A Wireless Sensor Node Powered by a PV/SuperCapacitor/Battery Trio

The Edward S. Rogers Sr. Department of Electrical and Computer Engineering
University of Toronto

ECE496Y Design Project Course - 2012
Group Final Report

Team Members:
Matthew Martino  matthew.martino@mail.utoronto.ca
Jordan Varley  j.varley@utoronto.ca

Supervisor:
Doctor Olivier Trescases  ot@ele.utoronto.ca

Administrator:
John Taglione
Executive Summary

The subject of our ECE496 project is Wireless Sensor Nodes (WSN), which are small devices that collect and transmit data, and are often placed in remote areas with the capability to extract energy from ambient sources to last long periods of time. The goal of the project is to produce such a WSN that will make use of a unique energy management scheme to ensure long-lasting operation. A solar panel will be the primary energy source, with secondary energy storage devices being a rechargeable battery and super capacitor.

Ultimately, the WSN must be able to collect particular environmental data using sensors. In this case the WSN will be utilized in a Smart Garden application where temperature, moisture, humidity and light intensity data relevant to a garden will be transmitted to a nearby base station. GPS data will also be collected to track the location of individual WSNs. Efficient energy collection from the solar panel using a Maximum Power Point Tracker is required, as is the capability to recharge secondary energy storage components. Continuous autonomous operation is key, especially during the night, and in sub-optimal lighting conditions. Lastly, a minimum size of 6 inches x 4 inches has been specified for the size of the device, along with a desired upper limit for the instantaneous solar power available of 200mW.

The WSN was implemented using 3 key modules: Energy Collection, Energy Storage and Delivery, and Energy Consumption. The collection module consists of the solar array and Maximum Power Point Tracking hardware, which extracts as much energy as possible from the solar array and delivers it to the Storage and Delivery Module. The Consumption module contains the overhead components used to implement the smart garden application and manage power flow throughout the system. Power flow is configured throughout the WSN to manage the available energy resources, charging/discharging of the storage devices as well as limit the smart garden functionality in low-energy situations.

Upon completion of the WSN design, a number of key achievements were made including lower-than-expected current consumption in the power management overhead components, accurate current regulation within the targeted requirement of +/- 5% as well as higher-than-targeted converter efficiency. Additionally, the complete smart garden application has been implemented with capabilities to measure geographical and environmental data, store and transmit the data via XBEE radio, and operate in sleep mode when measurements are not being made. Finally, the WSN features the ability to harvest available solar energy from the outdoor environment and deliver this energy to the super capacitor and battery storage devices. A long-term test (nine days) was completed, and its results coupled with an analysis of the WSN power consumption strongly demonstrate the WSN’s potential to operate autonomously for months.
Group Highlights & Individual Contributions

Throughout the WSN’s entire development cycle, significant effort was invested to carefully select, design and implement the WSN components to ensure the overall success of the project. A particular emphasis was placed on ultra-low power design (sub-milliwatt levels) and efficient energy conversion/utilization. With the large up-front investment during the design phase in Fall 2011, a number of key requirements were achieved by the WSN design upon completion in March 2012. These accomplishments include:

- A custom-designed DC/DC converter to interface a lithium-ion battery and super capacitor that achieves at least 82.33% efficiency across all operating conditions and consumes less than 470 uA of current to supply all required overhead components (CPLD, ADC)
- A successful implementation of the WSN’s target application (smart garden) including measurement and XBEE transmission of positional data (GPS), soil moisture, ambient temperature and light, and relative humidity with less than 900 uA of current consumption during measurements and 10 uA when in sleep mode
- The ability to harvest available solar energy using an array of panels that total 75 mW and store retrieved energy in a 5F super capacitor and 350 mAh lithium-ion battery
- A power management logic implementation using a CPLD operating on a system clock of 256 kHz for minimized overhead power consumption

The majority of the WSN’s requirements have been delivered upon completion of the project and the delivery of these hinged on a strong design foundation and effort invested during the problem definition and system architecture stages. During the development stage, throughout the latter half of the WSN design cycle, few technical issues arose at either the module level or system integration level, further illustrating the strong design foundation the team had built during the first semester. With the single major technical issue that did arise, relating to current delivery using a PFM control scheme, the team came together to understand the problem, identify a fix and drive the project forward to minimize the impact of the week’s delay introduced by this issue. No single module had failed to perform as designed nor introduced any system-level integration issues, resulting in a smooth development cycle through the latter half of the project. From the outset of the project, the team would have liked to run a long-term system test spanning a number of weeks or months, however due to timing constraints and project set-backs, the extended-duration test has been shortened to span about a week with the results of this test to be presented at the design fair.

Overall, the project is deemed a success and the challenges presented by such a project have allowed the team to grow on a personal and professional level.


**Individual Contributions - Jordan Varley**

During the span of both the WSN design and development phases, I have had the opportunity to contribute to a number of different aspects the project touches upon, including:

- System Architecture for the WSN target application and energy storage/management features
  - Power converter topologies and configurations
  - Energy storage devices and interconnection between these devices
  - Sensor placement for power management and smart garden application

- Basic Analog Design for data measurement and sensor signal chains used by the WSN
  - Design and placement of necessary gain stages, analog multiplexers and ADC components

- Component selection for the CPLD, Microcontroller and support components to implement power management and target application functionality

- Completion of Schematics for the WSN design

- Digital Design using Verilog and the system CPLD to implement:
  - PFM Current Regulator for the custom DC/DC Converter
  - Reference Voltage Generation for the MPPT
  - Measurement of Critical System Voltages and Currents
  - Power Management Logic to manage energy resources available to the WSN
  - Communication with the microcontroller for debug and system operation

- Board Development, Debug, Testing and Verification of:
  - Sensor signal chain for the Voltage and Current information supplied to the CPLD
  - DC/DC Converter operation in open and closed loop environments
  - PFM Regulator Performance
  - System Power Flow Operation and Performance
  - Solar Boost Converter Operation and Performance

The contributions I have made throughout the design of our WSN range across the entire development cycle and include stages in problem definition, architecture, design and testing/debug. The tasks I have been able to contribute to have had a large impact in the success of the WSN design and lead to the successful conditions assigned to a number of the project requirements, as illustrated in later sections of this document. Due to the extensive breadth and depth requirements of the WSN design, a number of excellent opportunities have been provided to further my technical knowledge throughout the course of project and I have found this to be a very rewarding experience overall.
Individual Contribution - Matthew Martino

During the course of the WSN design, build-up, and testing I contributed to a variety of activities, including:

- **Architecture and Module Design**
  - Energy storage device and power converter arrangement.
  - Performed the design work for the custom PFM converter
    - Generated parameters to be used for the current regulator, and determined component values.
  - Contributed to the power management logic prior to its implementation in the CPLD.
  - Selected the passive components used in all four power converters, along with a current sensor, parallel ADC, voltage references, and other devices.

- **PCB Layout**

- **Creation and testing of the Smart Garden application through software and hardware development**
  - Involved microcontroller programming using C to accomplish the following tasks:
    - Enable the Smart Garden sensors and take measurements using the ADC.
    - Develop functionality that allows the microcontroller to measure on-board electrical data.
    - Communicate with a GPS to collect location coordinates using the UART.
    - Communicate with a Xbee radio using the UART to transmit all data wirelessly.
    - Establish a communication link between the microcontroller and CPLD.
    - Identify methods of reducing power consumption within the microcontroller by reducing clock speed when possible, and entering sleep mode.
    - Implement a top-level algorithm that collects and transmits the aforementioned data at specific intervals using timers, taking into account the current energy status of the WSN.
  - Involved verification of the signal flow path between the sensors and microcontroller, namely through the analog multiplexer and op-amp.
  - Verified the functionality of the Smart Garden sensors.

- **Board build-up through soldering and general debugging**
  - Quickly learned useful techniques for soldering, and contributed fairly to PCB build-up.

- **Power Flow Testing**
  - Tested and characterized the off-the-shelf converters that regulate the 3.3V and 1.8V rails.
    - Measured device efficiency, verified the output regulation, and performed load step tests.
  - Assisted in part to the testing of the implemented custom PFM converter.
    - Parameters generated during the design stage required verification and adjustment.

- **Testing the Solar Boost Converter**
  - Selected the actual device, and tested a sample on a breadboard in the fall.
  - Assisted in the verification of the part on the PCB in the winter.

- **Conducted power measurements of the fully integrated system.**

This project offered many challenges and required knowledge of a diverse set of fields in electrical engineering including: hardware design, software design, energy systems, simple control principles, communication, and the process of PCB development. Great dedication was required to ensure that the correct steps were taken during each phase of the project. Ultimately, I am proud of the work that my team and I put forward, and the accomplishments we achieved.
Acknowledgements

Our team would like to acknowledge the contributions made by our Administrator John Taglione, our Supervisor Professor Olivier Trescases, and some of the graduate students supervised by Professor Trescases including Shahab Poshtkouhi, Pete Scourboutakos, and Soren Massoumi.

John Taglione provided us with guidance and feedback throughout the year through the evaluation of the documents we submitted. Professor Olivier Trescases met with us on a weekly basis to review our progress, answer our questions, and offer suggestions. He also made his lab stations available for use, providing us with the equipment required to complete the project. Shahab, Pete, and Soren provided us with some software IP to use as a base for our own development. The C-code and LabVIEW VI they provided allowed us to quickly learn the basics of microcontroller programming and GUI creation, so that we could more easily develop our own material.
Texas Instruments Parts Descriptions

**BQ25504 - Ultra Low Power Boost Converter with Battery Management for Energy Harvester Application**

The BQ25504 is a brand new part from TI that we used to transfer power from a sub-200mW solar array to a 5-Farad Super Capacitor. Incidentally, when our project began in September this part did not exist so we planned on implementing a discrete converter ourselves; while designing that module a simple google search yielded this device, which modified our plans. The BQ25504 appealed to us for four reasons:

1. It performed input voltage regulation to interface with a solar array.
2. It allowed an external reference input voltage to be provided, allowing us to perform Maximum Power Point Tracking according to our preferred procedure.
3. It had over-voltage checking to limit the output voltage magnitude.
4. It operated at voltage levels that were compatible with our already-selected components: < 2V solar array at the input, and < 5V Super Capacitor at the output.

The benefits of the BQ25504 were two-fold. First, this module ensured effective solar energy collection, required for long-lasting operation. Having ordered and tested samples during the design stage, we felt confident that the BQ25504 would perform as required to provide energy to the WSN. Second, acquiring an off-the-shelf IC to implement this module allowed us to focus on other design requirements. Implementing this module from scratch would have been extremely time consuming, jeopardizing the project’s success. Overall we are pleased with this device. A more detailed description of the BQ25504 in our design may be found on page , with testing results on page .

**TPS62203 - High Efficiency, SOT23 Step-Down, DC-DC Converters (3.3V Fixed)**

The TPS62203 is a buck converter with fixed 3.3V output that transfers energy from the Super Capacitor to almost all of the other devices on the WSN. This converter regulates the main 3.3V rail without which the WSN would not operate. The TPS62203 possessed the following useful qualities:

1. The ability to maintain high efficiencies at low output currents through Power Save Mode (PFM switching); a mode in which we operate about 98% of the time.
2. The ability to transition between Power Save Mode and regular PWM switching during high-power operations.
3. Easy to layout on a PCB, and easy to test.

Power Save Mode was particularly useful because we are operating on the miliWatt scale. Overall we were quite content with the performance of this device. A more detailed description of this device in our design may be found on page , with testing results on page .
**TPS62202 - High Efficiency, SOT23 Step-Down, DC-DC Converters (1.8V Fixed)**

The TPS62202 is a buck converter with fixed 1.8V output that powers the CPLD core from the Super Capacitor. This converter allowed us to operate our CPLD at a lower voltage to save power. The TPS62203 possessed the following useful qualities (similar to the TPS62203):

1. The ability to maintain high efficiencies at low output currents through Power Save Mode (PFM switching); a mode in which we operate about 100% of the time for this device.
2. Easy to layout on a PCB, and easy to test.

Power Save Mode was particularly useful because we are operating on the microWatt scale. Again we were quite content with the performance of this device given the low power levels. A more detailed description of this device in our design may be found on page , with testing results on page.

**SN74LVC1T45 - Single-Bit Dual-Supply Bus Transceiver**

The SN74LVC1T45 might diverge from the eligibility criteria for this contest, but we include it here anyways because of the unique and important application for which it was used in our design. The description for this device on the Texas Instruments website states that: ‘The SN74LVC1T45 is designed for asynchronous communication between two data buses.’* However, we found better use of the device as a gate driver in our custom bi-directional non-inverting buck-boost converter. Four level shifters were used to drive P/N-Mos Switches from the CPLD. The CPLD IO’s operate on 3.3V logic. The power converter connects the Super Capacitor and battery, therefore those voltages drove the output of the shifted signals.

Low power design was the focus of the project, therefore mosfets with small gate charge were selected in our custom converter (on the order of hundreds of picoCoulombs). However, proper gate driving was still required at voltages above the CPLD IO voltages. The converter transfers less than 100mW, meaning that typical gate drivers do not exist for this power level. Our only alternative was these level shifters, justified by the small gate charge of the mosfets (the driving current is ~10mA). Quite simply, without these level shifters the converter could not have been developed according to the goals and specifications of the WSN. Using the SN74LVC1T45 in a switched mode power supply, rather than for communication interfacing, demonstrates its unique role in our project. A more detailed description of the module that uses this device may be found on page , with testing results on page.

*http://www.ti.com/product/sn74lvc1t45
# Table of Contents

**Introduction** ........................................................................................................................................... 2  
  Background and Motivation ......................................................................................................................... 2  
  Project Goal .................................................................................................................................................. 3  
  Project Requirements ................................................................................................................................. 3  

**Technical Design** ................................................................................................................................... 5  
  System Level Overview ............................................................................................................................... 5  
  Module-Level Descriptions ......................................................................................................................... 7  
    *PV Array* .................................................................................................................................................. 7  
    *Solar Boost Converter with MPPT Functionality* ..................................................................................... 7  
    *Energy Storage and Delivery Module* ..................................................................................................... 9  
    *Logic Devices: CPLD and Microcontroller* ......................................................................................... 13  
    *Smart Garden Application: Sensors, GPS, Transmitter* .................................................................... 21  
  Assessment of Proposed Design .................................................................................................................. 22  

**Testing and Verification** .......................................................................................................................... 24  
  Requirement F1 - Regulating 3.3V .............................................................................................................. 24  
  Requirement F2 - Regulating 1.8V .............................................................................................................. 26  
  Requirement F3 - Collecting Environmental Data .................................................................................... 27  
  Requirement F4 - Collecting On-Board Electrical Data .......................................................................... 31  
  Requirement F5 - Wireless Data Transmission ......................................................................................... 33  
  Requirement F6 - Flexible Data Collection and Transmission Rates ..................................................... 35  
  Requirement F7 - Programming the Logic Devices .................................................................................. 35  
  Requirement F8 - Microcontroller - CPLD Communication .................................................................... 36  
  Requirement F9 - Implement a Maximum Power Point Tracker ......................................................... 38  
  Requirement F10 - Implement a DC/DC Converter for the Solar Array .............................................. 39  
  Requirement F11 - Implement a DC/DC Converter that Transfers Energy amongst the Storage Devices 41  
  Requirement F12 - Recharge the Devices with Surplus Solar Energy .................................................. 48
Requirement F13 - Execute a power management program to maximize the WSN’s operation .............. 49
Requirement O1 - Sub-200mW Solar Array .................................................................................................. 53
Requirement C1 - Measure no larger than 6 inches x 4 inches .................................................................. 53
Requirements C2, C3 - Use a Super Capacitor and Rechargeable Battery .............................................. 53
Requirement C4 - Day and Night Outdoor Operation ................................................................................ 53
Requirement C5 - Operate autonomously once deployed ........................................................................... 53
Requirement C6 - Remain operational for no less than 7 days .................................................................. 53

Power Consumption Analysis .................................................................................................................. 55

Summary and Conclusions ....................................................................................................................... 56

Future Work and Improvements ............................................................................................................ 58

References ................................................................................................................................................ 59

Appendix A - Gantt Chart History ........................................................................................................... 61
Appendix B – PCB Schematic and Layout .................................................................................................. 64
Appendix C – Expenses .............................................................................................................................. 71
Appendix D - Bi-Directional DC/DC Converter Design Work ................................................................. 72
Appendix E - Microcontroller Top-Level Algorithm ................................................................................ 82
Appendix F - Communication Bus Command Details .............................................................................. 84
Appendix G - Power Gating ....................................................................................................................... 85
Appendix H – Solar Panel Background ...................................................................................................... 86
Appendix I – Power Management Logic Chart ....................................................................................... 87
Appendix J – Power Measurement Captures ............................................................................................ 88
Appendix K – Long-Term Test Results ..................................................................................................... 89
**Introduction**

This report summarizes and highlights the intent, design, testing, and success regarding the development of a Wireless Sensor Node powered by Solar Panels, a Super Capacitor, and a Battery. It first outlines our desire to create a device that uses renewable energy efficiently to prolong its operating time. It then lists the requirements for such a device to achieve that goal, and describes the final design that implements those requirements. What follows is a presentation of the degree to which the requirements are fulfilled through our Wireless Sensor Node. The report concludes with comments on our accomplishments.

