School of Electronic and

Electrical Engineering

FACULTY OF ENGINEERING

UNIVERSITY OF LEEDS

ELEC5870M

Final Report

The Design and Development of a Nuclear Pipe Inspection Robot

Benjamin Evans

SID: 201216635	Project No. 12
Supervisor: Andrew Kemp	**Assessor: Zoran Ikonic**

School of Electronic and Electrical Engineering
FACULTY OF ENGINEERING

UNIVERSITY OF LEEDS

ELEC5870M MEng Individual Project

Declaration of Academic Integrity

Plagiarism in University Assessments

and the Presentation of Fraudulent or Fabricated Coursework

Plagiarism is defined as presenting someone else's work as your own. Work means any intellectual output, and typically includes text, data, images, sound or performance.

Fraudulent or fabricated coursework is defined as work, particularly reports of laboratory or practical work that is untrue and/or made up, submitted to satisfy the requirements of a university assessment, in whole or in part.

Declaration:

- I have read the University Regulations on Plagiarism[1] and state that the work covered by this declaration is my own and does not contain any unacknowledged work from other sources.
- I confirm my consent to the University copying and distributing any part or all of my work in any form and using third parties (who may be based outside the EU/EAA) to monitor breaches of regulations, to verify whether my work contains plagiarised material, and for quality assurance purposes.
- I confirm that details of any mitigating circumstances or other matters which might have affected my performance and which I wish to bring to the attention of the examiners, have been submitted to the Student Support Office.

[1] Available on the School Student Intranet

Student Name: Benjamin Evans Date: 24/03/2022

Signed: BEvans

Abstract

There are currently no commercially available 2” pipe inspection robots that can be used in the nuclear industry. To decommission Sellafield nuclear power station a miniature robotic platform has been developed which can navigate a 2” pipe network and map radiation. The commercialisation of the system developed aims to reduce the cost of the post operational clean out phase of decommissioning and make the environment safer for humans to work in. The system can manoeuvre complex pipe geometry, and the impact of the harsh environment it operates in has been explored. Preliminary testing has validated that system meets the mapping accuracy and controllability requirements set by the nuclear decommissioning industry.

Acknowledgements

I would like to thank Andrew Kemp for his amazing support throughout my final year project. With trusting me to select my own project idea and always being committed to meeting every week is something I am very grateful for. It has been great to get to know Andrew, and he taught me many life lessons that I will take with me after I graduate.

I would also like to thank Nicholas Castledine for allowing me to carry on his amazing research project and work on robotics which is my passion. It has been great working with Rory Turnbull on this project, and I am very thankful for all the advice he has given me.

Finally, I would like to thank Alastair Baker and Christopher Bulman for going out of their way to help me obtain a radiation permit and run experiments in the GM59 nuclear laboratory.

Contents

Table of Figures

1 Introduction

This project aims to develop an inspection robot that can navigate a 2” pipe network and map radiation to decommission the Sellafield nuclear power station. Previously the pipe in bore articulated inspection robot (PIBAIR) platform was developed as part of the robotics and AI in nuclear (RAIN) initiative [1]. The PIABIR platform needs to be updated to allow it to map and travel further into pipe networks.

To navigate a pipe network, the range of the PIBAIR system must be increased by miniaturising the current control electronics to fit inside the pipe. A new communication system must be chosen to enable control of multiple robot modules along a tether and allow a video camera to be integrated into the system. In order to map radiation in the pipe, hardware must be selected and designed to integrate a radiation sensor. Radiation resistance testing will be conducted on the electronics to validate function within a nuclear environment. A mapping system must be built with a complementary graphical user interface (GUI) to enable intuitive control of the whole PIBAIR system. This work is the continuation of the predoctoral research by Nicholas Castledine. It is now part of a research project in the Real Robotics laboratory and is completed in collaboration with Rory Turnbull.

Currently, there are no commercially available 2” pipe inspection robots for the nuclear industry. Many of the previous small pipe robot platforms struggle to manoeuvre complex pipe geometry reliably. The current PIBAIR system can already navigate pipe t-sections and zero-radius bends. This supports further research into developing this platform into a commercial product to be used at Sellafield. Developing this pipe inspection robot will reduce the cost of the post operational clean out phase of decommissioning and make the environment safer for humans to work in.

To upgrade the current PIBAIR system, many technical issues that come with designing 2” robots will have to be overcome. A unique solution for the mapping system will have to be engineered as traditional mapping methods do not work in pipes. All the electronic hardware developed must fit into the pipe and not hinder the manoeuvrability of the current PIBAIR platform. The whole system must withstand the harsh environment of a nuclear power station and the left-over contaminants in the pipes. Real-world testing will be used to validate the upgraded system's performance and help it reach the project objectives.

2 Background literature

2.1 Project History

Before any work on the project was conducted, significant time was spent understanding the previous PIBAIR system designed by Nicholas Castledine. The strengths and weaknesses of the previous system were analysed to see what upgrades were required, which helped define this project's objectives.

The previous prototype, detailed in Figure 1, successfully achieves many functions required to navigate pipe t-sections, zero radius bends and vertical pipes. The robot module has six legs with Dremel cutting wheels attached at the end. Each wheel is independently driven by a micro gear motor, and the legs are sprung outwards against the pipe walls using orthopaedic bands. The six motors allow the module to travel up to 2cm/s in a pipe. Two additional retraction motors are attached to the six legs with fishing line. To navigate pipe t-sections, the front legs are retracted together to allow the robot module to turn by driving the rear motors. A spring in the middle of the robot allows it to bend around the corners. By retracting the back legs, the robot's diameter can be reduced to 25mm to allow it to enter narrower pipe openings [2].

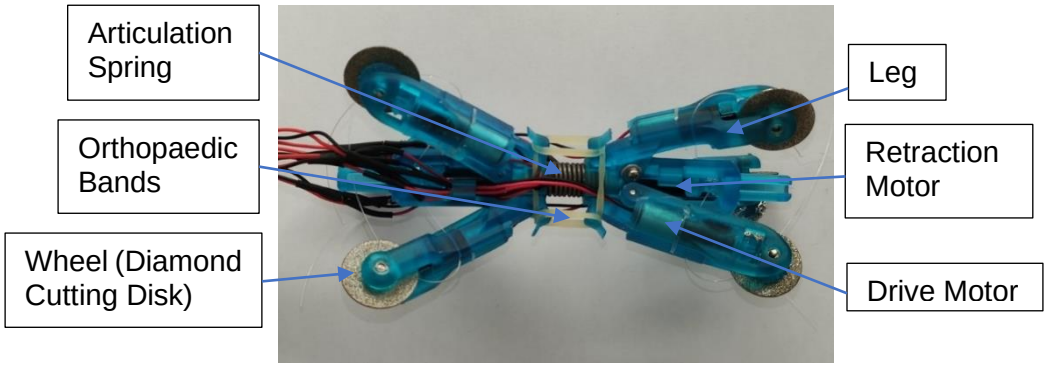

Figure 1: Previous Prototype of PIBAIR Developed by Nicholas Castledine [2]

The eight motors are controlled by four dual-channel motor drivers and a microcontroller that sits outside the pipe, as shown in Figure 2. These send pulse width modulation (PWM) signals to the robot module via a tether to control the motor speeds. PWM signals have a limited range, and therefore, the maximum operating distance of the previous prototype is only 2 meters. The robot is controlled via an Xbox controller and simple GUI, built using the Processing IDE, displays PWM values and system credentials. The previous project mainly focused on the mechanical hardware of the robot, so many improvements need to be made to the electronic hardware. The most significant improvement is increasing the system's range by miniaturising the motor control electronics to fit inside the pipe. Even though the mechanical design of the robot module is functional, there are still a lot of physical changes that need to be made. Rory Turnbull will focus on upgrading the mechanical design of the robot to make it less fragile and easier to manufacture. This will involve removing the use of fishing line, increasing traction in contaminated pipes and integrating a camera into the front module [2].

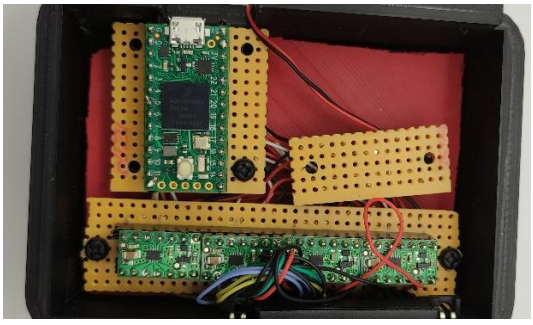

Figure 2: Previous PIBAIR Prototype Motor Control Electronics

2.2 Market Research

As the previous PIBAIR prototype uses a wall-press design, the review by Mills et al. [3] helped to understand how the mechanics of pipe robots function. Fortunately, a control system by J.Norton [4] has already been programmed for the previous prototype, which handles the mechanical motion control of a wall-press robot using motors. Therefore, this control system only needs to be adapted to allow multiple robot modules to move through the pipe collaboratively [2].

As this section of the project is focused on the electronics and software design, the review of “Localisation, Mapping, Navigation, and Inspection Methods in In-Pipe Robots” by Kazeminasab et al. [5] has been a vital source of the different technologies that could be potentially implemented into this system. This review gave many examples of other pipe robots in the industry and highlighted the current issues that pipe robots face. There have been other 2” pipe robots developed in the past, but these have a limited range [6] and struggle to navigate pipe geometry [7] consistently. With the previous PIBAIR prototype already being able to navigate complex pipe geometry reliably, this signifies the outstanding research by Nicholas Castledine to develop the previous system [2]. The project's previous success supports the need to develop the PIBAIR system further to push toward commercialisation.

The electronics will be miniaturised to fit inside the pipe to overcome the range issues that previous 2” pipe robots faced. A similar pipe inspection robot to PIBIAR by Brown et al [8], also part of the RAIN initiative, faced many difficulties in miniaturising the electronics to fit inside a pipe this small. This project, therefore, aims to overcome those challenges by using a unique modular solution to spread out the robot’s control electronics along the length of the tether to reduce the diameter. Adding this on top of the previous prototype's already successful mechanical platform should solve all the issues that other 2” pipe robots have faced.

2.3 Implementation

Many technologies could be implemented into the PIBAIR system to increase its range and add mapping functionality. As this system will be exposed to radiation and nitric acid, the chosen technologies should have some level of resistance to the surrounding harsh environment. Therefore, the implementation and radiation resistance of different communication, video and mapping systems in pipe bots have been analysed.

2.3.1 Mapping

The robot's movement through the pipe needs to be tracked to determine where radioactive contaminants are located in the pipe. The movement data can be used to build a 3D map of the pipe network that the robot has travelled through. This will then be combined with radiation intensity data to construct a map of the radiation intensity throughout the pipe network. As the pipes are contaminated, wheel encoders can not be used to measure the movement distance of the robot in the pipe, due to the wheels slipping. Encoders would also be very difficult to integrate into the small size of the current PIBAIR system. GPS sensors can not be used either to track movement distance as the metal pipes block the signal. Some pipe robots have had success with using sound-based localisation [9] or an RF system [10]. However, these systems are very complex, have varying accuracy in metal pipes and would require a long implementation time [2].

From the review by Kazeminasab et al. [5], the most successful and popular technique for mapping is to use an inertial measurement unit (IMU) to calculate the robot's orientation. Unfortunately, IMUs are only accurate over a short period. Therefore, they cannot be used to calculate the distance moved by the robot into the pipe as IMU output values suffer “integration drift over longer time scales” [11]. Therefore, the IMU data will need to be combined with another source of data which is accurate over long time scales and provides the distance that the robot has travelled into the pipe.

One solution proposed by Murtra and Mirats Tur [12] was to use an encoder to measure the length of the tether that travels into the pipe. Then use an IMU to tell if the robot has made a turn or gone round a bend. Another project by Lim et al.[13] also used a tether

encoder to successfully build a 3D map of a pipe network by tracking the movement data of the pipe robot. Out of all the mapping concepts, IMU and tether encoder data fusion seems the most promising solution for the PIBAIR system [2].

