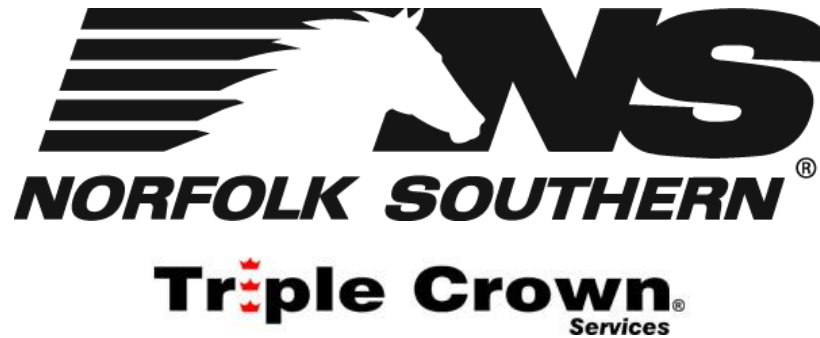




RoadRailer Air Brake Fault Localization

Final Report

ECE 480 Team Eight

Marcelo de Castro
Saurav Shrestha
Dilo Benjamin
Abdulaziz Najm
Tia Twigg

April 30th, 2010



Executive Summary

Air brakes in trains are operated by changing the pressure level in a pneumatic line which runs the length of a train. The brakes engage when the pressure drops inside the hose and released when pressure increases. This is called a 'fail-safe' system. However, if there is a substantial blockage in the line, application of brakes beyond that point may not be possible. Currently, the personnel can tell if a fault, i.e. if a leak or an obstruction, exists but they are not easily able to pinpoint the location. To locate the problem, they have to walk the entire length of the train which can reach up to one and an half miles in length.

Team 8 was tasked to find a fault localization solution. Using wireless pressure sensors and accompanying software, Team 8 created a solution that informs the user of fault and its approximate location. For this, the pressure sensors will be installed at certain intervals throughout the train. The sensors will then transmit the pressure readings to a central gateway location. With the back-end processing software that Team 8 created, the incoming pressure reading will be verified to see if it falls within the established normal range that user sets. If it does not, a message will be flashed in the screen notifying of abnormal pressure reading; hence indicating a problem. After checking which sensor transmitted the abnormal data, the user can walk directly to the location of the sensor and search that area for the fault in the air brake.

Acknowledgements

ECE 480 Design Team 8 would like to extend their heartfelt thanks and sincere gratitude to the following people for their assistance throughout the semester:

- Our sponsor, Mr. Michael Theisen of Triple Crown Services for his guidance and support throughout the semester.
- Our facilitator, Professor Subir Biswas; course instructors Professors Erik Goodman and Michael Shanblatt; ECE shop staffs Mr. Brian Wright, Mr. Gregg Mulder, and Ms. Roxanne Peacock for contributing their time, suggestions, and hard work, ensuring our success in the project.

TABLE OF CONTENTS

1. Introduction	6
2. Solution Space and Approach	7
2.1 Design Specifications and Design Factor Matrix.....	7
2.2 FAST Diagram	9
2.3 Total Cost	11
2.4 Major Changes in the Design.....	12
2.5 Gantt Chart	13
3. Technical Description	16
3.1 Hardware Components.....	16
3.1.1 Hardware Design	16
3.1.2 PS1 Integrated Data Acquisition.....	17
3.1.3 Glad-Hand Adaptor	18
3.2 Software	19
3.2.1 PS1 Test Application	20
3.2.2 Multi Test Application	21
3.2.3 Back-end processing application.....	21
3.3 Mesh Networking/Hopping.....	23
3.4 Base Station	24
3.4.1 XBee-PRO.	25
4. Test Data	29
4.1 Introduction	29
4.2 Notes and Assumptions:	30
4.3 Tests and Results	33
4.3.1 Obstruction Test with Leak:.....	33
4.3.2 Leak Test 1:	36
4.3.3 Leak Test 2:	39
4.3.4 Leak Test 3:	42
4.3.5 Leak Test 4:	44
4.3.6 Leak Test 5:	46

5. Final Notes	49
5.1 Summary	49
5.2 Difficulties and Obstacles	50
5.3 Conclusion	52
6. Appendix 1	53
6.1. Technical Roles and Responsibilities	53
6.1.1 Marcelo de Castro	54
6.1.2 Saurav Shrestha	55
6.1.3 Tia Twigg	56
6.1.4 Dilo Benjamin	58
6.1.5 Abdulaziz Najm	59
7. Appendix 2	60
7.1 References	60

1. Introduction

When Team 8 was presented the challenge of creating a fault localization system, we brainstormed on what should be the most important features in our solution. Our main goal was to come up with a solution that will save manual labor by indicating the range of the fault. Also, the pressure transmitted has to be accurate, and accurate within ± 0.1 psi range as specified by the customer. The pressure transmitters also have to be durable enough to endure rain and rough train travels. Thus, time efficiency, accuracy, and durability were three of our topmost criteria.

With that in mind, Team 8 designed a fault localization system where pressure transmitters are to be installed in the rail cars at regular intervals along the train. These transmitters will send the pressure data to a gateway location. This could save many hours of labor since walking the entire length of a train over a mile long with more than one hundred cars is extremely time consuming. We used Pressure Sensor One (PS1) manufactured by Electrochem for our project as it fits all of our criteria. The user can set the sampling rate for pressure measurement, which can be as low as one reading per second. This ensures time efficiency. The pressure we will be measuring ranges from zero to one hundred psi. PS1 provides 0.1% error, which fits the accuracy criterion for our purpose. The transmitter has a metal covering that is built to be used in harsh environments. This guarantees that the transmitter can withstand temperatures ranging from -20°F Michigan winters to 120°F Florida summers.

Our current solution will save manual labors and makes the entire process of fault finding time and cost efficient. Furthermore, the software allows users to store the reading in a text file, which can be referred to in the future, if needed. The back-end processing application our team created allows users to set a normal pressure range. If the incoming pressure reading is outside set range, the application will provide real-time notification. This eliminates the need for personnel to constantly watch for error to occur.

2. Solution Space and Approach

2.1 Design Specifications and Design Factor Matrix

The RoadRailer is a hybrid system, which is a road (truck) trailer that has been specially adapted to integrate with the railroad for transportation. RoadRailer trains are "Bi-Modal", meaning that the trailers can be used in two modes of service, both on a regular road/highway as well as on a standard rail. These trailers are built to lift their road tires. This allows the trailer to latch onto special rail-wheel adapter trucks called bogies. The air brake system on road trailers is similar to the air brakes used on a railroad car. One by one, the trailers are set onto the bogies, the air brakes are connected together using a Glad-Hand connector hose and then one (or more) locomotives can pull the line of connected trailers as a Unit Train.

Every approach for our design began with a device to be inserted into the air brake system by connecting with the Glad-Hands. This device will connect the air hose from the road trailer to the hose on the rail bogie. Once outfitted with an air pressure transducer, our unit can take accurate pressure measurements and transmit them wirelessly to a suitable base station such as a laptop or a portable hand-held device. Units will be placed on bogies at intervals along the length of the train. Once in place, pressure measurements can be taken at various locations along the train. Data is sent back to a computer/device where it can be analyzed by the user.

The initial meeting with the sponsor, Mr. Mike Theisen, revealed a project scope with three stages. The first stage was the selection of the pressure transducer for sensing the pressure and also the data transmission. The second stage was to design a way to connect the pressure sensor into the current air brake system using Glad-Hands. The final stage was to create the back-end software used to read the data. Once these tasks were accomplished, we would next have to come up with a creative way to simulate a leak and an obstruction in the air break line that we put together in the ECE 480 lab. This is because we would not be able to test our devices on the trains

themselves. Also revealed at the first meeting were the design specifications for power, size, cost, flexibility, accuracy, and safety.

For our power source, due to the lack of electrical wires along the train, and the fact that bogies and trailers are often replaced during a trip; power resources are not available. In order to meet our final goal power must be kept low so it can operate using batteries only. The final device should fit into limited available space, so it can connect between the bogies and the trains. The cost of the pressure transducer was dependent on the range of the wireless signal transmission. Since the trains can extend for up to 1.5 miles, the cost for the transducer was moderately high.

The design we considered relied mostly on a pressure transducer. A pressure transducer is fundamentally any device that converts an applied pressure into an electrical signal. Typically a pressure transducer output signal is one generated by the primary sensing element. Since it is difficult to achieve tight electrical tolerances with the sensing elements during manufacture, it is often necessary to add a circuit to compensate for errors over the operating temperature range. However the pressure transducer still retains the natural characteristics of the particular sensing technology employed such as linearity, hysteresis, repeatability, stability and frequency response. In fact these are the main reasons for using a pressure transducer which apart from the compensation circuitry are the purest form of pressure sensor. Pressure transducers are generally available with three types of electrical output; millivolt, amplified voltage and 4-20mA current loop. For our design, we used a 4-20 mA Output Pressure Transducer. These types of transducers are also known as pressure transmitters. Since a 4-20mA signal is least affected by electrical noise and resistance in the signal wires, these transducers are best used when the signal must be transmitted long distances.

The ultimate design needs to be flexible enough to be integrated into the current system, so our sponsor would not have to alter their current system. The device needs to be accurate and robust as to be able to operate in the high temperatures of Florida,

or in the cold temperatures of Ontario. Also, each device should be able to measure the pressure at a determined location and wirelessly send the data thru distances that could range from a few feet to one and a half miles. Finally, the device should not pose any safety threats to the workers that will be installing them.

The design decisions were based on a number of design specification discussed with our sponsor. These specifications were rated (as seen in Table 1) and their weights in the component matrix.

Design Criteria	Relative Order of Significance	Relative Weight
Accuracy	1	8
Robustness	2	7
Durability	3	6
safety	4	5
flexibility	5	4
Power	6	3
Cost	7	2
Size	8	1

Table 2.1: Design Factor Matrix

2.2 FAST Diagram

Our team decided to use a FAST Diagram (which is a visual layout of our project's function) in the initial design of our project which helped us to understand the functions to be eliminated or improved in the project. In order to develop our fast diagram we first identified the 'Basic Function', which is the principal reason for the product's existence. By asking the question: "How is this Function actually

accomplished?"; we placed the secondary Functions to the right of the Basic Function (secondary functions assist in, or are necessary for the realization of a Basic Function) and then checked the FAST diagram by starting at the right and working left. We asked the question: "Why this function must be performed?"

Sensor Components	Description	Functions
Locate Anomaly	Localize problem	Basic
Pressure transducer	Measure pressure	Secondary
	Convert signal	Secondary
Wireless transmitter	Transmit signal	Secondary
	Support protocol	Secondary
Wireless receiver	Receive signal	Secondary
	Update server	Secondary
Battery	Support device	Secondary
Glad-Hand adapter	Position device	Secondary
	Prevent leakage	Secondary
USB port	Transfer information	Secondary
PC interface	Collect data	Secondary
	Analyze information	Secondary
Enclosure	Increase aesthetics	Secondary
	Protect device	Secondary
Display monitor	Display measurement	Secondary

Table 2.2: Function Description

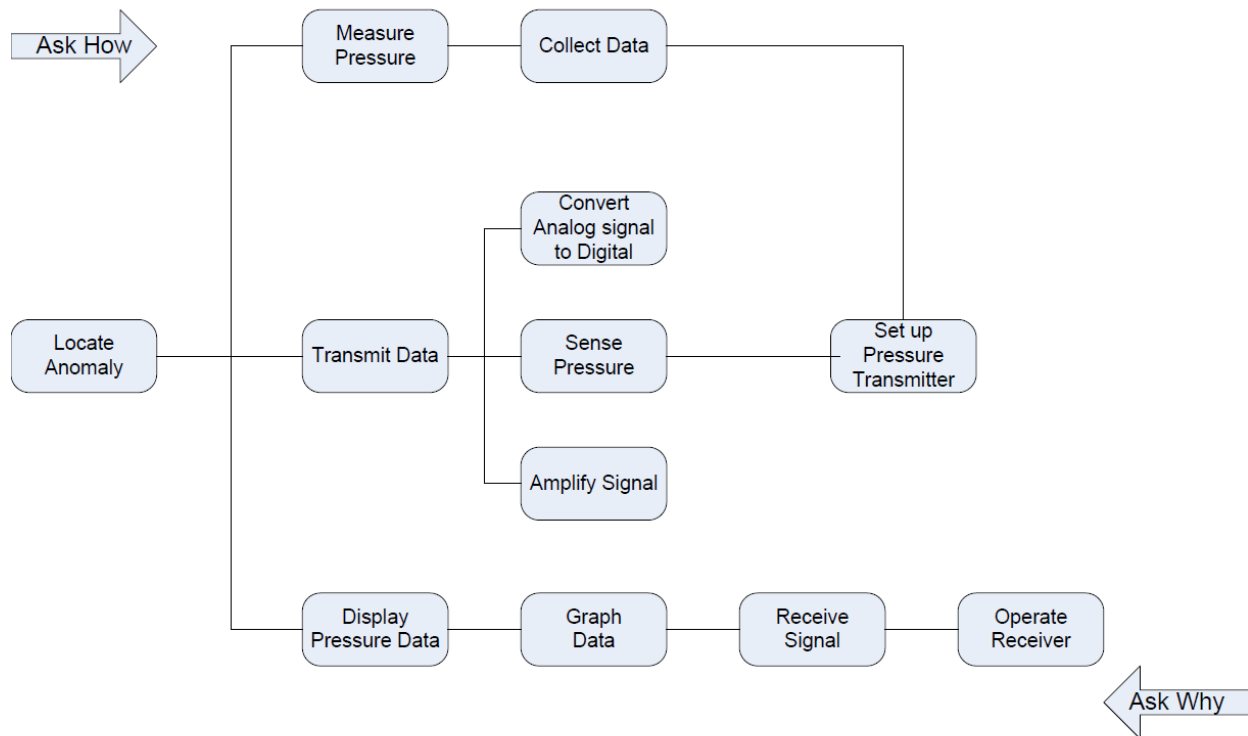


Figure 2.2: Functional Analysis System Technique Diagram (FAST)

2.3 Total Cost

Once the components required to implement the design were selected, an estimate for the budget was compiled. Each component was carefully researched, and our team is confident that these are the most affordable parts. Table 3 shows a breakdown of one unit cost. Our sponsor, should they decide to implement our system, will only need one base station and they can use as many sensors as they feel necessary.