**Background and Motivation**

The focus of our project is on Wireless Sensor Nodes (WSN's), which are small devices that collect and transmit data about the surrounding environment, such as: ambient temperatures, air pressure values, or wind velocities. These devices rely on ambient energy sources, including solar energy, and wind energy [1]. The usefulness of WSN's is their ability to survive for long periods of time, which requires careful management of the collected environmental energy. Currently, WSN's are commercially available from companies like Texas Instruments, which sells full Energy Harvesting Development Kits [2]. Companies such as Libelium [3] develop WSN's for purposes such as radiation-level detectors, and pollution detectors.

The motivation for this project lies in the exciting opportunity to create a device that relies solely on renewable solar energy, and to develop a strategy for managing that energy to extend the lifetime of the WSN. Our team has selected a Smart Garden use case where moisture, humidity, temperature, and light intensity data for a garden can be collected. A GPS is also included so that multiple units can be tracked. However, the focus is to manage and use low amounts of power to produce a long-lasting device. It is difficult to fully differentiate our proposed WSN with what is already available from several successful companies; one such group has even created a WSN for agricultural applications to monitor crop status, environmental data, and possible threats from pests [4]. From our research though, our team noticed that some of the WSNs developed by other academic research teams used photovoltaic cells that generate power on the order of 100-500mW [5]. An added challenge for the project is to see if we can develop a WSN that is long-lasting but uses solar panels that generate around 100-200mW of power. We also attempt to exploit the favourable characteristics of two distinct energy storage devices: a super capacitor, which has a high power density, and a rechargeable battery, which has a high energy density. We hope to harness the advantages of both of those devices, along with the energy collection abilities of the solar panels, to ultimately lengthen the WSN operation time.
**Project Goal**

Our project goal is to create a wireless sensor node that uses a unique power management scheme to coordinate power storage and delivery between a small solar array, and energy storage devices, ensuring long-lasting operation.

**Project Requirements**

*Functions*

The wireless sensor node shall perform the following functions:

1. Regulate a main-rail voltage to 3.3V within +/- 5%, at a minimum of 85% efficiency.
2. Regulate a sub-rail voltage to 1.8V within +/- 5%, at a minimum of 85% efficiency.
3. Collect environmental data including: soil moisture, humidity, temperature, light intensity, and geographical coordinates.
4. Collect on-board electrical data including: solar panel open-circuit voltage, battery voltage and current, and super capacitor voltage.
5. Transmit the collected data wirelessly to a nearby computer over a distance of 50m.
6. Allow changes to data collection and transmission rates, through software, prior to deployment.
7. Program the microcontroller and CPLD devices for testing and final operation.
8. Allow the on-board microcontroller and CPLD to communicate with each other.
9. Implement a Maximum Power Point Tracker for the photovoltaic cells operating at a minimum of 80% accuracy for Maximum Power Tracking.
10. Implement a DC/DC Converter to regulate the output voltage of the solar array as dictated by the MPPT. The converter should operate at a minimum of 80% efficiency, and regulate the voltage within a +/- 2% tolerance.
11. Implement a DC/DC Converter that regulates current through Pulse Frequency Modulation to transfer energy amongst the energy storage devices. The converter itself should operate at a minimum of 80% efficiency, and regulate the current within a +/- 5% tolerance.
12. Recharge the energy storage devices with surplus solar energy.
13. Execute a power management program to maximize the operational lifetime of the WSN.

*Objectives*

The wireless sensor node should exhibit the following objectives:

1. Use any number of solar panels of any rating, so long as the instantaneous power generated by the solar array is no more than 200mW.

*Constraints*

The wireless sensor node must adhere to the following constraints:

1. Measure no larger than 6 inches x 4 inches in size.
2. Use a super capacitor as an energy storage device.
3. Use a rechargeable battery as an energy storage device.
4. Operate outdoors during the day under varying lighting conditions and during the night.
5. Operate autonomously once deployed.
6. Remain operational for no less than 7 days.

Note that the wireless sensor node will exist in the form of a printed circuit board with all necessary components attached by means of solder, screws, or external connectors.
**Project Requirements Status Summary**

Below is a table indicating the status of the aforementioned functions, objectives, and constraints.

Table 1. *Project Requirements Status Summary*

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Description</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>Regulate 3.3V</td>
<td>Fully Completed and Tested.</td>
</tr>
<tr>
<td>F2</td>
<td>Regulate 1.8V</td>
<td>Completed and tested with Conditions.</td>
</tr>
<tr>
<td>F3</td>
<td>Collect Environmental Data</td>
<td>Not yet Tested to Completion</td>
</tr>
<tr>
<td>F4</td>
<td>Collect Electrical Data</td>
<td>Partially Completed</td>
</tr>
<tr>
<td>F5</td>
<td>Transmit Data</td>
<td></td>
</tr>
<tr>
<td>F6</td>
<td>Data Frequency</td>
<td></td>
</tr>
<tr>
<td>F7</td>
<td>Program logic devices.</td>
<td></td>
</tr>
<tr>
<td>F8</td>
<td>uC &amp; CPLD communication.</td>
<td></td>
</tr>
<tr>
<td>F9</td>
<td>MPPT</td>
<td></td>
</tr>
<tr>
<td>F10</td>
<td>Solar DC/DC</td>
<td></td>
</tr>
<tr>
<td>F11</td>
<td>Battery DC/DC</td>
<td></td>
</tr>
<tr>
<td>F12</td>
<td>Solar Recharging</td>
<td></td>
</tr>
<tr>
<td>F13</td>
<td>Power Management Program</td>
<td></td>
</tr>
<tr>
<td>O1</td>
<td>Sub 200mW Solar Array.</td>
<td></td>
</tr>
<tr>
<td>C1</td>
<td>6 x 4 inch size.</td>
<td></td>
</tr>
<tr>
<td>C2</td>
<td>Use a Super Capacitor</td>
<td></td>
</tr>
<tr>
<td>C3</td>
<td>Use a Rechargeable Battery</td>
<td></td>
</tr>
<tr>
<td>C4</td>
<td>Operate during day and night.</td>
<td></td>
</tr>
<tr>
<td>C5</td>
<td>Operate Autonomously.</td>
<td></td>
</tr>
<tr>
<td>C6</td>
<td>Operate longer than 7 days.</td>
<td></td>
</tr>
</tbody>
</table>

*Legend*

- **Fully Completed and Tested.**
- **Completed and tested with Conditions.**
- **Not yet Tested to Completion**
- **Partially Completed**

*The extra details regarding requirements that are not 'green' are flushed out in the Testing and Verification section.*
Technical Design

System Level Overview

Figure 1 depicts a block diagram of the most general modules and connections developed. The solid black arrows represent power flow connections, whereas the dashed black arrows represent data flow connections.

Figure 1. Higher-level blocks required for the Wireless Sensor Node.

The DC/DC Converter with Maximum Power Point Tracking (MPPT) Functionality interfaces with the Photovoltaic Array to extract solar energy. The Component Connection Scheme contains circuitry that will charge and discharge the energy storage devices. The output of the Component Connection Scheme powers devices on the right, namely the sensors used to collect data, the transmitter, the GPS, and the controlling units. Information is also transferred amongst the sensors, transmitter, and logic units. Figure 2 illustrates the more specific System Block Diagram, classifying the blocks according to their relation to energy:

Figure 2. More specific blocks required for the Wireless Sensor Node.
Below is a top-level view of the WSN with its main sections labeled; another picture is included.

Figure 3. The implemented WSN with labeled sections.

Figure 4. The implemented WSN.
Module-Level Descriptions

This section contains a description of the main modules required to implement the WSN. The areas of focus are: the PV Array, the Solar Boost converter, the Energy Management Module consisting of three DC/DC converters, the Sensors and Communication Module, and lastly the Microcontroller and CPLD. Note that the last two sections represent the load of the WSN. As each module is described, look for the small capture of our PCB layout on the right - the orange region represents the location of that module relative to everything else.

PV Array

The WSN has ten identical footprints for IXOLAR XOB17-04x3 solar panels to collect solar. The theoretical maximum array power is \((10 \text{ panels}) \times (V_{\text{max}}) \times (I_{\text{max}}) = 10 \times (1.53\text{V}) \times (11.7\text{mA}) = 179\text{mW}\). Actual measurements have produced a maximum power of \(10 \times (1.5\text{V}) \times (5\text{mA}) = 75\text{mW}\) in some cases. The panels are connected in parallel, because if shading occurs the current in one or more panels will drop, while the voltage will remain relatively constant, keeping the overall array functional [6] (see Appendix H). One drawback is that even though all panels will operate at the same voltage, that corresponding voltage might not produce the maximum power for each panel because of the shading. The output of the solar array is the boost converter stage with MPPT functionality. An eleventh footprint is included for a solar panel to act as the pilot cell for the MPPT Fractional Voc method. This panel will be left open-circuited, and its voltage will be periodically measured. The voltage of the pilot cell will be sampled using an Analog to Digital Converter (ADC) that will then pass the data through to a Complex Programmable Logic Device (CPLD).

Solar Boost Converter with MPPT Functionality

This module transfers solar energy from the PV Array to the super capacitor and load. To achieve MPPT functionality the PV Array output voltage, and hence the converter input, must be regulated, an uncommon feature for most Switched-Mode Power Supply (SMPS). It appeared that we would need to implement this module ourselves; however after some research we identified the BQ25504RGTT from Texas Instruments as a suitable candidate [7]. This device is a specialty part used for energy harvesting applications. It implements a PFM driven boost converter that performs adjustable input voltage regulation. Thus by means of a digital potentiometer, controlled by the CPLD, we are able to supply a desired reference voltage to the device to achieve maximum power point tracking.
The MPPT technique selected is similar to the standard Fractional Voc method that exploits the nearly linear relationship that exists between the open-circuit voltage at a given temperature and irradiance, and the corresponding voltage that produces the maximum power \( V_{\text{MaxP}} = K_1 \cdot V_{\text{oc}} \) [8]. However, rather than periodically measuring the PV Array's open-circuit voltage, we measure the open-circuit voltage of a similar pilot cell to dictate the instantaneous voltage for maximum power \( V_{\text{MaxP}} = K_1 \cdot V_{\text{oc}} = K_1 \cdot K_2 \cdot V_{\text{pilot}} \) [9].

The BQ25504 fits well with this MPPT technique because we can provide an appropriate reference voltage by measuring the pilot cell, then use the digital potentiometer to set the reference voltage and instruct the device. Lastly, the BQ25504 can detect overvoltage conditions on the output, disabling the boost converter. This function allows us to limit the output super capacitor voltage to its rated maximum of 5V. The output voltage of the module will be somewhat regulated by the super capacitor itself, so the voltage will not change quickly over time. Figure 5 shows the basic schematic of the device. Our implementation of the schematic differs in the following ways:

1. A battery is not used, pin 14 is left floating.
2. \( C_{\text{STOR}} \) is our 5 Farad super capacitor.
3. \( R_{\text{OC}2} \) and \( R_{\text{OC}1} \) are not used, instead we provide a voltage reference directly to pin 4.

![Figure 5. A typical application for the BQ25504 [7].](image-url)
Energy Storage and Delivery Module

Figure 6 contains the block diagram of the Energy Storage and Delivery Module which consists of the 5 Farad super capacitor, 350mAh rechargeable lithium-ion battery, two off-the-shelf buck converters, and one custom-made bi-directional non-inverting buck-boost converter. The three converters will be described in this section, along with auxiliary circuits relevant to their operation.

![Block diagram of the Energy Storage and Delivery Module](image)

**Figure 6. Block diagram for the MPPT Super Capacitor Connection Scheme.**

The Energy Storage and Delivery Module is responsible for:

- Storing input energy from the Solar Boost Converter into either of the onboard energy sources.
- Transferring energy from the Solar Boost Converter or the onboard energy sources to the load on regulated 3.3V and 1.8V rails.
- Providing signals for the battery voltage, battery current, and super capacitor voltage to be used by a central CPLD to control regulations and monitor energy usage.
- Managing energy effectively to ensure the WSN sustains operation once deployed.

Figure 7 shows where voltage and current sensors are placed to provide information to the CPLD for purposes of energy management and voltage/current regulation.

![Placement of Voltage and Current Sensors for the WSN Power Management](image)

**Figure 7. Placement of Voltage and Current Sensors for the WSN Power Management.**
Buck Converters 3.3V & 1.8V

Energy is transferred from the Energy Storage and Delivery Module to the load by means of two off-the-shelf buck converters, which regulate system rails at 3.3V and 1.8V. The 1.8V supply drives the CPLD core, while the 3.3V drives the CPLD IO's, the microcontroller, the Smart Garden sensors, and all other auxiliary devices. The selected devices come from Texas Instruments' TPS622XX series of buck converters, which operate in Pulse Width Modulation (PWM) mode for high-current loads, and Pulse Frequency Modulation (PFM) mode for low-current loads (sub 50mA). Our WSN will typically operate below 5mA under normal conditions, thus the efficiency boost provided by PFM mode is ideal. Figure 8 compares the efficiency of PWM mode on the left, with PFM mode on the right, for a sample device [10].

![Figure 8. Comparison of PWM efficiency (left) and PFM efficiency (right) [10].](image)

These devices are rather simple to work with, as they require a small number of components, and are straightforward to test. Test results regarding regulation and efficiency will be discussed in later sections. Below is the schematic we generated regarding those devices:

![Figure 9. Schematic of the Off-the-shelf Voltage Regulators.](image)
Bi-Directional Non-Inverting Buck-Boost Converter

This converter transfers energy between the battery and super capacitor rail, and supports bidirectional power flow. The converter regulates the battery current for charging and discharging purposes. Due to the overlapping nature of the battery and super capacitor voltage ranges (3V - 4.2V vs. 3.6V - 5V), a non-inverting buck-boost architecture was selected. The converter mode is referenced to the battery, meaning that the converter operates in boost mode when the battery voltage is lower than the super capacitor voltage. Three converter modes are available from this architecture: boost, buck, and buck-boost. Converter mode operating conditions are listed below:

<table>
<thead>
<tr>
<th>Converter Mode</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boost</td>
<td>$V_{bat} &lt; V_{sc}$ &amp; $</td>
</tr>
<tr>
<td>Buck</td>
<td>$V_{bat} &gt; V_{sc}$ &amp; $</td>
</tr>
<tr>
<td>Buck-Boost</td>
<td>$</td>
</tr>
</tbody>
</table>

Only a few operating scenarios occur where the Buck condition above is required, therefore the main converter modes are boost, and then buck-boost as long as $\Delta V \leq 0.5V$.

