Remote Trap Activation Detection using IoT

Muhammad Tanveer Ahmed

Supervisor: Dr Charith Perera

Moderator: Dr Hantao Liu

A report presented for the degree of
Computer Science BSc. (Hons).

Cardiff School of Computer Science & Informatics

Acknowledgements

I would like to acknowledge my close friends and family for being supportive throughout the entirety of this project as well as my degree.

Abstract

With the advancement in technologies and civilisation all over the world, natural habitats are being destroyed and minimised. The number of endangered animals is rising and so scientists and researchers are attempting to ensure that their habitats are secured and help rejuvenate areas in which endangered animals reside so that they can prosper. To help learn about endangered animals, they are captured for a short time, released and monitored so that scientists can understand their needs better, in order to ensure that all is done to prevent extinction and support their well-being. Currently, traps are monitored manually and not a lot of investment has been made to make the capture and release of an animal more efficient. This project aims at creating an IoT device capable of notifying researches/scientists when a trap has been activated to prompt a quick response in releasing the animal in order for it to not remain captured for prolonged periods of time.

Contents

1 Introduction **5**
1.1 Background and Research 5

2 Related Work **9**

3 Method of Approach **12**
3.1 Arduino Placement 12
3.2 Casing 13
3.3 Sensors 14
3.3.1 Motion Sensor 14
3.3.2 Sound Sensor 15
3.3.3 Vibration Sensor 16
3.3.4 Accelerometer Sensor 16
3.3.5 Temperature & Humidity Sensor 17
3.4 Notification Method 17
3.5 Limitations and Considerations 18
3.5.1 Environment and Accuracy 18

4 Implementation of Solution **19**
4.0.1 The Arduino and Components 19
4.0.2 Coding the Sensors 20
4.0.3 Coding the SMS Function 23
4.0.4 Combining the Code 23

5 Outcomes and Evaluation **25**
5.1 Test Results 25
5.1.1 Motion Sensor 25
5.1.2 Sound Sensor 26
5.1.3 Vibration Sensor 32
5.1.4 Accelerometer Sensor 33
5.2 Summarising Results & Changes to Arduino and Sensors 36
5.3 Re-configuring the Code & Creating Triggers 37

6 Conclusions **38**

7 Future Work **39**

8 Reflection on Learning **40**

Chapter 1

Introduction

Climate change is a huge topic of discussion affecting many different areas of life in this day and age. It is affecting every country globally in various ways. One major contribution to a chain reaction is deforestation around the globe. The knock on effect of deforestation is the loss of habitats for many different species of animals which can in turn lead to extinction.

To help with the conservation of animals in Borneo, scientists and researches at the Danau Girang Field Centre (DGFC) have been capturing, releasing and monitoring animals that could effectively become endangered from loss of habitat. During this process to capture animals, traps are placed within the rainforest and checked manually on a rotary basis to see if anything has been captured. This requires a lot of manpower, resources and time, whilst not being effective and possibly leaving animals trapped for long periods of time.

After researching for solutions to help with the efficiency of this process it came to my attention that an IoT device could be used to monitor these traps and notify a researcher when a trap has been activated using an Arduino and multiple sensors.

During this project I looked at developing this device and testing the sensors. After collecting data from the tests I analysed the results to detect which results most accurately represent the likelihood of a trap being activated and when a researcher should or should not be notified.

1.1 Background and Research

Focusing specifically on Borneo for this project, Borneo is the third largest island in the world and contains one of the oldest rain-forests worldwide [13]. Borneo is home to a diverse population of species which have been losing their habitat on a daily basis. Over a century ago Borneo was mainly forest land. Today 50%

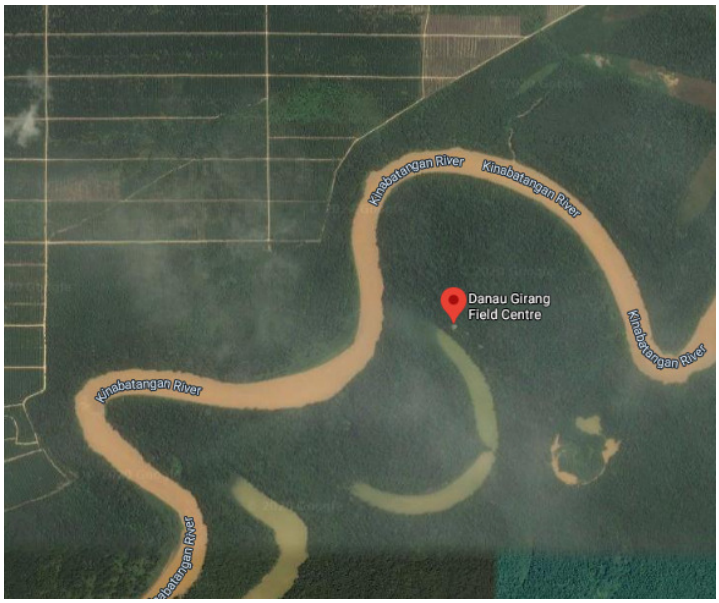

Figure 1.1: Danau Girang Field Centre

of the forest has been destroyed, and between the years 2000 and 2017 over 6 million hectares of forest land was lost in Borneo and was replaced by industrial plantations to collect resources such as palm oil and pulpwood [7].

Borneo is home to different species of animals some of which do not reside elsewhere on the planet. These species are at risk of becoming endangered and researches are monitoring them to ensure that their environmental needs are understood and met. Some of these wonderful animals include the "Clouded Leopard", "Sun Bear", "Malay civet" and the "Orangutan" [20].

The facility which accommodates researchers and scientists is called the Danau Girang Field Centre (DGFC). The DGFC is a collaborative research and training facility managed by the Sabah Wildlife Department and Cardiff University. The main goal of the facility is to further scientific research to help contribute to the long-term conservation of the rain-forest in Borneo and develop a deeper understanding of the environment and the animals that roam within. Another objective is to utilise advanced technologies to help explore the survival mechanisms employed by the flagship species of Borneo and then provide action plans to support the surrounding environment [8]. The DGFC is located in the Sabah area a couple of miles inland from the Kinabatangan River (1.1).

As mentioned previously the DGFC regularly set traps to capture animals along the Kinabatangan River to learn more about the wildlife in the area and help ensure they flourish in their surroundings whilst also understanding their capabilities more. They capture a range of different animals and focus on capturing key animals they believe are at risk of being endangered. They then track and monitor these animals to learn which areas they dwell and scan the areas thoroughly for poachers to safeguard these specific areas. To capture the animals they use a range of traps as can be seen in the figures below (1.2) (1.3).

Figure 1.2: Sunbear Trap

Figure 1.3: Leopard Cat Traps

After doing some initial research and discussing the issue in meetings about the technologies we can utilise to help researchers know when a trap has been activated, it was suggested that I use an Arduino as the IoT device. We needed a device that was small, consumed a low amount of energy and was cost effective. To ease the method of implementation I adopted the method of a Grove System. A Grove System is a modular, standardized connector prototyping system. Grove takes a building block approach to assembling electronics. It allows users to be able to assemble devices such as Arduinos with multiple sensors without having an electronic background and soldering components together [15]. Following the decision to use a Grove System, I researched the types of sensors and connectors that were available which would be suitable to detect whether a trap had been activated. The fundamental sensors that had been used to detect trap activation were primarily motion, sound, magnetic and accelerometers [18].

The final fragment of research done was on how the IoT device would communicate in order to alert researchers. There are six main communication protocols being used within the IoT standard [14] and after more meetings and discussions with a regular visitor of the DGFC I learnt that not all communication methods would be effective. Due to the humidity of the environment, Bluetooth technologies lose their range of communication profoundly and therefore would not be an ideal solution. The other issue with communication is the lack of infrastructure in and around the DGFC. This means that there are currently no networks setup and very few spots have consistent connectivity. Taking all factors into consideration we were left with two options:

- Build infrastructure which will allow the IoT devices to communicate.
- Use SIM cards to send SMS text messages.

Chapter 2

Related Work

In this section we will look at research papers that assisted me in my decisions to ensure that this project was successful. There were a number of research papers that helped me understand how we could monitor the traps and use the sensors to detect a trap being activated.

Previously a former student had done a project on using Bluetooth beacons to track poachers in Borneo [9]. Within this project I could see the technologies that were tested in Borneo and the results that were created to better understand what technologies would be best to use. After testing the Bluetooth beacons in the UK and confirming that his project was successful, unfortunately due to the humidity in Borneo, the Bluetooth's range was massively reduced and did not work the way it was intended. This project was similar to mine as it was needed for the Danau Girang Field Centre (DGFC) to help protect their endangered animals. The focus was on sending alerts to researchers when it was thought that poachers had entered a protected area. After reading this paper and understanding the technologies at hand, I decided to go with SMS as this seemed to be the most reliable technology for the environment in Borneo.