Parts	Price Per Unit
Pressure Transducer	\$995
Base Station	\$380
Battery 2 pack	\$25
Glad-Hand Adaptors	\$20
Total Budget	\$1420

Table 2.3: Project Budget Information

2.4 Major Changes in the Design

There have been some changes in our design since the beginning. The first major change was in the design of the adaptor for the pressure sensor onto the Glad-Hand. Initially, our team had thought of designing a case for the pressure sensor, connected with the Glad-Hands. Our current design was a much more convenient idea we had to connect the sensor into the air brake system, please refer to chapter 3.1 for further details.

Another change made to the design was in the software. This is considered a change because it was not originally part of the initial project requirements, but as the semester progressed we were asked to design back-end software for the program we were currently using to see the data collected by our pressure sensors (LabVIEW). The goal of back-end testing is to compare the incoming data against a set standard baseline to indicate whether the values are within the acceptable range or not. Should the values be outside the range, a message will display, indicating the receipt of an

abnormal value. To establish the baseline, we collaborated with our sponsor to determine an ideal value range for a given temperature.

LabVIEW was selected for the convenience of user interface development. The advantage of using LabVIEW is its ability to process data and display it on a computer screen in an easy to read fashion. This ultimately results in a reduction on power consumption in the system. The graphical interface available makes LabVIEW both easy to understand and appear visually appealing. The multiple sensor application mode in LabVIEW simplifies programming and troubleshooting by providing the user with visual feedback of data collected by all sensors.

2.5 Gantt Chart

The next stage in exploring the design solution for Design Team 8 was to develop a Gantt chart. The initial Gantt chart can be found below in Figure 2. The Gantt chart served as a timeline estimate that should be followed in order to meet the project goals by the deadline. After the initial meeting with our sponsor, a timeline was drawn up in a Gantt chart.

		Task Name	Duration	Start	Finish	Pred	Resource Names
2	☒	Design adaptor for testing	2 days	Tue 2/23/10	Wed 2/24/10	1	Team
3	☐	Build adaptor for testing	5 days	Thu 2/25/10	Wed 3/3/10	2	Marcelo/Team
4		Aquire adaptor parts	5 days	Thu 2/25/10	Wed 3/3/10		Abdul
5	☐	Test software	5 days	Tue 2/23/10	Mon 3/1/10	1	Tia
6		Debug software	5 days	Tue 2/23/10	Mon 3/1/10		Tia
7		Test Device in Lab (single sensor)	7 days	Thu 3/4/10	Fri 3/12/10	3	Dilo/Team
8		Test Prototype at Triple Crown Services	6 days	Mon 3/15/10	Mon 3/22/10		Dilo/Team
9		Refine prototype	7 days	Mon 3/15/10	Tue 3/23/10	7	Saurav
10		Order 2nd and 3rd Transmitters	5 days	Mon 3/15/10	Fri 3/19/10	7	Abdul
11	☐	Prepare Adaptors for 2nd and 3rd Transmitters	3 days	Wed 3/24/10	Fri 3/26/10	9	Marcelo
12		Aquire adaptor parts	3 days	Wed 3/24/10	Fri 3/26/10		Abdul
13		Test system (all three sensors) at MSU	7 days	Mon 3/29/10	Tue 4/6/10	11	Dilo/Team
14	☐	Test system at TCS	3 days	Wed 4/7/10	Fri 4/9/10	11	Dilo/Team
15		Gather Performance Data of Final Prototype	3 days	Wed 4/7/10	Fri 4/9/10	13	Team
16		Analyze final data	3 days	Mon 4/12/10	Wed 4/14/10	15	Tia
17		Troubleshoot	5 days	Thu 4/15/10	Wed 4/21/10	16	Team
18		Slacktime	5 days	Thu 4/22/10	Wed 4/28/10	17	Team
19		Design Day	1 day	Thu 4/29/10	Thu 4/29/10	18	Team

Figure 2.5: Design Team 5 Initial Gantt Chart

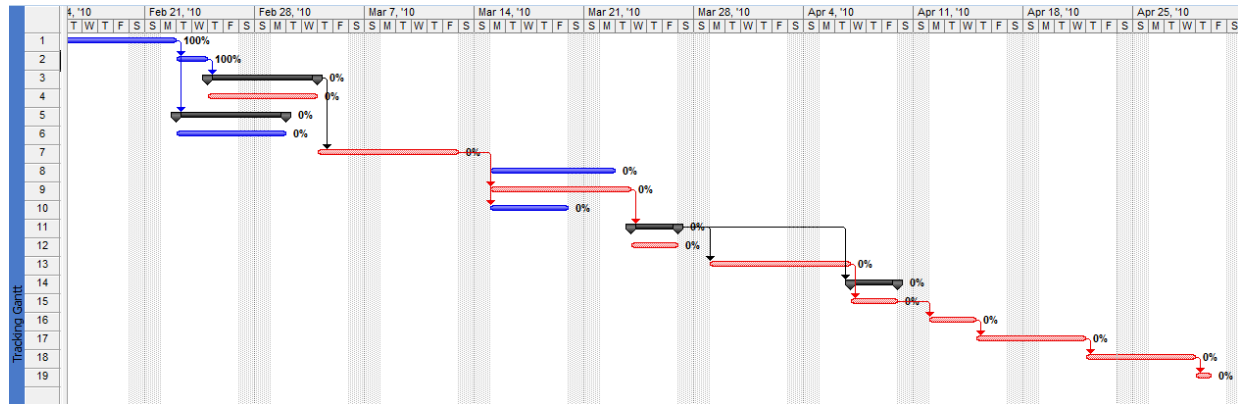


Figure 2.5.1: Design Team 8 Initial Gantt Chart Critical Path

Although the timetable was a convenient gauge of where we should stand in the design process, we were not able to adhere to it completely. As time went by, while working on the project our team reevaluated the Gantt chart and made some adjustments due to some issues that had arisen. Mainly, all changes in the Gantt chart were due to delays in acquiring parts, or parts that once received had to be returned due to manufacturing defects. We were not grossly off track, but the time required for troubleshooting was more than expected. Because of this, and the time needed for testing and modification our initial schedule suffered. Despite these discrepancies, the end goal was reached with time to finalize and refine.

	i	Task Name	Duration	Start	Finish	Pred	Resource Na
1	✓	Order Transmitter & Receiver	7 days	Fri 2/12/10	Mon 2/22/10		Abdul
2	✓	Design adaptor for testing	2 days	Tue 2/23/10	Wed 2/24/10	1	Team
3	✓	Build adaptor for testing	5 days	Thu 2/25/10	Wed 3/3/10	2	Marcelo/Tea
4	✓	Aquire adaptor parts	5 days	Thu 2/25/10	Wed 3/3/10		Abdul
5	✓	Test software	5 days	Tue 2/23/10	Mon 3/1/10	1	Tia
6	✓	Debug software	5 days	Tue 2/23/10	Mon 3/1/10		Tia
7	✓	Test Device in Lab (single sensor)	7 days	Thu 3/4/10	Fri 3/12/10	3	Dilo/Team
8	✓	Test Prototype at Triple Crown Services	6 days	Mon 3/15/10	Mon 3/22/10		Dilo/Team
9	✓	Refine prototype	7 days	Mon 3/15/10	Tue 3/23/10	7	Saurav
10	✓	Order 2nd and 3rd Transmitters	5 days	Mon 3/15/10	Fri 3/19/10	7	Abdul
11	✓	Prepare Adaptors for 2nd and 3rd Transmitters	3 days	Wed 3/24/10	Fri 3/26/10	9	Marcelo
12	✓	Aquire adaptor parts	3 days	Wed 3/24/10	Fri 3/26/10		Abdul
13	✓	Test with adaptor from Triple Crown	5 days	Tue 4/13/10	Mon 4/19/10	11	Dilo/Team
14	✓	Build our own adaptor	3 days	Mon 4/19/10	Wed 4/21/10	11	Marcelo
15	✓	Re-order sensors. One was damaged	5 days	Thu 4/22/10	Wed 4/28/10	14	Abdul
16	📅	Test system (only two sensors) at MSU	3 days	Fri 4/23/10	Tue 4/27/10	14	Dilo/Abdul/Tia
17	📅	Gather Performance Data of Final Prototype	2 days	Mon 4/26/10	Thu 4/29/10	16	Team
18		Analyze final data	1 day	Thu 4/29/10	Fri 4/30/10	17	Tia
19		Design Day	1 day	Fri 4/30/10	Mon 5/3/10	18	Team

Figure 2.5.2: Design Team 5 Final Gantt Chart

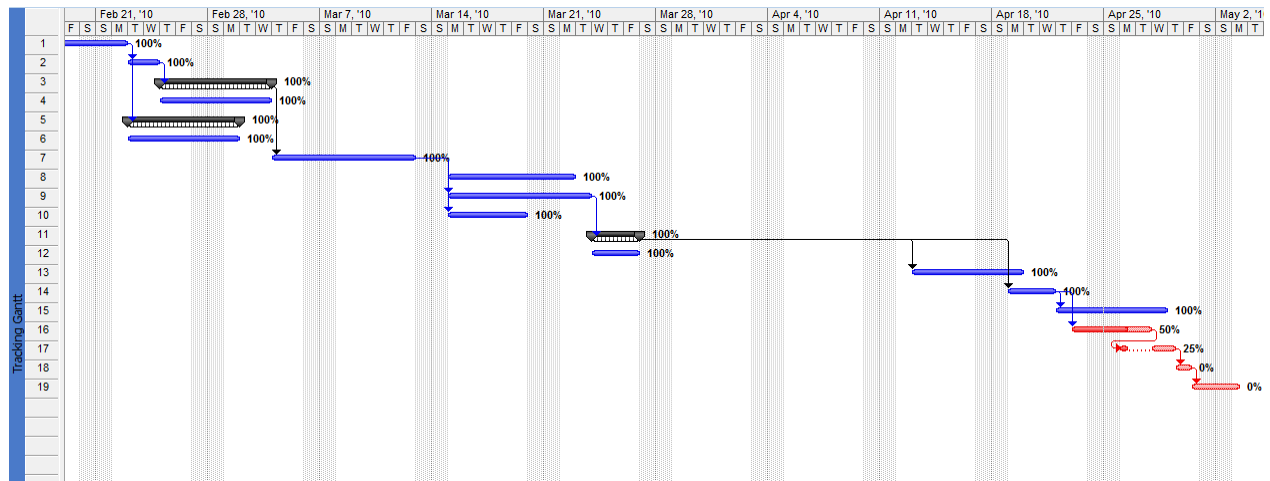


Figure 2.5.3: Design Team 8 Final Gantt Chart Critical Path

3. Technical Description

3.1 Hardware Components

3.1.1 Hardware Design

In creating our prototype, our hardware design was broken down into two main components: Glad-Hand adaptor (for our pressure transducer), and wireless pressure transducer. The following sections discuss in more detail their purpose to the team's prototype and how they were implemented in the final design.

The 'PressureSensorOne' (PS1) is a full-bridge strain gauge circuit with a, 4-20 mA output. Highly accurate pressure measurements are transmitted wirelessly to a suitable base station and passed to a host system such as a PC. Because the information is transmitted as current, the signal is relatively immune to voltage drops from long runs and noise from motors, relays, switches and industrial equipment. That is why 4-20 mA output pressure transducers, also known as transmitters, are a suitable choice when signals have to be transmitted over a long distance with the sensor at a remote location.

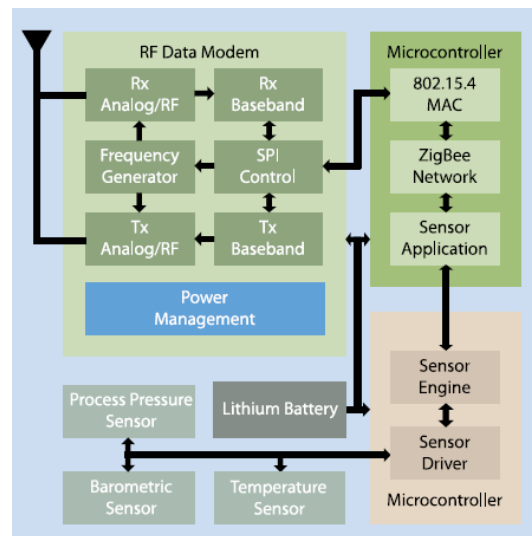


Figure 3: PS1 Function Block Diagram

3.1.2 PS1 Integrated Data Acquisition

The signals of the PS1 are amplified, sampled and digitized into 24-bit words at a rate of up to 1 Ks/S. This is accomplished using a precision mixed-signal microcontroller (MCU) which integrates an analog-to-digital converter (ADC) and a high-speed 8051-compatible CPU. Accurate measurements are passed to the wireless subsection for transmission. The PS1 also uses ZigBee wireless which combines an ultra-low power IEEE 802.15.4 transceiver with ZigBee software to form a wireless communication. This mesh network allows for ease of installation around obstructions, redundancy in communication paths and longer range.