Since this is a low power converter at sub-100mW, a PWM implementation would be too inefficient; rather Pulse Frequency Modulation is used. Various PFM techniques exist, but the one we chose uses a fixed peak inductor current dictated by the instantaneous $V_{bat}$ and $V_{sc}$ inductance [11]. To implement this, an ADC samples $V_{bat}$ and $V_{sc}$ which are then fed into a look-up table (LUT). This LUT outputs pre-calculated times $t_1$ and $t_2$ to energize the inductor to a desired 100mA peak then de-energize it to 0mA. The pre-calculated $t_1$ and $t_2$ times allow a precise amount of charge to be transferred in a switching cycle, and with the use of a third timing parameter, $t_{off}$, which is modulated, the average current can be regulated to deliver this amount of charge more or less often. Figure 10 illustrates the inductor current waveform using this PFM technique, and how $I_{ave}$ is calculated.
**Inductor Current in PFM: Fixed Peak Current Mode**

![Inductor Current Waveform](image)

**Figure 10. Inductor current waveform for PFM fixed peak current mode.**

Figure 11 illustrates the general regulation process:

1. The average battery current is measured using a sense resistor and amplifying current sensor.
2. This value is sampled by an ADC, which is compared to a digital reference current in the CPLD.
3. A bang-bang algorithm is used that simply increments or decrements the number of system clock cycles corresponding to \( t_{\text{off}} \) in order to achieve the desired average current.
4. In parallel to those steps, an ADC samples \( V_{\text{bat}} \) and \( V_{\text{sc}} \), and a LUT generates \( t_1 \) and \( t_2 \).
5. The converter mode is decided based on the relation between \( V_{\text{bat}} \) and \( V_{\text{sc}} \). The converter direction (battery charging or discharging) is dictated by other conditions.
6. The gate signals are produced to drive high-side p-channel and low-side n-channel mosfets. Mosfets with low gate capacitance were chosen which allowed level shifters to be used to drive the gates. Level shifters drive the gates with \( V_{\text{bat}} \) and \( V_{\text{sc}} \) instead of 3.3V.

*In boost mode the battery current equals the inductor current. In buck-boost mode the battery current does not equal the inductor current, so an adjustment is made to \( t_{\text{off}} \) to achieve the same battery current.*

**Figure 11. Current Regulation Algorithm using PFM fixed peak current mode and LUT's.**

See Appendix D - Bi-Directional DC/DC Converter Design Work for samples of the \( t_1 \) and \( t_2 \) calculations including a portion of the Perl Script used, the output of the script, and a graph of \( t_1 \) and \( t_2 \) for boost mode. Also included there are additional converter parameters, some verilog, and the schematic of the module.
Power Management Signal Chain

This module represents auxiliary circuitry used to capture electrical characteristics including the battery voltage, battery current, and super capacitor voltage. The chain makes use of a 4-channel analog multiplexer, a buffering unity-gain operational amplifier, and a 10-bit parallel ADC (see Figure 12). The ADC receives a 3.3V reference voltage that is fed from the super capacitor. The battery current is sensed using a bi-directional current sensor that receives a 1.024V reference voltage also fed from $V_{sc}$.

![Figure 12. Signal Flow for Power Management Signal Chain.](image)

To balance power consumption with effective system monitoring, the four signals of interest are sampled according to the rates below.

<table>
<thead>
<tr>
<th>Signal</th>
<th>Pilot Cell Voltage</th>
<th>Battery Voltage</th>
<th>Battery Current</th>
<th>Super Capacitor V.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sampling Frequency</td>
<td>66 Hz</td>
<td>66 Hz</td>
<td>66 Hz</td>
<td>66 Hz</td>
</tr>
</tbody>
</table>

Logic Devices: CPLD and Microcontroller

A microcontroller and CPLD perform operations and calculations to control various devices and execute the power management program. This section will describe aspects of the CPLD and microcontroller in detail.
Complex Programmable Logic Device

The XC2C256-7VQG100C, by Xilinx, was selected for the WSN. The device features 256 Macrocells and is a 100-pin package. Verilog was used to obtain the following CPLD responsibilities:

1. Monitoring the super capacitor voltage, battery voltage, and battery current.
2. Implementing a closed-loop controller for the power regulator.
3. Generating control signals for the power regulators.
4. Monitoring the amount of energy available in the super capacitor and battery.
5. Determining which energy source(s) can be used to power the devices.
6. Executing the MPPT algorithm.
7. Communicating with the microcontroller.
8. Controlling the digital potentiometer, bidirectional DC/DC, multiplexer, ADC, level-shifters, mosfets

![Figure 13. The high level block diagram of the CPLD modules.](image)

A high level block diagram of hardware modules implemented in the CPLD is depicted in Figure 13. In the following sub-sections, the CPLD-specific details will be explored.

CPLD Power Consumption

To facilitate low power operation, the clock frequency of the CPLD was selected to be as low as possible, as the power consumption is proportional to frequency [12]. A final clock frequency of 256kHz was selected, resulting in average CPLD consumption of:

- 90 uA for the 1.8V core voltage rail
- 450 uA for the 3.3V I/O voltage rail

These measurements were recorded when the CPLD was operating all interfaced hardware including the battery DC/DC converter, ADC, microcontroller communication interface and MPPT management hardware.
Power Regulation and Energy Monitoring

CPLD functionality described by items 1 – 3 are used to operate the Bi-Directional Non-Inverting Buck-Boost Converter. This functionality is implemented in both the Sensors and Data Management module as well as the Converter Management module. Data measurements are supported by ADC measurements which provide a 10-bit data value to the CPLD for each of the voltage and current measurements. This data is then used by the regulator hardware (implemented in Verilog) to determine the timing conditions for the DC/DC converter control signals which are then generated using a non-overlapping signal generator (see Appendix D) and fed out to the correct MOSFETs. Details of the current regulation algorithm are discussed in the Bi-Directional Non-Inverting Buck-Boost Converter module description as well as in Appendix D.

The CPLD’s energy monitoring capabilities are implemented using voltage measurements of both energy storage devices: the super capacitor and battery. The voltage of each component gives an indication of how much capacity each device has: the super capacitor is dictated by $E = \frac{1}{2}CV^2$ while the lithium-ion battery’s capacity is depicted in Figure 14.

![Figure 14. Typical Lithium-Ion Battery Discharge Curves [13].](image)

With this knowledge available to the CPLD, a power management strategy can be implemented to ensure long-lasting operation of the WSN, once deployed. Items 4 and 5 are contained within the Power Management module depicted in Figure 13. The selected power management strategy must support the following operating modes described in Table 2.
Table 2. WSN Power Management Operating Modes

<table>
<thead>
<tr>
<th>Operating Mode</th>
<th>Energy Sources Available</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal Day</td>
<td>Battery, Super Capacitor, PV</td>
<td>Sufficient ambient light to power the PVs</td>
</tr>
<tr>
<td>Normal Night</td>
<td>Battery, Super Capacitor</td>
<td>Insufficient ambient light to lower the PVs</td>
</tr>
<tr>
<td>Charge Battery</td>
<td>Super Capacitor, PV</td>
<td>Day Operation, PVs provide enough power to support load and charge both Super Capacitor and Battery</td>
</tr>
<tr>
<td>Charge Super Capacitor</td>
<td>PV and/or Battery</td>
<td>Insufficient power from PVs to support load from PVs and Super Capacitor only</td>
</tr>
<tr>
<td>Converter Off</td>
<td>PV and/or Super Capacitor</td>
<td>Super Capacitor has sufficient energy to support load</td>
</tr>
<tr>
<td>Emergency</td>
<td>PV and/or Battery, Super Capacitor</td>
<td>All energy devices unable to support load</td>
</tr>
</tbody>
</table>

The operating conditions for each mode are described in Table 13 in Appendix I, and Figure 13 below.

Figure 15. The voltage conditions that dictate the WSN power management operating mode.

While variable current regulation is supported, we decided to fix the current delivered through the Bi-Directional Non-Inverting Buck-Boost Converter to a constant 15mA (in either direction, and only when the converter is actually enabled) as this:

- Operates the converter at a high-efficiency point
- Decreases energy loss through the system as the converter is operating for shorter periods of time, due to faster charging

MPPT Algorithm

The MPPT algorithm, item 6, is used to track the maximum power point of the WSN solar array, which depends on both ambient light and temperature [6]. The hardware to support this algorithm is embedded within the MPPT Management module below. By characterizing the solar array used by the WSN, a relationship (a Scaling Factor) between open circuit voltage and maximum power point voltage can be found and stored digitally within the CPLD in an array of two-input look-up tables (LUTs). External Temperature and Light information (from the Pilot Cell Voltage) can be used to identify the appropriate
Scaling Factor and communicate this to the digital potentiometer (DPOT) that generates the reference voltage at which the Solar Boost Converter will regulate the solar array.

![Diagram of signal flow](image)

**Figure 16. CPLD Signal Flow for MPPT Module.**

Communication Interface

Finally, item 7, the communication interface, facilitates simple data transfers between the load and power management module for operation and debug. Details of this implementation are provided in the Communication Bus section to come.

Microcontroller

The PIC24F16KA102, by Microchip, was selected for the WSN. This device was preferred because of its low power consumption. C-programming was completed to perform the following tasks:

- Read environmental data from the sensors.
- Collect coordinate data from the GPS unit.
- Transmit the sensor and GPS information through the Xbee transmitter.
- Drive shutdown and enable signals for the sensors, transmitter, GPS, and other parts.
- Communicate with the CPLD.
- With help from the CPLD, read the pilot cell voltage, super capacitor voltage, battery voltage, and battery current.
- Control the sensors, multiplexer, op-amp, transmitter, GPS, and mosfets

The microcontroller accomplishes the above tasks by means of an algorithm that collects data at fixed intervals, and puts all corresponding devices to sleep between operations. The algorithm has two main operating modes: day and night.

**Table 3. Summary of Signal Sampling Times**

<table>
<thead>
<tr>
<th>Operating Mode</th>
<th>Data Collected</th>
<th>Collection Rate*</th>
<th>Data Transmission Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Day</td>
<td>Sensor, Energy, GPS</td>
<td>Higher</td>
<td>Once an hour if energy is available.</td>
</tr>
<tr>
<td>Night</td>
<td>Sensor, Energy</td>
<td>Lower</td>
<td>Data is stored overnight then transmitted in the morning.</td>
</tr>
</tbody>
</table>

*Note that the collection rate is adjustable in software, so users can choose a rate according to their needs. It is recommended that more data be collected during the day when more energy is available, and when the Smart Garden is subject to the sun, influencing the environmental conditions.
In both modes the algorithm operates on 1-hour loops that repeat. Below is a generalized, but accurate example of the algorithm's top level:

**Day Operation**

- Sensor Reading
- Energy Reading
- GPS Fix
- Xbee Transmission

**Night Operation**

- Sensor Reading
- Energy Reading
- TIMESTAMP

**Night to Day Transition**

- Xbee Transmission
- TIMESTAMP

*Figure 17. Example flow of the Smart Garden application throughout the day.*

*The associated top-level pseudo-code may be found on the next page.*
while(1) {
    // Day function -- timers are used to ensure this loop lasts 1 hour. This loop produces 4 sets of
    // sensor data and energy data, 1 timestamp, and 1 set of GPS coordinates per hour.
    do{
        sensor_block(0); // collect temperature, light, soil, humidity data
        sensor_block(1);
        energy_block(0); // collect device voltages and currents
        energy_block(1);
        gps_block(); // collect coordinate data
        sensor_block(2);
        sensor_block(3);
        energy_block(2);
        energy_block(3);
        get_timestamp(); // collect hour, minute, second data
        xbee_block(day_data); // Transmit Hourly Data
        talk_cpld(is_it_daytime); // ask the CPLD if it is still daytime
    } while(daytime)

    // Night function -- timers are used to ensure this loop lasts 1 hour. This loop produces 2 set of
    // sensor data and energy data, and 1 timestamp per hour.
    i = 0;
    do{
        sensor_block(i); // collect temperature, light, soil, humidity data
        sensor_block(i+1);
        energy_block(i); // collect device voltages and currents
        energy_block(i+1);
        get_timestamp();
        i++;
        talk_cpld(is_it_daytime?);
    } while(nighttime)
}

// Night-to-Day Transition -- transmit the stored night data
xbee_block(night_data);

The algorithm can be customized to collect sensor or energy data more or less often by adding or removing
sensor_block() and energy_block() calls. GPS data collection and Xbee transmission occur once an hour to
provide users with updated information, and to provide load steps. Those operations can be done more or
less often as well. Timestamp data is collected once an hour. The algorithm relies on communication with
the CPLD to determine if it is still daytime. In Night mode, data collection frequency is reduced to conserve
power. All data is stored during the night, and transmitted in the morning when solar energy is available. In
between the function calls above, the microcontroller is put to sleep, along with all the sensors, GPS, Xbee,
and other circuitry. Internal timers use an external 32.768kHz crystal to wait for the next major operation.
These timers are also used to ensure one loop lasts for one hour. See Appendix E for a more detailed
version of the above code. Again, only the top-level code is shown there.
A two-wire communication bus exists between the CPLD and microcontroller. Data transmission is facilitated using an asynchronous, bidirectional bus with the microcontroller acting as master. The bus operates at low speeds, on the order of 1700 bits/s. An acknowledge or “ACK” is expected from the CPLD after each transmission. If an ACK is not received within 1 second of the command being issued, the transmission is deemed a failure and the microcontroller will retransmit the command. Instructions are transferred in the following manner:

1. The microcontroller sends one high start bit along CPLD RX.
2. The microcontroller sends a certain number of pulses along CPLD RX; the number of pulses corresponds to a command as indicated in the command table in Appendix F.
3. The CPLD receives the command and upon completing the request, it either sends an ACK along CPLD TX if the command is a 'Send' command, or if the command is a 'Receive' command it sends an ACK along CPLD TX along with a certain number of pulses corresponding to specific information.

A total of 11 commands are available:

- Four commands tell the CPLD which of the four built-in scaling factors to use when determining the voltage for the MPPT algorithm.
- Another command asks the CPLD if the super capacitor voltage is high enough to support a high-power operations such as Xbee transmissions.
- Five commands instruct the CPLD to make a certain voltage or current available through its Signal Chain which also connects to one of the microcontroller's analog ports.
- The last command asks the CPLD if it is 'daytime' or 'night time' -- The microcontroller will switch over to day or night operation if the CPLD provides two consecutive replies in the positive.

See Appendix F for a table of the commands and a more detailed explanation. Screen captures of the communication module at work are included in the Testing and Verification section. The captures and explanations may shed further light on the communication process.
**Smart Garden Application: Sensors, GPS, Transmitter**

The Smart Garden Application is driven completely by the microcontroller, and represents the WSN's primary load. This module consists of the four sensors used to collect environmental features, namely the light sensor, temperature sensor, humidity sensor, and moisture sensor. Also included are a GPS, and a Xbee transmitter. Figure 18 below illustrates how the devices are connected, and includes a signal flow similar to that of the power management signal chain.

The environmental data is gathered by four analog sensors, three of which are off-the-shelf IC's. The fourth, a soil moisture sensor, can be implemented using the on-board microcontroller and passive components. Included in this module is functionality to completely cutoff the sensors and op-amp from the 3.3V rail, reducing power flow to 0W for those devices. This technique, called *power gating*, makes use of a p-channel mosfet in series with the VDD rail of the temperature sensor, humidity sensor, and op-amp; the Xbee, GPS, light sensor, and Amux have their own shutdown signals, and the soil moisture sensor is controlled directly by the microcontroller. The microcontroller drives the mosfet gate low to power the sensors, then once data acquisition is complete the gate is driven high to isolate the devices from power. Below is a summary of the power consumption impact for this module, with $V = 3.3V$. The sensors are only used for a few minutes each hour, so the 2mW savings greatly reduce system consumption.

