WIRELESS SENSOR ARRAY FOR FOREST FIRE DETECTION

By

Kenneth Adriatico

Steve Mesi

Travis Sauers

Final Report for ECE 445, Senior Design, Fall 2018

TA: Anthony Caton

15 December 2018

Project No. 2

Abstract

Current forest fire detection methods are unable to provide real-time fire detection and rely heavily on optical detection methods. Our project aims to provide real-time forest fire detection using a mesh network of distributed sensors. We tested the functionality of our design by creating three sensor units and measuring their response to the presence of smoke and fire. Our test results showed that the network was able to respond to the smoke and fire and quickly alert a user with a message containing the type of emergency and its location. Although our system functions as designed, alternative power supply options need to be considered to address issues found with the lithium ion batteries used in the design. Our project does in fact provide real-time forest fire detection and when distributed in a forest has the potential to reduce the damages due to forest fires by quickly alerting emergency personnel.

Contents

1. Introduction

Every day we can turn on the news and hear about a forest fire that is out of control, people's houses at risk of being destroyed, their lives in danger, or firefighters risking their lives. Forest fires have become a huge issue because of their destructive power and their effect on climate. These fires can cause many residual problems to human health as well through their effects on the environment [2]. They contribute to air and water pollution which has harmful effects on both wildlife and humans. Climate change also effects the frequency and severity of these fires because a warmer climate leads to dryer conditions which makes fires more prevalent. The fires release greenhouse gasses into the atmosphere which further contribute to climate change [1].

Our objective is to develop a system that can detect forest fires quickly and notify a user when an incident occurs. The first component of our system is the actual detection mechanism. This will consist of two sensors: a smoke sensor and a flame sensor. We are using two different sensors to account for the different drawbacks in each sensor. The smoke sensor may be more prone to false alarms due to ambient smoke concentrations in the air and the flame sensor will only be able to detect fires within a 60-degree angle in front of the unit [6]. Using these two sensors in conjunction will allow us to have accurate detection while avoiding false alarms. We will also incorporate a button on the sensor unit which will be used to send manual alerts when pressed. The purpose of the manual alerts is to help people who are lost in the wilderness or in any sort of danger. Because this is a remote application, it is important that the units require very little maintenance. To accomplish this, each sensor unit will be equipped with two 2.5-watt solar panels which will recharge four 18650 battery cells and allow the system to function indefinitely. We will be implementing a mesh network of sensors using the ZigBee communication protocol. The purpose of the mesh network is to eliminate the possibility of single node system failure. Without a mesh network, the entire system could be rendered useless if one of the sensor units is destroyed. With a mesh network, if one sensor unit is destroyed then the signal can be sent to any other node in the network without any issues. Forest fires can travel at up to 6 mph in forests which means that the sensing unit must be able to detect the fire and send an alert within 3.73 seconds assuming a detecting distance of 10 meters [14]. We constructed birdhouses that contain all the electronic components. This will protect the components from the environment, but it will also help reduce the impact that our design has on the wildlife by being more ecofriendly.

2 Design

Our design consists of four main blocks: The Power Circuit, the Communication Circuit, the Microcontroller, and the Sensor Block shown below in Figure 1. The Power Circuit powers the system and regulates voltages; the Communication Circuit forms the mesh network; the Microcontroller controls dataflow, and the Sensor Block detects emergencies and alerts the Microcontroller.

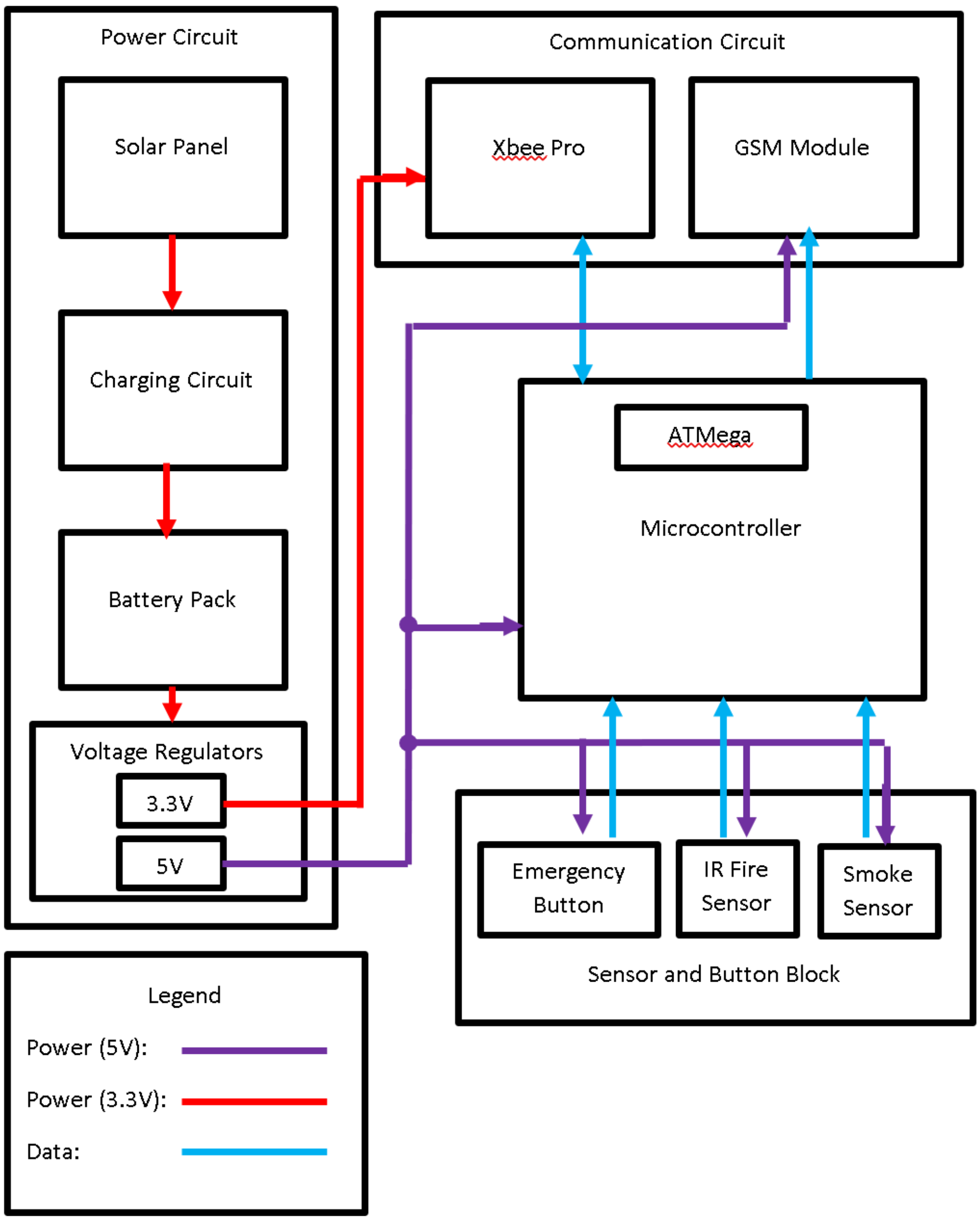

Figure 1: Block Diagram

Our physical design is meant to contain the electronics while also being environmentally friendly. For this reason, we designed the units to be contained in birdhouses. The electronics are separated from the rest of the birdhouse and are contained in a box mounted on the bottom as shown in Figure 2. There are cutouts for the IR flame sensors and the smoke sensor and the final units that we built and tested are shown in Figure 3.