Performance	
Pressure range	100 (psi)
Pressure reference	Absolute, Sealed Gauge, and True Gauge
Error	±0.1% of range
Output format	32-bit floating point (IEEE754)
Output rate	0.3 to 480 readings per second
Environmental	
Operating temperature	-40°C to 85°C
Compensated temperature	-40°C to 85°C
Thermal effects	Zero shift ±0.1% of range
	Span shift ±0.1% of range
Electrical	
Battery Type	Lithium thionyl chloride (Li/SOCl ₂)
Size	2/3 AA (1 required)
Voltage	3.6 V
Battery life	2 months
Wireless	
Range Indoor/urban	100m
Range Outdoor line-of-sight	1500m
Operating frequency	ISM 2.4 GHz

Table 3.1: PS1 Technical Information



Figure 3.1: PS1 Measurements Shown in Millimeters

3.1.3 Glad-Hand Adaptor

As mentioned in Chapter 2, one of the major changes in our project design was the design of the adaptor for the pressure sensor onto the Glad-Hand. Initially, our team had thought of designing a case for the pressure sensor, connected with the Glad-Hands. Our current design is much more convenient for connecting the sensor into the air brake system.

A threaded hole was drilled on the top of one of the Glad-Hands, and outfitted with a fitting tube which connects to a ‘hose end connector’ especially made for pressure applications. The pressure sensor was attached to the hose using a radiator clamp, and at the end of the sensor another ‘hose end connector’ was attached. In order to get the pressure into the sensor, we connected the two ends of the ‘end rose connectors’ with a black ‘INSTA-GRIP hose’, also especially manufactured for pressure applications. Both ends of our device are connected with Glad-Hand adaptors, that way we can attach the device to the current system in use at Triple-Crown Services.

Overall we did not encounter problems with the design of our Glad-Hand adaptor. The device has been tested with 120 psi of pressure and it does not leak. Our lab tests run at 80 psi but we tested the connection at 120 psi to ensure that they were air-tight and would not experience problems.



Figure 3.1.2: Glad-Hand Adaptor for PS1

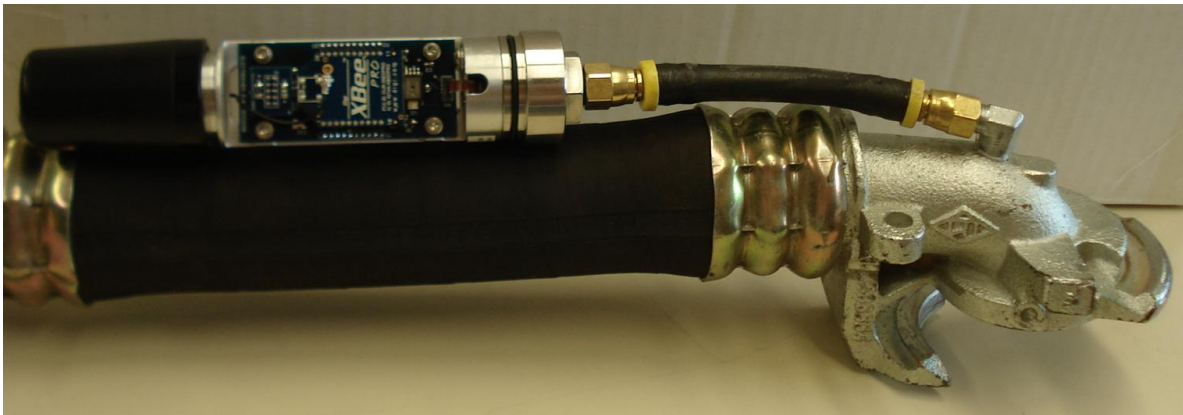


Figure 3.1.3: PS1 Attached to Glad-Hand Adaptor

3.2 Software

Two software programs PS1 Test Application and Multi Test Application came with our pressure sensors. PS1 Test Application is used to collect a single packet of measurements from one sensor. This packet of information includes measurements such as the strength of the received signal, primary and reference pressure, battery voltage as well as primary temperature. Multi Test Application is used to collect those same packets of information from multiple sensors.

Our team added a back-end processing application to the Multi Test Application. In this application, the user will enter a high limit and a low limit pressure value as the baseline pressure. The incoming pressure data will then be automatically checked to see if it falls within the range. If it does not, a message will flash stating that the pressure is above or below the limit as the case might be.

3.2.1 PS1 Test Application

PS1 Test Application is a user-friendly application that displays the primary pressure, reference pressure, and battery voltage as well as primary and module temperature in a graphical format. After a connection is made to the receiver using X-CTU software that came with the sensors, the user has to enter the pressure sensor serial number as well as the number of data counts. Pressure sensor will then start taking the measurements mentioned above and output them in the form of a graph. Figure 3.2.1 shows a figure of the PS1 Test Application in use.

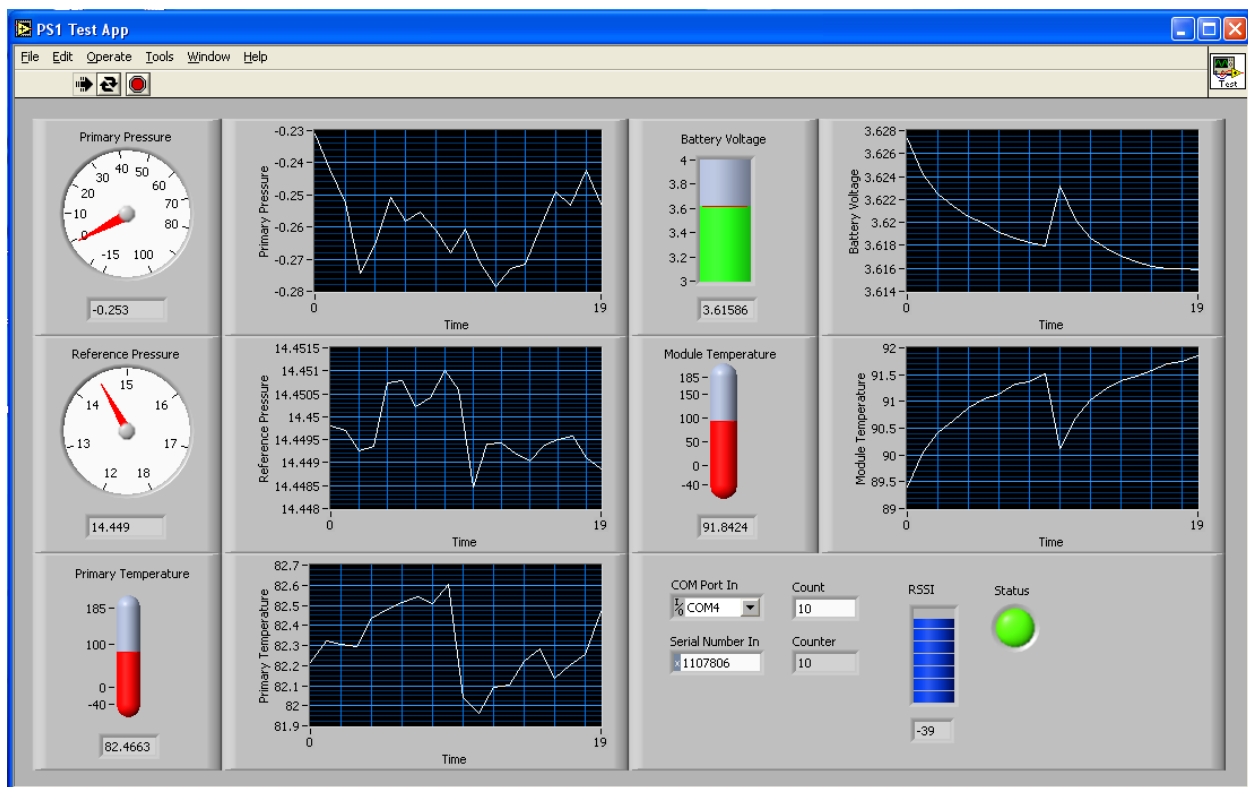


Figure 3.2.1: PS1 Test Application in use

3.2.2 Multi Test Application

Multi Test Application is an application that displays the primary pressure, reference pressure, and battery voltage as well as primary and module temperature in a numerical, tabular format. After a connection is made to the receiver using X-CTU software, the user has to enter the pressure sensor serial numbers as sampling period, i.e., the time interval between consecutive readings. Figure 3.2.2 shows a figure of the Multi Test Application in use that is taking data from two sensors.

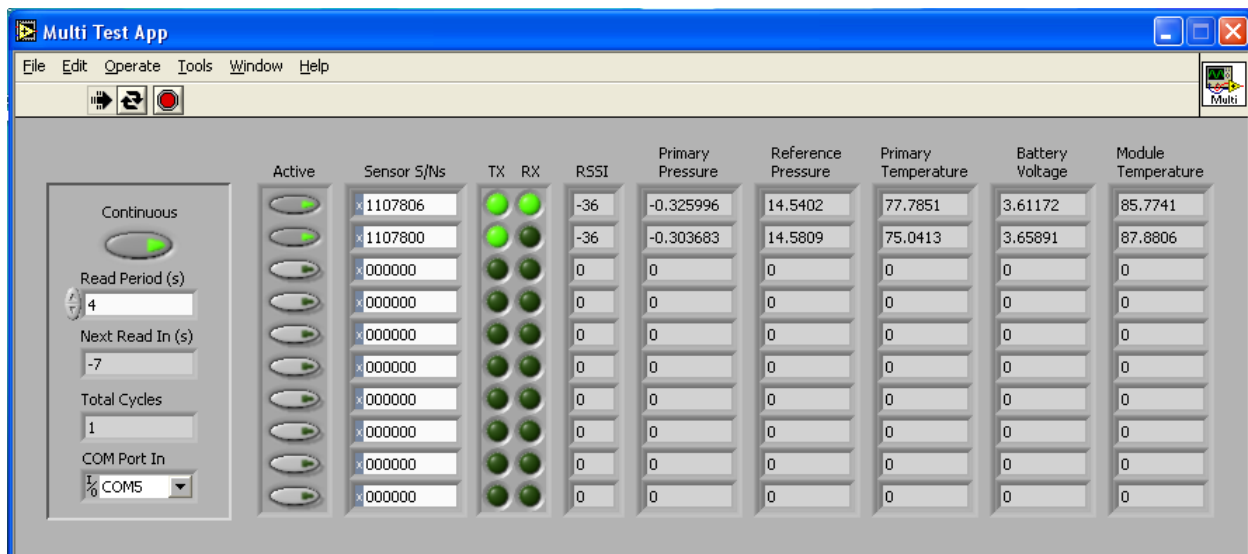


Figure 3.2.2: Multi Test Application in use (with two sensors)

3.2.3 Back-end processing application

Team 8 designed and added a back-end processing application to the Multi Test Application. Figure 3.2.3(a) below shows the back-end processing application added to the Multi Test Application. The user sets the low limit and high limit. Those values are graphed as a straight, horizontal line in the figure. The incoming values from each of the sensors are also plotted as separate waveform versus time (seconds) shown in the figure. If the waveform goes above the high limit or below the low limit, message indicating so will be displayed.

3.3 Mesh Networking/Hopping

During our initial research for the sensors, we looked into the mesh networking/hopping features of the sensors under consideration. Before we purchased the PressureSensorOne (PS1) we asked the technical sales representative if the sensor in question would be able to also act as a repeater and ‘hop’ signals from other sensors down the line to the base station. The salesman assured us, via email that the sensors would indeed be able to do that, so we decided to proceed with the purchase of one sensor and a base station. We did this with the idea that, if after some preliminary tests and inspection the sensor performed well, we would purchase two more sensors to complete our project tests.

Once the sensor arrived, we each took quite some time to study the instruction manual. Focus for this portion was on the mesh networking and hopping capabilities of the sensor. The manual clearly states, under the PS1 features section on page 7, that it “offers the building blocks for different types of networking known as ‘star, mesh, and cluster tree’”. It describes the benefits of such network routing schemes such as self-healing and longer range. Mesh networking is the function that allows for the hopping feature we wanted to implement. Upon closer inspection, the details of how to implement the mesh networking feature were not found. The search for the needed information was then carried to the internet. Several manuals were downloaded and read from websites such as ZigBee, IEEE (802.15.4 standard), and Crossbow.

After careful scrutiny of the available material, it was realized that the information needed to command the sensors to act as repeaters was not to be found. It was decided that a call needed to be made to the technical support staff at Electrochem. It was through this conversation with the technical personnel that we were informed that the sensors we had purchased from them were not yet capable of the hopping function that was desired. The Electrochem representative told us that that feature was in development, but had not yet been finished due to lack of demand. When we informed him of our intended use of the sensor he realized that there was a very good application for the hopping feature and told us he would pass on the request.