<table>
<thead>
<tr>
<th>Power Gating Status</th>
<th>Current Consumption</th>
<th>Power Consumption</th>
</tr>
</thead>
<tbody>
<tr>
<td>ON (Mosfet Open)</td>
<td>226uA</td>
<td>0.746mW</td>
</tr>
<tr>
<td>OFF (Mosfet Closed)</td>
<td>845uA</td>
<td>2.7885mW</td>
</tr>
</tbody>
</table>

*Note that the above numbers include the disabling of the light sensor, which has its own shutdown signal.*

**See Appendix G for the sensor schematic page, and for some oscilloscope captures.**
Assessment of Proposed Design

As the document has illustrated thus far, the proposed design is quite extensive. The entire design is split into three main sections: Energy Collection, Energy Storage and Delivery, and Energy Consumption. Within each of these sections very specific tasks and goals were established to ensure the requirements were met. The Energy Collection stage needs to efficiently capture solar energy and deliver it to the next stage. The Energy Storage and Delivery stage must efficiently and cautiously transfer energy throughout the system. Lastly, the Energy Consumption stage must run the system as efficiently as possible, and carry out its end use (a Smart Garden product) reliably.

A common theme throughout is the careful use of energy: each section is equally responsible for operating efficiently to ensure the WSN can last for potentially months at a time. Much effort was spent during the design/parts-selection stage to acquire low-power parts that would yield acceptable results. Energy conservation strategies were incorporated into the design in the form of exploiting sleep modes, low device duty cycles, and power gating. The power gating (PG) decision was actually quite important because it saves around 2mW with very little effort.

Take note that while the three main sections all depend on each other to work as a whole, each section is modular enough to operate on its own provided suitable input energy is available. To be more specific, there is minimal interconnection between the three modules to ensure successful system integration. Figure 19 illustrates the relationship between sections: the design leaves little to chance. Each large section can be developed on its own and nearly seamlessly come together. Combining the three large sections was quite trivial relative to the development of each module.

Figure 19. WSN Module Dependencies.
While the *Testing and Verification* section is presented next, we will divulge the most important result discovered as a means to assessing our proposed design. Upon combining all three sections for preliminary system-level tests, we measured the current flowing from the output of the 3.3V regulator (in the Storage and Delivery stage) that powers the entire WSN. The results are tabulated below:

**Table 4. Summary of System Current and Power Draw off of the 3.3V rail.**

<table>
<thead>
<tr>
<th>Operating Condition</th>
<th>Current Draw (mA)</th>
<th>Power Consumption (mW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPLD and Slave Devices On, uC Asleep, PG On</td>
<td>0.47</td>
<td>1.551</td>
</tr>
<tr>
<td>CPLD and Slave Devices On, uC On, PG On</td>
<td>0.68</td>
<td>2.244</td>
</tr>
<tr>
<td>CPLD and Slave Devices On, uC On, Sensors On/PG Off</td>
<td>1.33</td>
<td>4.389</td>
</tr>
<tr>
<td>CPLD and Slave Devices On, uC On, PG On, GPS Operating</td>
<td>28</td>
<td>92.4</td>
</tr>
<tr>
<td>CPLD and Slave Devices On, uC On, PG On, Xbee Transmitting</td>
<td>60</td>
<td>198</td>
</tr>
</tbody>
</table>

*Relevant System Parameters: $V_{sc} = 4.5V$, $V_{bat} = 3.8V$, $I_{bat} = 16mA$

The numbers are quite impressive. We put a lot of effort into the design and parts selection stages during the Fall Semester, but even we did not think we could achieve sub-3.3mW levels in our lowest consumption state. These results validate the design, and strengthen the WSN's chances for long-lasting operation. Of course this only considers the 3.3V rail, and does not mention losses. For now we will forgo additional analysis, but more information will be provided in the Power Consumption Analysis section near the end of the document. See Appendix J - Power Measurement Captures for the ammeter measurements.
Testing and Verification

This section demonstrates the degree to which our Wireless Sensor Node fulfills the stated requirements. Each requirement is addressed with supporting evidence and analysis to show why we believe it has been achieved. Some requirements have not been fully realized, and explanations for those cases are provided. Any additional material may be found in the appendices, which are explicitly mentioned when necessary.

**Requirement F1 - Regulating 3.3V**

<table>
<thead>
<tr>
<th>Target Tolerance</th>
<th>Actual Tolerance</th>
<th>Pass/Fail</th>
<th>Target Efficiency</th>
<th>Actual Efficiency</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>+/- 5%</td>
<td>Worst pk-pk is 4.5%</td>
<td>PASS</td>
<td>&gt; 85%</td>
<td>&gt; 85%</td>
<td>PASS</td>
</tr>
<tr>
<td></td>
<td>Average is 1.2%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Texas Instrument’s TPS62203DBVT 3.3V Buck IC was used to regulate the 3.3V rail that powers the CPLD IO’s and WSN load. Figure 20 below depicts efficiency curves at low, medium, and high input voltages representing expected super capacitor voltages. A DC load was used to target certain scenarios:

**Output Current**

<table>
<thead>
<tr>
<th>Condition</th>
<th>Output Current</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual base WSN operation</td>
<td>0.69mA</td>
</tr>
<tr>
<td>Predicted base WSN operation</td>
<td>3mA</td>
</tr>
<tr>
<td>Higher than predicted base WSN operation</td>
<td>7mA</td>
</tr>
<tr>
<td>Base WSN with GPS activated</td>
<td>30mA</td>
</tr>
<tr>
<td>Base WSN with Xbee activated</td>
<td>70mA</td>
</tr>
</tbody>
</table>

**Figure 20. Efficiency Curves of 3.3V Buck Converter**
The measured efficiencies meet the minimum 85% requirement. Below are various screenshots depicting a number of required scenarios. The green signal is the 3.3V output, and the yellow signal is the input coming from a power supply. Figure 21 shows the output regulated when \(I_{out} = 100\text{mA}\), figure 22 shows the output regulated when \(I_{out} = 0.69\text{mA}\), and Figure 23 shows the output regulated when \(I_{out}\) jumps from 3mA to 70mA. These, and other scenarios satisfy the tolerance requirements. The worst-case peak-peak variation is 4.5%, whereas the worst-case average is 1.2% within 3.3V. All of our major operating points are regulated above the minimum efficiency, so this requirement was satisfied.

Figure 21. \(V_{in} = 5V, I_{out} = 100\text{mA}\).

Figure 22. \(V_{in} = 5V, I_{out} = 0.69\text{mA}\).

Figure 23. Load step from 3mA to 70mA at \(V_{in} = 5V\).
**Requirement F2 - Regulating 1.8V**

<table>
<thead>
<tr>
<th>Target Tolerance</th>
<th>Actual Tolerance</th>
<th>Pass/Fail</th>
<th>Target Efficiency</th>
<th>Actual Efficiency</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>+/- 5%</td>
<td>Worst pk-pk is 2.2%</td>
<td>PASS</td>
<td>&gt; 85%</td>
<td>&gt; 70%</td>
<td>FAIL</td>
</tr>
<tr>
<td></td>
<td>Average is 0.87%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Texas Instrument’s TPS62202DBVR 1.8V Buck IC was used to regulate the 1.8V rail that powers the CPLD core. Figure 24 below depicts efficiency curves at low, medium, and high input voltages representing expected super capacitor voltages. A DC load was used to produce output currents of 0.2mA, 0.3mA, 1.1mA, 3.1mA, and 5.1mA. Testing has indicated that this rail uses less than 0.1mA meaning that this converter operates at a maximum of 77% efficiency – well below our desired 85% minimum. This result is not surprising as only microWatts are being transferred. TI’s power-save mode prevents the efficiency dropping to below 50%. The low efficiency is acceptable because the actual power losses are low. Figure 25 is a screenshot demonstrating the output regulation at worst case input voltage (5V), and Iout = 1mA. The yellow signal is the input voltage from a power supply, and the green signal is the regulated 1.8V output. The captured measurements show a worst-case peak-to-peak variation of 2.2%, with the average voltage falling within 0.87% of 1.8V.

![Efficiency Curves for 1.8V Buck Converter](image1)

![Screenshot of 1.8V Converter with Vin = 5V, Iout = 1mA, Peak-Peak variation = 2.2%](image2)

**Figure 24. Efficiency Curves of 1.8V Buck Converter**

**Figure 25. Screenshot of 1.8V Converter with Vin = 5V, Iout = 1mA, Peak-Peak variation = 2.2%**
Requirement F3 - Collecting Environmental Data

This section will comment on all the sensors used to collect environmental data, and demonstrate that the output of the sensors is sufficient for our needs. Note that this requirement was a SUCCESS.

Soil Moisture Sensor

This sensor is implemented using two resistors, two wire leads, and two microcontroller pins, as depicted in Figure 26 below (note that a 10kΩ resistor was used instead of 100Ω and 47kΩ instead of 57kΩ):

![Soil Moisture Sensor Implementation](image)

Figure 26. Soil Moisture Sensor Implementation [14].

The sensor operates by driving the rightmost terminal in the diagram high, while driving the leftmost terminal low. The sensor acts as a voltage divider that exploits the instantaneous resistance of the soil to indicate its relative water content [14]. Therefore, soil that is more moist has more water, provides a more conductive path from the sensed pin to ground, and produces a lower output voltage. A test was done to characterize the output voltage vs. added water, of a sample of soil of around 500mL. Portions of 5mL of water were added to the soil periodically, after which the soil was allowed to sit for a few minutes, and the voltage was then measured. The resulting curve is below:

![Output Voltage vs. Water Content for the Soil Sensor](image)

Figure 27. Voltage vs. Water Content Profile

There is an overall trend for the voltage to decrease as more water is added; the trend is gradual at first then there is a sharp decline. After the decline the trend seems to level out and adding more water does not seem to have a major effect on the reading. Therefore, we can use this trend to interpret measured values from the WSN. Any readings above approximately 1.8V will indicate to the user that the target garden requires watering. Anything below 1.4V will indicate that no watering is required.
Light Sensor
This sensor takes the form of an IC that we purchased (the APDS-9007-020 by Avago Technologies). It produces an on output current that follows a logarithmic relationship with measured lux. Attempts to validate the output data involved comparing measurements between the sensor and a commercial lux meter. A halogen lamp was used to shine light on the sensor and meter. The comparison is illustrated below:

![Light Sensor Lux Comparison](image)

**Figure 28. A comparison between the WSN light sensor and a lux meter.**

The blue curve is the output of the light sensor, the red curve is the output of the lux meter, and the green curve is the ratio of sensor:meter output. Some comments can be made regarding the results:

- Under typical ambient lighting conditions the readings match (points 1,2).
- When the lamp is used the sensor reading is constantly ~9 times higher than the meter.
- This fixed ratio might be explained due to the geometry and limitations of the setup.
  - The light sensor active area is 0.15mm$^2$; therefore light is incident uniformly over the surface. The meter uses a receiving half-spherical dome that is approximately 4cm$^2$; incident light was definitely not uniform over its surface, producing lower readings.
  - The halogen lamp had a front grill on it that would block some light entering the meter, but not the sensor.
- The sensor output does have a logarithmic dependence on incident light.
- Conclusion: the sensor provides sufficient output for our needs.

Temperature and Humidity Sensors
These sensors were purchased as IC's (the HIH-5031-001 by Honeywell and the MCP9701AT-E/TT by Microchip), and were easier to test than the others due to their interdependence. Below is the output of a Perl script that calculates the equivalent temperature and humidity values acquired by our WSN when moved to different locations within a house.
Figure 29. The output of a Perl script calculating temperature and humidity from the sensors.

The script manually takes in data from the 10-bit ADC sampling of the sensor output voltage. The sensor equations are used to produce actual results. The second reading was taken outdoors, and since it was night the temperature dropped and the Relative Humidity (RH) increased greatly. Below is a screenshot confirming the outdoor conditions at the same time that the measurements were taken:

![Screenshot from the weather network showing outdoor temperature and humidity](image)

**Figure 30. A screenshot from theweathernetwork.com displaying the temperature and humidity conditions.**

The sensor data was 9.6°C and 40% RH which essentially matches this reading. The WSN was then brought indoors and moved to warm locations near a fireplace and cool locations in the basement. The RH responds as expected since as temperature decreases, RH increases. Therefore, the sensors behave as expected, and as will be shown in Requirement 3, the data can be transmitted for later use.

Coordinate Data

A GPS, the A1035-H by NEX Robotics, is used to collect the latitude and longitude of the WSN when requested. The GPS is placed in a low-power state when not in use. A sample of the output GPS data may be viewed below (the general format of the data will be discussed in Requirement F5, but the highlighted section is what matters now).
All of these readings were taken just outside the Galbraith building at UofT. The following table provides statistics on the data:

<table>
<thead>
<tr>
<th></th>
<th>Average Latitude</th>
<th>Latitudinal Variation (max - min)</th>
<th>Distance Variation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average Latitude</td>
<td>43 39.5786 N</td>
<td>00 00.064</td>
<td>1.97m</td>
</tr>
<tr>
<td>Latitudinal Variation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Distance Variation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average Longitude</td>
<td>79 23.8162 W</td>
<td>00 00.0662</td>
<td>1.635m</td>
</tr>
<tr>
<td>Longitudinal Variation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Distance Variation</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

From the test above, the variation from reading to reading is reasonable. Comparing the average result to an external reference shows that the GPS is accurate as well. See the figure below. We were located on the other side of St. George St., but the coordinates should be reliable enough for our use case. Therefore, the GPS can adequately tell users where their WSN is located, which is helpful if multiple units are deployed.

Therefore, all the sensors behave as expected and reasonable data can be extracted from them. Given the scope and primary purpose of the project, the sensors provide sufficient data for one to understand some of the characteristics in the vicinity in which the WSN is located. This requirement has been fulfilled, and the Smart Garden application has been adequately addressed from the data collection perspective.
Requirement F4 - Collecting On-Board Electrical Data

This requirement is a SUCCESS on two fronts: both the CPLD and microcontroller can independently collect this data. The CPLD accomplishment is highlighted in Requirement F11, and will be addressed there. This section focuses on the microcontroller's success. The purpose of the microcontroller collecting this energy data is to transmit it using the Xbee, so that we can observe the energy status of the system when it is deployed. The success of this task depends on the proper functioning of the communication bus between the microcontroller and CPLD, which is addressed in Requirement F8. Essentially, the analog mux that interfaces with the sensed voltages is controlled by the CPLD. The microcontroller tells the CPLD that it wants to measure these voltages using its own on-board ADC. Once the data is collected, it is transmitted along with the sensor data, timestamp, and GPS data using the Xbee. Below is a screenshot of the transmitted data that is read using Putty (Requirement F5, right after this, describes the format of the transmitted data, and describes how we collect it). Suffice for now to say that the last four entries associated with I,V,C,P represent the desired energy data.

![Screenshot of transmitted electrical data collected by the WSN.](image)

Note that the transmitted data may range from 0-1024 because a 10-bit ADC is used on the microcontroller. It is much more accurate and efficient to create a script to post-process this data into meaningful values. Below are the calculations used to back-calculate the transmitted data into the actual data being measured. Note that the operation '*3.3V/1024' is required to convert the digital value to an analog value.
**Battery Current:**

**Formula:**

\[ I_{\text{actual}} = \frac{(V_{\text{adc}} \times 3.3\text{V})}{(1024 \times 1.56) - 1.024} \times \frac{1}{8 \times 1} \]

*The formula can be re-written as:*

\[ I_{\text{actual}} = \frac{(V_{\text{adc}} \times 3.3\text{V})}{(1024 \times A_2) - K_1} \times \frac{1}{(A_1 \times R_{\text{sense}})} \]

*A_2* is the op-amp gain after the current sensor, *K_1* is the off-set voltage of our bidirectional current sensor, *A_1* is the gain of the current sensor, and *R_{\text{sense}}* is our sense resistor.