2.3.2 Communication

The previous PIBAIR system microcontroller communicated with a computer via USB cable to control the robot module's movement. This means that the range of the system would be governed by the length of the USB cable that attaches to the microcontroller in the pipe. USB 2.0 cables have a range of five meters which will not be satisfactory for the range of the new system [14]. Therefore, a unique communication solution which can operate over longer distances needs to be implemented to control the robot modules and send video data back to the operator. It must also allow multiple robot modules to be controlled simultaneously, connect to the radiation sensor and camera, and be resistant to the noisy environment of a pipe network.

A long-range pipe robot by Chen et al. [15] used a controller area network (CAN) bus for the communication system. CAN bus is a very robust communication protocol and is used extensively in the automotive industry due to its excellent interference rejection and error detection. This allows multiple devices to communicate over the same wires in the network, which for the PIBAIR system, would reduce the tether thickness [2].

2.3.3 Video

A camera will need to be integrated into the first PIBAIR module to provide the operator with visual data of the inside of the pipe to make t-section turns. The camera must be small enough to fit in the pipe but still provide enough visual data to make the system controllable. The previous PIBAIR prototype experimented with using an endoscope camera, but this had limited range as it was connected via USB. The pipe robot by Kwon et al. [7] is a similar size to PIBAIR, and for their system, they make use of a micro CMOS camera to provide video data. The camera is directly connected to the microcontroller, and data is sent live over the tether back to the operator [2].

The camera choice and communication decisions both overlap and affect each other. The communication system can affect the camera's video quality if it has a maximum data rate. Then the selected camera could affect the type of communication system used as some cameras only connect via specific interfaces like USB. The decision for the type of camera and communication protocol will have to be made in parallel.

2.3.4 Radiation Resistance

As the motor control electronics are implemented within the pipe, they could potentially be exposed to ionising radiation. Gamma radiation can degrade electronics over time, especially semiconductor components. Generally, radiation-hardened electronics are used in the nuclear industry, which are more resistant to ionising radiation exposure. Unfortunately, radiation-hardened electronics are costly and require specialists to implement them, which is outside this research project's funding. Therefore, consumer electronics will continue to be used in this project and the effect of ionising radiation on the PIBAIR system will be studied.

Research conducted by Nancekievill [16] found that ionising radiation affects consumer electronic parts, such as voltage regulators and IMUs used in the PIBAIR system. It was found that the voltage output of a voltage regulator is reduced with an increasing total dosage of ionising radiation. This could cause the PIBAIR motor control module to stop working as it is not being provided with enough power. However, as shown in Figure 3, it was found that for a 5V voltage regulator, the output only drops by 0.55 V when it has been exposed to a total ionising dosage (TID) of 5 KGy(Si). This is the equivalent of the electronic device being “in a nuclear decommissioning environment for approximately 4 hours a day, 5 days a week for one year” being “exposed to an average dose-rate of 1.4 mGy(Si).s-1” [16]. A voltage drop of 0.55 V would have a negligible effect to the current PIBAIR system as all the electronic components can operate over a range of voltages, approximately 5 - 6 V.

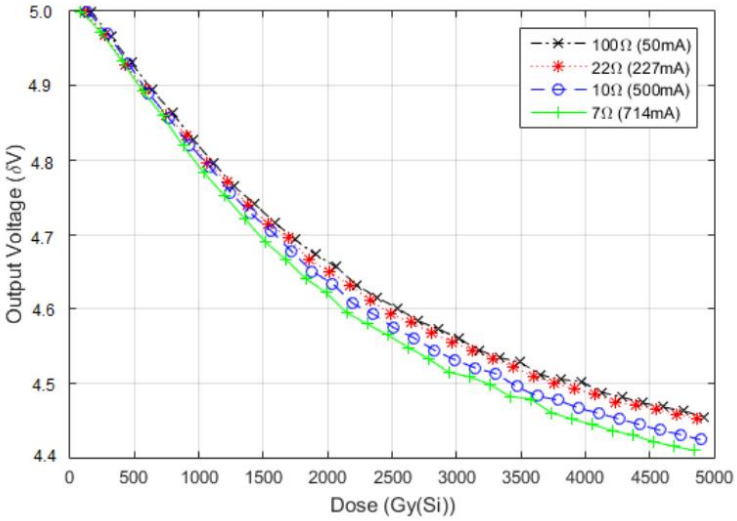

Figure 3: Degradation of KA7805 Voltage Regulator During Irradiation from [16]

Most parts of the PIBAIR system are single-use because they can be potentially contaminated in the pipes. Therefore, most electronics components will be replaced before being exposed to a TID close to the value tested above. It can also be assumed that if the robot's electronics fail due to too high of a TID, this will signify that there are intense levels of ionising contamination in that part of the pipe. Therefore, the operators need to be careful during decommissioning that section.

For IMUs the research by Nancekievil [16] found that their failure dosage was much lower than for voltage regulators. The three-consumer digital IMUs tested all failed after a TID of 800 Gy(Si), which is a fifth of what caused the voltage regulator change by 0.55 V. The raw output values of some of the IMUs were actually heavily affected from as low of a TID of 50 Gy(Si), shown in Figure 4. This would affect the orientation accuracy of IMU and therefore cause the mapping system accuracy to be reduced. As each IMU was affected differently by the ionising radiation, this supports running similar experimental tests on the effects of ionising radiation on the selected IMU for this project.

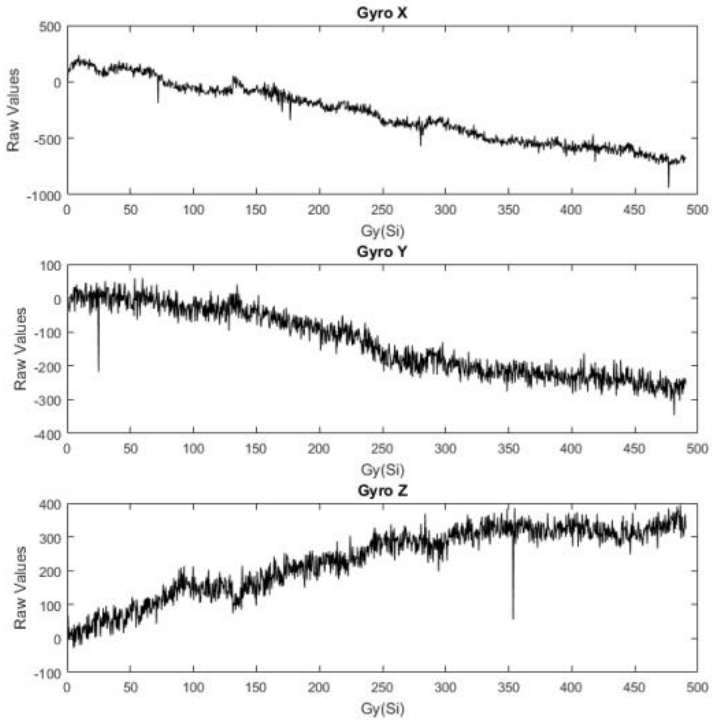

Figure 4: Degradation of LSM9DS0 IMU Gyroscope from [16]

2.4 Additive Manufacturing

With this being a mainly robotics-based project, 3D printing is utilised extensively throughout to produce the system's mechanical parts. Even though Rory Turnbull is responsible for all of the mechanical design of the PIBAIR modules, other parts of the system designed in this part of the project also require some mechanical design and 3D printing.

This part of the project uses the knowledge gained over the past three years in 3D printing to assist with manufacturing mechanical components. It also helps with understanding the mechanical parts designed by Rory Turnbull. The ability to operate 3D printers reduces the lead time of prototyping parts to a matter of hours. Understanding additive manducating is a fundamental part of robotics engineering and is a skill that is not taught in electronic engineering.

Fused deposition modelling (FDM) 3D printing will be used to iteratively prototype the housings for the tether encoder and motor control modules. As this project aims to be commercialised, the manufacturing techniques used during the prototyping phase have to be repeatable on a large scale. With 3D printing farms becoming less expensive, the same technology can then be used to manufacture the system at scale. This reduces development costs as tooling for injection moulding does not need to be made. It also means mechanical designs can be smaller and more complex than if injection moulding was used.

Different 3D printable materials can offer higher resistance to the nitric acid found in the pipes. As the robot modules are very small and intricate, stereolithography (SLA) 3D printing will be used to prototype designs. Formlabs SLA printers can print at a much higher resolution than FDM and use cured resins rather than plastic. A solvent compatibly table by Formlabs [17], details the resistive properties of their SLA resins to multiple solvents and acids. Unfortunately, there is no data for the resistance of the resin prints to nitric acid. Therefore, Rory Turnbull will conduct experimental tests to validate using SLA resins as a manufacturing material.

3 Initial System Design

The final PIBAIR system is very complex, with many different parts interacting with each other. Therefore, to ensure the final objectives are achieved, the system has been broken down into separate parts to clarify each engineering task. Industry-standard system engineering tools by S.Burge [18] have been utilised to clearly define the context, function, requirements and scope of the PIBAIR system.

3.1 Context

The PIBAIR system will be used in a commercial setting and must be able to withstand the environmental factors around it. The system will affect many different outside entities, and the boundary of its interactions must be defined. The context diagram in Figure 5 shows the interactions of the PIBAIR system within the environment of a nuclear power station [19].

