WEATHER SENSOR NETWORK

A THESIS SUBMITTED TO THE GRADUATE DIVISION OF THE UNIVERSITY OF HAWAI‘I AT MĀNOA IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

ELECTRICAL ENGINEERING

AUGUST 2015

By

Andy H.N. Pham

Dissertation Committee:

Anthony Kuh, Chairperson
David Garmire
Matthias Fripp

Keywords: Weather station, weather data, and anemometer
ACKNOWLEDGMENTS

I want to acknowledge the support of the Renewable Energy and Island Sustainability (REIS) Center and the Department of Energy (DOE) grant DE-OE0000394 for supporting my research on energy and sustainability projects. I also want to thank my advisor, REIS director, Dr. Anthony Kuh for mentoring and supporting my leadership decisions over the years.

I started working in in the Smart Campus Energy Lab (SCEL) in REIS as the project leader of a group of eight students working on a weather sensor network. It was my first time leading such a large team without any experience in the sustainable field. I want to thank all my previous and current teammates for being patient and gave me time to adapt to my role. I could not lead the SCEL team to where it is now without my leadership team; Kenny Luong (software lead/system engineer), Zachary Dormen (technical lead), and Christie Obatake (new project manager).

I would like to thank my thesis committee Dr. David Garmire and Dr. Matthias Fripp for their time and help throughout the last couple of years.

Finally, I would like to thank everybody who supported me throughout this work by way of extending kind hands, whenever necessary.
TABLE OF CONTENTS

Acknowledgments .................................................................................. 2

List of tables ........................................................................................... 6

List of figures .......................................................................................... 1

1 Introduction and Motivation ................................................................. 2
   1.1 Renewable Energy ......................................................................... 2
   1.2 Renewable Energy in Hawai‘i ...................................................... 5
   1.3 Renewable Energy and Island Sustainability ............................... 6
   1.4 Smart Campus Energy Lab ......................................................... 6
   1.5 Thesis Overview ......................................................................... 7
   1.6 Contribution ............................................................................... 8

2 Weather Sensor Network ..................................................................... 9
   2.1 Project Overview .......................................................................... 9
   2.2 Commercial Weather Module ................................................. 10
   2.3 Apple Weather Sensor Module ............................................... 13
      2.3.1 Cost Breakdown ................................................................. 16
      2.3.2 Sensors ............................................................................. 16
      2.3.3 Housing ............................................................................ 19
      2.3.4 Power System ................................................................. 19
      2.3.5 Testing and Debugging ...................................................... 20
   2.4 Cranberry Weather Sensor Module .................................. 20
      2.4.1 Cost Breakdown ................................................................. 21
      2.4.2 Sensors ............................................................................. 22
      2.4.3 Modules Sensors Comparisons ..................................... 23
3 Kiwi Anemometer

3.1 Introduction and motivation

3.1.1 Cup and Windmill Anemometers

3.1.2 Hot Wire Anemometer

3.1.3 Laser Doppler Velocimetry

3.1.4 Ultrasonic Anemometer

3.1.5 University of Colorado Boulder Passive Acoustic Anemometer

3.2 Kiwi Passive Acoustic Anemometer

3.2.1 Project Overviews

3.2.1.1 Capture Wind Speed and Wind Generator

3.2.1.2 Capture Wind Speed

3.2.1.3 Wind Generator

3.2.1.4 Capture Microphone Signal Experiment Setup

3.2.1.5 View Microphone Raw Data

3.2.1.6 Algorithm Development

3.2.1.7 Peak Detector

3.2.2 Peak Detector in Software

3.2.3 Peak Detector With First Order LPF

3.2.4 Peak Detector With Second Order Butterworth LPF

3.2.5 Correlate Peak Detector Output With Wind Speeds

3.2.6 Wind Speed Invariant With Changing Distance Experiment

3.2.7 Noise Sensitivity Experiment

3.2.8 Implement Algorithm on Arduino UNO

3.2.9 Kiwi Anemometer Demo
LIST OF TABLES

2.1 Apple module cost breakdown .................................................. 16
2.2 Cranberry module cost breakdown ............................................. 22
3.1 Equipment and part required for collecting the raw data from the microphone ... 34
3.2 Equipment and part required for collecting the raw data from the microphone ... 37
3.3 Kiwi anemometer constant wind speed invariant with distance test result .......... 55
3.4 Kiwi anemometer sensitivity to additive traffic noise ........................ 56
3.5 Kiwi anemometer sensitivity to additive rain noise ........................ 56
LIST OF FIGURES

1.1 Solar power world total power capacity ........................................... 3
1.2 Wind power world total power capacity ........................................... 4

2.1 WaspSense sensor module ............................................................... 10
2.2 Vantage Pro2 weather station ........................................................... 11
2.3 Texas Weather Instruments WR-25 ................................................... 11
2.4 Campbell Scientific Watherhawk 916 .............................................. 12
2.5 HOBO U30-NCR weather station ..................................................... 13
2.6 Justin Carland sensor module .......................................................... 14
2.7 Typical I2C communication bus ....................................................... 14
2.8 I2C communication bus with discrete devices .................................... 15
2.9 Apple module dimension ................................................................. 15
2.10 Apogee SP-110 .............................................................................. 17
2.11 BMP085 barometric pressure sensor .............................................. 17
2.12 SHT11 Sensirion digital humidity/temperature sensor ....................... 18
2.13 MTK3339 GPS ............................................................................. 18
2.14 Sensor module mounted on building pillar ....................................... 19
2.15 Cranberry bottom PCB stack ......................................................... 20
2.16 Cranberry top PCB stack ............................................................... 21
2.17 MPL115A2 barometer ..................................................................... 23
2.18 Humidity/temperature sensor HIH6031 ......................................... 23
2.19 Battery for previous sensor modules (left), and Cranberry battery (right) 24
2.20 Hub-and-spoke network model for long-distance data relaying ............ 25

3.1 Cup anemometer ............................................................................. 28
3.2 Windmill anemometer .................................................................... 28
3.3 Hot-wire anemometer ......................................................... 29
3.4 Laser Doppler velocimetry measuring air flow through a duct ............ 30
3.5 Ultrasonic anemometer .................................................... 31
3.6 A-E are the omnidirectional microphone, and M is the weather station use to validate the experiments result ....................................................... 32
3.7 Receiver configuration ...................................................... 34
3.8 Omnidirectional microphone ............................................. 35
3.9 Arduino UNO software used to capture data .......................... 36
3.10 Row 1 is wind speed in MPH, and the columns are raw data .......... 37
3.11 Wind generator setup ...................................................... 38
3.12 Experiment set up diagram .............................................. 38
3.13 Microphone and windmill anemometers setup ........................ 39
3.14 Experiment set up .......................................................... 39
3.15 Matlab code used to plots raw data .................................. 40
3.16 Microphone analog data with varying wind speeds with wind generator (fan) 18 inches from the microphone ................................................... 40
3.17 Full bridge rectifier schematic ......................................... 41
3.18 Equivalent system block diagram of full bridge rectifier ............... 42
3.19 Input sinusoid in red plot, and output across R1 in green plot .......... 42
3.20 Peak detector diagram ...................................................... 43
3.21 Kiwi modify peak detector .............................................. 43
3.22 Raw analog data of wind speed ranges from (0-15.3)MPH ................. 44
3.23 Implemented peak detector system of figure 3.21 in Matlab ............... 45
3.24 Peak detector output with alpha equal 0.5 ................................ 46
3.25 Peak detector output with alpha equal 0.01 ................................ 47
3.26 Peak detector output with alpha equal 0.001 ................................ 48
3.27 Peak detector output with alpha equal 0.0001 ................................ 49
3.28 Output of peak detector with $W_n$ equal 0.0001$\pi$ rad/sample ............... 50
3.29 Wind speeds verse peak detector output ............................................. 52
3.30 Wind speeds verse square root peak detector output .............................. 53
3.31 linear regression on the square root of peak detector output ................. 54
3.32 Constant wind speed vs. varying distance between wind source and anemometer . 55
3.33 Arduino code that measures wind speed in real time ............................. 58

