ASEE 2022 ANNUAL CONFERENCE
Excellence Through Diversity
MINNEAPOLIS, MINNESOTA, JUNE 26TH-29TH, 2022 ASEE

Paper ID #37974

A cost effective smart trough monitoring alert system

Emil H Salib (Professor)

A cost-effective smart-trough monitoring alert system

Dr. Emil H. Salib, Kabeer Mehdi and Matthew Chamberlain
salibeh@jmu.edu, mehdika@dukes.jmu.edu and chambemr@dukes.jmu.edu
College of Science and Engineering (CISE),
James Madison University (JMU), Harrisonburg, VA 22807

Abstract

As farmers embrace environmental stewardship, many plan to improve surface water quality by excluding cattle and establishing riparian buffers along streams and rivers. This requires establishment of mechanical watering systems for the cattle. In this senior capstone project, we developed a monitoring alert system that notifies the farmer in the event of a watering system failure. We recognized from the outset of the project the importance and high value of having to actively and professionally interact with our client. We have also realized the significant opportunity and risk we were presented with when the client offered us his farm as a platform to exercise our ideas and test our system prototypes.

To meet our client's requirements, two senior students, under the guidance of their senior capstone project advisor, designed a system that included the integrating of three (3) different wireless technologies (Cellular, Wi-Fi and Zigbee), the integrating data from multiple sources through a wireless gateway, and powering the system computing components (e.g., Arduino and wireless Xbee) using a solar panel at each trough. The water level at the troughs, which are located in the pasture field, are wirelessly collected and sent to a Raspberry Pi (RPi is a small mobile computer) in a designated water pump house. Additional data, water pressure, and ambient temperature at the pump house are also collected in the pump house. The RPi unit aggregates all of the data from all locations and stores it for further processing. If any measurement falls outside of a preset range, an text notification message is sent to the farmer's cell phone alerting him to the type of malfunction.

Our phased, incremental, and keep it simple approach in tackling the implementation and testing of the system design paid off handsomely when at the end, we were able to successfully test and evaluate the system in the field (that is, on our sponsor's farm). Our assessment of the system was very impressive. In it, we included detailed cost of the system, potential future cost savings, a comparison of our system against potential competitors, and a thoughtful list of what should be done next to commercialize the monitoring alert system.

1 Introduction and Background

Today, society is faced with a wide range of environmental problems that are continuing to grow in severity. If left unchecked, these problems can lead to long-term damage to not only the environment but also the living things in the environment. To tackle these problems, it is the responsibility of humans to develop a plan to reduce the severity of environmental problems. One such solution is that the United States Department of Agriculture in collaboration with local authorities has developed programs [1] that encourage farmers to cut off their lands from adjacent streams and waterways. By doing this, runoff from these farms does not fall into the waterway and contaminate watersheds. In return for fencing off their lands, the Department of Agriculture (through local authorities) will provide financial assistance to these farmers to help them return to normal operating functions at their farm.

One of our department faculty member owns a cattle farm in Madison County, Virginia and fenced off his land from a natural stream that feeds into the Rappahannock River watershed. Unfortunately, the streams adjacent to the farm are the main source of drinking water for the cattle. Without access to the water, the cattle would suffer from dehydration and that would affect the health of the cattle when calves would be born. To solve this problem, the farm owner had a mechanical system built on the farm where water troughs were installed at different pastures on the property. These water troughs would be connected to a pump house that would mechanically refill the water.

Currently, while the mechanical system can refill the water troughs, there are some issues. First, the amount of water can only be measured by visual inspection, that is, by going to the locations of each trough and seeing the water level of the trough. Second, the mechanical system is prone to failures that lead to the pump house losing pressure. The loss in pressure can also be caused by low temperatures, causing the pipes in the pump house to freeze. The farm owner reached out to two senior students and their senior capstone advisor to take on the opportunity of developing a solution to these issues. Under the sponsorship of the farm owner, the senior students, with the help of their advisor, developed a monitoring alert system. This system would be able to measure characteristics of water such as the water level at the water troughs as well as the pressure and temperature at the pump house feeding the water troughs. These characteristics would be measured and if they cross a predefined threshold value, an alert message would be sent in the form of a Short Message Service (SMS) text message to the farmer's or the farm manager's mobile device.

The paper is organized as follows. In section 2, we provide a brief overview of the opportunity definition and the solution goal and objectives of the senior capstone project described in this paper. In Section 3, we provide a brief description of the solution design and prototype. Section 4 focuses on the solution testing and pilot study stages and section 5 details the solution results. In section 6, we offer a project discussion, our conclusions, and potential next steps.

2 Definition and Objectives

In this section, we provide a brief definition of the senior capstone project opportunity. In addition, we describe the proposed solution and its goal and objectives.

2.1 Opportunity Definition

Our senior capstone project sponsor, the cattle farm owner, has water troughs located on various pastures. Each water trough is connected to a pump house that mechanically refills the water troughs at each pasture. Currently, there is no efficient method to monitor the characteristics of the water troughs except for physically going to each water trough and checking the water level or using an expensive camera to take pictures and send them to the owner's or the manager's mobile device. The farm owner wishes for a system that can send an alert to the owner's or the farm manager's mobile regarding the water level in the water troughs as well as the water pressure and temperature within the pump house.

2.2 Proposed Solution

To solve this problem, our senior capstone project proposed solution was to create a monitoring alert system. Sensors would be placed at the water trough and the pump house. The sensors at the water troughs would monitor the water level and those at the pump house would monitor the water pressure and air temperature. The information being monitored at the water troughs and the pump house would be collected and analyzed to send an alert message (SMS text message) in case the farmer's attention is required. By being able to receive information from the watering system remotely, a farmer or their farm hands can determine earlier which troughs they need to refill and therefore can spend the rest of the day taking care of other important tasks.