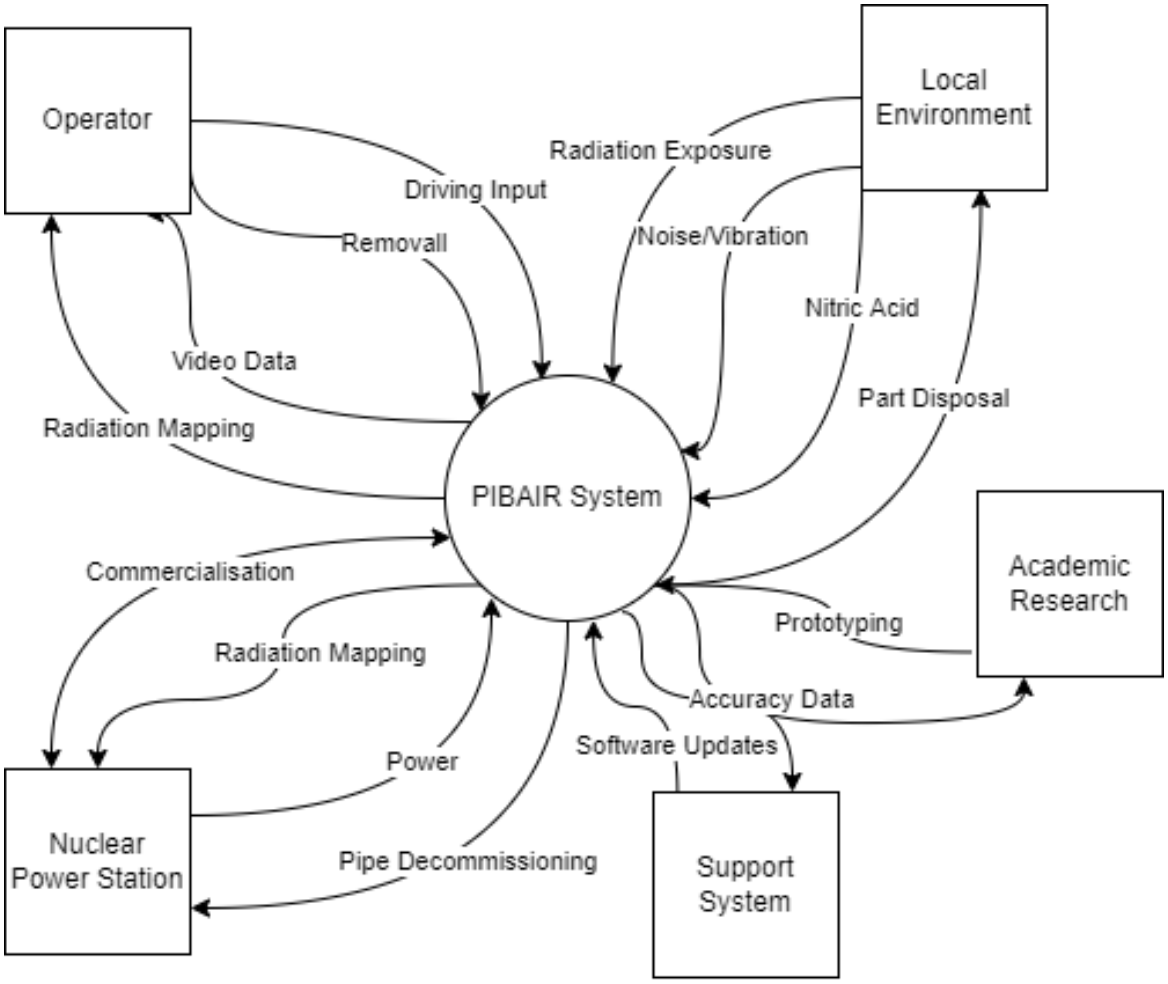

Figure 5: PIBAIR Context Diagram

When looking at commercialisation, the main objective of the PIBAIR system is to map radiation within a pipe for decommissioning. Therefore, the system must successfully communicate with the operator to keep them safe from coming into contact with any radiation. The system must also respond to the operator's input with intuitive controls. When operating in the harsh environment of a nuclear power station, the PIBIAR system has to withstand its local surroundings long enough to collect the required data. The accuracy of the mapping data must not be affected by noise or radiation exposure, as this could mislead the operator on the whereabouts of the radiation. These surrounding factors highlight the importance of the PIBAIR's reliability as it significantly affects operator safety.

With the current PIBAIR system being part of the Real Robotics research project, the system's short-term objectives are very different from the above. There is less of a focus on the safety-critical aspects of the system and more on designing the system's hardware. The current objective is to present a working prototype within a test environment to validate the hardware design and help source external funding for commercialisation. Therefore, by the end of this research project, the goal is to have two robot modules navigate collaboratively through a small pipe network and the accuracy of the mapping system to be obtained. Figure 6 visualises how two robot modules will be positioned in a pipe, separated by a payload. The results of this research project will be included in a Pipebots journal paper. Once two robot modules work together, the system can be expanded to include up to 5 robot modules required for it to go further into the pipe due to increased tether loads.

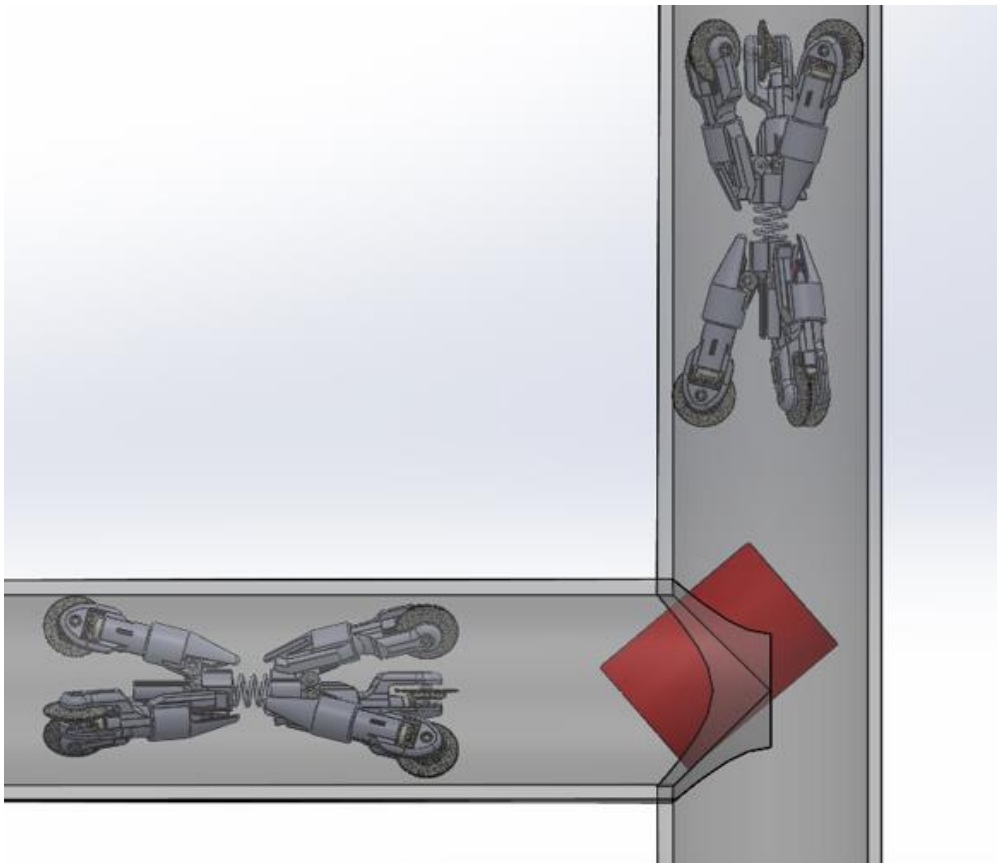

Figure 6: Payload Module Size Limitation for Cornering - Provide by Nicholas Castledine

3.2 Function

Looking at the background research by Kazeminasab et al. [5], it has been found that many different design options could be used to make a working prototype. Each part of the PIBAIR system has a separate function which can be implemented using various technologies. The different means of reaching each function, summarised in Figure 7, have been analysed to find the best combination of technologies that work together for the whole system [20]. It is essential to analyse the functions of all different parts of the engineering system. Otherwise, solutions that do not function as intended can be designed, and the project goals will not be achieved. Once the system parts have been designed and manufactured to accomplish these five functions, it should reach the project goals of navigating a pipe network to map radioactive contaminants.

Function	**Means**				
Measure Radiation Intensity	Gas Initiation	Scintillation Detectors	Semiconductor Detectors		
Mapping Pipe Network	Image Recognition	Tether Encoder	IMU	LIDAR	Ultrasonic Sensor
Navigation Control	Autonomous	Games Controller	Touch Control	Video Data	GUI
Drive Motion	Wall-press	Screw-Drive	Snake Type	Legged	
Data Communication	Tether	Wireless Radio	Magnetic Induction		

Figure 7: Function Means Analysis Table

The highlighted means in Figure 7 show the chosen technologies that will be used to achieve each function of the PIBAIR system. Some of the means were previously selected by Nicholas Castledine for the last PIBAIR prototype and will continue to be used as they have already been validated. These include using a wall-press design and a tether for communication. The system will require multiple means to achieve the navigation control and mapping functions. The system will initially continue to use a game controller for navigation control. However, in looking towards commercialisation, this will switch to touch control to make the system more intuitive to learn. A camera will be used to capture video data for the operator to use to control the robot and manoeuvre around t-sections.

Figure 7 was then used to create a simplified top-level system diagram for the PIBAIR system, shown in Figure 8. This diagram highlights the five essential parts of the PIBAIR system and their functions.

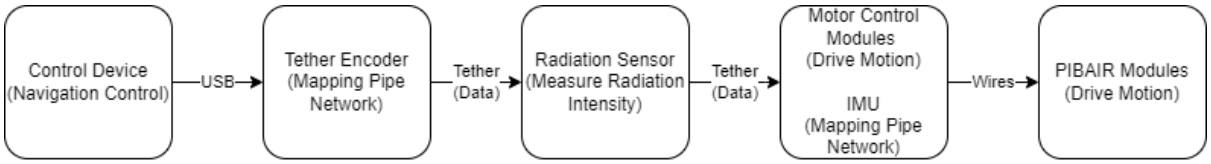

Figure 8: Simplified High-Level PIBAIR System Diagram

This system diagram details the key naming for each part of the PIBAIR system that will be used throughout the project. Each PIBAIR Module will require one motor control module. The front motor control module will include an IMU sensor to measure the robot's orientation in the pipe network and when a turn has been made. The whole system will communicate through a single tether that passes through the tether encoder at the open end of the pipe. The tether and tether encoder will connect via USB to a control device used by the operator. Finally, the radiation sensor will be positioned behind the PIBAIR modules, partway down the tether.

3.3 Requirements

The PIBAIR system has many requirements set by the project's objectives. From these, other requirements were defined that would have to achieved to reach the original requirement set by project objectives. Any future design or hardware decisions will consider all the requirements to make sure they collectively help reach the project objectives. Some of the requirements of the project are more important than others. Therefore, higher-ranking requirements will take precedence over lower raking requirements if a cost vs reward decision is made between the two [21]. The PIBAIR system has both functioning and non-function requirements which have been combined into one list for simplicity. It should be noted that the requirements are focused on achieving the objectives of this research project and not on commercialisation. The ten key requirements for the PIBAIR system, in order from most important to least important, are as follows:

1. **Implementation Time:** The chosen hardware, technology and designs must be implementable in the time frame that the project is run for as it affects a wider research project. It would be a waste of time to try and develop an overly complex solution that cannot be implemented in time, as this will affect the funding of the research project. This requirement helps to achieve the project completion goal.

2. **Part Availability:** Each part used in the PIBAIR must be available for delivery in a short time frame and cannot have long lead times. The current semiconductor and electronic parts shortage has impacted this requirement heavily. This requirement has to be achieved to meet the implementation requirement.

3. **Video Quality:** Without video data, the PIBAIR system is uncontrollable as the operator will not be able to drive it through a pipe network. Therefore, the most critical data that the PIBAIR provides is video. A minimum requirement for the video quality has been set to one frame per second at 360p resolution. This requirement helps to achieve the control objective and reach the navigation goal.

4. **Electronic Miniaturisation:** All hardware designed and selected must be able to fit around zero radius and t-section bends as seen in Figure 6. Nicholas Castledine calculated that the maximum dimensions for non-flexible parts are a diameter of 40mm and a length of 55mm. This requirement has to be achieved to meet the range requirement.

5. **Radiation Sensor Digitisation:** The previously selected Kromek GR1-A [22] radiation sensor communicates with a computer via USB. The PIBAIR system must allow the radiation intensity data to be sent down the tether to appear on the control device in real-time. This requirement needs to be achieved to meet the radiation identification objective and helps reach the map radiation goal.

6. **Range:** The first objective of the system is to have a target range of 30m. This is considered far enough for the operator to inspect a reasonable distance of pipe at a time, keeping them safe from radiation. Having a larger range would impact the design and performance of the PIBAIR system. This requirement has to be achieved to meet the range objective and navigation goal.

7. **Tether Thickness:** As the PIBIAR modules pull the tether further into the pipe, they will have to tow an increasing tether load. With the pulling force of each PIBAIR module being approximately 3 newtons, the weight of the tether could stop the modules from moving forward as it gets further into the pipe. Therefore, reducing the tether thickness by having fewer wires inside will reduce its weight. A thinner tether will also more easily bend around pipe geometry. This requirement assists with achieving the range requirement.

8. **Mapping Accuracy:** After speaking to a French nuclear decommissioning company, a mapping accuracy of ± 2cm was suggested as a good mapping accuracy target. This requirement needs to be met to reach the mapping objective and helps to reach the mapping radiation goal.

9. **Controllability:** The PIBAIR modules must be able to navigate pipe t-sections reliably. Therefore, the system must be intuitive to control and respond to the operator's inputs successfully. This requirement helps to achieve the navigation goal.

10. **Cost:** As the PIBAIR modules can potentially become contaminated and are too intricate to decontaminate, they will be the only single-use part in the system. Therefore, the cost to produce each PIBAIR module could be considered. However, the cost is the least significant factor. In the nuclear industry, approximately £10,000 is considered the maximum cost for single-use robots, which is far more than the whole PIBAIR system's expected cost. This requirement is associated with the project motivation of the commercialisation of the system.