4.1 Kiwi anemometer with addition amplifier .............................................. 60
4.2 Measures wind direction ................................................................. 61
1 INTRODUCTION AND MOTIVATION

1.1 Renewable Energy
Humans have been using fossil fuels as their primary energy source since coal was first discovered in the second and third centuries (100-200 AD). The energy from fossil fuels are used in pretty much everything in our daily lives; from charging cell phone batteries, washing machines and driers, to the light bulb in your office. However, fossil fuels are a finite source that is being rapidly being depleted, there are serious environmental concerns with fossil fuels, and fossil fuels are the leading cause of climate change.

One solution to reduce the dependency on fossil fuels is renewable energy. Renewable energy is energy source that is not depleted when used. 19% of the global total energy generation in 2012 comes from renewable energy. The common renewable energy sources are solar, wind, geothermal, hydro, and biomass. The total power capacity of these five sources is 1,721GW [1]. Figure 1.1 shown the solar power capacity is growing exponentially over the years. Wind power capacity is also growing exponentially shown in figure 1.2. Solar and wind energy are dependent on weather conditions and the time of day making these two energy sources intermittent. The power generation of these two sources fluctuates through out the day. Intermittent renewable sources can cause problems as the fundamental equation for the electric grid is that the amount of energy generated must equal the amount of energy used (load). This becomes more difficult with intermittent energy resources.

In 2014, the top five countries leading in solar power capacity are Germany (38GW), China (28GW), Japan (23GW), Italy (18GW) and the United States (18GW) [2]. Photovoltaic was first discovered in the 1830s. Photovoltaic materials emit energy when exposed to light. The modern solar panel consists of photovoltaic (PV) cells connected together in an array and can achieve an
energy efficiency as high as 46% [3]. The solar panel that are used for residential and commercial energy sources have an average efficiency of about 17.4%. Solar panel efficiency has been increasing and costing less as technology has advanced. The five leading countries in solar power make up about 70% of the total 180GW world solar power capacity.

In the same year China is leading the world in wind power capacity with 115GW, the United States is next with 66GW, Germany is the third with 39GW, Spain is the fourth with 23GW, and India is the fifth with 22GW [4]. The five leading countries in wind power made up about 72% of the total 367GW world wind power capacity.

Figure 1.1: Solar power world total power capacity
Figure 1.2: Wind power world total power capacity

- China: 114,763 MW (31.1%)
- United States: 65,879 MW (17.8%)
- Germany: 39,165 MW (10.6%)
- Spain: 22,987 MW (6.2%)
- India: 22,465 MW (6.1%)
- United Kingdom: 12,440 MW (3.4%)
- Canada: 9,694 MW (2.6%)
- France: 9,285 MW (2.5%)
- Italy: 8,663 MW (2.3%)
- Brazil: 5,939 MW (1.6%)
- Rest of the world: 58,275 MW (15.8%)
In 2013, the top five countries leading in hydropower capacity are China (280GW), Brazil (86GW), United States (78GW), Canada (76GW) and the Russia (47GW) [1]. The five leading countries in hydropower make up about 54.7% of the total 1,000GW world hydropower capacity.

In 2013, 100GW power capacity is from bio-power. The top five countries leading bio-power are mostly United States, Germany, China, India, and Italy [1].

In 2014, 74GW power capacity is from geothermal power. The top five countries leading bio-power are mostly United States, Philippines, Indonesia, Mexico, and Italy [5].

1.2 Renewable Energy in Hawai’i

Hawaii is highly dependent on the importation of energy. In 2012, 93% of the energy that the state of Hawaii consumed was imported. In addition to this, a majority of electricity was generated from petroleum. On December 2014, 529GWh of electricity was generated from petroleum, whereas only 102GWh was created from renewable energy sources [6]. Due to Hawaii’s heavy reliance on imported oil, it has the highest electricity prices per kWh in the nation. In January 2015, residents of Hawaii paid an average of 33.34 cents per kWh, which is over 20 cents above the national average [7]. In the future, our reliance on fossil fuels could be problematic due to scarcity of these resources, prices, and environmental concerns.

In 2008, the Department of Energy (DOE) and the State of Hawaii governor launched the Hawaii Clean Energy Initiative (HCEI), with the goal of achieving 70% clean energy by 2030 [8]. 30% will come from energy efficiency and conservation while 40% will come from renewable energy sources. More recently, in June 2014, the State of Hawa’i had set a higher goal of 100% renewable electricity generation by 2050 [9]. Furthermore, the State of Hawa’i legislature recently passed a bill, call for 100% renewable energy by 2045. Governor David Ige signed the bill in June, 2015 [10].
1.3 Renewable Energy and Island Sustainability

Renewable Energy & Island Sustainability (REIS) was established in 2009 at the University of Hawaii at Manoa (UHM). REIS is multidisciplinary team of faculty and students from engineering, social sciences, natural science, architecture, ocean and earth technology, tropical agriculture and human resources, law, and business. REIS works on cutting edge research and education problems in renewable energy and island sustainability. The initial REIS funding was from a $1M UHM sustainability grant. In 2010 the REIS Center received a $2.5M grant from the Department of Energy (DOE) on workforce training in Strategic Training and Education in Power Systems (STEPS).

1.4 Smart Campus Energy Lab

In fall 2013 I joined REIS as a researcher assistant under Dr. Anthony Kuh. My role was to lead the Smart Campus Energy Lab (SCEL) engineering team and advance the research on energy efficiency and sustainability.

Since 2013 the SCEL has grown from six engineering students to more than 20 engineering students in fall 2015. When I started in Fall, 2013 SCEL has one project, the Weather Sensor Network. SCEL has one project (the Weather Sensor Network) when I started. There are now three undergoing projects and we anticipate more projects for this Fall, 2015.

The Weather Sensor Network and Solar Irradiance Forecasting project is an ongoing project that started in 2012. The goal of the project is to monitor environmental conditions on the UHM campus so that we are in a better position to forecast future distributed renewable energy sources (i.e. solar PV) as it is deployed on campus. We deployed sensor nodes (weather sensor node boxes) on the rooftop of Holmes Hall to collect solar irradiation, relative humidity, ambient temperature, and barometric pressure data. Adaptive filter algorithms were applied to the data to forecast solar
The University of Agder in Norway also had similar research on forecasting solar irradiance for renewable energy application [11].

The Kiwi Passive Acoustic Anemometer project is an ongoing project that started at the beginning of 2015. The project goal is to design a small and low power consumption acoustic anemometer. We also want to see if it is possible to use a microphone to measure wind speed. The anemometer could be used in application to monitor building air ventilation.

The Tree Monitoring project is an ongoing project that started at the beginning of 2015. The tree monitoring system measures the soil moisture and make sure there is enough water for the young tree to grow. The project’s goal is to intelligently monitor important environmental variables associated with a tree.

