EECE 404 Senior Design II

Wireless Temperature Sensor Network (SensorNet)

<u>Final Report</u>

Submitted by:

Sean Grant

Kolby Lacy

Faculty Advisor:

Dr. Hassan Salmani

Date:

April 24, 2018

Department of Electrical and Computer Engineering
Howard University
Washington, DC 20059

I. Summary:

In certain buildings, the HVAC systems are implemented in such a way that they ‘work’ but are not optimized in a way that would allow for more efficient usage or energy consumption. Mostly because they often times rely on the input from a single source, which implements the same settings for various rooms across the building. What we want is to take an existing system such as this and be able to automate individual temperature readings for each room. With the creation of a sensor network, we can achieve this by having sensors in a room automated to record and report temperature readings to a central location that will process the data and take steps to adjust the temperature. Wireless temperature sensor networks are on the cutting edge of communication technology and many useful applications have been discovered in the military, health, and environmental science fields. What is useful about wireless senor networks is that hey can record much more information than a single source and can obtain valuable information that can be provided by a variety of sensors. Even more, this process can be automated, increasing its efficiency. Another convenience is that because this data is usually sent to a server, it can also be accessed remotely and interpreted by an expert if need be. There are numerous patents and research papers related to the implementation and security of these wireless sensor networks. Our project will implement a design for a sensor network exclusively for increasing HVAC operation.

II. Problem Statement:

Currently, the temperature management system within the Lewis K. Downing building through the use of a commercial HVAC is inaccurate and inefficient, raising a need for a customized, hybrid HVAC system which can sense and adjust the temperature in each room separately in real time. The project’s long term goal is to develop a design schematic for a wireless network system that records signals from temperature sensors, analyzes said signals, and then transmits output signals to control the HVAC system in place. On the contrary, the projects’ 2017-2018 academic school year goal is to implement a proof of concept network system in a demonstration in order to shoe the usefulness of the system and the management of electricity usage for heating and cooling. Also to improve the convenience of temperature management.

III. Design Requirements:

Our design is needed in order for a temperature sensors network composed of **2** sensors per room to be established in the engineering building. The design must allow the temperature of the room to be read by the nodes to within plus or minus **1** degrees. It must also offer temperature management capabilities. After the basic functions are established, secure connectivity is the next priority. We want our product to be as secure as it is effective. However, for this semester, the noly time constraint is to get a working temperature sensor network by the end of the semester. The design must also allow periodical alerts when detecting data anomalies.

We must also address the other constraints of our design. Our design must not interfere with wireless communications in the building. This is considered a **technological** constraint. Noting that we will only be using two nodes and one laptop connected to a router, there will not be enough traffic to cause any interference to users of the Howard wireless network.

In terms of power or energy consumption, our project has very low consumption. Our RPis use separate 9V power supplies and the laptop uses a standard 19.5V output charger. The other microcontroller is powered by the laptop. Our power supplies comply with the standard of a Class A external power supply as the supply less than 250 Watts of power.

This project is limited to just the inside of the Howard engineering building and is very small-scale. As such, the **social, cultural, and political** constraints are minimal. The only **social** constraints can be applied to a malfunctioning system that (in the future) allows the building to get to hot so that it is uncomfortable for the people in the building. It is imperative that the system functions correctly. In the same way there are no **environmental** constraints as long as all environmental regulations set forth by the FCC are followed.

One of the **economic** constraints set for by the project includes the unit price that will need to be paid for each node ($35 for each Raspberry Pi 3, $10 per temperature node, $10 per breadboard connector). Also the price of setting up the external database will need to be factored in but this depends on the size of database that is used.

Regarding the **time** constraints, the goal is for each temperature reading to be taken within 20-30 milliseconds of each other. This will provide a more accurate analysis of the temperature in the room and what actions will need to be taken by the corresponding HVAC system.

IV. Current Status of Art:

The wireless temperature sensor network project is centered around the current status of the data communications field. At the base of the project is the ability to transmit and receive signals at specific radio frequencies. The transmission of data wirelessly is related to the way in which the analog measurements taken from the temperature sensors will be transferred to the backend database that will monitor the system. The receiving of said signals will be the main task allocated to the backend server because once the signals are received it will be up to the backend CPU to process said data. However, having all of the processing power tied to the backend server is just one implementation idea that is being analyzed. Another idea that is being looked into is having a portion of the processing of the analog signals taken from the temperature sensors done on the actual sensor nodes themselves. This would be done with the purpose of reducing the bandwidth needed to transmit the data to the backend server. If the unprocessed data would consume less bandwidth than the output (processed) data then it would make sense to just transmit the unprocessed data to the backend CPU. However, if completing some level of processing on the

CPU of the temperature sensor itself and then transmitting the data saves bandwidth, then this is the route that our project will take.