There is much overlap between all the requirements which can cause decision loops when designing or choosing hardware. Therefore, the ranking system is vital to ensure that each decision made has the best final outcome for the end system. The order of importance for the requirement ranking hierarchy above was calculated by analysing how decisions made for one requirement will impact the other requirements. The matrix diagram in Figure 9, described the relationship between the different requirements and their relative strengths from 0 (No Relationship) – 9 (Strong Relationship) [23].

Requirements	**Implementation Time**	**Part Availability**	**Video Quality**	**Electronic Miniaturisation**	**Radiation Sensor Digitization**	**Range**	**Tether Thickness**	**Mapping Accuracy**	**Controllability**	**Cost**
Implementation Time		9	6	5	8	3	0	4	4	1
Part Availability	5		0	4	2	6	0	0	0	0
Video Quality	6	9		7	0	9	0	0	0	0
Electronic Miniaturisation	6	9	9		7	3	0	5	0	0
Radiation Sensor Digitization	7	0	5	0		0	0	0	0	8
Range	5	3	9	6	8		9	4	6	0
Tether Thickness	5	3	0	7	4	9		5	0	0
Mapping Accuracy	9	3	5	5	7	4	5		0	0
Controllability	7	3	8	5	0	1	2	0		0
Cost	9	9	5	7	7	5	4	2	0	
Total	59	48	47	46	43	40	20	20	10	9

Figure 9: Requirement Relationship Strengths

The above matrix diagram should be read in the direction of the x-axis requirement relationship to the y-axis requirements. Therefore, the relationship strength between two requirements can vary depending on which direction they are compared in. For example, each

requirement does have a relatively strong relationship to the final cost of the system. However, when looking in the other direction, the available budget of the system has no relationship with a lot of the requirements. The system's total cost will be much lower than the maximum cost for a single-use robot in the nuclear industry. Therefore, if the budget were to go up or down, it would not affect a lot of the requirements. Cost does have a strong relationship with radiation sensor digitisation as the software drivers for the radiation sensor are very expensive and outside the budget of this research project. The total values for each requirement signify their overall relationship strengths with the rest of the requirements for the project.

Implementation time is the most important requirement and has a strong relationship with all the other requirements. The work for this project is being used in a Real Robotic research project with separate deadlines and performance deliverables, so there is a responsibility for everything designed having to work. This is where this project's design decisions differ from other master's projects. It would be possible to design very complex concepts for miniaturising the electronics, but these would not be implementable in the project's time frame. Therefore, less complex design ideas that can be manufactured and validated within the time frame of this project will be used. This especially applies to part selection as all hardware that will be selected must be implementable into the current system in the time frame. A less complex system will have fewer problems, and it will be quicker to debug issues.

3.4 Scope

This project is the continuation of the previous work by Nicholas Castledine, and the current work conducted is in collaboration with Rory Turnbull. Therefore, the project scope in Figure 10, has been defined to clarify the work conducted, objectives and deliverables for this part of the project.

In Scope	Out of Scope
Miniaturise control electronics	Selection of microcontroller, radiation sensor, motors and motor drivers
Integrate radiation sensor	All mechanical design or manufacture of PIBAIR modules
Develop a mapping system	Housing for the motor control modules
Program a GUI and control software	Commercialisation
Select new PIBAIR hardware	Testing in a nuclear power station environment
Radiation resistance testing	Deciding the objectives for the overall system

Figure 10: Project Scope Definition

Being transparent, some of the in-scope items were split between this part of the project and Rory Turnbull. Figure 11 describes how the workload was shared between the two. It should be noted that the motor control module is made up of two parts, the USB Hub PCB and the Motor Control PCB.

Responsibilities	Benjamin Evans	Rory Turnbull	Both
Miniaturise control electronics	Motor control PCB	USB hub PCB, 6V to 5V PCB, Motor PCB conectors	System assembly
Integrate radiation Sensor	Software integration	USB hub hardware integration	N/A
Develop a mapping system	Design of system, tether module and all programming	N/A	N/A
Program a GUI and control software	GUI programming, build control tablet	N/A	N/A
Select new PIBAIR hardware	Selection of IMU, encoder, camera and USB extender cable	Wire connections, voltage regulators and power supplies	Communication protocol
Radiation resistance testing	IMU testing and research	N/A	Radiation permit training and Risk Assessments

Figure 11: Workload Share

3.5 Proposed Initial System Design

The understanding gained from the systems engineering tools above and background research has been combined to create an overall PIBIAR system design that will meet all the project objectives. The proposed initial system design in Figure 12 shows how the different parts of the PIBAIR system are connected at the software and hardware level. Any pieces of hardware being used from the previous version of PIBAIR are detailed, along with any technologies that will be utilised to design custom hardware. The required power electronics and mechanical hardware are also shown.

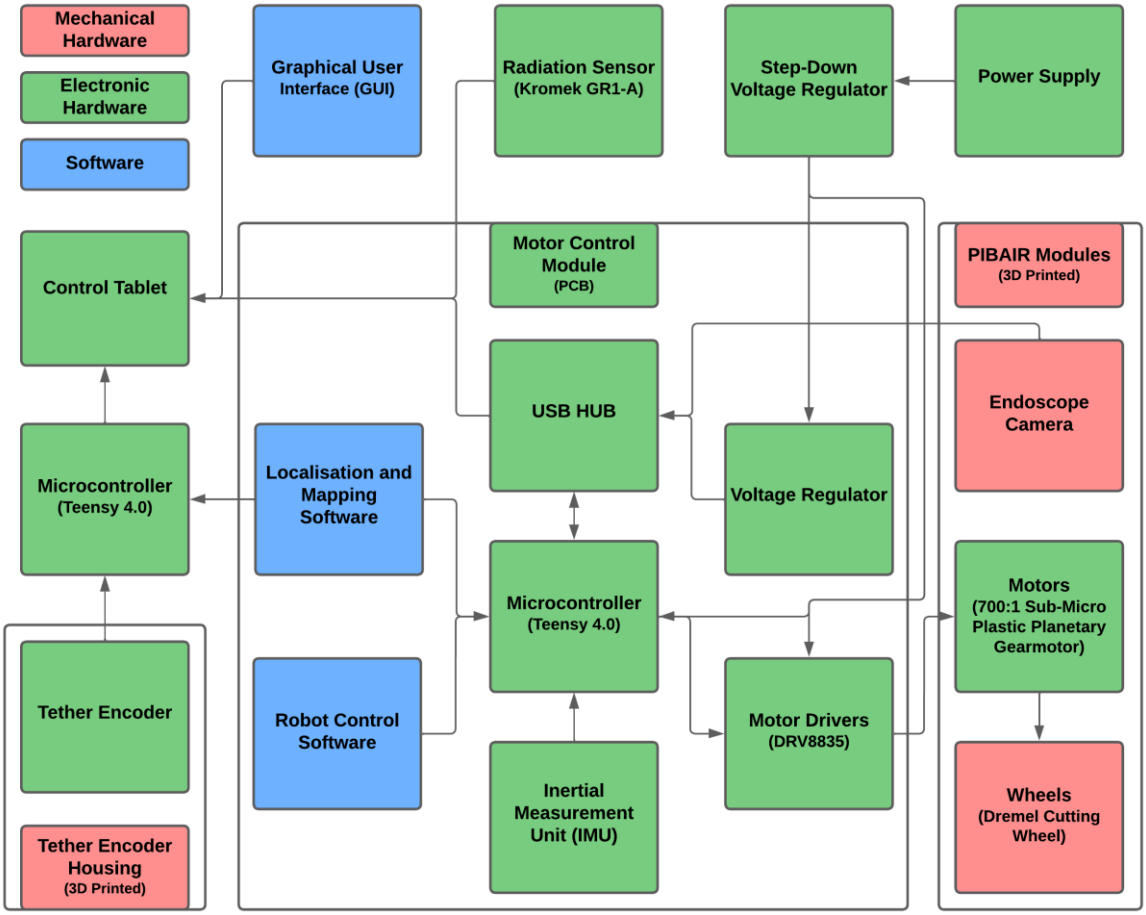

Figure 12: Initial System Diagram

The green blocks are the electronic hardware that must be selected and designed to produce a functioning system. The blue blocks indicate any software that must be programmed to create the GUI and mapping system. The red blocks are mechanical components which will need to be developed using computer-aided design (CAD) and then 3D printed. The PIBAIR system will require the use of multiple programming languages which run on different microcontrollers and computers. The project will continue to use the same microcontroller, Teensy 4.0, from the previous version of PIBAIR. This means the same code base for the motor control system can be used again, reducing software development time. All the pin and motor driver connections are already well documented in Appendix A, Figure 34, which will help with the PCB design. Initial research validated that Teensy 4.0 is the best microcontroller choice for this project. As there are many Teensy's already available in the Real Robotics laboratory, it has become a requirement to continue to use them. Therefore, there will be no further analysis into other microcontroller options.

4 Work Conducted

4.1 Hardware selection

A communication system, video camera, and mapping hardware must be chosen to reach the project objectives. To make hardware selection decisions that best meet the requirements, the systems engineering tools by S.Burge [18] have been used in combination with the requirements defined in part 3.3. Pugh Matrix diagrams will be used to compare how the different hardware options meet each system requirement. The Pugh Matrix takes advantage of people being more successful at comparing just two individual factors rather than many at the same time. It then combines all those comparisons to allow you to make more objective opinions on hardware decisions with many factors [24].

For a Pugh Matrix, one of the hardware options is selected as the baseline (0). Then for each requirement, the other hardware options are either ranked the same as the baseline with a 0, better than the baseline with a 1 or worse than the baseline with a -1. As some of the requirements are more important than others, the values assigned for each requirement are scaled by a requirement weighting of 1 (least important) – 5 (most important). The requirement weighting values are calculated using the ratio of the totals from Figure 9. The total score is then calculated for each hardware option. The option with the highest score is meant to be objectively the best choice. However, sometimes there is not always a clear best option. Although a Pugh Matrix will always help you identify the option that would be the worst decision or if any hybrid solutions could come from using two options together.

Any hardware or technology decision selected for comparison in the Pugh Matrix will have at least met all the minimum requirements. A comparison then follows to find which option meets all the requirements the best. A lot of technologies and hardware solutions were not even considered due to long implementation times and poor component availability caused by the chip shortage. The decision for the communication and camera systems was made simultaneously as the outcome of one decision affected the other.

4.1.1 Communication System

Selecting a suitable communication standard for the system is vital to reaching the range and control objectives. The communication standard also affects how the radiation sensor and camera are integrated into the system. Multiple communication standards have been analysed for use in the PIBAIR system. All the communication standards meet the minimum video quality requirement. To meet this requirement, the minimum data rate of the communication standard must be at least 519.2 kBps, calculated in Appendix A. The Pugh Matrix in Figure 13 displays how comparatively the different communication standards meet the requirements. CAN bus was selected as the baseline for this comparison as it has been used in other pipe robot systems [15].

Requirements	Weight	CAN bus [25]	USB [14]	RS-485 [26]
Implementation Time	5	0	-1	0
Part Availability	4	0	1	0
Video Quality	4	0	1	0
Electronic Miniaturisation	4	0	-1	-1
Radiation Sensor Digitization	3	0	1	0
Range	3	0	-1	0
Tether Thickness	2	0	1	-1
Mapping Accuracy	2	0	0	0
Controllability	1	0	1	0
Cost	1	0	-1	0
Total		0	1	-6

Figure 13: Communication System Standard Selection Pugh Matrix

The result totals from the comparison in Figure 13 show a common outcome for a Pugh Matrix. It has successfully identified one standard that is the worst option and is very close between the other two standards.