2.3 Solution Goal and Objectives

In collaboration with our sponsor, we set out a series of goals that the senior capstone project must achieve. These goals are as follows:

- Prepare the farm for a possible Environmental Easement. With the farm being cut off from natural waterways and streams, the cattle on the farm have no access to natural sources of drinking water. Our system needs to be able to assist the owner in preparing the farm to be eligible to receive an Environment Easement from the Department of Agriculture.
- Improve productivity. That is, the system is to improve the efficiency of the workers on the farm.

To meet these goals and collaborate with the sponsor (the farm owner), we developed the following set of requirements that the system should meet. These requirements are as follows:

- The alerts need to be triggered by a change in the water conditions: level, pressure, and temperature.
- There are multiple troughs and only one pump house. The system needs to allow for multiple troughs to communicate wirelessly with the pump house (the central hub) from where the alerts should be sent.
- The alerts must have enough information for workers to know which troughs and pump house need to be checked.
- The monitoring and alert system must have low operational, installation and maintenance costs. The materials need to be inexpensive in case they need to be replaced and they should be installed or replaced without a technician.

3 Solution Design and Prototype

The solution that our team designed consists of a central hub sub-system to be installed at the pump house and remote sub-systems/units at two water troughs, each at different nearby pasture. The water trough remote units would send the information about the changes in water conditions at the troughs to the central hub. The central hub would also receive the information regarding the pump house's well pressure and air temperature. The information regarding the water conditions would then be sent from the central hub to a mobile device as an SMS text message. In this section, we provide an overview of the central hub and trough subsystems of the solution and their components.

3.1 Central Hub Sub-system - Pump House

The central hub of the system is located in the pump house on the farm where the well pressure and air temperature are monitored. The components of the central hub are a water-pressure sensor, a temperature sensor, an Arduino UNO, a Raspberry Pi 3, an Xbee Pro S3B wireless transmitter/receiver unit, an Access Point, and an Android smart phone (Global System for Mobile (GSM) modem app).

Figure 1 shows how these components are interconnected. The water pressure and air temperature sensors are three wire sensors. The pressure sensor is hooked up to the micro-controller (Arduino) with power, ground, and data wires. The data wire of the pressure sensor is connected to the pin out A0 and the temperature sensor is connected to the pin out 5 on the Arduino. The Arduino unit is then connected to the Raspberry Pi via a serial connection. An access point is used to connect the RPi3 to the GSM modem running on the Android smart device. The Xbee Pro S3B was connected to the RPi3 via a USB connection that we ran outside of the pump house and mounted on the roof of the structure for a line-of-sight between the central hub Xbee and the water trough sub-system Xbee unit.

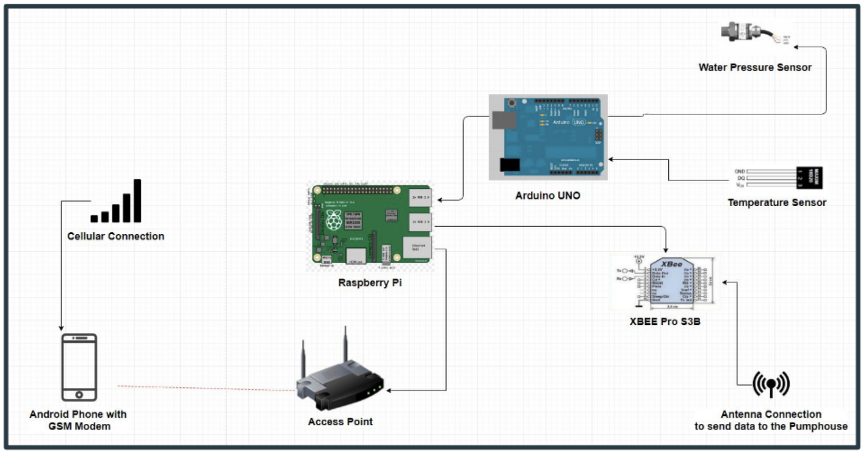


Figure 1: Central Hub - pump house sub-system components

The following are brief descriptions of the sub-system components.

Dfrobot Water Pressure Sensor

The well pressure is monitored using an analog water-pressure sensor (see Figure 2b) from Gravity

[2].

DS18B20 Temperature Sensor

The ambient temperature is monitored by a DS18B20 waterproof digital sensor (see Figure: 2a). We chose this sensor because it's waterproof [3] and suitable for a farm environment. This sensor has three wires: ground, power, and Data. Our Data wire ran to a breadboard that connected a 4.7 kilo-ohm resistor between the Data wire and the power wire per the supplier's recommendation.

(a) Ds18B20 Temperature Sensor.

(b) DfRobot Water Pressure Sensor.

Figure 2: Temperature and Pressure Sensors

Arduino Uno

The Arduino UNO was chosen as our micro-controller because it functions as a great analog to digital converter, which is required for the pressure sensor. It also supports the digital signal from the temperature sensor.

Raspberry Pi 3 (RPi3)

At the pump house we used a Raspberry Pi 3 as the central hub that receives data from the water trough sub-systems via the Xbee RF units. The RPi3 has the capability to connect to the access point via an Ethernet connection or a Wi-Fi connection.

Xbee Pro S3B

We selected the Xbee Pro S3B [4] wireless unit to send data from the water trough sub-systems to the central hub (pump house) sub-system. The main reason for selecting this specific model is that it operates at 900MHz, which is a band of RF that is able to travel long distances with relatively low power consumption.

Access Point

We used a number of different access points. At the end, we selected a Linksys Access Point due to its low cost.

Android Blu

We used an Android Blu device and a GSM modem application to send the alert SMS text messages to the farmer's mobile device. This Android Blu device is connected wirelessly (Wi-Fi) to the access point located in the pump house.

