Sensing the Scientists

A Wireless Sensor Network in BRKI 167

Tiago Bozzetti, Andrew D'abbraccio, Joe Duvall, Ethan Vynalek, Alex Wade, CSCI 379
Bucknell University
Lewisburg, PA, U.S.A.
amm042@bucknell.edu

***Abstract*—This paper describes an application of wireless sensor nodes to a computer science student laboratory located in room 167 of the Breakiron Engineering building on the Bucknell University campus. We were tasked to use sensors to capture data from the sensing of basic environment characteristics such as temperature and humidity. We then recorded sensor data over a period of 24 hours. This paper discusses difficulties encountered in sensing the data, the meaning of our data, and some of the outliers we observed.**

Index Terms—Sensor, node, wireless, efficiency, development, practicality, design.

I. Introduction

FOR centuries, humans have interacted with the world around us using only our five senses. Advances in technology, have paved the way for a new frontier of human perception – wireless sensors.

Wireless sensors are available in countless types and can be combined into just as many configurations, so the set of sensors we employed in our node represents but one such configuration. Despite the simplicity of our product, experiencing its development cycle proved to be an insightful and rewarding experience.

In this paper, we examine the premise, methods, and results of our design in an educational context, and address the impact of issues we experienced, considering their applications to within the context of complex system development.

II. Design

Our node was designed with simplicity and ease of implementation in mind. As such, it is bulky, and lacks the portability and durability required for use outside of a laboratory setting.

A. Node Hardware

We built our node around a SparkFun Fio v3 Arduino board, aboard an R.S.R. Electronics MB-106 Breadboard. To communicate with our base station, the node incorporated an Xbee RF radio module. The node carried two sensors: a Honeywell HIH6030 Temperature/Humidity Sensor and a Vishay BC Components NTC 10K 400mm Pipe Thermistor.

B. Base Station

Our base station centered around the Raspberry Pi platform. We powered the unit over PoE, and used it to receive communications from our node with an Xbee over USB interface adapter.

On the base station we installed a PostgreSQL database engine to store and organize our data. We wrote a Python script that we ran continuously, capturing data as it came in from the sensor node and inserting it into the database accordingly. We were able to monitor sensor data in real time through a web server running on the Raspberry Pi, accessible over the local network.

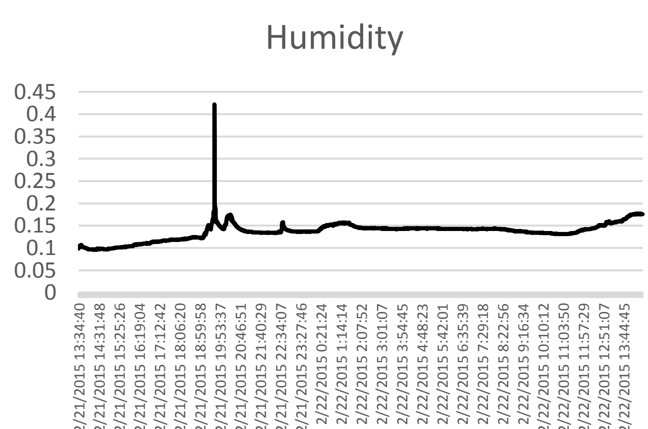

Fig. 1: Honeywell HIH6030, humidity (measured from 0.0 to 1.0)

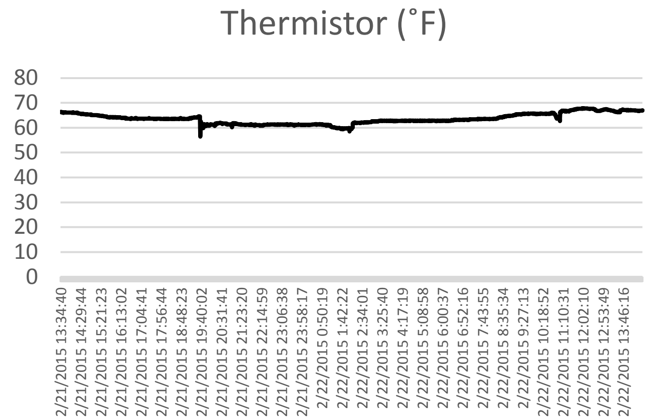

Fig. 2: Thermistor, temperature (in degrees Fahrenheit)

