Final Report: Pipe Mapping via a Pass Through IMU Based System

Benjamin Ask
Department of Electrical and Computer Engineering
Tennessee Technological University
Cookeville, United States
baask42@tntech.edu

Adison Heathcott
Department of Electrical and Computer Engineering
Tennessee Technological University
Cookeville, United States
ayheathcot42@tntech.edu

Caleb Thorne
Department of Electrical and Computer Engineering
Tennessee Technological University
Cookeville, United States
mcthorne42@tntech.edu

Elizabeth Russell
Department of Electrical and Computer Engineering
Tennessee Technological University
Cookeville, United States
errussell42@tntech.edu

***Abstract*—This document is a model and instructions for LaTeX. This and the IEEEtran.cls file define the components of your paper [title, text, heads, etc.]. *CRITICAL: Do Not Use Symbols, Special Characters, Footnotes, or Math in Paper Title or Abstract.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

1) Problem Statement: The main issue in pipe infrastructure is the money and time lost to locating the 'lost' pipes where the original surveys either were not recorded or were destroyed [9] [10]. The 'lost' pipes may malfunction due to a leak, need to be connected to a new system or may be ruptured due to construction that had no pipe on record which could be disastrous if the pipe were to be filled with gas or water. Issues that may result in a ruptured pipe can be as small as a leak to accidentally creating a spark and igniting gas in a pipe [8].

2) Document Summary: This project's goal is to create a probe that can be guided through a length of pipe (20 to 40 feet of pipe) and collect data. This data will then be uploaded to a computer, which will receive, visualize, and plot the collected data to create a map of the pipe system. The subsystems of the resulting design are Power, Storage, Data Analysis, and Data Collection. These systems are atomized in Figure 1, but a short description of said subsystems is as follows: Power – contains the system concerning the supply of power to the system Storage – contains the system concerning the storage of data gathered by the probe Data Analysis – contains the system concerning the calculation on an external device of the data collected Data Collection – contains the system concerning the collection of data from the probe Movement - contains the system concerning the traversal of the probe through the pipe.

3) Expectations: A few specifications and standards for the pipe network that the team will test in are 1"-4" diameter pipes, 90 degree turns only, such as 'elbow joints' no 'T-joints', and empty pipes with no water or gas contained within. Other concerns that may happen are things such as possible damage or error with the collecting of data which the team is planning to solve by having a redundant system to average with the main collection system to solve any error with the data collection. The ideal functioning of this probe and accompanying code will result in a probe that can map pipes in residential areas and will greatly reduce the cost of locating and uncovering said pipes.

A. Salient Outcomes

Our main successes came in the form of the mechanical testing system and the preliminary maps created during testing. We were able to create a fully functional testing system including a motor pulley system, probe casing compatible with said system and PVC pipe network.

B. Report Layout

Our report is organized as stated next: First a review of all the relevant literature, then our teams methodology, then our results and analysis, then a timeline and cost analysis of our project, then the ethical and professional considerations, followed by lessons learned throughout the process and the conclusion, with individual statements at the end of the paper.

II. Literature Review

A. Engineering Standards

The design of this project will adhere to all applicable standards, in both the professional and legal domains. The general PCB design, as well as the selection of its mounted components and peripherals, will adhere to the relevant IPC standards, which are listed below.

- IPC-2221B: Generic Standard on Printed Board Design
- IPC-A-600: Acceptability of Printed Boards

- IPC-A-610: Acceptability of Electronic Assemblies

The IPC standards will constrain this project in multiple ways, as these standards provide guidelines for virtually all aspects of PCB design [11]. Considering that the IPC standards will constrain the general layout of the PCB, such as trace widths, component distancing, and surface geometry, there will be a resulting constraint placed on which components must be selected in order to meet these overall standards. There will also be direct constraints placed on the components, as each component must be selected so that its rated values, such as power ratings, are not exceeded during any stage of the device's operation. The Environmental Protection Agency (EPA) imposes constraints on certain types of batteries via the Mercury-Containing and Rechargeable Battery Management Act. Given that many common coin-cell batteries contain mercury [12], and thus fall within the scope of this law, it is a possible constraint for this project. In the event that this project implements a coin-cell battery anywhere on the PCB, the PCB layout may need to be designed in such a way that the coin-cell battery is easily removable from the device in order to keep it in accordance with the law.

III. Methodology

A. Data Collection Subsystem

1) Microcontroller: The microcontroller is the center of the probe and will be doing the most in terms of computation and power usage for the system. The team believes a low power STM microcontroller and custom board will be the most cost and performance efficient option. The team also has more experience in programming STM microcontrollers specifically, so the development of software should be easier.

Due to the constraints from the peripheral hardware and software the team needs to lay specifications on the microcontroller subsystem. This will assist in finding an adequate microcontroller that will support the needed hardware.

Firstly, the microcontroller must be a 32 bit device to adequately support the 12 to 16 bit data it will be receiving from the peripheral IMU's. It will also need to support memory addressing for the storage subsystem, which may be larger than 16 bits can support easily. Next, the microcontroller must have an adequate number of GPIO pins to support the peripherals and storage hardware. As of now the estimated number of minimum GPIO pins necessary are 50. This will allow extra pins to exist in case of need later and will be supported by a wide variety of microcontrollers. Finally, the microcontroller must allow an external clock source to be input for more accurate clock signals, and must support I2C, SPI, or USART for peripheral communications.

Each of the systems in the data collection subsystem will connect to the microcontroller using the pins of the microcontroller. Most systems will connect through either power or GPIO, however, some systems will connect to special pins on the chip, such as reset. The connections and their relationships will be discussed under each subsystem.

2) Clock: The clock subsystem is important since most microcontrollers do not have stable internal clocks. Most internal clocks are made of resistors and capacitors which can cause the clock signals to be inaccurate and noisy. [6]

Timing is very important for calculating positions, communicating with peripherals, and ensuring the correct amount of power is used. Peripherals such as I2C and SWD need accurate clock signals to function correctly. The clock is also one of the most power consuming parts of a microcontroller and needs to be exact to ensure battery life calculations are accurate.

The teams solution is to connect an external crystal oscillator to the microcontrollers pins to supply a more precise clock signal to the device.

3) Programming: The programming subsystem will allow the team to upload new software to the device much easier during development and testing. Most microcontrollers do not support software uploading systems out of the box, so a system needs to be developed to allow the user to upload new software from a USB connection.