3.2 Water Trough Sub-system

In the water trough, a water-level sensor was placed. The water-level sensor had two wires, a ground and a power, that we ran to a bread board that split the power wire with a 560 ohm resistor to the A0 pin output and the 5V power supply. At the post, which is 25 feet away from the water trough, a water-tight enclosure containing an Xbee Pro S3B wireless unit that is mounted on an Arduino shield is placed. This unit would send information regarding the condition of the water at the water trough to the central hub at the pump house. The Arduino is powered by a solar cell and rechargeable battery. Figure 3 shows how these components are interconnected.

Arduino, Xbee & Arduino Xbee Shield

The difference between the Arduino used in the central hub and the one used in the water trough sub-system is that the later has an Arduino shield that enables us to mount the Xbee unit on the Arduino. This way the Xbee can be powered from the Arduino unit rather than a separate power feed. See Figure 3.

eTape Liquid Sensor

To measure the water level, we used an Adafruit eTape Liquid Level Sensor [5] because it came in a variety of lengths that could fit different water trough sizes.

Solar Panel & Battery

To power the systems in remote locations of the farm, we chose to go with a 20W 12V xExpert-Power 12 Volt 20 Ah EXP12200 Rechargeable SLA Battery. The solar panel and 12V 20 Ah battery is shown in Figure 4.

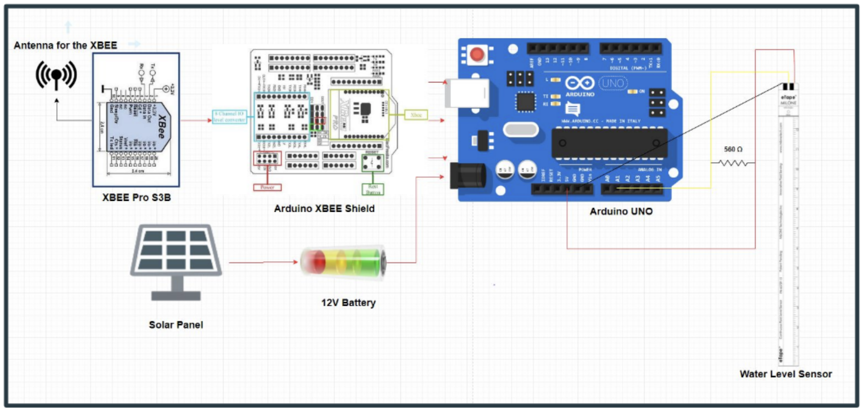


Figure 3: Water trough sub-system components

Figure 4: Solar panel and battery at the water troughs

4 Solution Implementation Phases

4.1 Overview

In this section, we describe the phases that we worked through during the course of developing this project. Figure 5 depicts the phases with a brief description of each phase.

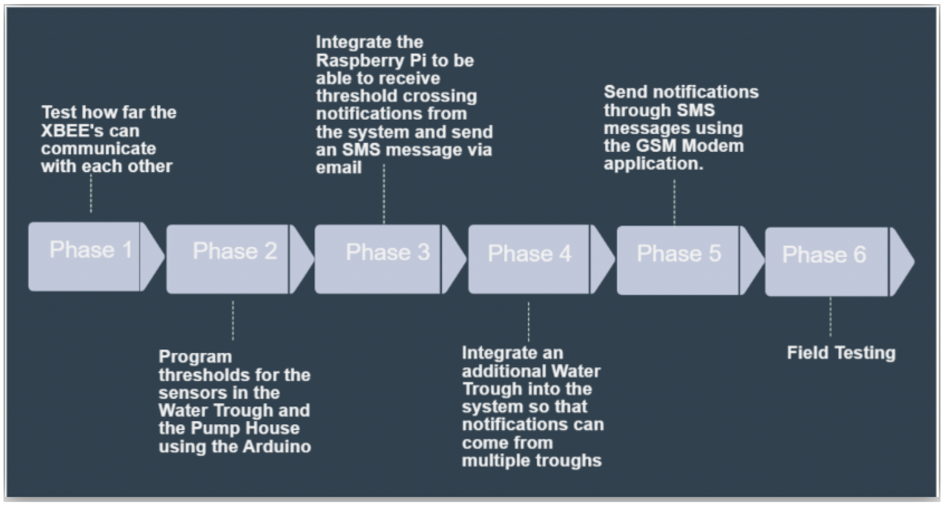


Figure 5: Project phases

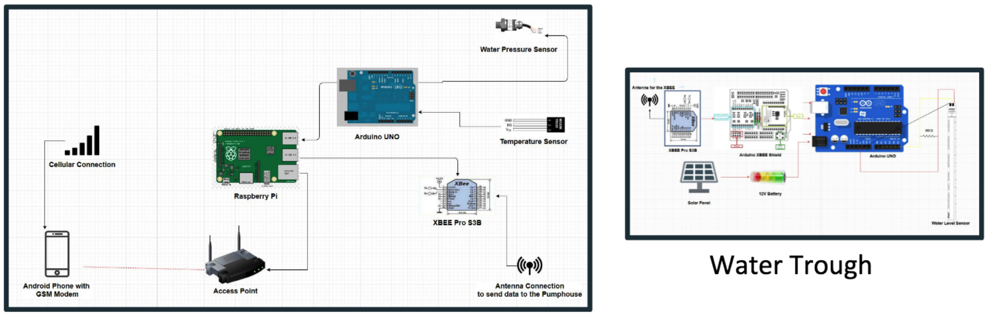
However, before we go into the details of these phases, we briefly describe the farm layout including the locations of the pump house (the central hub) and the two closest water troughs. In addition, we introduce the two system arrangement scenarios we adopted to conduct our pilot study in the field (that is on the farm).