We made one more attempt at finding a way to force the sensors to hop by contacting Digi, the company who made the XBee controller for the sensor. They informed us that it might be possible, but we would have to reprogram the firmware, which had been programmed at Electrochem. This might be able to be done by removing the XBee chip from the sensor and placing it into the setting on the base station. That way there would be a direct connection from the computer to the chip. An attempt to reprogram the chip wirelessly was not recommended. We decided that we did not have enough time remaining in the project to attempt this so we have not tried it.

This problem effectively ended the networking and hopping portion of our project and forced us to focus on the remaining areas of testing. The current proposed solution to the hopping problem is to implement an antenna attached to its own base station down along the train. This base station can then hop the signals from the outlying sensors back to the main base station at the controlling computer. The antenna and base station are both currently available for purchase from the Electrochem website. It is not confirmed how far the antenna can send the relayed signals. More than one may be necessary.

3.4 Base Station

The XBee-PRO RS-232 RF Modem is a ZigBee/IEEE 802.15.4 compliant solution that has RS-232 interfacing built-in to its design. Out-of-box, the modem is equipped to sustain excellent range (2-3x the range of standard ZigBee modems) and requires no additional configuration for immediate RF communications. The modem transfers a standard asynchronous serial data stream between two or more devices. Simply feed data into one modem, then the data is sent out the other end of the wireless link. Its built-in RS-232 interface allows for rapid integration into existing data systems. It is also very convenient to other sensors because it uses the most common technologies, such as a USB port to connect to the computer, Network compatible with other ZigBee/802.15.4 devices.

3.4.1 XBee-PRO.

The XBee-PRO Module includes the MC9S08GT60 micro-controller and MC13193 RF chip. The XBee-PRO Module also contains a switch that switches the antenna between RF transmit and receive modes; however, after the switch is a Low Noise Amplifier (LNA) for Receive Mode and a Power Amplifier (PA) for Transmit Mode. The XBee-PRO Module also contains:

- An analog switch that can switch the MC9S08GT60 A/D reference from the 2 volt regulated output of the MC13193 and a pin on the XBee-PRO module.
- A digital potentiometer that is used to adjust the power output of the PA.

Both the XBee-PRO RF Modem and host (PC) settings can be viewed and adjusted using Max-Stream's proprietary X-CTU Software

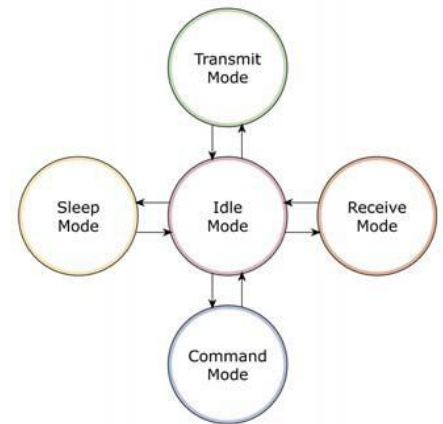
Here we explain how to configure the base station from the X-CTU

◇ Modes of Operation:

The XBee-PRO RF has 5 main modes of operation, they represents the stage at which the device is at, each one of these modes is there to serve a purpose and can be adjusted to what suites the user.

Sleep Mode: Sleep Modes enable the RF modem to enter states of low-power consumption when not in use. This is to help maintain a longer battery life and save power.

Command Mode: To modify or read RF Modem parameters, the modem must first enter into Command Mode – a state in which incoming characters are interpreted as commands.



Networking Modes: Before transmitting data over-the-air, the RF modem will first undergo CCA (Clear Channel Assessment). If the CCA fails, the packet will not be transmitted. This mode consists of two main branches:

- Transmit mode: to transmit commands to the sensors to get a reading
- Receive mode: actually receiving the readings from the sensors to be able to interpret and output the results.

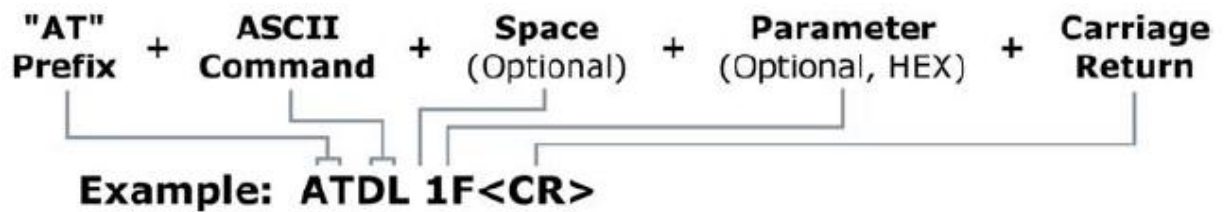
Idle Mode: this is the mode where the device is not performing any tasks, instead it is basically the transition mode in between any two of the previous modes, so the which switching between sleep mode and transmit mode, the device has to go back to the ideal stage before it moves to the transmit mode.

◇ **Programming the RF Modem:**

The first thing we do once we open the X-CTU program is that we go to the Modem configuration tab and press the Read button under “Modem Parameters and Firmware” this will enable the program to access the modem.

To enter command mode: go to the Terminal tab and you just Send the 3-character command sequence “+++” and observe guard times before and after the command characters. When a command is sent to the RF modem, the modem will parse and execute the command. Upon successful execution of a command, the modem returns an “OK” message. If execution of a command results in an error, the modem returns an “ERROR” message.

Any RF modem will accept a packet that contains a broadcast address. When configured to operate in Broadcast Mode, receiving modems do not send ACKs (Acknowledgements) and transmitting RF modems do not automatically re-send packets as is the case in Unicast Mode.



- Most of the commands that are sent or received in this terminal can be adjusted in the next tab which is the "Modem Configuration tab". Some of the parameters you can change or commands you can send include reset, restore battery life measure single measurement and many others.
- The other way to send packets or commands is through the Assemble packets button in the terminal Tab, you can choose to send your commands either using ASCII or HEX decimals; the more common one is HEX for this mode. You basically follow the below diagram in determining the command and parameters, most commands are typically 10 bytes long.
- To calculate Checksum: Not including frame delimiters and length add all bytes keeping only the lowest 8 bits of the result and subtract from 0xFF.

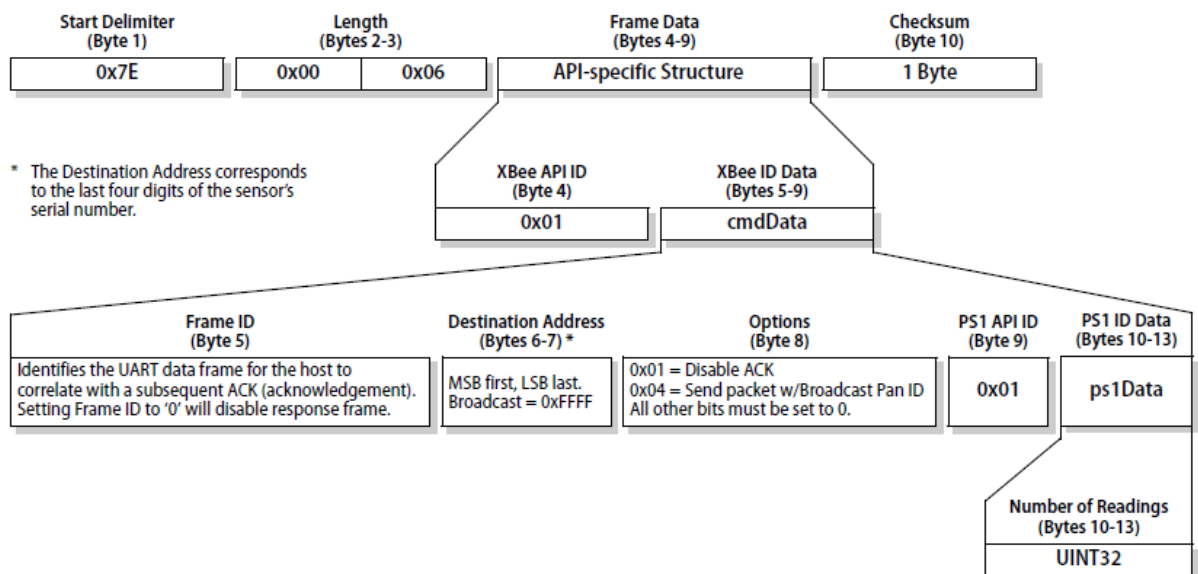


Figure 3.4.1(a): Command structure for assembled packet

- List of commands:

AT Command	Command Category	Name and Description	Parameter Range	Default
BD	Serial Interfacing	Interface Data Rate. Set/Read the serial interface data rate for communications between the RF modem serial port and host.	0 - 7 (custom rates also supported)	3
CC	AT Command Mode Options	Command Sequence Character. Set/Read the ASCII character value to be used between Guard Times of the AT Command Mode Sequence (GT+CC+GT). The AT Command Mode Sequence enters the RF modem to AT Command Mode.	0 - 0xFF	0x2B ('+' ASCII)
CH	Networking & Security	Channel. Set/Read the channel number used for transmitting and receiving between RF modems. Uses 802.15.4 protocol channel numbers.	0x0C - 0x18	0x0C (12d)
CN	AT Command Mode Options	Exit Command Mode. Explicitly exit AT Command Mode.	-	-
CT	AT Command Mode Options	Command Mode Timeout. Set/Read the period of inactivity (no valid commands received) after which the RF modem automatically exits AT Command Mode and returns to Idle Mode.	2 - 0xFFFF [x 100 ms]	0x64 (100d)
DB	Diagnostics	Received Signal Strength. Read signal level [in dB] of last good packet received (RSSI). Absolute value is reported. (For example: 0x58 = -88 dBm) Reported value is accurate between -40 dBm and RX sensitivity.	0 - 0x64 [read-only]	-
DH	Networking & Security	Destination Address High. Set/Read the upper 32 bits of the 64-bit destination address. When combined with DL, DL defines the destination address used for transmission. To transmit using a 16-bit address, set DH parameter to zero and DL less than 0xFFFF. 0x000000000000FFFF is the broadcast address for the PAN.	0 - 0xFFFFFFFF	0
DL	Networking & Security	Destination Address Low. Set/Read the lower 32 bits of the 64-bit destination address. When combined with DH, DL defines the destination address used for transmission. To transmit using a 16-bit address, set DH parameter to zero and DL less than 0xFFFF. 0x000000000000FFFF is the broadcast address for the PAN.	0 - 0xFFFFFFFF	0
GT	AT Command Mode Options	Guard Times. Set required period of silence before and after the Command Sequence Characters of the AT Command Mode Sequence (GT+ CC + GT). The period of silence is used to prevent inadvertent entrance into AT Command Mode.	0x02 - 0xFFFF [x 1 ms]	0x3E8 (1000d)
ID	Networking & Security	PAN ID. Set/Read the PAN (Personal Area Network) ID. 0xFFFF indicates a message for all PANs.	0xFFFF	0x3332 (13106d)
MY	Networking & Security	16-bit Source Address. Set/Read the RF modem 16-bit source address. Set MY = 0xFFFF to disable reception of packets with 16-bit addresses. 64-bit source address (serial number) and broadcast address (0x000000000000FFFF) is always enabled.	0 - 0xFFFF	0
P0	Diagnostics	PWM0 Configurations. Select/Read function for PWM0.	0 - 1	1
PL	RF Interfacing	Power Level. Select/Read power level at which the RF modem transmits.	0 - 4	4
RE	(Special)	Restore Defaults. Restore RF modem parameters to factory defaults. Follow with WR command to save values to non-volatile memory.	-	-
RN	Networking & Security	Random Delay Slots. Set/Read the minimum value of the back-off exponent in the CSMA-CA algorithm that is used for collision avoidance. If RN = 0, collision avoidance is disabled during the first iteration of the algorithm (802.15.4 - macMinBE).	0 - 3	0
RO	Serial Interfacing	Packetization Timeout. Set/Read number of character times of inter-character delay required before transmission. Set to zero to transmit characters as they arrive instead of buffering them into one RF packet.	0 - 0xFF [x character times]	3
RP	Diagnostics	RSSI PWM Timer. Enable a PWM (pulse width modulation) output (on pin 3 of the RF modems) which shows RX signal strength.	0 - 0xFF [x 100 ms]	0x28 (40d)
SH	Diagnostics	Serial Number High. Read high 32 bits of the RF modem's unique IEEE 64-bit address. 64-bit source address is always enabled.	0 - 0xFFFFFFFF [read-only]	Factory-set
SL	Diagnostics	Serial Number Low. Read low 32 bits of the RF modem's unique IEEE 64-bit address. 64-bit source address is always enabled.	0 - 0xFFFFFFFF [read-only]	Factory-set
SM	Sleep (Low Power)	Sleep Mode. Set/Read Sleep Mode configurations.	0 - 6	0
SP	Sleep (Low Power)	Cyclic Sleep Period. Set/Read sleep period for cyclic sleeping remotes. Maximum sleep period is 268 seconds (0x68B0).	0x01 - 0x68B0 [x 10 ms]	0x64 (100d)
ST	Sleep (Low Power)	Time before Sleep. Set/Read time period of inactivity (no serial or RF data is sent or received) before activating Sleep Mode. The ST parameter is only valid with Cyclic Sleep settings (SM = 4 - 6). Set ST on Cyclic Sleep Coordinator to match Cyclic Sleep Remotes.	0x01 - 0xFFFF [x 1 ms]	0x1388 (5000d)
VR	Diagnostics	Firmware Version. Read firmware version of the RF modem.	0 - 0xFFFF [read-only]	Factory-set
WR	(Special)	Write. Write parameter values to RF modem's non-volatile memory so that modifications persist through subsequent power-up or reset.	-	-