JTAG and SWD are the main standards for communication when it comes to software uploading. SWD uses two GPIO pins, only supports ARM architecture, and supports printing debug information. JTAG uses four GPIO pins and supports multiple architectures including ARM. [7]

The team has decided JTAG will be the most beneficial due to its wide support for architectures. The system will need to have connections to the microcontroller GPIO pins and the USB port.

4) External Communication: The external communication subsystem will take care of power and data from the USB attached to the probe. The circuitry will determine what pins from the USB are sent to the power subsystem, and what data pins are sent to the correct subsystem for retrieval or reprogramming. The physical representation of this on the board will be a USB socket and simple wiring for splitting the input pins.

5) Reset: The device will always be under the threat of software or hardware bugs that could lock up the system. To counter this a reset system will be implemented onto the device. This will give the user a button or switch to interact that will be connected to the reset port on the microcontroller.

Hard reset will always be accomplished by turning the probe off and resetting the power. This subsystem will supply the user a fast and efficient way to reset the software and peripherals without interrupting power supply.

6) State: The system will have multiple different states it can be in for operation. These states need to be defined in ways where the user can see the current state, and set new states.

For this a switch will be used to set the microcontroller and peripheral hardware into whatever state it needs. This will include a state for collecting data, retrieving data, writing new code, and enabling / disabling the device. This system will be connected to the microcontroller through GPIO pins that will tell the software what state to run in.

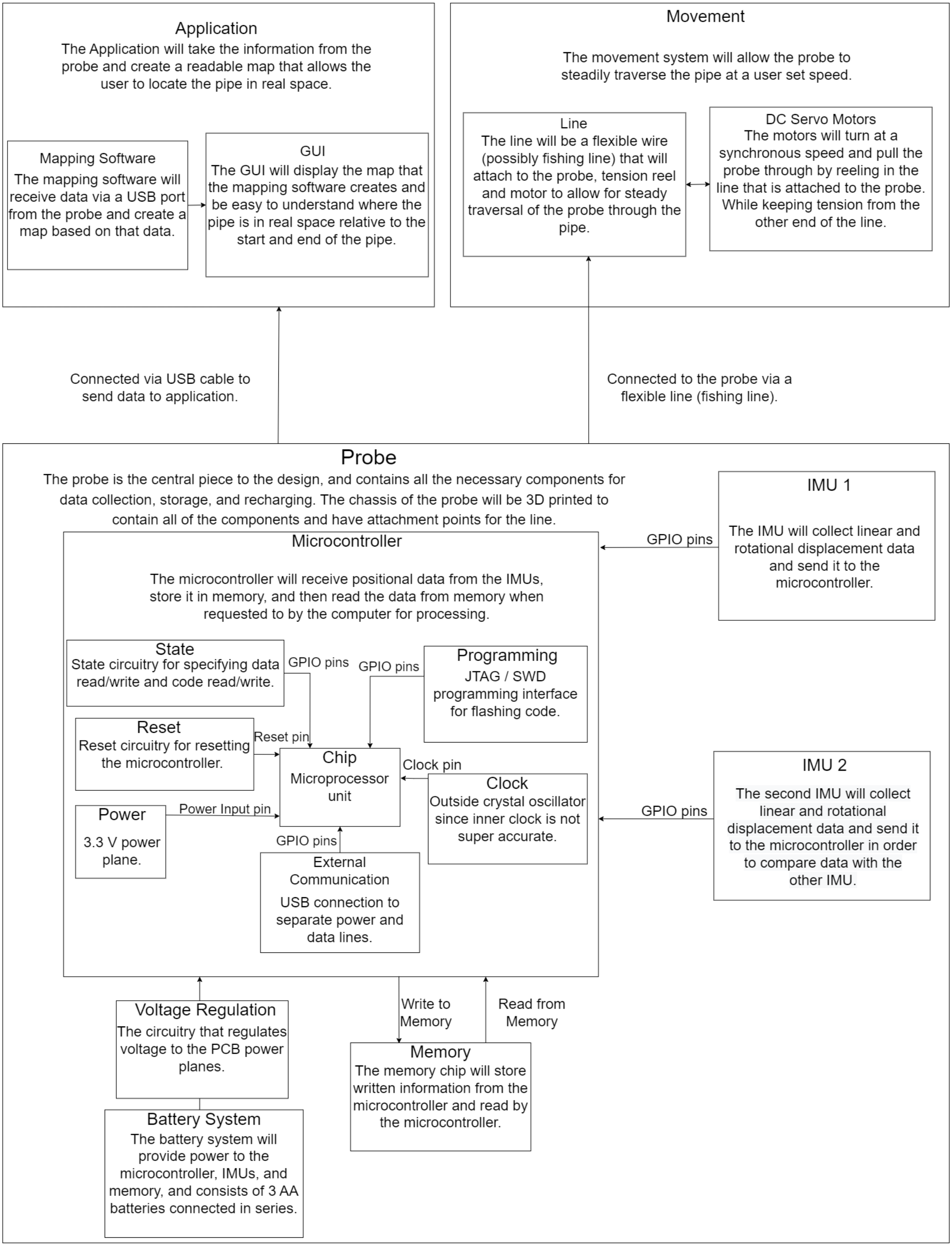

Fig. 1. Block Diagram.

7) Acceleration Measurements: Measuring the acceleration of the device as it traverses the pipe is an integral part of the mapping process. Rather than using discrete accelerometers and gyroscopes, this project will utilize inertial measurement units (IMUs) in order to collect the position data. One component of an IMU is its 3-axis accelerometer, which will be used to measure the device's acceleration through the pipe.

The accelerometer has very few constraints since it creates more than it needs to adhere to. Firstly, most accelerometers have more than adequate sensitivity for the project. Secondly, most accelerometers support I2C, SPI, USART, and temporary data storage which will be supported by almost any microcontroller. Most accelerometers also run at a high enough frequency to have more than enough data points and to minimize the error gathered through the process.

8) Rotation Measurements: Keeping track of the device's rotation is crucial for the task of mapping, as the rotational data will quantify any directional changes that the device makes. This measurement of the device's rotation, which will be taken by the IMUs 3-axis gyroscope, will be used alongside the acceleration data in order to obtain an accurate representation of the area traveled by the device.