1.5 Thesis Overview

Chapter 1 gives a summary of renewable energy sources with a focus on solar and wind energy. We discuss their intermittency and give current solar and wind generation capacity around the world. We then discuss the energy landscape in Hawaii along with the Hawaii Clean Energy Initiative and goals of achieving 100% renewable energy by 2045. The Chapter also briefly discusses the Renewable Energy and Island Sustainability (REIS) group and the Smart Campus Energy Lab (SCEL).

The two projects that will be discussed in the thesis are the Weather Sensor Network and Kiwi Passive Acoustic Anemometer. Chapter 2 covers the high level designs of Apple sensor module and the Cranberry sensor module of the Weather Sensor Network. Chapter 3 discusses both the design process and implementation of the Kiwi passive acoustic anemometer. Chapter 4 summarizes this thesis and suggests directions for further research.
1.6 Contribution

For the past two years I supervised the Weather Sensor Network project where I managed all: the tasks for both Apple and Cranberry revisions. I contributed in the selecting the necessary electronic components and PCB design. I also worked on making the communication protocol to the sensor module more robust. I also worked on the power system to make it longer lasting so that it would work during the rainy season when there were more cloudy days. One of the most challenging parts of the Apple design was the housing. We could not use the housing from previous revisions because water would leak into the sensor module and corrode the electronic components. I put together the housing for the Apple module and this solution using plastic components for the enclosure is still being used today.

My contribution is 90% of the work on the Kiwi anemometer. I developed the algorithm to measure wind velocity. I also worked on all the experiments: 1) Wind speed invariance with changing distance, and 2) Noise Sensitivity.
2 WEATHER SENSOR NETWORK

2.1 Project Overview

The objective of the project is to design a sensor network composed of environmental weather sensor network boxes to be deployed on rooftops around the UHM campus. Our goal is to eventually deploy about 50 to 100 sensor node boxes on UHM building rooftops to get good spatial and temporal resolution of UHM environmental conditions. Each node in the network is an autonomous self powered module that measures solar irradiance, barometric pressure, humidity, and ambient temperature. The sensors data is transmitted to the server in 493 Holmes Hall where the data is stored, and processed. The most important sensor is the solar irradiance (Watts per meter squared). The solar irradiance is used to estimate photovoltaic (PV) power generation.

Online adaptive filter algorithms; steepest decent (SD), least mean squares (LMS), normalized least mean squares (NLMS), and recursive least squares (RLS) can be used to forecast solar irradiance [12]. In forecasting solar irradiance it is also important to get proper representations of data. In earlier work we used the zenith angle as input to forecast solar irradiance [13]. Accurate forecasts help with grid stability as we are better able to anticipate large surges and drops in solar PV energy.

This chapter will discuss the sensor module hardware, the high level design, and fabrication overview. Commercial weather sensor modules are available to be purchased for at least $650. The price of the commercial sensor module does not fit in the budget. The module was designed and assembled in house by engineering students to have low cost. The design has undergone two modifications in the last two years. The module is self powered by a rechargeable lithium ion battery drawing power from a small solar PV panel. The power system of the first design iteration was not able to support the module when sun light was not available for a long period. The power
issue got addressed in the second design iteration. The first design iteration was named Apple and the second design iteration was named Cranberry.

2.2 Commercial Weather Module

In this subsection we review some commercially available weather sensor boxes, their sensors, and costs. This is to give a comparison with the costs and capabilities of our weather sensor box. Libelium WaspSense base costs $800 without the mounting part. The Libelium has 16 types of plug and play sensors available for purchase. The module can have up to 6 sensors plugged in at a time. Its optional for the module to have solar panel power. WaspSense does not have a solar irradiance sensor [14].

![Figure 2.1: WaspSense sensor module](image)

David Instruments Vantage Pro2 costs $650 without the mounting part. The module has rainfall, barometric pressure, temperature/humidity, and anemometer sensors. Its optional for the module to have a solar panel power [15].
Texas Weather Instruments WR-25 costs $1,200 without the mounting part. The module has wind speed, wind direction, barometric pressure, internal temperature, external temperature, humidity, and rainfall sensors. The module is powered by plugging into electrical outlet [16].
Campbell Scientific Watherhawk 916 costs $2,400 without the mounting part. The module has wind speed, wind direction, solar irradiance, rainfall, barometric pressure, humidity sensors. The module is self powered by battery and solar panel.

Onset Computer HOBO U30-NCR costs $1,400 without the mounting part. The module standard package has temperature, relative humidity, dew point, rainfall, soil moisture, solar radiation, wind speed and direction, barometric pressure sensors. The module is self powered by solar panel and battery [17].
2.3 Apple Weather Sensor Module

In 2013, Justin J. Carland M.S. Electrical Engineering student at the UHM designed and built a weather sensors module that measures barometric pressure, humidity, temperature, and luminosity. The sensors data is transmitted to the server in 493 Holmes Hall where the data is stored and analyzed. The module is self powered by a rechargeable lithium ion battery and a small solar PV panel [18]. The module was made of discrete devices shown in figure 2.6. Most of the discrete devices used in the module had an inter-integrated circuit (I2C) built with pulled up resistors on the serial data line (SDA) and the serial clock line (SCL). Integrating those devices in the same I2C communication bus resulted in large power consumption.
The typical I2C communication bus diagram is shown in figure 2.7. The SDA and SCL lines each have one pull up resistor. The values of these resistors are dependent on the length of the bus, and how many devices are on the bus. Figure 2.8 diagram shows the I2C communication bus with the discrete devices. The more devices on the I2C bus, the lower the SDA and SCL pulled resistors resistance, thus higher power consumption.
An improved module design named Apple was made in 2013. The module has lower power consumption by removing unnecessary resistors in the I2C bus. The module was designed with printed circuit board (PCB) in Eagle CAD, resulting in a smaller size. Figure 2.9 is the dimension of the Apple module electronic. The module used the same Xbee Pro transceiver and the communication protocol of the previous module [18].
### 2.3.1 Cost Breakdown

Table 2.1 is the cost break down of the Apple module. The total cost does not include the labor cost. The cost per module came out to be $524.05.

<table>
<thead>
<tr>
<th>Description</th>
<th>Quantity</th>
<th>Unit Cost (USD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arduino UNO - R3</td>
<td>1</td>
<td>29.95</td>
</tr>
<tr>
<td>XBee Pro 63mW RPSMA - Series 2B</td>
<td>1</td>
<td>44.95</td>
</tr>
<tr>
<td>INA219 High Side DC Current Sensor Breakout</td>
<td>1</td>
<td>9.95</td>
</tr>
<tr>
<td>Sensirion Temperature/Humidity Sensor</td>
<td>1</td>
<td>35.00</td>
</tr>
<tr>
<td>BMP085 Barometric Pressure/Temperature/Altitude Sensor</td>
<td>1</td>
<td>19.95</td>
</tr>
<tr>
<td>NCP1402-5V Step-Up Breakout</td>
<td>1</td>
<td>5.95</td>
</tr>
<tr>
<td>Op Amp - Dual Rail-to-Rail - 2.7-12V power @ 80mA output</td>
<td>1</td>
<td>2.95</td>
</tr>
<tr>
<td>Pyronometer Apogee SP-110</td>
<td>1</td>
<td>169.00</td>
</tr>
<tr>
<td>Large 6V 3.4W Solar panel - 3.4 Watt</td>
<td>2</td>
<td>69.00</td>
</tr>
<tr>
<td>USB / DC / Solar Lithium Ion/Polymer charger - v1.0</td>
<td>1</td>
<td>24.95</td>
</tr>
<tr>
<td>Lithium Ion Battery Pack - 3.7V 6600mAh</td>
<td>1</td>
<td>39.50</td>
</tr>
<tr>
<td>Housing</td>
<td>1</td>
<td>15.00</td>
</tr>
<tr>
<td>Clamps</td>
<td>1</td>
<td>50.00</td>
</tr>
<tr>
<td>PCB</td>
<td>1</td>
<td>7.00</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td><strong>524.05</strong></td>
</tr>
</tbody>
</table>

Table 2.1: Apple module cost breakdown

### 2.3.2 Sensors

The pyranometer (solar irradiance sensor) used in the sensor module is the Apogee SP-110 shown in figure 2.10. The pyranometer is self-powered, and has a sensitivity of $0.20 \text{ mV} / (\text{Watt} \times \text{m}^{-2})$. 
The module Arduino UNO has a 10 bit analog to digital converter (ADC) with a 4.88 mV resolution to convert the analog solar irradiance signal to digital.