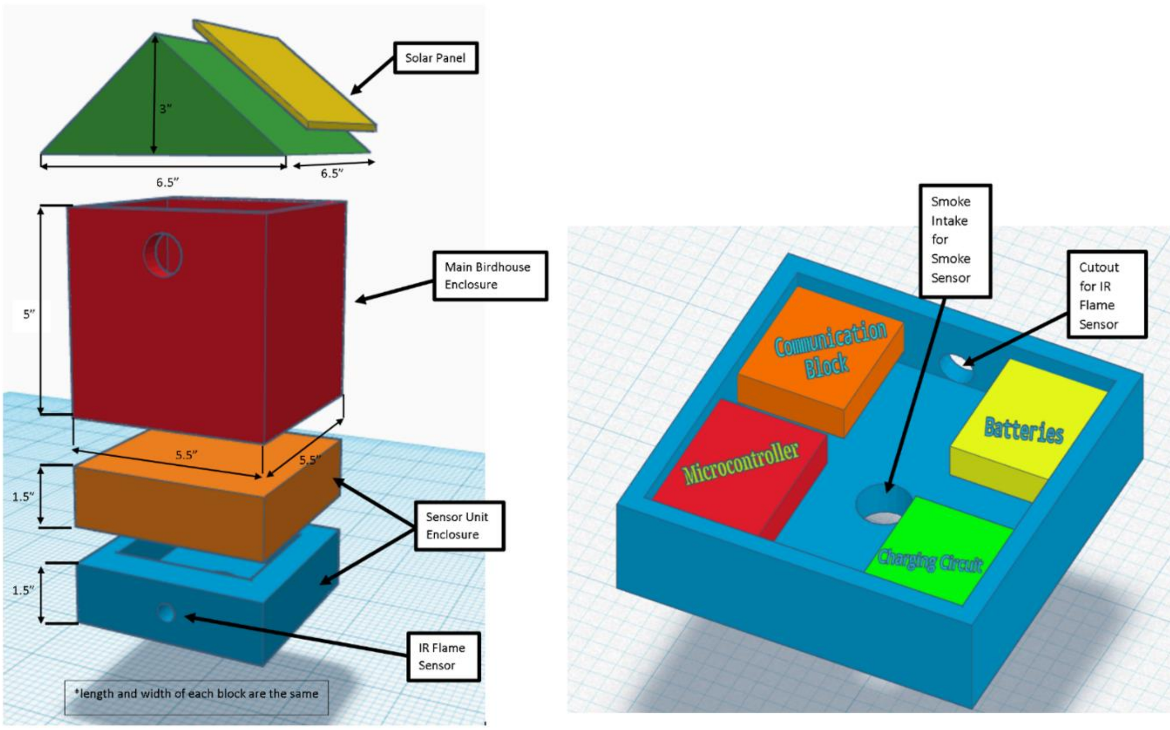

Figure 2: Original Physical Design

Figure 3: Completed Design

2.1 Power Circuit

The Power Circuit block is responsible for both the power generation and distribution for the entire sensing unit. This block is focused around the battery pack sub-block. The Power Circuit block also contains the Solar Panel and Charging Circuit sub-blocks which allow the unit to self-generate power so that it can continuously recharge the battery pack. Voltage regulators are used to regulate the variable battery voltage to the specified operating voltage of components.

2.1.1 Solar Panels

Each unit will be placed in a remote location in a forested area. For this reason, they will all be equipped with two solar panels to recharge the Lithium Ion batteries and in result, reduce the maintenance requirements. The two panels will be placed on either side of the birdhouse roof. We chose 5V, 2.5W polycrystalline solar panels because they are affordable, and when the solar panels are wired in parallel, they can produce a maximum output of 5W at 1000mA; this means the solar panels can work at roughly 50% efficiency on average and still produce enough current to charge the Lithium Ion batteries each day. Furthermore, the total average load draws 216.5mA which is 5196mAh per day, and with an average of 10 hours of daylight per day, the solar panels need to be able to produce an average of 519.6mA. This current needs to account for the effect of obstructions due to trees and cloud coverage.

2.1.2 Lithium Ion Battery Charger

For charging the batteries, we will be using the TP4056 Linear Li-Ion Battery Charger. This device takes an input voltage from the solar panel and its output terminals are connected to a single battery cell as shown below in Figure 4. The input voltage from the solar panel must be 5V and the current must be below 1A [9]. The charging circuit also has a few safety features to protect the batteries from over-charge, over-discharge, and over-current draw; the charging circuit is designed with an over-charge cut-off voltage of 4.2V±1.5%, an over-discharge cut-off voltage of 2.5V, and an over-current protection of 3A [9]. This provides us with an added layer of protection to ensure that the Lithium Ion batteries never become over-heated are being handled with care [13].

2.1.3 Battery Pack

This is a very crucial component of the design because choosing the correct battery capacity will determine whether the system will be able to function continuously or not. Our design has an average load of 216.5mA which amounts to 5196mAh per day. We chose to use four 3.7V, 2500mAh 18650 Lithium Ion batteries per sensing unit. We chose this battery because of its low cost, high capacity, and rechargeable capabilities. Initially, we wanted to wire all four of these batteries in parallel with each other to achieve a total theoretical capacity of 10000mAh. This design provided us with plenty of capacity even if the batteries did not work at their relative specified capacities. But, while trying to implement this design, we came across an obstacle; we were attempting to use a boost converter with no feedback control to step up the 3.7V output of the battery pack to 5V. This was achievable under no load and constant load conditions but not under a varying load. Our sensing units have a varying load depending on whether it is sending or receiving messages. Therefore, when the load would change, the voltage output of the boost converter would spike, and this ended up burning two of our ATmega328P processor chips. Consequently, we altered our design of the four-cell battery pack; we decided to wire two batteries in parallel in series with two other batteries in parallel as shown in Figure 4. This design provides a voltage output of 7.4V from the battery pack allowing us to easily step down the voltage to 3.3V and 5V using voltage regulators; however, this design yields a theoretical battery pack capacity of 5000mAh which is 196mAh below our requirement.