For this project to work, the device I chose needed to be power efficient so that it did not need to be maintained often. It had to operate under low power usage for a long amount of time to be sustainable and have a long uptime [2]. After doing research a lot of the time motion sensors were used to achieve low power consumption and accurately detect intrusion [11]. We could see in both of these papers how using motion sensors allowed for detecting intrusion and using the least amount of battery power to do so. This is why I decided to test a motion sensor with the Arduino and analyse the outcomes as it could be a powerful sensor to use. Following on from this conclusion the rest of the sensors I chose operated efficiently.

There were not many projects that directly linked to using Arduinos to detect a trap activation. Although there were many papers on how Arduinos had been

used to detected intrusion and for security purposes. The overall goal was the same, how could the Arduino and sensors be used to detect that someone or something had been detected in a certain space [3]. After reading through these papers we came across many common sensors that were being used to detect abnormalities. The motion sensor was the most common sensor but sound, vibration and accelerometers had also been used in scenarios linked to objects being moved such as a door being opened. I also looked at environmental monitoring and whether this could be useful for my project. Having the temperature and humidity sensor to monitor the environment was used in other papers to notify people of when certain environments surpassed a threshold and this would also be effective within a trap in case the conditions within a trap were unbearable for an animal [17][10][6].

I came across how Arduinos and other IoT devices were used in smart farms. This was a particularly interesting article as it mentioned having control of the enclosures as well as environmental monitoring [19]. It encouraged me to think that in the future if the traps were modified to have electronic doors and the researchers could not get access to the trap, they could remotely release the animal so that it would be able to survive in case of any issues. The article also mentioned measuring the climate quality but as this was not an important factor for the project I decided not to look further into these sensors. Climate quality would have been an important factor to ensure that the chicken farm could thrive and the livestock would not get ill but for the wild animals this would not be a concern.

I read an article that the National Wildlife Research Centre in the US published, about the difference in monitoring traps remotely rather than manually to try and gauge the positive impact it would have at the DGFC in managing resources. There were many benefits to remotely monitoring traps such as saving cost, time, decreasing fuel usage, decreasing need for human presence and enabling crews to prioritise traps that had been showing signs of activity. This made it evident that having a similar approach for traps around the DGFC would have a positive impact on the research centre and how they went about their daily activities.

Whilst doing research on remotely detecting trap activation there had already been some traps that had been created with the sole purpose of doing so, these trap designs had been patented, documented and published so I read the patents to see how they used sensors to detect trap activation [5]. We can see that for this particular design the use of a motion sensor and a micro-controller to control the doors was the only technologies being used. The motion sensor was placed at the top of the trap in the middle monitoring any motion below it within the trap, as soon as motion was detected, it would send a signal to the micro-controller which would electronically close the doors. The traps used at the DGFC are a lot more old fashioned and have a pressure plate within the centre of the trap and when activated both doors close. Although the designs are significantly different, the use of a motion sensor could still be used within

the same placement to detect an animal within the trap.

The last article that I read which was significant for my project was a wireless sensor network for rodent detection [1]. In this article they explain how they used micro-controllers and a smart network to see whether rodents had been detected in an area. This was an interesting article as if it was possible to set up a smart network around the DGFC we could have a lot more information on where animal are located around the rain-forest. Having this data would allow for the researchers to place traps accordingly and capture desired animals. Although currently with the lack of infrastructure around the DGFC this would be a future project and not compatible with the current project.

Chapter 3

Method of Approach

3.1 Arduino Placement

Before considering other aspects of the project, I needed to understand how the traps worked in order to create a device which would know when a trap has been activated. This had been done in the research stage which had led to the decision on where the IoT devices would be placed. After looking at how the traps worked I decided that for each trap there would be two Ardunios accompanying the cage. This decision had come based off of the fact that there were two types of traps: - Multiple door Traps (1.2) - Single Door Traps (1.3)

For the multiple door traps there would be an Arduino on each trap door (2 max)(3.2). If a trap had been triggered, both doors collapse to then block the entrances and exits of the enclosure. Given that this was the design of the trap it made sense to have an Arduino placed on each entrance. That way when a trap had been activated and the doors shut, the researchers could be notified. Having two points of data showing trap activation ensured a higher chance a trap has been activated rather if one door failed and closed by itself or was knocked by an animal causing it to shut this would possibly notify researchers of a false trap activation.

For single door traps there would be one Arduino on the entrance/exit and one placed within the trap (3.1). Again having an Arduino on the trap door can ensure that when the trap door closes the IoT device would detect that it has been activated and notify the researches. Unlike the multiple door traps there was no second source of verification, therefore an Arduino within the enclosure could monitor other variables to support the fact the trap had been activated and there is indeed an animal trapped inside.

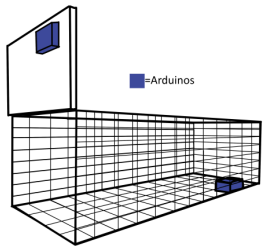

Figure 3.1: Single Door Trap Mockup

Figure 3.2: Double Door Trap Mockup

3.2 Casing

Having an exposed Arduino would cause multiple issues in the rain-forest. Firstly, it could result in an animal consuming the device and harming themselves or possibly picking it up and taking it away. Secondly it means the Arduino would be damaged easily by the weather due to the humidity and causing it to stop working, or possibly malfunction the sensors connected. These and other risks were assessed and meant that the Arduino must be enclosed in a case for durability and protection from the rain-forest. To ensure durability and compatibility with some of the sensors I bought a plastic transparent jar and made slight modifications to make testing easier. The cases dimensions are H = 15.2cm x W = 10.6 x D 10.6cm which created a volume of 0.8L (3.3). Although this case might not be directly used in the rain-forest I used something that was highly compatible and the closest prototype to what a real case may be, which provided durability and protected the components from the environment.

Figure 3.3: Case

3.3 Sensors

After understanding the architecture of where the Arduinos would be placed around the trap, I researched and looked at the fundamental sensors used to detect trap activation [18]. I decided to choose motion, sound, vibration and accelerometers. The reason for not choosing magnetic sensors is because it would require four Arduinos per multiple door trap. Not only would that be less cost effective, if the Arduino at the bottom at the trap was knocked over or moved by the animal, the magnetic sensor would not sense that the trap had been activated. Another issue would be that the magnetic sensors would have to penetrate through two layers of plastic when the trap had been activated which could have been an issue. Therefore, I decided to order the other four sensors to help detect trap activation. In each subsection there are more in depth reasons as to why each sensor was chosen and their purpose and how they were tested. All sensors used in this project and their specifications can be found on the Seeed website [16].

3.3.1 Motion Sensor

A motion sensor is a highly valuable sensor when monitoring movement, which in this case, could help us determine whether a trap has been activated. For single and multiple door traps a motion sensor would be used on both Arduinos, as when the door shuts, this would detect motion on both sensors. Also, on a single door trap, the Arduino placed inside the trap should continually detect motion if an animal is moving around inside. This along with the other factors would suggest that it is highly likely a trap has been activated when motion is detected. The motion sensor I used had a default 3 meter measuring range which can reach up to 6 metres, a holding time of 1-25s and a detecting angle of 120° (3.4).

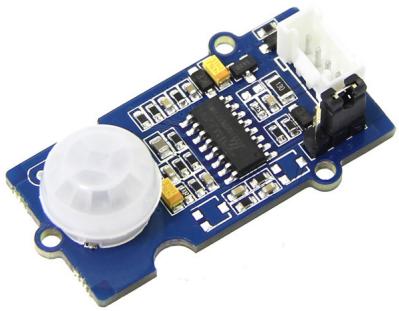

Figure 3.4: Motion Sensor

The way in which the motion sensor was tested, was by placing the sensor within the case and measuring out to 5 meters. From 0-5 meters I walked past

the motion sensor multiple times and noted every time motion is detected. After doing these tests it showed how accurate the motion sensor was within the case and whether it would always detect motion when necessary. I then performed the same tests with the motion sensor outside of the case to see if there were any differences and compared and contrasted the results. This left 2 different tests that were performed on the motion sensor:

- Sensor enclosed in case.
- Sensor outside of case.

3.3.2 Sound Sensor

The sound sensor was used on a single Arduino accompanying the trap. The microphone located on the sound sensor had a sensitivity of 52-48dB and a frequency of 16-20kHz (3.5). When a trap door closes the sound sensor attached to the door should see a spike in results due to the crashing sound. On a single door trap, the sound sensor located inside the trap should have multiple spikes in values if an animal is distressed and moving around inside. These consistent values on a single door trap would give the researchers more confidence an animal is trapped inside. On the other hand using the sound sensor to detect a crashing sound at the same time motion is detected and the accelerometer values change would give more confidence a trap door has shut.

Figure 3.5: Sound Sensor

The sound sensor required more comprehensive testing than other sensors. There were multiple tests done to the sound sensor both within and outside the casing. Firstly, to be as accurate as possible I conducted these tests outside in a more natural area rather than an urban area, so that sounds such as car horns and other city noises could not have given false representation of data. I tested the sound sensor for a significant period of time to create visual representations of the average ambient sounds received both within and outside the casing, comparing and contrasting the difference the casing makes to the surrounding sounds.