![Figure 2.10: Apogee SP-110](image1)

The barometric pressure sensor, Bosch BMP085 with I2C data transfer is shown on figure 2.11. The sensor measurement ranges from (300-1,100) hPa with an accuracy of 1.0 hPa and a resolution of 0.01 hPa. The incorporated temperature sensor has a range of $(-40 \ to \ 85)^\circ C$ with an accuracy of $1^\circ C$ [20].

![Figure 2.11: BMP085 barometric pressure sensor](image2)
The humidity sensor used in the module was Sensirion SHT11 with I2C data transfer is shown in figure 2.12. The sensor measures relative humidity from (0−100)% with an accuracy of 3.5% and a resolution of 0.03%. The sensor also measures temperature with a range of (−40 to 123.8)°C with an accuracy of 2°C and resolution of 0.01°C. The sensor was powered by 5V supply and consumes 550µA during measurement [21].

Figure 2.12: SHT11 Sensirion digital humidity/temperature sensor

MTK3339 GPS module with UART data transfer manufactured by GlobalTop Technology was used to get location and current time. The GPS was powered by 5V and draws an average of 22.5mA. The GPS module can track up to 22 satellites and the update ranges are (1 10)Hz. Position accuracy is within 1.8 meters [22].

Figure 2.13: MTK3339 GPS
2.3.3 Housing

The module electronic and battery were fitted inside a project box $2 \times 3 \times 6\text{in}^3$ from RadioShack. The solar panels are held to the project box by 90 degree aluminum bars. The module is held to building structure, like the building pillar shown in figure 2.14, with a 600 pounds grip clamp.

Figure 2.14: Sensor module mounted on building pillar

2.3.4 Power System

The module is powered by a 3.7V 6,600 mAh rechargeable Li-ion battery. The battery charges with dual solar panels with a total of 6 W at 6 V. The Apple module current draw is 130 mA at full operation mode, which is 50 mA less than the previous module. There is one digital switch in the Apple module to turn off the power of the sensors when the battery is low. The current draw is 70mA when the sensors are off. The switch cuts off the power when the battery is below 30% (3.7V).
2.3.5 Testing and Debugging

The Apple module was deployed onto Holmes Hall roof at the beginning of July, 2013. The module was mounted onto one of the pillars shown in figure 2.14. We tested the modules and the longest uninterrupted service was three months. Service stopped as the base station lost communication with the module as the battery discharged to below 30% as there was no sun to charge the battery for four days due to a prolonged storm producing cloudy weather. The Apple module was not designed to get through long cloudy periods that occur in the winter time.

2.4 Cranberry Weather Sensor Module

The Cranberry module was designed to improve the power system of the Apple module, and reduce cost and size. The communication protocol was also improved. The user can set the packet lengths that are wirelessly transmitted from the module to base station. The module is made of two PCBs $2 \times 2in^2$ shown in figure 2.15 and figure 2.16. The bottom PCB has the microcontroller, Xbee, battery charger, and the power switches. The bottom PCB is enclosed into the housing to protect it from the environment. The top PCB has all the weather sensors exposed to the environment. The top PCB is easy to replace if a sensor is damaged.

![Cranberry bottom PCB stack](image)

Figure 2.15: Cranberry bottom PCB stack
2.4.1 Cost Breakdown

Table 2.2 is the cost break down of the Cranberry module. The total cost does not include the labor cost. The cost per module came out to be $402.36.
<table>
<thead>
<tr>
<th>Description</th>
<th>Quantity</th>
<th>Unit Cost (USD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>XBee Pro 63mW RPSMA - Series 2B</td>
<td>1</td>
<td>44.95</td>
</tr>
<tr>
<td>Relative humidity/temperature sensor</td>
<td>1</td>
<td>9.89</td>
</tr>
<tr>
<td>Barometric Pressure Sensor</td>
<td>1</td>
<td>4.59</td>
</tr>
<tr>
<td>Pyronometer Apogee SP-110</td>
<td>1</td>
<td>169.00</td>
</tr>
<tr>
<td>Large 6V 5.6W Solar panel Solar panel - 3.4 Watt</td>
<td>1</td>
<td>49.95</td>
</tr>
<tr>
<td>Solar Panel Charging Chip</td>
<td>1</td>
<td>1.49</td>
</tr>
<tr>
<td>Lithium Ion Battery Pack 3.7V 15,600mAh</td>
<td>1</td>
<td>40.49</td>
</tr>
<tr>
<td>Housing</td>
<td>1</td>
<td>15.00</td>
</tr>
<tr>
<td>Mount</td>
<td>1</td>
<td>50.00</td>
</tr>
<tr>
<td>PCB</td>
<td>1</td>
<td>7.00</td>
</tr>
<tr>
<td>Cap, Res, switch, and etc</td>
<td></td>
<td>10.00</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td><strong>402.36</strong></td>
</tr>
</tbody>
</table>

Table 2.2: Cranberry module cost breakdown

### 2.4.2 Sensors

The pyranometer (solar irradiance sensor) used in the sensor module is the Apogee SP-110 shown in figure 2.10. The pyranometer is self-powered, and has a sensitivity of 0.20 mV per Watt m^2 [19]. The module Arduino UNO has a 10 bit analog to digital convertor (ADC) with a 4.88 mV resolution to convert the analog solar irradiance signal to digital.

The barometer pressure sensor, Freescale Semiconductor MPL115A2, is shown in figure 2.17. The sensor can measure from (50115)kPa with an accuracy of 1 kPa [23].
The humidity/temperature sensor, Honeywell HIH6031, is shown in Figure 20. The sensor measures relative humidity from (0 - 100)% with an accuracy of 4.5% RH and a resolution of 0.04%. The sensor also measures temperature with a range of $(5 - 50)^\circ C$ with an accuracy of $1^\circ C$ and resolution of $0.025^\circ C$ [24].

2.4.3 Modules Sensors Comparisons

The major sensors for the sensor network are pyranometer, barometric pressure, ambient temperature, and humidity. The Apple module, Cranberry module, Weatherhawk 916, and the HOBO U30-NCR have the same pyranometer, Apogee SP-110. Both Apple and Cranberry modules have better barometric pressure measurement accuracy than other the commercial weather stations mentioned in this chapter. The Weatherhawk 916 weather station has the best temperature measure-
ment accuracy of 0.5°C. Both the Apple and Cranberry modules have a temperature measurement accuracy of 1°C. The Apple module has the highest relative humidity measurement accuracy of 3.5%. The next highest is the Cranberry module with 4.5%.