(a) Farm Location. (b) Water Trough Locations. (c) Water Pump Location.

Figure 6: Aerial View Map

4.2 Farm Layout and System Arrangements

As seen in Figures 6a, 6b, and 6c, the pump house has two water troughs located adjacent to it in two different fields. The water trough that is more towards the south was designated as Water Trough 1 while the one to the Northeast was called Water Trough 2. Both water troughs posed challenges. While Water Trough 1 is closer to the central hub at the pump house, it does not have line-of-sight between the Xbee modules because a hill is in the way which could interfere with the RF signal from the water trough sub-system. The issue with Water Trough 2 is that while it has line-of-sight to the pump house, there is a line of trees in the way that could interfere with the RF signal. Fortunately, the Xbee modules were still close enough to broadcast a signal that the pump house could use to read the data and trigger alerts as appropriate.



Water Trough

Pump House

Figure 7: Arrangement 1 - One Water Trough and One Pump House

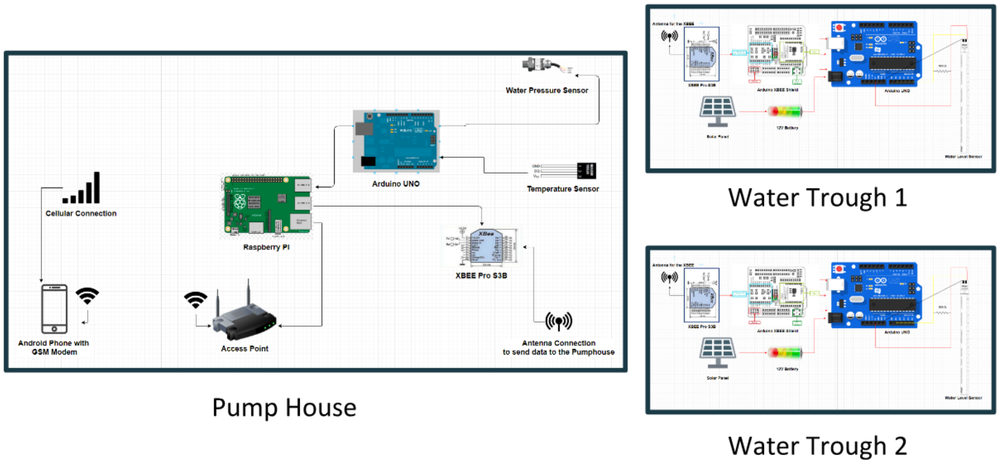


Figure 8: Arrangement 2 - Two Water Troughs and One Pump House.

To verify that the proposed solution is capable of handling one or more water troughs we decided to test two system and sub-system arrangements. In arrangement 1, the system consists of one water trough subsystem and one central hub as shown in Figure 7. In arrangement 2, the system consists of two water trough subsystems and one central hub as shown in Figure 8.

4.3 Implementation Phases

4.3.1 Phase One: Testing the Xbees

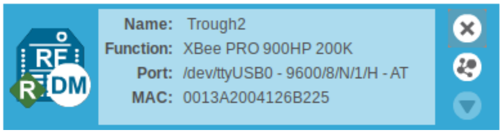
Before we could move forward with the implementation of our design, we had to develop a method that would allow the data on the water conditions at the water troughs to be sent to the pump house. We selected a wireless transmission module called an Xbee to do this. Specifically, we used a model of Xbee called the XBee Pro S3B. The Xbee utilizes a protocol called Zigbee that allows information to be sent at low power consumption over long distances (up to Superior Line Of Sight (LOS) of up to 28 miles with a high-gain antenna). An Xbee module was used at the water troughs to send data regarding the water condition and another Xbee was placed at the pump house to receive the data from the water trough sub-systems. See Figures 9a, 9b, and 9c to see the general settings that were used to set up the Xbees.

Name: Raspberry Pi
Function: XBee PRO 900HP 200K
Port: /dev/ttyUSB0 - 9600/8/N/1/H - AT
MAC: 0013A20040D7CAA9

Name: Trough 1
Function: XBee PRO 900HP 200K
Port: /dev/ttyUSB0 - 9600/8/N/1/H - AT
MAC: 0013A2004126DA34

(a) the Pump House set to AT mode.

(b) Trough 1 set to AT mode.



Name: Trough2
Function: XBee PRO 900HP 200K
Port: /dev/ttyUSB0 - 9600/8/N/1/H - AT
MAC: 0013A2004126B225

(c) Trough 2 set to AT mode.

Figure 9: Configuration details of the Xbee.

To test this arrangement, the Xbees were tested at different distances within the lab, and at different angles to determine how well they were able to send and receive data. Then, the Xbees were taken out to the field outside of the lab on campus to test how far we were able to maintain communication between the two Xbees. The result was that communication was successfully maintained to a long distance even with non-superior LOS and non high-gain antenna. Certainly, it was beyond the distance from the central hub and each of the two closest water troughs) [6].

4.3.2 Phase Two: Programming Thresholds

The next phase was to determine at what point (threshold) the sensors at the water troughs and the pump house would send information about the water conditions. To set these thresholds, we programmed the Arduino UNOs at both the pump house and the water troughs. The thresholds are the points/levels at which if the conditions of the pump house and water trough fall below or above, the Arduino UNO would send the notification data to the central hub. For the pump house, the threshold was a pressure of below 35 PSI or above 70 PSI. For the water troughs and the sensors used, if the water level fell below five (5) inches, the information would be sent to the central hub located at the pump house to trigger an alert.

Threshold for the Water Level Sensor

The water-level sensor at the water trough has two wires from the sensor itself: a power and a ground wire. The sensor had a length of 12 inches. We programmed the Ardunio unit so that an alert message is sent if the water level in the water trough is below five (5) inches. Figure 3 shows the water trough sub-system including the eTap water-level sensor.