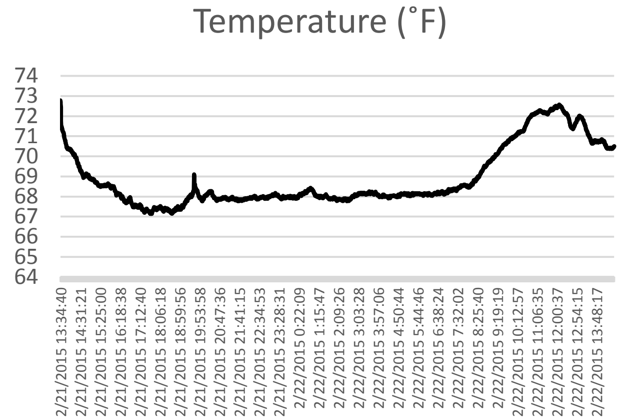

Fig. 3: Honeywell HIH6030, temperature (in degrees Fahrenheit)

III. Analysis

We sampled three data points reported from two sensors units on a single wireless node over a period of 24 hours. Our measurements were taken roughly every 24 seconds, resulting in 3,768 discrete data samples for each of our three data points, or 11,304 total data samples.

Our data was mostly consistent. Our only noteworthy deviation occurred around 20:00 Saturday, when the humidity sensor registered uncharacteristically high readings. We confidently attributed this to human interaction, in which an individual engaged in heavy exhalation directly onto the humidity sensor. When the human interference was removed, the humidity sensor resumed normal operation.

Initial problems in the experiment were corrected quickly. We learned how to use the different sensors individually, each member becoming familiar with their sensor, and then combining our code into one text.

The R.S.R. Electronics MB-106 Breadboard on which we built our sensor node was well-suited to our needs in that it allowed for easy connection and testing of the parts that compose the node, but is excessively bulky. Designed, as it was, for testing, the board's ease of disassembly was beneficial in development but would be a notable liability in any long-term deployment in which a wire might be pulled loose, rendering the node nonfunctional.

The 3.7V lithium-ion battery attached to our system was able to provide power throughout the entire window of data collection, even at our relatively-high sampling rates. We were not able to measure battery voltage nor current draw and thus cannot provide any meaningful conclusion as to the performance of this battery in a prolonged deployment. We are similarly unable to predict how it would behave in more extreme conditions, as our testing was localized to a single stable laboratory.

We had some difficulty with the implementation of the temperature and humidity sensor, which uses the I2C communication protocol – one with which we lacked familiarity. Fortunately, we found a library that was made specifically for the sensor, which simplified our work. Our challenge then became understanding the library, but this proved much simpler than learning the entire I2C protocol.

After writing the code, we began construction of the circuit. This proved to be a bit more complicated than our previous experience building sensor nodes, since we now had to use capacitors in addition to resistors.

An Xbee radio (using the 802.15 protocol) was employed as our medium of communication from the node to the base station. A public Arduino library was used to assist with our development of transmission and response requests. This library required specific serial commands to modify radio specific settings for use. While it carried with it a learning curve of its own, our development time was greatly reduced.

IV. Conclusions

In our implementation of a wireless sensor node, we were successfully able to capture environmental data. Throughout the development process we encountered some obstacles, but were able to overcome them with success. We have concluded through analysis that our data was both accurate and appropriate. Overall, our implementation satisfied the parameters of our initial goal.

There are still many areas for improvement: better sensor hardware, optimized sensor placement, a more robust radio communication protocol, and more in-depth base station analysis. These potential improvements represent how the field of wireless sensor networks is continually evolving. There is always something to improve – whether through enhanced reliability, expanded applications, or more thorough analysis.

We worked with many facets of the wireless sensor networking field as a part of this project. In the future, we hope to apply our experience to a marketable wireless sensor device.

The code for this project is available to authorized users at the following URL: https://gitlab.bucknell.edu/adw011/wsn-project-1.git

Acknowledgments

The authors would like to thank Prof. Alan Marchiori for his compilation of Arduino libraries and Python scripts used extensively as references and examples for inspiration.