2.4.4 Power System

The Cranberry module had an improved Li-ion battery 3.7V 15,600mAh shown in figure 2.19. The Cranberry module battery capacity is twice that of the Apple module. The battery is charged by a single 6V 5.8W solar panel. The MCP7381-2CAI/ML IC was used as the battery charger [25]. The module is run on 3.3V. The module had two IC power switches to turn on/off the sensors power bus and the Xbee transceiver. When all sensors and Xbee are used the total current drawn is less than 10mA.

![Figure 2.19: Battery for previous sensor modules (left), and Cranberry battery (right)](image)

2.4.5 Network and Communication Protocol

The mesh network graph was formed automatically by the coordinator node. Some nodes will have higher power consumption than others due to large amounts of relay data. This causes power instability in the network [26].
The network configuration in figure 2.20 was proposed to solve the power instability of the mesh network. There are three different types of nodes in the network. The first node type is the home station. Home station is where the server is located in Holmes Hall 493. All the data will be relayed back to be stored in the server. The second type of node is the base station. The base station manages its own local network and relays the data directly back to the home station. The base station is either self powered with a large power system or plug in. The third type of node is the sensor node. The sensor node collects weather data and only communicates with its own local base station. The sensor node does not need to relay other node data; therefore the sensor node can put its transceiver in sleep mode during data sampling to save power. This network configuration was tested in 2014, and we demonstrated its viability. A base station was deployed at the Hokulani Elementary School, about one mile away from the home base station. A single sensor node was deployed on top of a building close to the base node. The experiment duration was two months.

Figure 2.20: Hub-and-spoke network model for long-distance data relaying
The sensors sampling rate was limited to the Xbee maximum package size of 84 bytes. Integrating additional sensors into the module in the future may result in change of the packet size. The Cranberry module communication protocol divides larger amounts of data into smaller packet sizes for transmission. Each packet gets transmitted and gets stitched together at the destination [27]. Sensor sampling rate can be changed over the air by sending a request command.
3 KIWI ANEMOMETER

3.1 Introduction and motivation

An anemometer, or also known as wind sensor, is a device use to measure wind velocity and direction. Anemometer can be used in many applications, but the most common use is for weather station instrument and wind turbines. Acoustic, photon, heat & resistance, and mechanical are used in anemometer to measure wind velocity and direction. In this chapter I will present a passive acoustic anemometer named Kiwi that I designed. Kiwi anemometer is small and light, low power consumption, fast response, and does not have moving parts. Kiwi anemometer could be used in applications where space is limited. House and building ventilation can be monitored with Kiwi anemometer. About 60% of the energy used annually at the University of Hawaii at Manoa is for building cooling systems. The UHM energy costs have tripled over roughly ten years. In 2014 UHM electricity bill is about $34M [28]. There are many projects in at the UHM using natural ventilation to reduce energy usage. The Kiwi anemometer could potentially collect the air flow data for analysis and future building designs.

3.1.1 Cup and Windmill Anemometers

The most common type of anemometer is the cup anemometer shown in figure 3.1. The modern version of the cup anemometer also has a rear tail to determine the wind direction. Cup anemometer works by measuring the current generated by a generator when the wind spins cups. Another popular type of anemometer and similar to the cup anemometer is the windmill anemometer shown in figure 3.2.
Figure 3.1: Cup anemometer

Figure 3.2: Windmill anemometer
3.1.2 Hot Wire Anemometer

Hot wire is an anemometer that has no mechanical moving part, and it works by measuring resistance changing of the hot wire due to the presence of wind cooling it. Figure 3.3 is an example of a hot wire anemometer configuration. The heating element does not necessarily have to be a metal wire; it can also be a resistor. Hot wire anemometer is commonly used more in environments where there are relatively small changes in temperature. Hot wire anemometer is not commonly used for outdoor application, because the accuracy of the sensor is dependent on the surrounding temperature.

![Figure 3.3: Hot-wire anemometer](image)

3.1.3 Laser Doppler Velocimetry

Light can be used to measure wind speed, and the light anemometer is called laser Doppler velocimetry (LDV) [29]. LDV is used in experiments measuring flow of fluid or gas that required noninvasive measuring. Figure 3.4 is an example of a LDV setup measuring air flow of an air ventilation duct. Some other applications that LDV can be used to measure the blood flow near-surface artery, and also the exhaust of a rocket engine [30].
3.1.4 Ultrasonic Anemometer

Ultrasonic anemometer uses ultra sound to determine wind speed. Ultra sound pulse is transmitted between a pair of transducers, and measures the arrival time of the pulse [31]. The arrival time of the ultrasonic pulse is smaller when the wind flow propagates in the same direction of the pulse, and the arrival time is larger if the wind flow propagates in the opposite direction. Ultrasonic anemometers can also determine wind direction in two dimensions when it has four transducers in the configuration shown in figure 3.5.
3.1.5 University of Colorado Boulder Passive Acoustic Anemometer

Unlike an ultrasonic anemometer transmitting ultrasonic pulses, passive acoustic anemometer measures wind speed without emitting a pulse. The University of Colorado Boulder (UCB) passive acoustic anemometer measures air flow using the same technique used in sonar to measure the flow of fluid in water. UCB passive acoustic anemometer uses the cross colorations of the receiving noises between omnidirectional microphones placed in a small area [32]. A typical UCB acoustic anemometer has four to five microphones shown in figure 3.6.
3.2 Kiwi Passive Acoustic Anemometer

3.2.1 Project Overviews
The passive acoustic anemometer developed by UCB requires a large area, and independent sources of sound to measure the wind speed. A different passive acoustic anemometer was developed by the Smart Campus Energy Lab (SCEL), at the UHM named Kiwi. The Kiwi anemometer is small, low power, low cost, and low complexity.

The Kiwi anemometer was developed by Andy Pham, and Daisy Green. The anemometer name Kiwi was chosen by Daisy Green, because its size is comparable to a kiwi fruit. The next sections of this chapter will discuss the experiments examining the signal generated from and captured by the microphone due to the presence of wind. Additionally, we examine and discuss the develop-
ment of the Kiwi anemometer algorithm to ensure that the wind speed measurements are invariant to changing distances from wind sources, the Kiwi anemometer noise sensitivity, the implementation of the algorithms in hardware, and the Kiwi anemometer applications.

3.2.1.1 Capture Wind Speed and Wind Generator

In most signal processing applications having data is necessary to design the system, because you need to know the properties of the signal. This section of the chapter will discuss the experiments of capturing microphone signals generated by a known wind source with variable speeds. The data will later be used to design the anemometer algorithm underlying the wind speed detection system.

3.2.1.2 Capture Wind Speed

There are two constituent parts of the experiment setup to capture the microphone signals generated by the wind: one is the microphone as the receiver and the other is the fan as the wind generator. The equipment and parts that were used in the receiver setup are shown in table 3.1. Arduino UNO microcontroller ATMEGA328p have 10 bit analog to digital converter (ADC), therefore the ranges of representation are from (0-1023). The resolution of the 10 bit ADC is 5V/1023 if using 5 V for the microphone VCC, and is 3.3V/1023 if using 3.3 V as the microphone VCC. The Kiwi anemometer used data ranging from 0 to 1023, instead of the voltage of the microphone signal. For example: The voltage signal of the microphone is 2.44V using 5V supply, then the analog data equivalent is 2.44V/(1023/5V) = 499 when rounded to the nearest integer.