Figure 3.4.1(b): Table of commands for terminal tab

4. Test Data

4.1 Introduction

The customer requested that the product should be installed every 10 to 20 bogies. This is because they want the ability to localize the problem, which will reduce the time it takes to identify the location of said problem significantly. The device will be installed during the assembly of the bogies and the trailers. At that time, we will use an adapter that will be installed, by use of glad hands, between the hose from the road trailer and the rail bogie. The glad hands seal the pressure line from one cart to another. This adapter will contain the pressure sensing and transmission unit. The user has to keep note of which device was installed at which section of the train and in which order where the devices installed, the user takes note of the device by using the serial number on the sensor, so it could be later entered into the processing program to identify where the problem is. It will need to be durable since the train travels in various harsh weather conditions and a variety of climates. The device is powered by battery since there are no power lines along the length of the train. There are two options of device operation. The first option is to have the device operating all the time. The problem with this option is that the battery life would be greatly reduced, lasting for approximately two months. The customer stated that pressure information is not needed in transit since any unexpected braking of the train is ignored. A stop to examine a problem while the train is on a schedule is very expensive. The second option is to use an onboard timer to turn off the device which is based on a user specified configuration. The downside to this option is that the configuration is user based using the base station X-CTU program; therefore it may be necessary to input or change it every time it is installed. The pressure readings will be automatically transmitted wirelessly to a main receiver system located nearby, either on the train or in the train yard terminal. Once the train is fully assembled and pressurized, the pressure readings will be examined on the computer that the receiver is attached. The user will have to set the threshold levels or limits and then wait for the back-end processing program to indicate whether the pressure is above, under or just inside the limits, the indication is going to be via a

flashing signal that clearly states the current state of the pressure reading. The device will also send out battery level indicator information along with the pressure data for the convenience of maintenance. The transmitted information will be an encoded digital signal of the sensor data which will then be decoded at the receiver.

4.2 Notes and Assumptions:

This section will address the assumptions that were made since the tests could not be performed on the actual system at Triple Crown due to difficulties, safety and security issues and potential delays of trains; these assumptions were made carefully in order to replicate the required system. With these assumptions the tests were performed and the results were analyzed, this doesn't necessarily mean that the device will work in the exact same way or produce the same results under regular everyday usage if installed on the actual train. All of these notes are to explain the kind of environment that the testing took place.

- Since our test system is very small in relative to the actual train, we had to make some adjustments, we used to clamp part of the hose to represent resistance and simulate the bogies in the actual train. Without these clamps the sensors would be very close to each other and won't be able to detect any leaks or blockage since the change in pressure when a leak is introduced is very rapid and that close of sensors to each other will not be able to detect it. Our sponsor requested these sensors to be installed about 10-20 carts apart, air pressure would take time to change between these two locations and with clamping the hose is the only way we can simulate that time. We clamped the hoses in a way where if you give it enough time with no leaks or blockage they will stabilize to the same level, just with longer time. When leaks and obstructions are introduced later on in testing sensors will be able to detect them and localize them very easily.
- For most of our test we used 50 Psi instead of 75 or 90 (what the locomotive actually produces), the reason for this is that using less Psi ranges enable us to detect and control the leaks easier, for example we needed to simulate a small

leak in the system, using duct tape or just our finger or seal the hole we are able to completely seal the hole, while when dealing with 80 Psi pressure its a lot harder to completely seal the hole and wait for the system to stabilize. We had a second run on some of these tests where we actually ran 80 Psi to verify our results and it was a lot harder but in the end we were able to simulate and produce the same result.

- We used two different applications for testing purposes, one was a single sensor test application, which took a reading once every second and outputted the result directly on a graph, and the other was a multi- sensor application where it takes reading from two or more sensors and outputs the result on a text file. The reason behind that is that the multi test application is more efficient since it compares the result from all sensors relative to each other but it takes a bit more time, we were getting a reading once every about 5 seconds.
- Batteries started to become weaker once they hit the 3.4 volts Range, for some reason once the battery reaches that level it becomes slower in the transmitting a signal and sometimes even doesn't work. We suggest that the operator changes the battery when it reaches that level.
- All of our test were performed under the same regular room temperature
- We assume that the user is going to have to set the throughput limits manually, because in the train pressure decreases between the front and the back of the train based on many different variables such as temperature, length of train and the fact the pressure decreases not in a systematic way and many others. It would be practically impossible for us to set a limit that works for all sensors because each sensor should have its own limit based on its location.
- We assume that the train locomotive pressurizes the train system hoses in the same way a nitrogen tank pressurized our test hoses air flow will move the same and leaks would have the same effect.
- We assume that locomotive pressurizes the air inside the hoses at all times, because if it doesn't then with any leaks, the train would lose all the pressure in the hoses and cause it to trigger the emergency breaks.

- We assume that both sensors are calibrated with the same zero point reference.
- We have to determine reasonable time difference for each of the readings, even though the output file shows that readings from different sensors were taken instantaneously, testing shows that there is a certain amount of lag time between the readings, the way we deal with this is that for each test scenario we let the system stabilize before we consider and analyze the results.

4.3 Tests and Results

4.3.1 Obstruction Test with Leak:

Purpose:

The purpose of this test was to simulate what would happen if there was an obstruction somewhere along the pressure line. The actual lines become obstructed when there are dents in the system which restricts the passage of air. Depending on the severity of the obstruction, it can prevent certain bogies from pressurizing to the appropriate amount. What is known as the Ball Test is tested on each trailer to determine whether there is an obstruction. A ball that is a little smaller than the pressure line is inserted at one end of the trailer and then air is pumped in at the same side. If the ball manages to come out of the other side, then it is assumed that there is no obstruction; however, if the ball does not come out of the other side or if it takes a longer time than expected, it is indicative of an obstruction. This test is performed on every trailer before it is assembled. Although this test is performed in a matter of seconds, if a blockage was introduced to the system while the train is in motion, it is difficult to determine the location of the blockage. Also, the severity of the obstruction will be difficult to determine. This test was done to determine the effect of a blocked system and whether the blockage could be localized by installing pressure sensors at fixed intervals.

Procedure:

1. The glad hand hoses were assembled such that an air compressor was connected at one end and a mechanical pressure gauge was connected at the other end.
2. The glad hand adaptors were connected in between the hoses.

3. The pressure sensors were installed in the appropriate glad hand adaptor. Sensor number 1107800 was installed closer to the air compressor and sensor number 1107806 was installed in the adapter closer to the mechanical gauge.
4. The computer was booted up and the base station was connected to the computer via USB.
5. The X-CTU software was opened up to determine whether the ports were operating
6. The X-CTU was closed and the PS1 Multi App was opened.
7. The serial numbers were inserted then enabled and the parameters were inserted.
8. Part of the hose that was after the first sensor and before the second sensor was inserted into the clamp and tightened.
9. The PS1 Multi App was set to run.
10. Once data was received from the sensor and after waiting for two cycles, the pressure tank was turned on, regulating at 50 PSI.
11. The data was collected and analyzed.

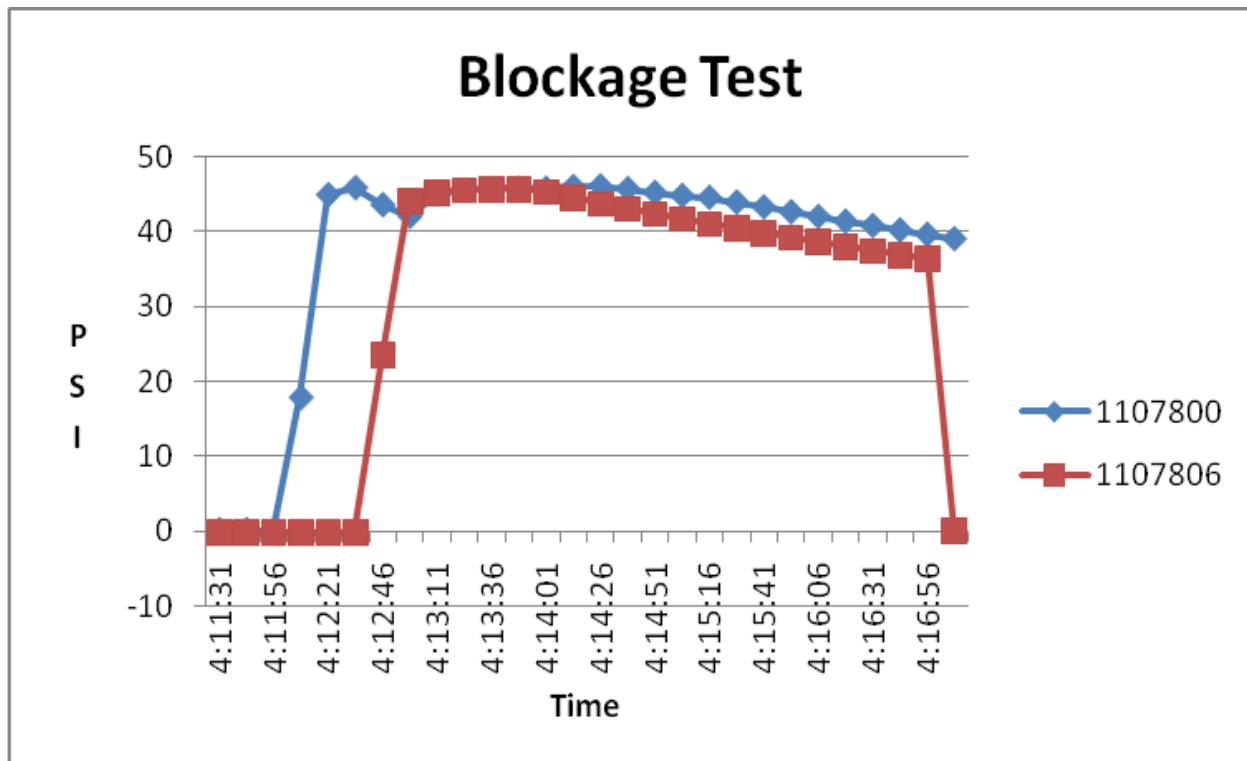


Figure 4.3.1: Plot of obstruction test with leak

Result:

The obstruction was not significant and the entire connection of glad hands and hoses were able to pressurize despite the obstruction. Although the pressure tank was set to regulate at 50 PSI, the pressure in the line varied from about 40 to 45 PSI. This was due to an error in the pressure tank and not due to the obstruction as indicated by the normal reference test. Once the tank was turned off, the pressure dropped faster

4.3.2 Leak Test 1:

Purpose:

The purpose of this test was to simulate what would happen if there was a leakage. Leakage is a major problem because if the pressure becomes too low (below 60), then the brakes on the bogie go off. Currently the only way to detect a leak is to visually observe any cracks or to hear any noise from the leak. The length of the train can be anywhere from a mile to one and a half mile, therefore, it takes a very long time to detect a leak – sometimes over 12 hours. Since the train runs on a tight schedule, any loss of time results in a big expense for the company. Because of this, if the train is in motion and if there are any leaks that cause the brakes to go off on some of the bogies, the train does not stop and travels with part of its brakes on. This test was done in an effort to localize leaks in the system. Since the actual train system is over a mile long, it takes about 45 minutes to pressurize the train. Since we are working with a model that is only ten to fifteen feet in length, an obstruction was created using clamps to reduce the flow of air in an attempt to replicate the actual system.

Procedure:

1. The glad hand hoses were assembled such that an air compressor was connected at one end and a mechanical pressure gauge was connected at the other end.
2. The glad hand adaptors were connected in between the hoses.
3. The pressure sensors were installed in the appropriate glad hand adaptor. Sensor number 1107800 was installed closer to the air compressor and sensor number 1107806 was installed in the adapter closer to the mechanical gauge.
4. A leaky hose was installed between sensor number 1107806 and the pressure gauge.
5. The computer was booted up and the base station was connected to the computer via USB.
6. The X-CTU software was opened up to determine whether the ports were operating

7. The X-CTU was closed and the PS1 Multi App was opened.
8. The serial numbers were inserted then enabled and the parameters were inserted.
9. Part of the hose that was after the first sensor and before the second sensor was inserted into the clamp and tightened.
10. The PS1 Multi App was set to run.
11. Once data was received from the sensor and after waiting for two cycles, the pressure tank was turned on, regulating at 50 PSI.
12. The data was collected and analyzed.

