (grouped) into a single yard. Range of Communications and Data Acquisition will be shown through video-conferencing with a person next to a hive 10 or more miles away, showing that when he plays with the sensors in the hive, the informatics in the application change. Data Latency will be shown with a timer, that less than 10 seconds after he played with sensors, the informatics have changed ("Live Feed"). Product Dimensions (in enclosure) will be shown to be less than 8" x 8" x 6" using a standard 12" ruler.

Security of the transmissions can also be shown by presenting showcasing a sniffing example using a software-defined radio receiver. This will simulate the attempt to achieve a man-in-the-middle attack on our communications end-product. They will not be able to decipher the data being transferred.

VI. Power Block Diagram

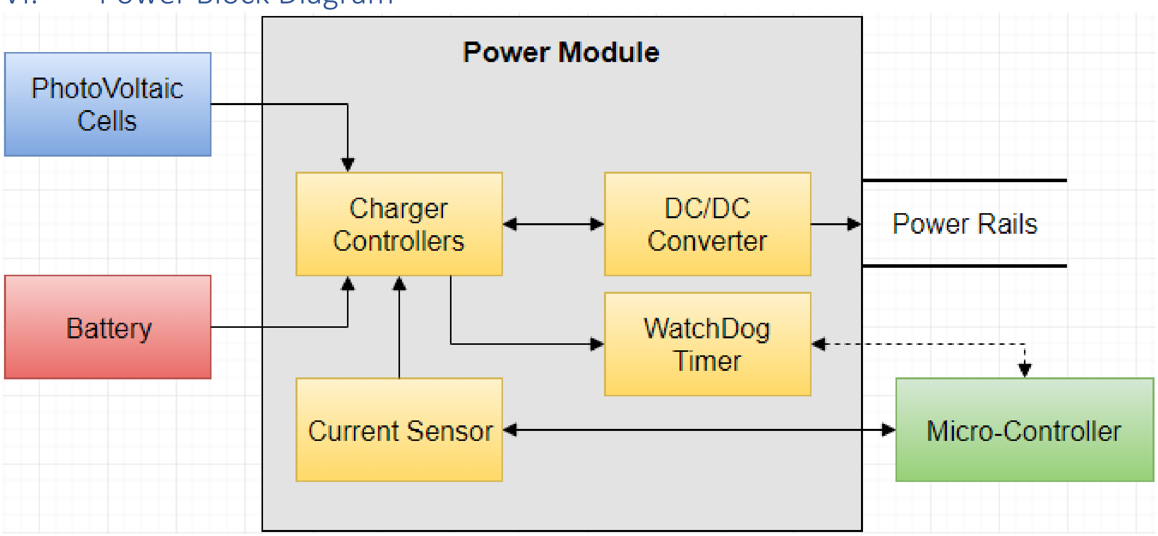


Figure IV-4: Power Interface Block Diagram

The Power Block Diagram figure above shows the power interfacing block diagram, illustrating how the Power Module helps the photovoltaic cells (PVCs) and specialized LiFePO4 battery to supply power to the Microcontroller Unit (MCU) and Power Rails to supply power to the peripherals (including sensors). The charge controller (trickle charger) will convert the PVC power to DC to charge the battery. The current sensor (controller) will output ultra-low current required for the MCU and the watchdog timer will ensure that software is terminated if it malfunctions, preventing erroneous looping. The DC/DC Converter will be used as a step-down converter to achieve 5 V and 3.3 V for various peripheral sensors. We may also need to add a stepdown to 1.8V as well for some of the onboard sensors. It is also worth noting that the CC1352R which we have chosen as our Wireless Microcontroller Unit (WMCU) has an programmable onboard DC-DC Converter for stepping down voltages.

VII. Hardware Block Diagram

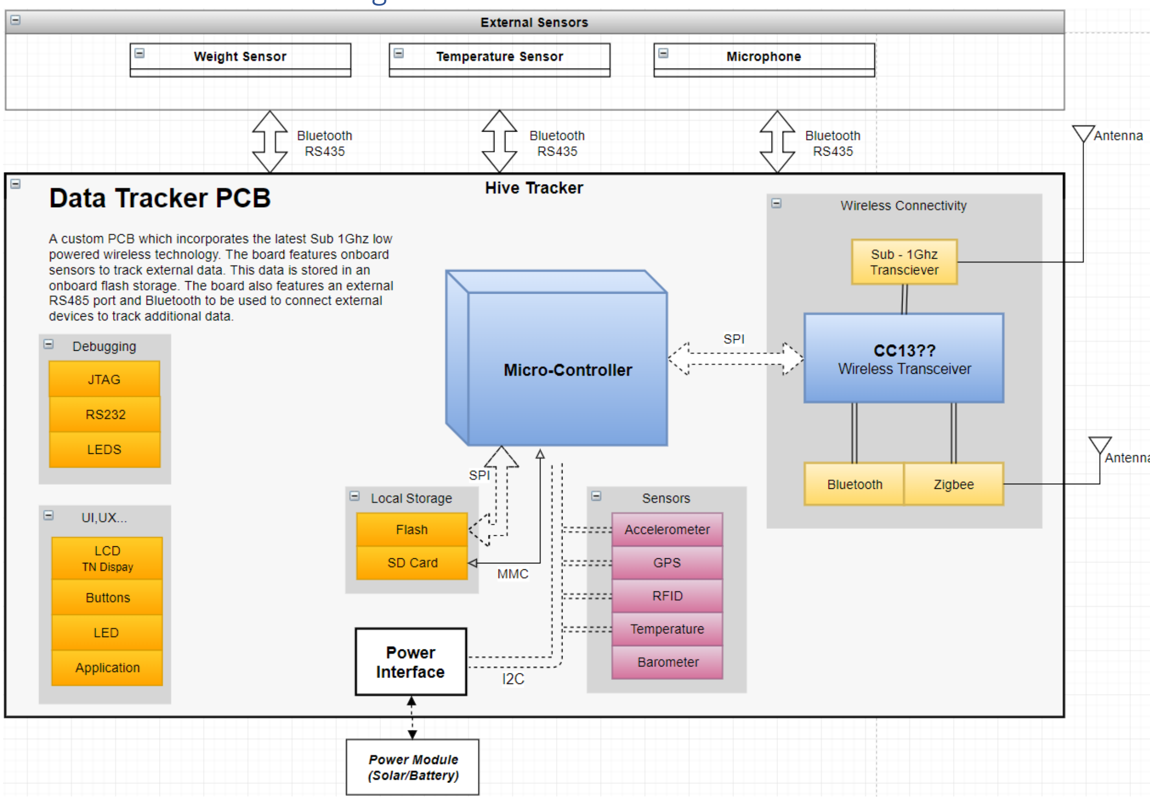


Figure IV-5: Hardware Interfacing Block Diagram Including Hive Tracker and Peripheral Sensors

The above diagram illustrates the target PCB design for the data-tracking unit. A standard Low Power Wide Area Network interface has not been selected yet. The pros and cons of the current standards are being considered. While the LoRa stack does provide a viable method of deploying a custom network, their wireless transceivers are lacking key features such as Bluetooth connectivity.

Texas Instruments, on the other hand, does provide transceivers with built in microcontrollers and Bluetooth; they also support various LPWAN protocols such as Sigfox. The listed sensors will be incorporated into the board along with an internal storage device. A debugging interface will be necessary for testing the ICs on the PCB. Although the standards for interfacing amongst the ICs have not been confirmed, modern chips do provide libraries which abstracts the low-level communication protocols. As a result, a widespread application could be developed and tested on a test board and ported easily to an arbitrary development board. A method for generating user feedback using either LEDs or an LCD display is also being considered. At this point it is too early to know how we are going to interface the CC1352R1 exactly, however we know that we will need an additional GPS module.

VIII. Software Block Diagram

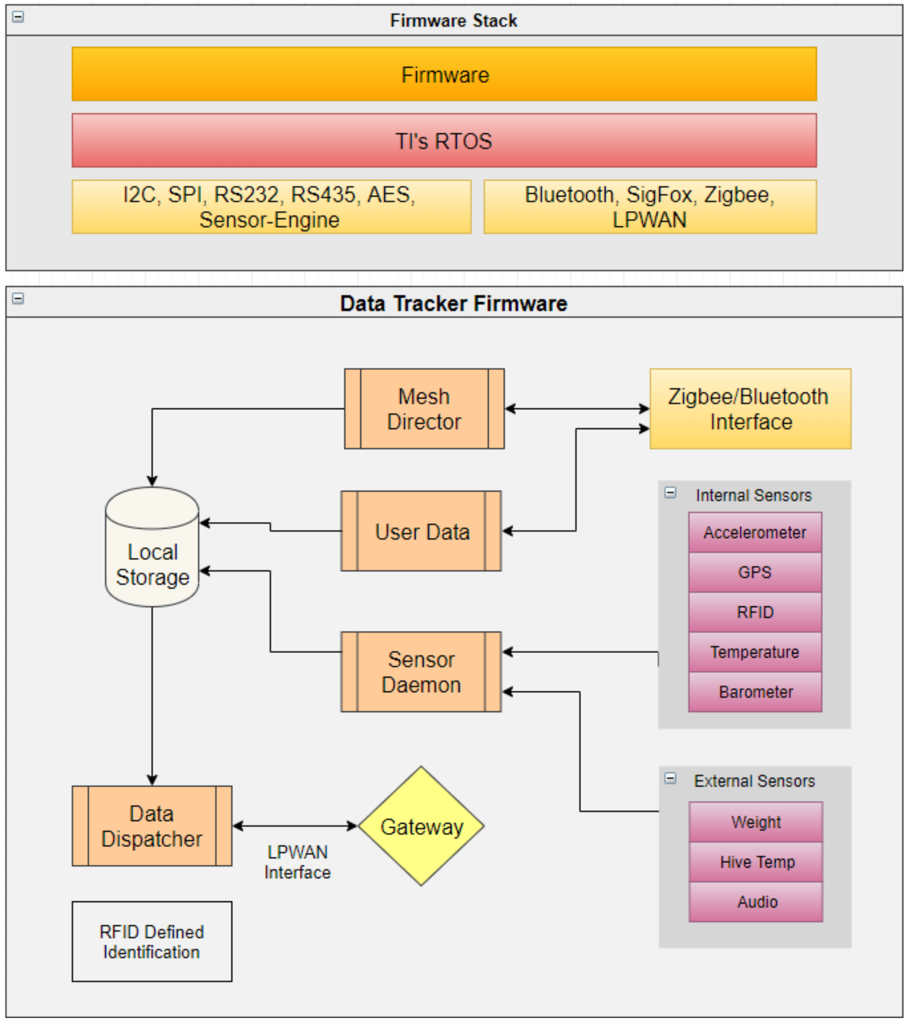


Figure IV-6: Software Interface Including Firmware Stack and Embedded Data Tracker Firmware

Texas Instrument provides a real-time operating system made specifically for IOT type devices. It employs various algorithms for reading and storing sensor data. Coupled with its Bluetooth and Zigbee interface, using Texas Instruments IOT platforms may ease the software development process. The firmware block diagram illustrated above describe the several types of tracked data. Internal sensors are those sensors internal to the tracking device (External to the hive). External sensors are those within the hive. The Bluetooth interface allows a user to connect to a hive using a phone, while the ZigBee interface allows that user to issue command to the entire yard at once. Data obtained from these data types are then dispatcher using the data dispatcher element to the gateway where it is translated to a TCP/IP packet and pushed to a web server.

V. Research and Part Selection

This section includes an overview of current technologies associated with our project, within the scientific research community and those already available in market. We have included technical research into the best technologies to use – arrangements of parts – to realize our project end-goals. Also featured here is the parts selection, where we look at schematics for the technologies presented, and discuss which parts are unavailable, available, and choose the most ideal parts to fit our project constraints. The orders, sample request forms, and other information for these chosen parts are included in the Administration section later in this document. Further documents including datasheets may be included in the Appendix.

I. Scientific Research

I. Power

For this project we chose to power using photo voltaic solar panels to give the most durability to our device as well help reduce the overall cost for both market and engineering purposes. After much analyzing of different powering specifications and charge systems, we have decided to go with one that offers a connection from the solar panel straight to a battery that is then connected straight to our board. The battery charges when it's reached certain threshold level. This version is preferred to one that still utilizes the solar panel when the battery is full instead of always relying on the charge of battery because it requires less components and helps with the ease of build while still providing enough charge. In the regular market, solar panels for projects such as these go the same as for large-scale residential use: prices vary directly to both area and watts generated per area. For this project we found it convenient to use a small 6.2-Watt solar cell that will not should give us more than enough charge.

The device will have battery then be connected directly to the boards to power them and this powering shall have to be regulated according to the how much charge is in the battery as well as how much power is being supplied by the solar panel. The board will also need to step down the voltage from the battery to that of the PCB. To choose a decent controller, two main ones were looked at since they are compatible with our communication system: the TI BQ24250 and the MCP73871. The BQ24250 offers some great features such as Selectable Battery Tracking Mode to Maximize the Charge Rate from Solar Panel Using DPM Feature, and 20-V Input Rating, With Overvoltage Protection (OVP), 1% Battery Voltage Regulation Accuracy, Current Charge up to 800 mA with 10% Charge Current Accuracy, Thermal Regulation Protection for Output Current Control, and Low Battery Leakage Current. It also offers MPPT capability which automatically helps get the maximum power point, at which the entire photovoltaic (PV) system operates with maximum efficiency and produces its maximum output power. However, the MCP73871 offers a much higher current capacity, higher integration such as internal loop compensation, more safety features such as Battery Absent Detection and Thermal Shutdown, as well as a different version of MPPT that helps the charger always pick pulling from the solar panel before the battery when available and helps charge both the internal power and the battery at the same time.

The charge at the regulation phases drops ensuring that a voltage is still present even when no current is available. This happens when the battery reaches below 1.2 Volts and needs to compensate the supply for its loss voltage. This system also protects the battery as any system such as this that lets its battery be completely depleted to 0 Volts will need to replace the battery.

II. Networking Topologies

Network topology is the “mapped” structure of a wired or wireless network that transfers data somehow from one place to another. These places are called “nodes”, and the connections themselves, as part of the communication theory topology, are called “links”. There are physical topologies and logical topologies. Physical topologies act as a transmission medium to link devices physically, whereas logical topologies are the way wireless networks communicate. Both types of networking topologies contain the same few types: ring topology, mesh topology, star topology, fully-connected topology, line topology, tree topology, and bus topology.

We will be using the mesh network topology in unison with Bluetooth in order to realize ZigBee protocols. The value of the meshed network in this case is that everything is proportional to the exponent of the number of subscribers which can be pulled from any member of the mesh network. Therefore, we can tell the number of hives connected via low-power Bluetooth to one-another and deduce which hives specifically are connected, forming a “meshed” yard. This is approximated by Reed’s Law. As you would assume, most nodes are interconnected via links to multiple other nodes, and a logical “mesh” pattern is formed in three dimensions.

We will be using the star network topology in unison with GSM in order to realize internet protocol standards to push data to a remote webserver. Our webserver will likely be either or a combination of Amazon AWS and IBM Watson. Star network topology is used where each network host is connected to a central hub by point-to-point connection – forming a 5-pointed star with a center point connected to each of them. The central node is called the “hub” which acts as a signal repeater, as our CC1352C transceiver gateway will, repeating signals to local area networks (LAN), which form the other nodes (access points). The logical connection between them is the link, in this case, and the fact that data transmitted from the hub can be decoded by ONLY the server we are using, sending data through an API Gateway (RESTful API).

III. Sensors Technology Applied to Honeybees

Weight

Weight monitoring is one of the simpler hive monitoring techniques, however it also provides extensive data feedback throughout the year. Measuring the weight of a hive over time can show whether the hive honey output is progressing or declining. This can give valuable insight on the health of that specific hive. Smaller weight changes each day can show when the bees come and go from pollination, or if there are any abnormalities with their behavior in general. Not only can weight measurement give details about the bees in a hive, but it can also provide information on what type of plants the bees are pollinating and when they flower. This is all obtained by weighing each hive individually for an extensive period; when the weights are applied, they do not have to be operated from that point onward, meaning they are very easy to maintain.

Temperature & Humidity

Honey Bees prefer a certain amount of humidity in the hive; there is consistent data to support honey bees preferring around 45%-55% humidity within the offspring nest of the hive. A humidity monitor can accurately reflect how close the humidity percentage of the hive is to ideal values. Bees also prefer certain temperatures as well; combining humidity and temperature data with weight data can accurately project how well a hive is doing and whether these conditions are affecting the health of the hive.

Lumens

Light sensors can help accurately predict the length of a day as well as the weather conditions of each day. Knowing whether a day is cloudy or extremely bright is important because of the way honey bees can be affected. The cycles that are produces each year from the length of time the sun is up, as well as seasonal weather, can help predict how the honey bees will act during these conditions. This information can also help apiarists know when they specifically want to go out to the hives for maintenance.

Rain Gauges

Rain gauges are another important aspect of hive measurement. Pollination USA has specifically requested a rain gauge device from Plan Bee. Rain gauges can give accurate measurements on the amount of rain that has fallen in a specific area. A heavy rainstorm will provide bees with less time for pollination, as most bees will seek cover inside of the hive. Knowing whether rain is a reason for abnormal foraging patterns is significant. Rain can also increase flowering, which may lead to an increase in pollination. Apiarists also need to maintain the hives and make sure that they are not flooded with rain water; knowing how much rain has fallen will help determine if the hive is having any technical problems worth monitoring further.

Video Recording

Video Camera Streaming of yards can be useful for monitoring theft, as well as pollination patterns of the bees. Any abnormal activity could also be viewed from these cameras and uploaded to the customers device from the gateway. Pollination USA has asked for some sort of visual production device, whether that is a camera that takes a picture or short video during certain time intervals, or a constant video stream.

Census & Counters

Bee counters can provide data on when bees enter and leave the hive. Knowing how many bees are in the hive can provide intel on pollination times and routes, as well as how well the hive is doing based on the number of bees counted in a specific hive and compared to a baseline of an average healthy hive count. Restricting the bees' entrance and exit to one specific and rather narrow point may be necessary, but this would not affect the bees themselves if applied correctly. Video footage could even be added to bee counters so that more information can be obtained about what type of bees are entering and leaving, as well as if the bees exhibit specific behaviors, such as swarming.

Accelerometers

Collected data on the specific vibrations of a hive could provide invaluable information. Bees exhibit certain vibration behaviors depending on the condition they are in. Recording these vibrations over time and comparing them to the health of the hive could provide accurate date on what vibrations of a healthy hive sound like. This data would not only be helpful for apiarists but would provide research information as well.

In-Hive Microphones

Another great research technique would be in-hive microphones. Knowing how the hive sounds year-round, especially when unable to open the hive and see the bees (such as in winter or during other intense whether patterns, like hurricanes), would bring certainty to how well the hive is surviving. Building a portfolio of bee communication techniques would also provide information on the different ways bees communicate, and what specific sounds mean.

ASM Pheromone Sensors

Pheromone sensors can give intel on mating inside of a hive, as well as if any predators are moving in on the bees, such as mites. This could prevent hive loss and spreading of invasive mites, as well as provide information on how well the hive is producing more bees.

II. Technical Research

I. Cloud Computing & Web Servers

There are several available options for cloud computing and web servers, Amazon Web Services (AWS), IBM Watson, Microsoft Azure, Google Cloud and Salesforce Watson. The TI CC1352R RF communications chip uses SigFox and limits us to AWS and IBM Watson for web services as our cloud computing base and web server. These two offer somewhat similar capabilities around flexible computing power, storage and networking, as well as the addon of machine-learning and artificial intelligence processing available. To compare the two, we have implemented a table below.

Cloud Computing Services		Amazon AWS	IBM Cloud & Watson
Pricing	Type of Discount	Reserved Instances (RIs)	Monthly Pricing
	Length of Commitment Required	1 or 3 years	Monthly Commitment
	Range of Discount	Up to 75%	Up to 10%
	Other Documented Programs	RI volume discounts, Spot instances	Negotiated
T	Technical Approach &	Excellent	Very Good

	Cloud Computing Services	Amazon AWS	IBM Cloud & Watson
	Documentation		
	Service Level Agreements	Very Good	Satisfactory
	Management Approach	Satisfactory	Very Good
Performance	Past Performance / Uptime (Confidence)	High	Moderate
	Security	Excellent	Excellent
Offered Services	Computing	Yes	Yes
	Artificial Intelligence & Machine Learning Ability	Satisfactory	Excellent
	Storage	Yes	Yes
	Web Hosting	Yes	No
	Databases	Yes	Yes
	Networking & Content Delivery	Yes	Yes
	Migration	Yes	Yes
	Developer Tools	Yes	Yes
	Management Tools	Yes	Yes
	Media Services	Yes	No
	Security, Identity & Compliance	Yes	Yes
	Analytics	Yes	Yes
	Mobile	Yes	Yes
	AR & VR	Yes	No
	Application Integration	Yes	Yes
	Customer Engagement Tracking	Yes	Yes
	Business Productivity Tracking	Yes	Yes
	Desktop App Streaming	Yes	No
	Internet of Things	Yes	Somewhat – only Watson is supported for IoT applications
	Game Development	Yes	No
	Cost Management	Yes	No
	Marketplace Software	Yes	No

Table 1: Comparison of Cloud Computing and Web Services

The yellow highlights support what we would like to make use of from their offered services, whereas the green highlights fundamentals, or bare necessities. IoT Applications MUST absolutely be supported for packet delivery and web hosting. This is not the case for IBM Cloud – only Watson is usable for IoT applications – so we decidedly be using

Amazon AWS as our web server, web application hosting and for APIs. This is because of the ease of use and integration with Texas Instruments, and low relative cost compared to IBM Cloud. Although there is a time commitment required, the first year is free in our case, and our customer will likely not want to migrate cloud computing servers because new development and associated costs will be required. We may or may not add IBM Watson services as well, specifically for better machine learning architectures and performance.

REST – Representational State Transfer (RESTful APIs)

REST is a type of architecture for creating web services using standards and technologies that have already been developed, such as HTTP. REST does not require the customer to know that it is running or being used. This way, the customer doesn't need to know about the server and the resources on that server ahead of time. Anyone familiar with HTTP will have ease of use with REST, which is a greatly beneficial feature. Any coding language can be used with REST as well, adding to its adaptability with various consumers. REST is often compared to SOAP, which is another competing style of architecture.

II. Frameworks

Frontend

There are many available frontend frameworks for web development, based upon the three frontend languages we talked about in Standards: HTML5, CSS3 and JavaScript. We will need one of these frameworks for adding formatted and styled content, pages and animations to our web-based application. There are only a few good development frameworks that are modular enough to get the job done, though, and of these we are tasked with deciding which will be the least complex, and modular enough to fit our web application specific needs. The table below compares the complexity and modularity of these different frontend frameworks, as well as other tangible items so we can make the best decision on a frontend web-app framework.

Some of the important items we will need to consider are the creator/publisher (reputable or renowned publisher), release date (we want something new that will be supported for a long time to come), current version, popularity, core-concepts and philosophy that the frontend framework is built upon, framework (the more data we use the more AWS is going to cost), pre- and post-processors, icon sets, formatting, and others. We also want to consider keywords and documentation, customizability, and browser support.

Frontend Frameworks	Bootstrap	Foundation	Semantic UI	Pure	UIkit
Creator/Publisher	Mark Otto, Jacob Thronton	ZURB	Jack Lukic	Yahoo	YOOtheme
Released	2011	2011	2013	2013	2013
Current Version	4.0	6.0	2.2	1.0.0	3.0.0
Popularity (stars on GitHub)	121,374	29,956	39,364	18,183	11,604

Frontend Frameworks	Bootstrap	Foundation	Semantic UI	Pure	UIkit
Description	Sleek, Intuitive, Powerful... for faster and easier web development	The most advanced and responsive in the world	UI components based on useful principles from natural language	set of small responsive CSS modules	light, modular for developing faster and powerful websites
Core Concepts/Principles	RWD and mobile-first	RWD, mobile-first, semantic	Semantic, tag ambivalence, responsive	SMACSS, minimalism	RWD, mobile first
Framework Size	578KB (minified and zipped)	197.5KB (zipped)	806KB (zipped)	3.8KB (minified and gzipped)	326.9KB (zipped)
Preprocessors	Sass	Sass	Less	None	Less, Sass
Responsiveness	Yes	Yes	Yes	Yes	Yes
Modular	Yes	Yes	Yes	Yes	Yes
Starting Templates/Layouts	Yes	Yes	Yes, very basic though	Yes	Yes
Icon Set	Not included	Foundation Icon Fonts	Font Awesome	None, Font Awesome usable	SVG icon system, library for outlines
Extras/Addons	None bundled, but many third-party plugins	Yes	None	None	Yes

Table 2: Comparison of Frontend Web Frameworks

Thus far it appears that Foundation by ZURB is the best option, because as of this point in the analysis, it offers the most flexibility while still retaining low framework size, which will be needed for speed (data latency). As well, extras and addons are available directly from ZURB without need for third-parties. The core concepts exceed that of Bootstrap 4.0 as well, but further analysis is still needed to decide between BS4 and Foundation.

Frontend Frameworks	Bootstrap	Foundation	Semantic UI	Pure	UIkit
Unique Components	Jumbotron, Card	Icon Bar, Clearing Lightbox, Flex Video, Joyride, Pricing Tables	Divider, Flag, Rail, Reveal, Step, Advertisement, Card, Feed, Item, Statistic, Dimmer, Rating, Shape	None	Article, Flex, Cover, HTML Editor embed
Documentation	Excellent	Good, many additional resources available	Very good, well organized and separate website guides through implementation	Good, but basic	Good, but basic
Customization	Options for seperate files for grid system and reboot, easy customization with Sass, no online customizer	Basic GUI customizer	No GUI customizer, only manual customization	Bsic GUI Skin Builder (online)	Advanced GUI Customizer available
Browser Support	Latest, All	Latest, All	Latest, All	Latest, All	Latest, All
Licensed by	MIT	MIT	MIT	Yahoo! Inc. BSD	MIT

Table 3: Comparison of Backend Web Frameworks

Of these, we will likely be using Bootstrap 4.0 because it has the best documentation of all these frontend frameworks, and offers greater customization options, although not necessarily as accessible as the other frameworks. As well, the mobile-first and minimalism of BS4 offer better accessibility as an actual handheld application, which will work better for our apiarists for which we are creating this end-product to begin with.

It is also worth noting that Angular.JS will also be used. Angular JS allows the rending of website directly in the web browser during development, was developed and is published by Google. Frameworks can be either JS-based, or CSS-based, and the frameworks discussed above are all CSS-based, or a combination of the two. BS4 uses JQuery (a JS library) to add dynamic functionality to its web elements and can be used with Angular.JS. Angular.JS really allows for enhancement of HTML and adding more complex functionality and customization than BS4 alone offers.

Backend

Because we have chosen to use Amazon AWS webserver services, the backend of our web applications will all be API driven. Any framework we use will be decided by the AWS server necessities, APIs and operating systems that power the web application's frontend.

Backend frameworks here are libraries of server-side programming languages that construct the backend structure of the website. The software stack comprises of database, server-side (backend) frameworks, the AWS server and the operating system (OS). We have a few options for backend frameworks:

Server-side Frameworks	Ruby On Rails	ASP.NET	Django	Express JS	Golang (Go)
Description	10x faster than JS or Java based frameworks, and easiest to use web framework, according to website	open source framework for modern web app services that are fast, simple and scalable	"pluggability" for easy creation for complex implementation; don't have to repeat yourself	light web application framework to help organize into MVC framework and use any templating language	object oriented, but not in the usual way; simple pieces connected by small interfaces
Language	Ruby	.Net languages	Python	Javascript	Go, gc assem, C++
Express Builds	single, multi and hybrid	single, multi and hybrid	single, multi and hybrid	single, multi and hybrid	single, hybrid
API Supported	Yes	Yes	Yes	Yes	Yes
Templating Engines	Yes	Yes	Yes	Jade and EJS	Yes

Server-side Frameworks	Ruby On Rails	ASP.NET	Django	Express JS	Golang (Go)
MVC Pattern	Movel-View-Controller architecture	Model-View-Controller architecture	Model-View-Controller architecture	Model-View-Controller architecture	Multi-paradigm, procedural object-oriented, concurrent
Platform	None	.Net	None	Node.JS	None
Operating System (OS)	Cross-Platform	Microsoft Windows, Linux (limited support)	Cross-Platform	Cross-platform	Cross-platform
Unique Components	Ajax functionality	No	Ports to other languages	Express Generator (automatic)	Go language
Documentation	Excellent	Good, but online expertise available	Excellent	Good, but several tutorials available	Excellent, with added templates and videos
Size	2.9 MB	4MB	7.6MB	2.8MB	Compiles to binary, very small
Stable Version	5.2.0	4.7.1	2.0.7	4.16.3	1.10.3
Stable Date	2018	2017	2018	2018	2018
Initial Release Date	2005	2002	2005	2010	2009
Creator / Publisher	David Heinemeier Hansson	Microsoft	Adrian Holovaty, Simon Willison	T.J Holowaychuk	Robert Griesemer, Rob Pike, Ken Thompson
License	MIT	Apache 2.0	Django Software Foundation (DSF)	MIT	Google
AWS Supported	Yes	Yes	Yes	Yes	Yes

Table 4: Comparison of Serverside Frameworks

We will be using Golang in combination with SQL language. We have chosen Golang because it is significantly faster than other backend frameworks and dependent upon usage and not package size, and because it is more regularly updated and directly supported by

Amazon AWS. As well, it allows for faster development, which is exactly what we need in a time crunch, and runs directly on Google development environments, which we will be using as they are free, as well.

III. Firmware

FreeTOS – Free Real-Time Operation System

FreeTOS is a real-time operating system developed for embedded devices. It was designed to be as simple as possible; the kernel only relies on 3 C files. Like TI-RTOS, FreeTOS features a small memory footprint and support scheduling, thread priority, dynamic memory allocation, static memory allocation, and semaphores and other memory management units. FreeTOS provides rapid application development and a robust enough framework to maximize application portability and extensibility.

While Texas Instruments' RTOS is based on FreeTOS, TI-RTOS also boasts additional features which severely decreases development time and aid in testing and deployment. The unfortunate decision to relegate TI-RTOS to Texas instrument designed devices severely limits application portability and even upgradability. Because of the time constraints allotted to complete this project we'd decided to use Texas Instrument development environment to increase application development time. Such a solution would allow us to fully integrate wireless and wired communication methods with ease. Texas Instrument's also provides from a significant number of integrated devices which are fully supported and across their many software tools and integration layer.

Data-Tracker Firmware

TI-RTOS – Real-Time Operating System

Texas Instruments provides a low-level real-time operating system deliberately called RTOS, or real-time operating system. This framework contains a micro-kernel, along with drivers, middleware and a hardware abstraction layer (HAL). These components provide an approachable programming interface which abstracts the application software from the underlying hardware. This promotes portability and usability as code will be allowed to be used across a diverse set of microcontrollers. TI-RTOS. was written in C and exposes 90% of the features available for the micro-controller. This significantly accelerates application development while also shielding the developer from the assemble interface. Texas Instrument also provides the full source code to the public and does not require a runtime license fee. Texas Instruments also furnishes a vibrant community, with active Texas Instrument employees offering costumer support and advice.

TI-RTOS provides a deterministic micro-kernel which dictates that kernel code should be executed within a pre-determined period. This minimizes time spent within the kernel code and allows application code to be executed more actively: as a real-time operating system should. The kernel also offers preemptive multi-threading and synchronization services. Such service allows application developers to employ multithreaded application with the valid tools needed to manage resource sharing and processing scheduling. TI-RTOS also offer device drivers which can initialize various micro-controller features such as UART, I^2 C and ethernet. These drivers provide an approachable application-programming-interface for interacting with said devices and managing device resources. (Texas Instrument, n.d.)

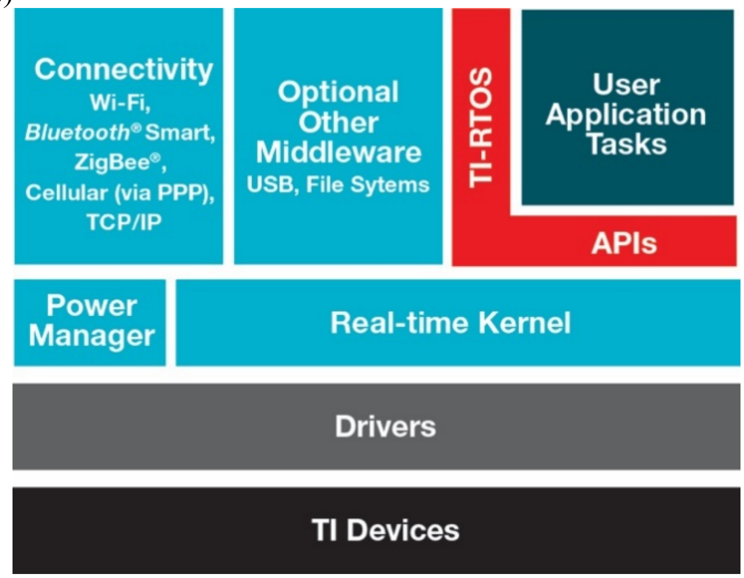


Figure V-1: TI-RTOS User Application Stack

TI-RTOS is also integrated with Texas Instrument wireless connectivity stack, provided by Texas Instrument's wireless transceivers boasting Simplelink capabilities. This feature includes Wi-Fi, Bluetooth, Zigbee, and Texas Instrument's Sub GHz network solutions. Additionally, TI-RTOS offers a comprehensive set of power saving features which permits application developers to leverage underlying hardware capabilities of their chosen microcontroller. (Texas Instrument, n.d.)

RTOS		TI RTOS by TI	ARMRTX by Keil	FreeRTOS by Real Time Engineers Ltd.	Micrium by Micrium
Scheduler Type		Pre-emptive	Round-robin (default), Pre-emptive, Cooperative	Pre-emptive, Cooperative	Pre-emptive
CMSIS Compliant		No	Yes	No	No
Services	Type I	Tasks, SWIs, HWIs, Software Timers	Tasks, SWIs, Software Timers	Tasks, Co-Routines (task share stack), Direct TASK notifications	Tasks, Software Timers
	Type II	Semaphores, Mutex, Mailbox, Events	Semaphores, Mutex, Mailbox, Events	Software Timers, Semaphores, Mutex, Events	Semaphores, Mutex, Mailbox, Events

RTOS	TI RTOS by TI	ARMRTX by Keil	FreeRTOS by Real Time Engineers Ltd.	Micrium by Micrium
Interrupt Latency	Zero latency interrupts supported (115 cycles for ISRs with TI-RTOS calls)	Zero latency interrupts supported	Zero-latency interrupts supported	Unknown
Device-Specific Support	Systick, Interrupts, Exception Handling	Systick	Systick	Systick
Device Drivers	SPI, SPI-SD, I^2C, UART, Timer, RTC, Timestamp, Watchdog	None	None	None
Flash Size	4K to 10K	Less than 4K	5K to 10K	
Low Power Support	Tick Suppression, Device-specific Power Manager	Tick Suppression, Device-specific support coming soon	Tick Suppression, Pre- and Post-sleep macros for customization, Idle task hook	Idle task hook, Tick suppression coming soon

Table 5: Comparison of Real-Time Operating Systems Firmware (RTOS)

TI-RTOS provides an extensive list of features and capabilities which eases application development for Texas Instrument micro-controllers. To increase application efficiency, Texas Instruments RTOS was written in a modular manner, whereby unnecessary features could be omitted from the final application build. This permits developers to further stretch the limited resources available on such restrictive platforms. We have made a table comparing RTOS above.

IV. Communications Technologies

LPWAN – Low Powered Wide-Area-Network

The project specifications require that the data collecting device be able to communicate with a user deployed access point at a minimum of 10 miles, while also maintaining cost effectiveness and power efficiency. Standards such as Bluetooth and Wi-Fi are incapable of communicating at the necessary range, while cellular standards require significant cost and limited coverage. This has led to an emerging market segment whereby low bandwidth devices are developed to communicate across very large distances using cost effective subscription models.

Emerging standards are being promoted to meet the growing demand for such device. Since the varying market segments, require different marketing specification and consideration, each standard should be meticulously examined for their advantages and disadvantages. As a result, this section seeks to decipher the best solution, which would supply the most robust and reliable connection while meeting the technical requirement set forth by this project. Listed below are the leading long-range wide area network standards currently being deployed to satisfy the growing LPWA demand.

Weightless

A proprietary wireless solution with open hardware and software stacks. It relies on a central gateway device to manage 1000s of connected devices. The primary factor differentiating such standard from the others is the wireless baseband used for communication. Weightless proposes using the wavelength radio transmission of unoccupied television band. Weightless also proposes AES encrypted end-to-end communication and guaranteed security and anonymity (Weightless, n.d.).

NB-IoT – Narrow-Band Internet-of-Things

This is a Low Powered Wide Area Network standard being developed by 3GPP; an organization with oversight over similar wireless cellular technology such as GSM, UMTS, and LTE. NB-IoT relies on a subset of the finalized LTE standard with a focus on low-cost, long range indoor coverage with high connectivity density. It operates on a single narrow band of 200 KHz and uses Orthogonal Frequency-Division Multiplexing (OFDM) for its downlink communication and Single Carrier Frequency-Division Multiple Access for uplink communication (OFDMA). (Huawei, n.d.)

SigFox

Currently the leading Low Powered Wide Area Network provider. Sigfox employs a proprietary wireless stack which runs on the Industrial, Scientific, and Medical ISM bands. It relies on thousands of carrier-deployed gateway devices, scattered across multiple contents; Sigfox professes it has accomplished "global coverage," but such claim has been regulated to urban environments. Sigfox, inherently is a low-cost subscription service whereby connected devices are meant to transmit small sets of data, on the order of 100s of bits per day. Sigfox subscriptions also run at about 1 dollar per year per device.

LoRaWAN – Long-Range Wide-Area-Network

An open-sourced low powered wide area networking stack designed to provide long range communication between device and gateway. While there are commercial providers expanding coverage by deploying gateway devices across the content, private users are also able to deploy their own gateways and avoid recurring subscription costs. Where the LoRa stack shines is in its open-sourced protocols and implementation details. LoRa Gateways could be created with off-the-shelf parts and deployed anywhere and maintain a significant range. This is accomplished by utilizing the Sub-1 GHz ISM bands. Additionally, LoRa relies on a chirp spread spectrum modulation scheme for both uplink and downlink connections to provide expanded range and reliability. This is all accomplished by wireless transceivers, provide by Semtech, the company behind the LoRa standard.

RPMA - Random Phase Multiple Access

This is a proprietary Low Powered Wide Area Network stack developed by Ingenu. RPMA relies on the ubiquitous 2.4 GHz ISM bands but can prove significant coverage using minimal access points. Using high gain antennas, Ingenu show that a single access point has the capability of hosting approximately 400,000 devices and is also capable of covering as much as 30 square miles. A down-side of this technology is cost, while not advertised, Ingenu seeks to market their hardware to large enterprises, and restricts private network deployment (RPMA access points are only rented to organizations within areas poised for coverage).

Texas Instrument SimpleLink

Texas instruments Low Powered Wide Area Network solution, which relies on the Sub-1 GHz ISM band. Unlike similar technologies, Texas Instruments utilizes narrowband channels, relying on less than 25 KHz, which has shown to strengthen signal co-existence performance in highly congested areas. In addition, Texas Instruments utilizes Gaussian Frequency-Shift Keying modulation to greatly extend the range of communication. The entire networking stack, is also open sourced, allowing users to deploy their very own private network without additional costs. Texas Instrument also offer a wide range of microcontrollers with integrated SimpleLink transceivers. These system-on-a-chip solutions also feature additional wireless standards such as SigFox, Low Powered Bluetooth, and Wi-Fi. Texas Instruments retains the record for the furthest LPWAN transmission, at over 90 kilometers.

Name of Standard	LTE-Cat M	SigFox	LoRaWAN	Ingenu RPMA
Frequency Band	Cellular	868/902 MHz ISM bands	433/868/780/915 MHz ISM bands	2.4 GHz ISM bands
Channel Width	1.4 MHz	Ultra-narrow band	64x125 KHz, Modulation: Chirp Spread Spectrum (USA)	1 MHz (40 channels available)
Range	2-5 km urban	30-50 km rural, 10 km urban, 1000 km LoS	10 km rural, 2-5 km urban	greater than 500 km LoS
End Node Transmit Power	100 mW	10 uW to 100 mW	less than 27 dBm (USA)	up to 20 dBm
Packet Size	Defined by User	10 bytes	12 bytes	Flexible 6 B to 10 kB

Name of Standard	LTE-Cat M	SigFox	LoRaWAN	Ingenu RPMA
Uplink Data Rate	Around 200 kbps	100-140 bps	900-100 kbps (USA)	Up to 624 kbps
Recurring Costs	Medium	Medium	Low	High
Download Data Rate	Around 200 kbps	Max 4 messages of 8 bytes/day	900-100 kbps (USA)	Up to 156 kbits
Devices per Access Point	20k+	1M	100k uplink, 1M download	364k
Topology	Star	Star on Star	Star	Star supported with an RPMA extender
End Node Roaming Allowed	Yes	Yes	Yes	Yes
Governing Standard Body	3GP	SigFox	LoRa Alliance	Ingenu (formely OnRamp)
Status	13 Released June 2016, In Deployment	In Deployment	Spec released June 2015, In Deployment	In Deployment
Implementation Costs	High	Low	Low	High
Recurring Costs	Medium	Medium	Low	High

Table 6: Comparison of Communications Communicaitons Technologies

The table above compares the major Low Power Wide Area Network technologies. To adhere to the cost constraints presented in the requirement specifications, standards such as Ingenu's RPMA, and LTE-Cat M, which required significant upfront costs, were disregarded. Similarly, closed-source services which relied on provider coverage, such as SigFox were also disregarded for their restrictive outlook towards privately deployed networks. Weightless and nWave are currently have not significant market penetration, and as a result, have limited wireless transceiver options or community created devices. This decision has relegated our final decision to either LoRa and Texas Instrument's SimpleLink.