Looking at the comparison of CAN bus and RS-485 first, it is clear that CAN bus would make a better option for the communication system than RS-485. CAN bus and RS-486 are very similar for most of the requirements as they both have approximately the same interference rejection, data rates and range. However, the implementation method of the two standards is where they differ. An RS-485 chip would be harder to implement on the miniaturised motor control module because it requires more external resistors than a CAN bus transceiver chip. RS-485 would also require one extra wire than CAN bus, increasing the tether width.

Continuing to use USB for the communication system comes with many advantages. However, to use USB, a unique solution to increase its range and allow multiple modules to communicate over the same wires is required. A 30-meter USB 2.0 extender cable or USB redriver chips can be used to increase the range of the communication standard to above its

original limit of 5 meters. USB hub chips can be used to allow multiple USB devices to communicate over the same USB wire.

It is very close when comparing the totals for USB and CAN bus. USB would take much longer to implement as a custom USB Hub PCB for each module and the radiation sensor would have to be designed to allow them to communicate over the same wire. However, the software development time for the system when using USB is much shorter, as it can make use of the same control code from the previous prototype. A new control protocol would have to be programmed to communicate over CAN bus and select between which motor control module to talk to. The largest advantage of USB is the USB endoscope camera from the previous prototype can be used. This has a higher video quality than the only camera solutions implementable over CAN bus due to USBs higher data rate. As video quality is very important for improving the controllability of PIBAIR, USB will be used for the PIBAIR communication system to help reach that objective. It will be implemented using the unique solution described above to meet the range requirements. There is also more of a selection of USB hub chips than CAN bus transceiver chips and the Kromek GR1-A radiation sensor is only connectable via USB. Therefore, a single-board computer does not have to be implanted in the pipe to allow the radiation sensor to communicate over CAN bus.

4.1.2 Video Camera

The camera attached to the front PIBAIR module provides vital video data to the operator. Therefore, a video camera needs to be implemented for the system to reach the control objectives of the project. Even though a USB endoscope camera was previously used, other camera types that could be implementable in CAN bus and RS-485 communication standards were explored. This was because of the range issues of USB discussed above, which had not previously been solved. As the endoscope camera was USB video class (UVC) compliant, there was an attempt to try to connect it directly to the microcontroller to allow it to communicate over the other communication protocols using the solution by miciwan [27]. However, it was found that this would not be possible due microcontroller not having enough flash memory to produce a high enough quality video. Therefore, multiple video cameras with different specifications and implementation techniques were compared in Figure 14, to see which best meets the requirements. The TTL serial camera was set as the baseline because it has been used in other similar-sized pipe robots [7].

Requirements	Weight	TTL Serial [28]	CVBS [29]	USB [30]
Implementation Time	5	0	1	0
Part Availability	4	0	-1	1
Video Quality	4	0	0	1
Electronic Miniaturisation	4	0	1	1
Radiation Sensor Digitization	3	0	0	1
Range	3	0	0	-1
Tether Thickness	2	0	0	1
Mapping Accuracy	2	0	0	0
Controllability	1	0	0	1
Cost	1	0	-1	0
Total		0	4	15

Figure 14: Video Camera Selection Pugh Matrix

The result totals from Figure 14 show that the USB endoscope camera best meets the requirements for the PIBAIR system. It should be noted that this Pugh Matrix considers the communication systems that would be required to implement each type of camera. Therefore, the USB camera does have a longer implementation time than the composite video (CVBS) because the USB hub PCBs need to be designed. The CVBS camera would require a capture card and an extra yellow RCA connection in the tether, making it thicker. The TTL serial camera module has a similar implementation time to the USB camera as the camera control PCB would require modification to attach it seamlessly to the PIBAIR module.

Sticking with the USB camera previously used for the PIBAIR prototype means the USB communication system chosen in part 4.1.2 can be used. This comes with the advantages of easier radiation sensor digitisation, availability as one is already in the

laboratory and greater image quality which helps to meet the control objectives. As the camera decision was in parallel with communication system and the USB endoscope camera was clearly the best choice this helped make the decision to use USB for communication. With the 30m USB extender cable solution solving the range issues of USB, the USB endoscope camera will now be used in the new PIBAIR system.

4.1.3 Mapping System

An accurate IMU and encoder must be selected to determine the robot's orientation in the pipe to reach the mapping system objective. One IMU will be implemented into the first motor control module. Nancekievil's research [16] suggests that the MPU-6050 IMU has "the greatest consistency of output for an accelerometer" when exposed to ionising radiation up to the failure TID. The MPU-6050 also has an excellent supporting library and small footprint, making it a suitable choice for the system. Unfortunately, the MPU-6050 has been decommissioned and cannot be used as this project needs to be commercialised.

The MPU-6050 IMU, the 6-axis version of the 9-axis MPU-9250, has now been updated and replaced by the ICM-20948 IMU [31]. This new IMU has greater accuracy than the MPU-6050 and includes a digital motion processor (DMP), which handles accelerometer, gyroscope and magnetometer data fusion, removing the need for a complementary filter [32]. The ICM-20948 has continuous automatic self-calibration, so the system does not need to remain still on start-up [2]. Tilt correction and a familiar library will reduce the software development time for the mapping system as the yaw direction will not change if the motor control module rolls over in the pipe. The only comparable IMU is the BNO085, which has many of the same features as the ICM-20948, but it does not meet availability requirement. As the same manufacturer makes the ICM-20948 as the MPU-6050, there is a possibility that it will have the same resistive characteristic to ionising radiation. Therefore, the ICM-20948 IMU on the Adafruit breakout board will be the IMU used for the mapping system. Further radiation resistance testing will need to be conducted to validate that the ICM-20948 output values are not heavily affected by ionising radiation. With this IMU having continuous automatic self-calibration, this may cause the results to differ from the MPU-6050 [2].

An encoder attached to a roller at the opening of the pipe will be used to measure the distance that the tether goes into the pipe. During cornering, the robot modules have to move back and forward, which could potentially cause the tether to move in and out of the pipe opening. Therefore, a quadrature encoder must be used which can tell the direction it is rotating into account for this back-and-forth movement. The robot will only pull the tether through the encoder rollers at a maximum rate of 2cm/s, which is below the max rpm of most rotary encoders. The B2-CWZ3E 6mm shaft rotary encoder by Yumo [33] has been selected for the tether encoder. This encoder can connect directly to a microcontroller and has a high resolution to achieve a mapping accuracy of ± 2cm. Due to the current electronic component shortage, it was the only encoder with the above specification that meets the availability requirement.

4.2 Hardware Design

As there are no commercially available solutions for a 2” pipe robot, a lot of the hardware for the PIBAIR system has to be custom as it cannot be bought. Therefore, hardware must be designed and manufactured to miniaturise the electronics and produce the mapping system.

4.2.1 Motor Control Module

For the motor control module, a PCB must be designed that connects the IMU, motor drivers and microcontroller together. It then must be attachable to the tether and motors in the PIBAIR module. The PCB with protective housing must be able to fit around pipe geometry to meet the range objective and, therefore, cannot be wider than 40mm and longer than 55mm.

It was first attempted to implement all motor control modules and IMU components onto one PCB to share power componentry. As debugging issues on a small PCB would be challenging, the larger board in Figure 15 was first designed. This board's schematic and further designs can be seen in Appendix C, Figure 37 and Figure 38. This board attempted to integrate the chips from the IMU and motor driver breakout boards directedly onto it. With the components being spread out and not underneath the microcontroller, connections could be tested with a multimeter and oscilloscope [2]. The larger PCB was manufactured in the university electronic workshop and then assembled for testing as shown in Appendix C, Figure 39.

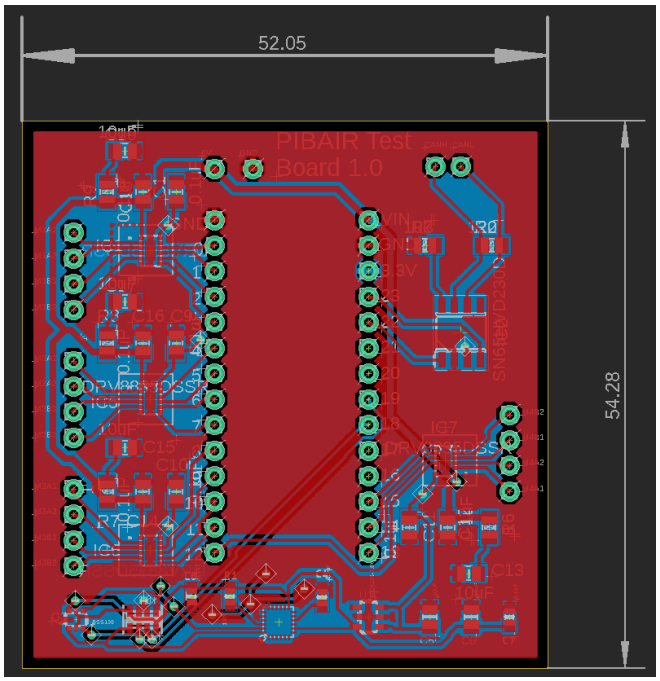

Figure 15: Large Test Autodesk Eagle Board Design [2]

Unfortunately, when testing the larger board, non-of the motor driver chips and IMU worked as expected. Using an oscilloscope and multimeter, signals from the microcontroller to the chips were correct. The power electronics for each chip were also providing the correct voltages and current. Due to the chip shortage, it was possible to purchase the chips for the IMU and motor drivers separately. This means it is not possible to get a pick and place machine to assemble all the PCB components. Therefore, the chips were de-soldered and re-soldered by hand from their breakout boards. As the IMU chip is an electro-mechanical device, it is very fragile. During de-soldering and re-soldering, the chip must have been damaged, causing it to produce no output. It was also challenging to solder chips by hand, which can cause improper connections. This was expected to be the main issue that caused neither the motor driver chips nor IMU to work.

With the test board showing it is challenging to manufacture a PCB with the chips directly on it, another PCB was designed, which the motor driver and IMU breakout boards could attach. The new board design in Figure 16 allows the motor driver breakout boards to attach to one side using short header pins and the Teensy microcontroller to the other. For the front motor control module, the IMU attaches to PCB with long header pins between the motor driver boards, so it sits above them. The board's schematic can be seen in Appendix C, Figure 40. This new stacked design allows the PCB to be as small as 24mm by 45mm as components are attached in other planes. This leaves space for the motor control module housing to go around it and still meet the miniaturisation requirements. Using the breakout boards meant it was much quicker to manufacture and assemble the motor control module as no chips had to be de-soldered and re-soldered.

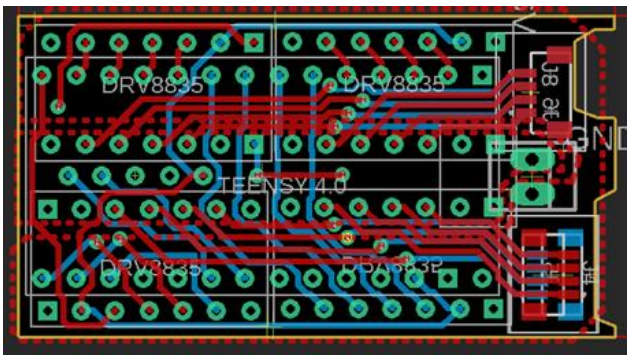

Figure 16: Miniaturised Motor Control Module Final Autodesk Eagle Design

The PCB has the same connections between the motor driver electronics and Teensy pins as the schematic shown in Appendix B, Figure 34. This meant the previous motor control code by Nicholas Castledine could be used as the connections to motors in the robot are coherent. Rory Turnbull then designed the USB hub PCB shown in Appendix C, Figure 41, which attaches to the end of the motor control PCB, as shown in Figure 17. The USB hub chip had to be implemented onto a separate PCB as there would not have been enough space on the motor control PCB without making it wider than the miniaturisation requirements. Notches and a 90 degree power header was added motor control PCB to allow the USB hub PCB. Picoblade Molex connectors were added to connect to the motors. Attaching the USB hub PCB to the end of the board also allows the Teensy's micro-USB port to be used.