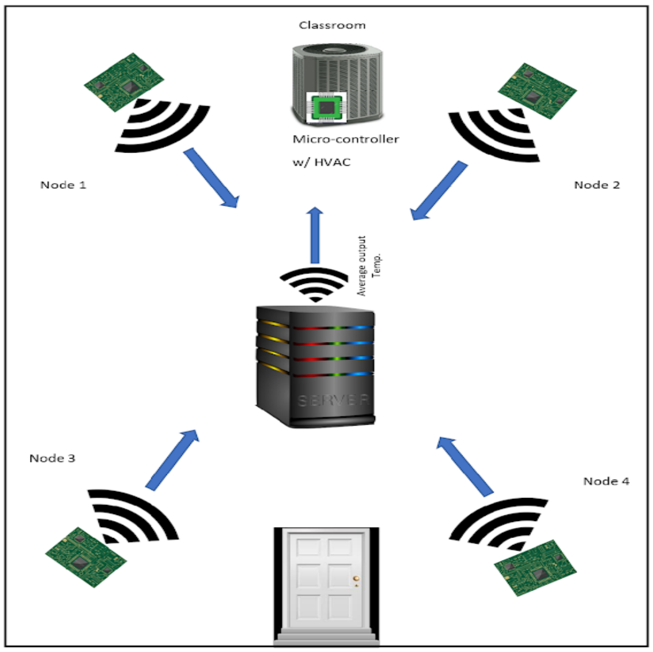
Currently, wireless data communications are done using radio frequencies. Any discussion of wireless communications and the use and security of wireless data transfer begins with analysis of the electromagnetic radio frequency spectrum. Wireless communications between devices can be thought of as a transfer of data through the air from one device to another, however, there are more elements at work throughout this process. Electromagnetic radio waves at specific radio frequencies, or channels, on the RF spectrum are used to transmit signals through the air where an electromagnetic signal receiver then receives the signal. Conceptually, the data being transferred is “attached” to an electromagnetic radio wave, as previously described, and then instead of sending the data to the receiver, the radio signal carrying the data is sent. Once the signal reaches the correct receiver (which is determined through identification addressing of the transmitter and receiver) the data is then extracted from the radio wave and analyzed accordingly. The electromagnetic radio frequency spectrum ranges from about 3kHz to 300GHz and each frequency (or channel) holds the capacity to transmit an individual signal. However, not every frequency, channel, or band (a range of channels) is available for use by the public. The Federal Communications Commission (FCC) assigns certain radio frequency bands to licensed entities (primary users) such as cellular phone carriers and government agencies. Any users who are not licensed to operate on a specific frequency band or have not paid to operate on the frequency band are considered secondary users. Secondary users can be any individual with a need to utilize a specific radio frequency channel or band, from wanting to set up their own WiFi network to wanting to establish their own data transfer protocol. The driving factor that separates primary users from secondary users is that primary users usually occupy a certain frequency band for commercial purposes, while secondary users may occupy a specific frequency band for more personal uses.

V. Solution Design:

Concept 1:

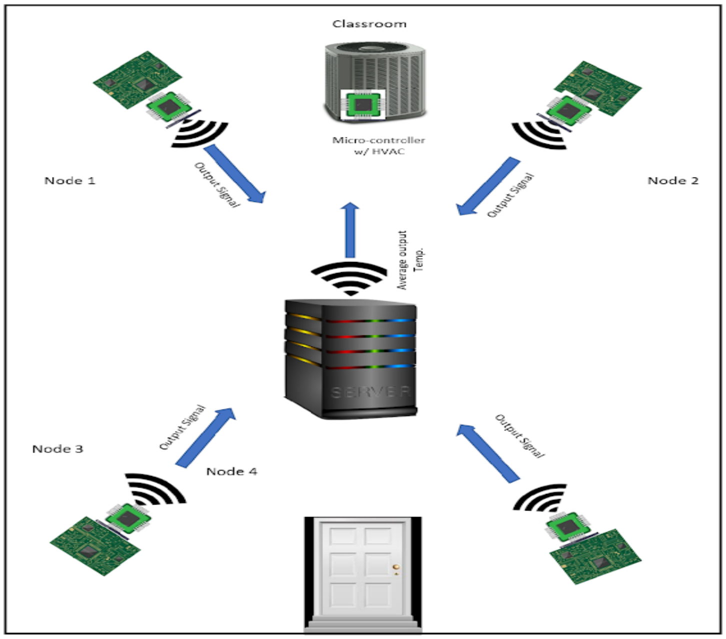
Having an external server that carries out all of the computations for the signals that will be collected from the temperature sensors. Once the data is sent from the sensor nodes to the database, all signal processing will happen on the database side, which will control the heating or cooling unit in the room. This will use less power from the nodes because the nodes are not performing computations. Also, several nodes will diagnose an area, which increases sustainability and will help detect any malfunctions or compromises.