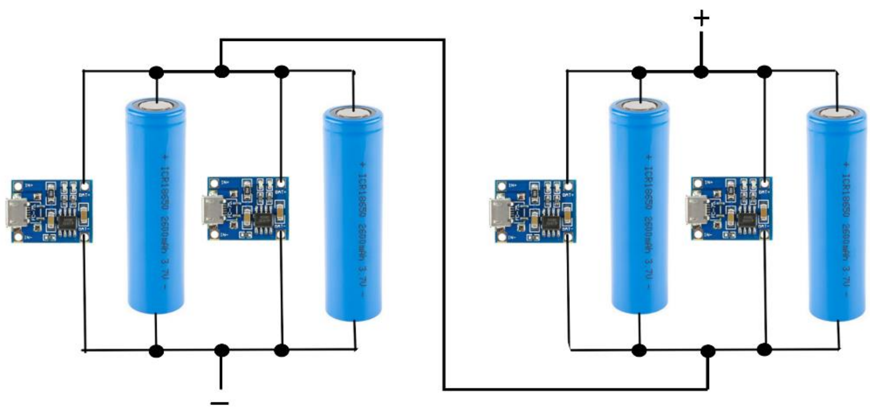

Figure 4: Battery Pack Design

2.1.4 Voltage Regulators

To achieve a constant output voltage of 3.3V and 5V we will use the UA78M33C voltage regulator and MC7805CT respectively. We chose the 5V regulator because it can convert an input voltage of 12±6V to a constant 5±0.2V output, and it also has a maximum output current of 2200mA which is well above what our system requires [3]. Next, we chose the 3.3V regulator because it can convert an input voltage of 15±10V to a constant 3.2±0.1V output, and it also has a maximum output current of 700mA [15]. These specifications provide flexibility in terms of the battery voltage output as well as flexibility in current capacity for possible future upgrades. No RV table is necessary for this device because there are no requirements that it must meet that are not specified in the datasheet.

2.2 Microcontroller PCB

Microcontroller block will be implemented using the ATmega328P. The ATmega328P will be programmed to handle signals from both Communication Circuit sub-blocks and all the Sensor and Button sub-blocks. The ATmega328P is the brains of the sensing unit and handles all I/O in the sensing unit and will determine when to send emergency alert signals notifying someone that there is a fire in a specific location.

2.2.1 ATmega328P

The microcontroller that will process and transmit all data from the various components is the ATmega328P 28 Pin PDIP by Atmel; it is an 8-bit AVR microcontroller in the megaAVR family [5]. This microcontroller was chosen for its cost-effectiveness and dynamic functionality. The chip is USART device making it compatible with a wide range of other devices. Also, the chip has 23 general I/O pins and a max processing speed of 16MHz, which is necessary since the processor will need to do multiple computations simultaneously for extended periods of time. This will also provide flexibility in terms future design upgrades to the units [5]. Most importantly, the ATmega328P has five different power saving modes that can be software enabled; these power saving modes could help drastically reduce power consumption and pro-long the battery life [5]. The processing power on the chip along with the simplicity of the Arduino based programming language will allow for extremely fast data analysis allowing quick Fire detection. The ATmega will be soldered onto the PCB shown in Figure 5.

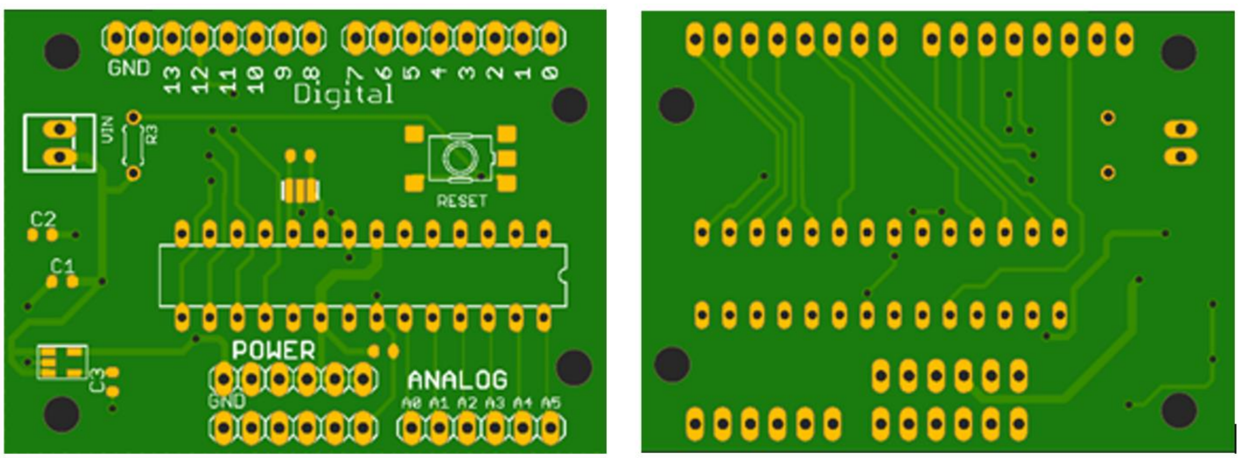

Figure 5: Top and Bottom Layer of Final PCB Design

2.3 Communication Block

The communication block consists of two sub-blocks, the XBee-PRO and the GSM Module. The XBee-PRO sub-block transmits and receives signals and is what allows the Mesh System to be possible as the signals transmitted by the XBee-PRO can be sent to multiple units and even relayed through another unit's XBee-PRO. The other sub-block in the Communication Circuit block is the GSM Module. This sub-block will only transmit signals and would only be installed on the master units. The sole purpose of this sub-block is to send emergency alerts via text message if any of the sensing units have detected a fire.

2.3.1 XBee-PRO

One of the most important aspects of this project is the mesh network functionality. This functionality is implemented using the XBee-PRO module. These modules use the Zigbee communication standard to provide low energy mesh network solutions for communication. They use the IEEE 802.15.4 networking protocol for RF communication. For our project to offer sufficient coverage, we require communication distances up to 100 meters in forested areas. For this reason, we have chosen a higher powered XBee-PRO module with a transmission power of 60mW. This module has a maximum communication distance of 1-mile LOS [4]. Considering trees and other obstructions, this module should be able to meet out 100-meter requirements.