I conducted tests mimicking the trap door sounds to help understand what

values would be released in the real world scenario. Furthermore, I attempted to create a surround sound environment where natural tropical rain-forest sounds are played. Again I conducted these tests both within and outside the casing and created visual representations to understand the general sounds created and what values would be normal in this environment.

3.3.3 Vibration Sensor

Initially the vibration sensor would have been located on the doors of the traps to detect if the trap had been activated. After considering the number of sensors already being used to detect the doors being closed it would be unnecessary and not as effective as the accelerometer to add this sensor to the door. The vibration sensor would be used primarily on a single door trap and connected to the Arduino placed inside the trap. The main purpose of the vibration sensor would be to detect consistent movement inside the trap verifying that an animal had been captured. The sensor itself is highly sensitive and waterproof meaning it could be used in traps in more saturated areas.

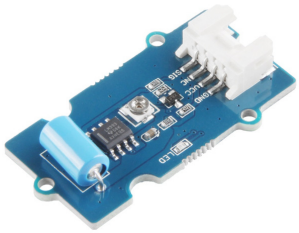

Figure 3.6: Vibration Sensor

The way the vibration sensor was tested, was by producing movement around the sensor and monitoring its values and outputs. Additionally if the animal knocked the casing this would also have created vibration. I mimicked the knocking of the case and assessed the values created.

3.3.4 Accelerometer Sensor

The main purpose of an accelerometer was to focus on the trap door alone. A 3-axis digital accelerometer was the key sensor in detecting when the trap door had shut. It did this by being aware of its orientation and space changing its values when it has been moved. The accelerometer I used has a test range of 16g and 10000g shock survivability (3.7). It was durable enough to be dropped over and over again without malfunctioning being reused for multiple tests.

I conducted drop tests with the Arduino in the case and the accelerometer attached. I did the drop test from multiple different heights to measure the change in values so that I could understand the values created when dropped to match the closing of a trap door. As the accelerometer would always have been

Figure 3.7: Accelerometer Sensor

in the case there was no reason to conduct the experiment outside the case as the values did not differ drastically.

3.3.5 Temperature & Humidity Sensor

The temperature & humidity sensor will not be part of the variables that factor whether a trap has been activated. If an animal is captured it could be ideal to know the current environmental conditions. Having this insight will allow researchers to see if the animal may be at a higher risk of dehydration if the current environment is more humid and tropical. For example if two traps have been activated and one traps micro-climate may be hotter than the other, researchers can prioritise on releasing that animal first (3.8).

Figure 3.8: Temperature & Humidity Sensor

3.4 Notification Method

As mentioned previously due to the environmental conditions only a few technologies would be available to notify researchers. I had identified that there were two possibilities, building infrastructure or using a SIM card to send an SMS text message. As we were unsure whether a Lora network would be available

for implementation at that stage I implemented an SMS text message function to notify the researches if a trap has been activated.

After certain variables from different sensors have been confirmed, the Arduino will send an SMS text message directly to researchers phones to notify them of a trap activation. Each Arduino will identify itself so that the users know which trap has been activated. I also ensured that the location for each Arduino has a unique identifier so that researchers know exactly which trap has been activated.

3.5 Limitations and Considerations

This section will briefly mention the possible difference in results compared to the results that may occur in Borneo. It will briefly talk about a few ways to tackle these differences and what was done to ensure that conclusively that when an SMS text message is sent there is a very high chance a trap has been activated with an animal inside.

3.5.1 Environment and Accuracy

Firstly due to the tests not being conducted in a rain-forest, for the sound sensor results would have been the most difficult to maintain accuracy. As the rain-forest will naturally have a different sound setting the average sound produced in a rain-forest will be different to the woodland in which the tests will be performed. This was taken into consideration and as mentioned in the testing phase I tried to create a setting in which rain-forest sounds were played to try and assess the difference and analyse them in results.

Secondly the vibration sensor would also have different results based off the different types of animals that could be captured. Although I conducted a range of tests to analyse vibration results different animals will respond differently to being trapped in a cage. Some animals will be a lot more frantic and cause a larger commotion creating larger vibrations other animals may be small and light therefore not being able to physically produce larger vibrations.

One final note is that any shortcomings will be assessed and modified during live testing in Borneo to create more validated variables to detect trap activation.

Chapter 4

Implementation of Solution

In this section I will solely talk about the implementation of the solution and not any of the testing methods or outcomes. This section will show the code being used in regards to each sensor, how the Arduino and its components fit together and lastly how the SMS text message function was implemented. All of the conditional variables and minor tweaking of code will be implemented and mentioned in the Outcomes and Evaluation section after the tests were been completed.

4.0.1 The Arduino and Components

Firstly I will list the physical components being used along with the Arduino itself:

- Arduino Uno
- Grove Baseshield
- GPRS Shield V2.0

Along with these main components we have the sensors attached which have all been mentioned in the Method of Approach section and the casing. In the figure below we can see what the physical device looks like with all the sensors attached along with the main boards (4.1).

The last component to mention is the SIM card which was fitted under the GPRS shield. To enable the GPRS shield to work there is a separate switch on the side of the board that must be pressed to enable the SIM card to work (4.2).

The components were easy to assemble thanks to the Grove system, it was all slotted together using connectors and each board was placed on top of the other. At this stage there was no programming required.

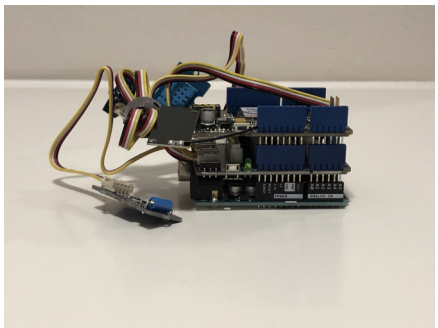

Figure 4.1: Arduino

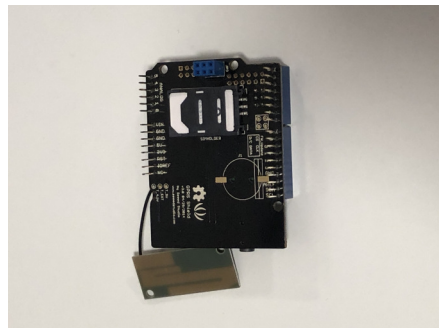

Figure 4.2: SIM Card

4.0.2 Coding the Sensors

All programming was done using the Arduino IDE. As the sensors were created by Seeed a lot of the functionality necessary had already been implemented. Therefore, I had used the base code for all the sensors from the Seeed website [16]. For each sensor I will briefly show the code and output, and provide explanations, as a lot of the code is self explanatory.

The first sensor I implemented was the motion sensor (3.4). For each sensor, initially you need to define which connector it is connected to. Then in the setup area, set the program to a specific Baud Rate so you can monitor the outputs. Also in the setup area, set the pinmode for the sensor. Then the loop function is where the work is done. For this sensor, it was really simple using the digital read function, if it detects movement then print "MOTION DETECTED" else print "MONITORING". This made assessing the output results very easy for testing.

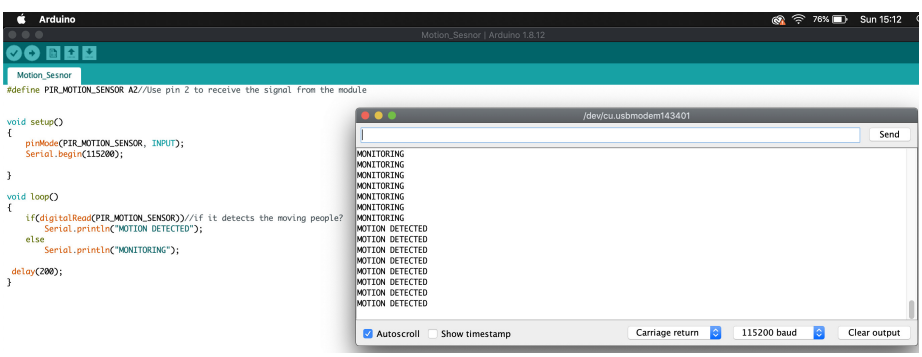

Figure 4.3: Motion Sensor Code & Output

The second sensor I implemented was the Temperature and Humidity Sensor (3.8). The Seeed website provided a library for the sensor called DHT. Using

this library and the Seeed website it was very easily implemented. As you can see the output for the results are neatly laid out and easy to interpret. When a researcher is notified of trap activation the most recent temperature & humidity results would be sent in the text message.