In this experiment the microphone VCC was connected with 3.3 V. The reason for using 3.3 V instead of 5 V is because the weather sensors module in Chapter 2 was designed to run on 3.3 V power bus. The Kiwi anemometer can later be integrated with the weather sensors module for outdoor applications. Figure 3.7 depicts the setup of the receiver. Arduino UNO is powered by a laptop through the USB cable. The USB port is also used for communication between the laptop and the Arduino UNO. The microphone output pin is connected to the Arduino UNO analog input pin 0. Other analog input pins can also be used for the microphone output, if the analog input pin
0 is not available. The ground pin of the microphone is connected with the common ground pin on the Arduino UNO board. The oscilloscope channel 1 probe is connected with the digital pin 4 of the Arduino UNO, and the channel 1 reference is connected with the common ground. Digital pin 4 on the Arduino will output a pulse on the oscilloscope every sampling cycle.

<table>
<thead>
<tr>
<th>Part and equipment name</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laptop with Arduino IDE software</td>
<td>1</td>
</tr>
<tr>
<td>Arduino UNO</td>
<td>1</td>
</tr>
<tr>
<td>USB A/B cable</td>
<td>1</td>
</tr>
<tr>
<td>Omnidirectional microphone</td>
<td>1</td>
</tr>
<tr>
<td>Colored jumper wire</td>
<td>1</td>
</tr>
<tr>
<td>Oscilloscope</td>
<td>1</td>
</tr>
<tr>
<td>Windmill anemometer</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 3.1: Equipment and part required for collecting the raw data from the microphone

![Figure 3.7: Receiver configuration](image-url)
The microphone used in the experiment was an omnidirectional microphone that has a frequency range from 20Hz to 20kHz. The microphone came equipped with the breakout board when purchased from an Adafruit vendor. The microphone breakout board has three pins; Vcc (2.3 to 5.5)V, ground, and analog output, all shown in figure 3.8. Even through the microphone is omnidirectional, the microphone sensing pattern has a strong listening main lobe. The main sensing lobe of the microphone was used to measure wind speed. The output signal from the microphone is small, thus the breakout board had integrated Op-Amp MAX4466, and a potentiometer to adjust the output gain [33].

![Figure 3.8: Omnidirectional microphone](image)

Figure 3.8: Omnidirectional microphone

Figure 3.9 is the Arduino UNO code used in the receiver in Figure 29. The computer is communicating with the Arduino UNO through the universal asynchronous receiver/transmitter (UART) at 115,200 bit per second. The Arduino UNO is powered by 5V, but the microphone is powered by 3.3V, therefore an external 3.3V reference is needs to be connected to pin AREF of the Arduino UNO. The Arduino UNO pin 4 was used as an indicator sampling time. The pin 4 is pulled HIGH at the beginning of the sampling cycle at time T1, and pulled LOW when the sampling cycle is finished at time T2. Oscilloscope measures the sampling period T = (T2-T1), and the sampling frequency is 1/T Hz.
The window terminal of the Arduino IDE was used to display the data. The data is continuously collected for 60 seconds. After the 60 seconds, then the data is copied into an excel sheet shown in figure 3.10. In the excel sheet the first row of every column is the wind speed in miles per hour (MPH). Row one was obtained by using the a commercial anemometer TPI 556. Each column is a set of data for a particular wind speed. For example: column 1 has a wind speed of 3.3MPH, and the first data point of this data set is 480, the second data point is 483, and so on. This is the raw data generated by the microphone which is converted to wind speed which is explained in the first paragraph of Section 3.2.1.2.
3.2.1.3 Wind Generator

The following equipment in Table 3.2 was used as the variable wind speed source. The wind source used was a fan with four speeds. An AC light dimmer was used to fine tune the fan wind speed. The set up is shown in Figure 3.11.

<table>
<thead>
<tr>
<th>Part and equipment name</th>
<th>Quantity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 Speed Fan</td>
<td>1</td>
<td>Winds Generator</td>
</tr>
<tr>
<td>120V 500W light dimmer</td>
<td>1</td>
<td>Fine tune wind speeds</td>
</tr>
</tbody>
</table>

Table 3.2: Equipment and part required for collecting the raw data from the microphone
3.2.1.4 Capture Microphone Signal Experiment Setup

The diagram in figure 3.12 is the experimental setup to capture the microphone signal generated by wind. The wind generator was placed 18 inches away from the receiver microphone along with the commercial windmill anemometer TPI 556 [34]. The microphone and the windmill anemometer are 0.5 inches away from each other so that they don’t interfere with each other. Figure 3.13 is the physical setup of the diagram in figure 3.12. Figure 3.14 shows the fan and the light dimmer placed on one of the two tables. The other table was used to hold the receiver microphone shown in figure 3.13. The microphone was mounted onto the table such that the main sensing lobe of the microphone is parallel with the surface of the table.
3.2.1.5 View Microphone Raw Data

Once the data is collected the next step is to plot it. The following Matlab code in figure 3.15 was used to plots the raw data shown in figure 3.16. Its important to know that the excel file must have the exact data structure as shown in figure 3.10.
Figure 3.15: Matlab code used to plots raw data

```matlab
%% This Code plot the raw data of the microphone
c1c
close all
clear all
A = xlsread('WindData 0 to 15pt3 mph.xlsx');
Dimension = size(A);
Dimension(1) = sum(~isnan(A(:,1)));
X = A(~isnan(A));
clear A

X = reshape(X,Dimension);
Speed = [num2str(fliplr(X(1,:))) ones(Dimension(2),1)*(' mph'))];
X = X(2:end,:);

n = 1:size(X,1);

plot(n,fliplr(X))
title('Microphone analog data')
legend(Speed)
xlabel('sample')
ylabel('Analog Amplitude')
```

Figure 3.16: Microphone analog data with varying wind speeds with wind generator (fan) 18 inches from the microphone
3.2.1.6 Algorithm Development

From observing the plotted data in Figure 38 the amplitude of the signal got larger as the wind speed increased. For example: 9.1MPH has larger amplitude than 5.2MPH. The peak detector algorithm was used to extract the signal peak ($V_p$).

3.2.1.7 Peak Detector

The full bridge rectifier schematic shown in figure 3.17 is a peak detector technique commonly used to convert alternate current (AC) into direct current (DC). The block diagram shown in figure 3.18 is the equivalent to the full bridge rectifier. $x(t)$ is the input signal sinusoid, and $y(t)$ is the output measure across resistor R1. The four diodes in the full bridge rectifier circuit are equivalent to the absolute value operator. There is a voltage drop across the diodes, thus the 1.4V is subtracted from the $x(t)$. The capacitor and the resistor R1 are the equivalent of a first order low-pass filter (LPF). In figure 3.19, the red plot is the input sinusoid passing through the full bridge rectifier (figure 3.17), and the output voltage across the resistor R1 is the green plot. The input sinusoid has 5 V peak, thus we want to detect the 5 V at the output $y(t)$. The green plot in figure 3.19 is the peak detector output, and it shows that the peak is 3.6V instead of 5V. The peak detector did not output 5V, because the signal dropped 1.4V across the two diodes before passing through the LPF.