LoRaWAN – Long-Range Radio Wide-Area-Network

Just 4 years ago, the word LPWAN was not even listed as a word; it is now a buzz-word in a rapidly growing market segment. The Internet of Things has gain tremendous backing and is proposed to continue its exponential increase within the coming years. While LoRaWAN is purported to have significant market share in the LPWAN space, its primary communication methods do present sever restrictions. LORA relies on the chirp spread spectrum modulation technique. Invented in 1933, this modulation technique is rarely used or for communication because of its co-existence limitation. Such limitation restricts data transfer rate of LoRa devices to 18 bits-per-second. At such speeds, a 30-byte message would take 2 minutes to transmit across a LoRa network. This greatly restricts data throughput and resiliency. In contrast, Texas Instrument implements a narrowband communication method for their simple link long range wireless communication. Such modulation technique can drive up to 1.2 kilobits per second and offers significant better co-existent performance. Because of such research, we've decided to use Texas Instrument's Low Power Long Range solution to drive our communication stack. Texas instrument also provides additional tools and supporting documents which will aid in implementing such a communication stack.

III. Parts Selection

This is the most important subsection within Research, where we will be deciding and finalizing the parts to be used in our final implementation. Three areas need to be covered for our application: power, communications and sensors. Power will include everything necessary for realization of the power block diagram presented in the Project Description section: voltage regulator, trickle charger, batteries, power IC for managing power storage, and solar panels.

Communications will include such things as necessary to realize the RF (or communications) portion and microcontroller of our design. This should include the following: antenna, antenna connectors (to PCB), GPS, RF switches, communications-specific chips (such as the TI CC1352R, and others), and flash memory chips for temporary data storage for transmission. It will also include many specialized resistors, capacitors, and inductors that we will need for power output management and filtering of RF components; this will be discussed further in the Implementation section.

The final part of Parts Selection will include sensors for everything we discussed implementing with Steven Eisele. This will include: temperature sensors (thermistors), internal measurement units (IMUs), pressure sensors (force transducers), light sensors (photodetectors), gyroscope, vibration sensors, humidity and moisture sensors, and a few others that will be further discussed within the Implementation section.

I. Power

Trickle Chargers

To utilize the photovoltaic cells to power our project, we will first run the solar panel's output voltage through a rechargeable battery that will then deliver the voltage gathered to the rest of the PCB. Certain parts of this battery system require control such as making the solar panels stop charging the battery when it is fully charged as well as measuring how much charge is left in the battery for the recharge cycle to be triggered to restart. For this former mentioned process, it is important to have MPPT (Maximum Power Point Tracking) to regulate the current and voltage delivered so that the maximum ideal power is achieved.

Texas Instruments already provides ready-made schematics and reference designs for MPPT-based power modules, that we can basically integrated with our already-made schematics for RF communications, GPS module, sensors-array, and user-interfacing. As well, it provides real-time tracking data for charge left in the battery, so this may be the ideal solution. Below, we compare the TI chip that we believe is ideal, with some other comparable chips. Keep in mind that the LT3652ISExPBF-ND by Analog Devices does not come with the same reference designs that the NCP1294EDBR2G by ON Semiconductors and the BQ24650RVAT by Texas Instruments do have available. As well, the TI reference design provides Energy Trace power tracking, which we find useful, as aforementioned.

Chips	**MCP73871** by Mirochip Technology	**BQ24650RVAT** by Texas Instruments	**NCP1294EDBR2G** by ON Semiconductors
Voltage Out (Max)	28 V	28 V	15 V
Voltage Out (Min)	5 V	5 V	3.8 V
Current Out (Max)	10 A	10 A	1 A
MPPT Setting	Yes/Special	Yes	Yes
Frequency-Switching	600 kHz	600 kHz	1 MHz
Fault Protection	Current Limiting, Over Temperature, Over Voltage	Current Limiting, Over Temperature, Over Voltage	Current Limiting, Over Temperature, Over Voltage
Operating Temperature	-40°C ~ 85°C	-40°C ~ 85°C	-40°C ~ 125°C
Control Features	Frequency Control, soft start, sync, preset voltage options	Frequency Control, soft start, sync	Frequency Control, soft start, sync
Cost	$1.84	$4.82	$2.13

Table 7: Comparison of Trickle Chargers

From our available options presented here, we have chosen the MCP73871 by Microchip Technology for a range of reasons. The main reason for our choice was due to the available voltage range per cost. We will however only be needing 1V8 to 6V0 volts, and likely only a few voltages in between this range as required by communications protocols RS485, RS232, UART, and sensors necessary. The MCP73871 delivers the most voltage for how cheap it is. It is also delivering higher power since it has a significantly higher output current compared to the other options.

Switching Programmable Voltage Regulator

Switching Programmable Voltage Regulator ICs (Programmable Switching Regs.) are used to step down the voltage from the rechargeable battery that is originally too high for the PCB. It provides a DC to DC transformation from the power that must be picked up at a moderate level from the solar panels to the much smaller values required to power our board effectively. Texas Instruments as well provides reference designs that are far more extensive in explanation than ON Semiconductors or STMicroelectronics. STMicroelectronics does not provide reference design at all. And these are worth noting before getting into the comparison table we present below. Otherwise, the NCP3064 and MC33063ADR are quite comparable. We will use the table below to help us finalize our decision on which part to choose.

Chips	**TPS65800** by Texas Instruments	**MC33063ADR** by Texas Instruments	**NCP3064** by ON Semiconductors
Voltage In (Max)	40 V	40 V	40 V
Voltage In (Min)	3 V	3 V	3 V
Voltage Out (Max)	5V	38 V	38 V
Voltage Out (Min)	1.25 V	1.25 V	1.25 V
Current Out (Max)	8mA	1.5 A	1.5 A
No. Outputs	3	1	1
Frequency Switching	100 kHz	100 kHz	150 kHz
Operating Temperature	-40°C ~ 85°C	-40°C ~ 85°C	0°C ~ 70°C
Precision Internal Reference	2%	2%	1.5%
Cost	$0.84	$0.57	$1.12

Table 8: Comparison of Switching Programmable Voltage Regulators

Though all options are very similar, the best choice for this project is the TPS6580 because although it carries an equal set of specifications in terms of temperature control and frequency shifting, it offers a lower, and more realistic output to what we want for this project. Also, the higher the frequency shifting, the lower the efficiency which would be a negative tradeoff for this sort of project. It also has a higher range for operating temperature.

Batteries

For the choice of battery, there are many types of rechargeable batteries that can be powered by solar cells. Below we have made a comparison of the different technologies available that we can choose from when coming to a reasonable battery choice. The most important parameters to our project are outdoor temperature range, durability, and rechargeability.

Battery Tech.	Lead Acid	NiCd	NiMH	Li-Ion		
				Cobalt	Mang.	Phosp.
Specific Energy (Wh/kg)	30-50	45-80	60-120	150-250	100-150	90-120
Internal Resistance	Very low	Very low	Low	Nominal	Low	Very low
Charge Time	8-16h	1-2h	2-4h	2-4h	1-2h	1-2h
Overcharge Tolerance	High	Nominal	Low	Low to no trickle charge		
Self-Discharge / Month (room temp.)	5%	20%	30%	<5% Prot. Ckt. consumed 3%/mo.		
Cell Voltage (Nominal)	2V	1.2V	1.2V	3.6V	3.7V	3.2 – 3.3V
Charge Cutoff Voltage (V/cell)	2.40 Fl. 2.25	Full charge detection by voltage signature		4.20 ty. some go higher		3.6V
Discharge Cutoff Voltage (V/cell, 1C)	1.75V	1.00V		2.50-3.50V		2.50V
Peak Load Current (Best Results)	5C 0.2C	20C 1C	5C 0.5C	2C <1C	>30C <10C	>30C <10C
Charge Temperature Range (Fahrenheit)	-4 to 122 F	32 to 113 F		32 to 113 F		
Discharge Temp Range (Fahrenheit)	-4 to 122 F	-20 to 65 F		-4 to 140 F		
Maintenance Required Every	3 – 6 months (replace ment)	Full disch. every 90 days in use		Maintenance Free		
Safety Requirements	Therm. stable	Thermally stable, fuse protection		Protection Ckt. Mandatory		
In-Use Since	l. 1800s	1950	1990s	1991	1996	1999
Toxicity	V. high	V. high	Low	Low		
Coulombic Efficiency	90%	70% with trickle, 90% with fast char.		99%		
Cost-per-Watt	Low	Moderate		High		
Cycle Life (80% DoD)	200-300	1000	300-500	500-1000	500-1000	1000-2000

Table 9: Comparison of Battery Technologies for Solar Applications

The figures in the table above comparing characteristics of commonly used rechargeable batteries are based upon the average ratings of commercial batteries that we have available to us for parts, at the time of publication. Specialty batteries with above-average ratings are excluded. These are more along the lines of industry averages. [14]

LiPO4 seems to be the best options in terms of temperature range, durability, and rechargeability. However, considering the product dimensions and durability, we may also choose to use a small, sealed, lead acid battery. These are quite common, inexpensive compared to LiPO4 and fit our standards somewhat appropriately. The specifications could be of a packaged lead acid battery is about 6 Volts at between 4000 and 8000 mAh (milli-Ampere Hours, mA*h). Below we compare the actual battery that we have to choose from to come to a final decision on battery technologies we intend to employ. Again, note that the most important parameters to our project are outdoor temperature range, durability, and rechargeability.

Batteries	PRT-1351 Li-Ion by SparkFun Electronics	ASR00007 by TinyCircuits	2750 Li-Ion by Adafruit Industries LLC
Battery Chemistry	Lithium-Ion	Lithium-Ion Polymer	Lithium-Ion
Voltage-Rated (V)	3.7	3.7	3.7
Capacity (mAh)	400	290	350
Termination Style	JST PH Connector	JST SH Connector	JST PH Connector
Package Size (mm)	26.5 x 36.9 x 5.0	23.9 x 23.9 x 5.8	36 x 20 x 5.6
Weight (g)	9	29	7.9
Standard Charge Current (mA)	100	N/A	N/A
Cost ($ USD)	4.95	5.95	6.95

Table 10: Comparison of Solar Battery Components

Considering that all three of these batteries use the same type of technology, have the same output voltage and connector type is relatively the same, our deciding factors will be package size, weight and most importantly cost. The dimensions of all three batteries are nominal and not necessarily the most principal factor, either. The SparkFun Electronics PRT-1352 Li-Ion battery for solar applications has the lowest weight (ideal) and lowest cost, undercutting both other competitors. Standard (Standby) charge current for this battery is high at 100 mA, but that can be generated using the trickle charger if we set up the battery already charged, and the trickle charger pulls small current from it anyways. So, this is the battery we will be utilizing for the intents and purposes set forth for our end-product.

Solar Panels

There are many options for solar panels. Most of these options are differentiated with respect to voltage- and power-supplied from them at optimal lighting conditions. We do not have a lot to spend on solar panels to remain within our budgeted targeted cost-per-unit, so the most important determining factors for us are wattage (power delivery) and cost-per-watt. We will likely only need a 1.5 o 3W panel, so we want to be firm in spending under about 7 USD per panel (10% of unit cost).

We are only comparing two options for solar cells here because there are so many options available which are essentially the same. The prices all fall around that of the Semi-flexible solar cell (10 Watts) by Crystal Solar, which we will likely be using. It is also worth noting that these were the lowest cost-per-watt solar panels with our specified parameters range available on Digi key.

Panel	**Semi flexible solar panel 10W** by Crystal Solar	**Medium 6V, 2W solar panel** by Adafruit Industries LLC
Max Power	10 W	2 W
Max Voltage	10 V	6 V
Connector	MC4 Compatible Connector	3.5 mm OD/1.1 mm ID DC jack
Cost/Watt	$0.98	$15

Table 11: Comparison of Solar Panels Available

For the voltaic cells we have chosen are ones that have a peak power of 10 watts with 10 volts at around 1,000 amperes. The reason for these specific parameters is because an input of 10V at 1,000A, though small, fills the battery after a day of full charging so it can properly discharge at night. It also has a decent amount of output voltage and current in terms of what our control chip can handle. We have also chosen this model because it is far cheaper than the rest that are around these specifications.

II. Communications

Antennas

For RF applications, the importance of the antenna is the interface between radio waves propagating through space and electrical currents moving in metal conductors, used with a transmitter or receiver. The antenna radiates current as electromagnetic radiation (EM waves or radio waves). It also has the power to produce electrical current at its terminals from incoming reception. There are several types of antennas we could use for our specific application, including dipole antennas, directional antennas, patch and microstrip antennas (onboard), sector antennas, yagi antennas, parabolic and dish antennas, and grid antennas. The table below compares a few lower-cost choices we have for antennas that fit our specific application.

Antennas	**Duck Antenna RP-SMA** by SparkFun	**YA90011 Yagi Antenna** by DMS Wireless	**AU09G6HQ-NF Omni Antenna** by Altelix
Operating Freq. (MHz)	900 or 1800	900	824 to 960
Output Power (dBi)	2	11	6
Range	Medium-to-Long	Very long	Long

Antennas	Duck Antenna RP-SMA by SparkFun	YA90011 Yagi Antenna by DMS Wireless	AU09G6HQ-NF Omni Antenna by Altelix
Nominal Impedance (Ohm)	50	50	50
Type of Antenna	Directional (Duck) Antenna	Directional (Yagi) Antenna	Omni Directional Antenna
Standards	SigFox, WiFi	Yagi, SigFox, Wi-Fi	GSM 850/900 Cellular, WCDMA, SCADA, LPWAN, LoRa, LoRaWAN, RFID, Multipoint and NLOS
Connector Type	RP-SMA Female	Wire-In	RP-SMA Female
ROHS Compliance	Yes	Unknown	Yes
FCC Compliance	Yes	Unknown	Yes
Price	$7.95 + shipping	$2.99	$24.99

Table 12: Comparison of 2.4GHz Antennas

Duck, Yagi and Omni-Directional are the most applicable to our project. The YA90011 antenna is very powerful and very long range but has a very narrow usage band and dampens the signal outside of that, so it is likely not feasible for 2.4 GHz. The AU09G6HQ-NF Omni-Directional Antenna would be perfect for our gateway. It is also insulated and has full compliance with all standards we are using – most importantly GSM for cellular communications channels. As well, the output power gain is good and will deliver long range, as needed.

For all other implementations excluding the gateway, we may want to use the cheaper option of the RP-SMA Directional Duck Antenna. It will function reasonably well at 2.4 GHz which we need for Bluetooth 4.0 and ZigBee implementation. As well, it will function for wireless transmission of data to the gateway as needed. Power delivery and range are not as good as the AU09G6HQ-NF, and directionality is not ideal either; however, they do not need to be. All we need these smaller antennas for is delivery to the gateway, like a middleman between the sensors and long-range transmission.

We may also look at PCB-specific RF antennas, and JSC-type antennas, or create our own path or microstrip antennas for short-range ZigBee applications. These will come in Implementation, as needed.

Antenna Connectors

Antenna connectors are likewise needed to connect the PCB to the antenna. There are a few standards for sizing of antenna connectors. The two types of antennas we purchased both used Reverse-Polarized SMA (RP-SMA) female connector types, as per FCC requirement for wireless PCBs in homes. However, we will not be in homes, so this requirement does not restrict us. And, as we may in future decide to go with a JSC-type antenna we have also included a JSC-type coaxial PCB connector by Murata. These are much smaller and less intrusive, but also lower range; as such, they may be used for ZigBee protocol short-range chips that we use within the "yards" and not necessarily for the Gateway.

Antenna Connectors	**Reverse Polarized SMA Connector** by Sparkfun	**RP-SMA-J / RP-SMA-K Connector** by Areyourshop	**MM5829-2700RJ4 Coaxial Series Cable Connector** by Murata
Supported Antenna	Reverse Polarized (RP) – SMA standard-following antennas	RP-SMA-J or RP-SMA-K standard following antennas	Coaxial JSC Type Connector
Connection Type	Female	Female	Female Plug
FCC Compliance	Yes	Unknown	Yes
Angle	90°	45° or 135°	90°
Price	$1.95 + Shipping	$5.35 + Shipping	$0.63

Table 13: Comparison of 2.4GHz Antennas Connectors

We have already ordered the RP-SMA connector by SparkFun, because it is matched by manufacturer and connection type to our RP-SMA Duck Antenna that we also purchased from SparkFun. We still intend to use these for non-gateway applications. However, we have also ordered the MM5829-2700RJ4 Coaxial Series Cable Connectors from Murata, because they are used directly in the reference design for the Texas Instruments CC1352R transceiver chip and MCU. We may still choose to go with an JSC type connector, once we begin PCB-level implementation, if the duck antenna does not work as well as expected for ZigBee and Bluetooth transmission.

RF Switches

Because we intend to use multiple channels, an RF switch is needed; a MUX cannot be used because of the high impedance, and digital signals will not be used. RF switches route high frequency signals through transmission paths by cleverly utilizing the signal frequency as the controller for output, in the same way the MUX uses binary logic for control. Because we will be using two frequencies for inputs and outputs, we will need a transfer switch for double pole, double throw (DPDT) which allows inputs from one channel and outputs to two channels, and likewise the reverse is true for signal intake.

Below a comparison is made as to which type of RF Switch IC is best for us to use. The XMSSJE30GOPA by Murata is no longer available, so an alternative or substitution must be made in order for us to realize the CC1352 RF communications reference design. The most key features for us are that the IC is bi-directional, so we can both transmit and receive data through the same RF antenna, that it is 2-to-1 (2 ports to 1 common communications channel) so we can interface the singular communications channel that will be routed to the antenna from the two ports (which each have a positive and negative input/output will pass through balun for single-line integration) on the CC1352R1. Cost is also very important for us to stay within budget targets.

Chips	PE4250 by Peregrin	XMSSJE 3GOPA by Murata	BGS13S4N9 by Infineon
Operating Freq.	0.1GHz-2.7GHz	0.1GHz-2.7GHz	0.1GHz to 3GHz
RF Impedance (Ohm)	±50	±50	±50
No. Inputs	1	1	1
No. Outputs	2	2	3
Bidirectional (Receiver/ Transceiver)	Yes	Yes	No
Temp. Op. Range	-40˚C ~ +85˚C	-40˚C ~ +85˚C	-40˚C ~ +80˚C
Price	$0.94/250 chips	$0.63/250 chips	$0.17/250 chips
Availability	Available	Out of Service	Available

Table 14: Comparison of 2-Channel Bidirectional RF Switches

The reference schematic design provided by Texas Instruments utilizes a Murata XMSSJJ3GUP-054 to effectively deal with RF switching for the CC1352 dual-band chips. The Murata XMSSJJ3GUP-054 routes the 900 MHz, and 2.4 GHz signals from the antenna to the Wireless Microcontroller Unit (WMCU). The chip and accompanying documentation could not be found.

A suitable replacement was found, which worked at the same frequencies and provided similar impedances. Peregrin's PE4250 operates at an equal operating frequency, requires similar input voltage and also incorporates an internal DC blocking capacitor. This removes the need for the DC blocking capacitor, C36, from the 900 MHz band, reducing overall unit cost and additional parts required. In addition, Peregrine's PE4250 is also well affordable, and is also well documented with included reference designs.

Communications Chips (Wireless MCU Transceiver)

The MCU and Wireless transceiver integrated Communications Chip is arguably the most important chip for our Internet-of-Things Sub-1GHz Communications Template project. There are few options for long-range sub-1GHz communications transceiver chips that support IP.

The most important features for our Wireless MCU Transceiver chipset are power usage (some are very high-power which we cannot afford to accommodate), cost, availability (some are no longer available or not-yet available), and implementation documentation, because none of us have extensive experience in RF design and will need extensive reference design documentation to understand and implement our own RF transceiver circuits.

LoRaWAN would be a nice addition, because it is less costly than SigFox (there is a small monthly fee associated with SigFox), but the range is significantly lower, power cost is higher, and setup cost is higher. As well, the most significant constraint for our project is the long-range and rate of transmission, in regard to which LoRaWAN would severely limit our project.

Of these chips, our best available choice is the Texas Instruments CC1352R. We have ordered and are currently developing for this chip. It was chosen for a multitude of reasons. The standby current is significantly lower than the other two chips, which will lower power required and overall maintenance costs in the long run. The range is reportedly as long as102 km, at 14 dBm, according to Texas Instruments. It supports many of the standards the we intend to use for the meshing and pushing protocols as well such as SigFox and Zigbee which requires Bluetooth. And it is capable of this feat because it can utilize the Sub-1GHz and 2.4 GHz bands simultaneously.

We will likely switch to the CC1352P which has a maximum range three-times the range of the regular CC1352R. As well, both chips come standard with a stepdown DC/DC converted (internal voltage regulator) for the MCU, which is programmable, exactly as we desired, so in testing we will decide whether or not we actually need a stepdown converted or voltage regulator still for further production. This will be done within the Implementation section.

Decibel-milliwatts is a measure used to indicate that a power ratio is expressed in decibels (dB) with reference to one milliwatt (mW), in other words dB * mW (dBm). It is used to express the power capacity of radio, microwave and fiber-optical communication networks and in our case, 50-Ohm impedance scale RF transceivers. Because the range of any RF device is most heavily dependent upon power output to the antenna, the dBm is the most important measure in determining range and is exponential as it increases. Another reason to switch to the CC1352P once available is that it outputs far more, using the same long-distance configuration as the CC1352R, and thus should more than double the effective communication range over the Texas Instrument CC1352 Sub-1GHz Transceiver.

Chips		CC1352R by TI	CC1352P by TI	SX1276 by Semtech	S2-LP by STM
Description		Support for Sub-1GHz + 2.4 GHz concurrently with the lowest power multi-band wireless MCU	Support for Sub-1GHz + 2.4 GHz concurrently with an integrated 20dBm PA enabling much longer range	RF transceiver/modem that provides ultra-long range spectrum	ultra-low power low data-rate transceiver, suitable for ISM bands and Wireless M-Bus
Cost		$6.85	Unknown	$6.17	$3.63
Availability		Available	Available in August	Available	Available
Implementation Documentation		Excellent	Excellent	Satisfactory	Good
Key Features	SimpleLink MCU Platform Compliant	Yes	Yes	No	No
	Application MCU	ARM Cortex M4F	ARM Cortex M4F	None	STM32 Nucleo
	Memory	352kB Flash, 256kB ROM, 80kB RAM	352kB Flash, 256kB ROM, 80kB RAM	None	128kB Flash, 64kB ROM, 16kB RAM
	Standby Current Consumption	0.8µA	0.8µA	9.9 mA	7 mA
	Output Power	5 dBm	20 dBm	14 dBm	16 dBm
	Communication Bands Supported	Sub-1GHz, 2.4GHz concurrently	Sub-1GHz, 2.4GHz concurrently	137 MHz to 1020 MHz	413-497MHz, 452-527MHz, 826-958MHz, 904-1055MHz
	Communications Technologies Supported	Bluetooth + Zigbee, Sub-1GHz, Wi-Fi, Multi-Standard	Bluetooth + Zigbee, Sub-1GHz, Wi-Fi, Multi-Standard	LoRaWAN, Sub-1GHz	Sub-1GHz
	Web Servers Supported	Amazon AWS, IBM Watson	Amazon AWS, IBM Watson	Unknown	Unknown
	Communications Range	long range	extremely long range	short range	long range

Table 15: Comparison of Wireless Microcontroller Units (WMCUs)

GPS Chips

Adding a GPS Chipset to our end-product will likely be difficult and expensive. But because of the necessity to Steven Eisele we are determined to find a cost-effective way to add active GPS-location from anywhere, anytime to our communications template end-product.

GPS RF Receiver	**SE4150L** by Skyworks Solutions	**SE4110L** by Skyworks Solutions	**CSRG0530B 01-ICKD-R** by Qualcomm	**STA8058-ND** by STMicroelectronics
Datasheet Available	Yes	Yes	No	No
Reference Designs Available	Yes	Yes	No	No
Availability	Available	Available	Available	Available
Minimum Order Quantity	1	1	1	1000
Price (USD)	$3.20	$3.59	$6.88	$6.65
Lead Time	Immediate	Immediate	20 Weeks	26 Weeks
Maximum Sensitivity (dBm)	Unknown	Unknown	Unknown	-159
Current – Receiving (mA)	15	10.5	Unknown	Unknown
Voltage – Supply (V)	2.7 – 3.6	2.7 – 3.6	Unknown	3 – 3.6
Operating Temp. Range (Celsius)	-40 to 85	-40 to 85	Unknown	-40 to 85
Data Interface	PCB, Surface-Mount	PCB, Surface-Mount	Unknown	PCB, Surface-Mount
Memory Size	none	none	Unknown	256kB Flash NOR Mem IC, 64kB RAM
Package / Case	24-VFQFN Exposed Pads	24-VFQFN Exposed Pads	Unknown	104-LFBGA
Device Packaging	24-QFN (4x4)	24-QFN (4x4)	Unknown	104-LFBGA (11x7)

Table 16: Comparison of GPS Receiver Frontends

With any GPS module we choose, within our price range, we will need to create a 15- to 25-mm size range patch antenna onboard PCB on the ground plane. We choose patch antennas because they provide better performance per size and lower cost and integration required compared to Linear GPS antennas (chip or dipole). As well, Linear GPS Antennas only receive signals from one axis resulting in less sensitivity and reduced coverage. We want greater coverage. Patch antennas sold by Taoglas Limited are our best bet for tested GPS module antennas and allow a simple GPS interface. More on this will be discussed in specific implementation details.

We have chosen to order both the SE4150L and SE4110L because they come with reference designs and ample information within the datasheets as to how to use these GPS receivers with other RF devices. This will be immeasurably helpful when integrating or interfacing the Texas Instruments CC1352R and CC1352P internal MCU with our custom GPS module. They do however consume a lot of current and will require their own power. As well, it is worth noting that we will have to add a lot of other parts to the PCB specifically for our custom GPS module featuring these receivers such as antenna, ASM sensors, and other filtering requirements for each input. This will add associated cost, as well as take up PCB space, so we have chosen the least expensive and smallest GPS receivers for that explicit purpose as well – the entire GPS module on our PCB will be expensive.

One possibility to decrease costs as well, is instead of using a baseband receiver IC to process the digital output of the GPS receiver, pushing all digital data from the GPS receiver immediately to the server. The digital data would then be processed using a software GPS baseband processor ("software GPS receiver") library for a programming language we are familiar with. One such library exists in the same spirit as Software-Defined Radio (SDR): GNNS Software-Defined Receiver.

In the meantime, we have also purchase the completed GPS module 474-DD-14239 Spark Fun GPS Receiver Module SMD-S1315F to be integrated with the communications template in the case that we cannot develop the GNNS Software-Defined Receiver in time.

Flash Memory Chips

Because the memory on the Texas Instruments CC1352R and CC1352P that we are using is used for transmission purposes only we will need external Flash Memory ICs. We have two options here, NAND Flash and NOR Flash Memory ICs. The most crucial factors for our project, for flash memory specifically, include access speed, standby power, capacity, and cost-per-bit. The table below compares these two options.

Flash Mem. Type	NAND	NOR
Standby Power	High	Low
Cost-Per-Bit	Low	High
File Storage Use	Easy	Hard
Code Execution	Hard	Easy
Capacity	Hard	Easy
Write Speed	Fast	Slow
Read Speed	Slow	Fast
Active Power *	Low	High

**note that active power id dependent upon how the flash memory is used. NOR is typically slower in writes and consumes more power than NAND for this action. NOR is typically fast on reads, which consume less power.*

Table 17: Comparison of Flash Memory Types

NOR memory seems to be the better option at this point, as the read speed is high, standby power is low, and capacity is high versus cost-per-bit. But because no figures are given, we will need further comparison to determine which technology is a suitable to interface with the CC1352R1 RF WMCU.

The table below further compares NAND and NOR Flash Operating specifications, similar to standards for particular flash memory cells that are available, including SLC NAND, MLC NAND, and MLC NOR. We are using this to further compare realistic figures on memory types to help us logically select an available technology replacement for the unavailable MX25R8035FZUIL0 by Macronix. It is worth noting before continuing to the following chart that the MX25R8035 is an NOR Flash Memory IC with 8Mb (1M x 8) SPI at 33 MHz utilizing 8-USON (2 x 3).

NAND and NOR Flash Operating Specifications

	SLC NAND Flash (x8)	MLC NAND Flash (x8)	MLC NOR Flash (x16)
Density	512 Mbits – 4 Gbits	1 Gbit to 16 Gbits	16 Mbit to 1 Gbit
Read Speed	24 MB/s	18.6 MB/s	103 MB/s
Write Speed	8.0 MB/s	2.4 MB/s	0.47 MB/s
Erase Time	2.0 mSec	2.0 mSec	900 mSec
Interface	I/O – indirect access	I/O – indirect access	Random Access
Application	Program/Data mass storage	Program/Data mass storage	eXecuteInPlace applications and Data mass storage

Table 18: Comparison of NAND and NOR Flash Operating Specifications

Both NAND and NOR flash memories have a place in embedded MCUs, however we will want to be using these MCUs for data storage, and not necessarily a lot of data so we do not necessarily care how much space we have. We will likely want to be using NOR in this case. The table below compares some options we have for NOR-specific Flash memory for data mass storage and program storage.

Flash Memory IC	**MX25R8035F M1L0** by Macronix	**S25FL116K0XMF I041** by Cypress Semiconductor	**SST25WF080B-40I/SN** by Microchip Technology
Memory Type	NOR Flash Mem.	NOR Flash Mem.	NOR Flash Mem.

Flash Memory IC	**MX25R8035F M1L0** by Macronix	**S25FL116K0XMF I041** by Cypress Semiconductor	**SST25WF080B-40I/SN** by Microchip Technology
Reference Designs Available	Yes	No	No
Cost	$0.77	$0.59	$0.94
Available	Available	Available, but End-of-life (EoL)	Available
Key Features	Ultra-low power mode, high-performance mode, wide range VCC (1.65 – 3.6V) for RWP operations, unique ID and secure OTP support	Variable read latency (number of dummy cycles) for faster initial access time or higher clock rate commands, industrial plus and extended temperature range	Ultra-low power mode, pin-count package which occupies less board space and lowers system costs, high performance fast CMOS SuperFlash technology
Key Features	multi I/O support (single, dual and quad), program interrupts	volatile config option in addition to legacy non-volatile config.	better long-term reliability and performance
Memory Size	8Mb	16Mb	16Mb
Max CLK Freq.	108MHz	108MHz	40MHz
Interface Type	SPI	SPI	SPI
Organization	4K x 16	2M x 8	1M x 8
Data Bus Width	8 bit	8 bit	8 bit
Supply V. Min (V)	1.65	2.7	1.65 V
Supply V. Max (V)	3.6	3.6	1.95 V
Supply I. Max (mA)	6	25	5
Min Op. Temp.	-40	-40	-40
Max Op. Temp.	85	85	85
Packaging	8-SOIC	Tube	8-SOIC
Pins	8	8	8
Connection Type	Surface Mount	Surface Mount	Surface Mount

Table 19: Comparison of NOR Flash Memory Chips Available

As it turns out, all of the NAND memory chips offered more memory, but were significantly more expensive and outside of our budget. So, we have only compared NOR Flash Memory ICs.

We have chosen the MX25R8035F M1L0 by Macronix because it has references designs already available for RF integration, and more specifically for the Texas Instruments CC1352R. It is also referenced specifically as the flash memory chosen by TI for the interface of the CC1352R, so it is the best possible choice in regard to this. Operating temperatures are within a suitable range, and the surface mount is ideal for soldering ourselves, as will be required of us. It is not the least expensive per memory, but it does have security and built-in features that will specifically aid our application, so while cost may remain in the middle-ground, it is well worth it to have the MX25R8035F as our NOR Flash Memory IC.

III. Sensors

Temperature Sensors (Thermistors)

The mount of this chip needs to be a surface mount, and the temperature sensor chip needs to be able to endure and sense temperatures from a minimum range of -40 ˚C to 85 ˚C. after taking all these specs into account, Plan Bee narrowed down possible chips. Some chips viewed include I^2C, but after reviewing multiple chips, a preferable output type not only includes I^2C but SMBus as well.

The TMP112AIDRLT chip has one of the best accuracies of ±0.5˚C, but is also more expensive, costing $1.37 per 250 chips. This chip also has a higher resolution of 12b vs a resolution of 11b or 10b. The other two chips with 12b resolution are the TMP112NAIDRLT and the TMP112NAIDRLR, which are both the same TMP112 chips, but slightly different models. The difference between these three TMP112 chips is accuracy and price. The TMP112NAIDRLT has an accuracy of ±1˚C, and is also one of the more expensive chips, costing $1.22 per 250 chips. These previously specified chips have a lower range of operation temperature. None of these chips can be bought individually, so free samples would need to be requested. The MCP9808T-E/MC chip operates at a slightly higher voltage and has a lower resolution than the other chips, but does have a higher accuracy of ± 0.5˚C. The MCP9808T-E/MC does have a preferred packaging/casing of 8-TSSOP, 8-MSOP (0.118", 3.00mm Width), which is a variation to be seriously considered. These four sensors are showcased in the table below. Plan Bee ultimately decided to order the TMP112AIDRLT and the TMP112NAIDRLR because they are the most accurate and have the highest resolution. These two chips are also the same TMP112 sensor, but different models. Ordering both will be useful for comparisons later and will help Plan Bee decide which of the two sensors to use for the final prototype.

Temperature Sensors	Manufacturer	Resolution	Accuracy	Temp. Operating Range	Price
TMP102AIDRLT	Texas Instruments	11b	± 2˚ C	-55˚C ~ +150˚C	$0.94/ 250 chips

Temperature Sensors	Manufacturer	Resolution	Accuracy	Temp. Operating Range	Price
TMP112AIDRLT	Texas Instruments	12b	± 0.5˚ C	-40˚C ~ +125˚C	$1.37/ 250 chips
TMP112NAIDRLT	Texas Instruments	12b	± 1˚ C	-40˚C ~ +125˚C	$1.22/ 250 chips
TMP112NAIDRLR	Texas Instruments	12b	± 0.5˚ C	-40˚C ~ +125˚C	$1.03/ 250 chips
MCP9808T-E/MS	Microchip Technology	10b	± 0.5˚ C	-40˚C ~ +125˚C	$.97/2500 chips

Table 20: Comparison of Temperature Sensor ICs

Internal Measurement Units (IMUs)

We expect to implement an Internal Measurement Unit (IMU) chip. The MPU-9150 that was included on the test board is now outdated and no longer offered, so therefore cannot be considered in this comparison. The MPU-9250, however, is active and the newest version of this sensor is available from InvenSense. When looking at the specs for IMU's, the question arose of whether I^2C was the only output type Plan Bee wanted to have on this chip. Plan Bee examined chips with only I^2C, as well as chips that offered a wider range of output types, such as SPI and UART. Some chips that only offered I^2C, such as the BHI160, are just as expensive, if not more expensive, than chips with a variety of output types, such as the BNO080. Including more output types in this chip would hinder the budget of the project but it could be helpful to have varying output types. The MPU-9250 only offers I^2C but is more advantageous in price. Next, Sensor type needed to be considered. All chips that contain I^2C, SPI, and UART have a wider range of sensor types than the chips with I^2C, which is another benefit of choosing one of the more flexible chips. Most chips with the ideal output type have sensor types of Accelerometer, Gyroscope, Magnetometer, and 9 Axis. The BNO080 and the BNO085 are both much cheaper if ordered in bulk (around $5), but that price rises to around $12 if bought individually. Free samples would preferably be requested. Both the BNO080 and the BNO085 chips are manufactured by Hillcrest Laboratories. The MPU-9250 was deemed to be the most useful due to the low cost. Low cost was ultimately deemed preferable to multiple output types.

Internal Measurement Unit	Manufacturer	Output Type	Sensor Type	Temperature Operation Range	Price
BNO080	Hillcrest Laboratories, Inc.	I^2C, SPI, UART	Accelerometer, Gyroscope, Magnetometer, 9 Axis	-40°C ~ 85°C	$13.48/chip
BNO085	Hillcrest Laboratories, Inc.	I^2C, SPI, UART	Accelerometer, Gyroscope, Magnetometer, 9 Axis	-40°C ~ 85°C	$11.85/chip
BHI160	Hillcrest Laboratories, Inc.	I^2C	Accelerometer, Gyroscope, 3 Axis	-40°C ~ 85°C	$12.40/chip
MPU9250	InvenSense	I^2C	Accelerometer, Gyroscope, 3 Axis	-40°C ~ 85°C	$9.32/chip

Table 21: Comparison of Internal Measurement Unit (IMU) ICs

Pressure Sensors (Force Transducers)

Pressure sensors will also be implemented in Plan Bee's design. The test board included the BMP-180 chip sensor, but this chip could not be compared because it is obsolete. A Surface Mount pressure sensor that operates under 4 Volts was prioritized due to the power source providing around 4V or less to the system. I^2C is a needed output type, but if other output types are included with certain chips, those are preferable. It was discovered that most chips with only I^2C did not include all the other specifications needed to progress in the selection process. Chips that fit the other ideal requirements seemed to have I^2C as well as SPI included in their output types. After narrowing down output type, almost all remaining chips have the same output of 24b, except for the SSCMNN015PA5A3, which has an output type of 12b. Maximum pressure did not need to be extremely high, so the 290.08 PSI is acceptable for most of the sensors. The SSCMNNN015PA5A3 has a maximum pressure of 30 PSI (206.84 kPa). Although a high maximum pressure is not needed, the highest degree of accuracy is crucial for Plan Bee's project; ±.015 PSI is as minimum of an accuracy range as possible that all chips compared need to have. The SSCMNN015PA5A3 does not meet this accuracy requirement and is also twice as expensive as all other chips compared.

The chip selected also needs to have a minimum temperature operation range of -40°C to 85°C. After narrowing chips down using these requirements, the remaining chips all have a few differences, which include operating pressure range, packaging/casing, the supplier device package, and price. These differences are previewed in the table below. Most chips below are produced by STMicroelectronics. Plan Bee decided to go with the most inexpensive chip, the LPS22HBTR.

Pressure Sensor	Manufacturer	Operating Pressure	Packaging/ Casing	Supplier Device Package	Price
LPS22HBTR	STMicroelectronics	3.77 PSI ~18.27 PSI (26 kPa ~ 126 kPa)	10-WFLGA Exposed Pad	10-HLGA (2.0x2.0)	$3.28/chip
LPS22HDTR	STMicroelectronics	_	10-VFLGA	10-HLGA (2.0x2.0)	$3.96/chip
LPS33HWTR	STMicroelectronics	3.77 PSI ~18.27 PSI (26 kPa ~ 126 kPa)	LGA	10-CCLGA (3.3x3.3)	$8.83/chip
SSCMNNN015PA5A3	Honeywell Sensing and Productivity Solutions	15 PSI (103.42 kPa)	8-SMD, J-Lead	8-SMT	$23.13/250 chips

Table 22: Comparison of Pressure Sensor ICs

Light Sensors (Photodetectors)

Plan Bee also expects to add a light sensor to the prototype board. For this project, the light sensor needs to be an ambient light sensor operating with an output type of I^2C (a variety of types are not available with this chip). The sensor chosen also needs to operate on less than 4 Volts and must be a surface mount type chip as well. Plan Bee has previously decided that the minimum temperature should never be above -40 ˚C, nor should the maximum temperature ever drop lower than 85 ˚C. Only specific sensors that have these minimum requirements are included in comparison. Intersil manufacturing produces many similar chips with these specs. The 5 Intersil chips are all extremely similar, mainly varying in price. The APDS-9250 sensor, produced by Broadcom Limited, would be cheapest if a free sample could be obtained. The OPT3001DNPR is also cheaper when bought in bulk but may not be available for Plan Bee. Both bulk order sensors have similar specs to the other chips, as compared in the table below. If free samples cannot be obtained, the cheapest chip left will be what Plan Bee decides to use, which is the ISL29023IROZ-T7. If free samples can be obtained, The OPT3001DNPR would be chosen due to the more ideal voltage supply range.

Light Sensors	Manufacturer	Voltage Supply Range	Temperature Operation Range	Price
ISL29023IROZ-T7	Intersil	2.25 V ~ 3.63 V	-40˚C ~ +85˚C	$3.43/chip

Light Sensors	Manufacturer	Voltage Supply Range	Temperature Operation Range	Price
ISL29020IROZ-T7	Intersil	2.25 V ~ 3.63 V	-40˚C ~ +85˚C	$4.26/chip
ISL29033IROZ-T7	Intersil	2.25 V ~ 3.63 V	-40˚C ~ +85˚C	$4.08/chip
ISL76683AROZ-T7	Intersil	2.5 V ~ 3.3 V	-40°C ~ +105°C	$4.23/chip
ISL76683AROZ-T7A	Intersil	2.5 V ~ 3.3 V	-40˚C ~ +105˚C	$4.64/chip
APDS-9250	Broadcom Limited	1.7 V ~ 3.6 V	-40°C ~ 85°C	$.91/2500 chips
OPT3001DNPR	Texas Instruments	1.6 V ~ 3.6 V	-40°C ~ 85°C	$1.27/3000 chips

Table 23: Comparison of Light Sensor ICs

Humidity and Moisture Sensors

Humidity and Moisture sensors are applicable for Plan Bee's Design as well. For this sensor to be useful for this project, it needs to be a surface mount, relatively accurate, support an output type of I^2C, have a full humidity range of 0% to 100%, and operate on under 4 Volts. After considering the specs above, there are no chips that Plan Bee could find that fit the requirements while also including RPSMA and SPI, so I2C alone was chosen.

Looking at response time, a fast as possible response time is preferred. Some chips have lower response times, such as 18s and 10s, far slower than the 8s and 5s of many other chips. Some of these types of sensors are also unavailable for immediate delivery; Plan Bee would like the chips delivered as quickly as possible. Because of response time disparities as well as delivery time, only 4 chips are compatible with Plan Bee's project. The factors most important to us for the selection of parts given below, that we used to filter these parts included cost, voltage-supply range, communication type (all of them use I2C as standard) and data output since we have limited bandwidth. These are further analyzed in the table and factor analysis below.

Humidity and Moisture Sensors	Manufacturer	Voltage Supply Range	Response Time	Output	Price
HPP845E031R4	TE Connectivity Measurement Specialties	1.5 V ~ 3.6 V	5s	12b	$2.88/400 chips
SHT21	Sensirion AG	2.1 V ~ 3.6 V	8s	12b	$4.21/400 chips
SHT25	Sensirion AG	2.1 V ~ 3.6 V	8s	12b	$6.97/400 chips

Humidity and Moisture Sensors	Manufacturer	Voltage Supply Range	Response Time	Output	Price
HDC2010YPAT	Texas Instruments	1.62 V ~ 3.6 V	8s	11b	$1.71/250 chips

Table 24: Comparison of Humidity and Moisture Sensor ICs

The HPP845E031R4 has the fastest response time out of the four chips, as well as one of the cheapest prices. The HDC2010YPAT is also cheaper especially because less parts are required per order; however, this chip has a slower response time. The output of the HDC2010YPAT is 11b whereas all 3 others have an output of 12b. The different packaging/casing is preferred to be a 6-TDFN Exposed Pad, which is what the first three sensors come in. The HDC2010YPAT was ultimately chosen for this project because of the wider voltage supply range and lowest available.