2.3.2 GSM Module

The main advantage that this sensor array has over current forest fire detection systems is its ability to detect fires before they are out of control and alert the user quickly. To send alerts to the user we will be using the GSM network to send text message alerts containing an emergency message with the location of the incident. This will be accomplished using the SIM900 GPRS/GSM board along with a SIM card. This allows the unit to access the 2G GSM network. We chose this board because it is an inexpensive solution that has a very low current draw of 1.5mA while in sleep mode [10]. The only time the board will be drawing significant power is when it is awoken and begins to transmit data which will only be when an emergency signal needs to be sent due to either a fire being detected, or the emergency button being pressed.

2.4 Sensor Block

The Sensor and Button block consists of three sub-blocks: Emergency Button, IR Fire Sensor, and the Smoke Sensor. The Emergency Button allows the sensing unit to function as both a fire detection unit and an emergency alert system for someone who is stranded in the woods. Since these units would theoretically be spread across the wilderness in remote areas, the idea was to have them also function as emergency alert systems. The IR Fire Sensor and Smoke Sensor sub-blocks are what allow the sensing unit to detect fires. When high levels of smoke or flames are detected, they will alert the microcontroller that a fire has been detected.

2.4.1 Smoke Sensor

The smoke sensor module is the first line of defense against forest fires. Smoke will be present before any flames are close enough for the flame sensor to detect. We chose to use the MQ-2 Smoke Detector Board because it is inexpensive and sensitive enough to detect smoke at concentrations of 300 to 10,000 ppm [11]. For our project, we do not want the sensitivity to be set at such a low value because this would likely cause many false alarms. The smoke sensor will be tuned to a specific value using a potentiometer. Additionally, we chose these smoke sensor modules because they are simple in design making it easy to comprehend how smoke sensors work. For example, this smoke sensor module uses the MQ-2 sensor which essentially acts as a varying resistor depending on the concentration of smoke or gas in the air; as the resistance of the MQ-2 sensor decreases, the voltage drop across the sensor decreases, outputting a higher voltage into the voltage comparator. This pulls the output of the voltage comparator to ground which then lights up the LED notifying there is smoke or gas in the air as shown below in Figure 6. Thus, now that we understand the functionality of the module, we can redesign this module onto our own PCB; as a result, we would be creating a cheaper product.

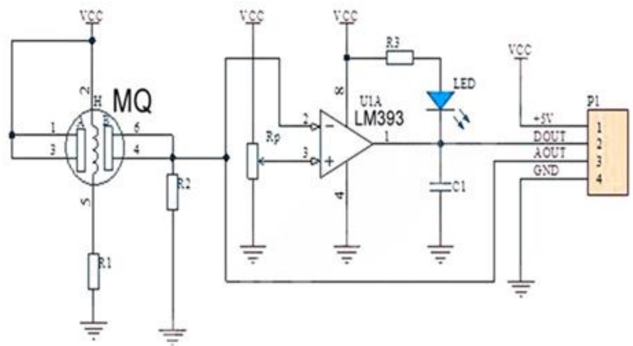

Figure 6: Smoke Sensor Schematic [7]

2.4.2 Flame Sensor

We chose the flame sensor because it is a cost-effective option that can detect IR radiation at a distance. The sensor has a detection angle of 60 degrees and detects wavelengths between 760nm and 1100nm [6]. The documentation states that the sensor can detect flame from a lighter at 80cm. We suspect that, with the increased size of the flame source, the sensor should be able to detect the flame a greater distance. For this sensor to be useful we require it to detect a flame at 10-meters. Additionally, we chose this flame sensor module for its simplicity as we did with the smoke sensor module. The flame sensor module works nearly the same way the smoke sensor module does; except, instead of the MQ-2 sensor acting as a varying resistor, the photo detector, indicated as the sensor in Figure 7, in the flame sensor module acts as a varying resistor depending on the amount of photon absorption. Again, now that we understand this module, we can redesign the module onto our own PCB; creating a cheaper more efficient product as a result.

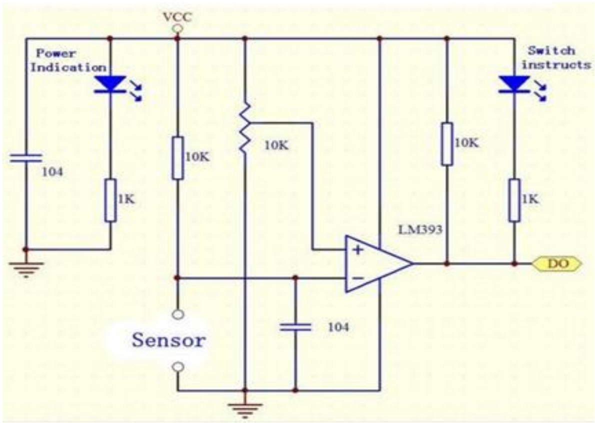

Figure 7: Flame Sensor Schematic [6]

2.4.3 Emergency Button

We will include an external two pin push button, depicted in Figure 12 below, which will allow people who may be lost in the wilderness or in need of help to request assistance. When the button is pressed, the sensor unit sends an emergency alert indicating that the emergency button has been pressed and tells the operator where the alert was sent from. This is an inexpensive addition to the project that is very simple but has the potential to save lives.

3. Design Verification

3.1 Power Circuit

3.1.1 Solar Panels

To verify the functionality of the solar panels, we had to test whether they could meet our systems energy requirement of 5196mAh of capacity production per day. This means that in the ten hours of sunlight per day on average two solar panels connected in parallel must be able to output more than 519.6mA on average throughout those ten hours. We performed a variety of tests on solar panels while connected in parallel. The current and voltage output measurements were taken in full sunlight, in a forested area, and on a cloudy day.

Based on our results shown below, we were able to verify that the solar panels would be able to meet the requirement of 519.6mA producing an average output current of 980mA in direct sunlight. Furthermore, figure 8 and figure 9 reveal how the power output of the solar panel correlates the amount of sunlight exposure the solar panel is receiving. As shown, as the solar panel gets covered, the current output of the solar panel exponentially decreases, and since the solar panels will be located in forested areas, the solar panels would not work efficiently enough to charge the batteries. Specifically, when we tested the solar panels in a forested area under a tree, the two solar panels wired in parallel had a current output of just 40mA. A current output of this magnitude would clearly not be enough to recharge the batteries daily. Thus, we would need to change our original design and mount the solar panels in a location where they would be fully exposed to sunlight rather than mounting the solar panels directly on top of the physical birdhouse which would be attached to a tree. An alternative option would be to wire each sensing unit's solar panels to the top of their relative tree that the birdhouse is attached to. This would fully expose the solar panels to sunlight allowing us to far pass the desired requirement of 5196mAh of capacity production per day.