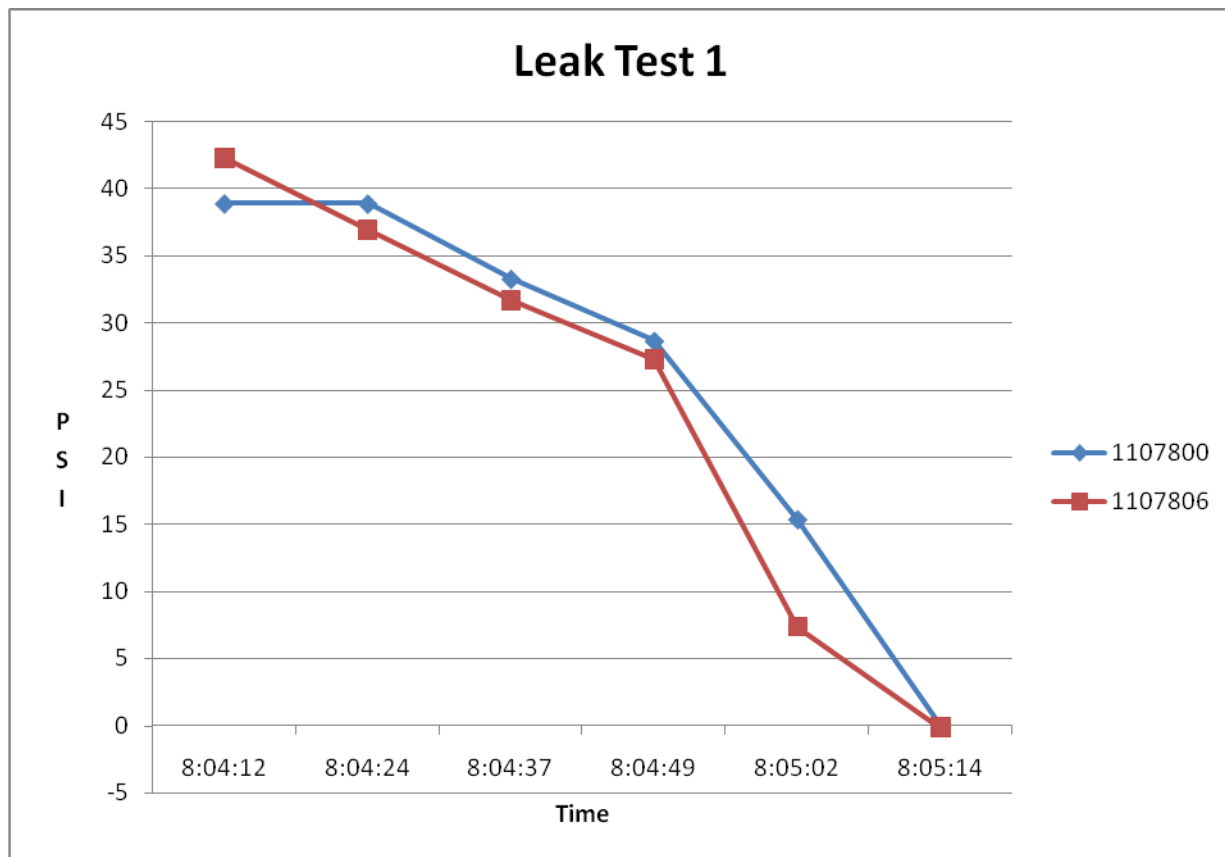


Figure 4.3.2: Plot of Leak Test 1

Result:

The restriction was not significant and the entire connection of glad hands and hoses were able to pressurize despite the restriction and leak. Although the pressure tank was set to regulate at 50 PSI, the pressure in the line varied from about 40 to 45 PSI. This was due to an error in the pressure tank and not due to the restriction or leakage as indicated by the normal reference test. Once the tank was turned off, the pressure dropped faster on the side with the leakage. Considering that the restriction was not significant to begin with, the sensor closer to the leakage (1107806) showed a lower pressure reading consistently from the time the tank was turned off. In this case, the leakage would be localized to the area of pressure sensor 1107806.

4.3.3 Leak Test 2:

Purpose:

The purpose of this test was to simulate what would happen if there was a leakage. Leakage is a major problem because if the pressure becomes too low (below 60), then the brakes on the bogie go off. Currently the only way to detect a leak is to visually observe any cracks or to hear any noise from the leak. The length of the train can be anywhere from a mile to one and a half mile, therefore, it takes a very long time to detect a leak – sometimes over 12 hours. This test is an extension of Leak Test 1. The leakage was sealed using electrician tape and then duct tape on top of it. The restriction was increased by clamping the hose even more. The restriction was varied throughout the test by tightening and loosening the hose. The loosening the tightening process represents bogies that are closer and further away from the sensors, respectively. Near the conclusion of the test, the sealed leakage was exposed. This was done to get results for various degrees of leaks. The sealed leakage represents a small leak while the exposed leakage represents a large leak in the system.

Procedure:

1. The glad hand hoses were assembled such that an air compressor was connected at one end and a mechanical pressure gauge was connected at the other end.
2. The glad hand adaptors were connected in between the hoses.
3. The pressure sensors were installed in the appropriate glad hand adaptor. Sensor number 1107800 was installed closer to the air compressor and sensor number 1107806 was installed in the adapter closer to the mechanical gauge.
4. A leaky hose was installed between sensor number 1107806 and the pressure gauge.
5. The computer was booted up and the base station was connected to the computer via USB.

6. The X-CTU software was opened up to determine whether the ports were operating
7. The X-CTU was closed and the PS1 Multi App was opened.
8. The serial numbers were inserted then enabled and the parameters were inserted.
9. Part of the hose that was after the first sensor and before the second sensor was inserted into the clamp and tightened.
10. The PS1 Multi App was set to run.
11. Once data was received from the sensor and after waiting for two cycles, the pressure tank was turned on, regulating at 50 PSI.
12. The data was collected and analyzed.

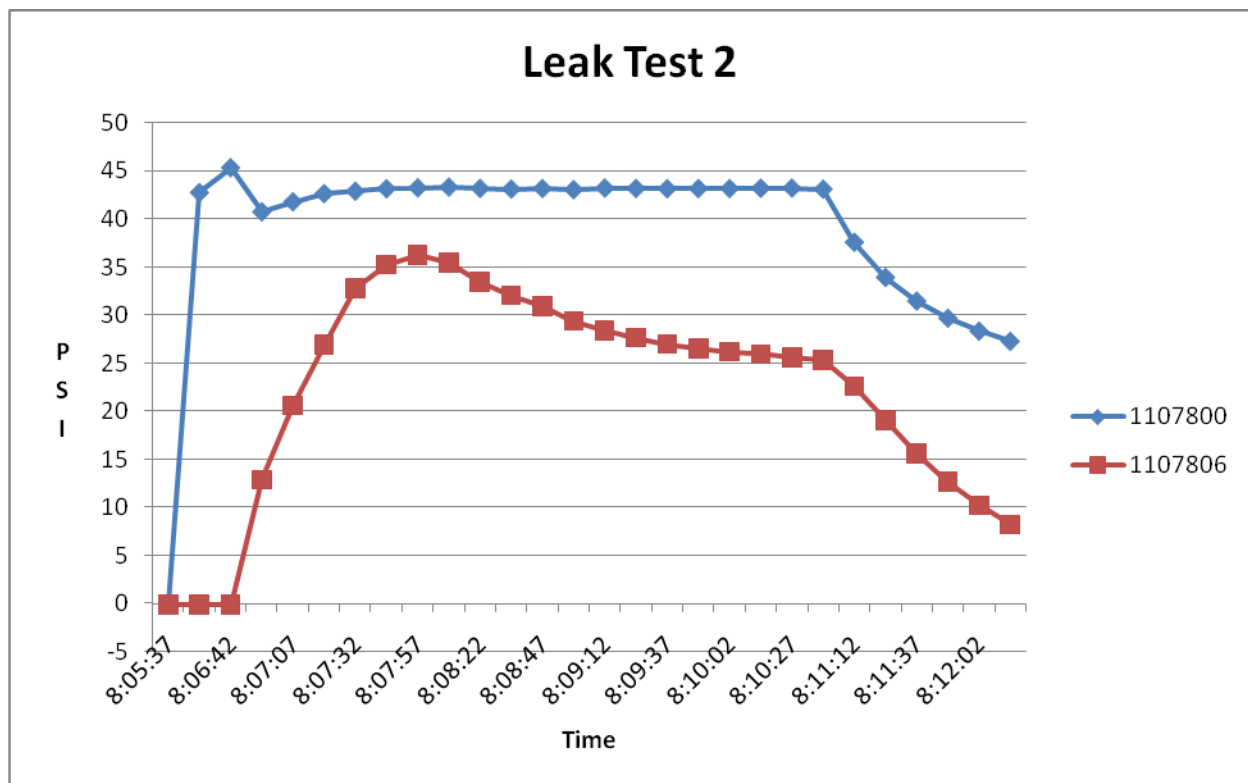


Figure 4.3.3: Plot of Leak Test 2

Result:

Initially the restriction was too big since the area after the restriction did not pressurize as evident by the information from sensor 1107806. Next, the restriction was reduced by loosening the clamp a little. Looking at the graph, this occurred after 8:06:42 where there is a sharp increase in pressure. There is also a decrease in the hose that is before the obstruction. It can be assumed that resulted from the air flowing from the unrestricted area to the newly opened up area rapidly. Shortly afterwards the restriction was increased, although not as much as it was initially. This occurred around 8:07:57 and it can be observed that the pressure after the restriction decreased while the pressure prior to the restriction remained constant. The pressure stabilized close to 25 PSI, where the leakage was equal to the air flow coming into the restricted area. The seal covering the hole was removed between 8:10:12 and 8:11:12. As a result, the leakage was fully exposed. There is a decrease in pressure throughout the whole system; however, the pressure in the restricted region drops much faster than the pressure in the unrestricted area when the entire graph is observed. Therefore, this test concludes that the leakage was present closer to sensor 1107806.

4.3.4 Leak Test 3:

Purpose:

The purpose of this test is to determine where the pressure stabilized when there is a constant leakage and restriction. The leak was covered using duct tape to simulate a small leak since the uncovered leak is too big causing the pressure to drop very low.

Procedure:

1. The glad hand hoses were assembled such that an air compressor was connected at one end and a mechanical pressure gauge was connected at the other end.
2. The glad hand adaptors were connected in between the hoses.
3. The pressure sensors were installed in the appropriate glad hand adaptor. Sensor number 1107800 was installed closer to the air compressor and sensor number 1107806 was installed in the adapter closer to the mechanical gauge.
4. A leaky hose was installed between sensor number 1107806 and the pressure gauge.
5. The computer was booted up and the base station was connected to the computer via USB.
6. The X-CTU software was opened up to determine whether the ports were operating
7. The X-CTU was closed and the PS1 Multi App was opened.
8. The serial numbers were inserted then enabled and the parameters were inserted.
9. Part of the hose that was after the first sensor and before the second sensor was inserted into the clamp and tightened.
10. The PS1 Multi App was set to run.
11. Once data was received from the sensor and after waiting for two cycles, the pressure tank was turned on, regulating at 50 PSI.

12. The data was collected and analyzed.

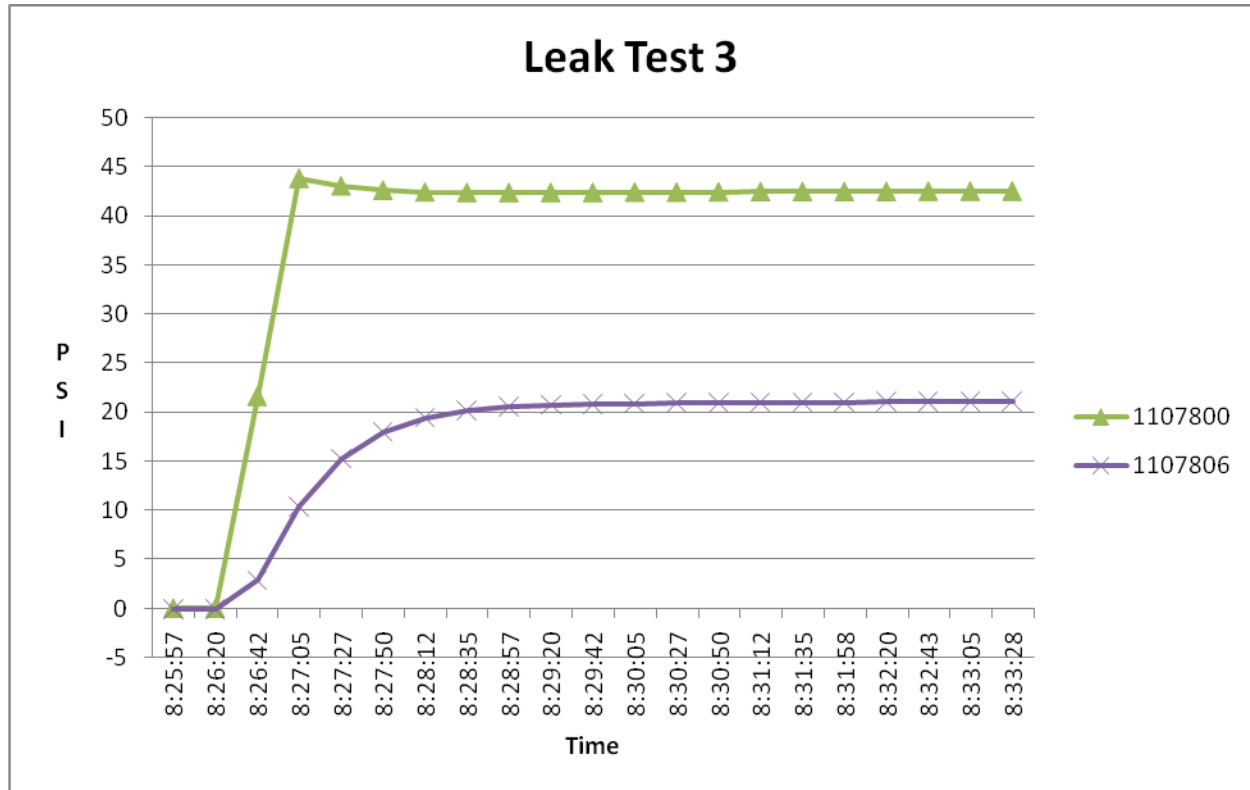


Figure 4.3.4: Plot of Leak Test 3

Result:

The area before the restriction pressurized normally and remained at a constant level (sensor 1107800). The area after the restriction pressurized slowly and stabilized around 22 PSI. This test indicates that in the actual system, when there is a leakage, the pressure might not reach the minimum 75 PSI required before the train can take off. Here, it is evident that the leakage is indeed closer to sensor 1107806 and not sensor 1107800.