IV. Project Components

After comparing many parts and sensors, it came time to buy the parts and have them shipped to Plan Bee for testing. It should be known that smaller components such as ceramic capacitors, capacitors, and resistors are not shown in the segment below but have been ordered as well. Each part in the section below is discussed in brief, stating facts such as why the part was chosen and from where it was ordered. Most parts were found using the Digikey and Moser suppliers. These suppliers can compare many parts with specified qualities in a speedy and informative manner. Using these tools helped Plan Bee find parts that had been previously unknown to the group, as well as helping narrow down which qualities are most ideal for each part. Some unknown qualities were revealed in parts that once seemed to be a first choice, but now have more apparent flaws than expected. Without this supplier comparison service, narrowing down the most efficient parts would have been a far more difficult process.

The TMP112, an ambient temperature sensor, was chosen because it was the most affordable part with the desired qualities of high resolution and high accuracy. Two different models of the TMP112 were ordered just in case there was a preferred packaging type decided upon arrival, etc. The TMP112AIDRLT and the TMP112NAIDRLR are shown below. Both parts were ordered from Texas Instruments, after using the Digikey site to narrow down comparisons.

The MPU9250 was chosen over other prospective Internal Measurement Unit parts because of how much more affordable it was compared to other competing sensors. The MPU9250 below was ordered from the Digikey site.

The LPS22HB pressure sensor chip was chosen over other prospects because of how much cheaper it was, as well as the more ideal packaging and casing that it came in. This sensor was ordered from Digikey as well and is exhibited below.

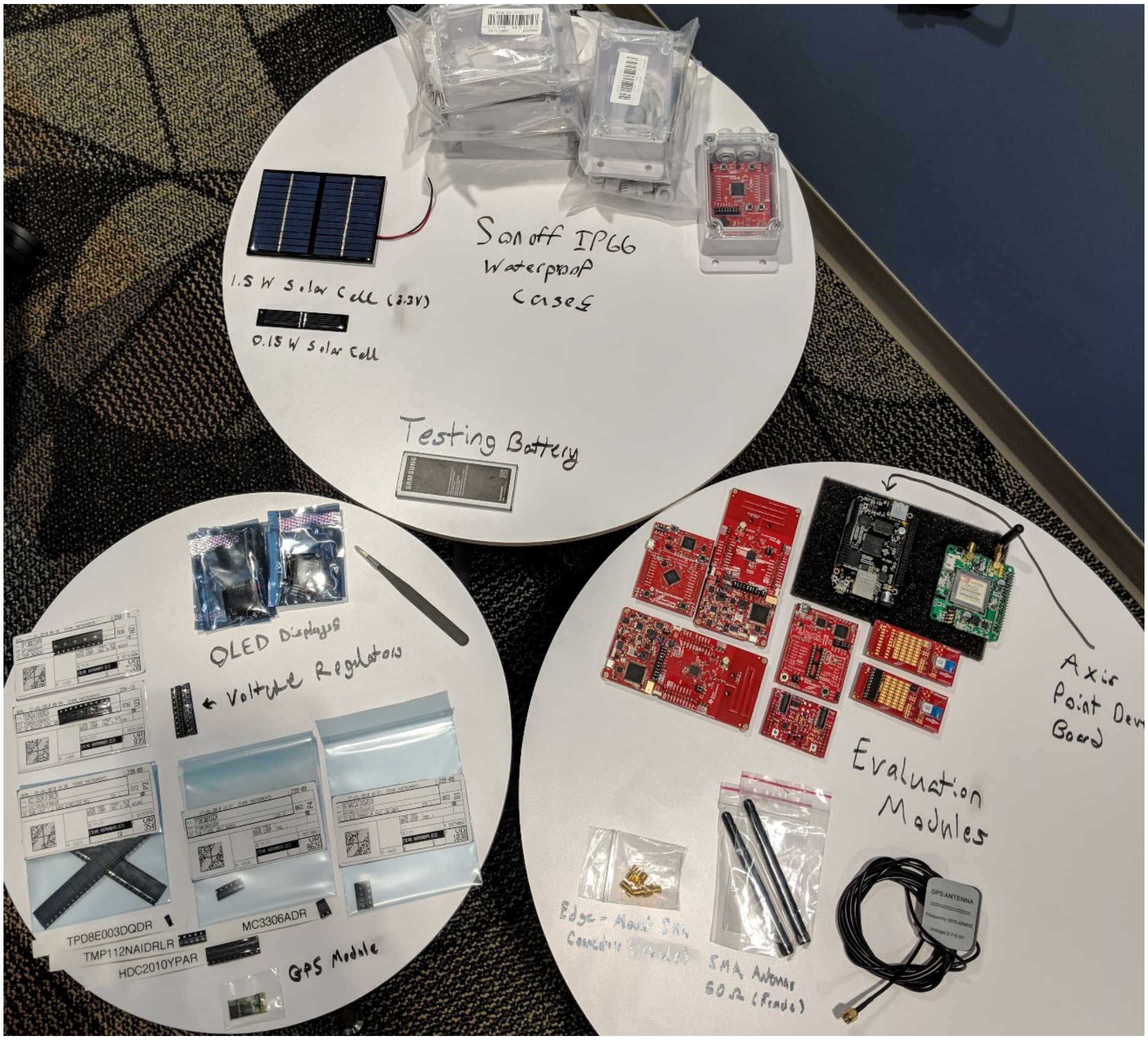


Figure V-2: All Utilized Project Components

The OPT3001 was ordered from Texas Instruments. This light sensor was the least expensive and had the most ideal voltage operation range. This part was found using Digikey but bought on the TI site due to bulk ordering restrictions on Digikey.

The HDC2010 humidity sensor was also ordered from the TI website. This part was the cheapest part with the accommodations Plan Bee required. Because of the ideal pricing, this part was chosen. It is showcased in the photograph below.

The MX25R8035F M1L0 was the flash memory chip chosen because of the reference designs it came with hat were also compatible to other components that Plan Bee was interested in as well. All other requirements were ideal as well; security procedures contained in this part made it worth a bit of extra expense when compared to other chips. This part found user Moser and is displayed below.

The SE4150L and SE4110L GPS chips were both ordered due to the far more helpful datasheets and designs that come with these parts. Both were ordered to test differences between the two; this will help find which is most useful for this project. Both are some of the most inexpensive GPS tracking chips, another reason for ordering them. These two parts are shown below.

The wireless MCU Transceivers CC1352P and CC1352R were ordered; both were the same part, but slightly different models. This chip was chosen because it had the best qualities of all othered compared Transceivers, such as lower required power, and operating on multiple frequency bands concurrently. Both chips were ordered to test differences in range as well as if other minimal differences will affect how well the part works with this project's overall design. These two parts are previewed below.

Peregrin's PE4250 show below was determined to be the most ideal RF chip per Moser. This chip was very affordable and included helpful reference designs, as well as having the best qualities to mesh with other chosen components. Therefore, this part was chosen and is shown below.

The Reverse Polarized SMA Connector by Sparkfun antenna connector was selected because the manufacturer and connection types were the same, making for a more fluid assembly process later. If the antenna chosen ends up not working as well as planned, this antenna connector will probably change as well. This part is shown in the picture below.

The Duck Antenna RP-SMA by SparkFun was the antenna chosen for this project. This was chosen because of how extremely cheap this antenna was, while still operating under required protocol set by Plan Bee. Some qualities of this antenna are lacking, which will be interesting to see play out when testing of this part begins.

The 10W Semi flexible solar panel by Crystal Solar was chosen as the solar panel for this project because it meets the most ideal voltage and current specs when considering this project. This part is also far cheaper than any other compared against it, making it an inexpensive pick for this project. This part is shown below.

SparkFun Electronics' PRT-1351 Li-Ion was the battery chosen for Plan Bee's project because it had the lowest weight available compared to all other batteries to spec, as well as being the least expensive. This was an easy choice and the part was ordered and is photographed beneath.

Deciding which switchable programmable voltage regulator would work best was also accomplished. the MC33063ADR by Texas Instruments was ordered and is shown below. This part was chosen not only because its overall specs were more ideal for this project than any other part compared, but also because of how inexpensive the part was.

The BQ24650RVAT by Texas Instruments was the chose trickle charger for this project because of the most ideal voltage delivered compared to price ratio. The output current is also more desirable than other options, which is why this part was ordered and revealed below.

VI. Design Constraints and Standards

Initial research into project constraints that will limit our end-product, such as interviews with the sponsor, Steven Eisele, in the Project Description section is expanded upon within this section. Design constraints follow the IEEE standards, going over Economic, Temporal, Environmental, Ethical, Health & Safety, Production, Sustainability, and Security constraints. Within the Standards subsection we discuss necessary standards and tutorials on how these standards will be applied to our project, and how each standard is grouped and related. Barriers to Market entry are also included here, because we may be marketing this device toward other beekeepers, and the device will need to be mass produced regardless. This means that the FCC, and other governing boards, as well as other companies, may impede us in realization of our end-product.

I. Design Constraints

I. Economic

The end-product will need to be economical enough to put on every hive within Pollination USA's apiaries. Because of this, it should not exceed more than about one-third of the fixed cost of setup of a new commercial beehive, according to Steven Eisele.

Type	Purpose	Expense
fixed one-time costs	small commercial hive	**$200.00**
fixed recurring costs (monthly)	food	$20.00
	medication	$30.00
	insurance including drought and commercial beehive	$1.20
variable recurring costs (monthly)	labor	$12.40
	transportation	(X)
	hive loss	(X)

* The (X) indicates that the expense is dependent upon contracts and other factors that cannot be linearly estimated on a monthly basis.

Table 25: Typical Expenses for Setup and Upkeep of an Indivual, Small Commercial Beehive

With the given constraint, $70 implementation cost per device, at 5,000 units – on the conservative end of the number of individual beehives owned by Pollination USA – gives us some leeway in design. Our most expensive individual components given our financial projections will be individual load cells and solar cells. We may still be able to reduce these costs further.

The cost of server maintenance or usage should also not exceed one-third of the monthly cost of labor and insurance combined. With this constraint we have a limitation of $4.54 per hive per month on cost of data analysis and storage. This is above the expected costs for using the available IBM Watson and Amazon AWS cloud services. So, these cloud services are still within our economic constraints. This is discussed further in Implementation.

II. Temporal

Our project needs to be completed a month before the end of Senior Design II to meet the requirements of our degree path. All design and development processes will be constrained by the five-month period, at the end of which our end-product must be functional and not just a prototype. We discuss this in more detail within the Milestones section below.

Steven Eisele will be in the Pacific Northeast fulfilling a contract until the end of Summer, as well. Although many beehives will remain in the Everglades during Summer, we will not be able to test our product with his oversight until – at the earliest – early September.

III. Environmental, Communications & Power

Our end-product will be attached directly to the side of a commercial-style beehive; the appearance of this is much like a dresser with drawers, which function as partitions of the beehive. These hives are commonly left in remote locations, more than twenty-five miles away from urban environments. Pollination USA's main yard is about one-hundred miles South-East of Fort Myers, within the Everglades. The tropical-swamp environment of the Everglades provides unique design problems for our sensitive electronics:

- high humidity and daily precipitation,
- salinity of near-ocean air,
- high pressure storm systems moving through South Florida,
- extreme heat (120 F)

As well, Mr. Eisele has noted that he has regular contracts with remote farms in California and in the Pacific-North-West, and the beehives he manages are transported there. Because of this, he would also like the same end-product to work in dry, desert environments and extreme-cold environments (down to 0 F) with snow.

Because of the remote location of the hives, cellular communications channels will not be available. We are thusly constrained to options including satellite communications, building our own cellular base station, or long-range radio-frequency (RF) communications. Satellite communication systems and cellular base stations are prohibitively expensive, so we are left to RF. The remote location will also present issue with powering the device. The best option for us is then to use solar technology in unison with an ultra-low power communications device.

IV. Ethical, Health & Safety

Our end-product will increase the efficiency of work completed, thereby reducing the number of man-hours required upon each beekeeper working for Pollination USA. This may displace jobs. Insurance cost per beehive may also decrease. And if we manage to create the functionality required for the feeding and medication control system that automates non-health maintenance of the individual beehives, then the number of jobs displaced will increase significantly. This is then entire intent of the system though, to save Pollination USA expense. This is not to say that there will not still be beekeepers – there will – just that these beekeepers may not be doing as much "beekeeping" as they would maintenance. To assist in the maintenance of our end-product, we should create a user-manual following this document, requiring minimal IT-experience so as not to require them to be completely retrained. We may, as well, be required to train them on the use and maintenance of our end-product.

On the issue of bystander health, it is possible that the radiofrequency (RF) signal energy could be potentially hazardous to those nearest to these signals, and thereby nearest to our devices, the beekeepers. The FCC has maintained wireless communications standards since 1996 on the issue of RF signal energy exposure limitations for human safety, for all communications devices produced and sold in the United States. [15] The FCC firmly limits exposure to RF energy from wireless communications devices to 1.6 watts per kilogram (W/kg) as averaged over one gram of tissue.

It is possible that a non-beekeeper passerby could be stung by Pollination USA's bees; it is also possible that they could have an allergy toward bee sting venom. Honeybees, most of the bees kept by Pollination USA, are typically docile until an – as determined by the bees – aggressive intruder approaches within ten feet from the hive. [16] It might be possible to somewhat prevent a non-beekeeping passerby from being stung by incorporating a warning that would somehow alert to the presence of bees, while simultaneously not disturbing the bees. This can be an additional feature of our system.

V. Production & Sustainability

Researchers have proven that honeybees' behavior can be disturbed radiofrequency-spectrum electromagnetic fields (RF-EMF). [17] With emitted RF-EMF band above 1 μW/m2 at honeybee-significant frequencies of between 110 Hz and 405 Hz – applied for longer than 45 minutes – is proven to intensify the amplitude and frequency of vibration of the honeybees and aggravate them into an "attacking" state beyond the ordinary limits of the hive, and out of the desirable production and brooding states. This research places a lower-limit constraint on the frequencies of communications we can use for long-distance communications.

The biggest environmental sustainability factor for our end-product is electronic waste and re-usability of the frame-worked device. Our biggest concerns are with the powering of the device – the solar-cell battery (of type Nickel-Cadmium, or NiCad) and the solar cell themselves. We are not as concerned with the durability of replaceable and recyclable sensors and chip electronics. NiCad batteries are the only type that can be used within our extreme temperature constraints, but they have hazardous downsides and typical durability is between seven to ten years. [18] They will however be easily replaceable within our system for the purpose of sustainability and lowering replacement (of whole electronics) cost, so when they are prime for replacement, we need to have an environmental-safety disposal plan ready. This will be an amended section to the user-manual.

VI. Security

The benefits to Pollination USA in our internet-of-things (IoT) sub-1GHz communication template implementation can instead become a huge liability of privacy and business-autonomy loss without proper security protocols in place. [19] As such, the data that is being sent should be secured and encrypted. Further security and encryption constraints domestically are discussed in the software standards section.

As discussed previously, beehive-theft is a somewhat-significant cause of loss to domestic commercial apiaries, like Pollination USA. [20] We have discussed the addition to our end-product whereby every IoT template is to include an embedded GPS locator module, in addition to its 10-mile sub-1GHz range. And an immediate alert whenever the hive is moved, using accelerometers as motion sensors. These must be non-detachable integrated parts within the beehive. With this added constraint, immediate feedback gives rise to immediate measures that can be taken to prevent beehive-theft losses.

Another concern that Steven Eisele had was natural security breaches, of the weather-related and animal-related types. He added that some apiaries, such as his own, already have electric fences installed around each yard to deter wildlife, and he would like a control system to see when the electric fence is tripped – perhaps by a bear – and when this occurs he would like to see images to ensure that the hives are generally unharmed for a period of time after. It would be generally useful to him to receive periodic image updates of each yard, every three-to-four hours, always. In addition to these periodic updates, whenever the motion sensors or weather sensors – perhaps by adding rainfall sensors – attached to the hives are tripped, image updates and alerts are sent in a "burst" over a brief period following the event. These added constraints give the apiarist peace-of-mind to know what security threats their honeybees are facing in any yard at any given time.

II. Standards

I. Hardware

For this section we sought to contrast the available standards and illustrate the reason behind our decisions to adopt the specified features.

Weather-Resistant Enclosures

IPMS – International Protection Marking Standards

As presented in the engineering and marketing specifications, a weather resistant enclosure will be a critical component to the overall device operability. As such, we have researched possible methods of weather proofing the data-collecting device. Enclosures are the most fundamental forms of protecting electronics from the weather. There are currently 2 standards available in United States, for protection; International Protection Marking Standard (IP), and NEMA weather resistance. The figure in the next page shows the IPxx marking system determinations. [21]

The bare minimum of protection we need is IP63, to protect fully against heavy rain and dust. However, IP66 is the most common weather-resistant enclosure standard abided and is more thusly more readily available and less expensive for our product dimensions. IP66 offers further protection against total dust ingress, total immersion protection between 15 cm and 1 meter below water but offers limited water ingress protection beyond this depth. [22]

Solids		Water	
1	Protected against a solid object greater than 50 mm, such as a hand.	1	Protected against vertically falling drops of water with limited ingress permitted.
2	Protected against solid objects greater than 12.5 mm, such as a finger.	2	Protected against vertically dropping with enclosures tilted up to 15 degrees from the vertical axis. Limited ingress permitted.
3	Protected against a solid object greater than 2.5 mm, such as a screwdriver.	3	Protected against vertically sprays of water up to 60 degrees from the vertical axis. Limited ingress protection for three minutes.
4	Protected against a solid object greater than 1 mm, such as a wire.	4	Protected against water splashed from all directions. Limited ingress permitted.
5	Dust Protected. Limited ingress of dust permitted. Will not interfere with the operation of the equipment. Two to eight hours.	5	Protected against jets of water. Limited ingress protection.
6	Dust Tight. No ingress of dust permitted. Two to eight hours.	6	Water from heavy seas or water projected in powerful jets shall not enter the enclosure in harmful quantities.
		7	Protected against the effects of immersion in water between 15 centimeters and 1 meter for 30 minutes.
		8	Protected against the effects of immersion in water under pressure for long periods of time.

IPxx – Ingress Protection xx:
the first x is the numerical standard abided by for Solids;
the next x is the numerical standard abided by for Water.

Table 26: Comparison of IPxx Standards

Figure VI-1: Sonoff IP66 Waterproof Case

The international protection marking standards classify and rate the degree of protection provided by an enclosure against external elements such as dust, rain, and physical force.

To meet the weather resistant and dimensions listed in the requirement specification, we've opted for a Sonoff IP66 rated waterproof enclosure, as shown in the image above.

The enclosure will be large enough to meet our needs while providing two additional outputs for attaching external sensors and an additional output for the Sub-1 GHz and Bluetooth antenna.

NEMA – North American Enclosure Types

NEMA outlines North American Standards for different indoor and outdoor enclosure types; each type of enclosure contains different standards. Because Plan Bee's enclosure will be outdoors, deciding which enclosure to use can be done by observing each type of outdoor enclosure's standards. NEMA has several different enclosure types. Plan Bee is focused on having a waterproof enclosure that is resistant to minerals as well.

Offers A Degree of Protection Against the Following Conditions	Type of Enclosure									
	1 *	2 *	4	4X	5	6	6P	12	12K	13
Access to Hazardous Parts	X	X	X	X	X	X	X	X	X	X
Ingress of Solid Foreign Objects (falling dirt)	X	X	X	X	X	X	X	X	X	X
Ingress of Water (such as dripping or light splashing)	…	X	X	X	X	X	X	X	X	X
Ingress of Solid Foreign Objects (circulating dust, lint, fibers, etc.) **	…	…	X	X	…	X	X	X	X	X
Ingress of Solid Foreign Objects (falling dust, lint, fibers, etc.) **	…	…	X	X	X	X	X	X	X	X
Ingress of Water (Hose-down and splashing water)	…	…	X	X	…	X	X	…	…	…
Oil and Coolant Seepage	…	…	…	…	…	…	…	X	X	X
Oil or Coolant Spraying and Splashing	…	…	…	…	…	…	…	…	…	X
Corrosive Agents	…	…	…		…	…	X	…	…	…
Ingress of Water (Occasional Temporary Submersion)	…	…	…	…	…	X	X	…	…	…
Ingress of Water (Occasional Prolonged Submersion)	…	…	…	…	…	…	X	…	…	…

**These enclosures may be ventilated.*

***These fibers and flyings are nonhazardous materials and are not considered Class III type ignitable fibers or combustible flyings. For Class III type ignitable fibers or combustible flyings see the National Electrical Code, Article 500.*

Table 27: Comparison of NEMA Enclosure Standards

The table above illustrates the degree of protection several types of NEMA enclosures offer. [23] Given the information presented in the table above, a set of enclosure type classifications were narrowed down into four distinct categories that we could utilize for our end-product:

1. Type 4 Enclosure indicates protection against falling dirt, light splashing, circulating dust/fibers, and settling dust/fibers.
2. Type 4X Enclosure indicates protection against all hazards that type 4 includes, as well as protection against corrosive elements.
3. Type 6 Enclosure indicates protection against all type 4 prevented hazards as well as Occasional temporary submersion.
4. Type 6P Enclosure indicates protection against all hazards protected under type 4X, as well as occasional temporary submersion and occasional prolonged submersion.

Hazardous Substances & Safety

ROHS – Restrictions of Hazardous Substances Directive

Restrictions of Hazardous Substances Directive is a standard developed and enforced by the European Union. This standard has been in effect since 2003. This standard promotes recycling of products in a cheaper way for consumers. Because the materials need to be recyclable, certain recourses must be barred from usage in the products under this standard. Such materials include lead, mercury, and certain harmful flame retardants.

PCB FR-4 – Flame Retardant

A NEMA designation for Flame Retardant Printed Circuit Boards. Made with Fiberglass-reinforced epoxy-laminated sheets; such boards are commonly used as the fundamental elements in circuit boards because of how versatile they are. They can withstand high temperatures, are low cost, and work well as an electric insulator. These boards work well in a variety of weather conditions. These boards work well with RF because of the dielectric capabilities and temperature variability. Moisture can also be an issue with RF, which is another reason why this board, with the ability to be highly weather resistant, is a capable choice. [24]

Interconnect Communications

UART – Universal Asynchronous Receiver-Transmitter

This is a configurable, hardware-driven, serial communication device. It is usually connected to an external driver circuit to meet the electric signaling levels required by the RS232 communication protocol. Numerous UART devices are commonly integrated into integrated circuits and are commonly used for serial communication. The CC1352R relies on UART to communicate between its microcontroller and wireless transceiver. There are also 2 additional UART devices: one of which will be used for debugging and testing.

SSI – Synchronous Serial Interface

This communication protocol utilizes synchronous data transmission to transmit information without the need of an external clocking signal. Synchronous data transmission relies on the synchronization of an internal clock which dictates elapse time for incoming data. This maximizes bandwidth while reducing the number of signals needed for serial communication. It is used extensively for RS-422 and RS-485 communication. Because of its benefits, it is routinely used for simplex communication which makes it well suited for sensor communication; it has become the de-facto standard among sensor manufacturer's. For this project synchronous serial interface will be used to log sensors data from externally connected devices; an RS485 driver has been proposed to be used to driver the necessary circuitry.

SPI – Serial Peripheral Interface Bus

Like SSI, SPI is a synchronous serial interface, commonly used for communication between onboard interconnects. SPI electric signaling relies on single-ended signaling instead of differential signaling. As a result, SPI communication is limited in range and will not be used for external device communication in this project. The CC1352R supports both SPI master and slave at a bus clock of up to 4 MHz.

JTAG – Joint Test Action Group

The standard used to verify and test printed circuit board designs after manufacture. To complement digital simulations, JTAG allows an engineer to perform on-chip instrumentation and electronic design automation (EDA). It defines a communication protocol and interface for accessing device level sub-blocks and provide device debugging, instruction tracing, and data tracing infrastructures. While JTAG define a broad set of protocols, many manufacturers routinely implement only a subset define implementation details. As such, it is important to refer to manufacturer specification for further insight.

Encoding & Modulation Communications

Encoding is the first step in transmission of a data across a communications channel. Modulation follows this, before the transmitter-antenna transmits this digital data.

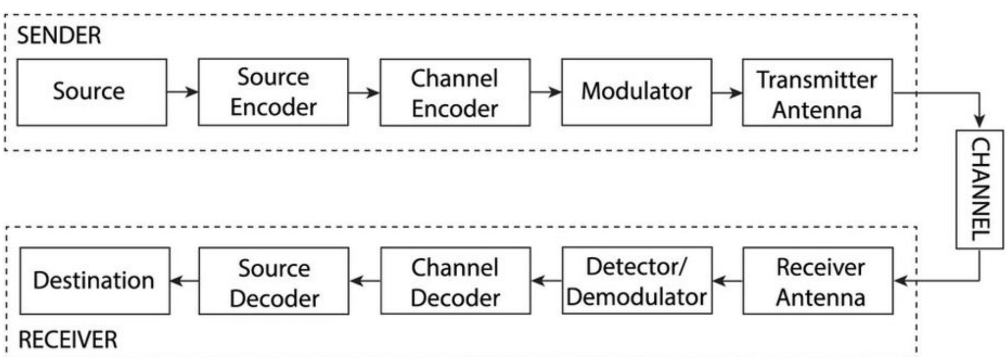


Figure VI-2: Digital Communication Systems Block Diagram

The following standards cover different methods of encoding and modulation that we are going to employ for our communications template end-product. [25]

GFSK – Gaussian Frequency-Shift Keying

There are two main methods whereby this is accomplished. The first, GFSK is a communications standard for utilizing frequency-shift-keying (FSK). FSK is a modulation technique, whereby the frequency of a carrier signal, used to transmit digital information, is altered to symbolize binary information. Gaussian frequency-shift keying applies a Gaussian filter to the data signal. This reduces sideband power and channel interference. This technique is employed by LPWAN manufacturers, including Texas Instruments, and Sigfox. Low powered Bluetooth 5 also relies on this GSK modulation to limit out-of-band interference.

GMSK – Gaussian Minimum-Shift Keying

Similar to GFSK, GMSK is a standard for minimum-shift keying (MSK) encoding of data. MSK is a type of FSK that encodes bits alternating between the quadrature components with the Q-component delayed by half of the symbol period. Dissimilar to FSK, the bits are encoded as half-sinusoids so that the resultant transmitted signal is constant-modulated (constant envelope) allowing for continuous-phase FSK scheme. In GMSK however, the digital data stream is first shaped by a Gaussian filter before being applied to a frequency modulator (encoder) and typically has a much narrower phase shift and transmission band; the benefit of this scheme is reduced sideband power, which in turn reduces out-of-band interference between the signal carriers and adjacent frequency channels. GMSK is used in GSM, for cellular communications, and by GPS-AIS for maritime navigation (under GPS-WAAS). [26]

CSS – Chirp Spread Spectrum Modulation

CSS, which we are going to called Chirp, is a spread spectrum technique that uses wideband linear frequency modulation chirp pulses to encode information. A chirp is a sinusoidal signal of frequency increase or decrease of time, often with a polynomial expression for the relationship between time and frequency. Below is a graph of a modulated upchirp; the vertical axis is amplitude and the horizontal axis is time, although values do not matter at this point. [27]

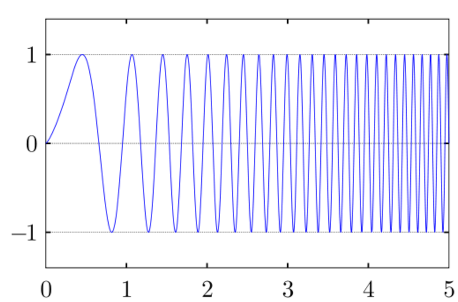


Figure VI-3: Linear Frequency Modulated UpChirp in the Time Domain

Chirp is a technique used to form the physical layer of LoRaWAN. Because LoRaWAN uses chirp to transmit over large distances in the sub-1GHz band range – specifically the 868MHz band – we will be using Chirp for transmitter-receiver modulation and demodulation. [27]

GPS – Global Positioning System

The GPS is a US government owned utility that provides user with real-time positioning, navigation and timing data. It consists of the space segment which includes operating satellites, the controls segment which monitors the satellite, and the user segment which will include our application. [28]

There are four GPS sub-standards:

1. GPS-SSP – Standard Positioning Service Performance is used by the general public "user" applications of GPS, such as cellular phones, vehicular navigation systems, and others similar. This is the GPS standard we will be utilizing for our project.
2. GPS-WAAS – Wide-Area Augmentation Service Performance is used for aviation and marine applications, which translates to usage in airplanes and boats for navigation in unique spatial settings.
3. GPS-PPS – Precision Position Service Performance is used for military applications and is only available to the US military and NATO allies.
4. GPS-CM – Civil Monitoring Performance Specification is always a compilation of requirements specifications and standards required to monitor the GPS civil service and signals based on top-level requirements to monitor all signals and is used by the Department of Transportation for tracking of navigation-enabled motor vehicles. This includes services utilizing Navstar. [29]

GSM – Global System for Mobile Communications

Published and updated by ETSI (European Telecommunications Standards Institute), GSM specifies the use of Gaussian Minimum Shift Keying (GMSK) modulation standards explained prior, with Time Division Multiple Access (TDMA) for the use of data transmission in the 900 and 1800 MHz bands in Europe, and the 800 and 1900 MHz bands in North America. As the name lends itself to be interpreted, GSM sets the global standard for cellular communications networks used by mobile devices, including cellular phones, tablets, and other fringe Internet-of-Things devices. This standard is up-to-date from 1G to 4G communications protocols.

4G-LTE – 4-GSM Long-Term Evolution

This is a high-speed wireless communication standard for phones and always connected devices. It is developed and managed by the 3rd Generation Partnership Project (3GPP). [30] The specification defined a downlink rate of 300 Mbit/s and uplink of 75 Mbit/s, and a transmission latency of less than 5 milliseconds in the radio access network. LTE relies on frequency division duplexing (FDD) or time division duplexing (TDD) to manage data transmission and reception.

GPRS – General Packet Radio Service

GPRS, established by ETSI, is a packet-oriented mobile data standard within GSM limiting packet size, formatting, and data speeds for guaranteed Quality-of-Service in 2G, 3G and 4G (LTE) networks. It will likely amend its standards for 5G networks which are being researched currently, continuing in relevance long into the future. The GPRS Core Network is the primary function of GPRS which allows 2G, 3G and WCDMA mobile networks to transmit IP packets to external networks through the Internet. Because we will be making use of the 3G and 4G technologies with TI CC1352R, and 1352P eventually, we will be utilizing this standard. The table below displays the coding schemes and associated speeds for the GPRS subset of standards:

GPRS Coding Scheme	Bitrate incl. RLC/MAC * (kbits/s/channel)	Bitrate excl. RLC/MAC (kbits/s/channel)	Modulation Type	Code Rate
CS-1	9.20	8.00	GMSK	1/2
CS-2	13.55	12.00	GMSK	2/3
CS-3	15.75	14.40	GMSK	3/4
CS-4	21.55	20.00	GMSK	1

** Cited in several sources (TS 45.01 Table 1), is the determined bitrate including RLC/MAC headers but excluding the uplink flag (USF) which is part of the MAC header, yielding a bitrate 0.15 kbits/s/channel lower.*

Table 28: Comparison of GPS Coding Sheme

The Bitrate including RLC/MAC overhead is the rate at which the RLC/MAC layer protocol data unit (PDU) is transmitted; this consists of the MAC header, RLC header, RLC data unit and spare bits. The RLC data unit represents the payload, the rest of "overhead". The PDU is encoded by the convolution code specified for a GPRS coding scheme, which yields the same PHY layer data rate for all coding schemes. We will not be able to utilize this full data rate. [31]

The next bitrate here is the rate at which the RLC/MAC layer payload (the RLC data unit) is being transmitted. As such, this bitrate excludes header overhead from the RLC/MAC layers. It is the amount of data we can use to transmit in kbits/s/channel.

UMTS – Universal Mobile Telecommunications System

UMTS is a subset of standards of the International Telecommunications Union IMT-2000 standards and sets telecommunications standards for cellular devices using third-generation GSM standards, 3GPP (3rd Generation Partnership Project). In other words, the use of UMTS will enable us to use the Basic 3G cellular network, allowing a maximum data transmission rate of 300 Kilobits per second within the Sub-1GHz communications band. `

Encryption Communications

AES – Advance Encryption Standard

This is an encryption standard used to secure data for remote transmission. The algorithm used to define the encryption is symmetric, meaning a single key will be able to encrypt and decrypt data files. AES has been adopted by the federal government and is the only key cipher approved by the NSA.

SHA2 – Secure Hash Algorithm

Algorithms that secure data by sending it through bit strings of a fixed size in only a single, non- invertible direction for it not to be intercepted. This is encrypted and analyzed on the receiving end and compared to a specific bit value, making it the only way to obtain the information.

ECC – Elliptic Curve Cryptography

Ecliptic Curve Cryptography is a replacement cryptography for Public Key Cryptography. As the NSA web archives state, "While at current security levels elliptic curves do not offer significant benefits over existing public key algorithms, as one scales security upwards over time to meet the evolving threat posed by eavesdroppers and hackers with access to greater computing resources, elliptic curves begin to offer dramatic savings over the old, first generation techniques." [32] ECC is a cost-efficient technology when compared to past technologies, especially over time. Past systems that applied public key were recommended for upgrade by NIST, and ECC was a viable successor.

RSA – Rivest-Shamir-Adleman Encryption

Rivest-Shamir-Adleman Encryption can also be called Public Key Cryptography. RSA bit-keys are considerably longer than the keys used today, such as ECC. RSA is a system utilizing public key; this system is more outdated than ecliptic key. The table below illustrates the NIST recommended key sizes, comparing Symmetric Encryption, RSA and Diffie-Hellman Encryption, and Elliptic Curve Encryption.

Symmetric Key Size (bits)	RSA and Diffie-Hellman Key Size (bits)	Elliptic Curve Key Size (bits)
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	521

Table 29: Comparison of TCP/IP Data Encryption Methods

The greater the key size in bits, the more encryption and safer the data being transferred through a wireless channel. We will be using RSA standards with Diffie-Hellman Key Agreement, explained below, as they are proven industry standard encryption methods wireless communications. [33]

Diffie-Hellman Key Agreement

Diffie-Hellman Key Agreement uses the public key system and private key system like the RSA does. When combining the use of a public key from one user with a private key from another, a secret encryption key may be shared between the two parties. This key can be used to encrypt data. [34]

DSA – Digital Signature Algorithm Encryption

The Digital Signal Algorithm is a Federal Processing Standard. A DSA falls under the Digital Signal Standard. The DSA is a standard used in the DSS. The DSS implements digital signature in leu of written signature. There are two components of the DSS, which are the creation of the digital signature itself, as well as the verification of that signature. Different keys are used for these two key components. Signature design itself uses a private key, whereas signature verification uses a public key. Public keys are used in many variations of systems, including RSA. [35]

II. Software

Software Communications

TCP – Transmission Control Protocol

TCP is the main protocol in the Internet Protocol (IP) Suite, at the "Transport Layer". It provides reliable, ordered and error-checked delivery of a stream of bytes between applications running on a hosted web server communicating via an IP network (usually masked for encryption). All web-based applications such as the World-Wide-Web, email, remote administration and FTP (File-Transfer-Protocol) rely on TCP. Because our application will be hosted on a web server, and require encrypted communication via an IP network, this standard is integral for server-side development. [36] [37]

IPvx – Internet Protocol Version X

IP, meaning Internet Protocol, is the principal communications standard for the Internet Protocol Suite, and sits at the "Internet Layer". It provides relaying of datagrams across network boundaries, and its routing function enables internetworking, i.e. the "Internet". IP specifies how data is divided into packets from the source host to the destination host based upon the IP addresses in the packet headers and encapsulates this data to be delivered. [38]

IPv4 is the predecessor of IPv6; the most prominent difference from version 4 is the size of the addresses, from 32-bits for addressing in IPv4 to 128-bit addressing in IPv6, allowing the number of internet addresses to expand by almost 29 orders of magnitude (x 1029). IPv4 is still the dominant IP standard, although IPv6 is growing quickly. The IP version we use will depend upon the web-server host that assigns our address. Because we will be establishing our server-side a new, we prefer to be using the newest standard under adoption, IPv6. [39] [40]

UDP – User Diagram Protocol

UDP mostly aids as an interface between high layer networking processes and the internetworking of IP. UDP is the successor to Transmission Control Protocol. It is less complex; UDP does not have some of the unnecessary qualities that TCP contained. This decrease in complexity also increased performance of UDP. Because of this lack in complexity, UDP cannot be used in as many applications as TCP. [41]

NanoIP – Nano Internet Protocol

NanoIP is a networking protocol; it works well with connectivity to devices, even when first connecting through a gateway. This technology is designed to bring networking to devices, but without the complexity of TCP. This standard contains nanoUDP and nanoTCP. NanoIP does not actually use IP addresses, but rather MAC addresses. NanoIP is faster and more simple than alternative protocol that uses IP. [27] [42]

CoAP – Constrained Application Protocol

CoAP is useful because it is designed to still be useful when more advanced technology comes about. It has also been designed to work on inexpensive technology, making it cost effective. With CoAP, resources can be easily accessed under a URL. This technology is efficient and has a similar style as a few other protocols, such as HTTP, making it user friendly. CoAP uses UDP as well as IP to operate with simplicity. [43]

DDS – Data-Distributed Service for Real-Time Systems

DDS is and API standard from OMG (Object Management Group). DDS assimilates components of a system together with low-latency and dependability. Using this protocol simplifies the process of passing data so that consumers can focus on creating their specific product. This standard has a high-performance rating when compared to similar technologies and condenses the amount of time and work that needs to be put into developing a product. [44]

SOAP – Simple Object Access Protocol v1.2 (latest version)

SOAP is known for its simplicity and flexibility. SOAP attains these features by excluding features such as security and reliability. SOAP has more protocol stacked on HTTP, such as WSDL; REST does not include these features. When HTTP proves to not be useful enough, SOAP is a helpful solution. This framework is designed to be autonomous of any explicit previous style used. [45]

IEEE P2413 – Standard for Architectural Framework for the Internet of Things (IoT)

This standard defines multiple protocol for the IoT. This includes IoT domains, compatibility of the overall system, and continuous market growth monitoring. The main goal of the IoT is expansion. [46]

WebSocket

WebSocket is an HTML standard; it is used for Web applications and communication. WebSocket is known for its communications procedure. WebSocket is usually used with TCP but can be used with UDP under certain circumstances. WebSocket also includes HTTP. This protocol allows information to be sent between two users. [47]

Weave (going to use this for Beacon)

Weave aids phone-to-device-to-cloud communication. Weave helps connect to the cloud through mobile phones as well as desktop. Weave can bridge the gap of communication for device-to-device, device-to-cloud, and device-to-mobile. Weave is known for having low overhead, scalable properties, flexibility, and security even on a vulnerable network.

This would be useful for Plan Bee's connection to the beacon. Being able to gain access from a mobile phone would be extremely useful in this situation as well. [48]

Web Server Development

JSON – JavaScript Objection Notation

JSON is based off of the JavaScript programming language, however it is independent of any particular coding language. Because it has similar conventions to other languages, JSON is known for its ease of use. JSON transmits data in a way that is understandable to humans, not just a computer. Because of this, more arranged data can be conveyed. JSON usually transfers data between an application and a server. [49] [50]

XMPP/XMPP-IOT – Extensible Messaging and Presence Protocol for Internet of Things Devices

XMPP stands for Extensible Messaging and Presence Protocol. XMPP was designed to be flexible and scalable; this is what extensible represents. XMPP also allows users to IM for communication purposes. This real-time messaging is represented by the M in XMPP. Presence in XMPP is characterized by showing other users when a user is online or offline. XMPP is a commonly used protocol that allows systems to communicate back and forth. This is another protocol used within the IoT. [51] [52]

SQL – Structured Query Language

SQL is a standard for the American National Standard Institute (ANSI) and for the International Organization for Standardization (ISO). SQL and is basically used for retrieving and editing databases. Different versions of SQL can be used depending on the specifications the user is looking for. SQL uses the Relational Database Management System to retrieve data for the user for editing. This is a useful and uncomplicated way to obtain data. [53]

NoSQL – Non SQL (Structured Query Language)

Model	Performance	Scalability	Flexibility	Complexity	Functionality
Key-Value Store	High	High	High	None	Variable (none)
Column-Oriented Store	High	High	Moderate	Low	Minimal
Document-Oriented-Store	High	Variable-to-High	High	Low	Variable-to-Low
Graph Database	Variable	Variable	High	High	Graph Theory

Model	Performance	Scalability	Flexibility	Complexity	Functionality
Relational Database (SQL-like)	Variable	Variable	Low	Moderate	Relational Algebra

** This data for this table was created by Ben Scofield and performance was rated for different categories of NoSQL databases using the YCSB benchmark.*

Table 30: Comparison of Relational Database Types Available for NoSQL

NoSQL database standards provide a mechanism for storage and retrieval of data that is modelled by means other than the tabular relations used in SQL, relational databases; it is a nonrelational database standard. It is however worth noting that SQL-like databases can exist within NoSQL databases, so NoSQL is broader, and "Not-only-SQL". There is not one way of handling data, but several, which are displayed in the table above. [54]

It would not be worthwhile to explore each individual method here, as whatever data basing method we use is likely going to depend upon the web host we choose, further explored in Research and Software Implementation sections. We would however prefer to use NoSQL for the flexibility it offers. [55]

Front-End Web Page Development

HTLM5 – Hypertext Markup Language v5 (Current)

HTML5 is a markup language for use in structuring and presenting content on the world-wide-web (WWW). It is the fifth and current version of the HTML standard. Web browsers receive HTML documents from a web-server and render the documents into structured web pages using CSS3 and JS standards, the triad of cornerstone technologies for the WWW. HTML allows the institution of constructs, images, interactive forms, text, and other web "objects". HTML also provides a means to create structured documents by denoting structural semantics of text such as headings, subheadings, paragraphs, lists, hyperlinks, quotes and other items. One of the most prominent features of HTML for us is the ability to hyperlink to other pages, allowing navigation on the WWW, and through our own application. [56] [57]

CSS3 – Cascading Style Sheets v3 (Current)

HTML5 uses CSS3 to "style" its pages, making them visually appealing and adding functionality. More formally, CSS it is used for describing the presentation of a document written in HTML. CSS enables us, in development of the front-end web application, to separate our presentation of content including grid, layout, colors, and fonts, improving content accessibility, providing more flexibility and control in the specification of presentation characteristics. CSS also enables multiple webpages to share the same formatting by specifying relevant (general styling) CSS in a separate ".css" file; this significantly reduces the complexity and repetition in structured content of our code. [58]

JS – JavaScript

Alongside HTML5 and CSS3, JS is one of the three core technologies that power the WWW. JS is a higher-level scripting language influenced by C (and Python) that enables functional and visual interactivity of web pages and is thusly essential for all web-based applications. As most websites on the WWW use it, all major web browsers (Firefox, Chrome, Explorer, Edge) have a JS engine to execute scripts. [59]

JS enables support for event-driven, functional, and imperative (including object-oriented and prototype-based) programming styles, and has APIs for working with text, arrays, dates, regular variable expressions, and basic manipulation of the document-object model (DOM) for the frontend of web-based application. As it is not a server-side language (as discussed above), JS does not include any I/O, such as networking, storage, or graphics facilities, relying for these upon the host environment in which it is embedded. [60]

III. Barriers to Market Entry

I. FCC Equipment Authorization for RF Devices

During prototyping, as Researchers, we can use any protocols we deem necessary to get a functional end-product we intend to use. This end product must fit within the national constraints of the FCC. Further, any RF Device that will be manufactured and produced at scale requires subjection to peer review through FCC Equipment Authorization for RF Devices, and if approved, listing within the FCC.