![Figure 3.17: Full bridge rectifier schematic](image)

Figure 3.17: Full bridge rectifier schematic
Using a full bridge rectifier circuit as a peak detector is difficult to implement. Unlike the input sinusoid, the signal of the microphone has a wider range of frequencies, and therefore it’s hard to determine what the LPF cutoff frequency is. The lump elements capacitor and resistor are sensitive to temperature, thus the properties of the LPF might changes from the design specification. The alternative method is to implement the peak detector via software. The properties of the LPF are easier to modify in software in order to reach more accurate specifications. The LPF in software also has less tolerance to temperature and noise.
3.2.2 Peak Detector in Software

Figure 3.20 is the modified peak detector diagram where the 1.4V dropped from the diodes is eliminated.

\[ x(t) \rightarrow |. | \rightarrow LPF \rightarrow y(t) \]

Figure 3.20: Peak detector diagram

There are two types of digital filters, infinite impulse response (IIR), and finite impulse response (FIR). FIR filters have the advantage of always being stable with bounded input and bounded output. FIR filters take more delay taps to be equivalent to IIR filters, thus resulting in high computation costs [35]. I decided to use IIR-type filters in the LPF design to reduce computational costs. The two IIR filters selected were first order LPF, second order Butterworth LPF. Second order Butterworth LPFs have better -3dB attenuation than the first order LPF, but the first order LPFs have the lower computation cost. The LPF that was chosen in the final design is discussed in a later section. The input signal in figure 3.16 is already positive, as the input comes from a circuit that outputs the absolute value of \( x(t) \) minus its mean value. The output \( y(t) \) is a weighted sum of present and past inputs, thus \( y(t) \) is equal to the mean of the \( x(t) \) at steady state instead of to \( V_p \). The mean of \( x(t) \) is required to be subtracted from the \( x(t) \) before it is passed through the absolute value operator. The modified version of the peak detector is shown in figure 3.21. The microphone signal has a raw data mean of 510.

\[ x(t) \rightarrow + \rightarrow |. | \rightarrow LPF \rightarrow y(t) \]

Figure 3.21: Kiwi modify peak detector
3.2.3 Peak Detector With First Order LPF

A larger set of data with wind speeds ranging from (0-15.3)MPH was used in implementing the peak detector using the first order LPF. Figure 3.22 is the plotted data used in the experiment. The sampling rate is obtained from the sampling period observed from the oscilloscope. The sampling time period is 1.33 milliseconds, therefore the sampling rate is 753Hz.

![Microphone analog data]

Figure 3.22: Raw analog data of wind speed ranges from (0-15.3)MPH

The implemented first order LPF system is

\[ y[n] = \alpha x[n] + (1 - \alpha)y[n - 1], \quad 0 \leq \alpha \leq 1 \]

The transfer function Z-transform is

\[ H(z) = \frac{\alpha}{1 - (1 - \alpha)z^{-1}}, \quad |z| < \frac{1}{1 - \alpha} \]
The impulse response is

\[ h[n] = \alpha(1 - \alpha)^n u[n] \]

The constant \( \alpha \) (alpha) was changed until the peak detector output \( y(t) \) appeared smooth. The Matlab code in figure 3.23 took the input data set \( x \), and the Mean (510) of the data, then output the peak of the data set as \( y \).

```matlab
function y = realtimeProcess(x, Mean)
    y = zeros(length(x)+1,1);
    alpha = 0.0002;
    for i = 1: length(x)
        y(i+1) = alpha*abs(x(i)-Mean)+(1-alpha)*y(i);
    end
    y = y(2:end);
    return
end
```

Figure 3.23: Implemented peak detector system of figure 3.21 in Matlab

The first alpha was chosen to be 0.5, and then decreased or increased as needed. Using alpha equal to 0.5 did not result in differentiable wind speeds, as shown in figure 3.24. The result of decreasing the alpha value to 0.01 is shown in figure 3.25. The output of the peak detector is much clearer with the smaller alpha value. When the alpha value is decreased to 0.001 the output of the peak detector is further smoothed, and the results are shown in figure 3.26. As expected, the result with smaller alpha is better in term of the smoothness of output \( y(t) \) and differentiable between the signals. The value of alpha was decreased further to 0.0001. The result is shown in figure 3.27. The output of the detector is much smoother for alpha equal to 0.0001. From observing the peak detector output smoothness shown in figure 3.27, using alpha equal to 0.0001 is sufficient.
There is a trade off for smoothening out the output $y(t)$. It took about 30,000 samples before reaching the steady state with alpha equal to 0.0001. With sampling rate of 753 Hz, it would take about 40 seconds to reach a steady state. Wind speeds vary at high frequency when the weather condition is unstable, thus using Arduino UNO to do Kiwi anemometer signal processing is not reliable. The wind generated from the fan is relatively constant, thus using Arduino UNO is acceptable for indoor testing.

![Figure 3.24: Peak detector output with alpha equal 0.5](image)

Figure 3.24: Peak detector output with alpha equal 0.5
Figure 3.25: Peak detector output with alpha equal 0.01
Figure 3.26: Peak detector output with alpha equal 0.001
3.2.4 Peak Detector With Second Order Butterworth LPF

Matlab signal processing tools box has the function Butterworth filter $[b, a] = butter(n, Wn, Ftype)$ [36]. The vectors $b$ and $a$ are the coefficients of the transfer function

$$H(z) = \sum_{k=0}^{n} \frac{b_k z^{-k}}{a_k z^{-k}}$$

where $n$ is the filter order. $Wn$ is the normalized cutoff frequency. $Ftype$ is the type of filter we want to implement. The system equivalent using recursion method is

$$y[\hat{n}] = \sum_{k=0}^{n} b_k x[\hat{n} - k] - \sum_{i=1}^{n} a_k y[\hat{n} - i]$$
The $W_n$ is varied until the output of the peak detector is smooth.

Figure 3.28: Output of peak detector with $W_n$ equal $0.0001\pi$ rad/sample

Figure 3.28 is the output of the peak detector using $W_n$ equal to 0.0001. The results of using Butterworth LPF are better than the first order LPF. Butterworth LPF took about one third less samples to reach a steady state than first order LPF. The vectors $\mathbf{b}$ and $\mathbf{a}$ of the Butterworth transfer function are

$$
\mathbf{b} = \begin{bmatrix} 0.2467 & 0.4934 & 0.2467 \end{bmatrix} \times 10^{-7},
$$

$$
\mathbf{a} = \begin{bmatrix} 1.0000 & 1.9996 & 0.9996 \end{bmatrix}
$$

Implementing the system of second order Butterworth LPF has some drawbacks. The microcontroller has to calculate many very small float numbers, thus the computational cost is high. In Matlab the time taken to process the data with second order Butterworth LPF is 107 milliseconds,
and first order LPF is 23 milliseconds. The first order LPF is better than second order Butterworth LPF in term of running time. For this reason, the first order LPF was chosen over the second order Butterworth LPF.