While the gyroscope is a separate component it adheres to the same constraints and specifications as the accelerometer since they will be coupled into the same device.

9) Error Measurements: There are a number of errors that can occur when using IMUs [2], though there are existing methods to help mitigate these errors. While most error reduction in this context will be accomplished by proper filtering, using multiple IMUs in tandem has been proven effective [3] in reducing the amount of IMU bias errors with respect to time. With this in mind, the team does intend to implement two IMUs in the design as a form of error prevention.

In regards to combating erroneous measurements via proper filtering, there are a few options that may be useful for this project. Utilizing a Kalman filter will likely be the best option however, as this method is specifically useful for predicting a device's location over time. [4]. Kalman filtering is also well documented as a filtering method for IMUs in particular [5], which makes it a promising method of error correction for this device. This filtering will be done via computer software once the device is connected.

B. Storage Subsystem

1) Memory Chip: The memory chip subsystem will handle the storage and retrieval of the data sensed by the probe. It will need to be able to hold at least 5000 feet of data and keep the data over a reset or power off signal. The memory chip will be connected to the microcontroller through GPIO pins and will be connected to the power subsystem through the power and ground lines.

The memory needs are difficult to predict, however, they can be roughly estimated. The worst case accelerometer data collection will be 1000 Hz with 16 bits of data. The longest pipe the team expects to support is 5000 feet. An estimated 20 minutes will be needed to traverse the pipe. At 20 minutes the device will collect around 10 MB of data. This will be the teams estimated constraints on the memory size for the storage subsystem.

C. Power Subsystem

1) Battery: As this device will be designed to operate without a constant supply of wired power, a rechargeable battery (or series of batteries) will be necessary in order to keep it powered as it traverses the pipe. The battery subsystem will have connections with the voltage regulation, voltage measurement, current measurement, heat measurement, and microcontroller subsystems.

To properly support the hardware the team is planning on getting a battery that at least supports a maximum 3.6 V - 5 V voltage with 1Ah. With the lower amperage needs of all the devices 1Ah should suffice and will be decently easy to acquire.

2) Voltage Regulation: The voltage regulation subsystem is responsible for regulating the battery voltage to the device while it is powered on, supplying the appropriate voltage to each of the other connected subsystems. This voltage regulation will be accomplished via a 3.3V LDO for powering the microcontroller, and a 1.8V LDO for powering the IMUs and memory. This subsystem will be connected to the battery subsystem, and the PCB power planes

D. Data Analysis Subsystem

To minimize the work and complexity of the microprocessor, the job of calculating mapping and visual data has been moved to an external program. The program will acquire data from the device using a USB and will use various programs to convert the acceleration and rotation data into 3D position data. It will also calculate and minimize error with software filters.

1) Mapping: The mapping application will be implemented on MATLAB due to its helpful libraries and strong computational power. The program will first acquire the acceleration and rotation data, convert them into 3D positional data, and filter out any noise or drift caused by integration. Then the points can be converted and charted to show distance, lengths, and other important information for location.

2) Filtering: A large part of the application will be filtering out extra noise produced by the variability of the gyroscopes and accelerometers. For this, the team will use software Kalman filters in either Matlab or Simulink.

3) Visualizing: The visualization application will be implemented on both MATLAB and any 3D model program possible, such as 3D viewer on Microsoft Windows. The program will first take the 3D data and convert this data into vertices that can be rendered to visibly show the pipe layout. These vertices will give users the ability to view to 3D pipe network using any program that will render obj files.

IV. Constraints, Specifications, and Analytical Verification

A. Power

The power subsystem must be capable of supplying 3.6V to 5V to the device, as this should be sufficient for powering the microcontroller as well as its attached peripherals. In addition, the battery must be able to store at least 1 Ah of charge, which should be sufficient for the low-power needs of an STM microcontroller and the sensors. Many STM microcontrollers and IMUs come equipped with various options for power-saving modes, which will also help to conserve battery usage on these already low-power devices. This battery must also be rechargeable, which will place a constraint on what type of batteries can be used for this project. Finally, as the device will begin recharging when plugged into a computer, the power subsystem will need to be capable of receiving input power via USB ports of a computer. Depending on what version the USB ports are (2.0 or 3.0), the input voltage to the power subsystem will differ, which means that the power subsystem will need to be capable of accepting a range of possible input voltages.

In order to verify that the specifications of the power subsystem are met, analysis may be performed via simulation software, such as LTSpice or Multisim, as well as via manual calculation using principles of circuit design.

B. Storage

The storage subsystem shares a mutual constraint with the microcontroller, as the memory chip will need to exchange information with the microcontroller at the appropriate times in the clock cycle. In order to meet the specifications defined in Section II.B of this document, the memory chip will be capable of storing 100 MB at minimum. In order for the memory chip to retain the data when the device is powered off or reset, as also specified in Section II.B, this subsystem will need to be designed with a nonvolatile variety of memory. The team believes that using a flash memory chip will be a suitable option to meet this design specification. This subsystem, like most of the other subsystems, will also be constrained by the physical size restriction placed on the device as a whole (being small enough to traverse a 1” to 4” diameter pipe). The memory chip chosen must be as small as possible, while still meeting all of the above specifications.

The memory subsystem may be analytically tested via hand calculations. The team will verify that the memory chip will be suitable with calculations based on the microprocessor clock speed, read and write times of the chip in question, power requirements of the chip, and any other relevant information that needs to be considered.

C. Microcontroller

The microcontroller will need to be a 32 bit system, since the IMU output data is typically either 12 or 16 bits. In order to have enough pins for interfacing with all of the peripherals, the microcontroller must also possess a minimum of 14 GPIO pins, as well as the ability to support JTAG and I2C communication protocols. The microcontroller must also be capable of connection to an external clock generator, as the microcontroller design will implement an external crystal oscillator to aid in maintaining clock precision. The microprocessor in particular will also have constraints placed on it by the IMUs and memory chip, as the clock frequency of the processor must be high enough to support all of these peripherals while maintaining a suitable sampling rate for these devices. The microcontroller must also fit within the maximum footprint of the device, which will need to be small enough to traverse a pipe as small as 1” in diameter.

The microcontroller design will be analytically tested via a suitable emulator with STM32 support, such as Qemu.