Our communications template device will specifically be classified as an Intentional Radiator Equipment, as it will feature an RF transmitter-receiver for long-range communication and short-range mesh-networking. We will be using the upper-ISM bands, and our power output (in Watts) and range will be limited by regulations governing the safety of animals and individuals, exactly as we intended. Intentional Radiators are defined in Section 15.3(o) in the Electronic Code of Federal Regulations guidelines. Exemptions to Intentional Radiators are listed in 15.201 of the FCC Federal Guidelines. These devices are subject to Equipment Authorization Program by the FCC. A detailed explanation of the FCC's' Equipment Authorization Process follows.

Requirements Imposed on Intentional Radiators

Frequency ranging from 902-928 MHz Hopping shall use min of 25 kHz spacing or the 20 decibels (dB) bandwidth, whichever is greater. For less than 250 KHz bandwidths, system shall use at least 50 channels and the average time on channel shall be less than 0.4 seconds in a 50 seconds interval. (section 15.247)

There are no special testing requirements. Frequency usage should eliminate the end channels to as to not violate the bandwidth rule given above. Frequency tolerance is not specified, only that the 20-dB bandwidth shall fall inside the 902-928 MHz band. (section 15.215c)

Environmental limits are not placed on the temperature range of the operating device (intentional radiator). Power output is limited to 30 dBm max specifications, with no more than a 6 dBi antenna. Implies Ptransmitter is 30 dBm including cable loss. Spurious levels specifically state emissions outside the band shall be negative 50 dBc or less attenuation. (section 15.209)

Equipment Authorization Procedures

Legal Proceedings

Section 2.906 – Supplier Declaration of Conformity (SDoC)

SDoC is a procedure where the responsible party, as defined in section 2.909, makes measurements or completes other procedures found acceptable to the Commission to ensure that the equipment complies with the appropriate technical standards. Submittal to the Commission of a sample unit or representative data demonstrating compliance is not required unless specifically requested pursuant to §2.945. SDoC is applicable to all items subsequently marketed by the manufacturer, importer, or the responsible party that are identical, as defined in §2.908, to the sample tested and found acceptable by the manufacturer. The responsible party may, if it desires, apply for Certification of a device subject to the Supplier's Declaration of Conformity. In such cases, all rules governing certification will apply to that device.

Section 2.907 – Certification

Certification is an equipment authorization approved by the Commission or issued by a Telecommunication Certification Body (TCB) and authorized under the authority of the Commission, based on representations and test data submitted by the applicant. Certification attaches to all units subsequently marketed by the grantee which are identical (see §2.908) to the sample tested except for permissive changes or other variations authorized by the Commission pursuant to §2.1043.

Section 2.908 – Identical Product Already Registered

As defined by the FCC, is used in this subpart, the term identical means identical within the variation that can be expected to arise because of quantity production techniques.

Approval

Supplier's Declaration of Conformity (SDoC)

The responsible party, as specified in the rules, warrants that each unit of equipment complies with the applicable FCC rules. The responsible party maintains all the required documentation demonstrating compliance with the applicable FCC rules. The responsible party prepares a compliance information statement to be supplied with the product at the time of marketing.

Certification

The responsible party, typically the manufacturer, obtains an FCC Registration Number (FRN) for a device requiring Certification. The FRN is a 10-digit number used to identify the individual or organization doing business with the FCC. The same FRN will be used for future approvals. After obtaining an FRN, the responsible party obtains a Grantee Code from the Commission by applying at the Grantee Registration website. A grantee code is required the first time a party applies for certification and can be used for all future approvals.

The responsible party files with a Telecommunication Certification Body (TCB) an application for a grant of certification. An application for equipment authorization requires submission of information about the product, as listed in Section 2.1033. The applicant must submit the required information to a TCB for review as part of the certification process. [For a list of FCC recognized TCBs see https://apps.fcc.gov/oetcf/tcb/reports/TCBSearch.cfm]

The TCB reviews all the supporting information and the evaluation results to determine if the product complies with the FCC requirements. Once the TCB decides to certify the product the supporting information is uploaded to the FCC Equipment Authorization Electronic System (EAS) – Database. A grant of certification is issued by the TCB on the FCC Equipment Authorization Electronic System (EAS) – Database.

Following Approval

Label, Manual and Record Retention

The manufacturers are then to correctly label the product and provide the required customer information. The manufacturer must also maintain all documentation as part of the responsibility for the retention of records and ensure that the manufactured products are complying. For more information, see Section 2.938 – Requirements for the retention of records of equipment subject to FCC approval.

Manufacturing, Importing and Market Approval

Changes to your product design may require an additional approval. KDB Publication 178919 gives general guidance when making changes to a previously approved product. See the permissive change rules in FCC Section 2.1043. Modifications that may be made to an RF device without filing for a new equipment authorization; Three distinct types of permissive changes; and identifies when a permissive change filing with the Commission is required.

VII. Design Implementation

This section includes discussion about hardware implementation, breadboard testing, schematic and block diagrams, software implementation, necessary flow charts, and hardware and software interfacing. Hardware implementation is shown through schematics of each "block" of the block diagram, and the breadboarding for each of these is also shown and discussed. Connections are then made at the general PCB-design level and discussed. The software implementation is shown through discussion and flow chart, and references to the Appendix section which includes relevant code snippets – A4. The backend and frontend software interfaces and technology implementation are also shown and discussed.

I. Hardware

I. Device-Specific Reference Design Details

Summary

This section features recommended device implementation details. We recount the manufacturer reasoning behind their implementations and discuss how such implementation details will be integrated into our final project. The overall design and implementation of the circuit will be done using Altium Designer. This program is easy to work with and extremely useful for PCB design. Plan Bee needs to focus on maintaining a seamless PCB design. This includes things such as making sure all components are of the same mount type (surface mount), as well as creating the most efficient design. This includes placing components in the easiest to work with and useful manner. By practicing these concepts, many of the schematics below have been created.

Power

BQ24650RVAT

The BQ24650RVAT comes designed to effectively provide DC power by having an entry side that cuts out any alternating signal. It also comes with a planned schematic for a thermistor to help sense its temperature besides the circuitry necessary for proper feedback for the MPPT system and battery level feedback system.

The BQ24650RVAT is a switching regulator with a switching frequency of 600 kHz which is ideal in order to have small inductor and capacitor values. The inductor current is directly proportional to the charging current as well as the ripple current which is in turn dependent of the input voltage as well as the duty cycle (D=Vout/Vin).

$$f_o = \frac{1}{2\pi\sqrt{L_o C_o}}$$

Figure VII-1: Switching Regulator Frequency

BQ24650RVAT circuit set up also comes with an input, as well as an output capacitor. Both input as well as output capacitors should have enough ripple current rating to absorb output switching ripple current so that the output will be an efficient DC power. For the input decoupling capacitor, a low ESR ceramic capacitor is preferred. Also, the voltage rating of the input capacitor must be higher than the normal input voltage level. As for the output capacitor, increasing both its capacitance as well as the value of the output inductor can help reduce the voltage ripple at certain input and output voltages. In order for the

BQ4650RVAT to have a stable internal loop compensator, it must have the resonant frequency of the output inductor and output capacitor must be designed between 12 kHz and 17 kHz with a preferred capacitor of 35 Volts or higher rating. It also may be necessary to choose a higher voltage rating or nominal capacitance value since the output capacitor shows a de-bias effect that can significantly drop the capacitance if the voltage is too high.

$$I_{CIN} = I_{CHG} \times \sqrt{D \times (1 - D)}$$

Figure VII-2: Input Capacitor Current

$$I_{COUT} = \frac{I_{RIPPLE}}{2 \times \sqrt{3}} \approx 0.29 \times I_{RIPPLE}$$

Figure VII-3: Output Capacitor Current

The IC set up for the BQ24650RVAT also requires for two N-channel MOSFETS when it uses a synchronous switching mode. The gates are connected to the IC at 6 Volts and 30 Volt rated MOSFETS are sufficient for at most a 20 Volt input. In order to choose a proper N-channel MOSFET its Figure of Merit (FOM) is examined. For the top-side MOSFET the FOM is defined as the product between the on-resistance and the gate-to-drain charge, and the FOM for a bottom-side MOSFET is the product of the on-resistance and the total gate charge. The lower the FOM, the lower the total power loss. The top-side MOSFET also includes conduction loss and switching loss.

$$FOM_{top} = R_{DS(on)} \times Q_{GD}$$

Figure VII-4: Top MOSFET Figure of Merit

$$FOM_{bottom} = R_{DS(ON)} \times Q_{G}$$

Figure VII-5: Bottom MOSFET Figure of Merit

The first item represents the conduction loss. Usually MOSFET RDS(ON) increases by 50% with 100°C junction temperature rise. The second term represents switching loss.

$$P_{top} = D \times I_{CHG}^{2} \times R_{DS(ON)} + \frac{1}{2} \times V_{IN} \times I_{CHG} \times (t_{on} + t_{off}) \times F$$

Figure VII-6: Top-Side MOSFET Loss

$$I_{on} = \frac{V_{REGN} - V_{plt}}{R_{on}}; \; I_{off} = \frac{V_{plt}}{R_{off}}$$

Figure VII-7: Gate driving current total

$$P_{bottom} = (1 - D) \times I_{CHG}^{2} \times R_{DS(ON)}$$

Figure VII-8: Bottom-side MOSFET Loss

For the BQ24650RVAT to work at its most efficient way, the MPP (Maximum Power Point) must also be set for maximum efficiency. Maximum power comes from the direct relationship between voltage and current and can be expressed by the solar panel's I-V curve.

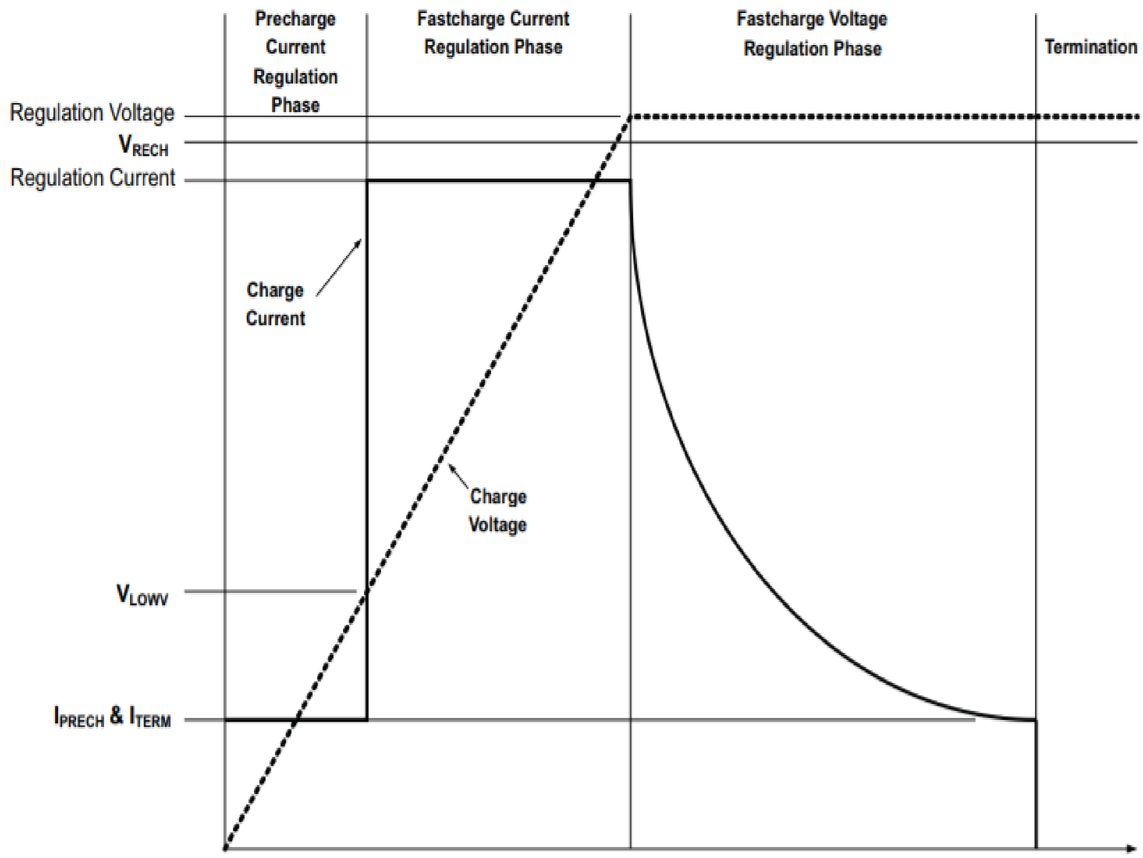


Figure VII-9: Charging Cycle

$$V_{BAT} = 2.1\ V \times \left[1 + \frac{R2}{R1}\right]$$

Figure VII-10: General Charge Transfer Function

The MPP can be derived by controlling the change between output current and output voltage. At no output current the voltage would be at its maximum, but its power would effectively be zero. As output current from the panel increases the output voltage decreases eventually no voltage and therefore no power, but before reaching that stage there comes a point in which the increased output current and decreased output voltage multiplied together give the highest possible output power which is the MPP and this point is measured to be most efficient at a temperature of 25°C. In order to control how much maximum voltage and how much maximum current can be acquired the setup of the panels themselves is controlled as solar cells that are connected in series increase output voltage while solar cells connected in parallel increase output current.

$$V_{MPPSET} = \frac{V_{IN} \times R_B}{(R_{EQ} + R_B)}$$

Figure VII-11: MPPT Voltage

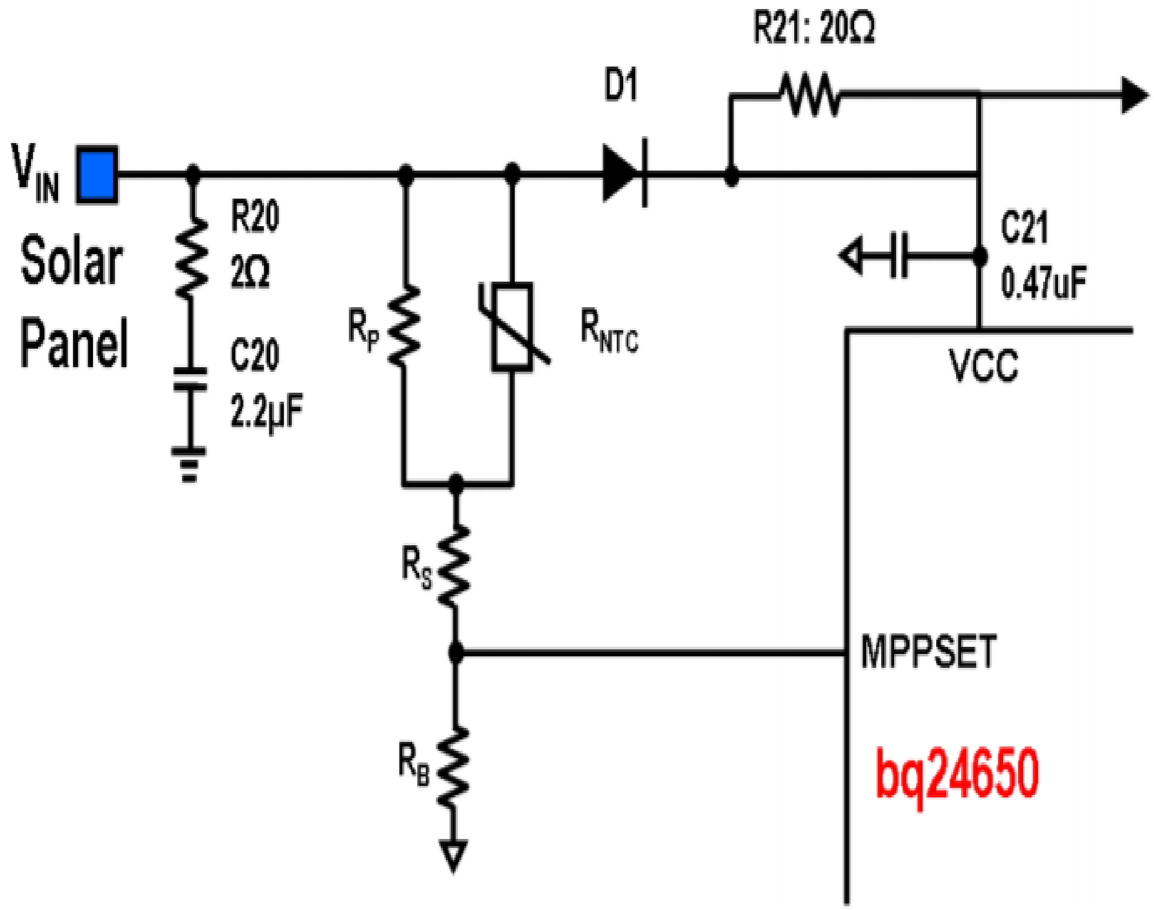


Figure VII-12: BQ24650RVAT Reference Design

Because the MPPT setting works better at certain temperatures than others, the BQ24650RVAT datasheet comes with directions for the implementation of a negative temperature coefficient (NTC) pack thermistor; negative is chosen because then the thermistor decreases in resistance as the temperature increases. As temperature increases, the established maximum point voltage decreases, and if the temperature increases too much then the IC even stops all charge. This thermistor along with a voltage divider create a voltage drop between the TS pin and ground that is compared to the internal settings of the IC to know if it should start up charge or if it should continue charging.

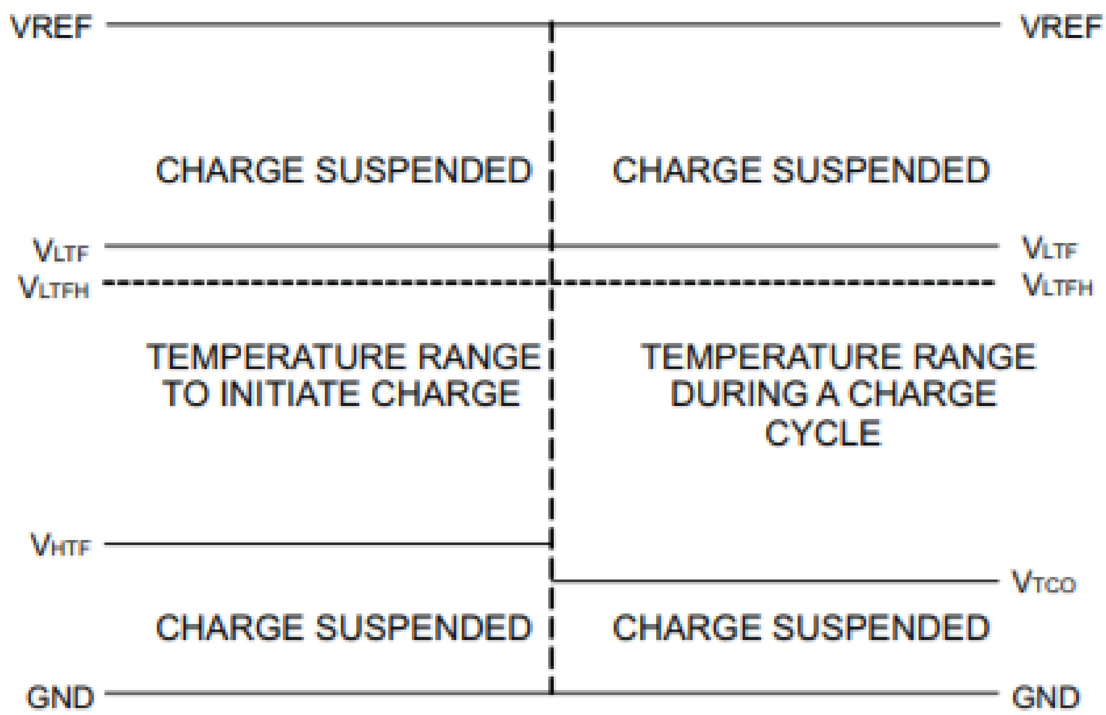


Figure VII-2 Thermistor Threshold

If the voltage between the positive and negative charge current sensor resistors (SRP/SRN) falls below 5mV, the IC enters into non-synchronous mode and the current goes through the body diode of the bottom-side MOSFET. This is in order to avoid negative conduction when the switching is occurring.

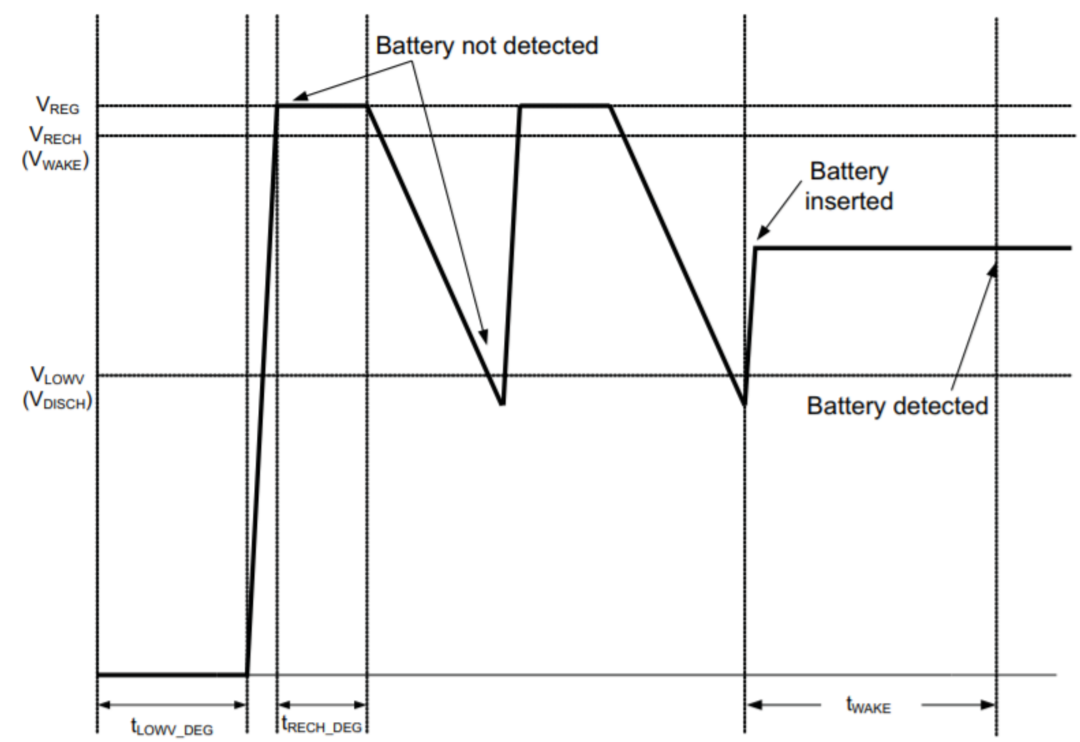


Figure VII-3 Battery Detect Timing Diagram

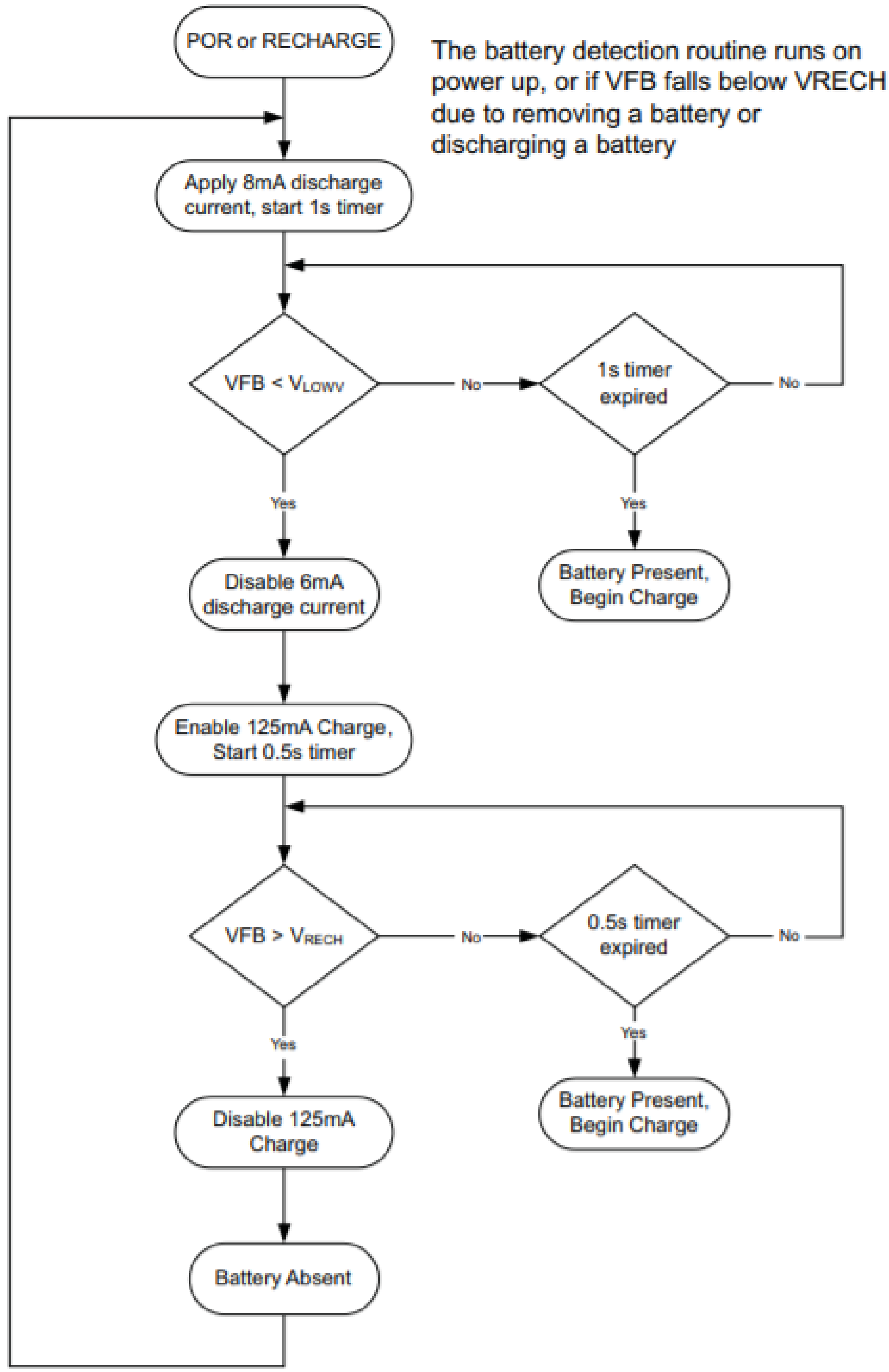


Figure VII-4 Battery Detect Flowchart

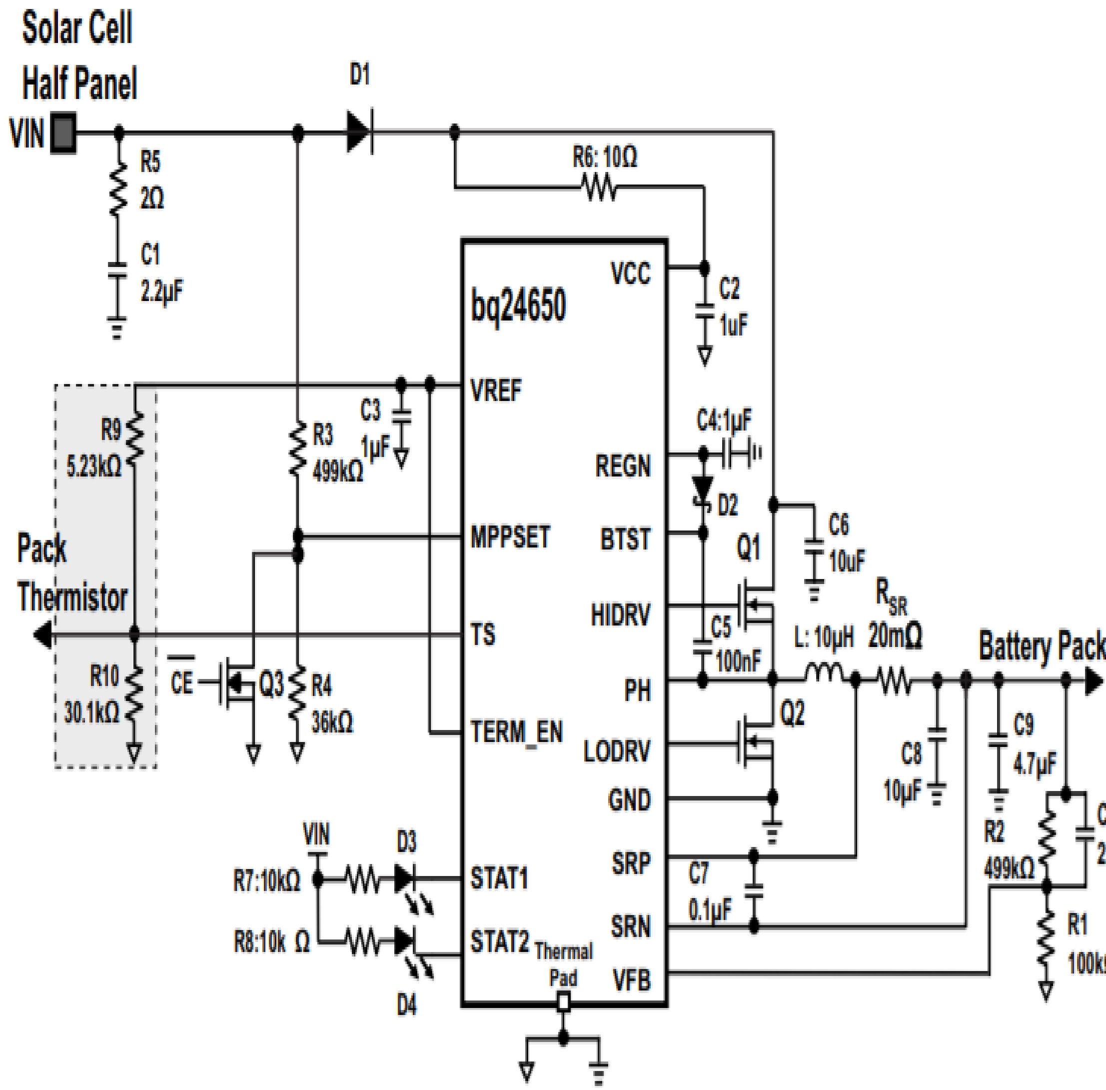


Figure VII-5: Solar Cell Half Panel Reference Design

MC33063ADR

This device is integrated so as to take any DC voltage and convert it to a DC voltage of a different value. This IC can be used for both a step-up converter and a step-down converter depending on the design and placement of its outer components. The device consists mainly of an internal temperature-compensated reference, a comparator, an oscillator, pulse width modulation controllers with active current limitation, a driver, and a high-current output switch.

For our project we will be stepping down the high voltage from the solar panel charged battery into a lower voltage and power suitable for the function of the device and there are different possible integrations for this. One such difference is whether or not our converter will need to push out more than 1.5 Amperes or not as then it would need an external BJT, but the main difference observed in regard to this project's application is the one between a non-synchronous DC-DC converter, and a switching regulator. Though the switching regulator provides many advantages such as a much faster switching efficiency, less energy is lost in the transfer requiring less thermal management and is the ideal and only way if we want to step up or 'boost' our voltage, it is much more convenient to use a DC-DC converter. This project will only rely on DC power from the battery and we will only need to step down voltage in this phase so non-synchronous converter would be sufficient. The non-synchronous also creates much less noise than the switching regulator. The DC-DC converter uses a simple mechanism of having all current flow through an output inductor which in turn turns into a voltage supply once it dissipates its current. How often it dissipates depends on the wanted duty cycle: the higher the duty cycle, the higher the output voltage.

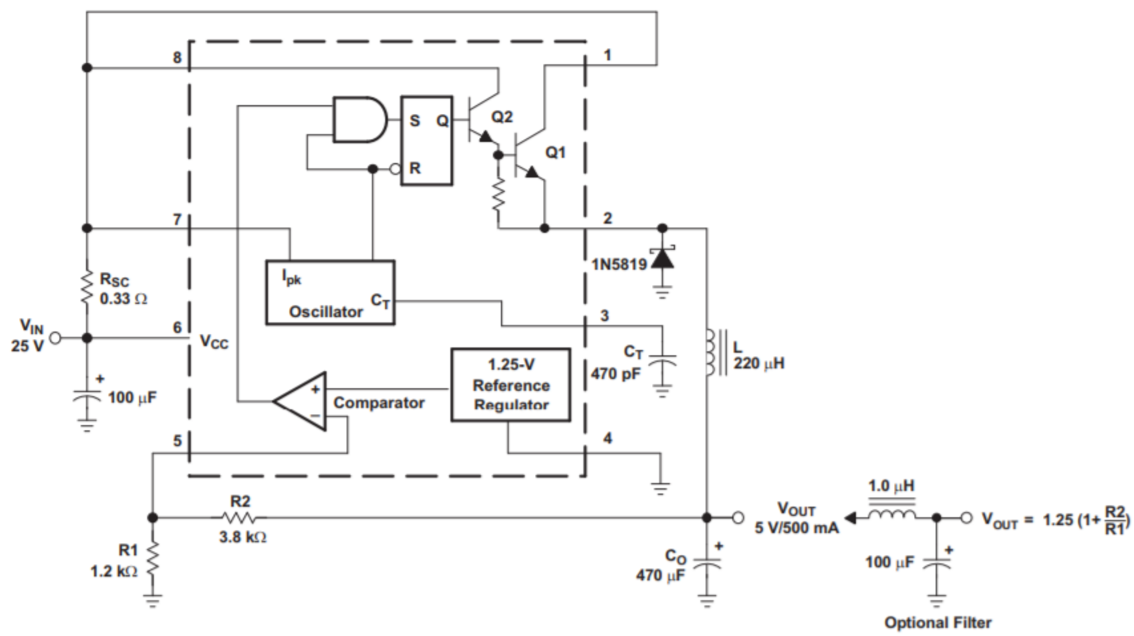


Figure VII 6: DC Step-Down Converter Reference Design

Sensors

II. TMP112

Because the TMP112 is a temperature sensor, making sure to alert the owner of temperature abnormalities is a requirement. This has been looked at by Plan Bee when considering board selection. The TMP112 temperature sensor has certain schematic configurations that increase its productivity and efficiency. This includes noise reduction configurations, filters, and required pullup resistors. Knowing what components are required for the TMP112 as well as what configurations help increase the sensor's abilities will help Plan Bee with the design and implementation of a PCB schematic that includes this chip.

The TMP112 needs pullup resistors on three pins; a value of around 5kΩ should be used for the SCL, SDA, and ALERT pins. This recommendation is for preventing the current from surpassing 3 mA at any of these three pins. A capacitor of .01µF can also be used as a bypass on the supply voltage. This capacitor should be placed near as possible to the supply and ground pins. This recommended design is shown below.

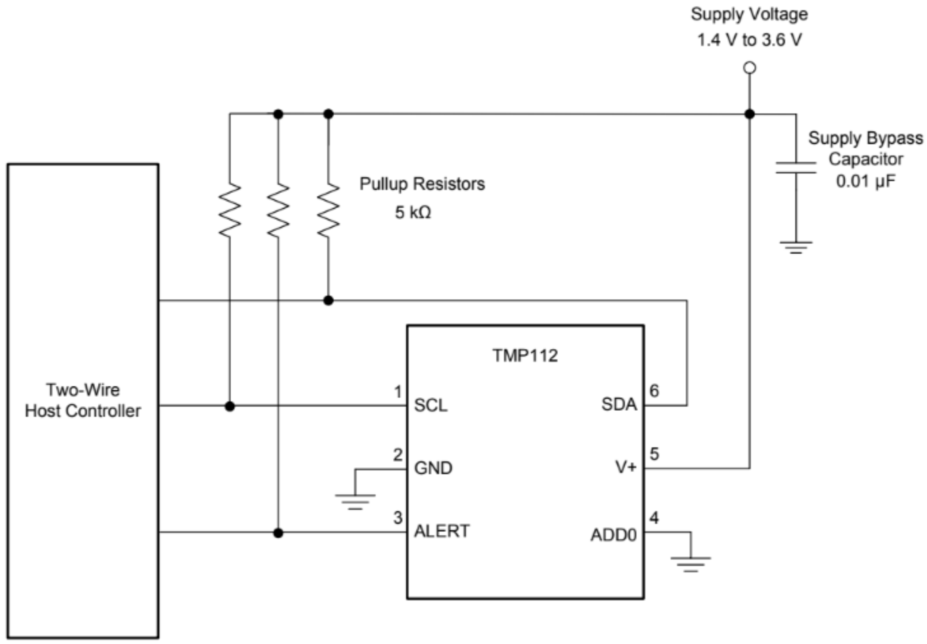


Figure VII-13: TMP-112 Sensor Reference Design

The TMP112 usually doesn't generate much noise, partly because of how little power it needs to run. However, if the power source has a high impedance, noise will be more likely to occur. Plan Bee is choosing a power source after choosing all other components that requiring power; this is to ensure a proper decision is made on how much power, voltage, and current are truly needed throughout the prototype device. Because of this, it would be safest to apply noise reduction techniques to the schematic including the TMP112. This will ensure as little noise as possible, even if the impedance from the power supply is higher than expected. One of the simplest noise reduction techniques would be an RC filter. This would include adding a resistor connecting from the supply voltage to the V+ node, while also adding a capacitor to the same V+ node. This capacitor would also be connected to ground. This noise reduction technique is shown in the schematic below. For this technique to truly be effective, the resistor value should fall below 5k, and the capacitor value should be above 10nF. These general component additions could help increase the abilities of the TMP112 sensor in Plan Bee's design. Implementing them into the PCB schematic is useful for preventative measures.

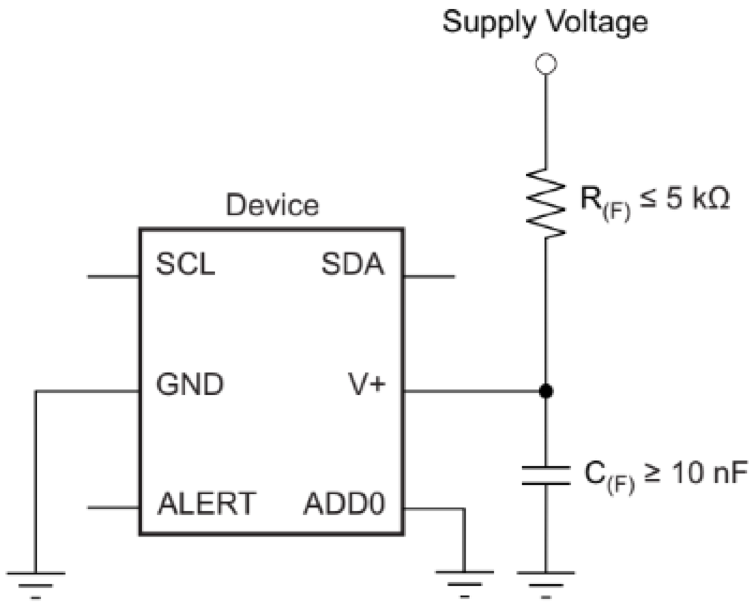


Figure VII-14: TMP-112 Sensor Intended Implementation of Reference Design

III. MPU9250

The MPU-9250 has twenty-four pins. Because the MPU can operate with I^2C and with SPI, there are different configurations that can be made for each interface. Capacitors would be placed at the same three pins regardless of which interface type is being used. REGOUT and VDD should have a .1µF capacitor that is also connected to ground; whereas VDDIO should have the same configuration, except the capacitor should be 10nF. The two .1µF capacitors would act as bypass capacitors and the 10nF capacitor would be a regulator filter capacitor. These three components are the main required exterior components for this sensor chip. There are two main differences between the I^2C and the SPI operation configurations, which are shown in the figure below. ￼ configuration is the left picture, whereas SPI is on the right. For ￼, the ninth pin is the AD0 pin, whereas for SPI it is the SDO pin. Another dissimilarity like this one occurs at pins twenty-three and twenty-four. Pin twenty-three is labeled SCL for ￼ and SCLK for SPI. For pin twenty-four, the I^2C operation configuration is SDA; SPI is SDI for pin twenty-four. The nCS pin (pin twenty-two) is connected to VDDIO under the I^2C operation, but this is not recommended for the SPI arrangement.