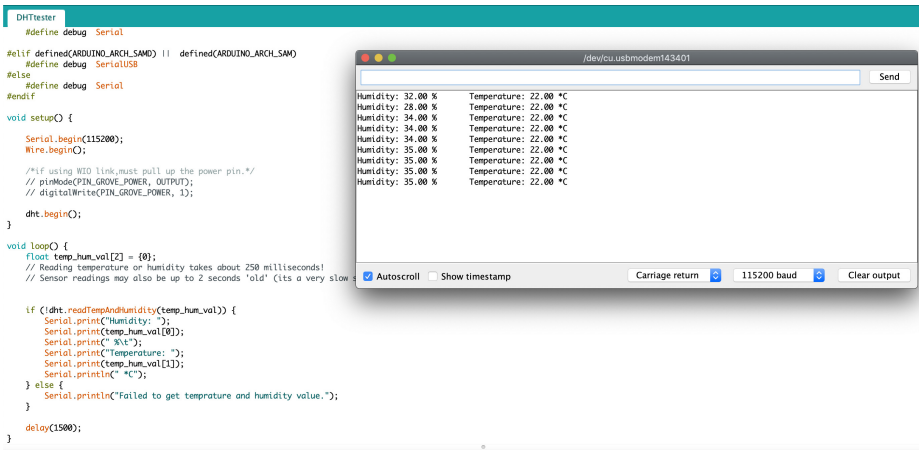

Figure 4.4: Temperature & Humidity Sensor Code & Output

The third sensor was the sound sensor (3.5). The implementation for this sensor was also found on the Seeed website and the results can be shown in a wavelength graphical view. When testing the sound sensor I took all of the different values and also re-created graphical representations to show my results. I also showed the highest, lowest and average values.

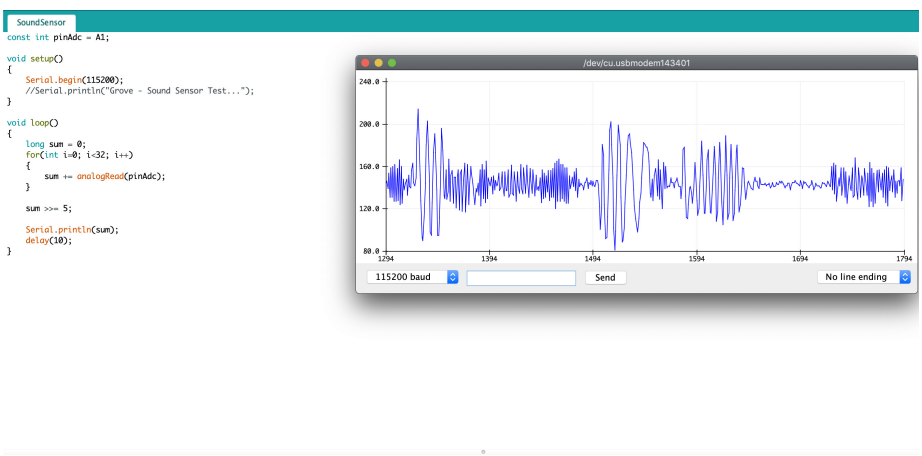

Figure 4.5: Sound Sensor Code & Output

The fourth sensor was the accelerometer (3.7). We can see that the code here

for the sensor was very complex and an external library was used, this had already been implemented for external use. As for the output the values, they were presented nicely. After doing the drop tests I monitored the Z values of the accelerometer, to determine the values which meant the trap door was fully closed.

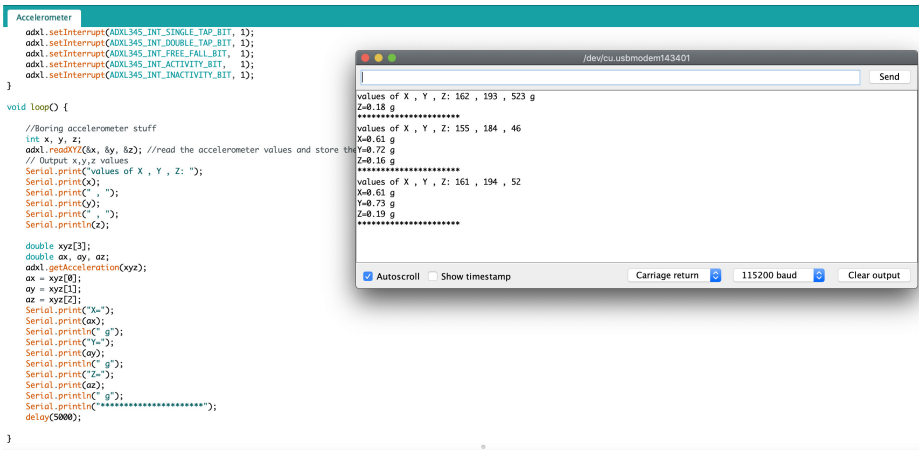

Figure 4.6: Accelerometer Code & Output

The final sensor I implemented was the vibration sensor, as this was the last sensor to arrive due to delays in shipping (3.6). This sensor was similar to the motion sensor as the output was one of two values, either the vibration was high or low. This made it easy during testing to analyse the results and reach conclusions.

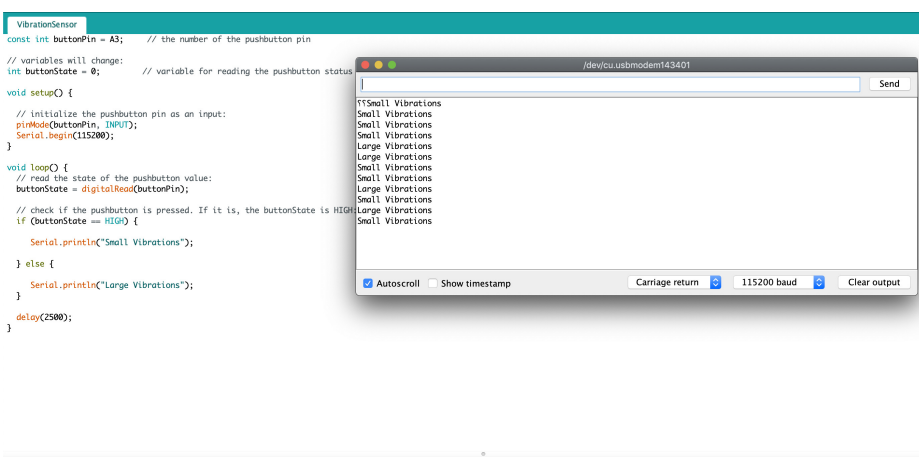

Figure 4.7: Vibration Sensor Code & Output

After implementing all of the sensors, small initial tests were done to ensure all the sensors worked correctly, to make sure when the official testing began none of the sensors were producing the wrong results.

4.0.3 Coding the SMS Function

As already mentioned, before trying to use the SMS function we had to establish there was a activated SIM card within the GPRS Shield and the GPRS Shield itself had been switched on and there was a cellular connection.

There was initially a dial function within the code that I removed as it would never be needed. With the initial testing code it was required that you parse the character 't' for a text message to be sent to a static number of your choice. When the SMS board originally boots up there is a green LED which blinks every 3 seconds if you have successfully connected to a network. After the text message has been successfully sent the board will respond with 'OK'. I have covered the number in the image for personal data reasons although the country code must be included in the number for it to work. As you can see in the figures below a text message was successfully sent.

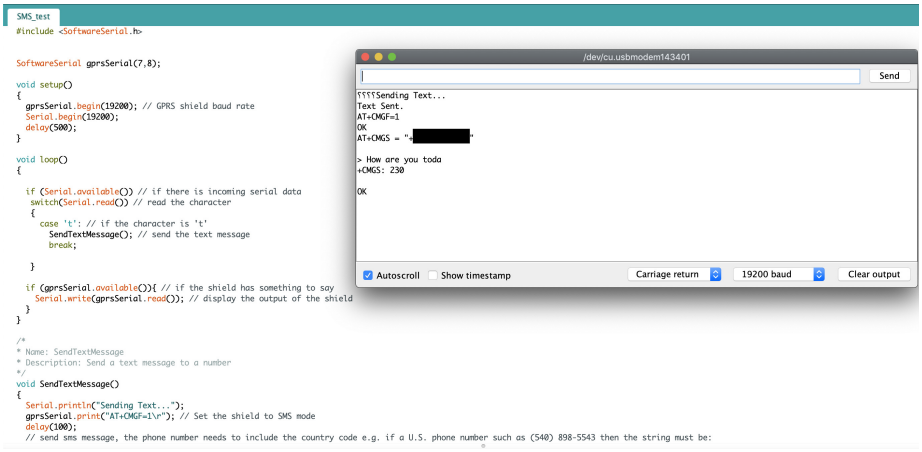

Figure 4.8: SMS Text Code & Output

4.0.4 Combining the Code

After all the initial implementation and tests had been finished I had to collaborate all the code into one file. I also reviewed the text and made any minor changes that were needed. The last functionality I changed before doing the tests on the sensors was how the text message was sent. I did one initial test that when high vibrations were detected a text message would be sent out, which was successful.