3.2.5 Correlate Peak Detector Output With Wind Speeds

In the previous section we observed that the amplitudes of the output of the peak detector at steady state for different wind speeds. This section will discuss the method of correlating the peak values with the wind speeds. The output of the peak detector is varied, and therefore the output was averaged from the steady state time. Figure 3.29 is the relationship curve between the wind speeds and the average of the peak detector output at steady state. The curve is not linear, and therefore using linear regression is not appropriate. Other methods that could be used are piecewise linear regression, or linear regression after transforming the curve into linear. The reason why the curve in figure 3.29 is not linear because the relationship between wind speeds and power produced by turbines is roughly cubic [37]. Figure 3.30 is the transformed curve of figure 3.29 by taking the square roots of the output of the peak detector. Linear regression was used to estimate the transformed curve shown in figure 3.31.
Figure 3.29: Wind speeds verse peak detector output
Figure 3.30: Wind speeds verse square root peak detector output
3.2.6 Wind Speed Invariant With Changing Distance Experiment

This section will discuss the experiment results that verified the wind speed is invariant with changing distance between the wind generator and the Kiwi anemometer. Figure 3.32 is the data set with a constant wind speed of 3.3 MPH, and the varied distances between the wind source and the anemometers: 18, 22, 26, 30, 34, and 38 inches. The TPI 556 windmill anemometer has wind speed measurement accuracy of 1%, therefore the wind speed reference is not exactly 3.3MPH. The testing results in table 3.3 showed that when we varied the distance between the wind source and the anemometers does not affect the Kiwi anemometer measurement significantly.
3.2.7 Noise Sensitivity Experiment

This section discusses the effects of additive noise had on the Kiwi anemometer. Traffic noise and rain noise were used in the experiment. A speaker was used to generate the additive noise. There were two parts to the experiment: 1) Varied rain noise intensity from low to high with constant
wind speed, 2) varied wind speed from low to high with constant car noise intensity. The Kiwi anemometer measurement of 0MPH wind speed with added noise was defined as noisy wind speed faulty measurement (NWSFM). The windmill anemometer was also used in the experiment as a reference. Table 3.4 showed the results of part one of the experiment. There was little effect on the Kiwi anemometer measurement when the NWSFM was lower than the noiseless wind speed. Table 3.5 is the results of part two of the experiment. The results showed that there was little effect on the Kiwi anemometer measurement for NWSFM 3.7MPH. A second NWSFM 6.1MPH was used to verify that the Kiwi anemometer measurement is reliable when NWSFM is lower than the wind speed. Videos of traffic noise [38] and rain noise [39] were recorded during experiments to demonstrate Kiwi anemometer noise sensitivity.

<table>
<thead>
<tr>
<th>NWSFM (MPH)</th>
<th>Kiwi Measurement (MPH)</th>
<th>TPI 556 Measurement (MPH)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.8</td>
<td>5.4</td>
<td>6.0</td>
</tr>
<tr>
<td>5.4</td>
<td>5.6</td>
<td>6.0</td>
</tr>
<tr>
<td>5.8</td>
<td>6.5</td>
<td>6.0</td>
</tr>
</tbody>
</table>

Table 3.4: Kiwi anemometer sensitivity to additive traffic noise

<table>
<thead>
<tr>
<th>NWSFM (MPH)</th>
<th>Kiwi Measurement (MPH)</th>
<th>TPI 556 Measurement (MPH)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.7</td>
<td>6.5</td>
<td>7.0</td>
</tr>
<tr>
<td>3.7</td>
<td>7.6</td>
<td>8.2</td>
</tr>
<tr>
<td>3.7</td>
<td>8.5</td>
<td>9.1</td>
</tr>
<tr>
<td>6.1</td>
<td>7.2</td>
<td>7.0</td>
</tr>
<tr>
<td>6.1</td>
<td>7.8</td>
<td>8.2</td>
</tr>
<tr>
<td>6.1</td>
<td>8.7</td>
<td>9.1</td>
</tr>
</tbody>
</table>

Table 3.5: Kiwi anemometer sensitivity to additive rain noise
3.2.8 Implement Algorithm on Arduino UNO

The code in figure 3.33 was programmed into the Arduino UNO used to measure wind speed. The code continuously processes the wind data, and displays the latest wind speed measurement every second on the terminal screen.
```c
int average = 0;
float output = 0;
float previous_output = 0;
int freq = 4000;

void setup() {
  int baud_rate = 9600;
  Serial.begin(baud_rate);
  Serial.println("Begin program");
  // analogReference(EXTERNAL);

  // Set frequency
  long period = ((float)1.0/(float)freq)*1E6;
  Serial.print("Period: ");
  Serial.println(period);
  long timer_delay_us = period;

  // Attach timer
  Timer1.initialize(timer_delay_us);
  Timer1.attachInterrupt(interrupt_loop);
}

void loop() {
  
  /* Here is the routine that is run when the timer interrupt goes off */
  void interrupt_loop(){
    int sensorValue = analogRead(A0);

    // Print out wind speed every second
    if(counter >= freq){
      previous_output = output;
      // Estimate wind speed
      output = sqrt(output)*1.283 + 0.0619;
      Serial.println(output);
      counter = 0;
    }
    // Sample, and process the data continuously
    else{
      // y[n] = alpha*(x[n]-MEAN)+(1-alpha)*x[n-1]
      output = 0.0001*abs(sensorValue - 510) + 0.9999*(previous_output);
      previous_output = output;
      counter++;
    }
  }
}
```

Figure 3.33: Arduino code that measures wind speed in real time
3.2.9 Kiwi Anemometer Demo

A live demo was video recorded of the Kiwi anemometer algorithm on Arduino UNO measurements with known wind speeds [40]. In the video, the wind speed measured by the Kiwi anemometer is really close to the commercial windmill anemometer. The Kiwi anemometer proved that using a microphone to measure wind speed is possible, and reliable for wind speeds greater than 2 MPH. The Kiwi anemometer has a faster response to the wind velocity changes than the commercial TPI 556 anemometer.
4 SUMMARY AND FUTURE WORKS

Apple module is still collecting weather data from January, 2015. The Cranberry module will be tested outdoors once the housing is complete. Placing the Cranberry module on rooftops will require working closely with UHM facilities concerning where we can securely place the modules and ensure safety conditions are met. We will also obtain a professional module to calibrate with the Cranberry module and look to efficiently process and analyze the data that we receive from each of the modules.

Kiwi anemometer cannot differentiate wind speeds below 2MPH as shown in figure 3.31. One reason could be that the microphone signal with wind speeds below 2MPH is too small and close to each other. One possible method to measure low wind speed is to integrate an additional amplifier with a large gain. This large gain amplifier will spread out the microphone signal before passing it through the peak detector. This amplifier will only be used when the Kiwi anemometer detects wind speed lower than 2MP. Figure 4.1 is the system block diagram of Kiwi anemometer with addition amplifier. The amplifier with gain G1 is used to measure wind speeds 2MPH or higher, and the amplifier with larger gain G2 is used when wind speeds are below 2MPH.

Figure 4.1: Kiwi anemometer with addition amplifier
In section 3.2.3 we mentioned that processing power of the Arduino UNO is not applicable for the Kiwi anemometer. Another developer board named Teensy [41] with faster microcontroller is replacing the Arduino UNO. Teensy has Atmel ATMEGA32U4 ARM microprocessor, and it can sample over 1MHz. Teensy is also fast at multiplying and dividing float numbers. The new processor will give Kiwi anemometer the ability to measure the fast rate of changing wind speed.

Other future work of Kiwi anemometer is to make it capable of measuring 2-dimensions wind direction. One possible way to measure wind direction is to use four microphones with the configuration shown in figure 4.2. We can use arctangent to calculate the wind angle $\theta$, and the equation is

$$\theta = arctan\left(\frac{Y}{X}\right), \quad Y = |Mic_B - Mic_D|, \quad X = |Mic_A - Mic_C|$$

Then to find what quadrant of the wind direction we can use if statements: if $Y, X > 0$ then quadrant 1; if $Y > 0, X < 0$ then quadrant 2; if $Y, X < 0$ then quadrant 3; and if $X > 0, Y < 0$ then quadrant 4.

![Figure 4.2: Measures wind direction](image)
We are also interested in conducting experiments to vary the angle from the fan to our wind sensor. This is to better understand how sound propagates at different angles and with multiple sensors we can eventually not only measure wind speed, but also obtain wind direction.

Other steps for improvement are modifying the hardware. This includes using an RMS to DC converter to replace the full bridge rectifier.
REFERENCES


2014.