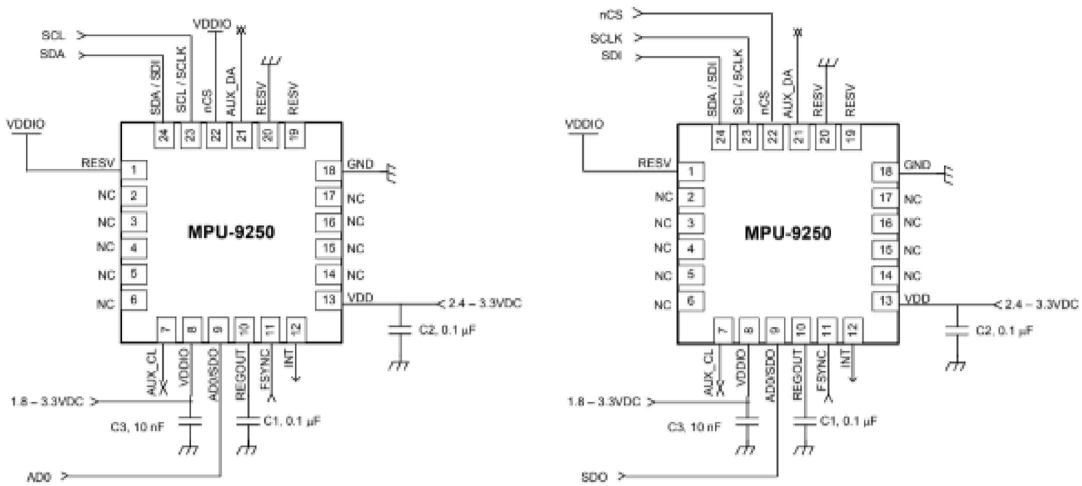


Figure VII-15: MPU-9250 Reference Design

Other sensors can be attached to the MPU-9250 chip. Common configurations are important for Plan Bee to understand so the schematic creation process for this sensor can be as streamlined as possible. The MPU-9250 would primarily run with I^2C interface for this project; the connection of another sensor to the MPU-9250 would be through the auxiliary I^2C bus. The system processor would link to the SDA and SCL pins; these connected wires would include pullup resistors as well. Pin configurations for this schematic design are important as well. Pins such as FSYNC, SDA, SCL, INT, and AD0 are all connected to VDDIO. The other sensor, that would need to be designed in a schematic on a case by case basis, would still be connected to VDDIO and MPU-9250 as well. This configuration is shown below, and although it is useful for basic schematic designing, it will need to be altered to fit the specific needs of this project.

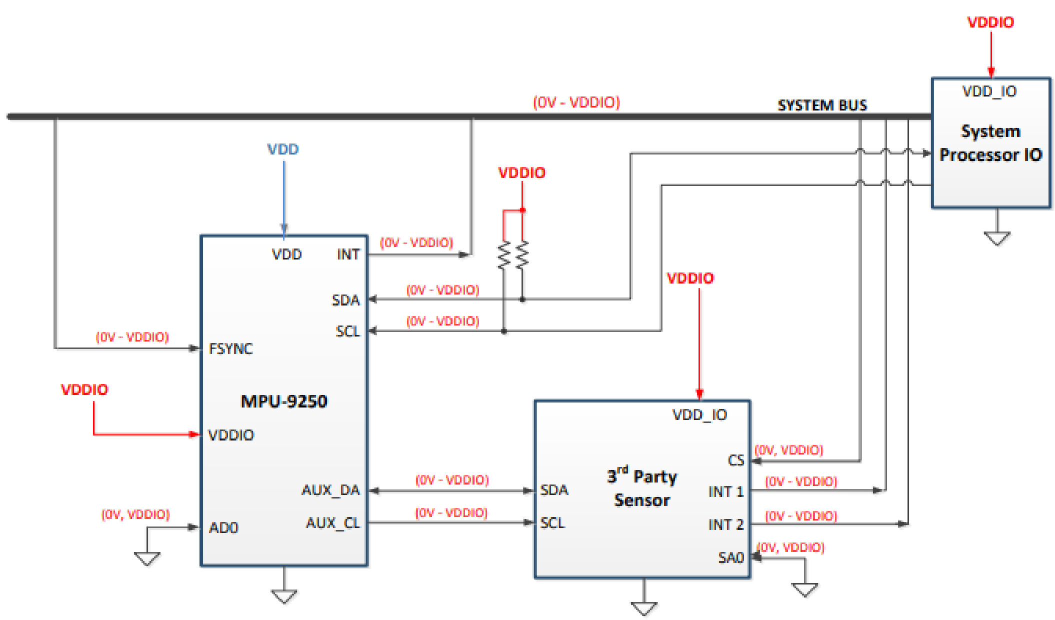


Figure VII-16: MPU-9250 Intended Implementation of Reference Design

IV. LPS22HB

The LPS22HB has ten pins, much more manageable than the MPU-9250's twenty-four. Because this sensor is less complex, the schematic design surrounding this chip will be simpler as well. This chip can also use two different interfaces, I^2C and SPI. Pin two is dual configured for either SCL (for I^2C) or SPC (for SPI). Pin four has an analogous situation, with I^2C configuration represented by SDA and SPI by SDI. The same type of I^2C/SPI pin differences apply to pin five as well, changing from SD0 to SAO, respectively. VDD is pin ten and has a 100nF capacitor attached as a bypass. This capacitor is recommended to be as close as possible to the supply pads of the device. This basic configuration is the start to designing this part of the PCB schematic. The figure below showcases this design.

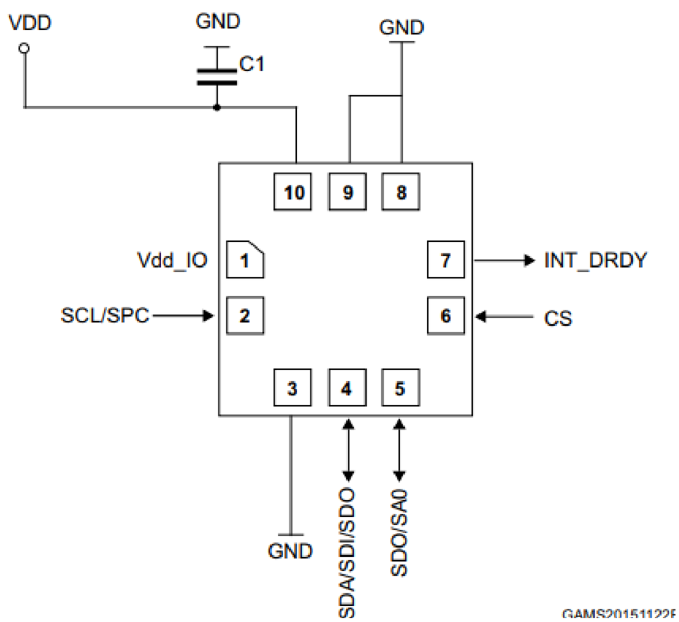


Figure VII-17: LPS22HB Sensor Reference Design

V. OPT3001

Much like the TMP112, The OPT3001's readings are also greatly affected by its surroundings on the board. Things that are especially reflective and can affect light, especially light that can be seen by the human eye, will greatly affect the measurements this sensor helps record. Taking special care to place surrounding components that fall into this category either above the sensor or away from the sensor is of upmost importance. This is a main concern when considering PCB layouts for this project.

Another recommendation is to include a bypass capacitor from the power supply going to this chip. The capacitor should be placed near the OPT3001. Making sure any light reflected from the capacitor does not reach the OPT3001 is very important. Having the capacitor raised to double the height of the OPT3001 would help this but may be difficult to accomplish. It is also recommended to connect the thermal pad of the OPT3001 to ground electronically through the board. A PCB trace would be the easiest way to do this and could be implemented in Altium. A PCB layout of this is shown below.

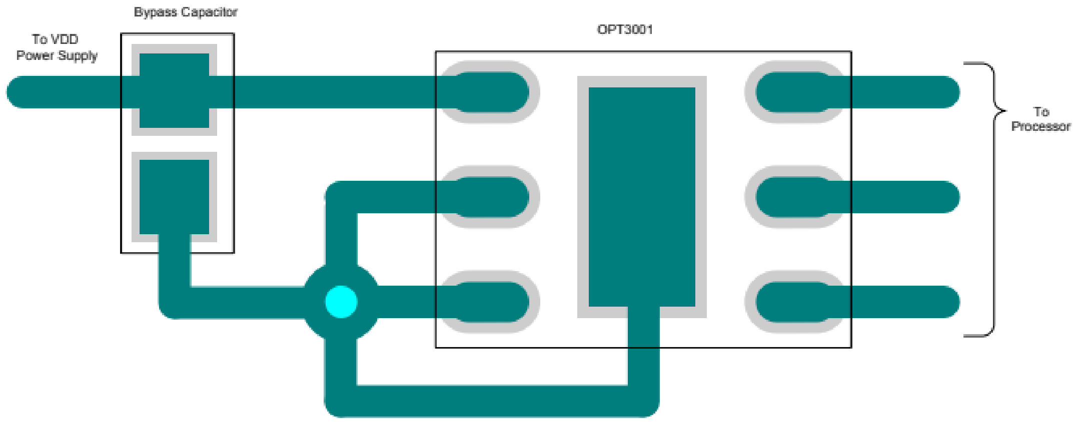


Figure VII-18: OPT3001 Sensor Reference Design

Because Plan Bee will most likely have the PCB board enclosed in protective casing, it is helpful to get an understanding on how to help make this chip most effective through casing and behind a dark window. There are examples of this schematic design, which include adding resistors between the VDD node and the SCL, SDA, and INT pins. This example is illustrated below. A capacitor is also applied at VDD and grounded. This setup would provide well for diverse conditions but would need to be tested thoroughly to ensure the OPT3001 is still providing useful and accurate information.

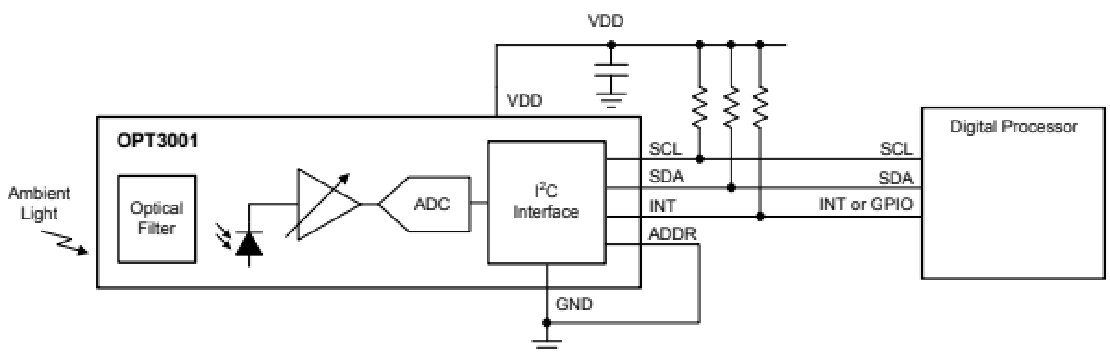


Figure VII-19 HDC2010 Reference Implementation

VI. HDC2010

To enhance thermal isolation, a slot should be made in the PCB for the HDC2010 to be laid. To maintain this thermal isolation, the only component that should be near the HDC2010 should be a bypass capacitor that is along the same wire as the voltage supply source. This bypass capacitor between VDD and ground is recommended to be multilayer and ceramic to prevent any heat from affecting the nearby sensor. The HDC2010 should be further away from any other components, preventing the heat they generate from affecting the accuracy of this chip. The slot in the PCB will increase the accuracy as much as possible and prevent the sensor from any slight environmental wear and tear or heat exposure that may occur. This setup is shown in the PCB schematic below.

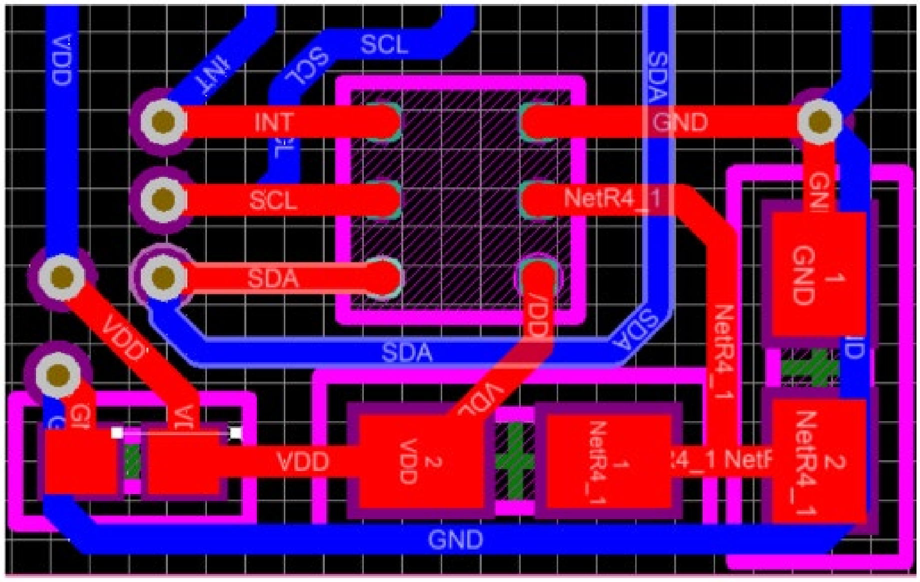


Figure VII-20: HDC2010 Intended Implementation of Reference Design

II. Software

I. Sensor Data Model Diagrams

Persistent data, such as sensor information, will be stored on the non-volatile storage of the board. This data will be obtained using the sensor controller of the main micro-controller. When stored, the data will be accompanied by additional data to manage transition from

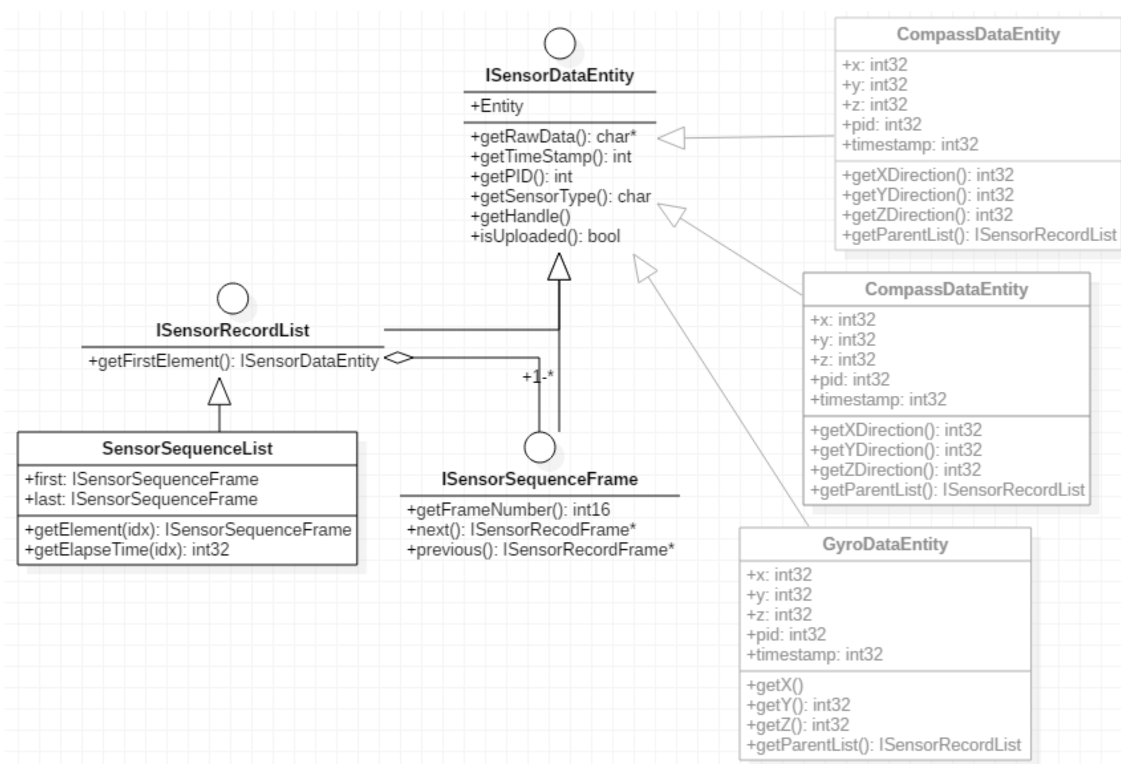


Figure VII-21 Data Interface

tracking device to an access point. While the data tracking device will be capable of storing information from various types of sensors, the underlying data-structure representing such information will all feature a common interface. This interface is called ISensorDataEntity. The ISensorDataEntity is an abstract class which declares 6 crucial methods in addressing raw sensory data. These methods are to be implemented by inherited classes wishing to integrate with the entire data transmission and storage process.
The declare methods are:

- ***getRawData***: This method returns a pointer to the raw data of a sensor data element. When implemented, the returned constant pointer should point to data stored in local memory of the MCU. It may also be noted that data be stored in non-volatile memory and cached to memory when needed.
- ***getPID***: This method takes zero argument and returns a 16-bit integer identification of the representing the indexed sensor data structure in memory.
- ***getSensoryType***: This takes zero arguments and returns a 8-bit integer type identifier. The type identifier represents what sensor type the stored data relates to. The sensor type descriptor, stored in non-volatile memory, will be associated with a single Sensor Type Identifier.
- ***getHandle***: This method takes zero arguments and return a 16-bit integer value representing the index of the value stored in memory. Such value is differentiated from a PID in that is a database value representing the specific element, while the handle could be synonymous with the location of the data element in memory.
- ***isDispatched***: This method takes zero arguments and returns whether the data element has been successfully dispatched.

Sensory data of the same sensor type may also be temporally linked. The interfaces, ISequenceList and ISensorFrame will be used to assert a temporal relation between data elements. ISequenceList will inherit from ISensorRecordList. This class will implement a generic linked list implementation. As a result, the elements forming an ISequencList will incorporate temporally related ISensorDataEntities. This sequence element was further differentiated from non -framed SensorDataElement by defining the ISensorFrame class. This class defines an element which encompasses an frame in an recording period. Such a recording element may be created when the sensor controller is prompted to record a series of sensor data to differentiate from the regular period measurements.

ISensorFrame is a circular linked list element, and, as such, implements a similar interface:

- **getFrameNumber**: Returns the index of the data point in the sequence.
- **next**: Returns the next element in the sequence or the first element in the sequence if all elements have been accounted for.
- **previous**: Points to the previous element in the sequence, if such element does not exist point to the las element int sequence.

The SensorSequenceList object, on the other hand, encompasses the circular linked list element. It provides access to a set of sequenced data point that is to be grouped accordingly. Such element may also inherits ISensorDataEntity, the root data element object. This will allow SensorSequenceList to be manipulated as a single element. SensorSequenceList element implements the following methods:

- **getElement**: This method returns the ISensorSequenceFrame at the specified location.
- **getElapseTime**: In manipulating the timestamp recording of each individual frame, this method returns the elapse time of the specified frame in the sequence.

To transmit a sequence to a webserver, each frame should transmitted first before a sequence list is transmitted. The sequence will then be processed on the server to gain further incites and stored in an online database.

Sensor Data Database Model

The figure below illustrates the database design for storing sensor data. The model consists of 4 major tables which serves as the root class for all sensor data types. In addition to the root

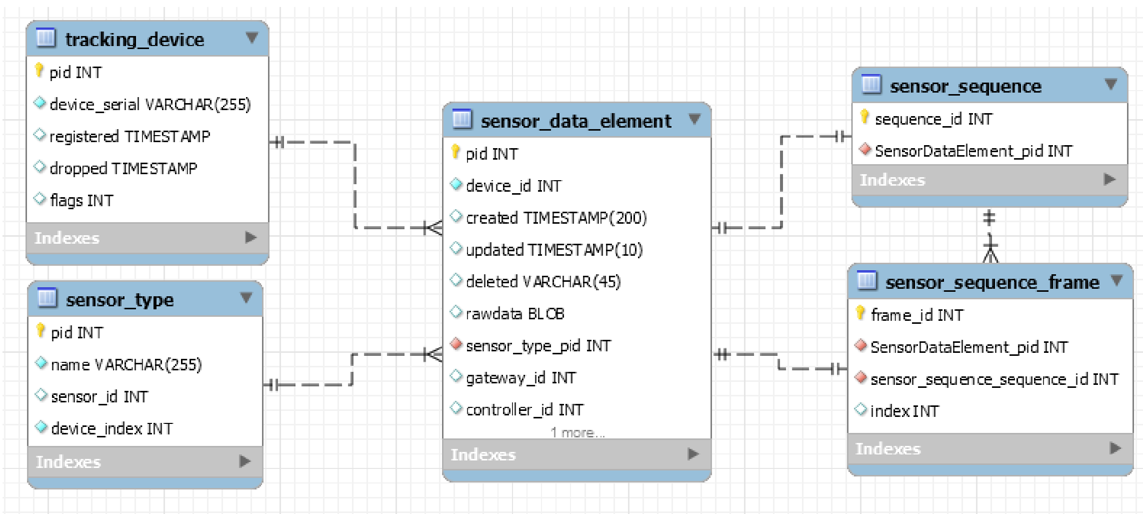


Figure VII-22: Core Database model for sensor data

The major tables are:

- **Sensor_data_element**: This is the base element where all other sensor data is derived. It consists of a primary key identifier, pid, which is resolved when the data is stored in the database. A device identifier column is also used to track which device said sensor data element is derived from. Additionally, a gateway identifier, gateway_id, is used to track from which access point the data had originated. The create, update, and delete timestamps are derived when the data aggregator stores the data to a local database.

  The controller_id column tracks the initial identification number stored on the tracking device. Because the tracking device id consist of a 16-bit value, the controller_id will only be used to refer to the data origination but will not be used to identify the sensor_data_element specifically. Each element in the sensor data table will also reference a sensor type row of the sensor type table. This value specifies what sensor type is correlated with the data.
- **sensor_type**: This table stores identifiers for the various sensor types registered with the data logger. Each sensor type declares a primary key to identify its location, in addition to a 32-bit ASCII name of the sensor. The sensor_type table

includes a sensor_id which will be define during sensor registration. The sensor registration process encompasses a controller providing a name and identifier for the sensor. Additionally, if there are numerous sensor of the same identifier, a device_index will be incremented to differentiate the different sensors attached to the data logger.

- **tracking_device**: This table stores the registered data logging devices. It consists of an database identifier to track device registration to a table. A timestamp for when a device has been registered and when the device has been dropped. The table includes a flag parameter which presents the state of the data tracker. Possible state include battery alerts, connection issues, or internal device alerts.
- **sensor_sequence:** sensor_sequence table keep track of sensor data elements sharing a temporal relationship. This will allow time dependent values to be collected and grouped, through there various elements are inherited from sensor data elements.

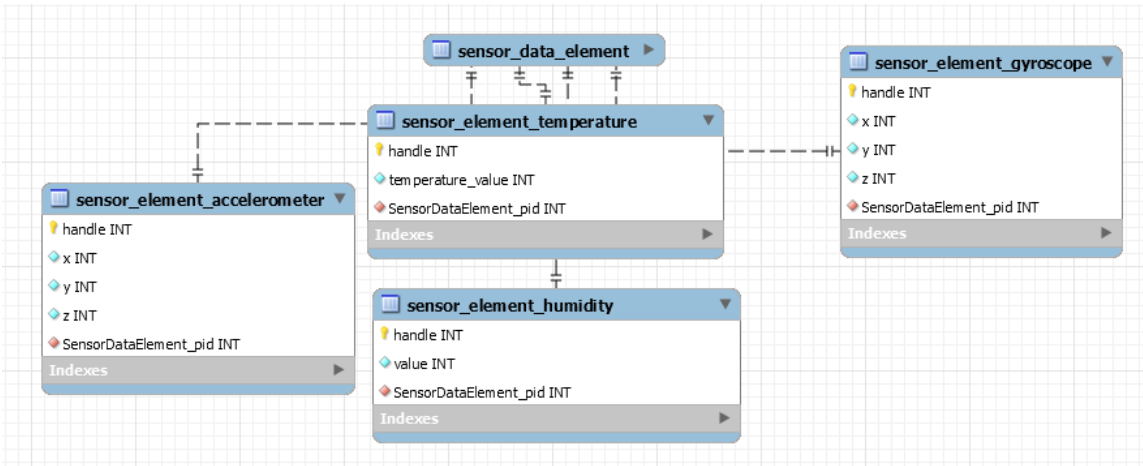


Figure VII-23 Concrete Sensor Data Elements

The above diagram illustrates the various sensor elements that could be inherited from the sensor data element. These tables store additional data parameters which are extracted from the raw data of the sensor data element. This data is stored external to easy database queries and provide direct access to sensor data. Inherited tables should contain indexes to the sensor data element table. In addition, inherited tables should also contain a handle which identifies the data element separate from the sensor data element. This abstracts the concrete sensor elements from the base class which provides ensure future expandability and refactoring.

III. Project Testing

I. General PCB Testing

In order for us to know that our breadboard has been built right and it is fully operational, we need to discuss the regular standards for breadboard testing and where we hope to get this done. For a PCB to be tested properly the main quality control is done as two separate processes: a fabrication test and an assembly test. Though these tests are generally easy to do on our own, for the sake of time, quality test precision, and available testing tools, we will be asking a local company that makes custom PCB's to also run the quality control tests for us.

For both fabrication and assembly testing, something that has to be considered is the proper placement of test pads as these have to be specifies for to whichever company will provide out PCB building and testing service. Test pads can normally be placed to be probed from the bottom, the top, or both sides. For the sake of price and ease of testing it is generally advised to only place test pads on one side, and Plan Bee will only need to our test pads on the bottom; having them at the bottom is also the most ideal factor for a bare-board test. Another consideration is which via points will be tented and which points won't be. Tenting a via means to close up any opening in a net with a soldering method. There are both pros and cons to tenting a via: pros being related to closing components/layers on top of each other so that no short circuits are experienced, while cons include leaving extra hole space that can be filled with corrosive materials when trying to fill in PCB's with multiple layers. It can also be beneficial to leave a via uncovered for first-time tests of prototypes as they leave behind easy access and these vias can be protected with the last-minute gold platin used at the end of the surface finish.

The first general examination to be done on the PCB for quality control is the fabrication test. This test is also known as bare-board testing because this testing is done only when the electricity node trails (also known as nets) are present and the board has not been populated with its components. For this part of the testing you generally do not need to put specific test pads on the nets unless you are dealing with fine pitch devices. This test can be done by a process using what is known as a flying probe device and it is as simple as having two independent probes come in contact with two separate nets, one probe sends a signal, while the other probe detects the presence (or absence) of that signal. Two main sub-tests that are done during this process are also open capacitance tests, and short resistance tests. An open capacitance test simply takes the probe that is receiving the signal and measures its capacitance (or lack of) to test for any open circuits, while the short resistance test measures the resistance of the receiving signal through a conducting trace to make sure there are no short circuits. Some limitations to this test include the spacing between the two nets being analyzed as the space between them must be big enough to fit the heads of the two automated probes. Fabrication testing can also be much slower than assembly testing though it is a simpler process.

Once the board is populated with all required components listed in the Bill of Materials, we can then undergo an assembly test. This testing is almost always (but not limited to) running a bed-of-nails test. Bed-of-nails test requires for all nets to have at least one point of connection being probed for each net simultaneously while having a specified reference net. Once each net is probed at the same time, a single signal is sent to each individual net while all the other nets are analyzed to see if they are receiving the signal through the probes. This process is done for every net one at a time. This sort of testing also requires for specific test pads at each net, and these might not be fully accessible all the time. For such issues, it is normally suitable to have specific spacing between test probes and test pads as this sort of testing requires a set grid of probes that can be in the way when it comes to big components.

For our testing, we have decided to go with the services of a local company called Quality Manufacturing Services Inc., located in Lake Mary, Orlando, Florida. This company provides many services ranging from full initial proof of concept incorporation and full PCB assembly, to quality control and box build assembly, to personal delivery. Their professional team partners up with our design team to satisfy all of our component needs including vias and testing pad placements, and then suggest an appropriate testing procedure based on these as well as any needed trouble shooting should there be any faults in design.

II. Breadboard Testing

This section reviews the application and methods used to test the various parameters encompassing the project. Initially, Texas Instruments CC1100 was used to testing low powered wide area network range. This was done using a custom application built by a company called Anaren. This application can send temperature data across 2 MSP430 using a Texas Instrument CC1100 wireless transceiver. While not as powerful as Texas Instrument's CC1352R used as the main MCU of this project, the CC1100 does feature TI's long-range communication stack.

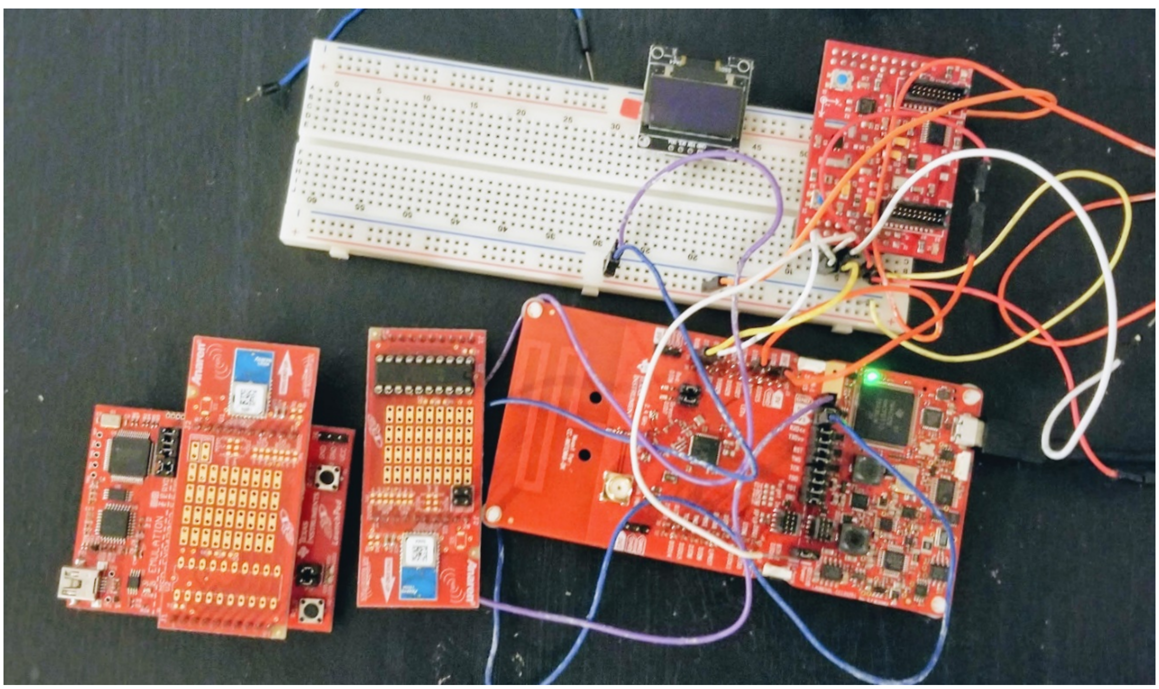


Figure VII-24 Breadboard testing

As such, the distance of communication has been touted to be capable of reaching 10 kilometers. The CC1100 development boards were flash with Anarens default application and asserted to be able to wirelessly communicate at short distances. One of the boards was then moved until communication halted. A range of approximately 50 ft was also recorded. Such an experiment leads us to further inspect the possibilities of choosing a alternate long range communication device. We happily sided with Texas Instruments CC1352R due to the additional gain promoted. Two development boards were purchased, and one attached to a breadboard in order to test Texas Instruments' sensor hub booster pack. This booster pack houses numerous sensor devices and is easily interfaced with the CC1352R development board using the I2C port. The following section goes into detail on the testing methodologies.

Sensors utilizing I^2C were purposefully chosen primarily for the ease of use and expandability. I^2C provides a robust method of communicating with many devices and allows additional devices to be incorporated in a preexisting design. Such an architecture allows the primary microcontroller to communicate with as much as 128 different devices. The Sensor Hub Booster Pack, designed by Texas Instruments, was used to test I^2C communication and sensory data acquisition. The Sensor Hub features an MPU9150 by Invensense; this device integrates a 3-axis gyro, 3-axis accelerometer, 3-axis compass and a temperature sensor in to a single device.

Additionally, the Sensor Hub Booster Pack Features Bosh Sensortec BMP180 pressure sensor, Sensirion SHT21 humidity sensor and ambient temperature sensor, an Intersil IS2903 ambient and infra-red-light sensor and Texas Instruments TMP006 non-contact infrared temperature sensor. Because this board was developed for Texas Instruments Tiva-C model processers, the headers were not pin compatible with the CC1352R development board. As a result, the device was modified, and its signals made available by attaching it to a breadboard. The necessary routings were then made between the CC1352 development board and the sensor hub.

To test communication and sensor data Texas Instruments Sensor Controller Studio was used. This application compiles code specifically designed for sensor devices, for the ARM thumb auxiliary micro controller integrated in the CC1352R. The sensor controller features a low powered ARM module fitting for communication polling transducers and other types of sensory devices. Communication is handled using semaphores between the main controller and sensor controller. This reduces power consumption and allocates computer time to the main controller. Code was then created to drive the listed sensors and the data was collected and reviewed. To aid with serial communication and logic analyzer was used to scan the I^2C bus. Sniffing the bus allowed us to assert the accuracy of the data being communicated between devices. Below we provide the result of each sensor that had been tested.

Testing Invensense MPU9150

The Invensense MPU9150 documentation provided the needed information necessary to initialize the sensor and read its data. The MPU9150 follows a similar implementation pattern as our final device, MPU9250. As a result, testing such a device will follow a similar vein as the final motion sensor. The initializing process for Invensense MPU9150 relies on power cycling the chip and waking it from sleep. From then the chip goes through a self-diagnosis stage were by it reads factory data and tries to calibrate its various sensors against a control system. The MPU9150 also features an interrupt line which can interrupt the master controller on specified programmed events. This feature, in addition to polling the system at 200 hertz was enabled. This code used to initiate and drive this device was developed using Sensor Controller Studio and will be integrated into the final project.

The sensor data of the MPU9150 is stored in multiple internal registers. Each sensor was able to provide a 16 bits value which was stored in 2 consecutive 8-bit addresses. To access the address the MPU9150's I^2C address, 0x68, was first placed on the bus, followed by a series of instructions and a final read with acknowledge value. The diagram below presents the I^2C bus as data was transmitted. As one could see, the main address was first placed on the bus, followed by additional the register to read and a read with acknowledge bit for each register value.

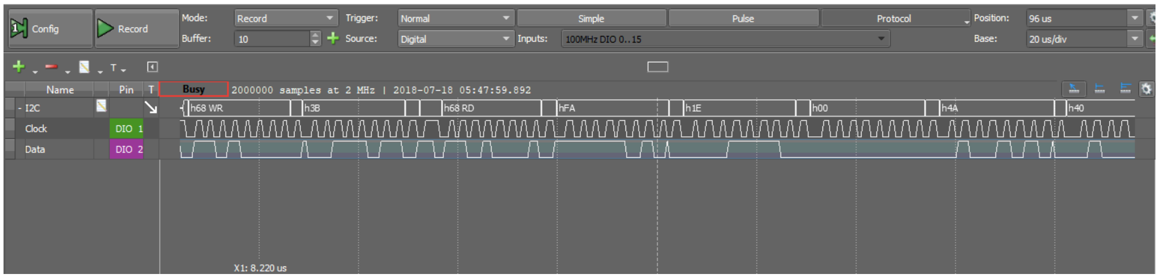


Figure VII-25 I2I2C Accelerometer Stream recorded using logic analyzer

In addition to the communication testing, the data transmitted across the I^2C bus was also tested and graphed against time. Force was placed on the accelerometer and the changing acceleration was graphed accordingly; this is presented in the figure 11, below.

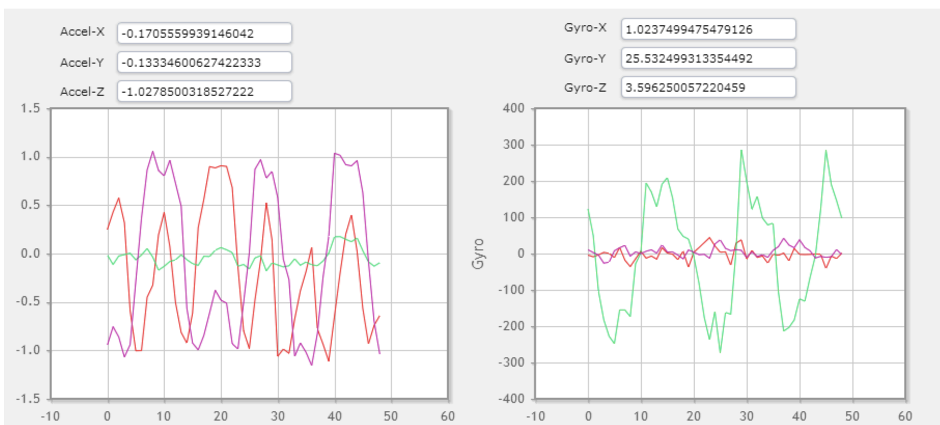


Figure VII-26 Accelerometer and Gyroscope Testing

Similarly, the gyroscope measured the centripetal speed of the sensor hub. The data gathered form the gyroscope against a period was is shown in figure 12. As the development board was moved the graph adjusted accordingly. The compass of the MPU9150 has been relegated to a secondary I^2C device. As such it utilizes a separate address from that of the main MPU9150. As a result, separate code was made to manage the compass. Like the MPU9150, data was read by applying a device address to the I^2C bus followed by a starting register address. A read with acknowledge instruction was the pass to the bus to read the necessary data. The figure above showcases compass data read form the MPU9150. As the development device was altered the compass also changed. While such data needed to be calibrated, the values and changes in values are significant testing blocks for the final application.

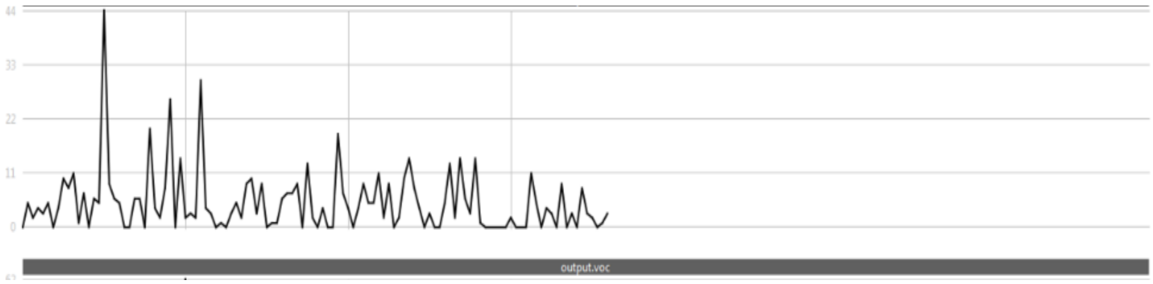


Figure VII-27 Compass Reading

Testing Sensirion SHT21

Sensirion's SHT21 integrates a capacitive humidity and temperature sensor. The humidity sensor can decipher humidity reading with a resolution of up to 14-bits. This chip is integrated with TI's sensor-hub booster module. As a result, to test this sensor the sensor-hub I2C connector was attached to the CC1352R development board. Like the accelerometer, Texas Instruments' sensor controller studio was used to query the data from the chip. The chip was first initialized as described in the datasheets. A logic analyzer was used to validate the initialization process. After initialization, the chip was then queried for humidity data. The SHT21 transmits data across 2 bytes packages; with the MSB transferred first. Each byte is followed by an acknowledgement. The transferred 16 bits of data comprises of a 14-bit value which represents the final raw data and a 2-bit status value. The raw data was queried over time and the data graphed using Texas Instruments' sensor controller studio. The Image below showcases the captured humidity sensor data plotted against time. As shown the data remained a constant value of 24524 as the humidity in the room remained constant.

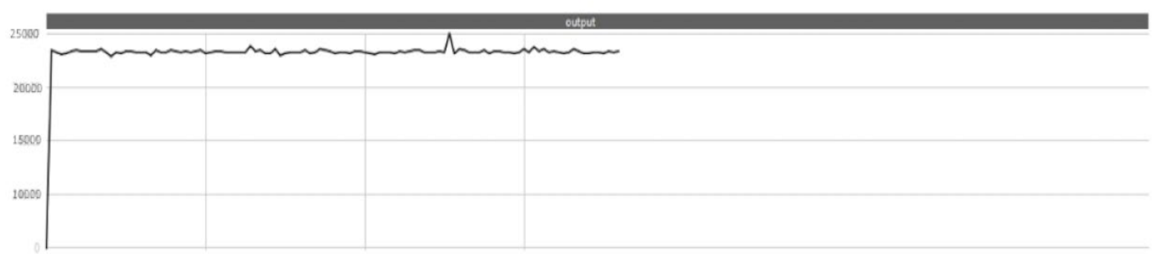


Figure VII-28 Humidity Sensor Reading

The relative humidity value could be calculated using the formula specified in the datasheet. That is;

$$Relative\ humdity = -6 + 125 * \frac{RawValue}{2^{16}}$$

Given the value of 24524, the relative humidity could be calculated as 40.775. This is in line with the humidity of the room, at 40% humidity. As a result, we concluded that the device does work, and the testing process was a success.

III. RF Testing

Long Range Communication Testing

To rapidly prototype wireless communication, Texas Instruments' SmartRF software tools was used to configure the RF transceiver to communication across various parameters and transmit an arbitrary set of data across differing ranges. SmartRF allows rapid application testing and can provide additional features such as wireless sniffing and wireless integration techniques using various wireless standards such as Bluetooth and Zigbee. To begin, a CC1352R development board was attached to a computer using USB. The devices were then activated in the SmartRF software and its features queried for transferring data. SmartRF offers a set of configuration options which presents an assorted amount of communication method to be used for communication. To test the long-range communication, the SimpleLink long range communication configuration was used.

Figure VII-29 Communication Range 523 meters

A variable length data packed comprised of randomly generated data was then created. The packet consisted of an initial 16-bit preamble followed by a 32-bit sync word. This is then followed by an 8-bit length value, then an 8-bit address value and finally the variable length data. Another CC1352R development board was attached to a second computer and the receive option was selected in SmartRF studio. This selection option was matched with that of the transferring board and the reception was initiated. The reception device presented the packet data as generated by the transferring device.

To test overall reception distance, the transfer method was set to continuous and the device on the receiving end was relocated. Given the transferring device was elevated 12 feet above ground, an estimated communication distance of about 1500 feet was achieved.

While this distance was significantly less than what is proposed, we do believe the distance could be improved by installing a better antenna and utilizing more power. In addition, the test took place in a moving vehicle within a rural environment. Both of which would have an adverse effect on the final operating distance. Further testing will be done to achieve a greater distance. Texas Instrument has also promoted their CC1352P; this chip features a higher power wireless transceiver and is also pin compatible with currently selected CC1352R. While the chip is not yet available; when it is, I do believe migration will be drastically simplified because of application abstraction made possible with Texas Instruments' RTOS.