Figure 4.9: Text Message Success

As we can see in the screenshot below all of the sensors are working together in one file and all the outputs are being printed (4.10).

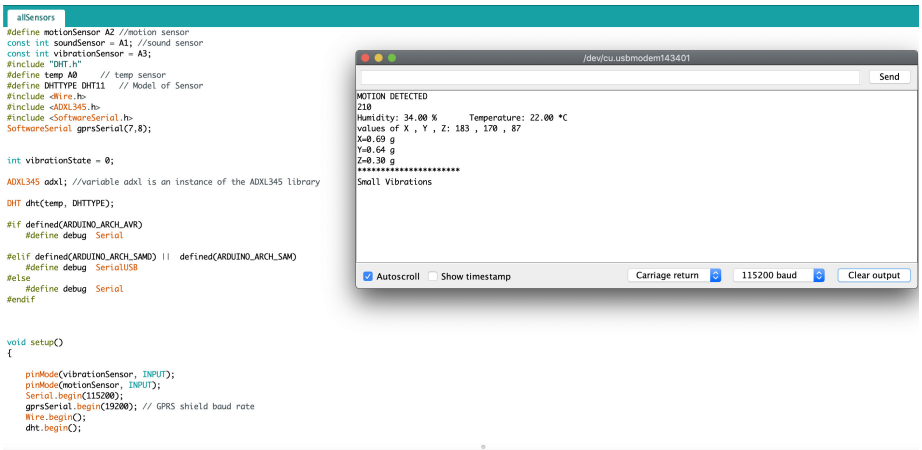

Figure 4.10: All Sensors and Outputs

Chapter 5

Outcomes and Evaluation

In this section we will examine the results of each sensor and try to understand the accuracy of each sensor and whether it can be used to detect and confirm that a trap has been activated. After all results have been evaluated this section ends with how the program was adjusted to send a text message when the conditions show that a trap door has been closed. Not only will the results be useful for understanding whether the sensors could be used to detect trap activation it could also show how accurate the sensors are to be used for other purposes in the rain-forest.

5.1 Test Results

5.1.1 Motion Sensor

As mentioned in the method of approach I would test the motion sensor inside and outside of the case to see if there were any changes in the accuracy of the results.

Firstly we will talk about the motion sensors capability of detecting motion when outside of the case. I created a small testing environment to see if motion was detected for up to 5 meters. The reason for capping the distance at 5 meters is because the motion sensors default detection range was 3 meters, and this would be used for detecting motion within the trap and no trap exceeded 5 meters in length. I did not leave the testing range at 3 meters as I wanted to see if it would detect motion further out as this would mean that motion could be detected outside of a trap, which could contribute to false alarms of when a trap had been activated.

As seen in the table below (5.1) we tested the motion sensor 10 times at 5 different ranges. As a result of the test, every time I walked into the 120°C

detecting angle of the motion sensor at all ranges motion was detected. The sensor was very accurate and at no point gave off false motion detection alerts.

	Result									
Meters	1	2	3	4	5	6	7	8	9	10
1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Figure 5.1: Motion Sensor Test Results

I then tried to conduct the test with the motion sensor within the case. This lead me to completely different results, as even though the case was plastic, transparent and only 2.5mm thick, the motion sensor could not penetrate through the casing and therefore did not read any results outside of the case. Alternatively, disturbing the case by either touching or moving it did detect motion. Therefore, this could also be used on the trap door, as when the door closes and the case moves and the motion sensor would be triggered.

Lastly, after testing the range for up to 5 meters, I decided to quickly see whether the motion sensor would be able to detect motion past 5 meters in length. To my surprise, even though the default range was 3 meters, it continued to detect motion up to 10 meters consistently. After 10 meters the accuracy decreased and it could not always detect motion at that range.

In conclusion, the motion sensor will not be able to detect motion of an animal within the trap if it is within the case, but if a perfect hole was cut for the motion sensor to sit in which would allow for it to not be restricted by the case it could be used to detect motion within the trap. Another point to note would be to try another case with either a different material or thinner casing as this could provide different results.

5.1.2 Sound Sensor

The first test that was conducted on the sound sensor was recording the ambient sound outside of the case in a quiet woodland area. Th helped us understand the normal values of the sensor when there are no extra variables to see what values could be considered abnormal. The way each test was conducted was recording the values from the sound sensor for 5 minutes, which gave us 300 values, and we conducted each test 3 times. Another reason to conduct this test was to see the difference of the ambient inside and outside of the case and compare and contrast them. Therefore, if a different case was bought for the sound sensor, depending on its build, it could give an idea to the user on how the sound inside would be different.

As we can see from the results in the graph (5.2) although we tried to keep

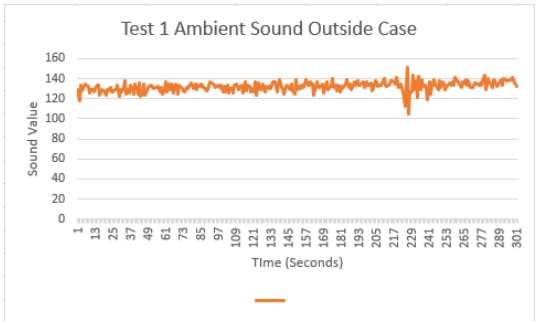

Figure 5.2: Sound Sensor Test Result (1)

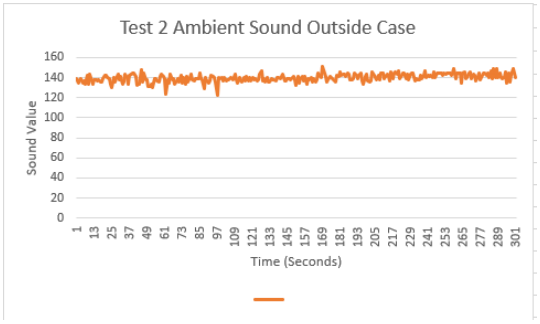

Figure 5.3: Sound Sensor Test Result (2)

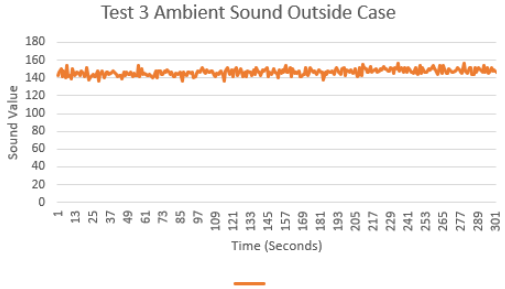

Figure 5.4: Sound Sensor Test Result (3)

the environment as controlled possible, at around 229 seconds there were some spikes in the results that came from seagulls that passed over the area. The average value of this sound test was 132 and the majority of the values stayed between 120-140.

In the second test (5.3) the results did not have any major spikes from the surrounding environment, but may have picked some sounds in the distance,

such as people or car horns. On average, for this test the sound was 139, which was higher than the previous test. We can see that the values never dropped below 120, but were a lot closer to 140 from the previous test, which would have been the cause of the increased average.

In the final test, (5.4) under this environment, the average value continued to increase and was 146. This was the most stable graph and had the least interruptions from the surrounding environment. Again, the values for this graph stayed between 136 and 156. From the first set of results it is possible that the sensor was still adjusting to the environment, as the average value kept increasing. Although, as this was not visually evident, I did not carry out anymore tests at this point in time.

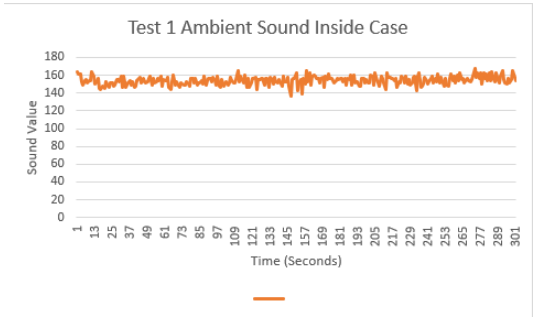

Figure 5.5: Sound Sensor Test Result (4)

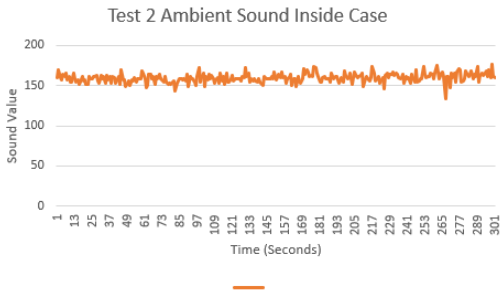

Figure 5.6: Sound Sensor Test Result (5)

The second set of tests conducted were of the sound sensor inside the case. We can immediately see that the ambient sound within the case was higher than outside of the case. This could be because the sensor is not in an open environment and therefore, due to its confined nature, the average values increased due to its limited surroundings.