D. IMUs

In order to ensure the highest data resolution possible from both of the IMUs, the team will select a device that provides 16 bit output resolution from both the accelerometer and gyroscope components. The IMUs must also draw a low amount of power, on the scale of milliamps, in order to ensure that the power subsystem is capable of supplying enough power to all device subsystems. The IMUs will also be constrained by the size specifications mentioned in the previous sections.

The IMUs will be analytically tested via Matlab and/or Simulink, as there are a variety of existing methods for IMU simulation on those platforms.

V. Results

A. Experiment 1

The first experimentation we did was on the mechanical subsystem which includes the motors, pipe network, and spool with fish line. The completed experiment consisted of a ”dummy” battery holder shell being pulled through the 24 foot long 2 inch diameter PVC network. While the probe was being pulled through we kept a running graph of the instantaneous motor velocity and found the largest difference between the desired velocity and the actual velocity of the motor (data shown below). Screenshots of the graphs have been provided below to show how the data was collected.

Motor Velocity Deviation Table

Run Number	Set Velocity (ft/sec)	Max Velocity Deviation (ft/sec)
1	1.39	0.016
2	1.36	0.060
3	1.34	0.038
4	1.41	0.019
5	1.27	0.052

B. Analysis & recommended Improvements

One problem that we encountered with this system is that while the battery shell easily traversed the network, the PCB shell often got stuck in the 2” 90 degree elbows due to a small lip that was not forseen in our initial design. Our next step is to redesign the PCB shell to have a more gradual edge rather than being very nearly a perfect cylinder. It is likely that both the battery shell and PCB shell would fit in a larger diameter PVC system but our cost for testing would have had to be much higher to confirm that theory. What this experiment verified was that our mechanical subsystem pulls at or above our minimum velocity (1 ft/sec) and pulls consistently with very little deviation so as to minimize the error during data collection. While we did not have a specific number constraining our velocity deviation we found that the deviation was always less than 0.1 ft/sec in each of our five tests which means if we run our probe around 1 ft/sec then we will never exceed our maximum velocity of 2 ft/sec which was calculated based on our sampling rate. Therefore this system successfully remained within the constraints while meeting specifications.

C. Experiment 2

The second experiment was designed to test the accuracy of the probe and to see if the data measured produced data within our specifications. Our constraints define a successful plot to be within a 1" radius of the measured pipes edge.

For the following iterations of the experiment, it was not possible to measure the probes location from inside a 2" pipe. Instead, the probe was run over four separate paths defined by duct tape on the floor and walls. For each of the experiments it is assumed that the start location of the probe is (0, 0, 0) on a 3D axis defined as (x, y, z) and the probe is facing the positive y direction.

1) Experiment Iteration 1: For the first iteration the tape line is plotted as the orange line on the graphs, however, a text description is included here as well. The first run consisted of a straight line exactly 6.14 meters long. The second run consisted of a straight line for 3.35 meters, a right turn, and another straight line for 3 meters. The third run consisted of a straight line for 2 meters, a turn upward, and a straight line for 4.14 meters. The fourth run consisted of a straight line exactly 2 meters in length.

The graph below shows the network number and run number for each experiment, the sample rate the data was captured at, the threshold value set in the software, and the final point of the measured data minus the final point of the baseline. The threshold value is set in the software and defines the minimum normalized acceleration that would constitute the probe as "moving". The difference of the final points is represented as (x, y, z) with each value being the distance between the two points on their respective axis.

Pipe Mapping Results

It is difficult to represent the deviation through the path without having tables full of thousands of data points, so the graphs below show the baseline (orange) and the measured line (blue). The endpoints have been measured to give some metric of how far off the final values are, but the intermediate values need to be evaluated graphically. Note in some of the graphed simulations the orange line is difficult to see but it is there.

2) Experiment Iteration 2: For the second iteration the same data from iteration 1 was used. The first iteration shows

Network - Run Number	Threshold	Endpoint Error (meters)
1 - 1	0.1	(7.05, 6.16, 2.32)
1 - 2	0.1	(4.98, 7.22, 1.02)
1 - 3	0.1	(5.54, 1.18, 2.22)
2 - 1	0.1	(0.41, 0.50, 2.72)
2 - 2	0.1	(13.0, 21.4, 8.72)
2 - 3	0.1	(13.3, 4.71, 4.76)
3 - 1	0.1	(97.1, 5.70, 50.3)
3 - 2	0.1	(10.7, 0.67, 2.18)
3 - 3	0.1	(27.4, 4.14, 29.8)
4 - 1	0.1	(87.9, 199.4, 612.3)
4 - 2	0.1	(24.4, 73.1, 214.1)
4 - 3	0.1	(18.3, 108.4, 183.8)

a large amount of variability in the final measurements from endpoints and does not always follow the baseline path shown in the graphs. Our first assumption is to test the gyroscope to see if this is the basis of our errors.

Below are graphs and table entries of Pipe 1 and Pipe 2 with the gyroscope data set to 0.

Network - Run Number	Threshold	Endpoint Error (meters)
1 - 1	0.1	(5.80, 2.44, 2.30)
1 - 2	0.1	(5.92, 1.38, 1.16)
1 - 3	0.1	(5.96, 3.09, 2.26)
2 - 1	0.1	(4.28, 2.24, 2.64)
2 - 2	0.1	(6.15, 1.07, 1.70)
2 - 3	0.1	(2.76, 4.54, 4.93)

Even though our endpoints are closer, the data does not follow the path of the baseline. This led the team to believe that the gyroscope should not be the issue. However, we are not ruling out the possibility.

3) Experiment Iteration 3: For the third iteration the team wanted to test whether the speed and acceleration had anything to do with our skewed results. For this a new set of data was generated with new experimentation. The probe was moved along new and old paths similarly to the methods before, except the speed was increased. Initially, the speed was around 0.8 feet per second. The new speed is around 2.5 - 3 feet per second. These values are estimated from the time and distance covered by the retrieved data, we do not have the means to travel at an exact speed since our system can not be integrated with the mechanical system.

The first pipe was moved along a straight line 9.144 meters in length. The second pipe was moved along the path of pipe 2 in the initial experiments, 3.35 meters straight and 3 meters to the right. For this iteration the gain value was adjusted to get better results. The gain value controls the strength of the gyroscope based error correction on the system.