VIII. Integration

This section includes overall integration, PCB design, and system testing information. This includes both software and hardware integration. Firmware and drivers integrated under the same "roof" of software are considered overall software integration, and will be shown and explained in detail, as well as referenced to the Appendix with relevant code. The Major PCB Design, including the Power, RF and Communications, and sensors cluster inputs will be shown and explained in detail. Testing pads will be available throughout the finalized PCB, and system testing will be done when this our PCB has completed fabrication.

I. Overall Schematic & PCB Testing Plan

The overall schematic is presented below as a logical block diagram of sorts, because we could not present all components integrated on a single schematic. It would have been far too complicated. We have split the schematic design into manageable schematic sheets that we can work on independently and integrate references and nets later. Each of the blocks presented its own schematic sheet with components and possible further blocks with sub-sheets.

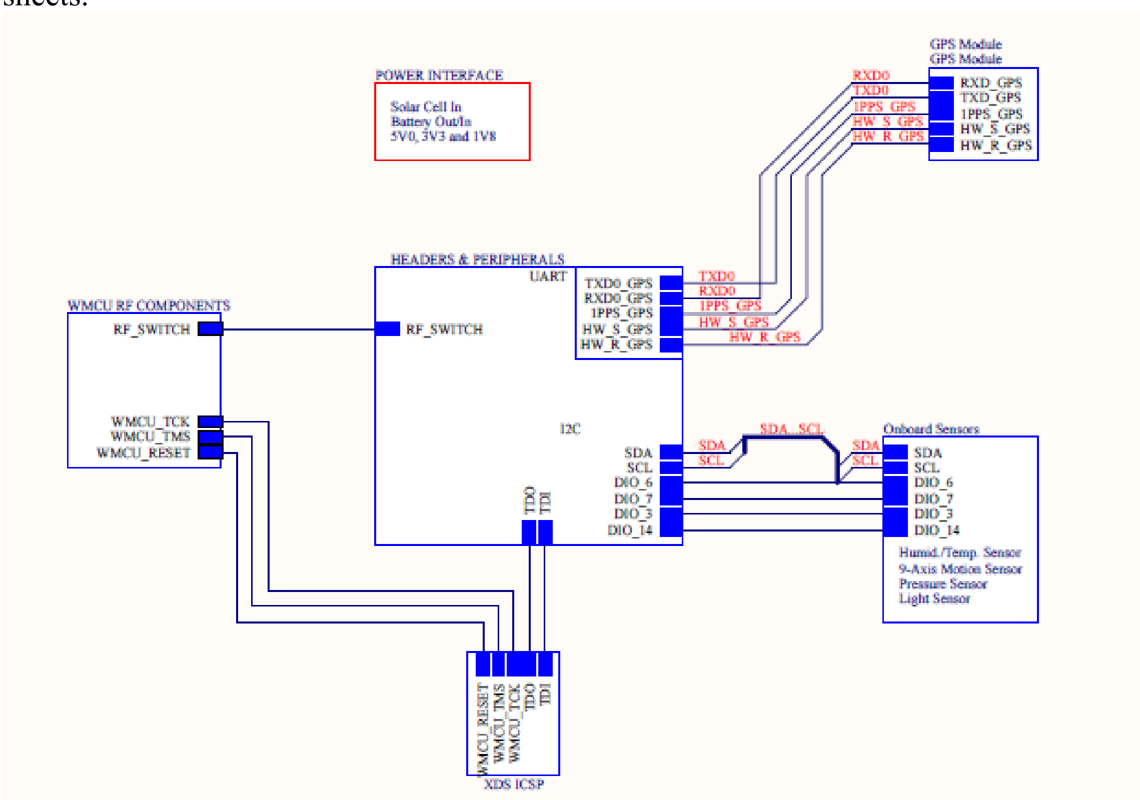


Figure VIII-1: Top Sheet for Overall Schematic

With the WMCU RF Component, we will be forced to use a four-layer-or-more PCB design, and with the communications buses, we will likely choose 8- or 10-layer when we are prepared for full fabrication. With this, our second-from-top layer of the PCB will be our power plane, and our second-from-bottom PCB layer will be ground plane for design convenience. As well, further heat dissipation can occur toward the bottom plane where components are not, helping us manage power output and heat. Separate layers will be accessed using copper infill vias. Power-specific planes will use 14 mils via diameter, RF components will use 12 mils vias as needed to access communications lines, and all other vias will be 8 mils in diameter.

Because of the way we are accessing power, we have not connected the power block. Instead, it is referenced throughout designs by a cycle-style power port component, which represents a via to this power layer. Grounds are indicated by the standard ground symbol, which will effectively be accomplished by vias also. More effective routing techniques may be used to reduce the number of infill vias, and the cost of the fabricated PCB within this next stage.

The effective blocks here that we are including in our major PCB design include the WMCU RF Components – which have been split off and will be on its entirely own domain on the final PCB – and the Headers and Peripherals Interface, the User Interface, the Programming Interface (XDS ICSP), the Power Interface, the on-board GPS module which will be included, several on-board sensors that we intend to include, and an interface for connecting to outside sensors for UART.

The Wireless Microcontroller Unit Radio-Frequency Communications (using ISM bands of 968MHz and 2.4 GHz) Components (WMCU RF Components) sub-sheet will include the first part of the CC1352R1 that is responsible for RF communications (U1A) and a balun (balance) for integrating differential (negative and positive terminals) input-and-output RF ports for the two ISM bands, as well as many filtering components and the RF Switch we have chosen. All RF communications necessities will be accomplished through this one schematic and balun sub-schematic (for balun) shown in the next section.

The Headers and Peripherals will include flash memory interfacing to the CC1352R1 embedded ARM Cortex M4 Microcontroller Unit. It will also include interfacing from the RF portion of the schematic to all other parts of the overall schematic, for sensors, for user interfacing, for power, and for all other interfacing requirements. This is the main interface, and the “heart” of our entire senior project design, so it will be discussed extensively. In the third section herein.

Interfaced with the Headers and Peripherals is the programming interface, called the XDS ICSP, meaning XDS In-Circuit Serial Programming, using X-window Direct Save protocols (XDS). The importance of this schematic cannot be overstated, because it will be necessary to program our required functionality into the MCU portion of the CC1352R1, which is why it is interfaced with the MCU and flash memory part of the Headers and Peripherals schematic. More on the internals of this schematic will be discussed in the second section herein.

The GPS module, the next most-important part of the overall schematic and our final end-product is featured at the top right. This can be accomplished using the theory of hardware-defined radio (as discussed within the Research and Parts Selection chapter) or through using an integrated module that communicates via UART RS485 protocols (using lower power than the external sensors which communicate via UART RS232 protocols). The CC1352R1 contains two UART modules internally, both multistrand and capable of RS232 or RS485, so these will be mapped accordingly within the Headers and Peripherals schematic sheet. More on this module will be discussed in the fourth section herein.

The Onboard Sensors schematic sheet is connected the Headers and Peripherals using the I2C protocol discussed within the Research and Parts Selection chapter. With this protocol, we can connect up to 127 seven-bit communications lines components using just two wires. These two wires are contained within the displayed bus, to each of the internal sensors (temperature sensor, three motion sensors, humidity sensor, pressure sensor and photodetector), each will take their own schematic and be discussed in the fifth section herein.

Similar to the GPS module, the Offboard Sensors Interface schematic sheet will be connected to the Headers and Peripherals schematic sheet using the UART RS232 protocols, which use greater power. Because UART is serial and not parallel like SPI or I2C, we will need to include an interface (hence the name) for the external sensors. This interface will be a differential (or dual) multiplexer for the two communications lines required for UART RS232 Protocol – the Rx (Receive) and Tx (Transmit) data lines. Select line will be needed internally, and we may need to include another microprocessor within the interface or may just route other General-Purpose Input-Output (GPIO) bidirectional pins to the dual multiplexer-demultiplexer control lines directly from the CC1352R1. These will be used to select the sensor data, switching between serial line and obeying UART communications and interfacing requirements. This will be further discussed in the sixth section herein.

The User Interface sub-schematic sheet, including two buttons, two switches and three onboard LEDs, as well as an OLED Display will be mapped to the Headers and Peripherals sub-schematic sheet using bidirectional GPIO pins available on the CC1352R1 for the ARM4 MCU utilize for bidirectional components (buttons, switches, and LEDs). And the OLED Display will be routed to the MCU using I2C again, which was discussed above. This will be further discussed in the last section herein.

I. WMCU RF Components

The main CC1352 chip has varying line thickness connections on the board, depending on where the lines are connected to from this chip. The lines from this chip to the crystals and to the Balun and Impedance Matching Interface are all 12-mil thick. These 50-ohm 12-mil thick lines are to those specs because it allows maximum power transfer from the power source to these RF systems. This is ideal because it allows for maximum power transfer from the RF antenna, increasing efficiency throughout the RF components of this end of the PCB, as well as RF transmission sensitivity. Thus, getting the routing as close to 50 Ohms as possible will increase our maximum communications range noticeably. The capacitors throughout the schematic are filtering capacitors that essentially “smoothen” the voltage power supplied to the CC1352R1 WMCU IC from the regulator and reduce unwanted frequencies from influencing the WMCU interaction with the RF side of the PCB.

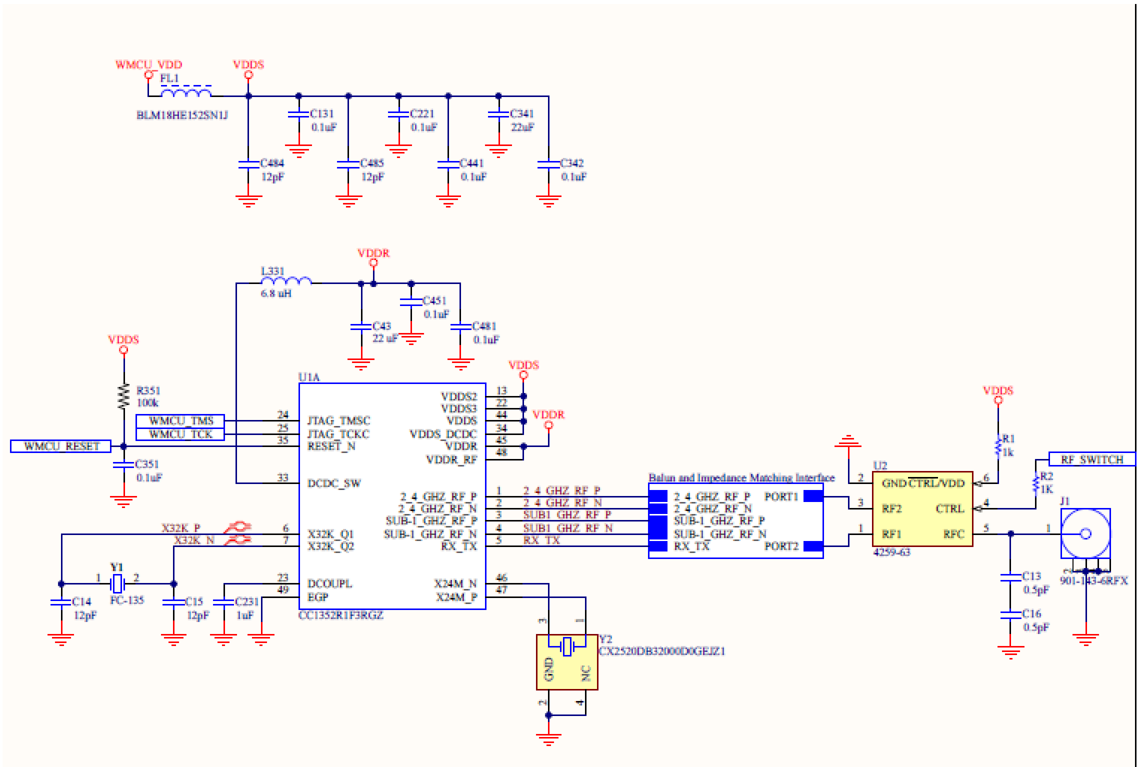


Figure VIII-2: Wireless MCU RF Components Schematic

The 32.768 kHz piezo-crystal shown on the left is used as the CLK for the internal ARM4 processor inside of the CC1352R1 WMCU. The 48MHz Crystal (much larger in packaging size) is shown in the righthand similarly and is differential. The signal from this piezo crystal is split between the X24M_N and X24M_P pins and turned into a 24MHz signal. This is used to drive the RF components of the WMCU and push the radio frequencies required.

The RF_SWITCH pin is used by another schematic referenced within this document, showing the MCU part of the WMCU in the CC1352R, in order to control the RF frequency chosen by the PE4259 RF Switch. Similarly, the WMCU_RESET, WMCU_TMS and WMCU_TCK pins are used in the XDS interface. The WMCU_RESET pin resets the WMCU every time a new program is finished uploading to the ARM4 processor, resetting the function of the chip to the new settings declared in firmware. The WMCU_TMS and WMCU_TCK are specifically used in pushing and pulling data to the WMCU during communications with the Integrated-Development Environment (IDE), Code Composer Studio, which will be used to program the WMCU.

The Balun and Impedance Matching portion of this schematic is used to integrate the differential signals (i.e. 2_4_GHZ_RF_N and 2_4_GHZ_RF_P represent the positive and negative components of a single 2.4 GHz transmission signal) and balance them into a single routed signal to the RF switch, for both 2.4 GHz and 986 MHz (Sub-1GHz) separately.

The antenna we are using to connect to the SMA port shown in the schematic on the right-right, is the Spark Fun Electronics 50-Ohm Female-SMA 2.4 GHz Duck Antenna, which matches the impedance of all RF components, maximizing power transfer.

Balun and Impedance Matching Interface

Similarly, all routing done here will be 12 mils in width in order to create 50-Ohm transmission lines, maximizing power transfer among RF components of the WMCU. This schematic is shown as a block on the previous sheet, and as states is used to integrate, balance, and impedance-match between the WMCU and RF-switching components, along with the SMA 50-Ohm antenna.

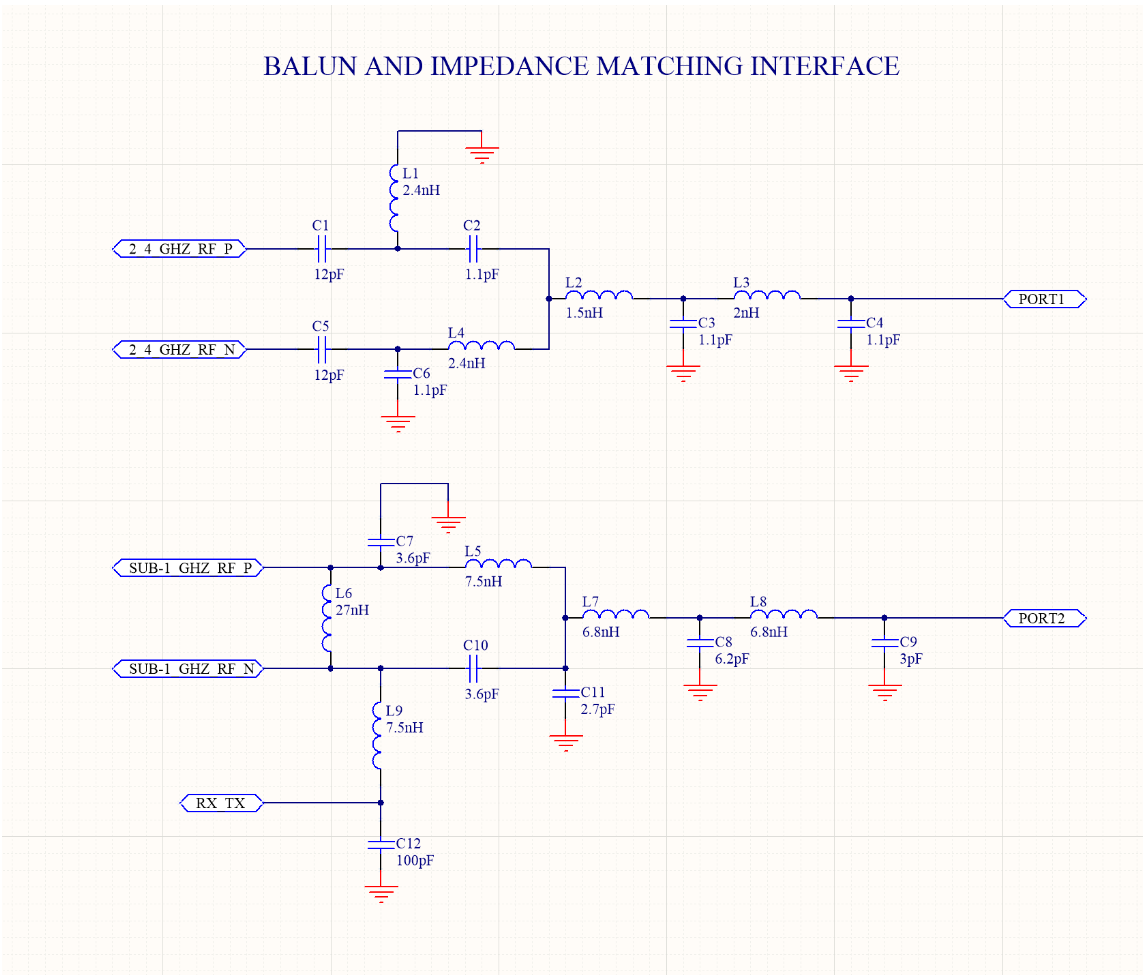


Figure VIII-3: Balun and Impedance Matching for RF Signals Schematic

This is similar to the reference design for the CC1352R1 given by Texas Instruments. However, we have chosen a different RF Switch, the PE4259 since the original reference-designed switch was unavailable. Note how there is no DC-blocking capacitor in line with the PORT2 output. This is because the PE4259 includes an internal DC-blocking capacitor, so the reference design was changed here. We also changed the location of capacitor C11 in order to simplify the design, without changing the frequency-filtering functionality. The inductors thereafter (single line) are for impedance matching of PORT2, whose values have also been changed to match the impedance of the second RF input port of the PE4259 RF Switch.

II. XDS ICSP

A lot of software in the backend of the hardware will enable our project to actually function. In order to add this functionality, we need an interface in order to program the WMCU and any external sensors we intend for it to communicate with – either for identification (such as with I2C protocols and SPI protocols), or for output and general logic necessities, as well as input and choosing which data to push and when. The XDS ICSP is that interface for our project.

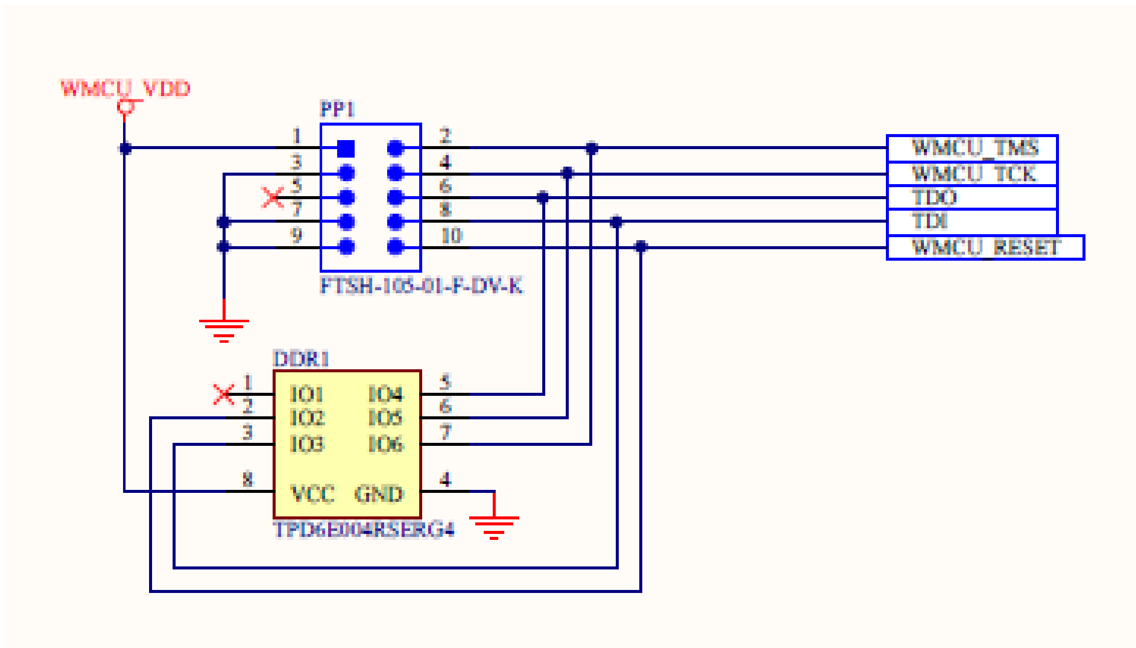


Figure VIII-4: XDS ICSP (Programming Interface) connected to WMCU Schematic

The XDS ICSP is a chip that can be connected to the FTSH-105-01 that Plan Bee is using on the main board. This connection allows the board to be coded to the required specs needed for this project. The FTSH-105-01 also has an electro static discharge (ESD) protector connected to it as well. This ESD protector is the TPD6E004 chip. This connection helps protect the FTSH-105-01 from any ESD damage that any group member could cause by touching the chip directly, up to 15kV which is far beyond what is needed. However, this is the recommended part by the reference design for the ARM4 processor we have embedded.

The XDS chip uses a modification of JTAG debugging protocol. This protocol is further detailed in the standards section. XDS ICSP stands for Extensible Development System in Circuit Serial Programming. This essentially defines what this chip does, which is to program the board itself while directedly connected to it. This type system is used on many different boards, such as the MSP430 and Arduino. Specifically, though, this header will be used to program via XDS the ARM4 Processor embedded in the Wireless MCU within the second schematic herein. This design idea has been realized in the schematic below using Altium Designer.

III. Headers and Peripherals

This schematic shows the second part of the Wireless MCU, and how the MCU part of the CC1352R1 is going to be interfaced with headers and peripherals. The pins shown herein are all DIO, which stands for Digital Input-Output ports, and can be used for either input or output as the acronym suggests. UART is interfaced on DIO_12 and DIO_13 to communicate with an external offboard MCU (via the header shown at the top right) in order to interface any external sensors. UART is also being used in interfacing of the GPS module, as required by the schematic of the GPS module. The pins used for this are DIO_18 and DIO_19. The RF Switch output from the WMCU to choose which communication channel to use (either Sub-1GHz or 2.4 GHz) with the PE2459 is the DIO_30 pin.

SPI is being interfaced with the Flash Memory IC, shown at the bottom within this schematic, and uses WMCU DIO pins DIO_8, DIO_9, DIO_11 as well as the pin for chip select using SPI, DIO_20. The Flash Memory is acting as external storage here in order to store any data and for the WMCU to be able to pull this data whenever necessary. This will be used to store any data not in RAM (the only available storage type in the WMCU) in the interim that is not immediately being pushed to the web server. The NOR Flash Memory IC here adds an additional 8 Mbits of storage to our interface.

Our User Interface schematic necessities are also being taken care of using the GPIO pins available in this section of the MCU. The buttons are interfaced with DIO_14 and DIO_15, the switches are being interfaced with the DIO_21 and DIO_22, and the LEDs are where they were in the afore-given reference schematic, DIO_5 and DIO_6. These are all general input-output (GPIO) pins and can be used for digital logic as programmed.

A second header is added here for solar panel input, for the 12V0, 1.5 Watts between the negative and positive terminals. These are to be routed in the power layer (the second layer under the RF layer) to power components shown in the Power Interface schematic herein. There, the voltage is stepped down to voltages usable by UART, TTL logic and other sensor necessities. These usable potentials are 1V8, 3V3, 5V0 as shown in the Top Level block for the Power Interface. The actual power interface schematic will be shown and talked about in detail below.

The Headers and Peripherals schematic is shown below.

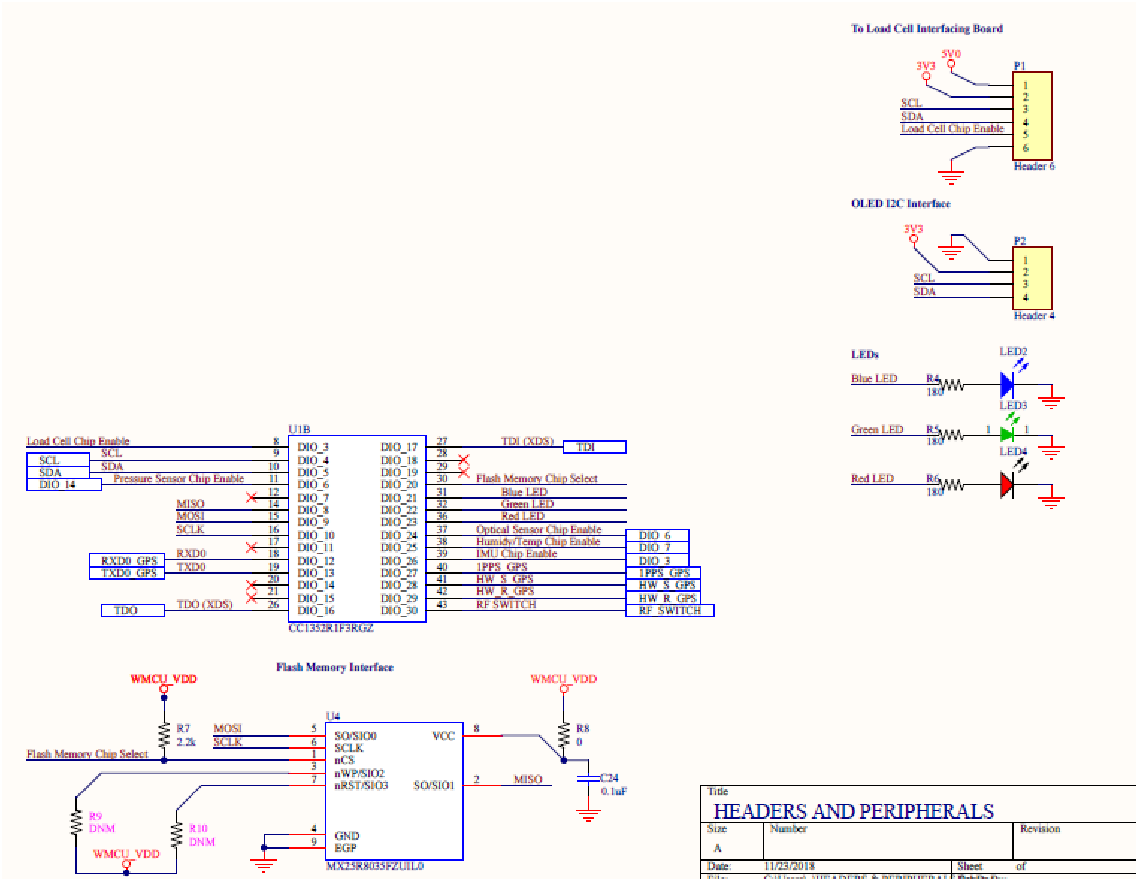


Figure VIII-5: Headers and Peripherals as Part of the WMCU Schematic

IV. GPS Module

We will be using an all-in-one GPS module that includes the receiver and necessary ARM core for processing data from the receiver. The module we are using is the Antennova M2M M10578 A2, which is further elaborated on in the Parts Selection section prior. The output of this receiver is not analog but digital and is communicated via UART, which is interfaced as shown in the Headers and Peripherals schematic shown previously. The output is given as GPS coordinates and time, as standardized by GPS Exchange NMEA Format.

The design we have chosen to implement is very similar to the reference design given by Antennova M2M for the chosen M10578 A2 all-inclusive GPS Module. We have also made a custom footprint to integrate it without external wires directly to the PCB, as per suggestion.

The 1PPS_GPS pin provides a satellite-provided 1-Pulse-Per-Second TTL digital signal that is more accurate than our own crystal clock can be for our WMCU. So, we are going to use this signal to correct our own internal WMCU clock signal as reference, and for incoming GPS signal clock.

The M10578 A2's HW_S hardware-set signal is going to be used to turn on and off the GPS chip when we need to use it. In use this chip consumes 9mA at 3.3V which is equivalent to 29.7 mW of power. To reduce power consumption in the end-product we are only going to enable the GPS when the motion sensor has been "tripped" enough to where the device can essentially be assumed moving. As stated earlier, this will prevent theft of and losing the hive during transportation. The HW_R pin is used to reset the GPS module while in use, resetting the signal if it is incorrect or unintelligible. We may need to use this too depending on testing outcomes.

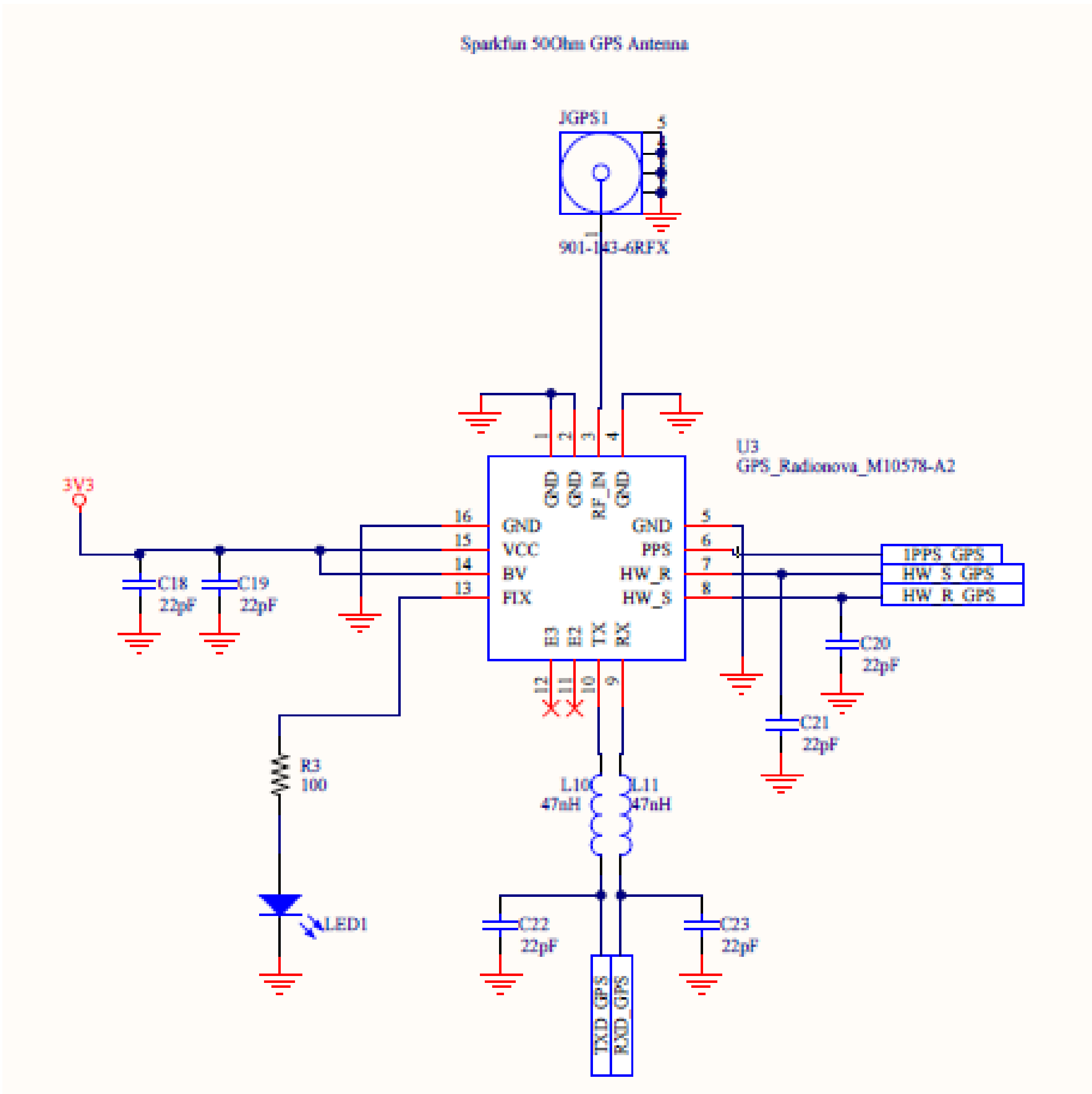


Figure VIII-6: GPS Module Connected to WMCU Schematic

Capacitor C30 and C31 are 22pF each and are for low-pass-filtering and "smoothing" the incoming 3V3 power signal. The GND pins are all attached to internal PCB ground (second to bottom layer) as specified in the datasheet for the Antennova M2M GPS Module. The FIX Pin pulses as a 555 Timer does at 1 Hz when a GPS signal from satellite is found and "fixed" upon – meaning that it is being retrieved – and because we want to be able to see this during testing we have set it to pulse a blue LED that will be by the module. Capacitors C32 and C33 are also for signal filtering for the digital TTL hardware-set and hardware-reset signals to the WMCU. The digital UART (RS232) communication lines are also being filtered by both L11 and L12 and by C34 and C35. This slightly changes phase of the digital signals and allows only low-pass-bands very close to DC.

We won't have backup power, but the solar panel is expected to function indefinitely, so we have routed the filtered (with the two 22 uF filtering capacitors) 3V3 power input directly to both the VCC (biasing voltage) and BV (Backup Voltage) pins, which differentiates our own design from the given reference design by Antennova M2M.

We are using the Spark Fun Electronics 50 Ohm GPS Antenna. The module itself has a 50 Ohm output impedance (out of the RF_IN port shown above). So, again to match this as we did in the RF WMCU module schematic, we have made the routing lines for the PCB 12 mils. This will maximize power transfer ensuring the strongest GPS signal is achieved into the GPS receiver module. This should ensure that we get useful readings in most conditions. For the GPS Antenna module, designed for 1.5241 GHz to 1.6000 GHz (GPS communication signal frequency band) narrow frequency band, uses a female SMA connected, and is sold by Spark Fun Electronics. To match this, we have purchased and included the accompanying Spark fun Electronics SMA male connector onboard specifically for this GPS module.

Antennova M2M has also offered experienced staff in order to aid us with successful implementation, which he have not used up to now. We may use their knowhow relating to the GPS Module if we encounter issues during our testing stage specifically with the GPS Module.

V. On Board Sensors

In the onboard sensors interface includes sensors for temperature, humidity, 3-directional motion, pressure and light. The sensors are going to use the third and fourth layer of the PCB and communicate using the SCL and SDA I2C signal input/output lines. Each sensor will have an identifier and push data when its ID is called by the WMCU, as the I2C protocol specifies. A separate schematic will be added for all of these sensors together on the I2C bus, when we are ready to fabricate the first PCB. However, the individual sensor schematics as required and referenced in our Top Sheet are currently found in the Design Implementation section, under Hardware.

VI. Off Board Sensors Interface

The offboard sensors interface will use the MSP430G2231 Ultra-low-power MCU to intake digital and analog sensors using I2C protocols as discussed for the onboard sensors. However, we will create a separate PCB for these external sensors similar to the onboard sensors interface. It will communicate with the main board via SPI, using the MSP430G2231 UART connection pins as well, as shown in the Headers and Peripherals schematic.

For this offboard module we will include sensors for temperature, humidity and light concentration. We will also be including sensors for the monitoring of food, medication, rain, vibration, and interfacing external-to-this-board load cells in order to track weight of the hive overall. A similar header to the ones shown previously will needed to be added with 5V0 available, as well as ground and I2C-required pins for signal input/output lines.

We plan to add Bluetooth capability to this module as well as backup power via a rechargeable coin-cell, so no extra wires are required. The MSP430G MCU is expected to be capable of running on the a 1000 mAh coin cell for more than three years when properly making use of the low-power mode when not in daily use. After this three years, the coin cell will need to be replaced so it will need to be easily accessible.

The placement of the Offboard sensors module will be on the inside wall of the hive opposite to the externally placed electronics. The reason for this placement is that the vibration sensors can be used in this manner to track the density of the hive, and thus give a good estimate of the amount of honey in the hive. The internal vibration sensor is also planned to be used to record bee activity (amplitude and frequency of vibration is measured) within the hive throughout the day. This will give the beekeeper a good idea of how and when the bees within the hive are active.

VII. User Interface

We are including a User Interface for our Senior Design project to add clarity of service and what each sensor is reporting during testing, to add attractiveness to the overall project, and for the ease-of-setup and continued maintenance of each end-product. The goal of the interface is basically to allow for effective operation and control of the machine from a human onsite, such as a beekeeper.

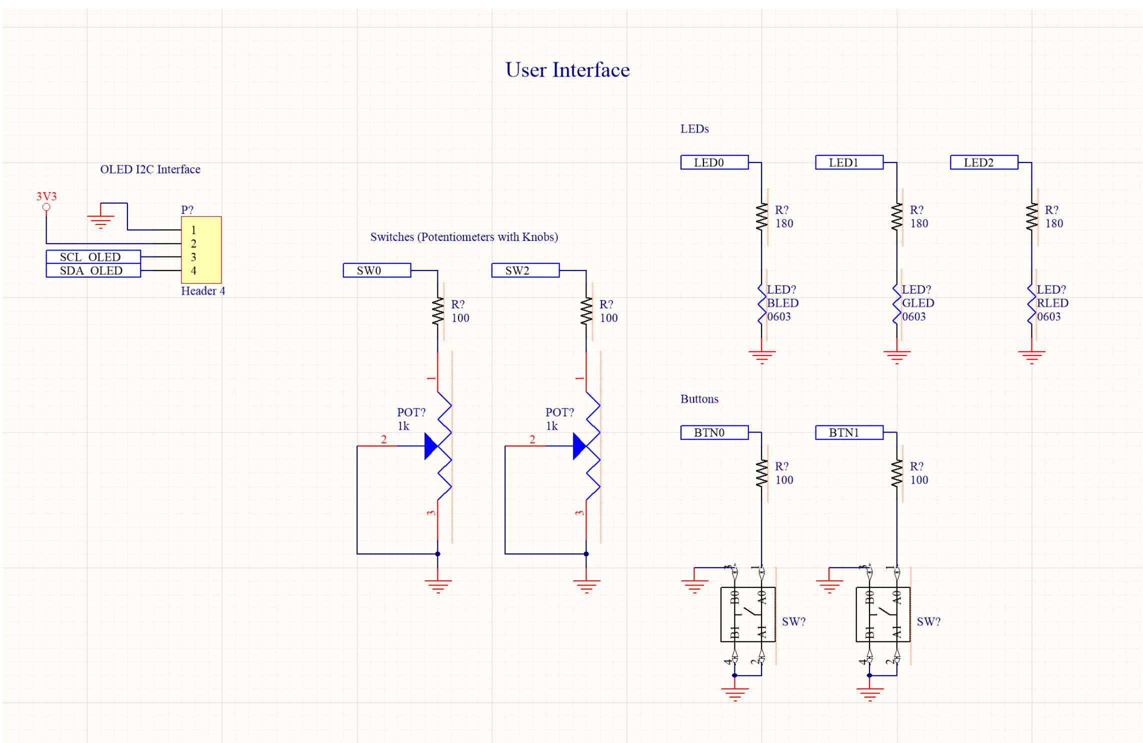


Figure VIII-7: User Interface connecting to WMCU Schematic

The components we have chosen to include in the user interface are an OLED display, two pushbuttons, two potentiometers for interacting and changing the user interface (for example moving up or down can be accomplished via a knob), three LEDs, and two pushbuttons. the pushbuttons will be used to select when food has been refilled and medicine has been refilled. The LEDs will each serve a different purpose, signaling communication via Zigbee is occurring, or 2.4 GHz signals are being pushed, for example. The OLED interface will provide the screen on which information will be displayed. We chose an OLED display for this project's interface because it is far more durable than comparable LED or LCD screens within the same price range. The brightness may cause an issue in broad daylight, but this can of course be corrected. All of the User Interface functionality will all be added within the programming phase of our project.

VIII. Power Interface

The power interface is quite simple and not entirely changeable. The solar panel takes in a maximum of 10 Volts and passes it through the charger regulator which loads the Lithium-Ion battery of 3.7 Volts. This voltage is later stepped down using a customizable buck converter unit that is set depending on the required input power which can change depending on which and how many sensors are on or off. The power rails shall then go along the third layer from the top of the PCB and ground will run along the second layer from the top of the PCB.