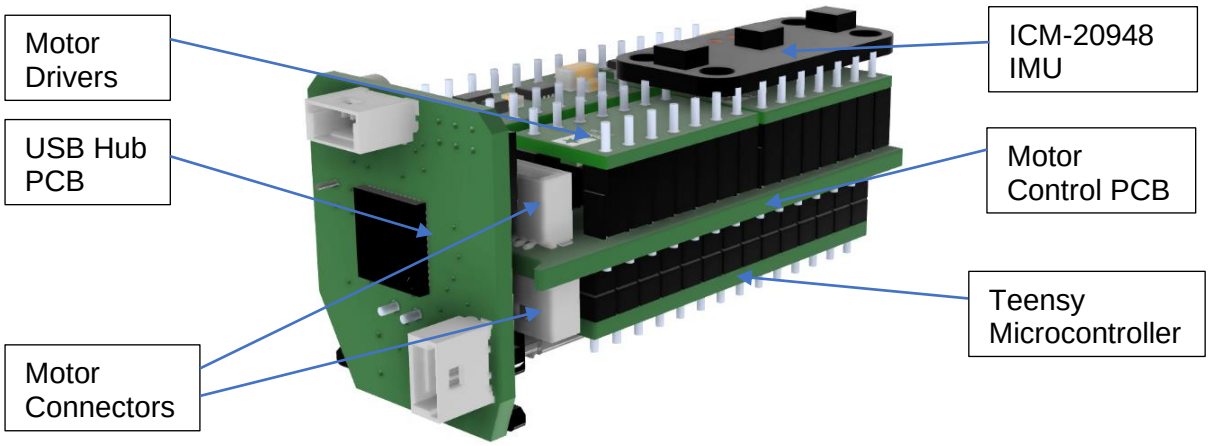

Figure 17: Motor Control Module Render

Another motor control board was manufactured without pins for the IMU, which would be used to control the second PIBIAR module. The stacked design of the motor control and USB hub PCBs was tested and found to be fully functional. The USB hub board allowed two

PIBAIRS modules to be controlled over the same USB cable and IMU data to be sent back to the computer for the mapping system.

During preliminary testing, the motors in the PIBAIR modules that contract the forward and backwards legs could break the mechanism if they were run when the legs were already fully contracted. Therefore, the PIBAIR module design was updated to include switches, which would trigger once the legs were fully contracted. These switches needed to be connected to the microcontroller to stop the contraction motors from running when the legs were fully contracted. With the time constraints of this project, it was not possible to get a new PCB manufactured with connections for the new switches. Therefore, wires connected to switches were soldering directly onto the Teensy pins as shown in Appendix B, Figure 35. After further testing the switch mechanism and final assembly, the design of the motor control module, shown in Figure 18, was complete.

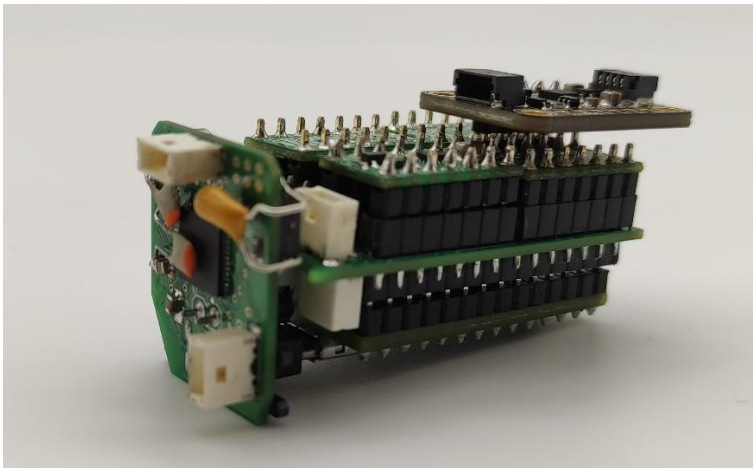

Figure 18: Assembled Motor Control Module

4.2.2 Tether Encoder

To reach the mapping objective, the distance of the tether that goes into the pipe needs to be measured and combined with the IMU data. The encoder selected in part 4.1.3 needs to be attached to the pipe's opening where the tether goes into. Therefore, a casing needs to be designed and manufactured to clamp around the pipe's opening to position the encoder in the correct place.

With an iterative design process and 3D printing, the tether encoder casing in Figure 19 was created using Fusion 360 CAD software. This design went through three iterations, with each one adding extra features and sizing adjustments.

Figure 19: Tether Encoder Casing Design

An exploded view of the tether encoder parts can be seen in Figure 20. The M3 casing bolts connect the two halves of the casing together around the end of the pipe. Spring have been added to the casing bolts. This allows the force that rollers clamp onto the tether to be adjusted. The springs also allow the housing to open wider when the thicker USB redriver chip in the 30-meter USB extender cable passes through the rollers. Metal inserts are required to repeatably screw the bolts into the bottom half of the casing. If the bolts were just screwed directly into the plastic, they would eventually wear down the threads in the plastic, causing them to become loose. M5 bolts on the casing sides lock it to the pipe to stop it from moving as the tether is pulled through. These locking bolts also allow the tether casing to attach to pipes smaller than 2" in diameter to accommodate for the narrow openings that PIBAIR modules can fit through. Finally, bearings allow the 3D printed knurled rollers to rotate freely and not resist the PIBAIR modules pulling the tether through.

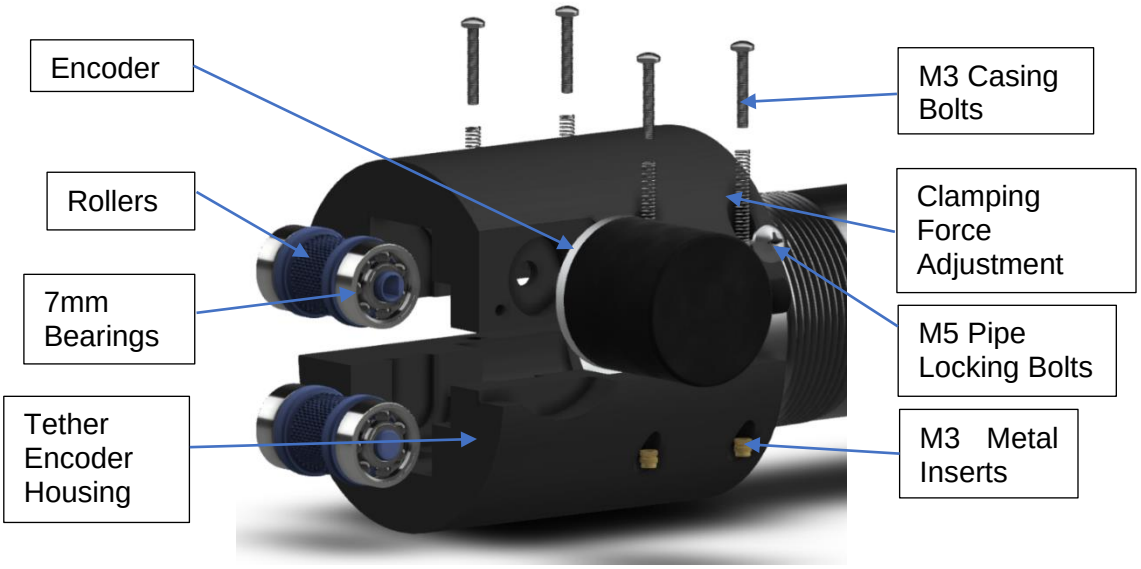

Figure 20: Tether Encoder Exploded View

3D printing made prototyping the tether encoder in a short time frame possible. The final printed design is shown in Figure 21. To 3D print, the design the slicer software, Cura, is used to break up the model into layers and to adjust print settings like infill density. To attach the metal inserts, they heated up with a soldering iron and pressed into holes in the plastic. The rollers required many prints to get them to rotate freely, and multiple sizes were made to adjust the gap that the tether goes through. Grip tape was added to the 3D printed rollers to help stop the tether from sliding over them.

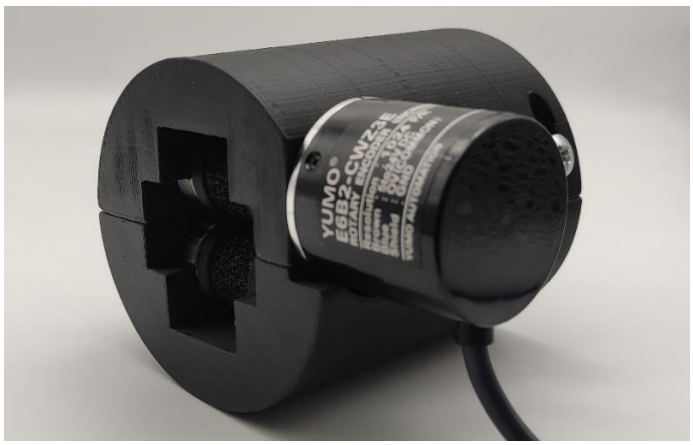

Figure 21: Assembled 3D Printed Tether Encoder Casing

4.2.3 Control Tablet

In the look towards commercialisation of the PIBAIR system, extra work has been conducted to design a touch screen control tablet to operate the robot, as shown in Figure 22. This will remove the need for a computer and Xbox controller and allows the operator to control the robot from one handheld device, helping to achieve the control objective. This control tablet will display the GUI and run USB drivers to read the radiation sensor in real-time. There are currently no commercially available Linux based touch screen tablets that have the correct I/O to connect to the PIABIR system, which is why one has been designed.

Figure 22: Control Tablet

As shown in Figure 23, the control tablet uses an 8.9” 1440p touch screen [34] and a display driver board which allows devices to connect to it via HDMI. The tablet is powered by the Raspberry PI Compute Module 4 (CM4) [35], a version of Raspberry PI 4 without any I/O ports. The CM4 is connected to a carrier board [36] which is a PCB with custom I/O ports. This allows the device to be much slimmer as only the I/O ports required to connect the tether and display driver board are implemented.

Figure 23: Control Tablet Internals

The tablet casing shown in Figure 24 was 3D printed. It has two USB A ports, one for the tether and one for the tether encoder. A USB C port powers the device. Eventually, the external power electronics for the PIBAIR system will be implemented in the control tablet as well.

Figure 24: Assembled 3D Printed Control Tablet

4.3 Software Design

To meet the control and mapping objectives, the microcontrollers of the PIBAIR system and GUI required programming. The previous prototype of PIBAIR used Arduino sketches (C++) to program the microcontroller and the Processing IDE (Java) to build a basic GUI. Processing is a graphical library that can be used to connect to microcontrollers from a computer and read an Xbox controller input.

Within the robotics industry, normally the Robot Operating System (ROS) framework is used for simultaneous localisation and mapping (SLAM) of robots as it also includes tools for visualising data. In a previous project by Evans [37] an Extended Kalman Filter (EKF) was implanted using ROS to fuse IMU and movement data from an encoder. Therefore, it could be helpful to use ROS to program the PIBAIR mapping system. However, the PIBAIR system only needs to map the pipe network. It does not require any localisation of the PIBAIR modules as they are controlled by an operator and not autonomously. ROS is a highly complex framework and requires a network to be set up for different devices to communicate with each other, which is very timely. As this system does not require all the ROS features, the Processing IDE will be used for the mapping system, which means the previous system code can be carried over[2].

This project previously had no online repository for version control and collaborative programming. Therefore the PIBAIR GitHub repository by Evans [38] was set up, which will allow future contributors to this research project to carry on the work. The Arduino sketch code was moved over from the Arduino IDE to the Visual Studio Code IDE (VS Code) using the PlatformIO extension. Another VS code extension allowed the Processing IDE code to be moved over as well.

4.3.1 Microcontroller Code

The PIBAIR system has three microcontrollers for this research project, each with different control code. One microcontroller reads the tether encoder, and the other two are for the motor control modules. As the communication system uses USB, serial port communication is used to send data between the microcontrollers and GUI running on the control tablet.