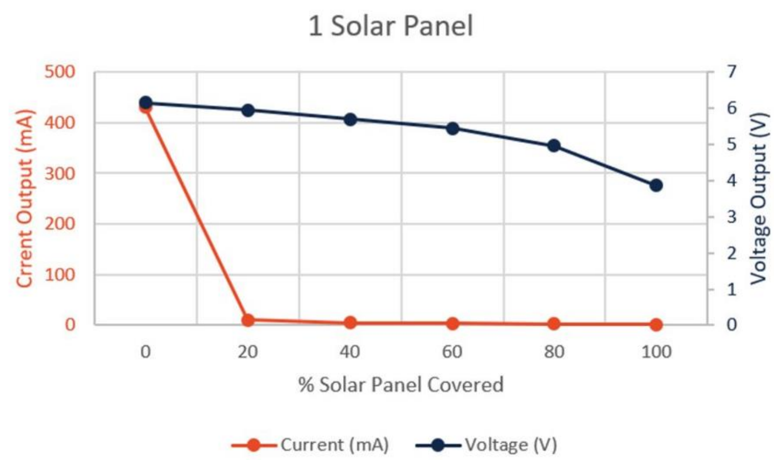

Figure 8: One Solar Panel Output Current

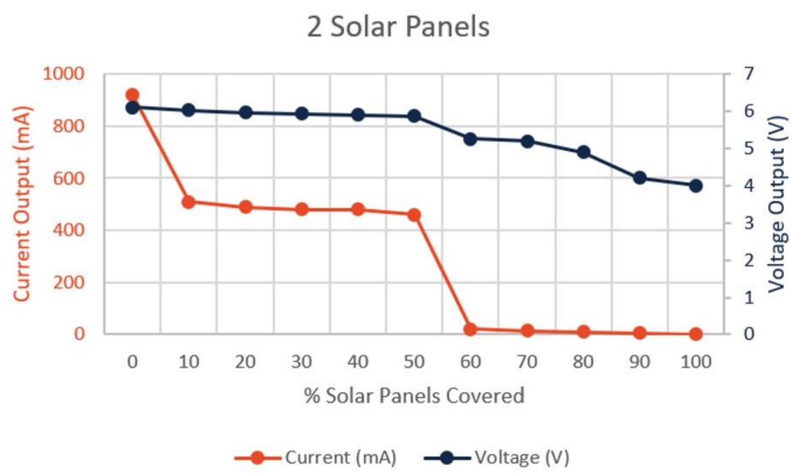

Figure 9: Two Parallel Solar Panel Output Current

3.1.2 Battery Pack

The first requirement for the 4-cell battery pack was that it must have a capacity of at least 5196mAh so that the sensing unit can last one full day without being charged; this is assuming the sensing unit draws an average current of 216.5mA over 24 straight hours. The second requirement was that the individual batteries must be able to be fully charged to 3.7V solely using the output of the solar panel. Thus, to verify our requirements, we had to test the battery capacity of each individual Lithium Ion battery as well as recharge the individual Lithium Ion batteries solely using the solar panels.

To verify the first requirement, we designed a battery draining circuit that drained the battery at a specified current draw and used an Arduino Uno microcontroller to measure the voltage every ten minutes. First, we drained a single Lithium Ion battery at a constant current draw of 225mA; with the battery having a specified capacity of 2500mAh, the battery should have taken 11.11 hours to fully drain. However, the battery fully drained to 2.5V, when the charging circuit applied its over-discharge cut-off, in just about 2.5 hours giving the battery a capacity of just 563mAh. This test was repeated with a constant current draw of 50mA; with the specified capacity stated earlier, the battery should have taken 50 hours to fully drain. However, the battery fully drained to 2.5V in just about 11.5 hours giving the battery a capacity of just 575mAh. Finally, to verify the second requirement, a solar panel charging test was performed on a single Lithium Ion battery; the battery used had been fully discharged from the previous test. Using a 60-Watt lamp, we were able to create a constant current output of 40mA and a voltage output of 5V from the solar panel. We then connected the terminals of the solar panel to the input terminals of the charging circuit and again, measured the voltage every ten minutes. With this setup, it took the single battery about 14.5 hours to fully recharge back to 3.5V giving the battery a capacity of just 580mAh.

The results of these tests are shown below by Figure 10 and prove that the actual capacity of the batteries is just about one-fifth of what the specification for the Lithium Ion battery stated. This will not allow us to meet our requirement of 5196mAh for the battery pack; with the new design and the actual capacity of these batteries considered, the capacity of our battery pack is an estimated 1200mAh. Ultimately, these results give us no other option but to do more research to find a new battery with a higher capacity even though we successfully proved the batteries can be fully recharge solely using the solar panels.

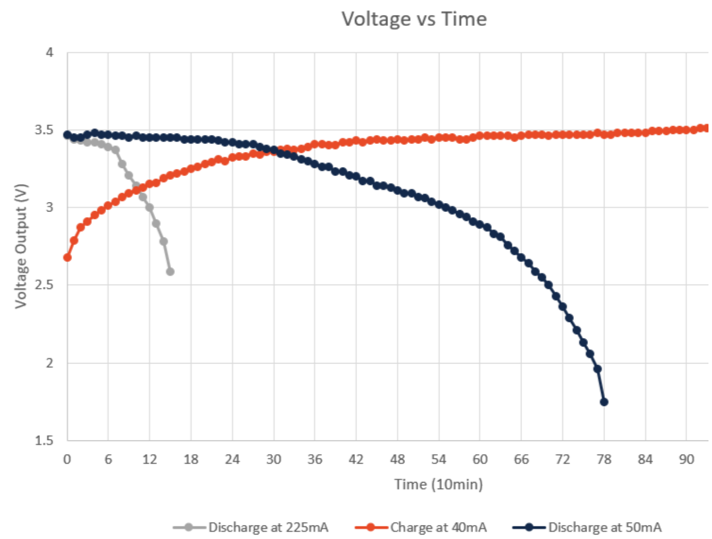

Figure 10: Charging and Discharging Batteries

3.2 Microcontroller PCB

3.2.1 ATmega328P

The ATmega is the brain of the sensing unit and is responsible for processing sensor data as well as hosting the mesh network and all communication. Our project had two requirements that the ATmega had to meet. The first was is that it needed to be able to process data from at least four input pins simultaneously, and the second is that it needed to be able to communicate with both the XBee-PRO and the GSM module through two different serial ports.

The final PCB design is shown below in Figure 11 with the ATmega and all other required hardware soldered to the board. To verify the requirements for this block we assembled the circuit shown in Figure 12 with the PCB at the center. This circuit has all our systems sensors and modules connected and should be able to function as expected if the requirements are met. After the circuit was assembled, we then tested each sensor to see if the ATmega could respond to all sensor inputs. We then tested the serial communication and were able to verify that the ATmega can indeed meet all our requirements.