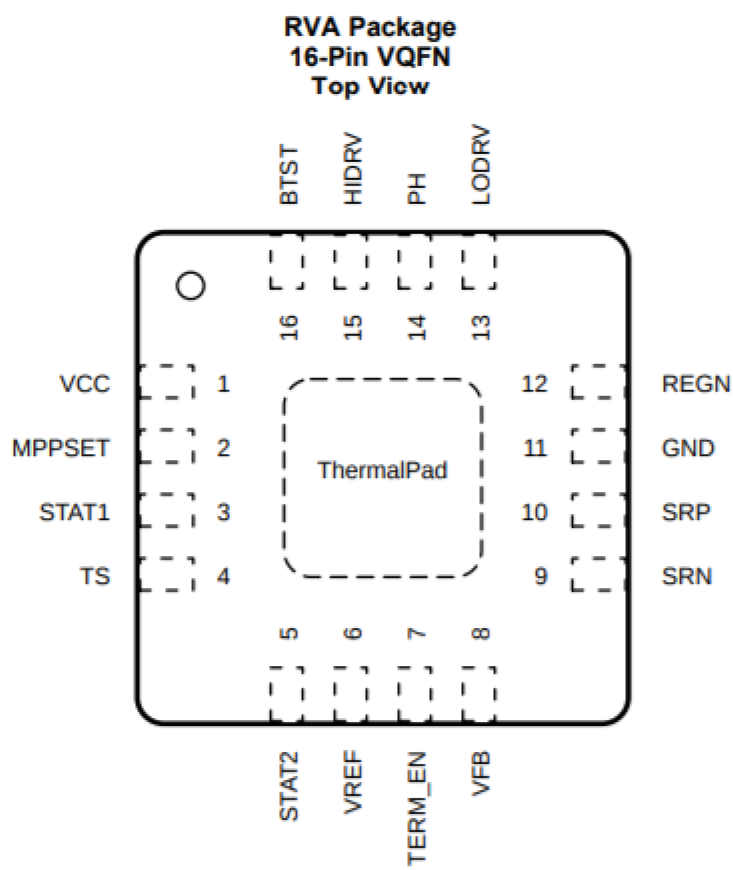


Figure VIII-8: Solar Panel Charger Controller

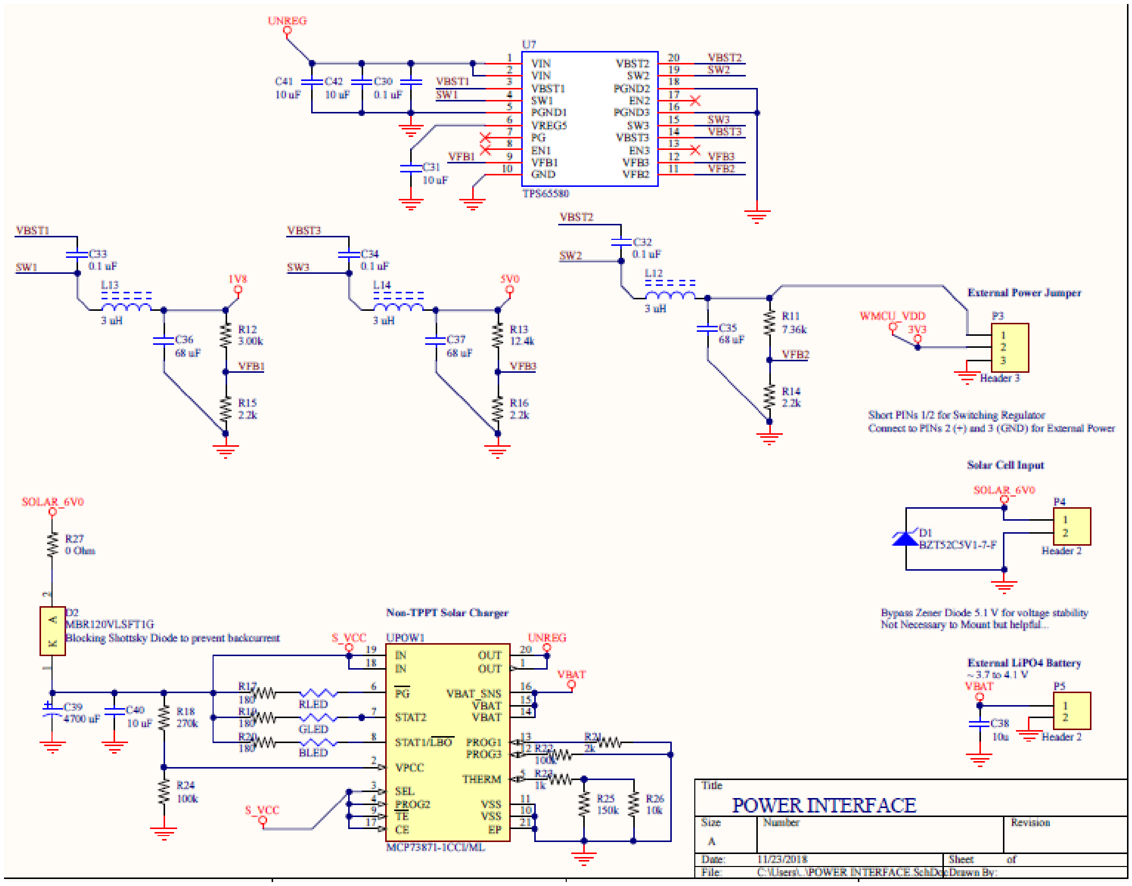


Figure VII-9:8 Power Iterface

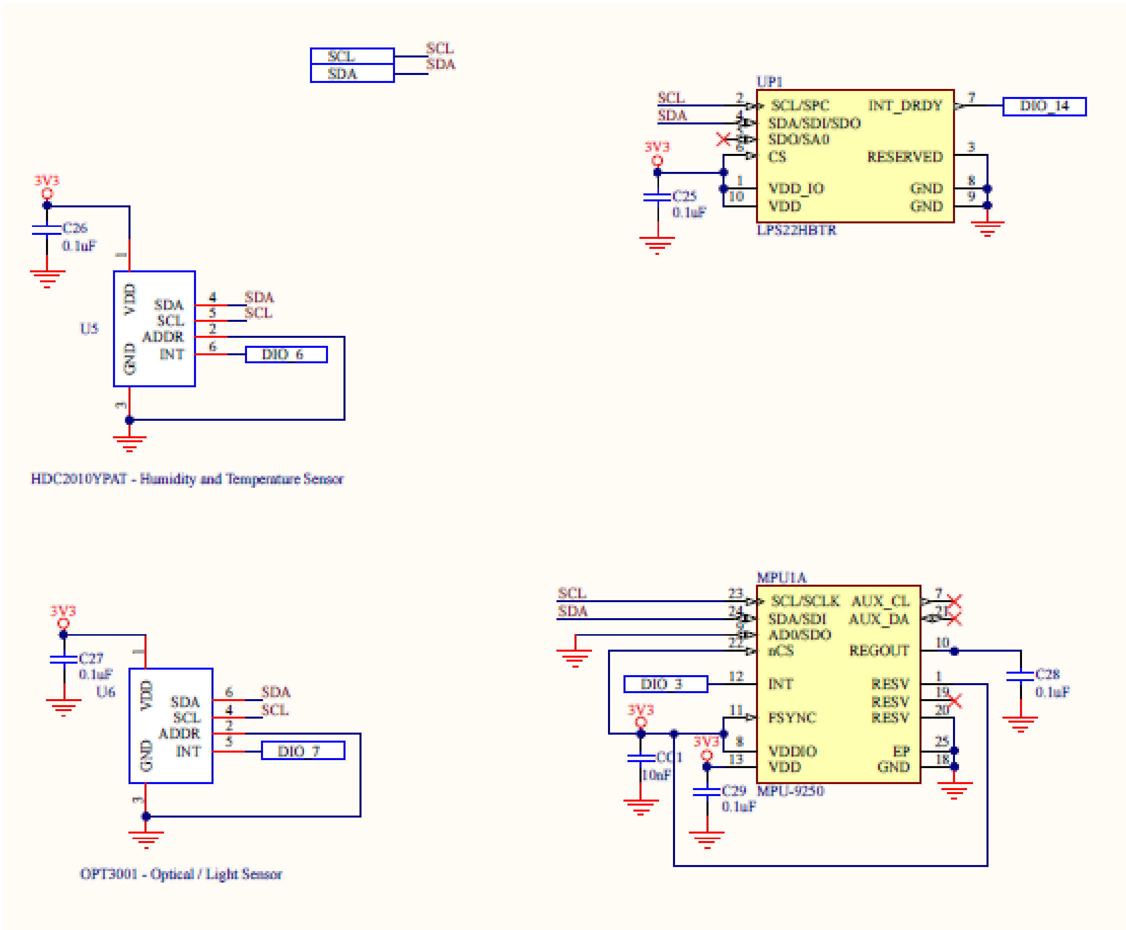


Figure VII-9:8 Onboard Sensors

IX. PCB Testing Plan & Referenced Testing Points

Although not shown in the above schematics, within the PCB custom testing pads and vias have been added throughout. They are 35 mils in diameter and plated. Vias have been added for the power components (second layer) and all communications lines (in the successive layers) for both RS232 and RS485 UARTs setup, the SPI setup, and the I2C interfaces and to-be-connected-offboard pins and are open in the top PCB-insulation layer as to allow for ease of testing (we do not have to open these pads ourselves by other means such as sanding). They are added inline with trace. Pads have been added for the GPS communications lines specifically to and from the antenna, as well as TTL logic lines to the WMCU. They have been added as pads only because they are nonintrusive and on the top layer. They have been added off-line with the traces and are connected to the traces because we want to maintain the 50 Ohm impedance of routing lines (traces) to allow for maximum power transfer to 50 Ohm components herein and maintain maximum integrity of RF signals.

All Pads and vias are spaced at least 5 mils apart from each other and are at least 5 mils in width. We will be using a multimeter probe to test these pads and vias throughout the PCB. We are expecting to need three rounds of PCBs printed and fabricated because of the complexity of our project and the number of components that could potentially be faulty. Most of our time from here will be spent on testing and fabricating.

II. Application Flow Details

I. Overall Application Layer Overview

This section attempts to describe the overall application integration details. The software portion of Plan-Bee will be divided into 5 stages. Each stage is assigned a range of responsibilities which will enable the project to accomplish the listed requirement specifications. The project's primary goal would be to present sensor data, collected from some arbitrary environment, to an end user. Each stage plays an integral role in accomplishing such task, and hence cannot be decoupled, or isolated from the rest of the toolchain.

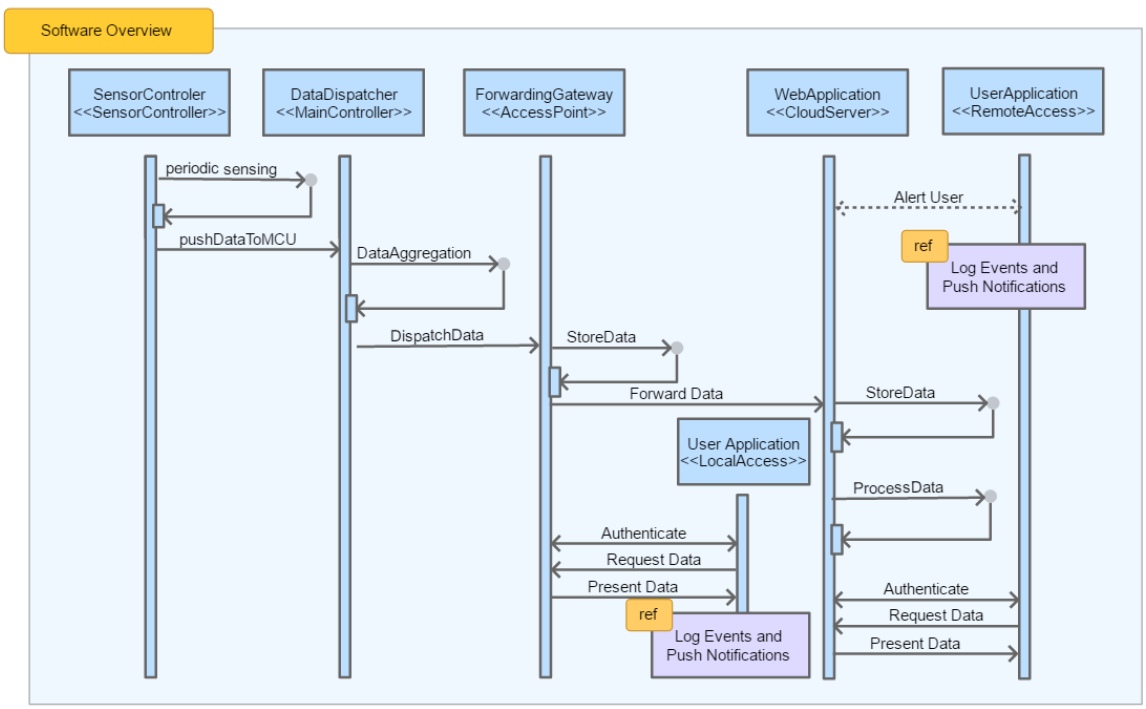


Figure VIII-9 Software Tools Overview

The software stack encompasses:

- **SensorController**: This application is responsible for gathering sensor data and pushing said data to the main controller unit. The application is executed on the low-powered ARM processor dedicated to reading sensory data and consists of multiple tasks being executed in a periodic manner. As a result, the overall periodic invocation of the application will be dependent on total sensory tasks to be executed. Each task is responsible for initializing a communication device, initializing the sensor device, reading sensor data, and terminating both sensor device and communication device. When the sensor data is read, it is then stored in main memory for the data logger to store the required data to non-volatile storage.

- **DataDispatcher**: The DataDispatcher is executed on the main controller unit (MCU). It is responsible for both logging and dispatching data and, therefore, comprises for two tasks, the DataAggregator and DataDispatcher. The data aggregator stage is usually invoked by the SensorController. Upon invocation, the DataAggregator reads the raw sensor data; does the necessary translation, depending on type; appends additional data elements to aid in the querying process; and finally stores the resulting data structure in non-volatile memory. The DataDispatcher complements the Aggregator by scheduling what types are to be transmitted to an access point, and transmitting said data while also managing routing and other transmission details.

- **ForwardingGateway**: The ForwardingGateway lives on the access point. This application is responsible for translating sensor data to valid values to be stored on a local database, and synchronizing said database with a registered cloud server or using a connected mobile device. Because of time constraints the synchronization process will encompass copying the database elements to an attached device and have the device manage synchronization. To keep transfers small, successfully synchronized data will be archived and removed for an active syncing database. The ForwardingGateway may also act as a quasi-application server: providing data to an authenticated mobile device. For a stretch goal, if the access point is unable to connect to the internet, the mobile application me be used as a network gateway to synchronize data with the cloud. This may permit users to continuously synchronize local data with the cloud, even in remote areas without wireless internet access.

- **WebService**: This application is responsible for receiving sensor data and presenting said data to an authenticated user through request or alerts. Sensor data could be received from an internet connected access point or from a mobile device. Upon receiving specific data, the WebService will inspect the data for abnormalities. If found, the WebService will trigger an alert for an authenticated end user. The WebService will also provide a robust API for accessing said data. This will allow both online applications and mobile applications to access and present the data using the most effective user interface for the device in use. We use Amazon AWS for data storage to a database and pushing the data to the end user application

- **UserApplication**: The end UserApplication will be executed either as a mobile application or a web application. The main responsibility for a user application is to obtain data from either a WebService application or ForwardingGateway application and present it to the end user in a intuitive fashion for the end user. Th e end user application should also be able to synchronize its local database with that of the WebService. This will allow data aggregated to a gateway to be pushed to a web server through the end user's mobile device.

Given brief description of the listed five stages, this software overview will further go in depth into how such application will be implemented and integrated into a cohesive experience for the end user. Because of time constraints, the major focus of this project will be on the application firmware of the data tracking tool. Other applications will rely mostly on third-party libraries to provide the necessary functionality in the constricted time. As such, such applications will have limited integration coverage.

II. Sensor Controller Application Sequence

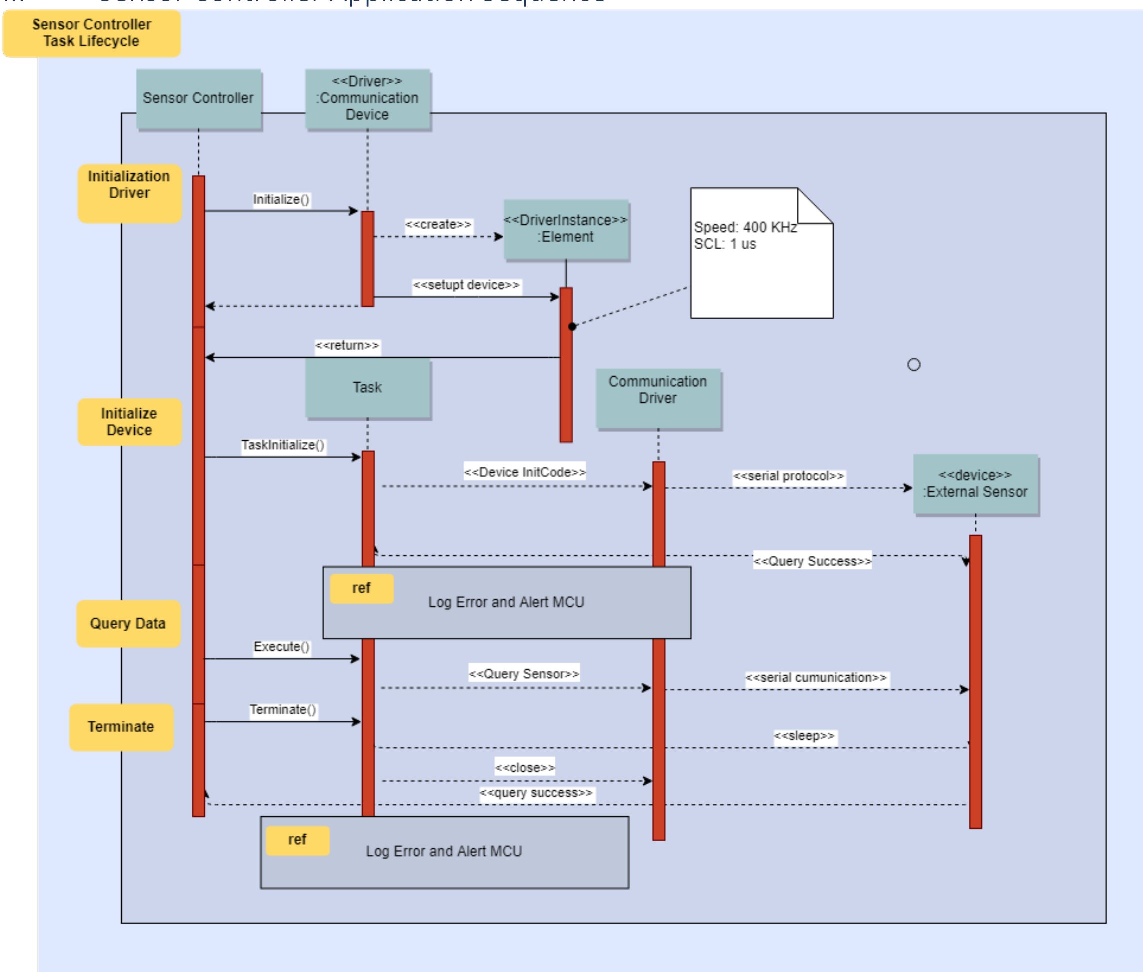


Figure VIII-10: Sensor Controller Application Sequence

The CC1352 incorporates an ultra-low powered ARM processor used for acquiring sensory data. This data is then shared with a task running on the main MCU using memory controlling elements. To execute code directly on the sensor controller; an application would be precompiled and integrated within the main program code. During execution the main processor transfers the compiled application binary to a memory mapped location, where the sensor application would be initialized to execute the code at the specified address. A sensor controller application utilizes a basic scheduler to manage a set of application processes. These processes are consigned to acquire sensory data form the external devices connected through UART, SPI or I^2C. When initialized, each task passes through a specific application lifecycle where it is responsible for managing memory allocation, and deallocation; application drivers; and passing data from the sensor controller to the main controller. The stages of the application lifecycle involve an initialization stage; execution stage; and termination stage.

The sensor controller would periodically wake from a sleep state and go through each stage before going back to sleep. This allows the CC1352 to utilize significantly little power; on the order of 0.08 micro-amps. This not only minimizes power consumption but also relegates more pressing task to the main micro-controller unit. For I^2C devices, the initialization stage involves setting up the I^2C bus. The bus is initialized with a frequency of 400 kilo-hertz and a stretch timeout of 1 microsecond. The I^2C communication is also setup to support 8-bit bus communication.

After initializing the communication protocol, the sensor controller module then initializes the specific devices to be queried. This is done using device specific code and relies on the sensor controller sending initialization command across I^2C bus. After the initialization stage, the sensor controller then queries the necessary data from the device. If the received data was returned corrupted, or communication was hindered during this stage, the sensor controller logs a communication error with the main controller and then continues to the next process. The troubled process will be return to on the proceeding cycle and the number of attempts logged. If data is acquired successfully, the sensor controller stores the output data in a special memory address where it could be accessed by the main controller. The sensor controller then closes the initialized buffers and drivers before proceeding to the next task in the task queue.

III. Main Dispatcher Application Sequence

Texas Instruments' TI-RTOS provides a robust application programming interface which is usually matched with their many testing tools and helper programs. The embedded kernel features task scheduling, memory management and protection using semaphores, and interrupt management. A task consists of a memory stack, heap location and is also assigned a task priority. Task are also divided into 4 types. Hardware interrupts (HWI), software interrupt (SWI), application task, and idle task. The types are also prioritized where HWI has the leading priority.

Hardware Interrupt Tasks (HWI) are specifically used to manage hardware interrupts created from external devices and are therefore blocking. As such, the scheduler executes task depending on their type and priority. Hardware interrupts cannot be preempted, which allows the task to run to completion.

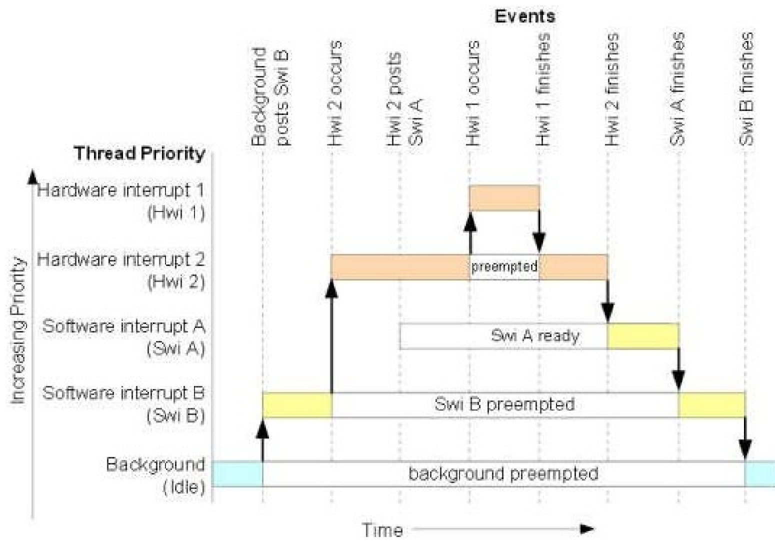


Figure VIII-11 Software and Hardware Interrupting Schemes

The Figure above describes the priority preemption rules utilized by Texas Instruments RTOS. Software interrupts are able to be preempted by higher priority software interrupts and also hardware interrupts. Tasks running on TI-RTOS are classified lower than HWI and SWI and as such can be preempted out. This may pose an issues as important tasks may be preempted while completing time constraint processes. To combat this issue, important tasks will be made non-preempting using semaphores and other memory protection elements.

The main Data Logger firmware will depend on RTOS tasks to complete their specific agendas. Depending on their goals, these tasks will be executed using a scheduled timer, or using a software interrupt. Software interrupts are used to pass data from sensor controller to the main controller unit. These interrupts are assigned an interrupt services routine which is carried out before it is remove form he scheduler. The firmware will incorporate such task to complete a variety of problems; important of which would be sensors data collection, computation and storage.

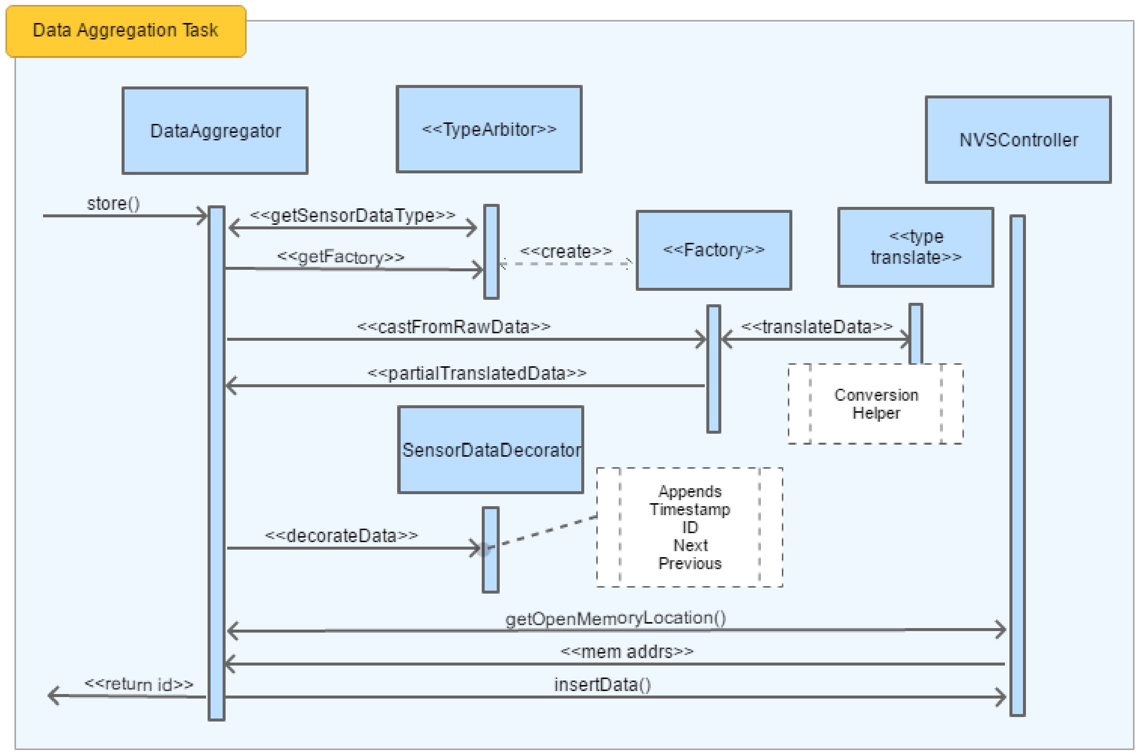


Figure VIII-12 Data Aggregation Sequence Diagram

Data Aggregator

The above figure describes the Data Aggregation sequence diagram. This illustration 3 major classes which encompass this element. The DataAggregator class acts as a front end to process sensor controller data. To elaborate, during the sensor controller task's execution stage, the collected output data is places at a specific location in memory. Upon termination the sensor controller then triggers a software interrupt on the main micro controller unit which is then executed using the Task scheduler. The software interrupt first copies the raw data to another location in memory to avoid blocking the sensor controller. The data is then passed to a DataAggregator; a class which follows the singleton pattern. This class acts as mediator between raw data collected from the sensor and data stored and the data to be stored to the local datastore.

The DataAggregator class begins by receiving a pointer to raw sensor controller data located in memory. First, the DataAggregator copies the raw data from its memory location and store said data to a specific location in memory. This data is then passed to the store Method of the DataAggregator class, where it is further pass to a TypeArbitor for obtain a handle to the sensor data's type descriptor. Using a typing hinting value stored in the data structure produced by the sensor controller the TypeArbitor is then able to provide a DataStoreFactory object corresponding to the data type hinted to by the data-structure.

An instance of this factory is the loaded and returned to the DataAggregator. The factory is then used to convert the raw data stream into actual data values represented by the sensors. These values are calculated using a specific translation routine for the type of sensor data. After receiving the processed data, the data aggregator then decorates the data structure. The Decoration process involves adding additional data elements not available in the raw data output. These data elements include a timestamp, an ID to identify the data element in the database, a pointer to the previous element in memory, a pointer to the next element in the database, a value specifying whether the data has been uploaded to the cloud, and a value specifying the index of the data element. After processing the data structure, the DataAggregator then accesses the NVSController. Meaning Non-Volatile Storage Controller, this class is responsible for store and accessing memory elements from main memory. To store such elements to non-volatile memory, such elements should be properly validated and processed.

The NVSController stores the address of the top of the memory heap. This address is first obtained, then the data is stored, using such address. At this point said header pointer is then updated to the new top of the storage stack. After the data has been successfully placed in non-volatile memory, the indexed value of said data point is returned to the data aggregator. This index could be used to retrieve the data element when needed. After completion the data aggregator class deletes the memory location used and logs the result of the return value before terminating. The value at the top of the stack will be stored at a specific where it will be logged for future querying.

In addition to becoming invoked by the sensor controller, the DataAggregator may also be manually invoked to store data not collected by the sensor controller task. This will be accomplished by storing data to a specific location and calling the DataAggregator with an optional argument to hint at the type. The DataAggregator will always clean the memory used upon completion. Meaning the Class should be reinitialized upon calling the start function. The returned identification will be managed by the DispatchCoordinator. This class will be used to count the number of objects that object that have not been dispatched for a specific class. Given, that each Datatype will be assigned a priority, the DispatchCoordinator will calculate the Datatype which should be next to be dispatcher to an access point. This data will also be stored to non-volatile memory for future references.

Data Dispatcher

The DataDispatcher class will be used to complement the Data Aggregation task, in further transmitting the collected sensor data to an access point. The DataDispatcher class is implemented using the singleton pattern. The class will be invoked using a clocked task which will be called during specific intervals, approximately twice an hour or depending on the amount of data to be Dispatched to an access point.

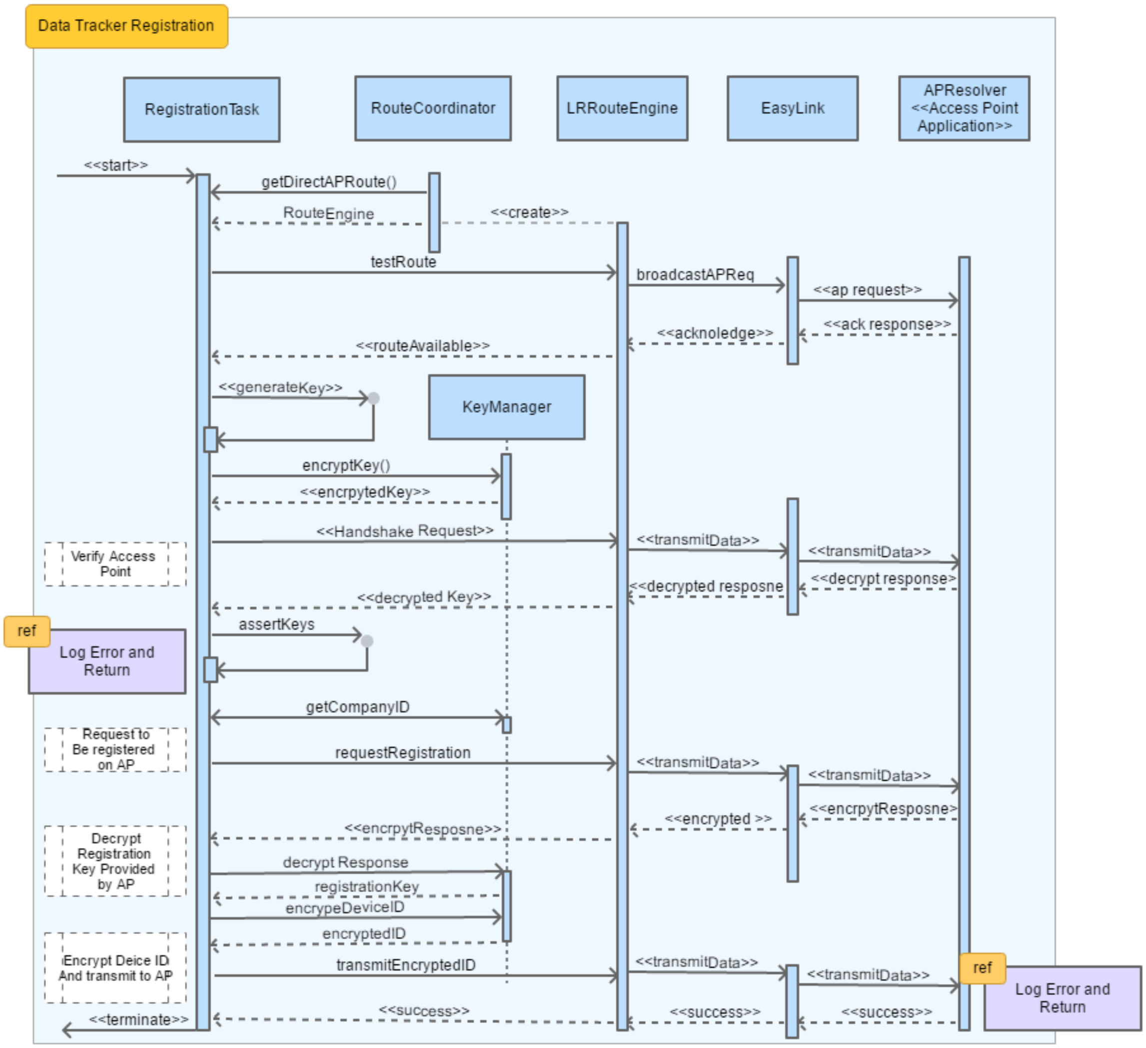


Figure VIII-13 Device Registration

In order to dispatch data to an access point, the tracking device must be pass through a security approval process which first involves ensuring the access point is authenticated as a valid device of the company. To initialize this process the registration task is invoked. The task then request a valid route to an access point. Upon testing the route, the data tracking device will encrypt a randomly generated number and transmit the value to an access point. The access point will then decrypt the value using an internal private key and return the decrypted response. The decrypted value will then be compared against the stored created value for success. Upon asserting that the values match, the will then attempt to register it's self with the access point. This process is done by first transmitting an assigned company's ID number whereby the access point will respond with an encrypted registration key. The registration task will then used said registration key its device identification. This encrypted identification is then transmitted to the access point where it will be decrypted and stored in the registration database. Upon success the access point will return an acknowledgement which will allow the data tracking device to also save the access point ID and company ID its internal database before terminating. While the registration process is similar to that of HTTPS, it lacks many of the details handshaking routines due to the lack of a TCP/IP stack. As a result a simplistic, and maybe naïve, authentication process was taken. After an initial registration, the device may then be able to join the herd of other devices where data could be transmitted across multiple devices to reach a final access point destination.

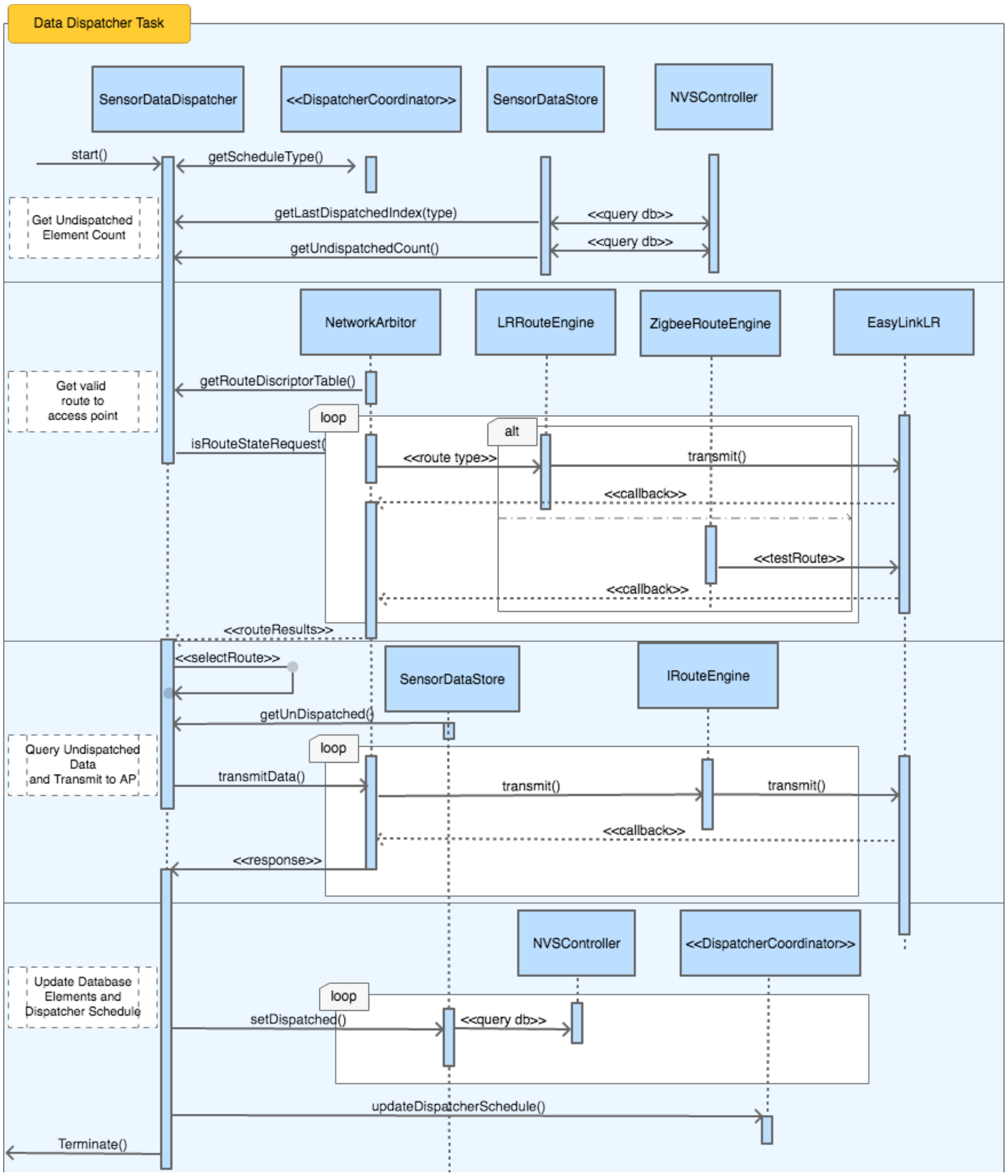


Figure VIII-14 Data Dispatching Sequence Diagram

The DataDispatcher class may also be invoked by a software interrupt. Such an interrupt will be triggered from an access point attempting to communicate with the specific data logger device. The access point will first send a broadcast signal alerting all data logging devices in range to wait for a time period to respond to a data request message. The access point will then transmit a request message, which includes an address to the specific data tracking devices the request is directed toward. While all other data trackers will ignore the request, the addressed data tracker will then trigger the software interrupt to trigger a DataDispatcherTask. The entire communicate scheme will be encrypted end to end. The DataDispatcherTask class will be made non-preemptable to disallow other task from interrupting execution and data transmission. This will also protect from data corruption during modification of the non-volatile storage, or during transmission.

The figure below illustrates the means of dispatching data to an access point. When invoked, the DataDispatcherTask first checks to see what type of data is scheduled for dispatching. This is done by invoking the DispatchCoordinator, another singleton whose role is to track and schedule sensors types that has been previous dispatched. Not shown in the illustration is the DispatchCoordinator integration with non-volatile storage. In essence, the DispatchCoordinator uses a special scheduling algorithm to select what sensor-data-type should be dispatched next. The sensor-data-type are all assigned specific priorities and an allotted amount of dispatches per period in time. If a type is dispatched successfully, the DispatchCoordinator logs the event, the total amount of data transmitted, and amount of data remaining that is specific to the transmitted type.

After requesting the next scheduled type to be transmitted the DataDispatcher then queries the last dispatched index from the non-volatile storage. The DataDispatcher also request the number of un-dispatched data from the specific type. This is used to check and validate whether data is available to dispatched from the datastore, and to also alert the access point as to how much data to expect. The DataDispatcher requests such data from the SensorDataStore. This class is meant abstract data access utilizing the non-volatile memory. More specifically, SensorDataStore manages only sensory data, providing an easy to use interface for creating, reading, updating, and deleting sensor data stored in non-volatile memory. After obtaining a specific count of the number of data elements to be transferred the Dispatcher then attempts to locate a possible route which could be taken to accessing the access point. The NetworkArbitor class stores and manages all attempted routes to an access point device.

This is done using Zigbee and the long-range wireless protocol, where each device attempts to communicate with a local access point. If an access point is found, the endpoints will note that it's able to communicate with an access point. It will then store the value of the received signal strength indicator (RSSI). This route will be logged in a database accessible by the NetworkArbitor.

If a device is unable to communicate with an access point, it may broadcast an SOS signal to neighboring devices. Each device will then listen in on the SOS signal. Using listen before talk (LBT), a single device, which have had success communicating with an access point will respond to the SOS within the allotted time. This initial communication will then be followed by a device address and route table to its access point. This newly defined route will be logged in the device, unable to communicate directly with an access point. As a result, the route will be available as a possible solution for the NetworkArbitor routes. These routes will also be assigned expiration dates, as to when they should be purged from the routing table database.

The DataDispatcher class receives a list of RoutingDiscriptor from the NetworkArbitor class. A RoutingDiscriptor can be considered a handle to a possible successful route stored in NetworkArbitor database. If a route is not received, the DataDispatcher logs such error and returns. The log allows the system to go into SOS mode where it attempts to obtain a possible route to an access point, as described above. If a RoutingDiscriptor list is returned with multiple routes, the DataDispatcher will first attempt to communicate with the previously connected access point directly (this happens even when the access point is not available in the provided routing table).

If a direct route is successfully made to an access point, the search for additional routes will be abandoned and said initial access point will be returned. If a direct route to an access point is not found, the NetworkArbitor will then loop through each additional route and attempt communication. The RSSI value will then be compared and the return list be sorted from best route to worst. Routes will be attempted using Texas instruments EasyLink api. This allows seamless wireless communication across both long range 0.9 GHz and shorter range 2.4 GHz. This is done whereby the current device will be attempting to communicate with whatever device is defined by the route and seek to obtain access to an access point.

Upon a successful route, the un-dispatched data is finally requested from the SensorDataStore. This data will be validated and, if the count is greater than maximum transmitted, a portion will be transmitted using the previously determined route. If the transmission fails, the DataDispatcher task will log said failure and exit. If the transmission was successful, the DataDispatcher class will update the Dispatched flag of the data elements stored in the non-volatile memory. The DataDispatcher will then update the dispatched schedule by decrementing the total dispatches achieved within the specified period and assigning a timestamp on the dispatch database element for the DispatchCoordinator to use.

IV. Access Point Data Receiver

Using the integrated sensor controller, data is recorded and dispatched to an access point where it will be forwarded to a cloud server for processing and storage. If an access point has limited to no access to a cloud server, the access point will store the allotted information until access to the cloud platform is created, or a user can decide to connect to the access point directly to download the newly generated sensor data. The storage process is illustrated in the figure below. After an authentication process, a push request could be initiated by either an access point or data logger. Data will then be transmitted using Texas Instruments' EasyLink protocol. Data transmission will be managed by the access point GatewayReceiver process. This process is a daemon running as long as the gateway is operation. After reception, the process stores the received data to a local MySQL database using a Database abstraction layer.

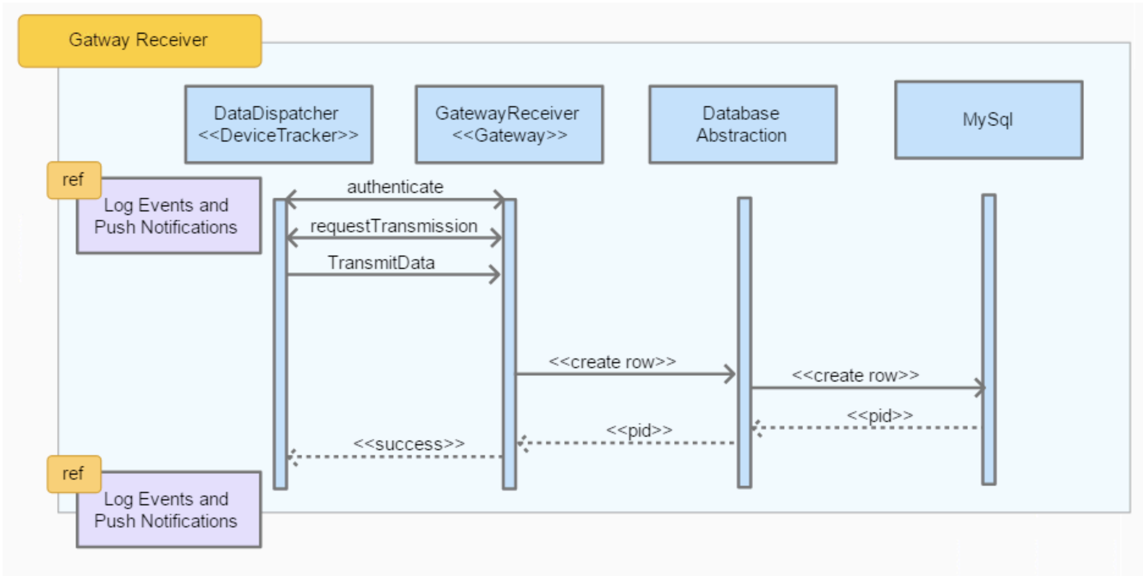


Figure VIII-15: Access Point Receiver Sequence Diagram

Because of the time constraints this process will be written in python. If data is successfully appended to the database, the gateway receiver will transmit a success packet bac to the data logging device. This will enable the logging device to log the entries as transmitted and void duplicate data element being transferred – sensor data identification will also avoid duplicate transfers. If the transmitted data is already located in the local database, the gateway receiver will send a success packet to let the tracking device mark the data as dispatched. At any point an error is thrown, the GatewayReceiver will back track from any data that has been stored, log the error, and responded with an error packet for the tracking device to be notified.