Threshold for the Water Pressure Sensor

The Ardunio unit was programmed to trigger a text alert if the pump house pressure went out of the threshold of 35 PSI to 70 PSI. Figure 1 shows the block diagram of the central hub sub-system including the water pressure sensor.

Threshold for the Temperature Sensor

Once we were able to reliably get the temperature of the surrounding area of the pump house, we set a trigger to send an alert once the temperature dropped below 32 $^\circ F$.

4.3.3 Phase Three: Introducing the Raspberry PI.

In this phase, we introduced the Raspberry Pi 3 to the central hub (pump house) sub-system. The Raspberry Pi would be receiving the information from the water troughs through the Xbees module. To receive information regarding the pump house, the pressure sensor would be connected to a micro-controller that would allow it to analyze the pressure readings. The micro-controller (Arduino) is connected to the Raspberry Pi which sends the readings to the user's mobile device through an Simple Mail Transfer Protocol (SMTP) gateway, meaning that it would take the data and format it into an email. That email would then be sent to the SMTP gateway and then it would be changed into an SMS message. Note that in Phase 5, we decided to use a different approach known as GSM modem.

At this point, a Python script was developed on the Raspberry Pi 3 that would take the information sent from the water trough and the pump house sub-systems and send the data as an SMS message through the SMTP Gateway. This script must run on a regular basis so that the alerts are sent to the farmers in a timely manner [7].

4.3.4 Phase Four: Add another Water Trough

Next, we added a second water trough sub-system to the monitoring alert system. We used the same components as in the first water trough. By doing this, we could test the system and see how the system could handle and process data sent to the central hub from multiple water troughs.

In the lab environment, the system was able to easily send information from the components that would be at the water troughs and the components that would be at the pump house. When we went to the farm, we encountered a problem where the information from the water troughs did not trigger alert messages to the farmer's mobile device. After some troubleshooting, we determined that the issue was that too much data was sent at the same time causing the system to experience collision and overload conditions. To resolve this issue, we enhanced the Python scripts to control the time and the rate at which each water trough sub-system sent data to the central hub sub-system.

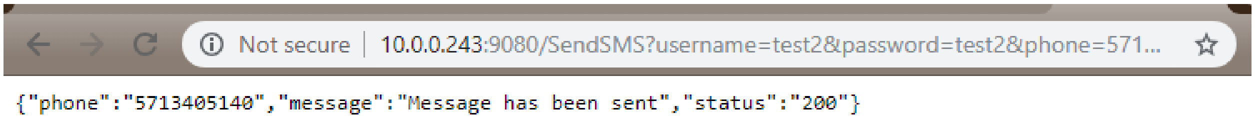
4.3.5 Phase Five: GSM Modem

In this phase, we adopted a new method to send information from the system to a farmer's mobile device. We incorporated an application called GSM Modem [8]. This is a mobile app that is available in the Android app store. It allows a mobile device to operate as a GSM gateway. This method allows for the alerts from the water troughs and the pump house to be sent to the farmer's mobile device as an SMS message through a cellular connection.

To make use of the function of the GSM Modem application, we needed to understand the mechanics of a commmand line tool called cURL [9]. Typically, cURL is used to send information to a specific IP address. Figure 10 represents a test of cURL. Once, the application was installed onto a mobile device, a number of settings was required to grant the app permission to send information from the mobile device [8].

For the GSM Modem application [10] [11] to work with our system, it was connected to the Access Point located in the central hub. Along with the mobile device that has the GSM Modem application, the Raspberry Pi was also connected to the same Access Point so that the data that was received from the water troughs and pump house could be sent to the GSM enabled mobile device to be sent to the farmer's mobile device.

To be able to send the information with the use of the GSM Modem application, new python scripts on the Raspberry Pi were written to accommodate the new method of sending the alert to a farmer's mobile device [12] [13]. Also, we decided to create a log [14] so that there is a record of the alerts sent out of the system.



```
10.0.0.243:9080/SendSMS?username=test2&password=test2&phone=571...
```

```
{"phone":"5713405140","message":"Message has been sent","status":"200"}
```

Figure 10: Test of the curl function

4.3.6 Phase 6: Field Testing (Pilot Study)

To test the viability of the system in a real-world environment, the alert monitoring system was installed on the farm. This would give us a good idea of how the system would withstand the farm's harsh environment with actual cattle roaming around. That is, how the system works when the cattle are doing their daily routine. The system would be at the farm for a few weeks. This should be enough time to receive a few alerts according to the farm's owner.

Installation at the Water Troughs for the Pilot Study

At the water troughs, there is a compartment that can be opened. The water level sensor was placed within this compartment. Once the sensor was properly positioned, the wire of the sensor was run out of the water trough, as shown in Figure 11. Once the wire was run from the trough, the compartment was closed. The wire was then run through a conduit and the conduit was buried 6 inches to a foot underground. The conduit was run to a post that housed the micro-controller that would run the code that would operate the water-level sensor. The micro-controller box with the connection to the water trough was connected to a solar-powered battery for a power source.

Figure 11: The compartment of the water trough that was removed to place the water level sensor within it as shown with a red circle.

Installation at the Pump House for the Pilot Study

At the pump house, vital components such as the Raspberry Pi 3, Arduino UNO, and the Android device were placed inside a hard box to keep them safe. The water-pressure sensor was connected to a valve that fed to a pressurizing tank for the underground well. The water-pressure sensor was then connected to the Arduino UNO. The Arduino UNO was then connected to the Raspberry Pi from the USB-A port on the Arduino UNO to the USB-B port on the Raspberry Pi. The Raspberry Pi 3 was then connected to the access point through an Ethernet connection. The Android device was then connected to the access point wirelessly.