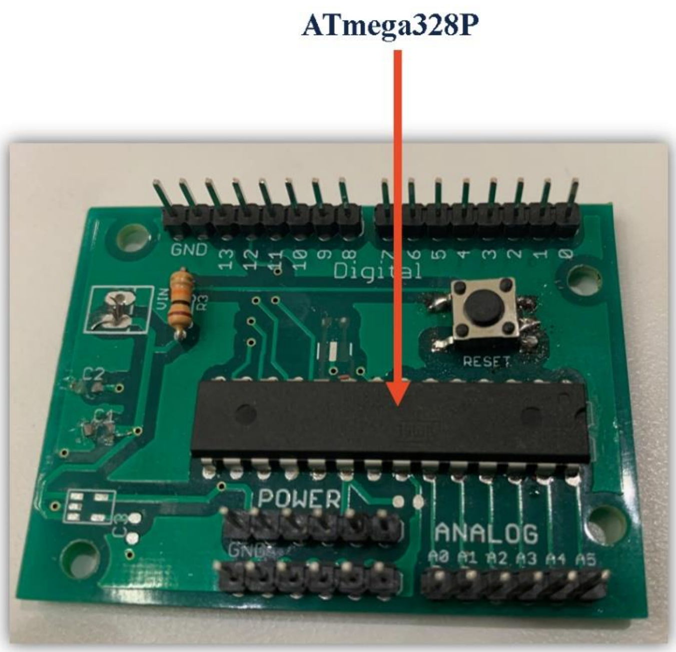

Figure 11: Final Assembled PCB

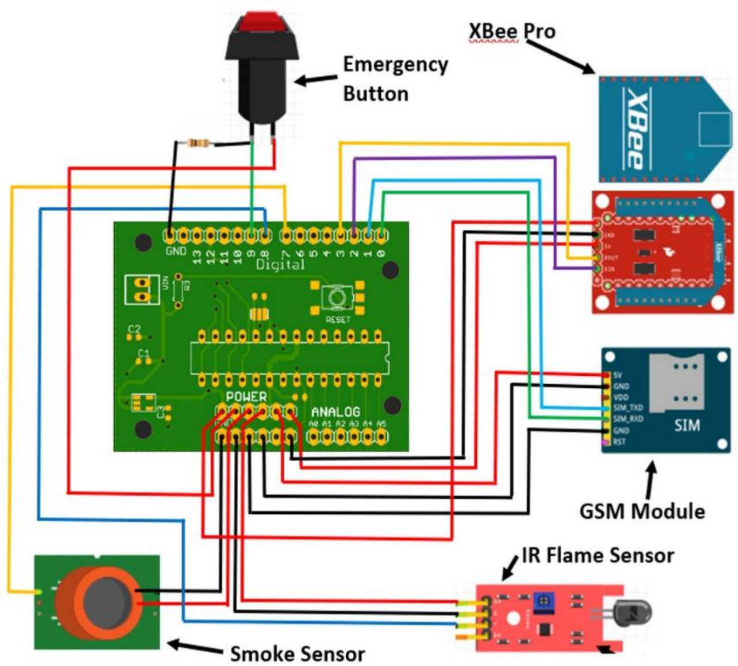

Figure 12: Full Sensor Unit Circuit Setup

3.3 Communication Block

3.3.1 XBee-PRO

The XBee-PROs are a crucial component in our design because they form the mesh network. Our requirement for these modules is that they needed to be able to communicate at up to 100 meters in a forested area. To verify this requirement, we began by connecting two XBees to two different Arduino Uno microcontrollers. We then went to a forested area and tested the maximum communication distance by progressively moving further apart until communication was no longer possible. We performed four different trials, obtaining the data shown in Table 1 below.

Each of the communication trails yielded a result that was greater than the required 100-meter distance and an average distance of 161-meters. Based off these results, we were able to conclude that the XBee-PRO units functioned as required.

Table 1: Maximum XBee Communication Distance

Trial Number	Maximum Communication Distance (meters)
1	110
2	157
3	108
4	269
Average	**161**

3.3.2 GSM Module

The GSM module requirement is that it must be able to reliably send SMS messages while located in a forested area. To test this, we connected the GSM module to an Arduino and wrote a sketch that sends a simple SMS message to a cell phone. We then went to a forested area and attempted to send messages and see if they were received. The results of this test are shown in table 2.

This requirement was easily verified, and 100 percent of the messages sent were received successfully.

Table 2: GSM Communication Test

Trial Number	Message Received Successfully (Y, N)
1	Y
2	Y
3	Y
4	Y
5	Y
Percent Received	**100**

3.4 Sensor Block

3.4.1 Smoke Sensor

Our original smoke sensor requirement was that it could detect smoke concentrations of 0.5±0.25% ppm. As the project progressed, we realized it would be difficult to measure an exact smoke concentration for testing. For this reason, we instead created a concentration of smoke in the air that would be a clear sign of a fire. To do this, we created different size small fires using paper and carboard. We then connected the smoke sensor to the Arduino Uno and measured the output to see if it could detect the smoke. The results are shown below in table 13.

Using the potentiometer to change the sensitivity of the smoke sensor, we were able to calibrate it to detect a wide range of smoke concentrations. It could be set to detect a small amount of smoke from a burning 1”x1” piece of paper. This setting could produce false positives, so we set it to a higher value where it would detect a larger amount of smoke from burning cardboard. Testing this sensor ethically was difficult because we were unable to measure an exact smoke concentration to test. Based on the data shown in Figure 13, we were able to verify that the smoke sensors function as desired for our application but further testing to determine exact smoke concentration detection levels is necessary.