**Battery Voltage:**

**Formula:**

\[ V = \frac{2 \times (V_{\text{adc}} \times 3.3\text{V})}{1024} \]

*A scaling factor of 2 is used to undo the voltage divider we created.

**Super Capacitor Voltage:**

**Formula:**

\[ V_{\text{scactual}} = \frac{2 \times (V_{\text{adc}} \times 3.3\text{V})}{1024} \]

*A scaling factor of 2 is used to undo the voltage divider we created.

**Pilot Cell Voltage:**

**Formula:**

\[ V_{p\text{vactual}} = \frac{(V_{\text{adc}} \times 3.3\text{V})}{1024} \]

Calculated values to be compared to the screenshot:

*Battery Current* = 15.8mA  *Battery Voltage* = 3.89V  *S. Cap Voltage* = 3.94V  *Pilot Voltage* = 1.26V

Below is an image of our test setup. The power supply on the left is displaying the voltage on the battery (3.90V), and the bottom multimeter is displaying the current through the battery 15.52mA. The voltage for the super capacitor cannot be seen because we used a dual power supply, but its voltage was 3.95V. Overall we consider this requirement to be completely fulfilled by the system.

![Figure 35. A picture depicting the battery current and voltage measured by the microcontroller.](image-url)
**Requirement F5 - Wireless Data Transmission**

<table>
<thead>
<tr>
<th>Target Function</th>
<th>Actual Function</th>
<th>Pass/Fail</th>
<th>Target Range</th>
<th>Actual Range</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transmit Collected Data to a Computer</td>
<td>Transmits Collected Data to a Computer</td>
<td>PASS</td>
<td>&gt; 50m</td>
<td>~ 33m max</td>
<td>FAIL</td>
</tr>
</tbody>
</table>

This requirement was marked yellow in the Project Requirements section because the desired transmission functionality has been achieved, however the intended transmission range was not reached. The maximum transmission range during an outdoor line-of-sight test was around 33m, even though the product is advertised to have around a 90m range [16]. We do not believe that our PCB somehow reduces the effective range. A future solution may be to simply buy a more powerful Xbee device. The different products share the same footprint, therefore no PCB adjustments would be required.

![Figure 36. Xbee Product Comparison [16].](image)

Now on to demonstrating the actual transmission functionality. To receive data, the Xbee on our PCB communicates with a Xbee connected to a custom PCB that acts as a serial port for a computer. Figure 28 below depicts a Putty terminal that is reading data from the 'Xbee Serial Port', and writing this data to a file.

![Figure 37. Putty Terminal Receiving Transmitted Data.](image)
The data is transmitted in the following format:

Table 5. Format of transmitted Data from left to right, up to down. Data is comma delimited when sent.

<table>
<thead>
<tr>
<th>Flag</th>
<th>Meaning</th>
<th>WSN Identifier</th>
<th>O</th>
<th>N</th>
<th>H</th>
<th>L</th>
<th>S</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>WSN</td>
<td></td>
<td>Operation</td>
<td># of sets of</td>
<td>Humidity</td>
<td>Light</td>
<td>Soil</td>
<td>Temperature</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>data per hour</td>
<td>0-1024</td>
<td>0-1024</td>
<td>0-1024</td>
<td>0-1024</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>&gt; 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 38 below shows the results of a test meant to verify the microcontroller ADC, algorithm, and Xbee device. The collected data is not ‘real sensor data’, rather a voltage divider was connected to the PCB. The data was imported into excel as a csv from the text file putty creates during transmission.

![Figure 38](image)

Figure 38. Parsed Data from successful Microcontroller Testing

Certain features from the microcontroller algorithm can be observed from the data above:

1. Data for both day and night operation has been collected.
2. The number of sets of measurements made per loop during day operation is larger than during night operation (3 sets vs. 2 sets).
3. The time interval between transmissions is consistent within day and night operation, but differ when compared to each other (every ~12 minutes vs. every ~22 minutes).
4. GPS data is collected only during the day, but because the test was indoors the GPS could not achieve a fix, so the algorithm flagged a timeout and moved on.

*Note that each loop was quicker than 1 hour for the sake of transmission verification.*
While the low transmission range is a problem, it can be mitigated. We are still able to successfully transmit data from our WSN to a central computer, which is the main purpose of the requirement. This requirement also confirms the correct operation of the microcontroller signal flow chain, and the correct operation of the various microcontroller modules including the ADC, UART, Timers, and RTCC.

**Requirement F6 - Flexible Data Collection and Transmission Rates**

This requirement is software oriented. The **SUCCESS** of this requirement is alluded to in the Technical Design section that describes the microcontroller algorithm on page 18. The displayed pseudo-code demonstrated that sensor, energy, and GPS data can be collected and transmitted as often as desired by adding or removing the associated higher level function calls. The modular nature of the function blocks makes customization of the top-level algorithm easy and unrestricting. The only conceivable restriction in our case would be the increased power consumption resulting from increased activity. Note that Requirement F5 provides additional evidence that this requirement was achieved, as samples shown contain timestamp values whose intervals differ.

**Requirement F7 - Programming the Logic Devices**

The **SUCCESS** of this requirement is implied by the fact that the CPLD and microcontroller execute the code we developed. Note that this requirement refers to the physical hardware included on the PCB used to program these devices, i.e. the JTAG header for the CPLD, and the header that connects to the PICkit2 device used for PIC microcontrollers. These circuits were tested early on, and obviously function correctly.
Requirement F8 - Microcontroller - CPLD Communication

This requirement is considered a SUCCESS as demonstrated through the captures depicted below of 3 of the 13 possible transmissions that can occur between the CPLD and microcontroller (all 13 were indeed tested). The signals are the same in all three captures as specified below:

<table>
<thead>
<tr>
<th>Signal Number</th>
<th>Signal Name/Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>CPLD_RX ... Microcontroller sends data along this line.</td>
</tr>
<tr>
<td>4</td>
<td>CPLD_TX ... CPLD sends data along this line.</td>
</tr>
<tr>
<td>3</td>
<td>CPLD Analog Multiplexer Channel Select [1]</td>
</tr>
<tr>
<td>2</td>
<td>CPLD Analog Multiplexer Channel Select [0]</td>
</tr>
<tr>
<td>1</td>
<td>MPPT K Value Selector [1]</td>
</tr>
<tr>
<td>0</td>
<td>MPPT K Value Selector [0]</td>
</tr>
<tr>
<td>Bus 2 = [3:2]</td>
<td>AMux Select [1:0]</td>
</tr>
<tr>
<td>Bus 1 = [1:0]</td>
<td>MPPT K Select [1:0]</td>
</tr>
</tbody>
</table>

All three captures illustrate the correct sequence of instruction and data exchange between the two logic devices. Transmission begins with the microcontroller sending a start bit, followed by a series of pulses equaling the command number; note Figure 39 is command 3 so 4 total pulses are sent, Figure 40 is command 10 so 11 total pulses are sent, and Figure 41 is command 13 so 14 total pulses are sent. The CPLD responds with an ACK within 15ms, as per our design. Command 3 selects K value 2 within the CPLD, so Bus 1 equals 2. Command 10 requests that the battery voltage be selected by the analog multiplexer, which is channel 2 when indexed from 0, so Bus 2 equals 2. Lastly, Command 13 requests that the CPLD tell the microcontroller if it is daytime. The CPLD responds with 5 pulses after the ACK indicating that it is night time - the correct response at the time. Therefore the communication ability between the CPLD and microcontroller has been confirmed, and is found to be quite reliable.
Figure 39. Command 'SendKValue3' (3)

Figure 40. Command 'SendAmuxBV' (10)

Figure 41. Command 'ReceiveIsDay' (13)

Figure 42. Schematic showing that the battery voltage BATT+_DIV is connected to Amux channel 2.
Requirement F9 - Implement a Maximum Power Point Tracker

The current status of this requirement is listed as Completed and Tested with Conditions because the characterization of the WSN’s solar panels to determine a relationship between the panel’s peak power point and its open circuit voltage remains incomplete. The remainder of the requirement, however (the hardware used to regulate the solar panels at a given operating point) is implemented (using a combination of Verilog and PCB Design) and functional on the WSN and a summary of its verification follows below.

To control the digital potentiometer (DPOT) used to select the solar panel operating point, a Verilog module generates SPI signals (see Figure 44). In Figure 43 below, a pair of oscilloscope captures illustrate both the digital SPI communication between the CPLD and the DPOT in addition to the solar panel operating point tracking the MPPT reference voltage. In Figure 44, the corresponding signals are:

- Pink: SPI Data from CPLD to DPOT
- Yellow: SPI Clock from CPLD to DPOT
- Green: MPPT reference voltage from DPOT
- Blue: Solar Panel operating voltage

To verify functionality, the full range of expected operating voltages (between 1.2V and 1.75V) were used, ensuring complete hardware support for the MPPT module. The DPOT used is the AD5160BRJZ100-RL7 by Analog Devices Inc.

![Figure 43. Block Diagram of Verilog Module to Control DPOT.](image)

![Figure 44. Oscilloscope captures to verify MPPT hardware functionality.](image)

Additionally, the open circuit voltage of the solar array is provided to the CPLD via a pilot cell and ADC.
**Requirement F10 - Implement a DC/DC Converter for the Solar Array**

<table>
<thead>
<tr>
<th>Target Tolerance</th>
<th>Actual Tolerance</th>
<th>Pass/Fail</th>
<th>Target Efficiency</th>
<th>Actual Efficiency</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>+/- 2%</td>
<td>0.6% - 1.8%</td>
<td>PASS</td>
<td>&gt; 80%</td>
<td>&lt;64.3%</td>
<td>FAIL</td>
</tr>
</tbody>
</table>

This requirement has been noted as *Completed and Tested with Conditions* due to lower measured converter efficiency than expected from the converter’s datasheet. Apart from this caveat, the converter is able to perform the following tasks without issue:

- Regulate the solar panel operating voltage to the reference voltage supplied by the MPPT to within +/- 2% of the reference voltage
- Detect and handle an over-voltage condition on the output super capacitor
- Transfer power from the solar panel array to the super capacitor with less than 64.3% efficiency

Regulation of the solar panel operating voltage was verified as part of Requirement F9, in addition to the accuracy measurements presented below in Table 6. Across the range of expected operating conditions, the Solar Boost Converter is able to successfully regulate the solar panel operating voltage to within 0.6-1.7% of the reference voltage, depending primarily on the solar panel operating voltage. This results in a **SUCCESS** condition for this aspect of the requirement.

**Table 6. Solar Panel Operating Voltage and Regulation Accuracy Measurements.**

<table>
<thead>
<tr>
<th>Target PV Voltage [V]</th>
<th>1.2</th>
<th>1.3</th>
<th>1.4</th>
<th>1.5</th>
<th>1.6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regulation Accuracy [%]</td>
<td>1.67</td>
<td>1.54</td>
<td>1.43</td>
<td>0.67</td>
<td>0.63</td>
</tr>
</tbody>
</table>

The Solar Boost Converter has a selectable maximum output voltage, at which point the converter stops transferring power from the solar array. In our design, the maximum allowable super capacitor voltage is 5.0V, thus the maximum output voltage was set at this level. The Solar Boost Converter would stop transferring power at a measured super capacitor voltage of 4.93V, by open-circuiting the solar array. When the super capacitor voltage would fall below this threshold, the converter would return to normal operation to recharge the super capacitor until it crosses the maximum voltage threshold again.

Using a range of irradiance conditions and operating voltages for the solar array, the solar boost converter efficiency was measured and plotted in Figure 45. As expected, the boost converter efficiency drops with increased voltage difference between the input and output. Additionally, at higher PV voltages, the solar...
array is able to supply less current to the converter (as it approaches its open circuit voltage, see Appendix H), resulting in more significant power overhead and ultimately the converter efficiency drops. The lower than expected efficiencies from the boost converter IC cause this requirement to have its *Completed and Tested with Conditions* status. The root cause of the lower efficiency could be determined with further investigation.

![Accuracy v. PV Voltage](image)

**Figure 45.** *Regulation Accuracy Plot for the Solar Boost Converter under various solar array (PV Voltages) and super capacitor voltages (Vsc).*

![Efficiency v. PV Voltage](image)

**Figure 46.** *Efficiency Plot for the Solar Boost Converter under various solar array (PV Voltages) and super capacitor voltages (Vsc).*
Requirement F11 - Implement a DC/DC Converter that Transfers Energy amongst the Storage Devices

The Battery DC/DC converter is used to transfer power to/from the battery depending on system operating conditions. Due to the varying operating voltages of both the battery and super capacitor, the converter can operate in both Boost and Buck-Boost mode. Additionally, the ability to charge and discharge the battery enforces a bidirectional requirement for the converter.

The DC/DC converter was tested with the following conditions:

- Super capacitor voltage: 3.6 to 4.9 [V]
- Battery voltage: 3.1 to 4.1 [V]
- Converter current: -15 to +15 [mA]

Throughout this range of operating conditions, the following high-level metrics were found, as described in the Table 7 below.

Table 7. Current measurements supplied from the solar array to the energy storage devices.

<table>
<thead>
<tr>
<th></th>
<th>Full Current Range</th>
<th>Excluding Currents &lt; 3 [mA]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Efficiency [%]</td>
<td>Accuracy [%]</td>
</tr>
<tr>
<td><strong>Maximum</strong></td>
<td>93.36</td>
<td>39.62</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td>86.65</td>
<td>2.17</td>
</tr>
<tr>
<td><strong>Minimum</strong></td>
<td>78.76</td>
<td>-20.50</td>
</tr>
</tbody>
</table>

As indicated in Figures 47-52 a number of trends can be observed about the converter operation:

- Boost conversions (when Vsc – Vbatt > 0.5V) are significantly more efficient than Buck-Boost primarily due to less switching loss (2 switches for Boost, 4 switches for Buck-Boost)
- As the converter current decreases in magnitude, the overhead power (gate drivers, switching loss) become more significant, reducing efficiency
- As the converter current decreases in magnitude, the regulation accuracy decreases primarily due to the smallest measurable current in the regulation loop being 0.6mA.
From the observed trends, the DC/DC converter is best to operate for current magnitudes > 3mA to improve both regulation accuracy and efficiency. As a result, the DC/DC converter meets the efficiency target for this requirement (a minimum of 80%) as well as the accuracy requirement (within +/- 5%) for the bulk of operating conditions. The most extreme inaccuracies were observed to occur when the battery was extremely discharged (< 3.1V), a rare operating condition for the WSN.

Figure 47. Efficiency Plot for the Battery DC/DC Converter with a fixed battery voltage, various super capacitor voltages (Vsc) and operating currents.
Figure 48. Efficiency Plot for the Battery DC/DC Converter with fixed super capacitor voltage, various battery voltages (Vb) and operating currents.

Figure 49. Efficiency Plot for the Battery DC/DC Converter with fixed battery voltage, various super capacitor voltages (Vsc) and operating currents.
Figure 50. Efficiency Plot for the Battery DC/DC Converter with fixed super capacitor voltage, various battery voltages (Vsc) and operating currents.

In the following Accuracy plots, a trend emerges where poor regulation accuracy develops with low regulation currents, a result that’s expected due to our minimum detectable current being 0.6mA.