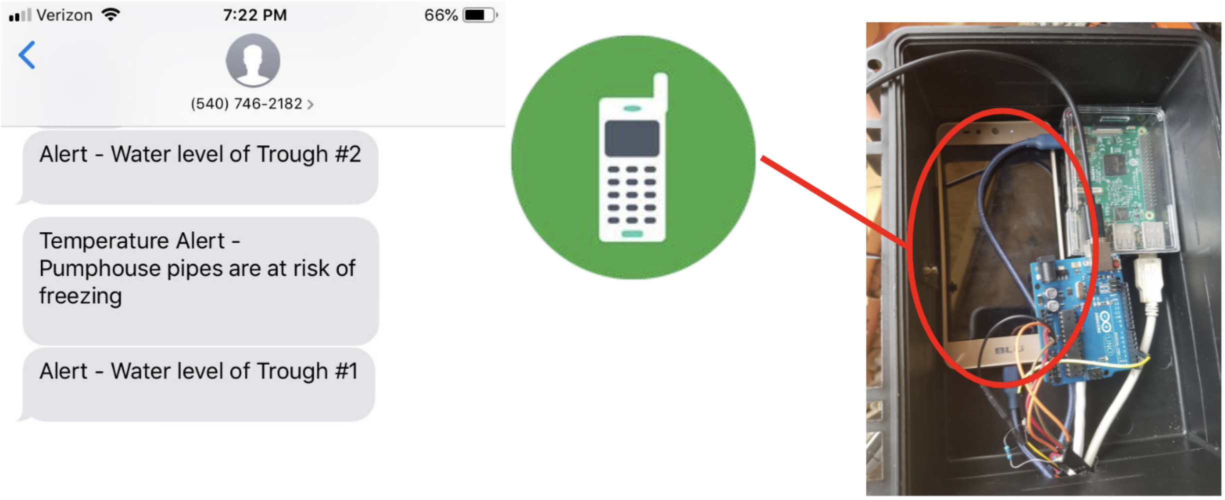
5 Solution Field Assessments

5.1 Overview

On the farm, we conducted several field tests. The focus of the field tests was to make sure that the water troughs were able to communicate to the central hub at the pump house. These field tests were then followed by the pilot study where we installed both the water trough and the central hub sub-systems.

5.2 Field Tests Assessment

We launched our field tests to determine the durability and robustness of our water-trough alert system. During the first few tests, we were able to determine that the water troughs were close enough and had clear enough line of sight to communicate with each other and with the central hub. However, our system ran into an issue sending the text message alerts when the text messages that were sent contained messages for both trough 1 and trough 2. It was at this point that we changed the text message delivery system from using emails to send text messages via ann SMS gateway to using an Android phone running a GSM modem app to send the text alert. We enhanced our Python scripts to take advantage of the new SMS message delivery and resolve the collision issue. While the cellular signal from the Cricket Wireless mobile service provider did drop when the phone was in the pump house, the mobile phone was still able to send text messages as shown in Figure 12. At this point, we decided that we were ready to move on with a Pilot Study where we went for an all-out system test where we would install and try out both the water troughs and the pump house sensors for an extended period.



Alert - Water level of Trough #2

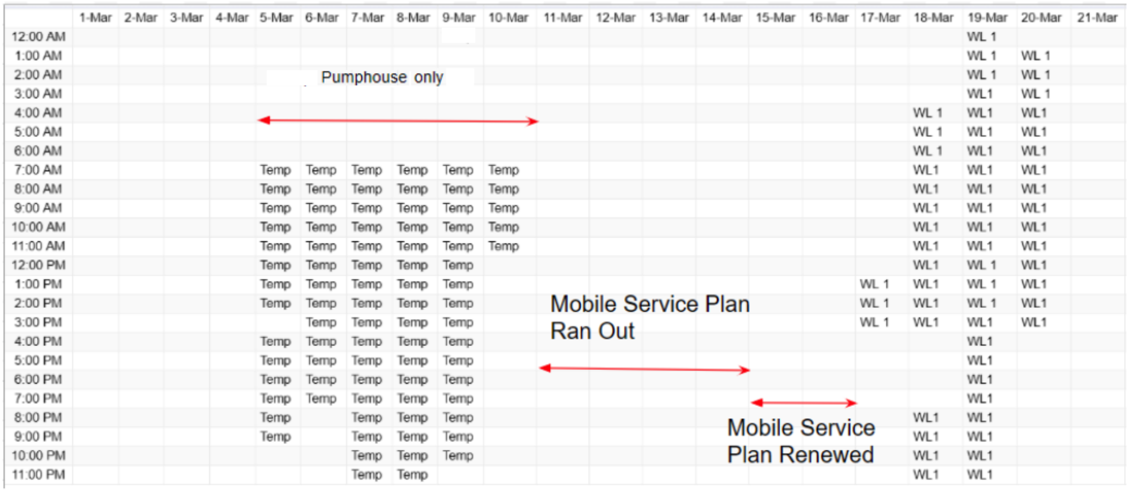
Temperature Alert - Pumphouse pipes are at risk of freezing

Alert - Water level of Trough #1

Figure 12: Text Message Alerts from the Farm

5.3 Pilot Study Assessment

We went to the farm with our equipment with the intent to complete the installation of the water trough sub-system in one day. This did not happen and we had to revise and extend our installation plans for yet another visit. At the end, and due to our lack of knowledge as to how challenging this type of installation is on a farm, we had to make several visits until we had the full system installed and operational. And we were able to receive alerts from the water troughs but we ran into an issue where the RPi would try and send multiple alerts jumbled up together. We ended up with alerts triggered by the pump house only from March 5th to March 10th as shown in Figure 13. From March 11th to March 14th, we received no alerts because our mobile service plan ran out. The plan was renewed on March 15th but we received no alters until March 17th. We received water level (WL) alerts from March 17th to March 20th. On analyzing these alerts and visiting the farm again, we discovered some cattle, and decided to investigate the micro-controller box at water trough 1 and eat an exposed section of the wire. This led to the resistance value taken in by the Arduino to be registered as infinity, which was technically above the threshold set to send a text message alert every 15 minutes. See Figure 13 for the information received from the monitoring alert system during the period from March 5th to March 20th.