Network - Run Number	Threshold	Endpoint Error (meters)
1 - 1	0.21	(0.63, 1.60, 1.76)
1 - 2	0.21	(2.63, 1.94, 1.08)
1 - 3	0.21	(2.70, 2.00, 1.10)
2 - 1	0.12	(0.94, 5.38, 2.65)
2 - 2	0.2	(1.73, 1.86, 1.73)
2 - 3	0.2	(3.01, 1.05, 0.217)

These results look closer to the baseline, however, they are not perfect. This increase in speed has shown to give better data, most likely due to more pronounced data in the IMU.

Most experiments using this new iteration turned out similarly to those shown above. The team believes that if the speed was more consistent throughout the experiment the data would be even closer to the baseline.

4) Experiment Iteration 4: In the third iteration we increased the speed to obtain better results. Increasing the speed means a faster experiment time, which in turn means less data points in the experiment. Due to this we decided to increase the sampling rate to see if the data would be more defined.

If you increase the sampling rate, you increase the amount of data obtained in the same amount of time. This data then has to be written to the memory for storage. This in turn poses a trade-off which we had to consider. The faster the sampling rate, the longer the memory would take to write, and the more gaps we would have in the data. The slower the sampling rate, the shorter the memory would take to write, and the more continuous the data would look. Due to this we settled on doubling the sampling rate from 26 Hz to 52 Hz.

Network - Run Number	Threshold	Endpoint Error (meters)
1 - 1	0.55	(3.75, 2.92, 0.95)
1 - 2	0.35	(1.00, 2.04, 0.75)
1 - 3	0.45	(2.70, 2.00, 1.10)

These results show that the increased sampling rate helped with more accurate turns. However, it still wasn't fast enough to show any increased accuracy in the measurements. As stated above, changing the sampling rate provides a trade-off. The team feels as if 52 Hz was the last possible rating for the IMU that would give decently accurate data without adding large holes in the sampling.

5) Experiment Iteration 5: The fifth experiment was used to test if the 'free fall' of the IMU would cause errors in the measurements. For this we held the probe 3 feet in the air and dropped it suddenly. Measurements were taken for a short time before and after the fall to measure the baseline of no movement.

Network - Run Number	Threshold	Endpoint Error (meters)
1 - 1	0.1	(0.022, 0.55, 0.46)
1 - 2	0.25	(0.009, 0.045, 0.088)

This experiment showed that even though the drop was not always consistent and correct, it shouldn't be causing any large error in our calculations. The errors seen were predicated as the IMU was not expected to always give accurate results in free fall. We had hoped to have more runs and larger heights, however, concern over damaging the probe caused us to end experimentation fairly quickly.

We performed this experiment to test if our previous assumptions of free fall were true and to test the scale of impact. Previous experiments had no reason to be affected by free fall since none of the were dropped uncontrolled. We are now confident that the current system we have employed will handle downward motion adequately and that the chance of free fall will not cause any large errors.

6) Experiment Iteration 6: The sixth experiment was used to test the baseline of no movement. This could be used to see if the IMU had any internal errors or if the software was interpreting still movement incorrectly.

Network - Run Number	Threshold	Endpoint Error (meters)
1 - 1	0.3	(0.0015, 0.002, 0.0005)
1 - 2	0.3	(0.0015, 0.002, 0.0001)

The experiment results show that the still movement gives accurate results and that the error is extremely minimal. It is difficult to see the baseline in the graphs but they are present at (0, 0, 0).

7) Experiment Iteration 7: The seventh experiment was used to test if the constant speed of the motors would give better results. The experiment itself had some minor problems that were unavoidable. Firstly, the path was only able to be set up in a short straight line. We were only able to mount the motors to the table so the track could only be the length of the table, 2 meters.

Secondly, the probe had to be put on a box that was then pulled to keep the probe, and the wires safe. The extra friction and caused the probe to move differently than it would have through the pipe in the casing. Also the motor spools needed room to be able to spin so the edge of the table was utilized. This also caused the box to need extra stabilization which caused some tilts in the readings.

The experiment was engineered to be the best it could given the circumstances, but it is not perfect.

Network - Run Number	Threshold	Endpoint Error (meters)
1 - 1	0.4	(2.71, 1.45, 1.42)
1 - 2	0.4	(1.99, 0.99, 0.22)

The results were not what the team was expecting but they still seem promising. Due to the experimentation problems the results were predicted to come out with some error. This level of error was not predicted but is explainable.

Again, the team feels that if the correct hardware and experimentation parameters were met, the results would be much more accurate.

8) Analysis and Recommended Improvements: Our initial experimentation showed large amounts of variability in the final measurements from the endpoints and didn't always follow the baseline path. Error was to be expected initially, but not to this degree. We conducted further experiments to isolate the cause of the error. In these further experiments, the gyroscope was ruled out as a culprit and the speed was brought to light. The later experiments still show error in the recordings but we think we have plausible evidence on what is causing the error.

Firstly, the state of the current drone is causing some of the error, or inability to prevent error, that is difficult to mitigate without redesign. The size and fragility of the current probe keeps it from being tested in its originally designed fashion. The probe cannot fit in the pipe since the wires and extra hardware takes up much more space than would fit inside the pipe. The fragility of the wires also prevents us from testing the system with the mechanical subsystem, as the wires are too delicate to risk such an experiment.

Our second source of error is very similar. Due to the fact that we cannot use the mechanical subsystem in the current design, we cannot move the drone at a consistent speed. Experiment iteration 3 shows that increasing the speed gave better results and the team believes a more constant speed would also help after further investigation into the software.

We have also come to the conclusion that we need to increase the speed of the probe. This is possible, however, we have selected the current speed to be the least damaging to the probe and the pipe. The redesign would need to ensure that the pipe could endure faster speed than previously calculated and that the new casing would be designed to safely travel at those higher speeds.

The last cause of error, is due to the software using dead reckoning to map the data points. Dead reckoning is at its base an estimation to be able to map a path using the data points, this error will always exist in our program. Dead reckoning uses the position measured and velocity to plot your position.

Dead reckoning (originally called deduced reckoning) will always introduce slight error into the system, one reason is due to the fact that to get the velocity needed for the position calculation, you must integrate the acceleration which adds an unknown C constant. Then to get a position from that velocity it must be integrated again which changes the equation to end in Cx+D where C and D are unknowns. This C and D introduce error, as the number of integrations increases over time to gain the needed positions the error will grow.