The code for the first motor control module is described in the flow diagram in Figure 25. This module has an IMU and sends this data via serial to the control tablet. The GUI then sends motor PWM values via serial back to the motor control module. Therefore, a serial handshake must occur between the GUI code and the microcontroller code to allow data to be sent back and forth.

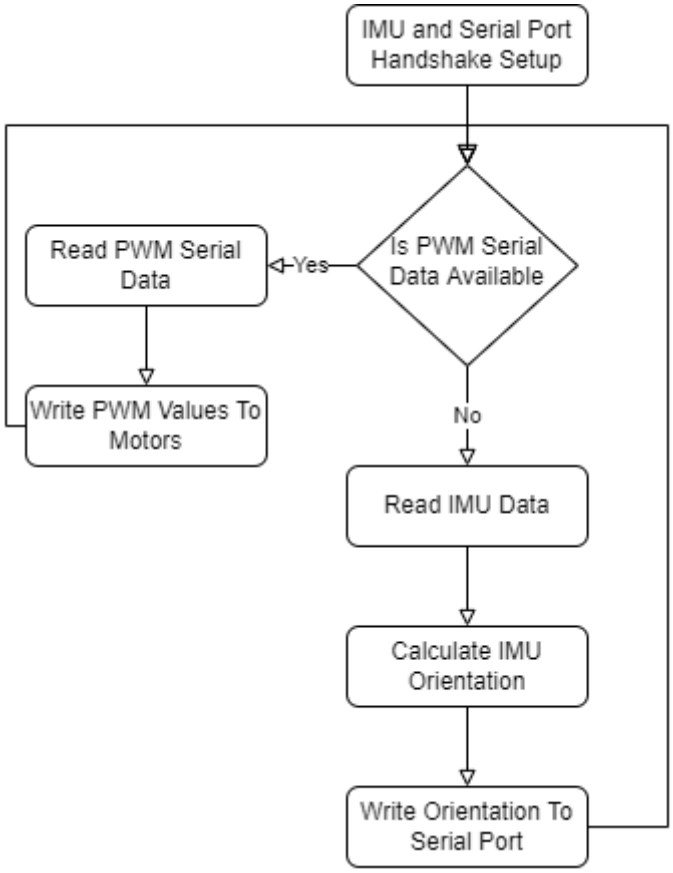

Figure 25: Motor Control Module with IMU Program Flow Diagram

To calculate the orientation of the robot module, the SparkFun ICM-20948 library example [39] was modified first to get the roll, pitch and yaw angles of the IMU. Then based on these angles, the direction of the IMU relative to the starting direction was then calculated. Originally it was thought that the method used by Lim et al. [13] would be required to detect a turn as their IMU values tend to drift over long-time frames. They calculated the change in direction by looking at the difference in the angles over a set time frame when a turn was made. If you were just looking at the absolute angle value, if this value were to drift, this would cause an incorrect turn to be detected. This method would have been very challenging to implement as the PIBIAR system can take a varying amount of time to make a t-section turn. However, over a two-hour test, it was found that the output angles did not drift significantly for the selected ICM-20948 IMU. With DMP activated in the IMU to enable continuous auto-calibration of the IMU sensor, this made IMU accurate over longer time periods. Therefore, the direction that the IMU is pointing in could be calculated just by looking at the absolute angle of the IMU. If the direction changes, this indicates the motor module has made a turn. To validate this method, 50 random 90 degree turns of the IMU were made over a two-hour period, and it detected the correct turn direction each time.

The IMU was set up only to calculate the roll, pitch, and yaw angles using the accelerometer and gyroscope values. Using the magnetometer as well, which is meant to help stop the values from drifting, slowed down the yaw angle response when a direction change was made. It also meant the direction that the IMU was in was relative to its direction in the earth's magnetic field and not to the direction that it entered the pipe. It was also found by Nancekievill [16] that the magnetometer is the most affected sensor in an IMU by ionising radiation, so it is better to avoid using it.

The second motor control module does not need an IMU. Therefore, only motor PWM data is being sent to that module. The previous PIBIAR prototype made use of the Firmata library, which handles the serial communication between the control tablet and microcontroller. This library turns the microcontroller into a slave device so the Processing code can directly control it. Therefore, the code for the second microcontroller just uses the Firmata library example to allow it to be controlled by the Processing code. The code for the tether encoder microcontroller simply reads the encoder counts and the direction it is rotating in. If the encoder rotates in the direction of the tether moving into the pipe it increases the total count value. If the encoder rotates in the direction when the tether moves out of the pipe, this decreases the total count value. The total count value is then continuously sent over the serial port to the control tablet.

4.3.2 Mapping & GUI

The mapping and GUI code were implemented in processing using multiple of its libraries to read the input control, draw a 3D map, display the video data, read the radiation sensor and tether encoder, and communicate with the motor control modules. The flow diagram in Figure 26 describes the overall software architecture of the whole PIBAIR system.

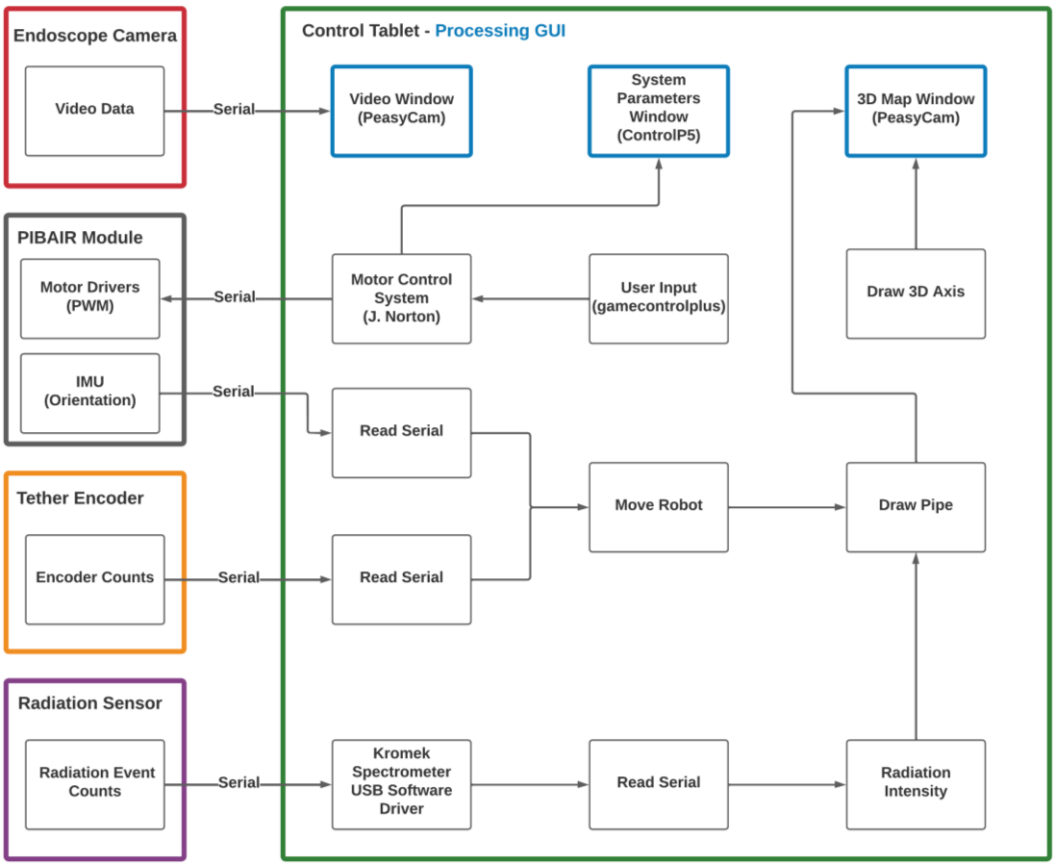

Figure 26: Overall Software Architecture Flow Diagram

For the mapping system to fuse the encoder and IMU data, the code first checks if there has been a set change in the total encoder count value. This set change is defined by the number of encoder counts per centimetre. If the tether has moved into the pipe by a centimetre, the robot's position is updated depending on the direction it is pointing in. The IMU is behind the front PIBAIR module in the pipe. Therefore, the position for the robot mapping is offset by the distance between the two. This is also the case for mapping the radiation intensity data as the radiation sensor is behind all the PIBAIR modules, so its position is offset from the IMU value.

The blue blocks in Figure 26 are responsible for the system's GUI, which produces the visual output shown in Figure 27. The PeasyCam library connects to the endoscope camera and produces a video box in the GUI. The ControlP5 library displays the motor control parameter visuals for all the motor PWM values. The PWM values can also be manually changed by clicking on the visuals, and toggle switches can be used to select between which module to control or to control both modules simultaneously. The PeasyCam library was used to allow the operator to move around, zoom and rotate the 3D map of the pipe network.

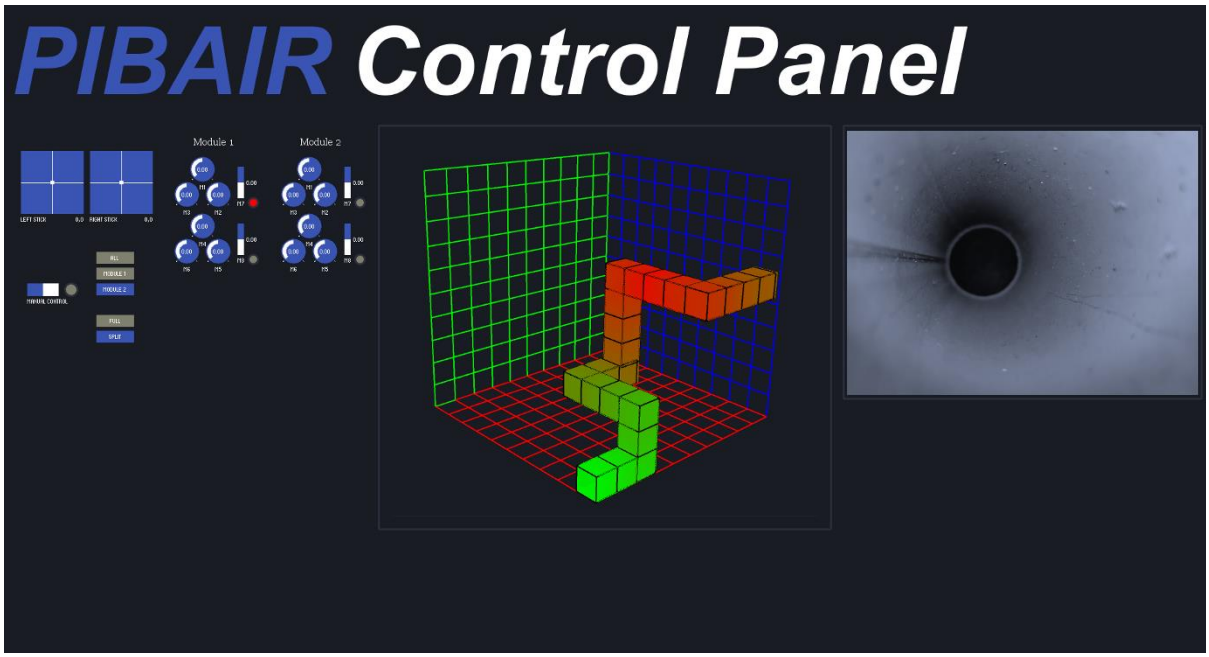

Figure 27: GUI

4.3.3 Linux

The control tablet runs the Raspbian operating system, which comes with touchscreen drivers. To make the Processing GUI code compatible with the Linux based operating system, the video camera code had to be modified as the Linux driver for reading USB cameras is different to windows. Drivers had to be installed to allow the Xbox controller to work with the control tablet. Touch screen driving controls will be added in the near future to the GUI to remove the need for the Xbox controller.