Pumphouse only

Mobile Service Plan Ran Out

Mobile Service Plan Renewed

	1-Mar	2-Mar	3-Mar	4-Mar	5-Mar	6-Mar	7-Mar	8-Mar	9-Mar	10-Mar	11-Mar	12-Mar	13-Mar	14-Mar	15-Mar	16-Mar	17-Mar	18-Mar	19-Mar	20-Mar	21-Mar
12:00 AM																			WL 1		
1:00 AM																			WL 1	WL 1	
2:00 AM																			WL 1	WL 1	
3:00 AM																			WL1	WL 1	
4:00 AM																		WL 1	WL1	WL1	
5:00 AM																		WL 1	WL1	WL1	
6:00 AM																		WL 1	WL1	WL1	
7:00 AM					Temp	Temp	Temp	Temp	Temp	Temp								WL1	WL1	WL1	
8:00 AM					Temp	Temp	Temp	Temp	Temp	Temp								WL1	WL1	WL1	
9:00 AM					Temp	Temp	Temp	Temp	Temp	Temp								WL1	WL1	WL1	
10:00 AM					Temp	Temp	Temp	Temp	Temp	Temp								WL1	WL1	WL1	
11:00 AM					Temp	Temp	Temp	Temp	Temp	Temp								WL1	WL1	WL1	
12:00 PM					Temp	Temp	Temp	Temp	Temp									WL1	WL 1	WL1	
1:00 PM					Temp	Temp	Temp	Temp	Temp								WL 1	WL1	WL 1	WL1	
2:00 PM					Temp	Temp	Temp	Temp	Temp								WL 1	WL1	WL 1	WL1	
3:00 PM						Temp	Temp	Temp	Temp								WL 1	WL1	WL1	WL1	
4:00 PM					Temp	Temp	Temp	Temp	Temp										WL1		
5:00 PM					Temp	Temp	Temp	Temp	Temp										WL1		
6:00 PM					Temp	Temp	Temp	Temp	Temp										WL1		
7:00 PM					Temp	Temp	Temp	Temp	Temp										WL1		
8:00 PM					Temp		Temp	Temp	Temp									WL1	WL1		
9:00 PM					Temp		Temp	Temp	Temp									WL1	WL1		
10:00 PM							Temp	Temp	Temp									WL1	WL1		
11:00 PM							Temp	Temp										WL1	WL1		

Figure 13: The information that was received from the farm during the Pilot Study.

6 Discussion, Conclusion & Next Steps

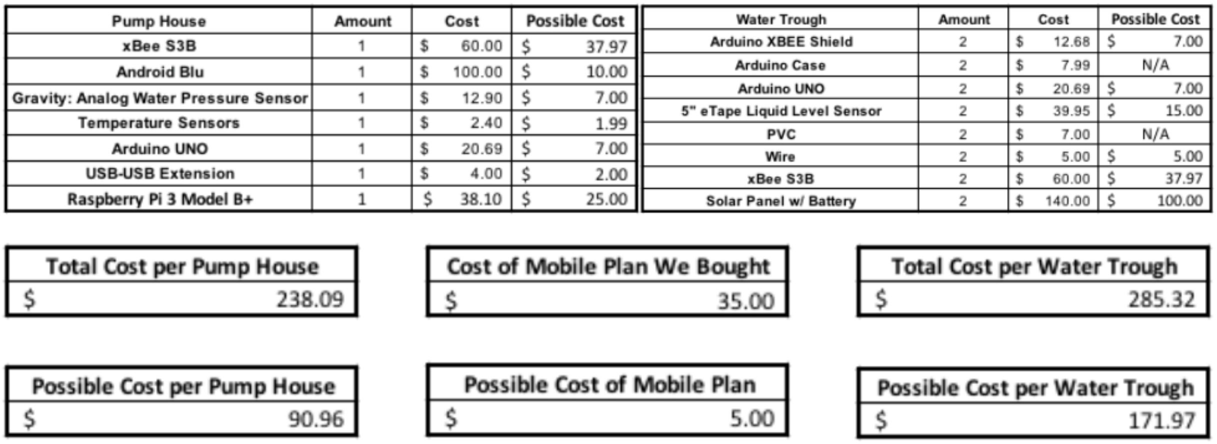
6.1 Discussion

Technical Challenges

While the alert monitoring system was able to send alerts to a mobile device, there were a number of challenges through the course of testing the system in the field. The first challenge was trying to use the GSM Modem application. We had to understand how the cURL utility works and how it is exercised on the app. The second challenge was when we went to the farm to install the system. We ran into many manual and physical issues that we were not prepared for considering the limited time we had. The third issue was how to make the water-trough sub-system ”cattle” proof.

Cost Analysis

With the materials that we used to create the monitoring alert system, the cost of the system was broken down between the components at the pump house, the components at the water troughs, and the cost of the mobile plan. For the pump house, the cost was $238.09. The cost at the water trough was $285.32. The cost of the mobile data plan was $35. After some research, the price of the pump house could come down to $90.96, the cost at the water trough could come down to $171.97, and the price of the mobile data plan could be reduced to $5. For more details, see Figure 14.