Figure 51. Accuracy Plot for the Battery DC/DC Converter with fixed battery voltage, various super capacitor voltages (Vsc) and operating currents.
Figure 52. Accuracy Plot for the Battery DC/DC Converter with fixed super capacitor voltage, various battery voltages (Vb) and operating currents.

A number of oscilloscope captures are presented in the following figures detailing the following converter properties:

- Regulation in Pulse Frequency Modulation (PFM) Mode
- Operation in Buck-Boost and Boost Mode

Figure 53. Oscilloscope capture portraying gate signals (Blue, Pink) and the Boost Switching Node operating in Pulse Frequency Modulation (PFM) Mode.
Figure 54. Oscilloscope capture portraying the Boost Switching Node operating in Pulse Frequency Modulation (PFM) Mode.

Figure 55. Oscilloscope capture portraying the Boost Switching Node and Buck-Boost Switching Node, respectively, operating in Pulse Frequency Modulation (PFM) Mode.
**Figure 56. Oscilloscope capture portraying the switching nodes transitioning from Buck-Boost to Boost Modes.**

Additional components of interest include:

<table>
<thead>
<tr>
<th>Device Function</th>
<th>Device Part #</th>
<th>Device Manufacturer</th>
</tr>
</thead>
<tbody>
<tr>
<td>1MHz Oscillator</td>
<td>LTC6906CS6#TRMPBF</td>
<td>Linear Technology</td>
</tr>
<tr>
<td>10-bit Parallel ADC</td>
<td>AD7813YRZ</td>
<td>Analog Devices Inc.</td>
</tr>
<tr>
<td>Bi-Directional Current Sensor</td>
<td>LT1787HS8#PBF</td>
<td>Linear Technology</td>
</tr>
<tr>
<td>Voltage Reference</td>
<td>ISL60002BAH333Z-TK</td>
<td>Intersil</td>
</tr>
<tr>
<td>4 – Channel Analog Multiplexer</td>
<td>ADG804YRMZ</td>
<td>Analog Devices Inc.</td>
</tr>
</tbody>
</table>
Table 8. Current measurements supplied from the solar array to the energy storage devices.

<table>
<thead>
<tr>
<th>Operating Condition</th>
<th>Current from Solar Boost Converter [mA]</th>
<th>Current into Super Capacitor [mA]</th>
<th>Current into Battery Converter [mA]</th>
<th>PASS/FAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Charging Super Capacitor from PV and Battery</td>
<td>3.00</td>
<td>14.96</td>
<td>-12.10</td>
<td>PASS</td>
</tr>
<tr>
<td>Charging Battery from Super Capacitor and PV</td>
<td>3.00</td>
<td>-16.10</td>
<td>18.75</td>
<td>PASS</td>
</tr>
</tbody>
</table>

The final design allows power to flow (at the control of the WSN) from the solar panels, to the load, super capacitor and battery. The ability to recharge the two energy sources hinged on the correct operation of our power converters. The solar Boost converter transfers power from the solar panels into a central node which connects to both of the energy storage devices: the super capacitor and lithium-ion battery (Figure 57).

Current flow into either of these devices is controllable by adjusting the solar panel’s operating point (and ultimately the current delivered from the solar panels) as well as the current taken by the converter that transfers power to/from the lithium-ion battery. When the load current is small, the super capacitor current will be the difference of the battery current and solar panel current, allowing either storage device to be recharged; this is the primary PASS/FAIL condition.

![Diagram of Energy Collection and Storage Device Connections](image)

**Figure 57. Diagram of Energy Collection and Storage Device Connections.**

The two main configurations were tested (charging either the super capacitor or the battery) to verify correct power flow throughout the system under expected operating conditions. In Figures (58 to 59) as well as in Table 8, a summary of the test results are presented.
Figure 58. Setup used to verify Requirement F5 consisting of a Halogen lamp (light source), ammeters and power supplies to monitor conditions on the WSN PCB.

Figure 59. Ammeter readings for test conditions listed in Table 6.

Requirement F13 - Execute a power management program to maximize the WSN’s operation

The power management program used to maximize the operational lifetime of the WSN is designed and implemented in Verilog on the CPLD. The actual logic was described in the CPLD section of the Module-Level descriptions, with an additional chart provided in Appendix I. A brief three-day test was conducted to test the power management logic and see how the WSN performs. The figures below illustrate measured data from this test. The test commenced at 06:55 on a Friday until 09:25 on the following Monday. Note that only daytime data is displayed – night conditions are less dynamic and excluded in this section. Data points are recorded approximately every four minutes.
**Sensor Data**

In Figure 60, the pilot cell voltage and the light sensor measurements for Friday are displayed. Note the logical correlation between the two data sets.

![Figure 60. Plot of measured pilot cell voltage and light intensity from the WSN long-term operation test.](image)

Figure 61 illustrates measured relative humidity and temperature throughout the WSN operation. A logical correlation exists in this relationship as well, with the relative humidity decreasing when temperature increases. This is an expected result as relative humidity is temperature dependent, with warmer air having a higher moisture capacity and thus a low relative humidity, for a constant amount of moisture.

![Figure 61. Plot of measured humidity and temperature data from the WSN long-term operation test.](image)
**Electrical Data**

On-board electrical data was collected to verify the execution of the power management program and observe the conditions of the WSN over a three day period. Various observations are made for each of the graphs below.

- Positive battery currents (blue) occur when the battery is discharging.
- Negative battery currents occur when the battery is charging.
- When there is a Solar Excess, and the Capacitor voltage (red) is low, the Capacitor is charging.
- When there is a Solar Excess, and the Capacitor voltage is high enough, the battery is charging.
- During Solar Deficit, the Capacitor voltage is kept at higher levels, and the battery is never charged.
- **Analysis:** The Power Management Program effectively oversees the battery and Super Capacitor.

![Daylight Hours: Super Capacitor Voltage and Battery Current](image)

**Figure 62. Plot of measured SuperCapacitor Voltage and Battery Current from the WSN long-term test.**

- Night data is not displayed, so during the transition to the next day the battery voltage (pink) drops because it powers the WSN at night.
- The battery is charged on Friday and Saturday during periods of Solar Excess and high Capacitor voltages; conversely there was no replenishment during Sunday due to poor solar input.
- Periods of Solar Excess and Solar Deficit are defined by a Pilot Cell Voltage threshold of ~1.44V.
- **Analysis:** If there is sufficient solar energy to replenish the battery, the WSN will be sustained, and long-term operation is possible.
The graph below is just of the Friday data.

- Take note of the local minimums of the SuperCapacitor voltage wherever the signal intersects a vertical gridline – these samples represent high-power operations, namely using the GPS and XBee.
- It is common for the battery to be charging the Capacitor (+ve blue waveform) for those samples.

Through examination of the graphs above during the three day test, it is clear that the power management program responds correctly to external and internal variations – changes in solar input and device voltages respectively. That test, along with simpler in-lab experiments, confirm the operation of the program.
Requirement O1 - Sub-200mW Solar Array

**SUCCESS** - The maximum rated power of the entire array is:
\[
(10 \text{ panels}) \times (V_{\text{pmax}}) \times (I_{\text{pmax}}) = 10 \times (1.53V) \times (11.7mA) = 179\text{mW}.
\]
Actual measurements have produced a maximum power of \(10*(1.5V)*(5mA) = 75\text{mW}\) in some cases.

Requirement C1 - Measure no larger than 6 inches x 4 inches

**SUCCESS** – Board measures 6.0 inches x 4.0 inches.

Requirements C2,C3-Use a Super Capacitor and Rechargeable Battery

**SUCCESS** - Proven through examination of our PCB.

Requirement C4 - Day and Night Outdoor Operation

**SUCCESS** - The confirmation of Requirement 10 regarding the Solar Boost converter ensures that energy will be available during the day which the WSN can use to operate. Furthermore, confirmation of Requirement 11 ensures that energy can be transmitted to and from the battery, making both day night operation feasible.

Requirement C5 - Operate autonomously once deployed

**SUCCESS** - The WSN uses on-board processing devices including a microcontroller and CPLD to automatically control hardware, perform calculations, and monitor the system. Users may apply an external reset upon deployment if required.

Requirement C6 - Remain operational for no less than 7 days

**SUCCESS** – Once all other functionality was tested and confirmed, including the power management program, a longer test was run to see how the WSN would perform. Below is a graph depicting the various energy-related waveforms collected by on-board circuitry, and transmitted. Note that a full-page equivalent is included in Appendix K for easier viewing. The WSN was placed on a window sill indoors (since a physical covering has not been designed for the device), and left to sit for about nine days.
The following observations can be made by analyzing the waveforms (note that samples taken during ‘daytime’ are about 4 minutes apart whereas ‘nighttime’ samples are about 20 minutes apart):

- the Pilot Cell voltage (blue) is ‘semi-circular’, corresponding to the variation of incident sunlight throughout the day; ‘night time’ is clearly visible over the smallest values; each day is one blue ‘hill’
- the first four days were sunny enough that the battery was charged by solar energy (negative battery current – purple, and increasing battery voltage – green)
- the next few days saw less intense sunlight (Pilot Voltage lower – blue), therefore the WSN relied primarily on the battery for energy (decreasing battery voltage – green)
- occasionally there was enough sunlight to partially charge the battery again (Day 6, 8, 9); the battery voltage increases accordingly
- the Super Capacitor voltage (red) is consistently regulated between 4.0V and 5.0V, as desired

Figure 65. Schematic showing that the battery voltage BATT+_DIV is connected to Amux channel 2. This test demonstrates the typical performance of the WSN. If there is sufficient sunlight the battery will be recharged and longer operation will be possible. Conversely, the WSN can sustain itself over multiple days of cloudy weather if the battery has adequate charge. Of course the ideal operational lifetime of the WSN is many months, but a nine day test is sufficient for this time-constrained project. Note that the WSN is still reliably charged when the test is stopped \( (V_{bat} = 3.6V, V_{sc} = 4.9V) \). The behaviour of the WSN over this nine day test coupled with the power analysis results, which will be presented shortly, strongly demonstrate the potential for long-term operation.
Power Consumption Analysis

The table below contains a summary of the power and energy consumption of the WSN for different operating conditions, and is averaged over one day’s typical load profile.

Table 9. Power and Energy Measurements of the WSN.

<table>
<thead>
<tr>
<th>WSN Power Consumption Summary</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th># of Seconds On per Day</th>
<th>Base WSN</th>
<th>Sensors</th>
<th>GPS</th>
<th>Xbee</th>
<th>Additional Devices and Losses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load Current (mA)</td>
<td>78148</td>
<td>7680</td>
<td>480</td>
<td>92</td>
<td>86400</td>
</tr>
<tr>
<td>Load Pout (mW)</td>
<td>0.519</td>
<td>1.33</td>
<td>28</td>
<td>60</td>
<td>NA</td>
</tr>
<tr>
<td>Average Load Pin (mW)</td>
<td>1.713</td>
<td>4.389</td>
<td>92.4</td>
<td>198</td>
<td>NA</td>
</tr>
</tbody>
</table>

The ‘Weighted Sum’ values are calculated using the data in the top-most chart. The load profile is based on a typical day with:
- 8 Xbee transmissions lasting 4 seconds each plus 1 transmission lasting 60 seconds
- 8 GPS acquisitions lasting 60 seconds each
- 64 sensor readings lasting 120 seconds each

§ Base WSN means that only the power management CPLD and the most critical devices are operating. All other devices are in Sleep Mode or are Power Gated.

* The total WSN consumption (yellow) equals the sum of the final values in the purple and oranges boxes.

The range of WSN power and energy use depends on how long the custom PFM converter operates.

Please note that the Base WSN, Sensors, GPS, and Xbee operated off of the 3.3V rail. There are a few select devices, namely the current sensor, the level shifters, pull-up/down resistors, and voltage dividers that operate off of the super capacitor rail and battery rail. The CPLD core operates on the 1.8V rail. This fixed power loss is represented on the far left under ‘Additional Devices and Losses’.

55
Analysis

- A consumption target was not selected during problem definition, as values cannot be reliably predicted.
- The total WSN consumption is low enough that the solar array and battery can sustain the system.
- The graphed results from F13 and C6 illustrate the potential for the WSN to operate for long periods.
- Therefore the WSN is deemed to be a success.

With all that in mind, we believe it is realistic to expect that the WSN will be able to operate over long periods of time. The solar array should be able to match the average system power consumption during a typical summer season, with the battery as a backup.

Summary and Conclusions

Upon completion of the WSN design project, the goals and requirements established at the onset have been primarily met. Four requirements remain “Completed and Tested with Conditions” due to test results not meeting specified targets from the product datasheets or hardware-only support for the Maximum Power Point Tracking requirement. Nonetheless, the bulk of the functionality has been implemented, tested and validated to meet or exceed requirement targets. The results from the validation and testing completed on the WSN design indicate correct operation and functionality of each of the WSN’s modules, as well as correct system operation.

During the testing and validation of both module and system operation, an extensive suite of tests were used to ensure correct and desired functionality. Each test was run repeatedly and over extended periods of time (hours or days, where possible) to validate the design in long-term operation. The ease of system integration once the modules were fully tested illustrates the extent of testing completed and how strongly the final design has been proven.

The design ideas developed and implemented throughout the WSN project have been proven out quite effectively. The custom-designed Bi-Directional DC/DC Converter is a prime example of a proven design idea, where the converter operation met or exceeded our requirement targets (accuracy and efficiency) for the module. Additionally, the complete functionality for the data gathering and transmission aspect of the smart garden application was implemented successfully, allowing a number of requirements to be met with the circuitry consuming very little power.
System integration has verified the power flow premise that serves as a foundation for the WSN design. Current flow is configurable by the WSN to charge the super capacitor or battery from solar energy as well as send power from the battery to the super capacitor for night operation. Additionally, measurements for average power used by the WSN’s consumption module show that a maximum average power of 4.303mW is actually used by the WSN, corresponding to an average of 371.76J of energy consumed in any given 24 hour period.

A long term test was run that saw the WSN operate for nine days before the test was stopped due to time constraints. The successful test demonstrates the effectiveness of the power management logic to configure power flow and manage the energy resources. These results validate the low-power design effort put forth throughout the project, and strongly suggest that long-term operation is possible.

A number of conclusions can be drawn from the successful completion of the WSN design and are as follows:

- The Smart Garden Application can be fully implemented while consuming low average power
- A discrete implementation of the custom-designed DC/DC converter can be done while meeting our regulation accuracy and power efficiency targets
- Intelligent power management for light loads (~1mA) can be done with little power overhead
- The underlying energy management and power flow configuration principles are sound

The results from the WSN design are not restricted to the Smart Garden Application. Emphasis was placed on intelligent power management, low-power design and high-efficiency conversions. Consequently, the power management aspect of this project could be applied to any general light load. Additionally, the abundance of WSNs and their wide utilization in many applications can yield industrial applications of this WSN design. Moreover, our WSN addresses a fundamental issue with many existing WSNs: the use of a rechargeable lithium-ion battery and super capacitor in conjunction aim to reduce maintenance requirements once deployed. In prior designs, a low-cost but non-rechargeable battery would be used, requiring a service technician to be deployed to the possibly remote location of the WSN to replace the depleted battery. In contrast, our WSN has the potential ability to manage its resources efficiently and effectively, maximizing run-time once deployed, thus minimizing the amount of maintenance required by any given node.

Overall, the WSN design was determined to be successful and a number of the design ideas developed early in the project have been proven out effectively. Throughout the duration of the project, both team members have found the experience to be rich in technical experience and as a whole extremely rewarding.
Future Work and Improvements

While the WSN developed does perform its intended functions, there are some aspects that may be improved or optimized to enhance reliability, lower cost, and increase operation time. The following lists present areas of future work and their potential benefits:

Lowering System Cost and Size

1. Decrease the number of solar panels, and decrease the size of energy storage devices: To accomplish this, build multiple WSNs with differing numbers/sizes of said components and see how long the WSNs run. If there is a redundancy, then use the WSN with fewer/smaller components.
2. Investigate the lower than expected efficiency results for the Solar DC/DC Converter. Improving energy collection efficiency means that fewer solar panels can be used.
3. Redo the PCB layout to optimize for size. Our initial PCB was simply a prototype with extra jumpers and spacing to make development easier.