4.3.5 Leak Test 4:

Purpose:

This test is similar to the previous test except the leak is fully exposed. After the restricted area reaches equilibrium, the leak will be covered using fingers. This test will simulate pressure changes when the leak is significant. Sealing the hose with finger will simulate how fast the hose will pressurize.

Procedure:

1. The glad hand hoses were assembled such that an air compressor was connected at one end and a mechanical pressure gauge was connected at the other end.
2. The glad hand adaptors were connected in between the hoses.
3. The pressure sensors were installed in the appropriate glad hand adaptor. Sensor number 1107800 was installed closer to the air compressor and sensor number 1107806 was installed in the adapter closer to the mechanical gauge.
4. A leaky hose was installed between sensor number 1107806 and the pressure gauge.
5. The computer was booted up and the base station was connected to the computer via USB.
6. The X-CTU software was opened up to determine whether the ports were operating
7. The X-CTU was closed and the PS1 Multi App was opened.
8. The serial numbers were inserted then enabled and the parameters were inserted.
9. Part of the hose that was after the first sensor and before the second sensor was inserted into the clamp and tightened.
10. The PS1 Multi App was set to run.
11. Once data was received from the sensor and after waiting for two cycles, the pressure tank was turned on, regulating at 50 PSI.
12. The data was collected and analyzed.

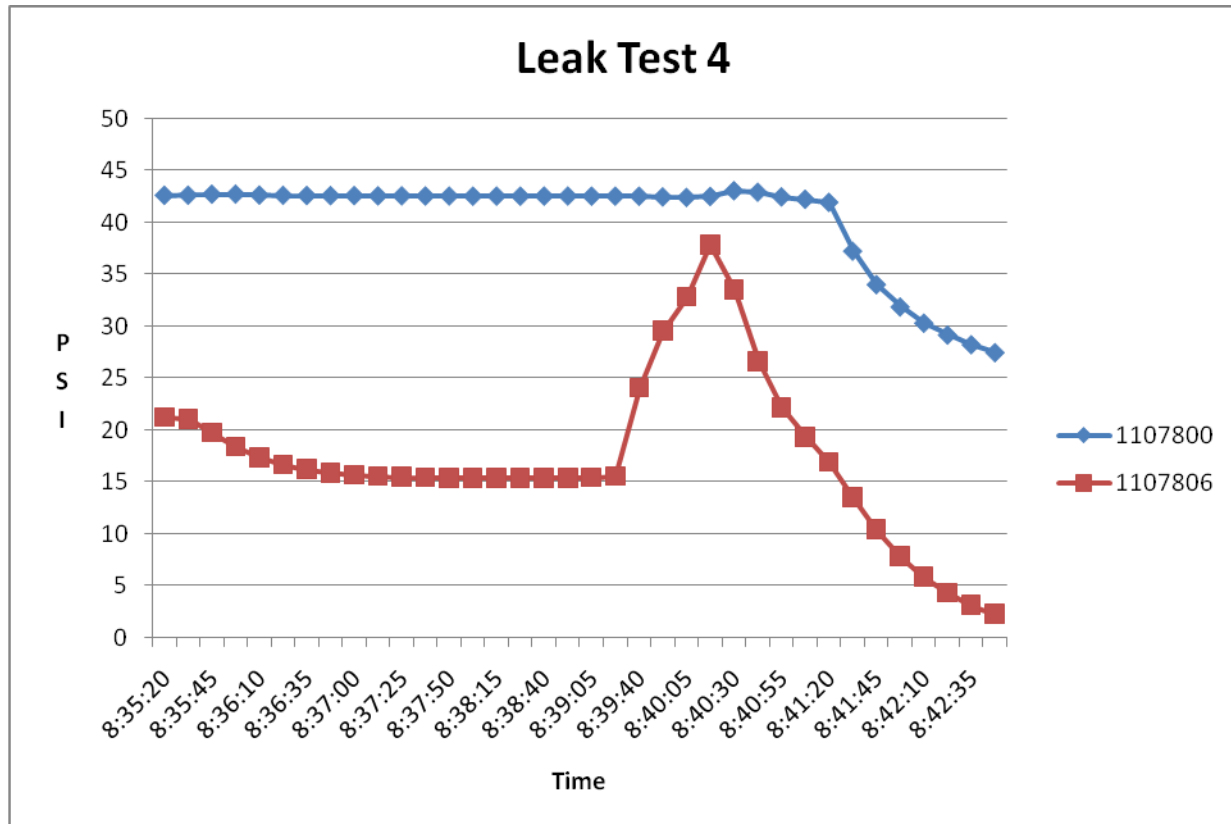


Figure 4.3.5: Plot of Leak Test 5

Result:

Since the leak is significant, the restricted area stabilized at only 15 PSI. The area before restriction remained at a constant 43 PSI. Throughout this whole test, the restriction remained constant. At time 8:39:40, the leakage was plugged using fingers. In few cycles, the pressure was almost the same in the restricted and unrestricted area. As the pressure continued to increase, it became more difficult to plug the hole; therefore, the finger was released and the pressure rapidly decreased. Close to 8:41:20, the pressure tank was turned off and as evident from the previous tests, sensor 1107806 lost pressure much faster than sensor 1107800.

4.3.6 Leak Test 5:

Purpose:

This test is similar to the previous test except it is conducted at 80 PSI. First, the test will be conducted with the leakage uncovered. This will determine how the area after the restriction will pressurize. Next, the restriction is varied and the results observed. The final part of the test is observing the pressure changes once the tank is turned off.

Procedure:

1. The glad hand hoses were assembled such that an air compressor was connected at one end and a mechanical pressure gauge was connected at the other end.
2. The glad hand adaptors were connected in between the hoses.
3. The pressure sensors were installed in the appropriate glad hand adaptor. Sensor number 1107800 was installed closer to the air compressor and sensor number 1107806 was installed in the adapter closer to the mechanical gauge.
4. A leaky hose was installed between sensor number 1107806 and the pressure gauge.
5. The computer was booted up and the base station was connected to the computer via USB.
6. The X-CTU software was opened up to determine whether the ports were operating
7. The X-CTU was closed and the PS1 Multi App was opened.
8. The serial numbers were inserted then enabled and the parameters were inserted.
9. Part of the hose that was after the first sensor and before the second sensor was inserted into the clamp and tightened.
10. The PS1 Multi App was set to run.
11. Once data was received from the sensor and after waiting for two cycles, the pressure tank was turned on, regulating at 80 PSI.

12. The data was collected and analyzed.

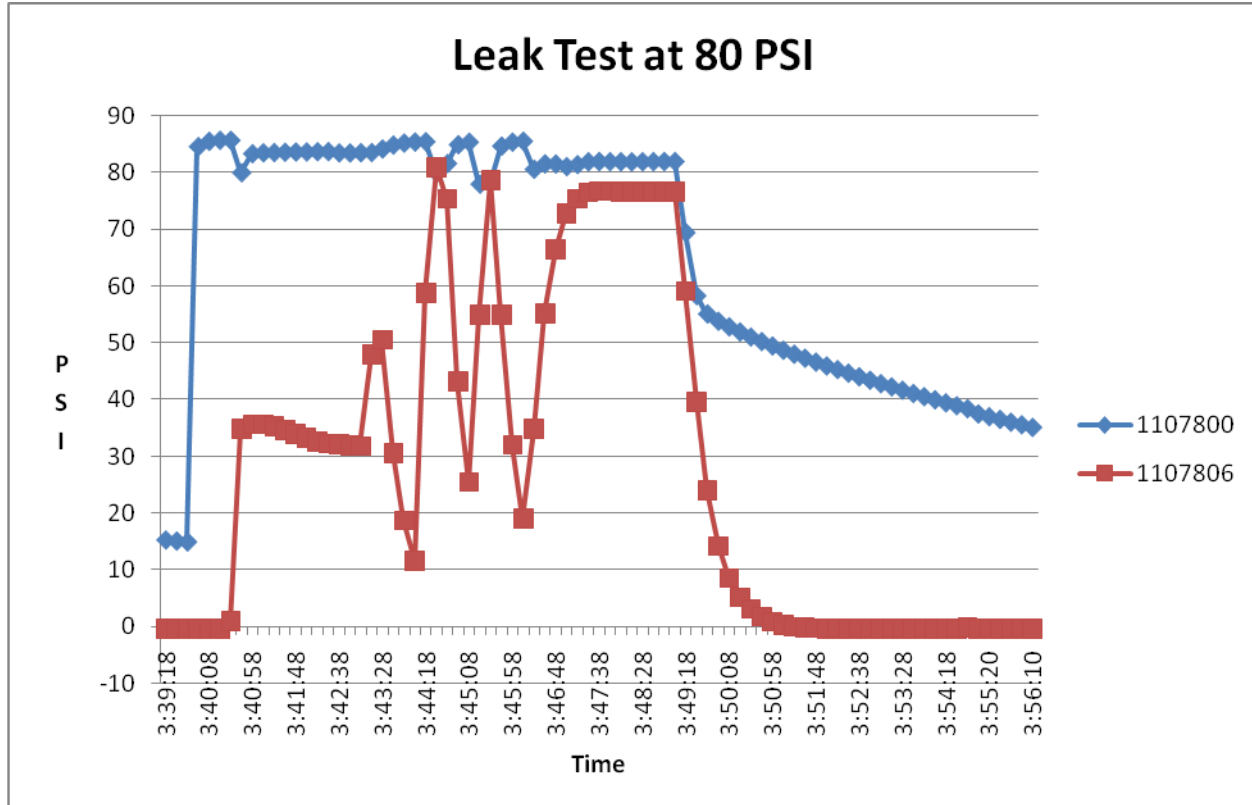


Figure 4.3.6: Plot of Leak Test at 80 psi

Result:

Initially, the restriction was too great; there was no air flowing into the hoses after the restriction. Next the restriction was slightly adjusted to be lower. This resulted in air flowing into the hoses after the restriction. For the next few cycles, the restriction was adjusted and the conclusion was that minor adjustments with the restriction produced rapid change in pressure. At the conclusion of the test, the pressure tank was turned off. What was observed is that the pressure in the restricted area lost pressure much more rapidly than the pressure in the area before the restriction. The conclusion of this test is that at higher pressures, small changes produce a significant change in the

pressure. As indicative of the test, it is difficult to pin point where exactly the obstruction or the leakage will be since pressure propagates rapidly; however, it is very reasonable to localize the leakage to a region considering that the train is over a mile long. At this length, the propagation is delayed and the sensors read data quick enough to the point where it is able to read pressure differences in different areas of the hoses. By observing the pressure differences, the leakage or obstruction can be localized and time can be saved in identifying the location of the leakage.

5. Final Notes

5.1 Summary

ECE 480 Design Team 8's final project design was successful. With the support of Dr. Subir Biswas, Design Team 8 has designed a wireless network of pressure sensors and nearly on schedule. With the exception of wireless communication between pressure transducers, which would've been done if our team hadn't been a victim of false advertisement, all proposed goals have been met. The prototype built could have a substantial impact saving many hours of labor since walking the entire length of a train with more than one hundred cars is extremely time-consuming.

Due to unexpected equipment failure the project was pushed slightly behind schedule. Our schedule, seen in Figure 2.5.3., was maintained up until a few weeks ago. Many deadlines hit and other classes had projects as well. Design Team 8 was able to stay on track though, only falling a few days behind schedule. Spending extra time in the ECE 480 lab working has since realigned the project schedule. Also, due to sound project management, the original plan included strategic slack time and therefore there was no detriment to the delivery date. The final design was completed as scheduled for presentation on Design Day, April 30th

The design of the 'RoadRailer Air Brake Fault Localization' was a success. The product prototype cost of one unit was \$1340. When put into mass production, the cost per-unit this is the cost per unit; it's important to keep in mind that the base station cost is \$300, and the sensor cost is \$995. So, only one base station is needed, and the number of sensors used in the system up to the sponsor. This is slightly higher than budget allocated to the design teams at the start of the semester. However, wireless pressure sensors aren't cheap devices, and after extensive research our team can

guarantee the pressure sensor used in our design is the most affordable device in the market as of right now.