There are two ways to mitigate this error, one is to zero out the error every so often when the software is run and the second is to have a secondary unit such as a GPS, LIDAR, or external sensor to gain the true location so that the error can be calculated out.

Unfortunately, as we are only using data from 6-axis IMUs dead reckoning is the only calculation that was feasible, as our group is only using an accelerometer and gyroscope to gather data. Another fix would be to get a higher quality IMU where it has a secondary unit included. Our code does have error mitigation already added, so to properly lowered the error to near zero we would need to utilize a secondary sensor.

Our solution was originally planned to mitigate some of these errors. However, as we faced many difficulties during implementation, we had to forego components and methods that were going to assist in reducing this error. Due to this we do not meet our 1" goal for the error range, however, we do have ideas on ways to mitigate this error and design a better system.

Firstly, having the second IMU to check measurements and give better filtering data would be extremely helpful and is already implemented in the redesign. Secondly, having the probe fit inside the casing, and thus the pipe, would hopefully give the sensors a more stable and constant environment to record data in. We could mitigate a lot of sudden movements, accidental offsets, and we could keep the speed much more constant. Finally, having secondary systems, like a GPS or magnetometer, would help tremendously since IMUs have difficulty measuring the incline/decline in a path and are sensitive to sudden changes in motion.

Even though the team was not able to get the data within the margins of success, we believe we have determined the cause of the error and have hypothesised on possible fixes to this error. We have taken individual tests of all the subsystems, software, and hardware pieces. The memory is tested in Experiment 3 and shows perfect execution. The software has been tested with example data and shows to be a promising asset. Finally, the IMU is tested in this experiment, however, it is difficult to break down the system any further than we already have.

We believe that the best way to fix the error in our system would be to move forward with the redesign and conduct the original experiments in the way they were first imagined, with one small change. We would like to run the experiments with varying speeds to truly test what constant speed would give the best results before the sampling rate would need to be increased.

D. Experiment 3

The third experiment was designed to test the capabilities of the memory subsystem. We used this experiment to validate that the memory would be able to hold up to 5000 feet or 20 minutes worth of data, and to keep the data when the device was powered off and on. The experiment consisted of running the device for various time lengths, measuring the amount of data that was stored in the memory, verifying that the memory had not overflowed, and verifying that the data was still intact and unmodified.

We were able to use the timestamp provided by the IMU to measure the exact amount of time data was measured. The IMU was running at 26 Hz and polled both the accelerometer, gyroscope, and timestamp. Each data entry consisted of 7 bytes, 1 tag byte and 6 data bytes. Since the data rate and size were constant the data looks exactly as predicted.

Run Time	Bytes Stored	Bytes Retrieved	Bytes Remaining
1 minute	32,760	32,760	16,744,456
5 minutes	163,800	163,800	16,613,416
10 minutes	327,600	327,600	16,449,616
15 minutes	491,400	491,400	16,285,816
20 minutes	655,200	655,200	16,122,016

1) Analysis and Recommended Improvements: The memory subsystem was easily able to collect data continuously for 20 minutes which is well above what it should ever actually have to collect in a real application. Recommended improvements are purely to increase the performance of the system and convenience of the programmer. The only improvement that the team could find was to change the memory to be able to be written to without being erased. The current memory has to be erased every time you want to change the data in a byte so this caused a bit of a performance hit on the system. It was not enough to change the data but if we sampled at a higher frequency, it would have caused a problem.

E. Experiment 4

The fourth experiment focused on verifying that the device's battery life was within the specification defined within previous documents. This specification stated that the batteries must last long enough for the probe to collect 20 minutes worth of data at minimum. For this experiment, the device was powered only via the series connection of 3 AA batteries connected to the Vin and GND pins of the Arduino Nano. Load current was measured using a DMM, and then the initial battery voltage was measured. After recording the initial battery voltage, the device was allowed to continuously poll data for 20 minutes uninterrupted. After the 20 minutes were up, the battery voltage was measured again. This test was performed 3 times. The data is shown in the table below.

Test	Run Time	Initial Voltage	Post-Test Voltage	Load Current
1	20 minutes	5.08 V	5.03 V	15-18 mA
2	20 minutes	5.03 V	4.98 V	15-18 mA
3	20 minutes	4.98 V	4.92 V	15-18 mA

Given the initial specification of 20 minutes minimum run time, this experiment verifies that the device comfortably meets our requirements. It should be noted that due to the dropout voltage of the 3.3V LDO, the device will only remain powered until the total battery voltage reaches around 3.67V. However, even though the device will only be functional until the battery voltage approaches this threshold, it is clear that the device exceeds our minimum run time requirement. Given the load current in the table above, and the manufacturer discharge curve for the batteries used in this experiment (Duracell Coppertop AA's), we can estimate the battery life of the device to be around 50-100 hours when using these batteries.

VI. Timeline and Cost

VII. Ethical and Professional Responsibilities

A. Ethical Considerations

There are some considerations to be made regarding ethical design choices in this project, and the team has attempted to address all of them throughout the design process thus far. The primary ethical considerations for this project include the following.

1) Device Construction: The outermost surface of the device must be constructed in such a manner that it will neither damage, nor introduce potentially unwanted contaminants into the pipe network, whether during normal operation or in a failure case. This means that special care will need to be taken in regards to choosing the material making up the outermost surfaces, as well as properly protecting any potentially harmful chemicals contained within the device, such as battery acid.

2) Device Lost/Stuck in Pipe: The possibility that the device could become, lost, stuck, or otherwise irretrievable from the pipe network is also an ethical consideration, as this event could potentially result in negative consequences affecting humans, structures, and/or the environment in certain failure cases. If the device gets stuck in an empty water pipe, and water begins to flow into the pipe before the device is removed, the blockage caused by the device could lead to pressure buildup that is excessive enough to cause the pipe to rupture. In this case, there could be direct harm caused to any individuals within close proximity to the ruptured pipe, as well as the surrounding structure. This break in the line would also prevent water from flowing to its intended destination, which could result in an outcome ranging anywhere from inconvenient to catastrophic, depending on what the water source was supplying. Depending on the chemical contents of the leaking water, (i.e., wastewater or drinking water), the burst pipe could also present an environmental hazard. In the case that the pipe carried gas rather than water, the pressure buildup resulting from a blockage caused by the device could result in a rupture of the pipe, which could then result in ignition of gas present in the line, depending on the properties of the gas.