Node 1 represents the temperature sensor node that would be placed in the far left corner of the room. **Node 2** represents the temperature sensor node that would be placed in the far right corner of the room. **Node 3** represents the temperature sensor node that would be placed in the adjacent left corner of the room. **Node 4** represents the temperature sensor node that would be placed in the adjacent right corner of the room. **Server** represents external server that will receive and record all of the measurements from each node as well as prioritizing the measurements based on when they are received by the server. This component will also transmit a signal to the HVAC system controlling its operation and, in turn, controlling the temperature in the room. **Microcontroller** represents the Arduino microcontroller that is used to receive the output signal sent from the server (database) and compare that value with the desired temperature in the room in order to control the HVAC's operation.



Concept 2:

Having separate processors on the sensor nodes so that once the input data from the temperature sensors (the measured temperature) is gathered from the sensor, all of the signal processing takes place on that node and then the results are passed to the database and microcontroller that will control the heating or cooling. This would allow for faster transmission because less information will be sent to the database.



Concept 3 (Developed by Sean):

Attaching sensors to the heating and AC HVAC units themselves to minimize the distance that the signals would have to travel in order to be processed and outputted in order to control the temperature in the room. Attaching each sensor node to the HVAC unit provides minimal latency from receiving temperature measurements. It would also give faster readings. In addition, this concept could be used to diagnose mis readings on the HVAC's original system.

Concept 4 (Developed by Kolby):

Using an FPGA board to control the input signals from the sensors and the output signals that would control the heater and AC unit in the room. The board would act as the middleman between the sensor and the HVAC. Each unit would be individually controlled by a board. Also, this solution would provide easier troubleshooting capabilities with minimal latency.

Selection Process:

The categories used in the decision matrix included: Cost, Functionality, efficiency, Reliability, and Practicality.

Concept 1**:**

Pros:

- Not having processors on each node cuts down on cost
- Having an external server provides storage for recording measurements
- Easy to troubleshoot

Cons:

- May cause a slight increase in latency
- Increases the strain on the server's processing power

Concept 2:

Pros:

- May slightly decrease latency
- Reduces strain on server's processing power

	Decision Matrix	
	5 (Most Feasible)--------------------------------1(Least Feasible)	
	Concept 1	**Concept 2**
Cost	4	3
Functionality	5	4
Efficiency	3	4
Reliability	4	3
Practicality	5	3
Total:	**21**	17

After weighing through the pros and cons of each solution concept in each category, the team decided to narrow down the selection to **Concept 1.**

VI. Project Implementation Plan:

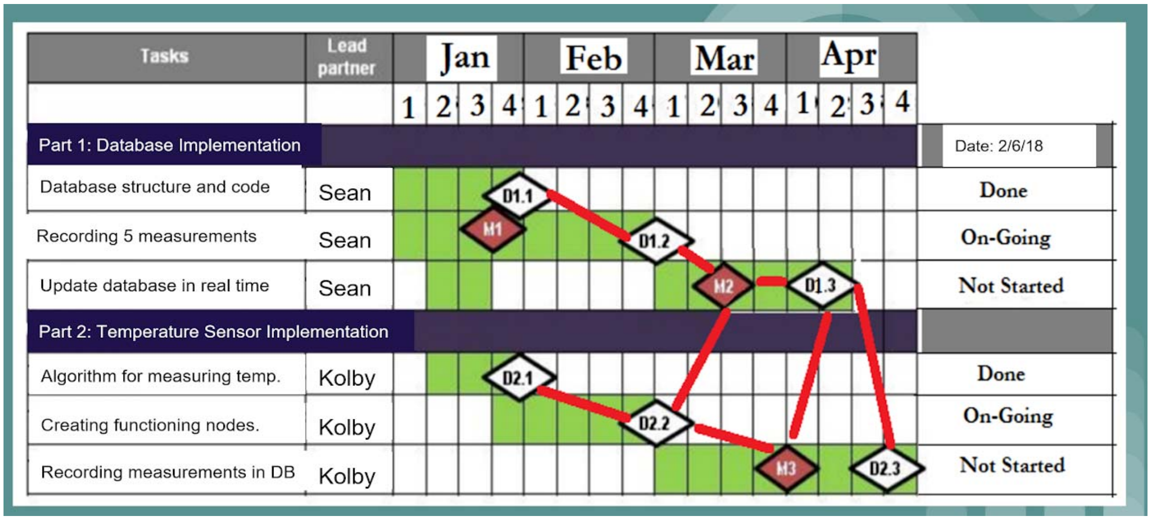
For this project, we need the following components: 2 Raspberry Pi 3s, 2 Waveshare 10 DOF IMU Temperature Sensors, 2 breadboards, 2 GPIO breakout kits for RPi, 1 wireless router, 1 Arduino Uno board.