Increase Operation Time

1. Further optimize certain circuitry for efficiency. Voltage dividers and pull-up/down circuits were already designed with high resistances, but further improvements to these and other circuits can be made.
2. Further optimize the custom DC/DC converter to operate even more efficiently around a couple operating points. Originally the converter was designed for a large number of operating points, then tested afterwards for the best ones. A more targeted re-design may yield better results.

Enhance Reliability

1. Increase the range of the radio transmitter by purchasing a more powerful device.
2. Additional circuitry may be added to address corner-case under-voltage conditions. We can better utilize the ‘Enable’ pins on the TI Buck converters for instance.

Other Work and Improvements

1. A physical case needs to be constructed to house the device, in order to use the WSN outdoors. Our project was simply the electrical side of things, so a mechanical design team would be required to design the case.
2. Simply run more tests to see how long the WSN can actually operate. By the end of the semester we did have some preliminary test results, but it would be interesting to see how long the device can really sustain itself.
References


Figure 66. Gantt Chart representation of project planning as of March, 2012
Figure 67. Gantt Chart representation of project planning as of January, 2012
Figure 68. Gantt Chart representation of project planning as of October, 2011.
Appendix B – PCB Schematic and Layout

Microcontroller

XBees Radio
Sensors

Sensors Signal Chain

AMUX to use Single Opamp for Sensors

Buffer Amplifier for Low Impedance Source for ADC
GPS Module

Power Sources
Rail Regulators and Voltage References (uses TI’s and )

Solar Boost Converter (uses TI’s BQ25504)
Complex Programmable Logic Device

Custom Bi-Directional Buck-Boost Converter

For 'Real' resistor values... R2 and R4 can go a bit higher... R3 and R5 can go either way.

Drive NMOS FETs directly from Logic Signal but Keep Logic Buffer Placeholder for Low Level FETs in case needed.
Power Management Signal Chain

Additional Power Management Signal Sources

Make the following values equal: R58,59,60,61,...can probably go higher than 300K for *Real Values

Choose R20 as sense resistor

Choose C63 = 1uF, might change to 10pF someday

Choose C62 local supply cap for Reference

C63 output filtering cap with 20K internal R

Choose R23/4 for Opamp NonInverting Gain (1+R24/R23) want final gain of ~1.5555
PCB Layout
Appendix C – Expenses

The table below outlines the various expenses incurred for parts and services.

Table 10. Additional converter parameters.

<table>
<thead>
<tr>
<th>Date</th>
<th>Vendor</th>
<th>Description</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oct 19 2011</td>
<td>MegaBatteries</td>
<td>4 350mAh LiIon Batteries</td>
<td>$ 40.00</td>
</tr>
<tr>
<td>Nov 14 2011</td>
<td>Proto Advantage</td>
<td>QFN Protoboard</td>
<td>$ 20.69</td>
</tr>
<tr>
<td>Jan 5 2012</td>
<td>Digikey</td>
<td>Initial Parts Order for 2 Boards</td>
<td>$ 282.36</td>
</tr>
<tr>
<td>Jan 5 2012</td>
<td>Mouser</td>
<td>2 GPS Modules</td>
<td>$ 65.40</td>
</tr>
<tr>
<td>Jan 6 2012</td>
<td>Newark</td>
<td>4 Analog Multiplexers</td>
<td>$ 22.10</td>
</tr>
<tr>
<td>Jan 18 2012</td>
<td>Rush PCB</td>
<td>PCB Fabrication (5 PCB’s)</td>
<td>$ 147.10</td>
</tr>
<tr>
<td>Jan 27 2012</td>
<td>Digikey</td>
<td>XBEE Radios</td>
<td>$ 81.41</td>
</tr>
<tr>
<td>Feb 8 2012</td>
<td>Digikey</td>
<td>Additional Passives</td>
<td>$ 11.41</td>
</tr>
<tr>
<td>Feb 23 2012</td>
<td>Digikey</td>
<td>Replacement OPAMPS</td>
<td>$  4.80</td>
</tr>
<tr>
<td>Feb 27 2012</td>
<td>Digikey</td>
<td>Replacement CPLDs</td>
<td>$  55.11</td>
</tr>
<tr>
<td>Mar 19 2012</td>
<td>Creatron Inc</td>
<td>XBEE Explorer Board</td>
<td>$  30.68</td>
</tr>
</tbody>
</table>

**Grand Total** $ 761.06
Appendix D - Bi-Directional DC/DC Converter Design Work

This appendix contains the following additional information regarding the bi-directional converter:

- Samples of the $t_1$ and $t_2$ calculations including a portion of the Perl Script used, the output of the script, and a graph of $t_1$ and $t_2$ for boost mode. Also included is the Verilog implementation code relating to $t_1$ and $t_2$.
- A summary of additional converter parameters, and the schematic of the module.
- The schematic of the module.
- Supplementary efficiency and accuracy graphs illustrating the converter’s performance.

Perl Script - Used for calculating $t_1$ and $t_2$ and also used to size the inductor and choose a system clock.

#!/usr/bin/perl
use warnings;
use POSIX qw(floor);

my @sc = (5, 4.95, 4.9, 4.85, 4.8, 4.75, 4.7, 4.65, 4.6, 4.55, 4.5, 4.45, 4.4, 4.35, 4.3, 4.25, 4.2, 4.15, 4.1, 4.05, 4, 3.95, 3.9, 3.85, 3.8, 3.75, 3.7, 3.65, 4.3, 4.25, 4.2, 4.15, 4.1, 4.05, 4, 3.95, 3.9, 3.85, 3.8, 3.75, 3.7, 3.65, 3.6, 3.55, 3.5, 3.5, 3.45, 3.4, 3.35, 3.3, 3.25, 3.2, 3.15, 3.1, 3.05, 3, 2.95, 2.9, 2.85, 2.8, 2.75, 2.7);

my @boost = ();
my @bb = ();

my $ind = 0.00047;
my $ipk = 0.1;
my $clk = 1000000/256000;

open OUTPUT_CSV2, ">times_boost.csv" or die "unable to open boost.csv";
open OUTPUT_CSV3, ">times_bb.csv" or die "unable to open bb.csv";

my $i = 0, $j = 0, $m = 0, $n = 0, $e = 0;

for($i = 0; $i < $#sc; $i++)  #sc loop
    { 
        for($j = 0; $j < $#batt; $j++)  #batt loop
            { 
                #boost
                if(($sc[$i] > $batt[$j]) && (($sc[$i] - $batt[$j]) > 0.5))
                    { 
                        $boost[$n][0] = 1000000*$ind*$ipk/($batt[$j]);
                        $boost[$n][1] = 1000000*$ind*$ipk/($sc[$i] - $batt[$j]);
                        $boost[$n][2] = $boost[$n][0]/$clk;
                        $boost[$n][3] = $boost[$n][1]/$clk;
                        $boost[$n][4] = $batt[$j];
                        $boost[$n][5] = $sc[$i];
                        $boost[$n][6] = $batt[$j];
                        $boost[$n][7] = ($sc[$i] - $batt[$j]);
                        
                        if($boost[$n][2] < 3.35)
                            {$boost[$n][8] = 3;}
                        elsif($boost[$n][2] < 3.85)
                            {$boost[$n][8] = 3.5;}
                        elsif($boost[$n][2] < 4.35)
                            {$boost[$n][8] = 4;}
                        else
                            {$boost[$n][8] = 4.5;}
                    }
            }
print OUTPUT_CSV2
"$boost[$n][0],$boost[$n][1],$boost[$n][2],$boost[$n][3],$boost[$n][4],$boost[$n][5],$boost[$n][6],$boost[$n][7],$boost[$n][8]\n";

$n++;
}

$bb

$bb[$e][0] = 1000000*$ind*$ipk/($batt[$j]);
$bb[$e][1] = 1000000*$ind*$ipk/($sc[$i]);
$bb[$e][2] = $bb[$e][0]/$clk;
$bb[$e][3] = $bb[$e][1]/$clk;
$bb[$e][4] = $batt[$j];
$bb[$e][5] = $sc[$i];

if($bb[$e][2] < 2.85)
{ $bb[$e][6] = 2.5; }
elsif($bb[$e][2] < 3.35)
{ $bb[$e][6] = 3; }
elsif($bb[$e][2] < 3.85)
{ $bb[$e][6] = 3.5; }
elsif($bb[$e][2] < 4.35)
{ $bb[$e][6] = 4; }
elsif($bb[$e][2] < 4.85)
{ $bb[$e][6] = 4.5; }
elsif($bb[$e][2] < 5.35)
{ $bb[$e][6] = 5; } else
{ $bb[$e][6] = 6; }

if($bb[$e][3] < 2.85)
{ $bb[$e][7] = 2.5; }
elsif($bb[$e][3] < 3.35)
{ $bb[$e][7] = 3; }
elsif($bb[$e][3] < 3.85)
{ $bb[$e][7] = 3.5; }
elsif($bb[$e][3] < 4.35)
{ $bb[$e][7] = 4; }
else
{ $bb[$e][7] = 4.5; }

print OUTPUT_CSV3
"$bb[$e][0],$bb[$e][1],$bb[$e][2],$bb[$e][3],$bb[$e][4],$bb[$e][5],$bb[$e][6],$bb[$e][7]\n";

$e++;
}

#bb

close(OUTPUT_CSV2);
close(OUTPUT_CSV3);
Excel File - A portion of the Perl script output file calculating \( t_1 \) and \( t_2 \) in terms of microseconds, clock cycles, and rounded clock cycles. The LUT's were created from this type of data.

<table>
<thead>
<tr>
<th>( V_b )</th>
<th>( V_{sc} )</th>
<th>( V_{sc-V_b} )</th>
<th>( t_1 ) ( V_b )</th>
<th>( t_2 ) ( V_{sc-V_b} )</th>
<th>( t_1 \text{ clks} )</th>
<th>( t_2 \text{ clks} )</th>
<th>( t_1 \text{ clks}_{\text{rnd}} )</th>
<th>( t_2 \text{ clks}_{\text{rnd}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.05</td>
<td>4.6</td>
<td>0.55</td>
<td>11.60494</td>
<td>85.45455</td>
<td>2.970864</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>4</td>
<td>4.55</td>
<td>0.55</td>
<td>11.75</td>
<td>85.45455</td>
<td>3.008</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.95</td>
<td>4.5</td>
<td>0.55</td>
<td>11.89873</td>
<td>85.45455</td>
<td>3.046076</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.9</td>
<td>4.45</td>
<td>0.55</td>
<td>12.05128</td>
<td>85.45455</td>
<td>3.085128</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.85</td>
<td>4.4</td>
<td>0.55</td>
<td>12.20779</td>
<td>85.45455</td>
<td>3.125195</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.8</td>
<td>4.35</td>
<td>0.55</td>
<td>12.36842</td>
<td>85.45455</td>
<td>3.166316</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.75</td>
<td>4.3</td>
<td>0.55</td>
<td>12.53333</td>
<td>85.45455</td>
<td>3.208533</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.7</td>
<td>4.25</td>
<td>0.55</td>
<td>12.7027</td>
<td>85.45455</td>
<td>3.251892</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.65</td>
<td>4.2</td>
<td>0.55</td>
<td>12.87671</td>
<td>85.45455</td>
<td>3.342222</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.6</td>
<td>4.15</td>
<td>0.55</td>
<td>13.05556</td>
<td>85.45455</td>
<td>3.389296</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.55</td>
<td>4.1</td>
<td>0.55</td>
<td>13.23944</td>
<td>85.45455</td>
<td>3.437714</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.5</td>
<td>4.05</td>
<td>0.55</td>
<td>13.42857</td>
<td>85.45455</td>
<td>3.487536</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.45</td>
<td>4</td>
<td>0.55</td>
<td>13.62319</td>
<td>85.45455</td>
<td>3.538824</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.4</td>
<td>3.95</td>
<td>0.55</td>
<td>13.82353</td>
<td>85.45455</td>
<td>3.589296</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.35</td>
<td>3.9</td>
<td>0.55</td>
<td>14.02985</td>
<td>85.45455</td>
<td>3.646061</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.3</td>
<td>3.85</td>
<td>0.55</td>
<td>14.24242</td>
<td>85.45455</td>
<td>3.693091</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.25</td>
<td>3.8</td>
<td>0.55</td>
<td>14.46154</td>
<td>85.45455</td>
<td>3.741154</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.2</td>
<td>3.75</td>
<td>0.55</td>
<td>14.6875</td>
<td>85.45455</td>
<td>3.790215</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.15</td>
<td>3.7</td>
<td>0.55</td>
<td>14.92063</td>
<td>85.45455</td>
<td>3.839683</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>3.1</td>
<td>3.65</td>
<td>0.55</td>
<td>15.16129</td>
<td>85.45455</td>
<td>3.888129</td>
<td>21.876363</td>
<td>4</td>
<td>22</td>
</tr>
<tr>
<td>3.05</td>
<td>3.6</td>
<td>0.55</td>
<td>15.40984</td>
<td>85.45455</td>
<td>3.937418</td>
<td>21.876363</td>
<td>4</td>
<td>22</td>
</tr>
<tr>
<td>3</td>
<td>3.55</td>
<td>0.55</td>
<td>15.66667</td>
<td>85.45455</td>
<td>4.004667</td>
<td>21.876363</td>
<td>4</td>
<td>22</td>
</tr>
<tr>
<td>4.1</td>
<td>4.65</td>
<td>0.55</td>
<td>11.46341</td>
<td>85.45455</td>
<td>2.934634</td>
<td>21.876363</td>
<td>3</td>
<td>22</td>
</tr>
</tbody>
</table>

Graph - This graph depicts the various \( t_1 \) and \( t_2 \) clock cycles required to achieve peak inductor currents of 100mA and 0mA in boost mode, regardless of \( V_{bat} \) and \( V_{sc} \). This graph helped establish a cut-off point where the converter should operate in buck-boost instead of boost mode.

![Graph of t1 and t2 clock cycles for boost mode](image)

**Figure 69. Calculated \( t_1 \) and \( t_2 \) values for boost mode.**
**Verilog Implementation**

The digital implementation of these $t_1$ and $t_2$ values were done using an array of single-input look-up tables (LUTs). To be able to count half clock cycles, a dual-edge triggered counter was implemented within the CPLD when generating the DC/DC converter control signals. A sample of the Verilog used by the CPLD is provided in Figures 61 to 62 below.

```verilog
130 //only care about 50mV steps. This is
131 //first do the ton (same for boost and
132 always @(*)
133 begin
134    case (MeasBattVolt[9:4])
135       6'h28: ton <= 6'd6;
136       6'h27: ton <= 6'd6;
137       6'h26: ton <= 6'd6;
138       6'h25: ton <= 6'd6;
139       6'h24: ton <= 6'd6;
140       6'h23: ton <= 6'd6;
141       6'h22: ton <= 6'd6;
142       6'h21: ton <= 6'd7;
143       6'h20: ton <= 6'd7;
144       6'h1f: ton <= 6'd7;
145       6'h1e: ton <= 6'd8;
146       6'h1d: ton <= 6'd8;
147       default: ton <= 6'b0;
148    endcase
149 end

150 reg [5:0]toff_boost;
151 always @(*)
152 begin
153    case (VoltDifferenceAbs[4:0])
154       5'h09: toff_boost <= 6'd48;
155       5'h0a: toff_boost <= 6'd44;
156       5'h0b: toff_boost <= 6'd40;
157       5'h0c: toff_boost <= 6'd37;
158       5'h0d: toff_boost <= 6'd35;
159       5'h0e: toff_boost <= 6'd32;
160       5'h0f: toff_boost <= 6'd30;
161       5'h10: toff_boost <= 6'd28;
162       5'h11: toff_boost <= 6'd27;
163       default: toff_boost <= 6'd0;
164    endcase
```

**Figure 70. Verilog specifying the LUTs used to produce $t_1$ and $t_2$ for the PFM converter.**
Verilog that counts the number of clock cycles specified by the LUTs for the gate signals.