With this being said, it is critical that the device be designed and constructed in such a manner that the probability of the device becoming stuck inside the pipe is minimized. The device will be designed with a minimal footprint, also taking into account the reduction in space that will occur in the bends of the pipe, such as 90 degree elbows.

VIII. Lessons Learned

* We ran into a time crunch due to shipping delays. In the future we would definitely prioritize early ordering of parts. * During assembly we had a key part (our microcontroller) burn out. This caused us to have to completely reconfigure our design. In the future we would either negotiate to order double what is needed, or plan a backup part if the chip/part is defective. * When at all possible, avoid using leadless IC's when designing a surface-mount PCB. Without the right equipment, it is incredibly challenging to solder correctly. * We found that our leadless SMD components were extremely difficult to solder. In the future we would make sure to get a stencil for reflow soldering the board. * We found that our error was much larger in the mapping than expected and it turned out that we likely need another component e.g Lidar, magnetometer or GPS to decrease this error. In the future we would add more systems to increase mapping accuracy and in general be wary of the accuracy of a single IMU with little filtering. The team determined that using a set of 9-axis IMUs rather than the 6-axis set we used would greatly assist in keeping the gyroscope measurements accurate.

IX. Conclusion

In conclusion we as a team have accomplished a great deal. We were able to create a working test setup for the mechanical side of things (Probe casing being run through the PVC). We were also able to build a fully operational hardware setup for testing. We were able to generate 3D maps from our testing. For future work we would most like to improve upon the PCB and the mapping accuracy. We have a fully ready PCB rework and have many ideas on how to increase the accuracy of our mapping software.

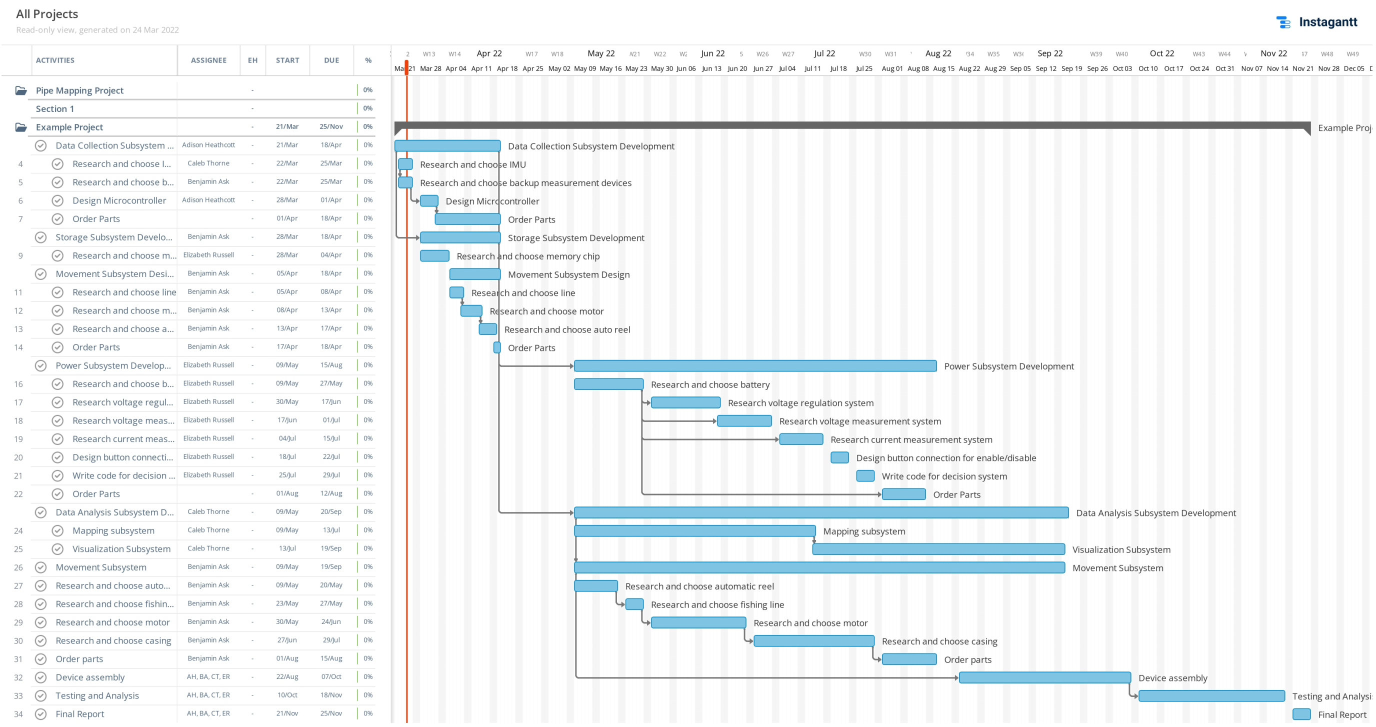

Fig. 2. Gantt Chart

Individual Statements

A. Adison Heathcott

In previous coursework I learned a lot of things that were very helpful in this project. Some are listed below:

- Circuit and Electronics Design and Analysis
- Microcontroller Design and Programming
- Signal Processing
- Physics

During the course I learned, and had to teach myself, a lot. Examples are listed below:

- PCB Design and Debugging
- Electronics Debugging
- Python and KiCAD
- Working long term with a team on time sensitive projects.
- Improved Presentation Skills
- Improved Project Planning Skills

B. Benjamin Ask

In previous coursework I learned...

- How to analyze a circuit
- How to integrate electronics
- Calculus based physics
- Professionalism (how to present work, how to work in a team)
- How to program using C++
- What a microcontroller is and how to implement it

During the course of this project I learned...

- Gained a deeper understanding of motors, and motor controllers (for the mechanical subsystem)
- Gained a deeper understanding of the physics involved in pulling probe through PVC (for the mechanical subsystem)
- How to better operate within a team (during project planning and assembly)
- Better presenting skills for technical presentations (design phase 1 presentation and final presentation especially)
- How to create and use a Gantt Chart for timeline management (gave us an understanding of how long we had to work on each piece of the project)
- How to use a motor controller to achieve consistent velocity of a motor (for the mechanical subsystem testing)
- How to brainstorm solutions to a unique problem (encompasses the entire project proposal)

C. Caleb Thorne

Previous coursework equipped me with several skills that were helpful in this capstone project, such as:

- Analog circuit analysis
- Calculus based physics
- Signal processing fundamentals
- General soldering techniques
- Use of test equipment (Oscilloscopes, DMMs, etc.)