In the first test, (5.5) we can see that the readings are a lot more jagged than when conducting the test outside of the case. Whilst the sound sensor was

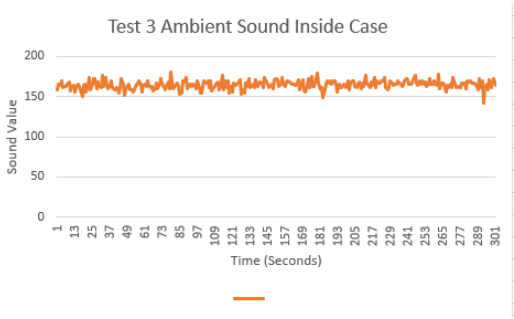

Figure 5.7: Sound Sensor Test Result (6)

outside the case, it normally stayed within 20 values.Whereas inside the case, the sound sensor exceeded that limit and the values in the first test are anywhere from 137 and 168 which is a 30 value difference. The average reading for this test was 154 which is a higher average than all of the tests conducted outside of the case.

The next two tests (5.6)(5.7) followed the same pattern as the tests conducted whilst the sound sensor was outside the case. The values continued to increase over these tests and the average for the second test was 159, whilst the average for the final test was 164. Nothing had been changed when conducting the second and third test, yet similarly the values continued to increase. One possible explanation for this would be that the sound sensor had not fully adjusted to its environment and so the values continued to increase until it reached a baseline value. The only way to test this theory is to conduct more tests for a longer duration.

The later two tests were achieved in an environment which created sounds that mimicked the rain-forest in Borneo in a surround sound setting. This would help us understand what sort of sound shielding the case provided and the difference between the sound sensor inside and outside the case when extra variables were put in place. We used a video to mimic the sounds in a rain-forest and played the same part of the video for each test so that they would bring us accurate results and we could compare and contrast the spikes for each test.

Looking at the results (5.8) (5.9) (5.10) produced by the surround sound test outside of the case, there were four occasions in which the graphs spike. At this point during the testing video, there was some thunder which created a very deep bassy sound. Comparing to the tests that were done where the sound sensor was outside of the case with no surround sound, the averages were slightly different as it was picking up the background sound, but they did not change drastically and the values were very close to the old average values. The same trend of the average values increasing over the course of the 3 tests also happened when evaluating these results. The average values for these tests were 149 (5.8), 177

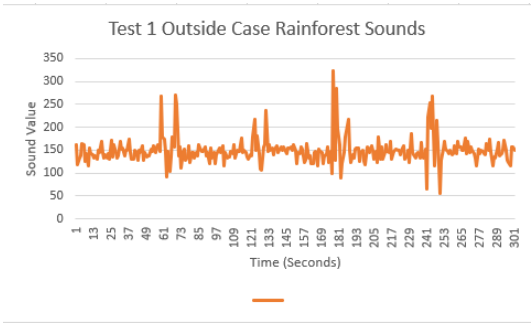

Figure 5.8: Sound Sensor Test Result (7)

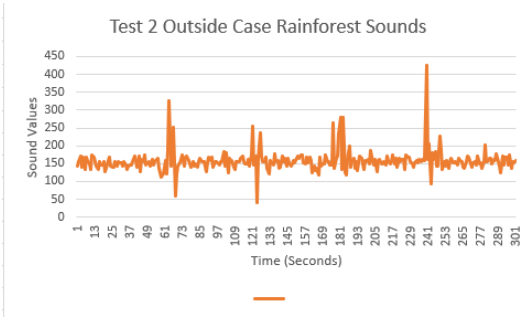

Figure 5.9: Sound Sensor Test Result (8)

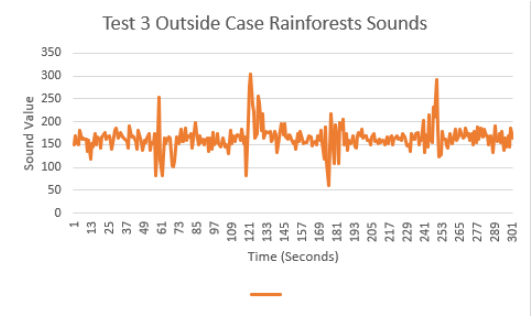

Figure 5.10: Sound Sensor Test Result (9)

(5.9) and 163 (5.10).

Lastly we look at the results generated from the sound test being tested with surround sound whilst the sound sensor was inside the case (5.11) (5.12) (5.13). Comparing these graphs with the previous tests done with the sound sensor inside the case without any surround sound (5.5) (5.6) (5.7), clearly they are very similar. Without the spikes they are almost identical and would have

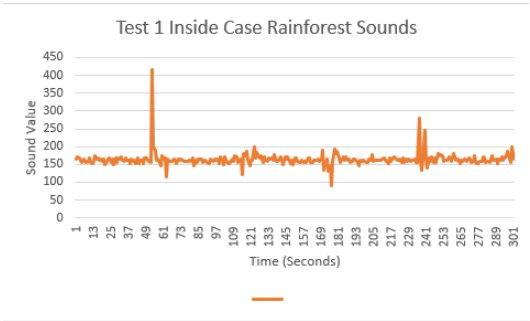

Figure 5.11: Sound Sensor Test Result (10)

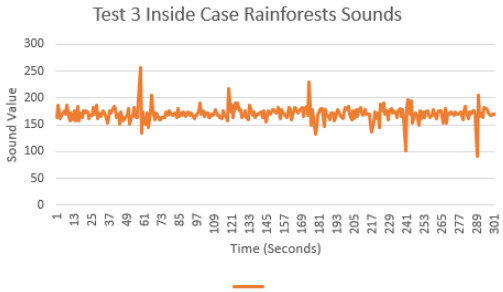

Figure 5.12: Sound Sensor Test Result (11)

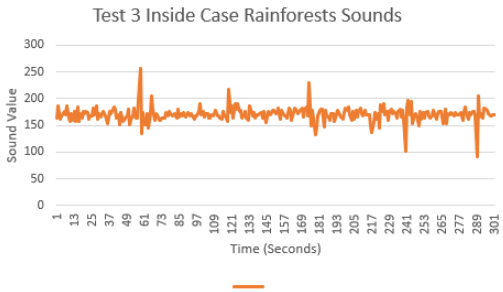

Figure 5.13: Sound Sensor Test Result (12)

the same average values. Due to the impact of the thunder happening in the video the spikes in the values increased the average value. We can also see that the casing does provide some sound insulation from the surround sound as the values were a lot more contained than when the sound sensor did not have a case protecting it.

To conclude, on the sound sensor, more tests would need to be done. For

every test completed in a different environment, the average value continued to increase and therefore, could not be used to detect trap activation, or if an animal is within a trap without further understanding of the correct values the sound sensor operates at. Using the sound sensor from these tests could send false alarms of trap activation. We have learnt that the casing does have a different ambient sound compared to an open space and due to this, similarly to the motion sensor, it would be better to create a bespoke case which enables for the sensor to poke outside the case and monitor the sound values from outside. Therefore, if we used the sound sensor in the trap and the sound level had increased or spiked, we could assume that an animal such as a monkey could be trapped inside, but an animal such as a civet would cause less commotion and noise and therefore not affect the values as much.

5.1.3 Vibration Sensor

All the vibration tests were conducted whilst the sensor was in the case. There were 3 different tests done to see how effective the vibration sensor was at picking up vibration, light tapping, moderate tapping and knocking the case. The light tapping would mimic if the trap was being shaken and therefore the case would also be slightly shaken, possibly triggering the vibration sensor. Moderate tapping would be a possible larger animal moving around inside the trap which could cause the trap to move around more, therefore, disturbing the casing more and creating moderate vibrations to trigger the sensor. Lastly the knocking of the case would be as if an animal was trapped inside and had accidentally directly hit the casing with enough force for it to move. The vibration sensor did not output integer values, similarly to the motion sensor, it only detected "Small Vibrations" and "Large Vibrations". Each test was conducted 20 times and the results can be seen in the table below (5.14).

	Results																			
Test	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Light Tap	✓	✗	✓	✗	✗	✓	✗	✗	✓	✓	✗	✗	✓	✓	✓	✗	✗	✗	✗	✓
Moderate Tap	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✓	✓	✓	✓	✗	✓	✓	✓	✗	✗
Knocking Case	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Figure 5.14: Vibration Sensor Test Results

As we can see from the results, the light tapping of the case had a 45% chance of creating vibrations large enough to be picked up by the sensor. This means that if a small lightweight animal was trapped within the case, such as a Malayan Civet then it is very possible that unless it directly hit the case, the amount of commotion it would case in the trap would not be significant enough for it to trigger the sensor.

The moderate tap increased the chance of triggering the sensor to 70%, which is considerably higher. This means, that if an animal such as a Clouded Leopard was within a trap, it could cause enough commotion to trigger the vibration sensor, which would be promising that an animal has been captured.

Lastly, the knocking of the case triggered the sensor 100% of the time, meaning that if any trapped animal directly knocked the casing with enough force to move it, then it would trigger the sensor.