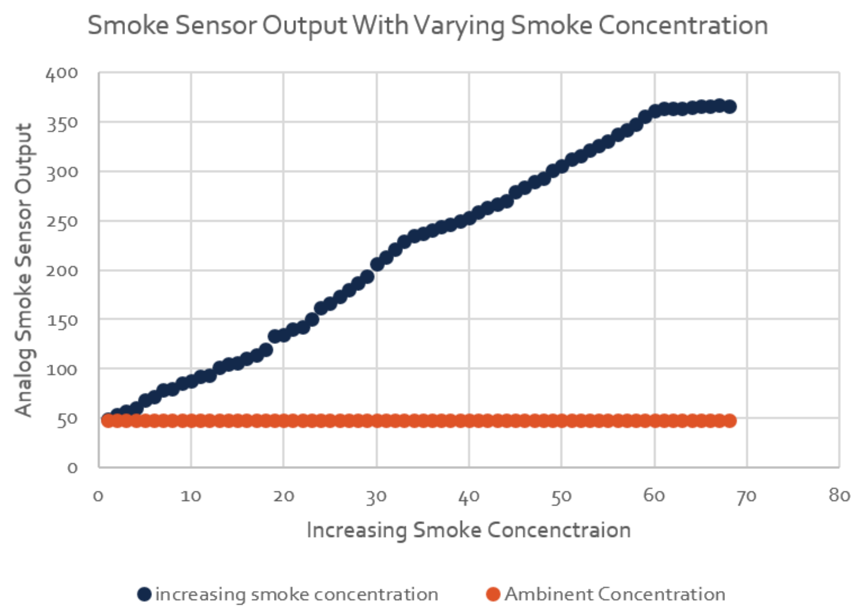

Figure 13: Smoke Sensor Analog Voltage Output

3.4.2 Flame Sensor

Verifying the flame sensor requirement proved more difficult than we originally expected. Our requirement for the flame sensor was that it must be able to detect a flame at 10-meters. To verify the flame sensor requirement, we measured the maximum detection distance that the flame sensor could reliably detect a flame. To do this we created flames using a lighter for one trial and burning paper for another. We then moved the sensor away from the flame until it could no longer detect it. We recorded this distance as the maximum detection distance for that flame size. The results of these tests are shown below in Figure 14.

We were unable to verify our requirement because we could not find a way to create a large enough flame source in a safe and ethical way. We were however able to show that increasing the size of the flame source increases the maximum detection distance. For future tests, we will research safer methods of creating a large fire or alternative ways of simulating the fire.

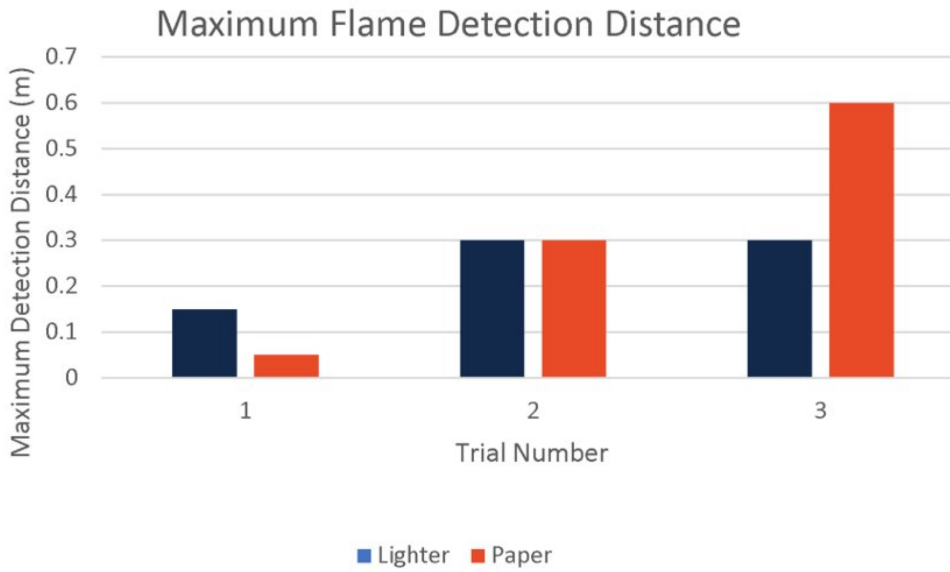

Figure 14: Maximum Flame Sensor Detection Distance

4. Costs

4.1 Parts

Table 3: Part Costs Per Node

Part	Manufacturer	Retail Cost ($)	Bulk Purchase Cost ($)	Actual Cost ($)
18650 Batteries	Samsung	24.63	11.00	24.63
XBee-PRO	DIGI	39.45	N/A	39.45
GSM Module	SainSmart	32.99	N/A	32.99
Solar Panels	ALLPOWERS	15.99	N/A	15.99
Charging Circuits	XCSOURSE	4.79	N/A	4.79
Smoke Sensors	SUKRAGRAHA	6.99	0.99	6.99
Flame Sensors	Atomic Market	6.99	1.80	6.99
Buttons	Gikfun	0.87	N/A	0.87
Wood	Home Depot	8.12	N/A	8.12
SIM cards	Ting	5.99	N/A	5.99
XBee Adapter	Waveshare	13.99	N/A	13.99
ATmega328P	Atmel	3.67	1.79	3.67
Total		**160.48**	**137.77**	**160.48**

4.2 Labor

Labor costs constitute most of the cost of the project. We estimate that each group member spent an average of eight hours on the project over the past 16 weeks and that the cost of our labor is $40 per hour. With three group members we obtain the result in Equation 1:

$$Labor\ Cost = 3 * 16\ weeks * \frac{8h}{week} * \frac{\$40}{h} * 2.5 = \$38{,}400 \tag{1}$$

5. Conclusion

5.1 Accomplishments

Designing, building, and testing the fire detection system was a difficult endeavor that required a considerable amount of research to complete, but by the end of the semester, we were able to demonstrate our project functionality with few issues. The communication block of the design functioned well without any issues and met all our systems requirements. The XBee-PROs and GSM modules were able to communicate and send alerts quickly. The Microcontroller block also met all its requirements and was able to process sensor data and control the communication and mesh network functionality. The emergency button proved to be very responsive and could be used to send for help in the event of an emergency. As for the power circuit, we were able to verify that the solar panels, charging circuits and voltage regulators. After testing each block component individually, we were then able to integrate the components and verify the functionality of the entire system.

5.2 Uncertainties

Although we were able to demonstrate the systems functionality, there were a few concerns we had with the design. The main concern we had with our design was the batteries. Our battery pack requirement stated that 5196mAh of capacity was necessary for our system to operate indefinitely without the need for manually recharging the batteries. We were unable to meet this requirement in our verification tests. We also had issues with our verification for the smoke and flame sensors. The problem with verifying these sensors was that it required creating a large fire which proved to be more difficult to accomplish ethically than were initially anticipated.

5.3 Ethical Considerations

When testing all parts of our project, we followed IEEE code of ethics Section 7.8.1 by making sure all our tests were done in a safe an ethical manner and did not endanger the health or welfare of the public or the environment [12]. This is especially important because testing our systems involves creating controlled fires and smoke. For this reason, we were unable to properly verify the requirements for the flame sensor because creating a fire large enough would have been a violation of the IEEE code of ethics. We also followed Section 7.8.7 by asking for and accepting any criticisms of our current design. We will address any issues that are present in the design and give credit to those that contribute.