In the month of January, we will gather all documentation on the microcomputers that we will use for nodes (originally Intel Joule boards but switched to Raspberry Pi 3s). We will also look up and compile the code to connect the temperature sensors to the RPis and take temperature readings.

For the month of February, we will determine a method of communication suitable for the connection between the nodes and the server. Once this is established, we will research how to implement this between one node and the server.

In March, we want to modify the server code so that it can accept multiple connections from the nodes. Then, we will attach multiple nodes and test simultaneous data collection.

In April, we will test the limits of the temperature sensors. We will also switch from our local area network over to the howard network and prepare for the presentation.



VII. Project Implementation Process:

The decision to use the Raspberry Pi 3 as our main node was one of availability. The intended Joule Boards did not come with the correct power supplies, so we had the Rpi as our alternative.

The setup of our node is shown below. The temperature sensor is directly connected to the Raspberry Pi while the RPi is connected to the server via TCP.

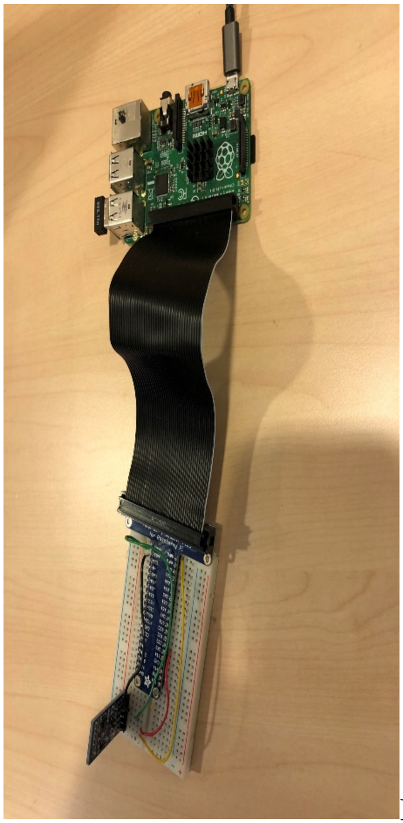
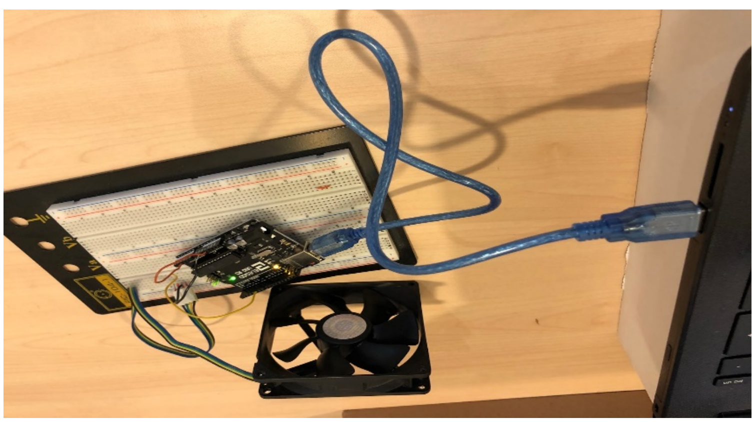


FIG. Node Setup

The setup for our microcontroller to “cool the room” is also shown below. This consists of the Arduino Uno and a 12V fan.