I also did some further tests such as walking around and creating vibrations around the case which were never picked up. This concluded that if the case was not directly affected or moved, vibrations would not be picked up from its surroundings. This is very important, as if an animal walked around the case but without enough force to make the trap move it, would not trigger the sensor.

For this sensor to be effective in monitoring whether an animal has been trapped, it must be placed within the trap at close proximity to the animal. Without the casing being moderately disturbed, it will not be able to contribute any evidence that an animal is trapped.

5.1.4 Accelerometer Sensor

All the accelerometer tests were done whilst the accelerometer was inside the case. The way the accelerometer was tested was a controlled drop from 3 different heights 40cm, 60cm and 80cm. The reason for these heights were because they are the same distance that different trap doors would fall to fully close. I attached the case to a pole to drop the case straight down at free fall speed. The reason for a controlled drop was for more accurate results, as when a trap door would shut, the case would stay in the same position straight down. Allowing the case to fall with no control could mean it changes direction and this could affect the accelerometer results. At the same time as measuring the accelerometer values, I also thought it would be important to note the sound, vibration and motion sensor values, as using these multiple variables would give more confidence that a trap door has shut. Each controlled drop at each height carried out 20 times.

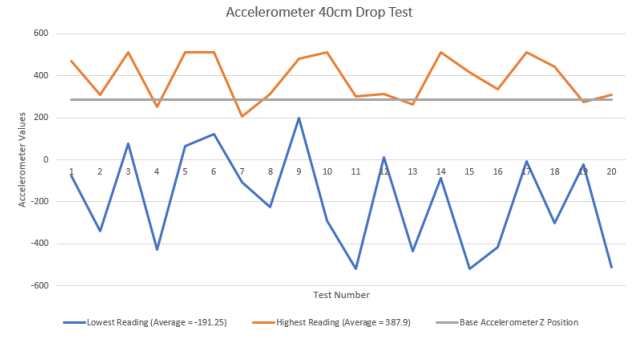

Figure 5.15: Accelerometer 40cm Drop Test Results

As seen in the results, we recorded the highest and lowest offset values produced when the drop test was conducted. The average position was also recorded

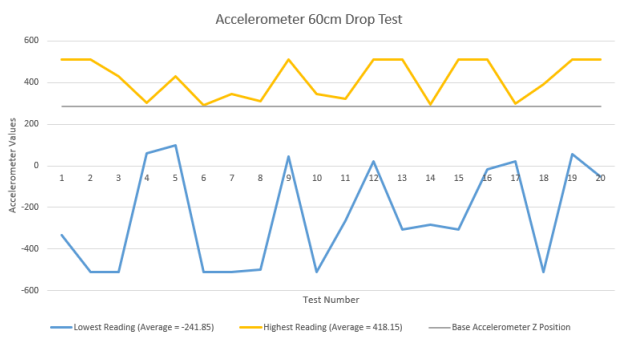

Figure 5.16: Accelerometer 60cm Drop Test Results

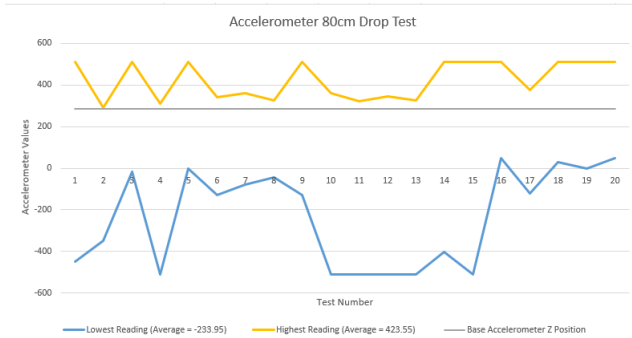

Figure 5.17: Accelerometer 80cm Drop Test Results

prior to doing the drop tests. As the orientation of the accelerometer was never changed, this base value stayed the same throughout the entire testing period and was monitored to ensure that it did not change. The base value is represented by the grey line on the graph and the value is 285.

We can see in the 40cm drop test (5.15), that the highest and lowest average readings were lower than the 60cm and 80cm drop test. On average, the displacement from the base value was often around -476 followed by +103. On some occasions, the results were less drastic when the trap door had shut. We can see that the closest values to base value on a drop test were 197 and 206, meaning that the difference was -88 and -79. When reading the results, the output was every 0.15 seconds, meaning some values could have been lost within these milliseconds and therefore not recorded. Therefore, the result where the highest offset value being 206 which was -79 from base value, and the lowest offset value being 196, -89 from the base value are considered to be anomalies as the most accurate offsets were lost. As the program would be recording every millisecond of data, I believe that we can discount these offsets for more accurate representations in the real world scenario. Looking at these values, if

we were to place the accelerometer on a trap door which travelled 40cm to fully close, I changed the program so that if the Z value dropped by 150 or exceeded 70 from its base value, this would be significant evidence that the trap door has closed. The reason for this is if we discount the anomalies the maximum lowest offset was 122 and the minimum highest offset was 253.

The results of the 60cm drop test (5.16) differ significantly from the 40cm drop test. This could be because there was more distance and speed in which the accelerometer fell and therefore, the accelerometer was more displaced creating different values. The offsets of these tests stayed higher in comparison to the 40cm drop test. The closest values to the base value here were 97 for the lowest offset and 293 for the highest offset. The average value for the lowest offset was -241 which is -526 away from base value and the average value for the highest offset was 418 which is +133 away from base value. Comparing to the 40cm drop test, it shows that the average for the lowest offset was lower and the average for the highest offset was higher. We can see that the offsets were on average further away from the base value compared to the 40cm drop test. After looking at the results produced by this drop test, every time the case was dropped the lowest offset fell below 97 as a maximum value for the lowest offset. The highest offset was a lot closer to the base value on a lot of the tests. As mentioned, this could be due to the way the results were produced and printed, and there could be a larger offset that was not captured in the data. If we were to place a condition in which if the lowest offset was below 100, I believe this could be a single condition that shows that a trap has been activated if the door had to travel 60cm to fully close.

Lastly the 80cm drop test (5.17) was very similar to the 60cm drop test. The values are slightly different by having an overall average lower reading at -233 and higher average for the highest offset which was 423. The closest highest offset value was the same as the 60cm drop test, which again was 293, but this time the closest lowest offset value was 48, which was again considerably lower than the 60cm drop test value. After looking at the three sets of results and the graphs, we can see a trend that as the drop test height was increased the lowest offset value got lower. Although the average lowest offset for the 60cm drop test is lower than the 80cm drop test, the maximum value from the lowest offset was 48, which was considerably lower than 97. Although, based on these conditions, there is still not much difference between the 60cm and 80cm drop test and therefore, I believe that having the same condition as the 60cm drop test, that if the maximum lowest offset was to fall below 100, this could be significant evidence that the trap door has closed.

One factor to note that could not be replicated was the weight of the trap door, this could change the values of the drop test.

Every time the drop test was conducted we also looked at the vibration, motion and sound values. Every time the case landed, the vibration picked up "Large Vibrations", motion was always detected and the average sound value spiked to 473 with base value of 167. These tests were conducted at the same as every

drop test, meaning the results were replicated 60 times.

5.2 Summarising Results & Changes to Arduino and Sensors

After creating and evaluating all the test results from each sensor, this showed how the sensors could be used to the best of their ability to detect a trap being activated. There are 5 sensors in total; vibration, sound, motion, temperature/humidity and the accelerometer.

After testing the vibration sensor and the accelerometer, and evaluating their outcomes, I concluded that they will be used with the Arduino that sits on the trap door. The reason for using the vibration sensor on the door, as shown from the results is whenever the trap door shuts the vibration sensor inside the case is disturbed and will sense that vibrations have been created. Alongside the vibration sensor the accelerometer is the perfect sensor to detect whether the trap door as shut. As when the door shuts, it senses the change in motion and the positional values change drastically, then measuring the offset values shows whether the trap has been activated. Using both of these sensors, when the vibration sensor is triggered and the accelerometer values change a text message will be sent informing the recipient that the trap has been activated. Having only two sensors on the trap door allows for the code to run more efficiently and take up less memory and also prolong battery life of the Arduino as it is only powering two sensors.

The second Arduino will be inside of the trap and will have the motion sensor, sound sensor and the temperature/humidity sensor attached. As the animal will be trapped inside the case, the environment inside the trap could be slightly different compared to outside the trap. Therefore, it is important to know the temperature and humidity within the trap so that if the conditions are extreme the researchers know that they need to release the animal as soon as possible. After running tests on both the motion sensor and sound sensor the results show that inside the case the values are affected by the casing. Following these results, the ideal scenario would be to create perfectly sized holes to allow the sensors to poke out of the casing to get the most accurate readings. The motion sensor inside the trap could detect any movement confirming that an animal is trapped inside the case. If there happened to be some motion detected and an animal was not trapped, but some debris had rolled inside the trap and triggered the sensor, the sound sensor would be the supporting factor that if abnormal sound values were created, this could support the evidence that an animal is trapped inside. Again minimising the amount of sensors attached to the Arduino and less code, increases the efficiency in the program and prolongs battery life for the Arduino.