Pump House	Amount	Cost	Possible Cost
xBee S3B	1	$ 60.00	$ 37.97
Android Blu	1	$ 100.00	$ 10.00
Gravity: Analog Water Pressure Sensor	1	$ 12.90	$ 7.00
Temperature Sensors	1	$ 2.40	$ 1.99
Arduino UNO	1	$ 20.69	$ 7.00
USB-USB Extension	1	$ 4.00	$ 2.00
Raspberry Pi 3 Model B+	1	$ 38.10	$ 25.00

Water Trough	Amount	Cost	Possible Cost
Arduino XBEE Shield	2	$ 12.68	$ 7.00
Arduino Case	2	$ 7.99	N/A
Arduino UNO	2	$ 20.69	$ 7.00
5" eTape Liquid Level Sensor	2	$ 39.95	$ 15.00
PVC	2	$ 7.00	N/A
Wire	2	$ 5.00	$ 5.00
xBee S3B	2	$ 60.00	$ 37.97
Solar Panel w/ Battery	2	$ 140.00	$ 100.00

Total Cost per Pump House
$ 238.09

Cost of Mobile Plan We Bought
$ 35.00

Total Cost per Water Trough
$ 285.32

Possible Cost per Pump House
$ 90.96

Possible Cost of Mobile Plan
$ 5.00

Possible Cost per Water Trough
$ 171.97

Figure 14: A cost analysis of the monitoring alert system.

Comparison Analysis

We compared our alert system to a number of comparable and commercially available systems. Gallagher Group Limited has a system that monitors the level of water within a water trough and sends alerts but not to a mobile device. They sell separate device that you have to install in a building to use. This company is based in New Zealand and their system is only available in New Zealand and Australia. There are two other companies that offer similar systems: Bentek Systems (Industrial Pipeline Monitor) and Spartan Cam (wildlife Camera System). Again, these are limited in scope and capabilities and more expensive than our system

6.2 Conclusion & Next Steps

In conclusion, we were able to create a monitoring alert system that was able to send alerts on changes to the water level and/or pump house conditions on the farm to a mobile device as an SMS message. This system is still in the prototype phase and still needs additional features to be considered a system hardened for long-term use. Some of these features are as follows:

- **Remote access to the system** A feature that would be greatly needed for the monitoring alert system would be the ability to connect to the system from outside the system. By allowing for remote access to the system, an engineer or technician could diagnose and troubleshoot the system without being physically on site.
- **Protection of vital components** With the system being damaged through the interactions with the cattle, it's vital that the wiring of the water trough sub-system be protected. The conduit underground would need to be uncovered and buried again once it has been properly adjusted.

- **Creation of a Mesh Network** The Xbee modules have a function that allows them to have a special network configuration. The Xbees can form a mesh network. A mesh network is that each Xbee has a connection to each other. If the connection between an Xbee (on the water trough sub-system) and that on the central hub is lost, then the water trough Xbee can still send the data through connections to other water trough Xbee modules that are closer to the central hub sub-system. This will keep the system functional despite the lost connection.

Acknowledgment

The authors would like to thank our sponsor for giving us the opportunity to work on our project on his farm. We would like to thank the Department and College of Science and Engineering (CISE) at James Madison University (JMU) for their support throughout the Senior Capstone Project process and the development of this paper. Special thanks are to Mr. Paul Goodall, our sponsor, for his continuing support and guidance and his willingness to offer his farm for our pilot study. Our thanks go also to Mr. Safaa Al-Badry for his lab support and Paul Henriksen for his proofreading of the early version of the paper.

References

[1] "Chesapeake Bay Program." https://www.chesapeakebay.net/news/blog/fencing_out_manure_pollution, Retrieved: 2022-02-14.

[2] "Gravity Water Pressure Sensor SKU SEN0257." https://wiki.dfrobot.com/Gravity__Water_Pressure_Sensor_SKU__SEN0257/, Retrieved: 2022-02-14.

[3] "Programmable Resolution 1-Wire Digital Thermometer." https://github.com/May-DFRobot/DFRobot/raw/master/DS18B20.pdf, Retrieved: 2022-02-14.

[4] "Digi XBee-PRO 900HP RF Module." https://www.digi.com/products/embedded-systems/digi-xbee/rf-modules/sub-1-ghz-rf-modules/xbee-pro-900hp, Retrieved: 2022-02-14.

[5] "12" eTape Liquid Level Sensor ." https://www.adafruit.com/product/464, Retrieved: 2022-02-14.

[6] "Digi XBee® Industrial Gateway." https://www.digi.com/resources/documentation/digidocs/PDFs/90001542.pdf, Retrieved: 2022-02-15.

[7] "Scheduling Jobs with python-crontab." https://stackabuse.com/scheduling-jobs-with-python-crontab/, Retrieved: 2022-02-15.

[8] "GSM Modem." https://apkpure.com/gsm-modem/dvms.smsoffline, Retrieved: 2022-02-15.

[9] "curl.1 the man page." https://curl.haxx.se/docs/manpage.html, Retrieved: 2022-02-15.

[10] "GSM Modem SMS - Free SMS Gateway - Android App." https://sindhitutorials.com/videos/android-gsm-modem/android-app, Retrieved: 2022-02-15.

[11] "Send FREE SMS in PHP - Receive FREE SMS in PHP." https://sindhitutorials.com/videos/android-gsm-modem/send-receive-free-sms-php, Retrieved: 2022-02-15.

[12] "How to use Python to execute a cURL command?." https://stackoverflow.com/questions/25491090/how-to-use-python-to-execute-a-curl-command?lq=1, Retrieved: 2022-02-15.

[13] "Execute curl command within a Python script." https://stackoverflow.com/questions/26000336/execute-curl-command-within-a-python-script, Retrieved: 2022-02-15.

[14] "Logging Cookbook." https://docs.python.org/3/howto/logging-cookbook.html, Retrieved: 2022-02-15.