Over the course of this project, I also learned:

- Techniques for soldering SMD components by hand
- How to use a reflow oven and stencils to solder leadless SMD packages
- Methods of troubleshooting PCB prototypes, and improvising creative solutions in the case of hardware failure

- General project management skills
- Improved presentation skills; particularly technical presentations
- How to successfully cooperate with a team that is designing both hardware and software in tandem
- How IMUs can be functionally implemented for mapping, and how much software/filtering work is required to achieve adequate results with them

D. Elizabeth Russell

Topics that I learned in prior courses which were helpful in this project are listed below.

- Circuit and electronics analysis
- Electromagnetic fields
- C++ and programming logic
- Calculus based physics
- Telecommunication and signal fundamentals

During this course I learned:

- To utilize and read Python code
- Application of telecommunication techniques
- How to improve my public speaking and presentation skills
- The ways IMU data can be used in mapping
- Ways to troubleshoot code
- How IMU data can compound errors and ways to solve those errors
- To successfully communicate and cooperate with a team in relation to time sensitive projects
- What ways are best when finding facts on lesser known information
- Improved problem solving skills

REFERENCES

[1] G. Zeiss, “Inertial mapping of Underground Pipe Networks,” Between the Poles, 10-Dec-2019. [Online]. Available: https://geospatial.blogs.com/geospatial/2019/12/inertial-mapping-of-underground-pipe-networks-.html. [Accessed: 12-Feb-2022].

[2] W. Heinselman, “What is trenchless pipe replacement and how much does it cost?,” What is Trenchless Pipe Replacement and How Much Does it Cost?, 04-Aug-2017. [Online]. Available: https://www.expresssewer.com/blog/what-is-trenchless-pipe-replacement-and-how-much-does-it-cost#: :text=Conventional%20dig%20and%20replace%20pipe,are%20 longer%20than%2050%20feet. [Accessed: 17-Feb-2022].

[3] “The Tools Necessary For Underground Pipe Installation,” https://www.catrentalstore.com/en_US/blog/underground-pipe-installation.html. [Online]. Available: https://www.catrentalstore.com/en_US/blog/underground-pipe-installation.html. [Accessed: 17-Feb-2022].

[4] PVC Chemical Resistance Guide, 1st ed. Oakville, Ontario: Ipex, 2020, p.6 [Online]. Available: https://www.ipexna.com/media/11116/chemical-guide-en-ipex-pvc.pdf. [Accessed: 17- Feb- 2022]

[5] Charlotte Pipe and Foundry Company, [Online] https://www.charlottepipe.com/price_lists.aspx. [Accessed:17-Feb-2022]

[6] Digikey Electronics, [Online] Available: https://www.digikey.com/en/products. [Accessed:17-Feb-2022]

[7] “History of plumbing in America,” PlumbingSupply.com. [Online]. Available: https://www.plumbingsupply.com/plumbing-projects-and-information/toilet-history-america. [Accessed: 23-Feb-2022].

[8] “Fact sheet: The biden-harris lead pipe and paint action plan,” The White House, 16-Dec-2021. [Online]. Available: https://www.whitehouse.gov/briefing-room/statements-releases/2021/12/16/fact-sheet-the-biden-harris-lead-pipe-and-paint-action-plan/. [Accessed: 23-Feb-2022].

[9] “NFPA 70®,” NFPA 70®: National Electrical Code®, 25-Aug-2019. [Online]. Available: https://www.nfpa.org/codes-and-standards/all-codes-and-standards/list-of-codes-and-standards/detail?code=70. [Accessed: 23-Feb-2022].

[10] P. Musungu, “Utilizing inertial navigation technology for 3D Mapping, locating and managing underground pipelines,” Construction Review Online, 21-Aug-2021. [Online]. Available: https://constructionreviewonline.com/machinery-equipment/utilizing-inertial-navigation-technology-for-3d-mapping-locating-and-managing-underground-pipelines/. [Accessed: 26-Feb-2022].

[11] “How to track underground water pipe?,” EngineerSupply, 04-Oct-2021. [Online]. Available: https://www.engineersupply.com/How-To-Track-Underground-Water-Pipe.aspx. [Accessed: 26-Feb-2022].

[12] ”NCHRP 15-44 GUIDELINES FOR THE USE OF MOBILE LIDAR IN TRANSPORTATION APPLICATIONS”, 01-Feb-2013. [Online]. Available: https://onlinepubs.trb.org/onlinepubs/nchrp/docs/nchrp15-44_finalguidelines.pdf

[13] “VR positional tracking,” Wikipedia, 18-Feb-2022. [Online]. Available: https://en.wikipedia.org/wiki/VR_positional_tracking. [Accessed: 27-Feb-2022].

[14] Sagar, “Applying IPC-2221 standards in circuit board design,” Sierra Circuits, 18-Jan-2022. [Online]. Available: https://www.protoexpress.com/blog/ipc-2221-circuit-board-design/. [Accessed: 27-Feb-2022].

[15] “Mercury in Batteries,” EPA, 07-Jan-2021. [Online]. Available: https://www.epa.gov/mercury/mercury-batteries. [Accessed: 27-Feb-2022].

[16] “Low power radio - general information,” Federal Communications Commission, 01-Sep-2021. [Online]. Available: https://www.fcc.gov/media/radio/low-power-radio-general-information. [Accessed: 27-Feb-2022].

[17] “Guide to Wireless Regulations in the United States,” 09-Feb-2015. [Online]. Available: https://www.linxtechnologies.com/wp/wp-content/uploads/fcc_resource_document.pdf. [Accessed: 27-Feb-2022].

[18] “Simultaneous Localization and Mapping for Inspection Robots in Water and Sewer Pipe Networks : A Review,” sept-2021. [Online]. Available: https://www.researchgate.net/publication/354874891_Simultaneous_Localization_and_Mapping_for_Inspection_Robots_in_Water_and_Sewer_Pipe_Networks_A_Review. [Accessed: 27-Feb-2thebibliography022].