V. Access Point Data Forwarding.

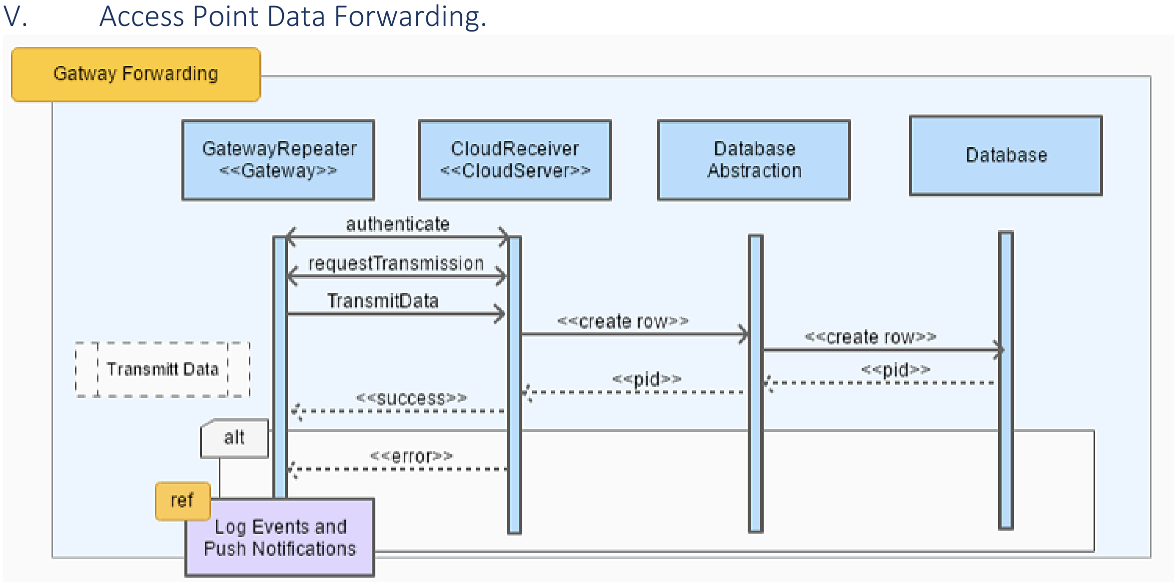


Figure VIII-16 Gateway Forwarding Sequence Diagram

In addition to the receiver process, the repeater service will also be running on the gateway device as a daemon. This process will be executed as long as the gateway is active. Its responsibilities are to synchronize the local database with a cloud database. To do so, the repeater service will rely on MySQL's sync API. This is a collection of function to aid with database synchronization across database on differing networks. As a result, the gateway process will be left to manage authentication and ensure data has been successfully transmitted and synchronized with a cloud server. If a failure occurs the cloud receiver will backtrack on all commits, log the error, and return an error packet to the gateway repeater. The gateway repeater will then attempt to send the data again. If the number of attempts reaches a certain could, all attempts will be discontinued, and an error will be logged for the end user to manage.

VI. CloudService

As previously noted, the cloud application will utilize Amazon's AWS hosting services. The application is responsible for managing cloud database access and appending data to and from the database. The data may be provided from an access point's repeater process, as described below, or the user's mobile application, as presented in our stretch goals.

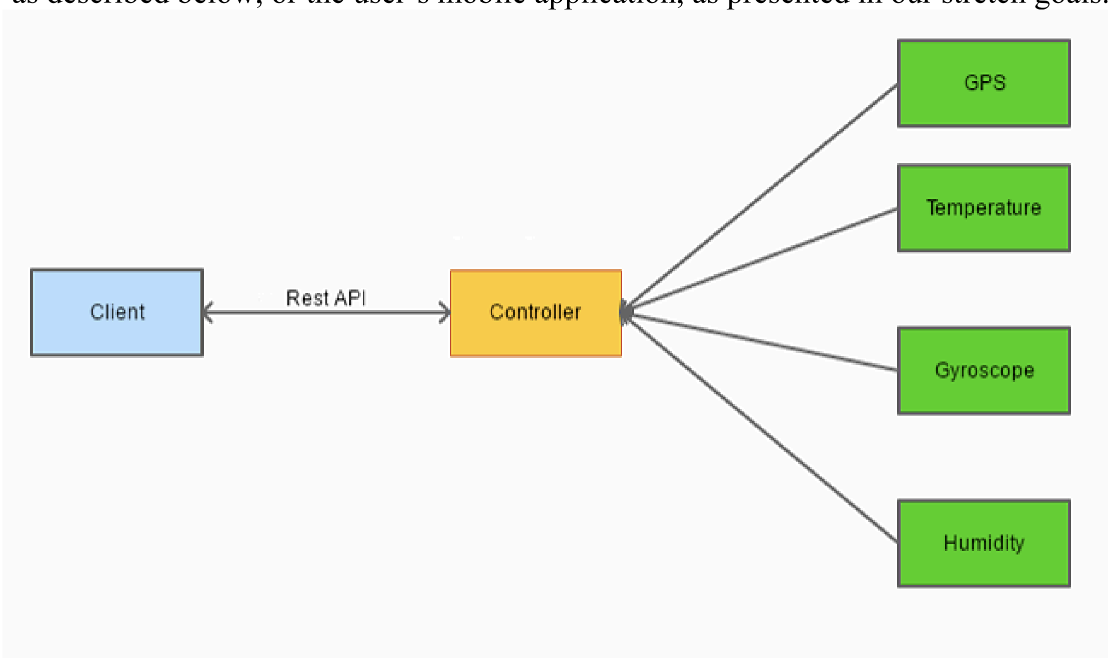


Figure VIII-17 Cloud Service Data Flow

The CloudService's second responsibility is to provide an application programming interface API for accessing said data to be presented by a frontend framework. The API will follow a generic REST interface which relies on create, read, update, delete requests on an authenticated user. The controller will only expose read only calls for sensor data to prevent end users from manipulating previously recorded temperature data.

VII. Mobile Application

The mobile application sequence is a simple app user authentication structure. The user asks for access from the application and the application acknowledges the user back through a form of identification. Once the user has accessed the application, the user can request for bee yard data.

The application accesses information from the cloud service which has been gathered from each database of each individual sensor holding unit. The cloud service then delivers the desired data to the mobile app that is being accessed by the user. These units also send push notifications for select settings straight to the user back through the same path.

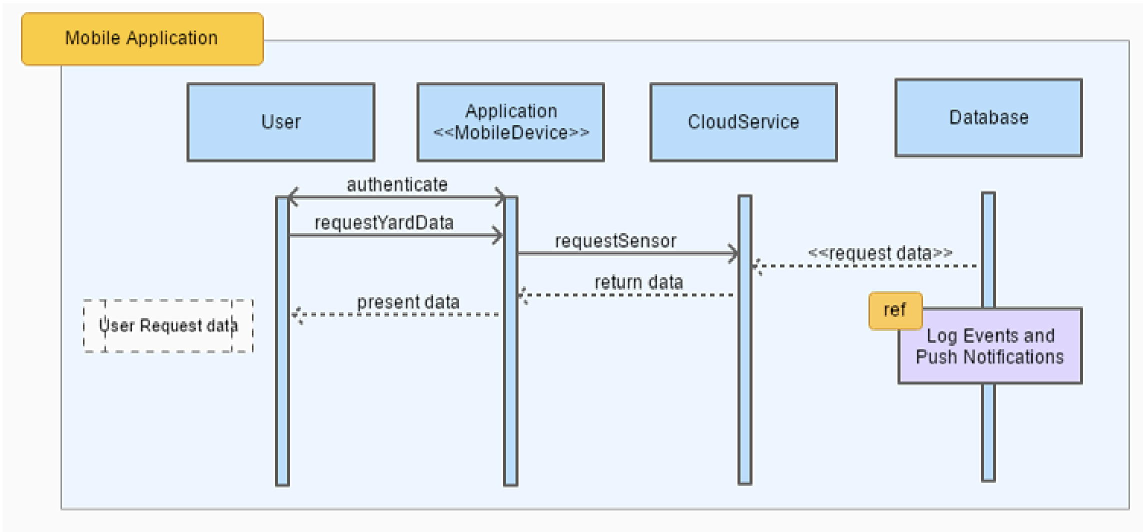


Figure VIII-18: Mobile Application Sequence Diagram

IX. Administration

This section discusses implementation costs and membership responsibilities. Member responsibilities includes a breakdown of the technologies present within our end-product, and who is expected to be responsible for those technologies. Milestones includes a week-by-week listing of deliverables, and who is doing them, until the very end of senior design II, to realize our project in a timely and efficient manner. Under implementation cost, we have included an initial budget. This is an initial budget and by no means final but is also an overestimate to accommodate any unexpected expenses in our projections. Financials also includes Running Tabulations, which will include receipts from all parts ordered and any associated project costs to the sponsor, as well as a running tabulation from these receipts of how much of our budget we have spent. At the end of senior design II, we will request at least three (3) quotes from different manufacturers for our end-product in full implementation – at least one of these must feasibly remain under $70 to meet our product design constraints discussed in the Constraints section. *This section is subject to updates and change consistently, as we progress.*

I. Member Responsibilities

The diagram below shows our initial team member responsibilities. Software development will largely fall into the responsibilities of Yannick Roberts. Tariq Ausaf will be responsible for some software design and implementation, but more so the major PCB design, RF Communications, and the LPWAN Gateway. Giovanny Reyes will be responsible for power delivery applications and assisting Tariq in PCB design. Katelyn Winters will be responsible for web application development and a large segment of software development with Yannick Roberts.

Technology-in-Development	Primary	Secondary
Firmware	Yannick Roberts	Tariq Ausaf
Server	Yannick Roberts	Katelyn Winters
Back-End	Yannick Roberts	Katelyn Winters
Web Application	Yannick Roberts	Tariq Ausaf
Design (Beautification)	Katelyn Winters	Yannick Roberts
PCB (RF)	Tariq Ausaf	Giovanny Reyes
PCB (Weight Sensors)	Tariq Ausaf	Katelyn Winters
PCB (SAW Sensors)	Tariq Ausaf	Katelyn Winters
PCB (Internal Sensors)	Tariq Ausaf	Katelyn Winters
PCB (Power)	Giovanny Reyes	Tariq Ausaf
PCB (Storage)	Giovanny Reyes	Tariq Ausaf
PCB (Interfacing)	Tariq Ausaf	Yannick Roberts
LPWAN Gateway	Yannick Roberts	Giovanny Reyes
Bluetooth	Tariq Ausaf	Giovanny Reyes
Zigbee	Giovanny Reyes	Tariq Ausaf

Table 31: Membership Responsibilities

II. Milestones

In this section we present a set of smaller, week-by-week deliverables that we can feasibly accomplish together, to make significant progress toward the end goal of completed our senior design project by November. Although our senior design presentation will not be until mid-December, we would like to have our project completed in enough time for ample testing; and in case we lag the agenda presented here, we will have scheduled time to recuperate. The current week is bolded to stand out, showcasing what we have accomplished thus far, and what we still have ahead of us. Two milestone-important weeks have been bolded and italicized; the first being the week we should have the project completed by, and the second being the week in which we are expecting to present our project.

Week No.	Dates	Milestone Deliverables
1	5/13 - 5/19	Senior Design Group Formed. Preliminary Project Idea Submitted on 5/18 @ 12 pm. Purchased Senior Design Books.
2	5/20 – 5/26	Received TI MSP430 Microcontroller. Received TI Sensors Booster Pack Plug-In Module.
3	5/27 – 6/3	Received TI MSP430-EXP430G2 Microcontroller for Comms. Interfacing. Received TI CC1300x Ultra-Low Power Wireless MCU. Ordered TI CC1352R Simple Link Ultra-Low Power Dual-Band Wireless MCU.
4	6/3 – 6/9	**10-Page Initial Project Documentation (Divide & Conquer) Submitted on 6/8 @ 12 pm.** Preliminary Prototyping Work Completed on Temperature Sensor.
5	6/10 – 6/16	**First Meeting with Dr. Lei Wei, Dr. Ritchie on 6/13 @ 2 pm. Updated 10-Page Initial Project Doc. Submitted on 6/15 @ 12 pm.** Group became Familiar working with MSP430-EXP430G2, CC1300x MCU, and the sensors module and began initial coding.
6	6/17 – 6/23	15 pages of Group Contributions to 60-Page Documentation. Began estimation and ordering of parts needed for functional power requirements. Began a specific parts list and itemized (continuous) budget. Ordered Load Cells. Setup CC1300x for Bluetooth Comm., setup sensors module to track vibrations from accelerometer, humidity from humidity sensor, and temperature from temperature sensor.
7	6/24 – 6/30	20 pages of Group Contributions to 60-Page Documentation. Became familiar with common ASM pheromone sensors, vibration sensors specifications for measuring beehive communications and density of the hive. Began experimenting with load cells.

Week No.	Dates	Milestone Deliverables
8	7/1 – 7/7	15 pages of Group Contributions to 60-Page Documentation. **Second meeting with Dr. Lei Wei and Dr. Ritchie on 7/2 @ 2 pm. 60-Page Draft of Final SD1 Documentation Submitted by 7/6 @ 12 pm.** Begin integration of Load cells into prototype MCU.
9	7/8 – 7/14	25 pages of Group Contributions to 100-Page Documentation. Order more sensors (vibration sensors, ASM sensor parts, others). Continue integration of Load cells into prototype MCU.
10	7/15 – 7/21	15 pages of Group Contributions to 100-Page Documentation. **100-Page Draft SD1 Documentation due 6/20 @ 12 pm.** Begin integrating sensors into prototype MCU. Begin major PCB design.
11	7/22 – 7/28	15 pages of Group Contributions to 120-Page Documentation. Begin adding sensors support to PCB design. Integrate pheromone sensors using ASM into prototype MCU.
12	7/29 – 8/4	15 pages of Group Contributions to 120-Page Documentation. (Over) **120-Page Final SD1 Documentation due 7/28 @ 12 pm.** Finish and order first PCB.
13	8/5 – 8/11	Summer-Fall Break
14	8/12 – 8/18	Summer-Fall Break
15	8/29 – 8/25	Updates to Senior Design Project Documentation. Finish testing first PCB. Redesign components for second PCB. Setup Amazon AWS for server usage. Begin server-side software development. Begin application development.
16	8/26 – 9/1	Updates to Senior Design Project Documentation. Begin integrating new any new components into second PCB. Continue work on server-side software development and application development.
17	9/2 – 9/8	Updates to Senior Design Project Documentation. Complete and order second PCB. Continue work on server-side software development and application development.
18	9/9 – 9/15	Updates to Senior Design Project Documentation. Continue work on server-side software development and application development.
19	9/16 – 9/22	Updates to Senior Design Project Documentation. Continue work on server-side software development and application development.
20	9/23 – 9/29	Updates to Senior Design Project Documentation. Finish testing second PCB. If necessary, begin iteration for third (and final) PCB design, and have testing done in the first week of November. Continue work on server-side software development and application development.

Week No.	Dates	Milestone Deliverables
21	9/30 – 10/6	Updates to Senior Design Project Documentation. Continue work on server-side software development and application development.
22	10/7 – 10/13	Updates to Senior Design Project Documentation. Continue work on server-side software development and application development.
23	10/14 – 10/20	Updates to Senior Design Project Documentation. Continue work on server-side software development and application development.
24	10/21 – 10/27	Updates to Senior Design Project Documentation. Continue work on server-side software development and application development.
25	10/28 – 11/3	Updates to Senior Design Project Documentation. Continue work on server-side software development and application development. Have Senior Design Project Completed.
26	11/4 – 11/10	Begin Applicable field testing. Begin writing User Manual Section of Senior Design Documentation.
27	11/11 – 11/17	Continue Applicable Field Testing. Continue Writing User Manual Section of Senior Design Documentation.
28	11/18 – 11/24	Complete Applicable Field Testing. Complete Writing User Manual Section of Senior Design Documentation.
29	11/25 – 12/1	
30	12/2 – 12/8	Practice Demonstrations. Practice Presentations.
31	12/9 – 12/15	Senior Design Project Committee Presentations

Table 32: Milestones

III. Financials

I. Initial Projected Budget

In this section we present a preliminary project itemized budget. This budget includes projected costs for each technology we intend to integrate into our final device. It is important to note that these costs are preliminary and will not necessarily be accurate until we finish the design and fabrication of the final device. As such, we have overestimated the technology and power requirements, and projected cost for some of the items in this budget, such as in the solar section. We have chosen to overestimate these costs so that when fallbacks occur we will stay under budget, so as not surprise our sponsor.

In order to meet the requirement specification of a reproducible unit cost of less than $70 USD, for 5,000 units - the number of commercial beehives - we need to pay attention to the individual unit costs and unit cost breakpoints associated with suppliers that we choose to use.

Type	Item/Description	Supplier	Part No.	Unit(s)	Projected EPU ($) *	Projected Exp. ($) *
Power	Solar Cells (6.2 W)	Adafruit	1525	3	75.00	225.00
	High Efficiency Synchronous Switch Mode Charger	TI	BQ24650	3	20.00	60.00
	$LiFePO_4$ Batteries (1000 mAh)	Westinghouse	N/A	3	25.00	75.00
	Solar Panel Mount	Renology	N/A	2	15.00	30.00
	** Miscellaneous Solar Costs					150.00
Hardware Prototyping	Microcontroller including RF Comms Chip	TI	MCU: MSP-EXP430G2 Comm Chip: CC1300x RF	2	120.00	240.00
	Sensors Kit for MSP430 MCUs	TI	Multiple	2	35.00	70.00
	Single-Strand 22 Ga. Copper Wire for ICs	N/A	N/A	100 feet	0.55	55.00
	Varied Basic Electronic Components Kit (Rs, Caps, MOS, Diodes, Op Amps, etc.)	N/A	N/A	4	15.00	60.00
	**Miscellaneous Hardware Prototyping Costs					200.00
Hardware Fabrication	PCB Fabrication Stage 1	Altium	N/A	2	30.00	60.00
	PCB Fabrication Stage 2	Altium	N/A	2	30.00	60.00
	RF Comm. Chips	TI	CC1352R	4	25.00	100.00
	MCU Chips	TI	MSP432	4	12.00	48.00
	**Sensors Costs & Miscellaneous Hardware Fabrication Costs					200.00

Type	Item/Description	Supplier	Part No.	Unit(s)	Projected EPU ($) *	Projected Exp. ($) *
Software Setup	Amazon AWS Server	N/A	N/A	1	0.00	0.00
	IBM Watson AI Usage	N/A	N/A	1	0.00	0.00
	Application (Dashboard) Development	N/A	N/A	1	0.00	0.00
	** Software Hosting & Miscellaneous Development Costs					375.00
Miscellaneous	Final PCB Packaging	N/A	N/A	2	20.00	40.00
	Weatherproofing Enclosure	N/A	N/A	2	30.00	60.00
					Total Projected Cost	**2158.00**

** Projected Costs: These are not the final costs, just the best estimate we can provide with the information we have at this time. As such, actual costs in these categories may vary.*
*** Miscellaneous Costs: we are using this to account for unforeseen costs associated with incorporating these technologies. We have not fully estimated the power requirements and other items for our project, either, because it is not fully designed and fabricated yet.*

Table 33: Initial Projected Budget

Our project is being sponsored in full by **Steven Eisele**, the acting President of **Pollination US Inc**., a State-of-Florida registered commercial company. This sponsorship is contingent upon us fabricating a finished device to the marketing specifications listed in the third table of the Requirements Specifications section above. In other words, he would like us to frame our generalized Internet-of-Things Tracking/Communications System for his commercial beehive tracking needs.

This Initial Projected Budget has been approved by our sponsor, and he has agreed to spend **$2,158.00 USD** on realizing an end-product which can be produced to suit his needs. If any additional expense is required, this will be amended and approved below by our sponsor again before proceeding, to keep an openness and integrity between our team and the customer.

II. Running Tabulations (Receipts)

This section is used to keep a continuous track of where our budget is going and when. This is accomplished using Microsoft Excel and automatically updated whenever an additional item is added, or an item is amended or changed.

III. Manufacturing Quotes (@ 5,000 Units)

Upon project completion, as per our agreement to approve the Initial Projected Budget, we will get three or more quotes for manufacturing our end-product for 5,000 individual beehives. The price-per-unit should not exceed $70 USD. These manufacturing quotes will be included below.

X. Conclusion

Through the making of Plan Bee, we were challenged to be able to be flexible in our thinking when it came to mixing components. Though there is prior research in bee/hive tracking and similar communication architectures, it did not come close to our goals. What Plan Bee strived towards was the creation of a data gathering system that could be designed and integrated for further future expansion. What this project aimed towards was a way to track different health and production signs of honey bees, gather all information under a single unit while keeping room for more, and communicate this data across a large distance to a certified user through a customized digital dashboard.

Our personal reasons for creating this project were both morality, and opportunity. Honey bees are a vital part of our ecosystem and it is no secret that humanity would find it difficult to live without them. Commercially, they are also a great asset as they help the global food and crop industry with pollination. We found ourselves in a position where we can reach into the honey bee renting market and make an impact with a new design scheme. Thanks to Pollination USA for full funding and support of this project, we were honored to be able to use our resources for the betterment of the lives of the honey bees; for the betterment of the global ecosystem and its well-being.

The main design concept behind Plan Bee was a general-purpose asset tracking system through a mutual communication of connected devices. Through the creation of a Sub-1GHz Internet-of-Things template format, this project gathered data from a series of connected sensors that monitored different aspects of the honey bee hive and pool together their data into a single communication platform. The TI CC1352R1F3RGZ was a suitable component for compiling data from sensors and communication through RF. Through mesh networks and the use of technologies like Bluetooth, Zigbee, and Wi-Fi we were able to mesh together all data and send it remotely to a main hub unit that can send this information across a farther distance. With the use of Amazon AWS web services, we were able to pull together a platform that helps communicate the gathered data to a specified user.

Another important aspect of Plan Bee for both its sustainability and affordability was the use of solar power. Each main unit comes with a solar panel that charges a battery at controlled instances for powering this project. This was a suitable choice as solar power is becoming cheaper and more accessible every day. The Lithium Ion battery it charges also provides longevity and makes this product even more desirable.

After it is all completed and put together, this project carries great promise. There were challenges along the way for figuring out the best way to integrate all needed information and compile it together for a simple way a user can handle, but we were able to pool enough research and information regarding our subject matters for this project to succeed. Plan Bee can be a prominent precedent in the honey bee industry and we hope that it can also lead to influence other services and technologies.

XI. Appendix

The Appendix herein will include A1 for References, A2 for Copyright requests both satisfied and unsatisfied, as needed for content included throughout this document, A3 for relevant Beekeeping Terminology as needed for content written in the initial section within this document, Bill-of-Materials (BOM) including all line items we have needed, ordered or purchased for our part of Senior Design, and Relevant Code Snippets. Please note that Relevant Code Snippets will not be completed until we have begun firmware development in complexity, after we have completed the first pass of PCB fabrication at the very least.

I. A1. References

[1] "How much does agriculture depend on pollinators? Lessons from long-term trends in crop production.," [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2701761/.

[2] "Why We Need Bees: Nature's Tiny Workers Put Food On The Table," [Online]. Available: https://www.nrdc.org/sites/default/files/bees.pdf.

[3] "Beekeeping By The Numbers," [Online]. Available: https://americanbeejournal.com/beekeeping-by-the-numbers-2/.

[4] "Honey Bee Disorders: Honey Bee Parasites," [Online]. Available: http://www.caes.uga.edu/departments/entomology/research/honey-bee-program/bees-beekeeping-pollination/honey-bee-disorders/honey-bee-disorders-honey-bee-parasites.htmlH.

[5] "Honey Bee Parasites, Hive Pests, & Bee Diseases," [Online]. Available: https://www.uaex.edu/farm-ranch/special-programs/beekeeping/hive-pests-diseases.aspx.

[6] "Comparative Analysis of Profitability of Honey Production Using Traditional and Box Hives," [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5478371/.

[7] "Temperature and Thermoregulation in the Beehive," [Online]. Available: https://www.arnia.co.uk/temperature-and-thermoregulation-in-the-beehive/.

[8] "Hive Humidity," [Online]. Available: http://www.arnia.co.uk/hive-humidity/.

[9] "Honeybee Colony Vibrational Measurements to Highlight the Brood Cycle," [Online]. Available: http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0141926.

[10] "Sound Communication in Honeybees," [Online]. Available: https://beesource.com/point-of-view/adrian-wenner/sound-communication-in-honeybees/.

[11] "Bee Learning and Communication," [Online]. Available: https://en.wikipedia.org/wiki/Bee_learning_and_communication .

[12] "Honey Varietals," [Online]. Available: https://www.honey.com/about-honey/honey-varietals .

[13] "Internet of Things Market to Reach 1.7 Trillion by 2010," [Online]. Available: https://blogs.wsj.com/cio/2015/06/02/internet-of-things-market-to-reach-1-7-trillion-by-2020-idc/ .

[14] "BU-107: Comparison Table of Secondary Batteries," [Online]. Available: http://batteryuniversity.com/learn/article/secondary_batteries.

[15] "Wireless Devices and Health Concerns," [Online]. Available: https://www.fcc.gov/consumers/guides/wireless-devices-and-health-concerns.

[16] "Pollination: Managing Bees for Pollination," [Online]. Available: http://www.caes.uga.edu/departments/entomology/research/honey-bee-

program/bees-beekeeping-pollination/pollination/pollination-managing-bees-for-pollination.html.

[17] "Honeybees Behaviour," [Online]. Available: https://www.jscimedcentral.com/Behavior/Articles/behavior-2-1010.pdf .

[18] "Battery Types for Solar Electric Systems," [Online]. Available: https://www.solar-electric.com/learning-center/batteries-and-charging/battery-types-for-solar-electric-systems.html .

[19] "A Survey on Security and Privacy Issues," [Online]. Available: https://ieeexplore.ieee.org/document/7902207/ .

[20] "Beekeepers are Stealing Each Others Hives to Survive the Cutthroat Industry," [Online]. Available: https://qz.com/1275258/beekeepers-are-stealing-each-others-hives-to-survive-the-cutthroat-industry/ .

[21] "Ingress Protection Rating Explained," [Online]. Available: http://blog.marianinc.com/blog/ingress-protection-rating-explained.

[22] "NEMA Enclosure Standards," [Online]. Available: https://www.nema.org/Products/Documents/nema-enclosure-types.pdf.

[23] "NEMA Enclosure Types," [Online]. Available: https://www.nema.org/Products/Documents/nema-enclosure-types.pdf .

[24] "PCB R4 Guide," [Online]. Available: https://www.mclpcb.com/fr4-guide/ .

[25] "Modern Digital and Analog Communications," [Online]. Available: https://www.amazon.com/Modern-Digital-Analog-Communication-Systems/dp/0198073801.

[26] "Electromagnetic Devices," [Online]. Available: https://www.york.ac.uk/electronic-engineering/research/communication-technologies/applied-electromagnetics-devices/ .

[27] "IEEE Explore," [Online]. Available: https://ieeexplore.ieee.org/document.

[28] "GPS Systems," [Online]. Available: https://www.gps.gov/systems/gps/.

[29] "SPS Performance," [Online]. Available: https://www.gps.gov/technical/ps/2008-SPS-performance-standard.pdf .

[30] "4G LTE Telecommunications," [Online]. Available: https://en.wikipedia.org/wiki/LTE_(telecommunication).

[31] "GPRS Electronic Notes," [Online]. Available: https://www.electronics-notes.com/articles/connectivity/2g-gprs/coding.php .

[32] "Elliptic Curve," [Online]. Available: https://web.archive.org/web/20090117023500/http://www.nsa.gov/business/programs/elliptic_curve.shtml .

[33] "Elliptic Curve Cryptography," [Online]. Available: https://www.globalsign.com/en/blog/elliptic-curve-cryptography/ .

[34] "IETF," [Online]. Available: https://www.ietf.org/rfc/rfc2631.txt .

[35] "DFA NIST," [Online]. Available: https://web.archive.org/web/20131213131144/http://www.itl.nist.gov/fipspubs/fip186.htm .

[36] "Principles of Network Design," [Online]. Available: https://web.archive.org/web/20160304150203/http://ece.ut.ac.ir/Classpages/F84/PrincipleofNetworkDesign/Papers/CK74.pdf .

[37] "TCP Parameters," [Online]. Available: https://www.iana.org/assignments/tcp-parameters/tcp-parameters.xhtml .

[38] "IPxx Technet," [Online]. Available: https://technet.microsoft.com/en-us/library/bb726991.aspx?f=255&MSPPError=-2147217396 .

[39] "IP Networks," [Online]. Available: http://www.kerisys.com/keri_response/pdf/TCP_IPNetworkCommunicationPhysicalAccessControlV1.pdf .

[40] "MBS Standards," [Online]. Available: http://www.sis.pitt.edu/mbsclass/standards/viar/TCP-IP.html .

[41] "UDP Overview," [Online]. Available: http://www.tcpipguide.com/free/t_UDPOverviewHistoryandStandards.htm.

[42] "NanoIP," [Online]. Available: http://www.cwc.oulu.fi/nanoip/index.html .

[43] "CoAP," [Online]. Available: http://coap.technology/.

[44] "Portals DDS: The Proven Connectivity Standard for IOT," [Online]. Available: http://portals.omg.org/dds/.

[45] "Simple Object Access Protocol (SOAP) v1.2," [Online]. Available: https://w3.cs.jmu.edu/bernstdh/web/common/lectures/slides_soap.php.

[46] "2413 - Standard for an Architectural Framework for the Internet of Things (IoT)," [Online]. Available: https://standards.ieee.org/develop/project/2413.html.

[47] "Web Sockets," [Online]. Available: https://html.spec.whatwg.org/multipage/web-sockets.html#network.

[48] "Weave," [Online]. Available: https://www.codeaurora.org/project/weave.

[49] "JSON," [Online]. Available: http://json.org/.

[50] "What is JSON?," [Online]. Available: http://www.secretgeek.net/json_3mins.

[51] "XMPP," [Online]. Available: https://xmpp.org/.

[52] "XMPP IoT," [Online]. Available: https://xmpp.org/uses/internet-of-things.html.

[53] "SQL," [Online]. Available: http://www.sqlcourse.com/welcomead?_qstu=%2Fintro.html.

[54] "NoSQL Database," [Online]. Available: http://nosql-database.org/.

[55] "Yahoo Cloud Serving Benchmark," [Online]. Available: https://blogs.oracle.com/bestperf/yahoo-cloud-serving-benchmark%3a-sparc-s7-2-and-oracle-nosql-advantage-over-x86-e5-2699-v4-server-per-core-under-load.

[56] "HTML5 Introduction," [Online]. Available: https://www.w3schools.com/html/html5_intro.asp.

[57] "HTML5 Responsive Web Design," [Online]. Available: https://msdn.microsoft.com/en-us/magazine/hh653584.aspx?f=255&MSPPError=-2147217396.

[58] "CSS: Cascading Style Sheets," [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/CSS.

[59] "JavaScript," [Online]. Available: https://www.javascript.com/.

[60] "Developing in HTML with Javascript and CSS3 Jump Start," [Online]. Available: https://mva.microsoft.com/en-us/training-courses/developing-in-html5-with-javascript-and-css3-jump-start-8223?l=lCnp5kIy_5104984382.

[61] "NEMA Enclosure Standards," [Online]. Available: https://www.nema.org/Products/Documents/nema-enclosure-types.pdf .

II. A2. Copyright Requests

Katelyn Winters 11 days ago

Hello! my name is Katelyn Winters and I am an Electrical Engineering student at UCF. I am in my final year of classes and am currently taking a Senior Design class where my group and I have to come up with a project and write a research paper about it. I found you and you team's research paper on Bee colony temperature monitoring and thought it was extremely helpful! I was hoping to include some writings about your work and an image of one of your proposed system architectures as well. I need to request the image from the source before I am able to include it. I am writing to you to request this image. Please let me know if the image is okay to use! i have included the image for clarification. Thank you for your time.
Sincerely,
Katelyn Winters

SD1_ex1.PNG

Aleksejs Zacepins to you 5 days ago

Hi.
Thank you for contacting me. It is great that you found our work interesting.
Yes, you can use our images, of course making reference.
Wish you luck writing your paper.

Aleksejs.

III. A3. Beekeeping Terminology

A **Beehive** (or **Colony**) is collection of bees making residence, around the location of the queen bee's choosing. A colony generally contains a single queen bee (a fertile female), a few thousand drone bees (fertile males) and tens of thousands of worker bees (sterile females). The **Queen Bee** is the "leader" of the hive. The **Drone Bees** – while they do not have stingers – guard the hive and mate with the queen bee. The **Worker Bees** clean the hive, feed the larvae, and retrieve pollen.

A collection of beehives in a single area is called an **Apiary** (or **Yard**). An **Apiarist** (or **Beekeeper**) is someone who looks after the beehives in an Apiary.

A **bee brood** (or **hive brood**, or **brood nest**) is a state of the hive when a new queen bee, new worker bee, or drone bee eggs are being laid by the current queen bee and are in various stages of development up to and including the larvae stage.

Swarming occurs when a new colony is formed away from the original. The queen bee signals and leaves with a large group of worker bees, taking them with her in starting her new colony.

In honeybees, **trophallaxis** is the exchange of food or honeybee pheromones to communicate information about the resources available, bees and beehive.

Honey Varietals describe the assorted flavors of honey honeybees produce based upon the climate (temperature and rainfall), and flowers blooming which the honeybees had visited.

IV. A4. Bill of Materials (BOM)

The Bill of Materials (BOM) presented here is an initial estimate of all of the products, parts, components, and otherwise that we will need to complete our project in the required timeframe for Senior Design II in the Fall 2018 Semester. The table is found below.

Status	CPU	Part	Quantity	Description	Manufacturer
Development Boards and Kits					
Received	40.00	LAUNCHXL CC1352R1	2	WMCU Dev Board	Texas Instruments
Received	40.00	296-35762-ND	2	Sensor Dev Board	Texas Instruments
Received	45.55	BBB01-SC-505-ND	2	BeagleBone Black Dev.	GHI Electronics
Waterproof Enclosures					
Received	4.00	IM171017001	5	IP66 Sonoff Waterproof Enclosure	Sonoff
User Interface Components					
Received	2.00	SSD1306	8	OLED Display	AdaFruit Industries
Received	0.14	150060RS75000	100	Red LED (0603)	Wurth Electronics
Received	0.14	150060VS75000	100	Green LED (0603)	Wurth Electronics
Received	0.14	150060BS75000	100	Blue LED (0603)	Wurth Electronics
Received	0.26	TC33X-2-201GG	100	200 Ohm Trimmer Resistor (Potent.)	Bourns
Received	0.46	SKQGADE010	100	Button	ALPS
WMCU and RF Components					
Received	2.89	SMA-10V21-TGG	5	SMA Fem. Connector	HUS-TSAN
Received	7.50	WRL-09143	3	2.4 GHz Antenna Duck	SparkFun Electronics
Received	16.40	SAM-M8Q-0-10	3	1.56 GHz Antenna GPS	U-Blox America
Received	1.39	XMSSJJ3G0PA-054	5	RF Switch IC	Murata

Status	CPU	Part	Quantity	Description	Manufacturer
Received	3.20	SE4150L	5	GPS Receiver	SkyWorks
Received	1.29	PE4259	10	RF Switch IC	Peregrine
Received	6.85	CC1352R1F3RGZ	25	WMCU IC	Texas Instruments
Received	15.23	237-M10578-A2	3	GPS Module	Antennova M2M
Received	0.76	MX25R8035FZUIL0	10	8Mb NOR Flash Memory IC	Macronix
Received	0.66	CX2016DB48000C0 FPLC1	10	48 MHz XTAL Res.	Kyocera
Received	0.27	FC-135 32.7680KA-AG0	10	32.8 kHz XTAL Res.	Epson
Power Components					
Received	3.34	BQ24650RVAT	10	Solar Charger	Texas Instruments
Received	0.39	MC33063ADR	10	Switching Volt. Reg.	Texas Instruments
Received	4.09	AM-7E04CAR-ND	3	12V 1.5W Solar Pan.	Panasonic BSG
Received	0.79	TPD6E004RSERG4	20	ESD Suppressor	Texas Instruments
Received	1.56	FTSH-105-01-F-DV-K	20	XDS Header	Samtec
Received	0.16	M50-3930242	100	2-Pin Header (Will be used to make 4- and 6-pin headers)	Harwin
Received	0.05	BLM18HE152SN1	100	Ferrite Bead (0603)	Murata
Sensor Components					
Received	0.46	HDC2010	10	Humidity and Moisture Sensor	Texas Instruments
Received	9.32	MPU-9250	3	9-axis IMU	TDK InvenSense
Received	2.34	TMP006	10	Temperature Sensor	Texas Instruments
Received	3.19	ISL29023	10	Light Sensor	Renesis/Intersil

Status	CPU	Part	Quantity	Description	Manufacturer
Received	7.75	SHT21	10	Humidity and Temp. Sensor	Sensirion AG
Received	1.20	BMP180	10	Pressure Sensor	Bosch Sensortec
Basic Components					
Received	0.022	GRM0225C1ER50BA03L	1000	0.5 pF Capacitor (0402)	Murata
Received	0.024	GRM0225C1E1R1BA03L	1000	1.1 pF Capacitor (0402)	Murata
Received	0.016	GRM0225C1E2R7CA03L	1000	2.7 pF Capacitor (0402)	Murata
Received	0.016	GRM0225C1E3R0CA03L	1000	3 pF Capacitor (0402)	Murata
Received	0.016	GRM0225C1E3R6CA03L	1000	3.6 pF Capacitor (0402)	Murata
Received	0.024	GRM0225C1E6R2BA03L	1000	6.2 pF Capacitor (0402)	Murata
Received	0.024	GRM0225C1E120GA03L	1000	12 pF Capacitor (0402)	Murata
Received	0.024	GRM0225C1E220GA02L	1000	22 pF Capacitor (0402)	Murata
Received	0.016	GRM0225C1E101JA02L	1000	100 pF Capacitor (0402)	Murata
Received	0.017	GRM0225C0J111JA02L	1000	110 pF Capacitor (0402)	Murata
Received	0.020	GRM022R71A102KA12L		1 uF Capacitor (0402)	Murata

Status	CPU	Part	Quantity	Description	Manufacturer
Received	0.016	GRM022R60J222KE19L	1000	2.2 uF Capacitor (0402)	Murata
Received	0.016	GRM022R60J103ME19L	1000	10 uF Capacitor (0402)	Murata
Received	0.025	GRM022R60J223KE15L	1000	22 uF Capacitor (0402)	Murata
Received	0.010	GRM033C80J223ME01D	1000	22 uF Capacitor (0603)	Murata
Received	0.027	GRM022R60G473ME15L	1000	47 uF Capacitor (0402)	Murata
Received	0.040	LQG15HS1N5C02D	1000	1.5 nH Inductor (0402)	Murata
Received	0.047	LQG15HZ2N0S02D	1000	2 nH Inductor (0402)	Murata
Received	0.034	LQG15HS2N4S02D	1000	2.4 nH Inductor (0402)	Murata
Received	0.034	LQG15HN7N5J02D	1000	7.5 nH Inductor (0402)	Murata
Received	0.040	LQG15HN27NH02D	1000	27 nH Inductor (0402)	Murata
Received	0.047	LQG15HZ4N7S02D	1000	47 nH Inductor (0402)	Murata
Received	0.037	LQG15HS68NJ02D	1000	6.8 uH Inductor (0402)	Murata
Received	0.110	MLF2012E6R8KT000	1000	6.8 uH Inductor (0805)	TDK
Received	0.010	CR0402-FX-10R0GLF	1000	10 Ohm Resistor (0402)	Bourns

Status	CPU	Part	Quantity	Description	Manufacturer
Received	0.005	CR0402-JW-300GLF	1000	30 Ohm Resistor (0402)	Bourns
Received	0.010	CR0402-FX-1000GLF	1000	100 Ohm Resistor (0402)	Bourns
Received	0.009	CR0402-FX-1800GLF	1000	180 Ohm Resistor (0402)	Bourns
Received	0.010	CR0402-JW-201GLF	1000	200 Ohm Resistor (0402)	Bourns
Received	0.010	CR0402-FX-2200GLF	1000	220 Ohm Resistor (0402)	Bourns
Received	0.009	CR0402-FX-2611GLF	1000	1k Ohm Resistor (0402)	Bourns
Received	0.003	CR0402-JW-182GLF	1000	1.8k Ohm Resistor (0402)	Bourns
Received	0.008	CR0402-JW-222GLF	1000	2.2k Ohm Resistor (0402)	Bourns
Received	0.024	CR0402-FX-2262GLF	1000	2.6k Ohm Resistor (0402)	Bourns
Received	0.010	CR0402-FX-1332GLF	1000	3.3k Ohm Resistor (0402)	Bourns
Received	0.008	CR0402-JW-512GLF	1000	5.1k Ohm Resistor (0402)	Bourns
Received	0.009	CR0402-FX-5622GLF	1000	6.2k Ohm Resistor (0402)	Bourns
Received	0.010	CR0402-FX-1022GLF	1000	10k Ohm Resistor (0402)	Bourns

Status	CPU	Part	Quantity	Description	Manufacturer
Received	0.008	CR0402-JW-104GLF	1000	100k Ohm Resistor (0402)	Bourns