5.3 Re-configuring the Code & Creating Triggers

After doing all the tests, I cleaned up the code and created the programs that would be used in both scenarios and tested them multiple times to make sure they worked. Both of these values are temporary and would be changed when monitoring a trap in the real world. After creating the triggers I also created a code snippet which would send a text message every few hours to show that the Arduino was still monitoring the trap and nothing had happened.

Firstly, we can look at the program that would be used for the Arduino sitting on the trap door. This contained the vibration sensor and accelerometer. The setup function contains all that is needed to initialise the vibration sensor, accelerometer and GPRS shield. The loop function calls the vibration function and accelerometer constantly monitoring the values. When the vibration sensor and accelerometer met their conditions a text message was sent, saying that the trap had been activated.

```
void loop() {
  // put your main code here, to run repeatedly:
    const int vibration = VibrationSensor();
    const int accel1 = AccelerometerSensor();

    if(vibration == 1 && (accel1 > 400) || accel1 < 0){

          SendTextMessage();
    }


}
```

Figure 5.18: Code for Arduino Monitoring Trap Door

Secondly, we have the program which is used to detect if an animal is inside the trap to give further confirmation a trap has been activated. As in the first program, all variables are declared before hand, the setup function is used to initialise the motion sensor, sound sensor and temperature/humidity sensor. In the loop the sound sensor and motion sensor values are constantly being monitored. If the conditions are met, then a text message is sent with the added data of the current temperature and humidity within the trap.

```
void loop() {
  // put your main code here, to run repeatedly:
    const int motion = MotionSensor();
    const int sound = SoundSensor();


    if(motion == 1 && (sound > 400 || sound < 100)){

        const float humidity = ReadHumidity();
        const float temp = ReadTemp();

        SendTextMessage(temp, humidity);
        Serial.println("D");

      }

}
```

Figure 5.19: Code for Arduino Monitoring Inside the Trap

Chapter 6

Conclusions

Firstly, we understood the problems and difficulties that have been taken place in Borneo. Due to deforestation and loss of the rain-forest, animals that once thrived in Borneo have slowly been diminishing in population and becoming endangered. We did some background research into the problem at hand and understood what the researchers and scientists at the Danau Girang Field Centre (DGFC) could use to help them save these endangered species. They had been using traps to capture animals to monitor their well being and understand their needs so that they can be taken care of. The researchers needed a way of being notified when a trap had been activated so that animals captured would not be put at risk. In the current situation traps were monitored and checked manually on a rotary basis which was not very efficient and could leave animals trapped without food and water for long periods of time. We looked at the traps that were being used and understood how we could effectively monitor traps and notify researchers when an animal had been trapped inside. After running multiple tests on the various sensors chosen, the results show that using an Arduino and these sensors, we would be able to provide researchers with a notification and a high probability that a trap has been activated. We concluded that having two different placements of the Arduinos would be able to ensure multiple points of data to support when a trap has been activated. Using the test results, we programmed the Arduino to send a text message when certain conditions were met that would notify a researcher when a trap has been activated.

Chapter 7

Future Work

Although the project has been completed, there is much to be done to help the endangered animals in Borneo and improve this notification system. The tests done were not in the same environmental conditions as in the real world scenario and could differ when implemented in Borneo. This project has been successful and could be used to detect trap activation although more tests could be done to ensure accuracy. Unfortunately due to Covid-19 and the current pandemic I will not be able to travel to Borneo to test this application first hand and tailor it to the environment. In the future this would be essential as doing tests on the traps used in Borneo could slightly differ and if researchers were to rely on these notifications then they would need to be tested thoroughly for the safety of the animals.

The current application uses SMS text messages to notify the researchers and has a very simple system in place due to the lack of infrastructure around the Danau Girang Field Centre (DGFC). If the infrastructure was to improve, a different way for the devices to communicate could be a much more reliable way of receiving notifications. For example, if a LoRa network was introduced then we could create a cloud application which handles all the data and monitors all the Arduinos using a single interface. One person could effectively ensure that they are monitoring the application and whenever a trap is triggered a notification could appear on the application pinpointing exactly which trap was activated and the location. Furthermore, it could send constant data and updates, and if a trap was not triggered but the results or values of the sensors looked abnormal this could prompt a researcher to check the trap just in case.

Chapter 8

Reflection on Learning

During this project I have learnt a lot and improved numerous skills. Firstly, I had never worked with an Arduino or coded in C. I thought these two factors would be my biggest challenges, but overall the learning curve was really smooth. I spent the first few weeks really understanding how the Arduino and sensors would work together and how I could use them to solve the problem of detecting trap activation. There was a lot of online support and having a background in other IoT related work helped my understand how I could use this device to its full potential and achieve a highly accurate product. When unsure I would ask for feedback from my supervisor which gave me guidance on which sensors to use, as well as creating a well thought out time plan and method of approach. This gave me a clear structure to my project which I worked with accordingly to finish the project on time and meet the product expectations. The documentation for all the sensors was very clear and allowed me to fully understand how each sensor could be used.

A few things that had impacted my learning and the project to be more accurate was the prevention of more testing. Due to the current situation of Covid-19 and the pandemic I was not able to test the sound sensor more thoroughly outdoors. The areas which I used to do the testing were off limits and this was a non-essential reason to go outside to other areas in which the tests had been conducted. I would have also liked to have tested more sensors such as a magnetic sensor further down the line, but due to shipments being delayed from China this would have impacted my time plan and delayed the project even more, as at the start of the testing period my time plan was delayed by two storms in the UK which were not ideal conditions to test some sensors outdoors. The last change I would like to have made and included in this project was using different types of casing to compare the results, but this would have been an expense of cost which was not funded.

Overall my learning experience throughout the project has been positive. I have

been able to complete an entire project successfully myself with an end product. I managed to stick to my initial time plan and conduct all of the tests necessary on my sensors which gave me adequate results to create an application that would notify researchers with a high probability of a trap being activated.

References

[1] Carlos Cambra ; Sandra Sendra ; Laura Garcia ... "Low cost wireless sensor network for rodents detection". In: (2017).

[2] Chakchai So-In ; Comdet Phaudphut ; Smarn Tesana ... "Mobile animal tracking systems using light sensor for efficient power and cost saving motion detection". In: (2012).

[3] L. Bhavani Annapurna; Kalicheti Mounika; K. Chakradhara Chary ... "Smart Security System using Arduino and Wireless Communication". In: (2015).

[4] United States Department of Agriculture Animal and Plant Health Inspection Service. "Evaluation of Remote Trap Monitors". In: (2007).

[5] United States Department of Agriculture Animal and Plant Health Inspection Service. "United States Patent No. US 9,439.412 B2". In: (2013).

[6] Yanping Wang ; Zongtao Chi. "System of Wireless Temperature and Humidity Monitoring Based on Arduino Uno Platform". In: (2016).

[7] BARBARA FRASER. *Is deforestation in Borneo slowing down*. URL: https://forestsnews.cifor.org/59378/has-borneos-deforestation-slowed-down?fnl=en.

[8] Benoît Goossens. *Danau Girang Field Centre*. URL: http://www.dgfc.life/home/dgfc/.

[9] Karan Juj. "Tracking poachers using The Internet of Things [BLE]". In: (2019).

[10] Emily Gertz; Patrick Di Justo. "Environmental Monitoring with Arduino". In: (2012).

[11] Prakash Kumar; Pradeep Kumar. "Arduino Based Wireless Intrusion Detection Using IR Sensor and GSM". In: (2013).

[12] Sheikh Ferdoush; Xinrong Li. "Wireless Sensor Network System Design using Raspberry Pi and Arduino for Environmental Monitoring Applications". In: (2019).

[13] Donna Marchetti. "Borneo's Wild Side". In: 10 (1998).

[14] Bernhard Mehl. *6 Communication Protocols Used by IoT*. URL: https://www.getkisi.com/blog/internet-of-things-communication-protocols.

[15] seeed. *Grove System*. URL: http://wiki.seeedstudio.com/Grove_System/.

[16] seeed. *Sensors*. URL: http://wiki.seeedstudio.com/Sensor/.
[17] Akshay D. Deshmukh ; Ulhas B. Shinde. "A low cost environment monitoring system using raspberry Pi and arduino with Zigbee". In: (2016).
[18] McBee Moore Vanik. *SENSOR FOR A WIRELESS ANIMAL TRAP DETECTION SYSTEM*. URL: http://www.freepatentsonline.com/y2019/0037829.html.
[19] Siwakorn Jindarat ; Pongpisitt Wuttidittachotti. "Smart farm monitoring using Raspberry Pi and Arduino". In: (2015).
[20] WWF. *Found only in Borneo*. URL: https://wwf.panda.org/knowledge_hub/where_we_work/borneo_forests/about_borneo_forests/borneo_animals/.