5.4 Future Work

Our projects design worked well in many ways but there are many ways in which it can be improved. For future work on this design, we first will research alternative power supply options and try to move away from using the 18650 lithium ion battery cells. We also will research a more safe and ethical way of verifying the smoke and flame sensors to find an optimal way to avoid false alarms and ensure that they meet our requirements. Aside from these three components, the rest of the design functioned as expected and the main area of the design that we want to improve upon is the cost. The bulk price of $137.77 per node is expensive when hundreds of the nodes would be needed to implement our system in a large-scaled forest. For this reason, we will research alternative options for some of the more basic components. One alternative option that we are considering is integrated some of the commercial components into the PCB design; specifically, looking into designing our own flame and smoke sensor modules on our own PCB.

References

[1] R. Barbero, J. T. Abatzoglou, N. K. Larkin, C. A. Kolden, and B. Stocks, "Climate change presents increased potential for very large fires in the contiguous United States," *International Journal of Wildland Fire*, Jul. 2015.

[2] C. Fowler, "Human Health Impacts of Forest Fires in the Southern United States: A Literature Review," *Journal of Ecological Anthropology*, vol. 7, no. 1, pp. 39–63, 2003.

[3] Texas Instruments, "µA78Mxx Positive-Voltage Regulators," UA78M33C datasheet, Jun. 1976 [Revised Jan. 2015].

[4] DIGI, "XBee/XBee-PRO S1 8.15.4 (Legacy)," XBee-PRO datasheet, Dec. 2015 [Revised May. 2018].

[5] Microchip, "AVR® Microcontroller with picoPower® Technology," ATmega328/P datasheet, Jun. 2016 [Revised Nov. 2016].

[6] SEEED, "Grove - Fingerprint Sensor," The Grove - Flame Sensor datasheet, Sep. 2015.

[7] SEEED, "Grove - Gas Sensor (MQ2) User Manual," The Grove - Gas Sensor(MQ2) datasheet, Sep. 2015.

[8] "Illuminance - Recommended Light Level," *Young's Modulus of Elasticity for Metals and Alloys*. [Online]. Available: https://www.engineeringtoolbox.com/light-level-rooms-d_708.html. [Accessed: 04-Oct-2018].

[9] NanJing Top Power ASIC Corp., "TP4056 1A Standalone Linear Li-lon Battery Charger with Thermal Regulation in SOP-8," TP4056 datasheet.

[10] SIMCom, "SIM900D_Hardware Design_V2.02," SIM900D datasheet, May. 2012.

[11] HANWEI ELETRONICS CO.,LTD, "TECHNICAL DATA MQ-2 GAS SENSOR," MQ-2 datasheet.

[12] "IEEE Code of Ethics," *IEEE - Advancing Technology for Humanity*. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 04-Oct-2018].

[13] Salz, "Safe Practice for Lead Acid and Lithium Batteries." 04-Jul-2016.

[14] E. Siegel, "The Terrifying Physics Of How Wildfires Spread So Fast," *Forbes*, 06-Sep-2017. [Online]. Available: https://www.forbes.com/sites/startswithabang/2017/09/06/the-terrifying-physics-of-how-wildfires-spread-so-fast/#478e757e7791. [Accessed: 16-Oct-2018].

[15] ON Semiconductor, "1.0 A Positive Voltage Regulators," MC7805CT datasheet, May. 2013

Appendix A

Table 4: System Requirements and Verifications

Requirements	Verification	Verification status
Solar Panels 1. Supply an average of 519.6mA at 6±2V to the charging circuit to meet the 5196mAh daily requirement	1. Place the solar panel in direct sunlight ~107,527 Lux [8]. 2. Use a voltmeter to measure the output voltage of the panel and make sure it is below 8V (the maximum for the charging board) [9]. 3. Use an ammeter to measure the output current and make sure that it is less than 1000mA (the maximum current for the charging board 4. Repeat the above steps under tree cover and measure the output current to ensure that the average current will be above 519.6mA	Y

Battery Pack 1. The three-battery system must have a total capacity greater than the basic system requirement of 5196mAh 2. The individual batteries must be able to be charged to 3.3V using the output from the solar panel	1. Construct the circuit shown in Figure 4 2. Connect a load to the positive and negative terminals and hold current constant 3. Measure the amount of time it takes to discharge the batteries and ensure that it is greater than 5196mAh 4. Connect the output of the solar panels to each of the charging circuit boards 5. Use and ammeter to measure the current into each of the individual cells when the charging circuits are on.	N

ATmega 1. Must be able to continuously process data from at least four input pins 2. Must have the ability to communicate with both the GSM module and XBee-PRO via serial communication port	1. Connect all four input lines to sensors and XBee-PRO to ensure the chip has enough computing power to process all the information 2. Transmit data from one XBee-PRO to another and check the signal integrity to ensure the serial communication port is compatible with the XBee-PRO 3. Transmit data from the GSM module to a phone and check the signal integrity to ensure the serial communication port is compatible with the GSM module	Y
XBee Modules 1. Must be able to communicate with other XBee modules at distances up to 100 meters in forested area.	1. Connect two XBee modules to separate Arduino Unos for testing 2. Find a forested area such as a park or forest preserve 3. Measure distances in 10-meter increments between the two XBees and test communication between them 4. Move further away until the modules can no longer communicate 5. Record the maximum communication distance and make sure it is greater than 100 meters.	Y

GSM Modules 1. Must be able to connect to a 2G GSM network when the sensor unit is in a remote location	2. Connect the GSM module to the Arduino’s serial pins 3. Find a forested area such as a park or forest preserve 4. Use AT commands to send SMS alerts to a phone 5. Make sure that the GSM signal is strong enough to alert the user	Y

Smoke Sensors 1. Must be able to accurately detect smoke at concentrations of 0.5±0.25% ppm. 2. Must be able to distinguish between real forest fires and false alarms such as nearby controlled camp fires 3. Must be able to distinguish between real forest fires and false alarms such as nearby controlled camp fires	1. Connect the gas sensor to an Arduino for testing 2. Produce a concentration of smoke between 2500 and 7500 ppm. 3. Monitor the digital I/O pin on the smoke sensor which will indicate when it detects the smoke 4. Record the time it takes to detect the smoke 5. Measure smoke concentrations in air on a burn day 6. Measure smoke concentrations when having a nearby camp fire 7. Measure smoke concentrations in smaller confined space with built up smoke 8. Determine a threshold smoke concentration for consistent general fire detection 9. Add a constraint that requires the threshold smoke concentration to be present for 5 seconds before triggering 10. Test these parameters on a variety of fire types and adjusts the parameters as needed	Y

Flame Sensor 1. Must be able to accurately detect a flame at 10-meters	2. Connect the flame sensor to an Arduino for testing 3. Create a 1 square meter flame source in a fire pit 4. Monitor the digital I/O pin on the smoke sensor which will indicate when it detects the smoke 5. Move further and further away from the fire until the sensor can no longer detect the flame 6. Record the maximum detection distance and verify that it is greater than 10-meters	Y