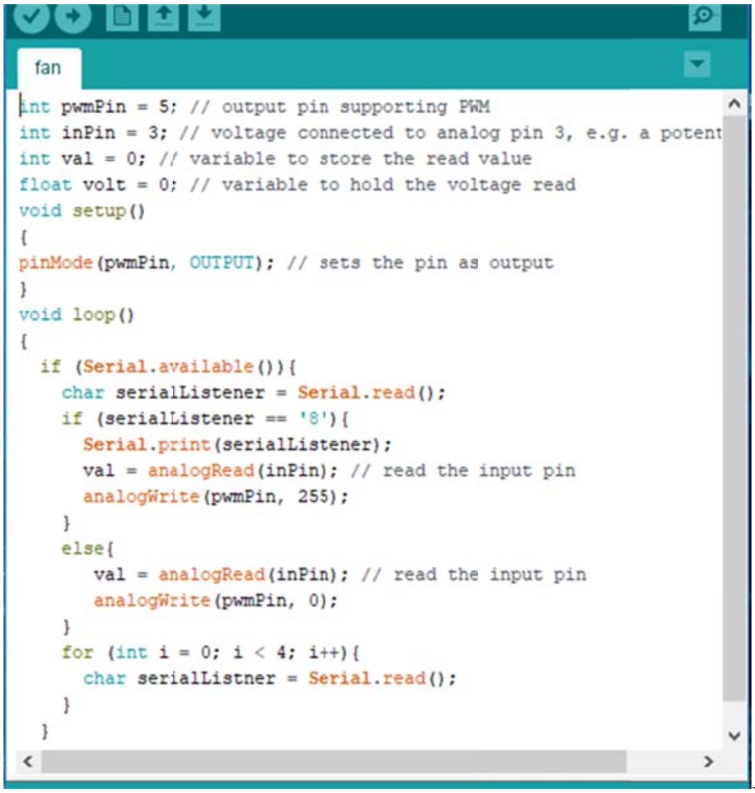
To read the measurements, create the tcp connection, and activate the fan. we used some python scripts and C code for Arduino. Some of the code snippets are shown below:

```
#!/usr/bin/env python

from Adafruit_BME280 import *
import socket
import time
import openpyxl

sensor = BME280(t_mode=BME280_OSAMPLE_8, p_mode=BME280_OSAMPLE_8, h_mode=BME280_OSAMPLE_8)

degrees = sensor.read_temperature()
degrees = degrees * 9 / 5 + 32

TCP_IP = '192.168.0.128'
TCP_PORT = 1234
BUFFER_SIZE = 1024


s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect((TCP_IP, TCP_PORT))
while 1:
    degrees = sensor.read_temperature()
    degrees = degrees * 9 / 5 + 32
    s.send(str(degrees))
    time.sleep(2)
data = s.recv(BUFFER_SIZE)
s.close()

print "Temperature:", data
print "Session closed"
```

Node script

```
#!/usr/bin/env python
import socket
import time


TCP_IP = '192.168.1.3'
TCP_PORT = 1234
BUFFER_SIZE = 1024  # Normally 1024, but we want fast response

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.bind((TCP_IP, TCP_PORT))
s.listen(2)
print "Waiting for connection..."

conn, addr = s.accept()
print 'Connection address:', addr
while 1:
    data = conn.recv(BUFFER_SIZE)
    if not data: break
    if (float(data) >= 80):
      print "Turning on fan"
    print "Temperture:", data[0:5], " F"

conn.send("Stream done")  # echo
conn.close()
```

Server script



```
int pwmPin = 5; // output pin supporting PWM
int inPin = 3; // voltage connected to analog pin 3, e.g. a potent
int val = 0; // variable to store the read value
float volt = 0; // variable to hold the voltage read
void setup()
{
pinMode(pwmPin, OUTPUT); // sets the pin as output
}
void loop()
{
  if (Serial.available()){
    char serialListener = Serial.read();
    if (serialListener == '8'){
      Serial.print(serialListener);
      val = analogRead(inPin); // read the input pin
      analogWrite(pwmPin, 255);
    }
    else{
       val = analogRead(inPin); // read the input pin
       analogWrite(pwmPin, 0);
    }
    for (int i = 0; i < 4; i++){
      char serialListner = Serial.read();
    }
  }
```

Arduino Code

VIII. Conclusion:

This year we were able to successfully design and develop a small-scale temperature sensor network. Though we were not able to get far enough to implement security measures, we were able to simulate a functioning temperature sensor network. Our system works and responds in real time to any changes. In the future, this project will be used to integrate with the HVAC in the engineering building and will continue as a hardware and network security project.