For correct converter operation, the gate-drive signals need to be non-overlapping. A simple circuit depicted in Figure 72 was utilized using I/O pins of the CPLD and a Verilog implementation of the digital gates.

```verilog
//produce Non-Overlapping Signals
NonOverlapping NOGDS(
    .GateSignal(GateSignal),
    .GateSignal_N(GateSignal_N),
    .DELAY_IN(Delay_In),
    .DELAY_OUT(Delay_Out),
    .GateSignalSource(DriveSignalInt)
):

//Logic to introduce the PFM Aspect (both signals off during rest period)
assign DriveSignal = GateSignal;
assign DriveSignal_N = (state == gate_rest || state == zero_state) ? 0 : GateSignal_N;
//reg [4:0]time_counter;
//Timer Counter for Decisions
always @(posedge CLOCK or negedge CLOCK)
beginn
    if(!RESET_N) time_counter <= 13'b0;
    else if(state == zero_state) time_counter <= 13'b0;
    else if(time_counter == done_cycle) time_counter <= 13'b0;
    else time_counter <= time_counter + 13'b1;
eend
```

Figure 71. Verilog that counts the number of clock cycles specified by the LUTs for the gate signals.

Figure 72. The equivalent circuit that produces the delay to implement non-overlapping gate signals.
Figure 73. Verilog for the non-overlapping gate signals.

Table 11 provides other converter parameters worthy of attention:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inductance</td>
<td>470uH</td>
<td>Higher inductance allows system clock to be lower, saving power.</td>
</tr>
<tr>
<td>Inductor R₄₉</td>
<td>~ 0.66 Ω</td>
<td>TSL1112-471KR72-PF [17]</td>
</tr>
<tr>
<td>Inductor Rated Current</td>
<td>750mA</td>
<td>&quot;</td>
</tr>
<tr>
<td>System Clock</td>
<td>256 kHz</td>
<td>&quot;</td>
</tr>
<tr>
<td>P-Channel Rₐ₉</td>
<td>~ 0.45Ω at V₉ₐ = -4.5V</td>
<td>Diodes Inc. DMG1013T [18]</td>
</tr>
<tr>
<td>P-Channel Qₜ₉₉</td>
<td>~ 622pC at V₉₉₉ = -4.5V</td>
<td>&quot;</td>
</tr>
<tr>
<td>N-Channel Rₐ₉₉</td>
<td>~ 0.28Ω at V₉₉₉ = 4.5V</td>
<td>Diodes Inc. DMG1012T [19]</td>
</tr>
<tr>
<td>N-Channel Qₜ₉₉</td>
<td>~ 736pC at V₉₉₉ = 4.5V</td>
<td>&quot;</td>
</tr>
</tbody>
</table>

Care was taken to select parts that had low resistances to optimize efficiency. An analysis of efficiency and current regulation accuracy may be found in later sections. Figure 74 is the schematic for the bi-directional converter with the associated level shifters. The left side is the 'battery side' and the right side is the 'super capacitor side'. Note that pull-up and pull-down resistors were included for safety purposes (R48, R49, R50, R62), and are actually 1MΩ. Also, a single 10uF capacitor was used on either end of the converter, the second footprint is simply a placeholder.
Figure 74. Schematic for the bi-directional non-inverting buck-boost PFM current regulator.

Figure 75. Efficiency Plot for the Battery DC/DC Converter with fixed battery voltage, various super capacitor voltages (Vsc) and operating currents.
Figure 76. Efficiency Plot for the Battery DC/DC Converter with fixed super capacitor voltage, various battery voltages (Vsc) and operating currents.

Figure 77. Efficiency Plot for the Battery DC/DC Converter with a fixed battery voltage, various super capacitor voltages (Vsc) and operating currents.
Figure 78. Efficiency Plot for the Battery DC/DC Converter with fixed super capacitor voltage, various battery voltages ($V_b$) and operating currents greater than 3mA.

Figure 79. Accuracy Plot for the Battery DC/DC Converter with fixed super capacitor voltage, various battery voltages ($V_b$) and operating currents.
Figure 80. **Accuracy Plot for the Battery DC/DC Converter with fixed super capacitor voltage, various battery voltages (Vb) and operating currents.**
Appendix E - Microcontroller Top-Level Algorithm

C code - Top-Level Algorithm in the microcontroller that implements data collection and transmission in a particular order. Note the differences between day and night operation.

//Local Main Variables
int d_n = 0;//0:0 day, 1: night
int tx_delayed = 0;//0:0 transmission completed, 1: means transmission delayed
int gps_no = 0;//0:0 means acquired GPS data, 1: means did not
int already_one = 0;//0:1 means tx_delayed is already 1
int isday;//0:1 means day, 2: means night
int yes_day;

while(1)
{
  //###############DAY Function##############//
  do
  {
    d_n = 0;
    yes_day = 0;
    
    sensor_block(d_n,0,tx_delayed,already_one);
    energy_block(d_n,0,tx_delayed,already_one);
    
    gps_no = gps_block(tx_delayed,already_one);
    if(gps_no == 0)
    {
      gps_set[tx_delayed,already_one] = 1;
    }
    else if(gps_no == 1 && (tx_delayed,already_one == 0))
    {
      gps_set[tx_delayed,already_one] = 0;
    }
    sensor_block(d_n,1,tx_delayed,already_one);
    energy_block(d_n,1,tx_delayed,already_one);
    
    //Get Timestamp
    mRtccWaitSync();
    RttcReadTime(&TimeVal);
    hourBIN[tx_delayed,already_one] = (TimeVal.f.hour & 0b00001111) + ((TimeVal.f.hour & 0b11110000) >> 4)*10;
    minBIN[tx_delayed,already_one] = (TimeVal.f.min & 0b00001111) + ((TimeVal.f.min & 0b11110000) >> 4)*10;
    secBIN[tx_delayed,already_one] = (TimeVal.f.sec & 0b00001111) + ((TimeVal.f.sec & 0b11110000) >> 4)*10;
    
    if(tx_delayed == 1)
    {
      already_one = 1;
    }
    tx_delayed = xbee_block(tx_delayed,0);//0 for daytime
    if(tx_delayed == 0)
    {
      already_one = 0;
      gps_set[0] = 0;
      gps_set[1] = 0;
    }
  //************ASSUMPTION: tx_delayed will not be 1 three times in a row**********//
    isday = talk_cpld(1,RIsDay);//check if still daytime
    if(isday == 5)
    {
      yes_day++;
      Timer_Config(1);//1 minute
      while(!timer_up){};
      isday = talk_cpld(1,RIsDay);//check every 1 minute if it is day time
      if(isday == 5)
      {
        yes_day++;
      }
      else
      {
        yes_day = 0;
      }
    }
    else
    {
      yes_day = 0;
    }
  }while(yes_day != 2);//confirmed twice that it is day time
  //***********END DAY***********//
}
//NIGHT Function/////////////////////////////////////
  d_n = 1;
  night_counter = 0;
  yes_day = 0;

  while(night_counter < 16 && yes_day == 2)//collect data for a max of 16 hours
  {
    sensor_block(d_n,0,0);
    energy_block(d_n,0,0);

    //Get Timestamp
    mRtc.lWaitSync();
    RtccReadTime(&TimeVal);
    hourBIN[night_counter] = (TimeVal.f.hour & 0b00001111) + ( ( (TimeVal.f.hour & 0b11110000) >> 4)*10);
    minBIN[night_counter] = (TimeVal.f.min & 0b00001111) + ( ( (TimeVal.f.min & 0b11110000) >> 4)*10);
    secBIN[night_counter] = (TimeVal.f.sec & 0b00001111) + ( ( (TimeVal.f.sec & 0b11110000) >> 4)*10);

    night_counter++;
    isday = talk_cpld(1,RIsDay);//check if still nighttime
    if(isday == 6)
    {
      yes_day++;
      Timer_Config(1);//1 minute
      while(!timer_up){};

      isday = talk_cpld(1,RIsDay);//check every 1 minute if it is day time
      if(isday == 6)
      {
        yes_day++;
      }
      else
      {
        yes_day = 0;
      }
    }
    else
    {
      yes_day = 0;
    }
  }

  while(yes_day != 2)
  {
    Timer_Config(1);//1 minute
    while(!timer_up){};
    isday = talk_cpld(1,RIsDay);//check every 1 minute if it is day time
    if(isday == 6)
    {
      yes_day++;
    }
    else
    {
      yes_day = 0;
    }
  }

  }//confirmed twice that it is day time

  //END NIGHT#######################################

  //NIGHT-TO-DAY Function#############################
  xbee_block(0,1);//0 for no delay, 1 for night time...transmit the data collected at night

  //END NIGHT-TO-DAY################################
Appendix F - Communication Bus Command Details

This appendix outlines the various commands that are sent along the communication bus.

**Table 12. Summary of Communication Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Command ID</th>
<th>Received Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>SendKValue1</td>
<td>0x01</td>
<td>0d1</td>
</tr>
<tr>
<td>SendKValue2</td>
<td>0x02</td>
<td>0d2</td>
</tr>
<tr>
<td>SendKValue3</td>
<td>0x03</td>
<td>0d3</td>
</tr>
<tr>
<td>SendKValue4</td>
<td>0x04</td>
<td>0d4</td>
</tr>
<tr>
<td>ReceivePowerStatus</td>
<td>0x07</td>
<td>0d1 for yes power available 0d4 for no</td>
</tr>
<tr>
<td>SendAmuxBI</td>
<td>0x08</td>
<td>NA</td>
</tr>
<tr>
<td>SendAmuxSCV</td>
<td>0x09</td>
<td>NA</td>
</tr>
<tr>
<td>SendAmuxBV</td>
<td>0x0A</td>
<td>NA</td>
</tr>
<tr>
<td>SendAmuxPV</td>
<td>0x0B</td>
<td>NA</td>
</tr>
<tr>
<td>SendAmuxDone</td>
<td>0x0C</td>
<td>NA</td>
</tr>
<tr>
<td>ReceiveIsDay</td>
<td>0x0D</td>
<td>0d2 for day, 0d5 for night</td>
</tr>
</tbody>
</table>

*Note that commands 0x5 and 0x6 are no longer implemented.*

The first four commands tell the CPLD which of the four built-in scaling factors it should use when determining the voltage for the MPPT algorithm; that voltage is sent to the digital potentiometer and onto the Solar Boost Converter. The ReceivePowerStatus command asks the CPLD if the super capacitor voltage is high enough to support a high-power operation such as acquiring coordinate data through the GPS, or transmitting collected data through the Xbee. The next five commands are used so that the microcontroller can capture the same energy data that the CPLD normally collects. The CPLD is instructed to make a certain voltage or current available through its Signal Chain which also connects to one of the microcontroller's analog ports. The last command asks the CPLD if it is 'daytime' or 'nighttime' in regards to the current open-circuit voltage on the pilot cell, which depends on incoming light. The microcontroller will switch over to day or night operation if the CPLD provides two consecutive replies in the positive.
Appendix G - Power Gating

This appendix provides supplementary images relating to power gating. Immediately below is the schematic page for the sensors. Note the mosfet inline with the VDD rail for the humidity and temperature sensors. This mosfet is controlled by the gate signal Sensor_Shutdown.

![Sensor Schematic Page](image)

Figure 81. Sensor Schematic Page.

The two screenshots below illustrate that the power gating function works correctly. The capture on the left is of power gating off, meaning that the p-channel mosfet is closed (confirmed as the purple gate signal is low). Notice that the three other signals are non-zero, as they represent the outputs of the three sensors controlled by Sensor_Shutdown. The capture on the right is of power gating on, meaning that the p-channel mosfet is open (confirmed as the purple gate signal is high). The three output signals are now all zero, as they have been isolated from the 3.3V rail.

![Capture with Power Gating Off](image)

Figure 82. Capture with Power Gating Off.

![Capture with Power Gating On](image)

Figure 83. Capture with Power Gating On
Appendix H – Solar Panel Background

In the following figure, a number I-V characteristics are depicted for a higher powered solar panel. Under varying irradiance conditions, the panel’s short circuit current varies proportionally with incident light while the open circuit voltage remains relatively constant.

The dashed curves illustrate the Power-Voltage relationship of the panel at varying irradiance conditions and the panel voltage corresponding to the peak power point changes minimally, by about 10%. This relatively constant maximum power point operating voltage, paired with the open circuit voltage serves as a foundation for the open-circuit voltage Maximum Power Point Tracking algorithm being implemented in the WSN design.

Figure 84. Depicting the I-V and P-V solar characteristics for varying irradiance levels [6].
**Appendix I – Power Management Logic Chart**

This chart outlines the relationship between WSN operating mode and the voltages of the Pilot Cell, Super Capacitor, and Battery.

**Table 13. Power Management Logic as Dictated by Device Voltages.**

<table>
<thead>
<tr>
<th>Operating Mode</th>
<th>Pilot Cell Voltage</th>
<th>Super Capacitor Voltage</th>
<th>Boolean Condition</th>
<th>Battery Voltage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal Day</td>
<td>&gt; 1.0V</td>
<td>&gt; 4.0 V</td>
<td>AND</td>
<td>&gt; 3.5 V</td>
</tr>
<tr>
<td>Normal Night</td>
<td>&lt; 1.0 V</td>
<td>&gt; 4.0 V</td>
<td>AND</td>
<td>&gt; 3.5 V</td>
</tr>
<tr>
<td>Daytime Charge Battery</td>
<td>&gt; 1.0 V</td>
<td>START: &gt; 4.7V STOP: &lt; 4.5 V</td>
<td>AND</td>
<td>&lt; 4.1 V</td>
</tr>
<tr>
<td>Daytime Charge Super Capacitor</td>
<td>&gt; 1.0 V</td>
<td>START: &lt; 4.1V STOP: &gt; 4.3V</td>
<td>AND</td>
<td>&gt; 3.0 V</td>
</tr>
<tr>
<td>Daytime Converter Off</td>
<td>&gt; 1.0 V</td>
<td>4.3 – 4.5 V</td>
<td>OR</td>
<td>&lt; 3.0 V</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&gt; 5.0 V</td>
<td>OR</td>
<td>&gt; 4.1 V</td>
</tr>
<tr>
<td>Night Charge Super Capacitor</td>
<td>&lt; 1.0 V</td>
<td>START: &lt; 4.5 V STOP: &gt; 4.7 V</td>
<td>AND</td>
<td>&gt; 3.0V</td>
</tr>
<tr>
<td>Night Converter Off</td>
<td>&lt; 1.0 V</td>
<td>&gt;4.7 V</td>
<td>OR</td>
<td>&lt; 3.0 V</td>
</tr>
<tr>
<td>Emergency</td>
<td>ANY</td>
<td>&lt; 4.0 V</td>
<td>OR</td>
<td>&lt; 3.5 V</td>
</tr>
</tbody>
</table>
Appendix J – Power Measurement Captures

The following captures were made while performing power measurements on the system-level. The output of the 3.3V buck converter was measured since this is the primary system rail that drives almost all the devices.

Figure 85. The power measurements for various system conditions listed in the Assessment of Final Design section.
Appendix K – Long-Term Test Results

The following graph represents the electrical data collected during the nine day test.

Figure 86. A zoomed-in version of the waveforms for the nine-day test.