While the project successfully fulfilled many of the Critical Customer Requirements there is room for improvement. Due to false advertisement, our team purchased a sensor that originally was supposed to act as a mesh network. Meaning that, the sensors would communicate with each other acting as repeaters, so their signals would all ultimately reach the base station. However, after we received the third and second sensors, and we were able to test the devices we found out that the hopping feature is not available within in the device. Although our team already has a solution for the problem, there is not enough time to purchase the necessary equipment and run new tests. This would be a good improvement in our project.

5.2 Difficulties and Obstacles

- Electrochem had inaccurate information, bad support and bad service.
- Our device got stepped on the two weeks before design day; a new sensor had to order and by the time of arrival, only one week was left to perform the necessary tests.
- Unable to perform a practical test because the sponsor didn't allow the installation of the device on trains and didn't provide enough information on how the current system works.
- Our device doesn't read data instantaneously which makes it harder to detect leaks on small systems.
- The air compressor regulator wasn't accurate enough.
- When testing with leaks, the leakage was difficult to completely cover.

- Base station and devices were not robust; the reset button on the base station had to be used at the beginning of most tests.
- Unable to open more than one testing application because the program freezes.
- As soon as the device goes out of wireless range, the program freezes.
- The device was too expensive for what it had to offer in return.
- Finding alternative solutions for the wireless range either by using antennas or replacing the chip inside with a newer chip.
- Find a device with a built in diagnostic functions that are able to output an error message whenever the device does not get a reading.

5.3 Conclusion

Team 8, designed a system where pressure transmitters are to be integrated into the air brake system of the RoadRailer trailer cars at regular intervals along the train. The final product includes a successfully designed adaptor for the glad-hand connector hose that attaches to the rail bogie, which was outfitted with an air pressure transducer. Our transmitters, installed into the air brake line, will send the pressure data to a gateway location. This device can save many hours of labor since walking the entire length of a train with more than one hundred cars is extremely time consuming. Our devices will make the system more efficient, as the transmitters will provide precise pressure information and location of the data source. We have designed a sensor system to that operates efficiently in harsh environments of all kinds. The transmitters work flawlessly in temperatures ranging from -20°F Michigan winters to 120°F Florida summers. The data transmits wirelessly, reporting measurements such as pressure and battery levels to a central PC, from bogies located at various points along the train. Also, our back-end software allows the user to set a range for each individual sensor. If the pressure goes out of range a warning is displayed on the screen.

The final product includes a successfully designed adaptor for the glad-hand connector hose that attaches to the rail bogie, and was outfitted it with an air pressure transducer. The data transmits wirelessly, reporting measurements such as pressure and battery levels to a central PC, from bogies located at various points along the train. Our back-end software allows the user to set a range for each individual sensor. If the pressure goes out of range a warning is displayed on the screen.

6. Appendix 1

6.1. Technical Roles and Responsibilities



Design Team 08, from left to right: Dilo Benjamin, Tia Twigg, Marcelo De Castro, Abdulaziz Najm, and Saurav Shrestha

6.1.1 Marcelo de Castro

Marcelo's particular technical responsibility has been to design and built the adaptor for the pressure sensor so our team can connect it to the glad hands, and also to assure that the communication between the sensors and the base station is working properly.

At first, Marcelo had an idea to design an adaptor that would contain a male glad hand on one side and a female glad hand on the other side so that way our team could insert this device into the current system by connecting it in between the bogie and the train using glad hands. This 'housing' would assure the safety of the sensor, which was his main concern. However, his plan changed once our team actually ordered and received the sensor. It was noticed that the circuit inside of the sensor was well protected, and building a 'housing' for the sensor would be not only a waste of time and money, but also unnecessary. That was when Marcelo came up with our team's new and current design which was a much more convenient idea I had to connect the sensor into the air brake system.

A threaded hole was drilled on the top of one of the glad hands, and outfitted with a fitting tube which connects to a 'hose end connector' especially made for pressure applications. The pressure sensor was attached to the hose using a radiator clamp, and at the end of the sensor another 'hose end connector' was attached. In order to get the pressure into the sensor, I connected the two ends of the 'end rose connectors' with a black 'INSTA-GRIP' hose, also especially manufactured for pressure applications. So, now both ends of the device are still connected with glad hand adaptors, that way it can be attached to the current system in use at Triple-Crown Services without having to change it. The prototype has been tested, with 120psi of pressure, which is more than we actually need, and it works perfectly. Meaning, it does not leak.

The second part of Marcelo's technical role was to assure that the sensor and the base station are communicating with each other without any problems. Over Spring break Marcelo decided to run some tests with our pressure transducer and our base station, and found out that the base station was not able to communicate with the sensor. After spending some time with one of the engineers from Electrochem on the phone, he found out that the reset button on the base station wasn't working. On the same day, Marcelo sent it back, and a couple days later he received a new one which now works. That saved our team some time since everything was working fine when everyone came back from Spring break.

Additionally, Marcelo have built the adaptor for the air compressor to be connected into the break line system we're using for testing in lab, since the one we've borrowed from Triple Crown Services needs to be returned. That way our team will have more time to run the tests. Also, he set up the line for testing in the ECE lab, using a nitrogen tank as our air compressor, and a pressure regulator to keep the pressure constant. By keeping the pressure constant in the air break line our team was be able to better test it for leaks and obstructions.

6.1.2 Saurav Shrestha

Saurav Shrestha was in charge of the software portion of the project. A huge portion of this included creating a back-end application in LabVIEW that will work in conjugation with the Multi Test Application software. To do this, he had to teach himself LabVIEW which he accomplished by checking out books from libraries as well as researching online. During the course of his research on writing application note, he found a simple temperature back-end processing application *tempsys.llb* on LabVIEW demo. He studied it in detail to understand why and how it worked. He also wrote his application note on using LabVIEW for simple back-end processing based on his study of this demo.

While writing his application note, he realized that this type of application could be implemented on our project to create a back-end processing application for pressure

readings. He used a small portion of that back-end processing application and researched the topic of enabling it to read incoming pressure data from sensors. After successfully completing this step, he made several improvements to it, which required him to learn more LabVIEW concepts. He was able to pick up those new ideas by working closely with a graduate student, who is proficient in using LabVIEW. After this, the back-end processing application worked flawlessly and as envisioned before. All in all, by taking the responsibility of creating a back-end process in LabVIEW, Saurav gained tremendous knowledge on using this program in a relatively short period of time.

Besides creating back-end processing application, Saurav also worked collaboratively with his fellow team members in various phases of the project. For example, he worked with Marcelo during the process of first testing using an air compressor. By using air compressor, several air brake hoses connected through glad hands and multiple sensors, they were able to create a very similar situation that would be expected in a real field. The success of this experiment showed them that this solution would work well when applied to a real situation.

6.1.3 Tia Twigg

Initial Technical role: Power supply

Final Technical Role: Network meshing and 'hopping'/testing

The first task of our project was to research different sensors to find to right one to fit our design needs. This began with researching the air brakes themselves. I learned that the brakes begin to engage when the pressure in the line drops below 60-65 psi. This was helpful in setting a lower threshold for the tolerances during testing.

The next topic to be addressed was power. Since there is no power available on the trains, except at the locomotive, it was necessary to find an independent power supply. After much discussion, research and consideration it looked like batteries would be the easiest and most logical solution. My research turned up a large number of battery powered pressure sensors that were especially designed to prolong battery life.

Many sensors were considered, but the one that was chosen had what the team thought was all the features that would suit our needs.

Once we settled on the PressureSensorOne from Electrochem, it was my job to work on the distance testing and mesh networking, or 'hopping'. I spent quite a bit of time studying the instruction manual that came with the sensors and base station. I had to carry my hopping research to the internet when I discovered that the information in the manual was insufficient. After downloading and reading several more manuals, including the independent manual for the base station, and the instruction site for the XBee chip, I realized that I was going to have to ask a real person for assistance. Further details of my findings can be found in chapter three under the networking/hopping section. It was discovered that the sensors we had purchased were not capable of the hopping function. This aborted my technical role and I was asked to help Abdul and Dilo with further sensor testing.

During one of our visits to Triple Crown Services, we were able to run a signal distance test in the train yard. It was a very rainy day, so we performed the test from inside a car whilst holding a transmitting sensor out the car window. The base station and the computer to which it was connected remained inside the main office at the station, just like it would be if the system were put into active use. Even in the rain and through walls and traffic and around houses we were able to receive signal for up to one half mile. Although the conditions were not ideal, it is doubtful that the signal distance would improve much with better weather conditions. This is due to the large number of obstacle in the way and the flow of traffic between the sensor and the base station. It is concluded that in order for sensors further than one half miles from the base station to have their data heard, an antenna attached to a relay base station further down the train would need to be implemented.

After we ran the distance test, I helped Dilo and Abdul with the pressure testing in the 480 lab. We decided that it would be reasonable to simulate the slowing of the air pressure propagating down the train through the bogies, with the use of a bench vice grip. When we lowered the air pressure inside the hose from 80 psi to 50 psi, this

worked quite well and we were able to continue with our obstruction and leak tests. Data and results were collected and interpreted. Those results are addressed in chapter three of this paper.

6.1.4 Dilo Benjamin

The technical role that Dilo was responsible for was testing. The purpose of this technical role was to create a system that represented the actual pressure system used at Triple Crown. This was due to the fact that testing was not permitted at Triple Crown due to safety hazards and time constraints since they operate on a tight schedule. In order to conduct the tests, hoses and glad hands were acquired from Triple Crown. Once the glad hands were modified to use the pressure sensors, they were connected to make a long chain that was used to perform all the tests. Next, the air compressor was acquired. It was used to pressurize the hoses so that tests could be performed. Once everything was assembled – the chain of glad hands, hoses, pressure sensors, and air compressor, the tests were performed. Two things were tested, obstruction and leakage. In the actual system, these were the two biggest problems. The obstruction was easily tested and located. It was easy to locate the obstruction when the system was pressurized because the area with the obstruction pressurized much slower. The leak test was a little more difficult because our system was very small, at about 4 meters. The pressure sensors did not display information rapidly enough to accommodate the fast pressure changes that were occurring. Therefore, a restriction was created using clamps to restrict the air flow. Once that restriction was created, it was much easier to notice the pressure changes caused by leaks. The conclusion is that, this product will help localize the problem to the general area, but it will be able to pin point the location within four or five meters.

6.1.5 Abdulaziz Najm

Abdulaziz was involved in several technical roles throughout this project; it started with an intense research on sensors/transducers to choose the best suitable option for the purpose of this project. He worked on installing the base station on the computer and then learning how to communicate (Send/Receive) commands to and from the sensors. His official technical role was Testing / Demo. Him and Dilo worked closely to come up with an accurate representation and try to visualize a model that replicated the one at Triple Crown, he had to come up with many different scenarios to cover all main aspects of leakage and obstructions and a way to localize the problem whenever it occurs. After every test they both worked on analyzing the output and stating whether the result was expected, after that they had to make a decision whether this product would be able to meet the costumer's requirements. Abdulaziz also worked on testing of the wireless range of the devices. Abdulaziz also worked on troubleshooting the devices, whenever there was an error or faulty system the team referred to Abdulaziz and Dilo to solve those errors and get the system working again and the readings consistent.

As for the Demo, Abdulaziz had to design the way the team is going to present their product, since our project is slightly different than other projects in the sense that, due to noise and spatial constraints, we cannot demonstrate all of the applications. The air compressor generates far too much noise for a practical demonstration in a crowded room. Likewise, our sensors have a long range, up to 1.5 km. This makes demonstration of the hopping, or the actual range, problematic. Our solution will consist of filming all of the tests and making a brief movie to show the different applications and scenarios.

During the presentation, we will have a prototype available for the audience to visualize the physical dimensions and connections. Also, we will run the software to produce an output of the normal air pressure in the room we will also be able to change the pressure slightly by blowing into the sensors. The point is to show the accuracy of the sensors and how the pressure is read and measured.

7. Appendix 2

7.1 References

IntelliSensing PressureSensorOne/Two User Manual

Available at the Electrochem website: <http://www.electrochemsolutions.com/index.aspx>

IntelliSensing Wireless Sensor Products user Manual, www.intellisensing.com , 2007

XBee/XBee-PRO Code Development, www.maxstream.net , 2006

X-CTUConfiguration & Test Utility Software,
<http://www.digi.com/support/eservice/login.jsp>, 2008

XBee-PRO PKG-R™ RS-232 RF Modem, www.maxstream.net, 10/08/2005

"LabVIEW Tutorial 1 ". National Instruments. Apr 4, 2010.
<http://www.youtube.com/watch?v=Em5R_RM8E08>.

"LabVIEW Tutorial". Buffalo University. Apr 3, 2010.
<<http://www.ee.buffalo.edu/faculty/paololiu/edtech/roaldi/tutorials/labview.htm>>.

"Front Panel". Connexions. Apr 1, 2010 <<http://cnx.org/content/m12194/latest/>>.

"LabVIEW Tutorial Manual". National Instruments. Mar 30, 2010.
<<http://www.ni.com/pdf/manuals/320998a.pdf>>.

"Lab I: Introduction to the TI DSK6711 DSP". Michigan State University. Mar 30, 2010.
<<http://www.egr.msu.edu/classes/ece480/goodman/ForMiniprojects/Lab1.pdf>>.

"Why use LabVIEW?". stackoverflow. Apr 4, 2010.
<<http://stackoverflow.com/questions/226010/why-use-labview>>.