USB drivers for the GR1-A sensor can be installed to read the radiation intensity in real-time. This will allow a program to read the intensity value over the serial port. Unfortunately, the USB drivers are too expensive for the funding that is available for this project. Therefore, the free Kromek Kspec spectroscopy software from the radiation sensor manufacturer will be used to map the radiation. This software records the radiation intensity over time, and then this data can be exported as a CSV file. After driving the PIBAIR modules in the pipe and exporting the data, the Processing code can then read this CSV file and update the radiation intensity data on the 3D map in the GUI. For the map, red areas represent a high radiation intensity, and green areas represent a lower radiation intensity. When PIBAIR has

been commercialised, the USB drivers for the radiation sensor should be purchased to have the intensity shown on the map in real-time.

4.4 Final System Design

Now all the hardware has been selected and designed, the initial system design from part 3.5 has been updated to the final system design block diagram in Figure 28. The final system diagram shows all the connections between the parts of the front motor control module, PIBAIR module, tether encoder and control tablet. A 12V power supply powers the PIBAIR system, the schematic in Appendix B, Figure 36 shows how the voltage is stepped down along the tether and how the shared USB connection goes through the hub chips.

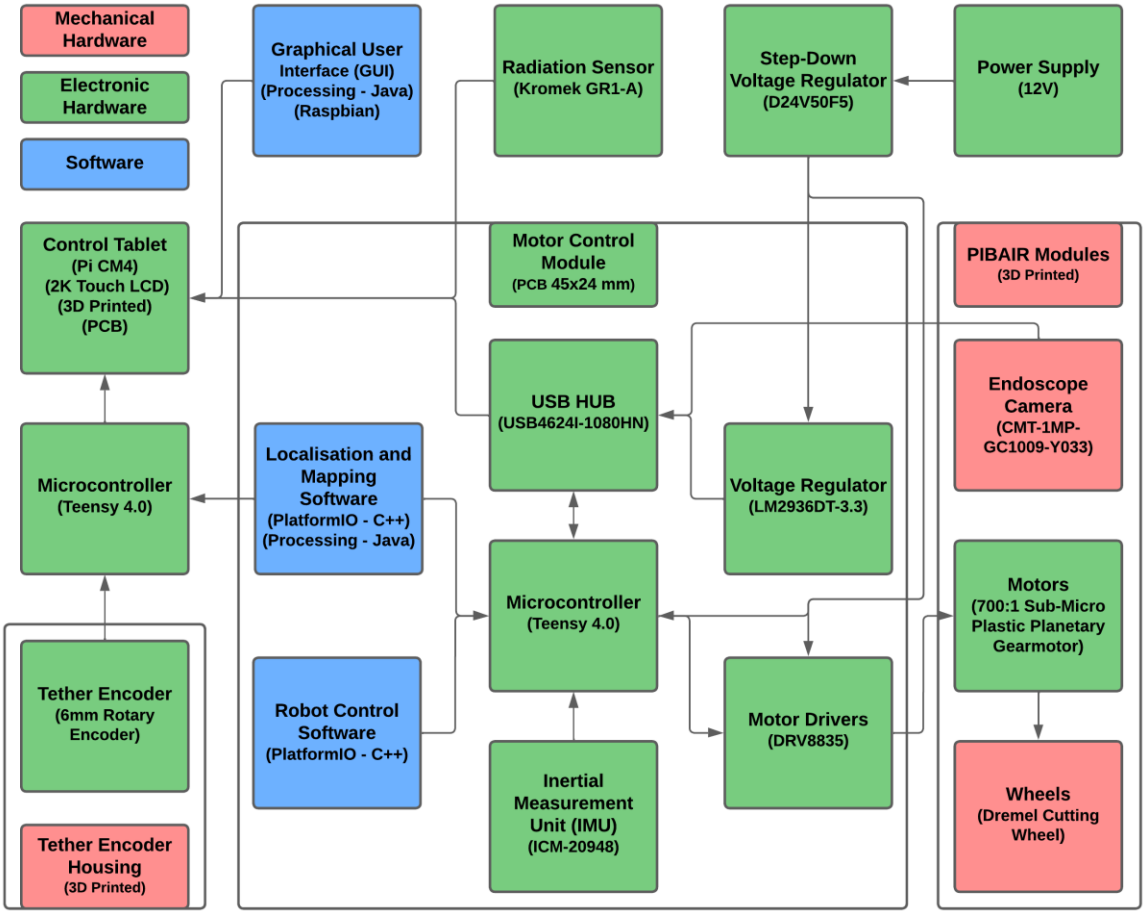

Figure 28: Final System Design Block Diagram

The 30m USB extender cable is used as the tether for the system. The 5V voltage line of the extender cable is disconnected from the USB port, and 12V from the power supply is run down it. The ground line for the cable is kept connected to the USB port as there has to be common ground for the USB commutation to work. Technically a maximum of two 30m USB extender cables can be connected to increase the range of the system. However, as the 30m USB cables weighs 1.2 kg and the maximum vertical pulling weight of each PIBAIR module is 300g, it would require a lot of robot modules to pull the 60m cable. As 30m is the set range requirement, this means that five robot modules should be able to pull the weight of the tether.

In the wider project, Rory Turnbull has been upgrading the mechanical design of the PIBAIR modules, the front module is shown in Figure 29 and the second module is shown in Figure 30. To complete the USB communication system, Rory Turnbull has also designed a

USB hub board for the GR1-A radiation sensor and a 5V regulator board for the endoscope camera shown in Appendix B, Figure 42 and Figure 43.

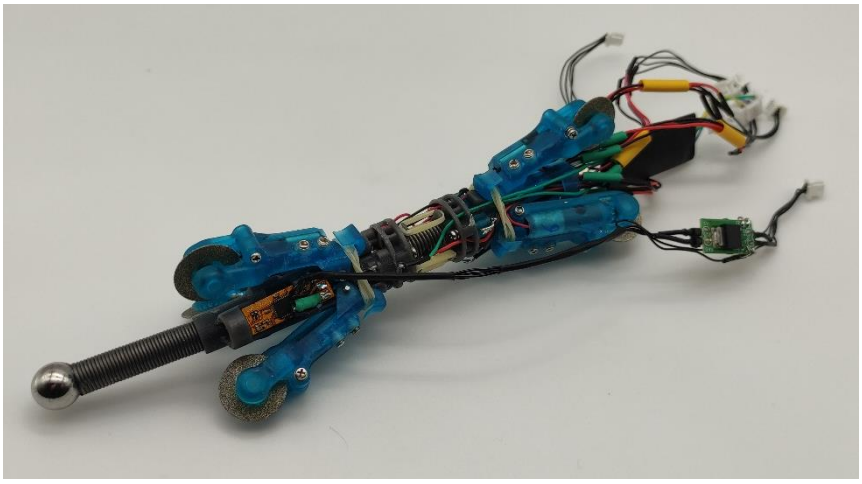

Figure 29: Updated Front PIBAIR Module with Endoscope Camera

The PIBAIR modules now use a screw mechanism to contract the forward and backwards legs to remove the need for fishing line. The legs are assembled using small bolts rather than glue, making the modules easier to repair. The outward pressing force of the legs has increased to improve the traction that wheels have on the pipes. Switches have been added to the contraction mechanisms to tell the operator if the legs are fully closed or open, as they will not be able to see in a closed pipe.

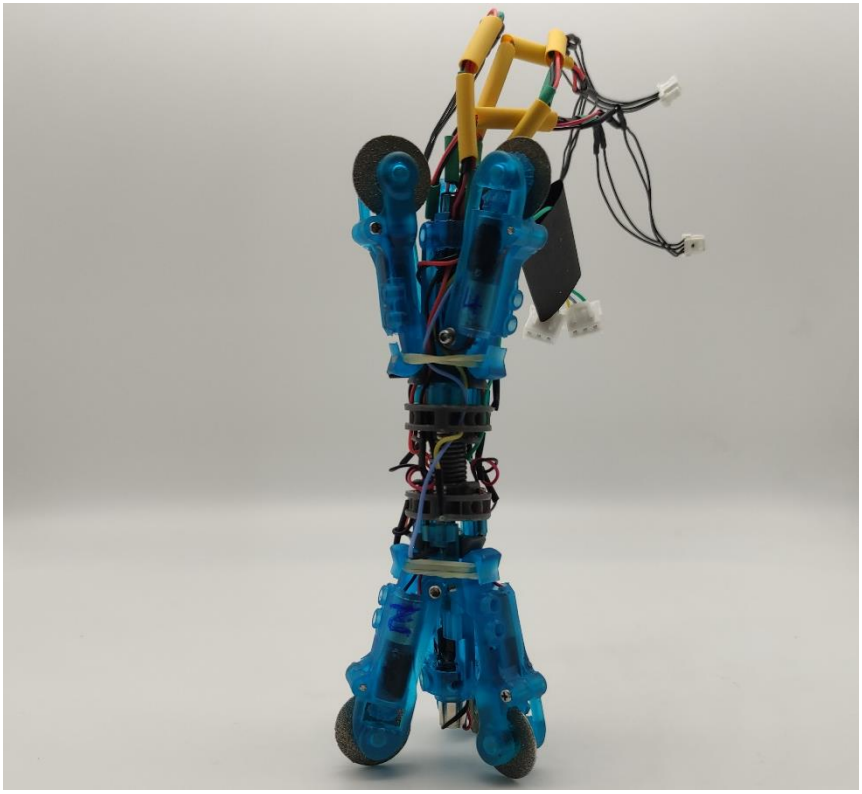

Figure 30: Updated Second PIBIAR Module

4.5 Verification and Validation

Once the system had been fully assembled, it underwent verification and validation testing to verify that it met all the requirements. This included running tests on each part of the system individually and then on the whole system together. These tests will validate the system's ability to meet the mapping radiation and navigation goals.

4.5.1 Radiation Mapping Accuracy

To test if the mapping system has met the accuracy requirement of ± 2cm, the IMU and tether encoder devices were tested separately whilst the rest of the system was still being built. This meant the experimental conditions for each could be controlled. The IMU is only used in the mapping system to tell if a turn has been made in the pipe network. Therefore, it is assumed that as long as the IMU calculates all the correct turns made, the tether encoder is solely responsible for the system's accuracy. This assumption is especially true if the robot makes no turns and just goes through a long straight section of pipe, which at Sellafield there will be many. The IMU test detailed in part 4.3.1 shows that it correctly calculated 50 turns over a 2-hour period.

The only factor affecting the IMU's ability to calculate turns is if the output values change when exposed to ionising radiation. Nancekievill [14] found that the MPU-6050 output values did not vary when exposed to lower levels of ionising radiation, but as this system uses the ICM-20948 IMU, its resistance to ionising radiation may vary. Therefore, a test for ionising radiation's effects on the ICM-20948 will be conducted.

Within the university GM59 laboratory, there is access to radioactive sources that can be used for this experiment. Gamma radiation is the only type of radiation that will affect the IMU as Alpha and Beta radiation are blocked by the motor control module housing. The strongest gamma radiation source available in the GM59 laboratory is cobalt-60.

The gamma dose rate of the cobalt-60 source available is 40.6 µSv/h at 1cm away, which is much lower than was used in research by Nancekievill [16]. Nancekievill [16] showed that some IMU output values can vary with a TID as low as 50 Gy(Si). The cobalt-60 does rate is not high enough to make any of the electronics fail, but it might be high enough to impact the IMU output readings. Therefore, an experiment will still be run with the cobalt-60 source to see if it affects the ICM-20948 IMU.

To begin with a control test was run on the ICM-20948 without any ionising source. It was kept still over a period of two hours, and its roll, pitch and yaw output values were recorded to analyse how much they drifted. The test ran for two hours as the system's operating time would not be longer than this for driving the robot over a 30m section of pipe. The variation in the IMU output values over time are shown in Figure 31. The control test showed no significant variation of any of the IMU output values over two hours, which would cause an incorrect turn to be calculated.

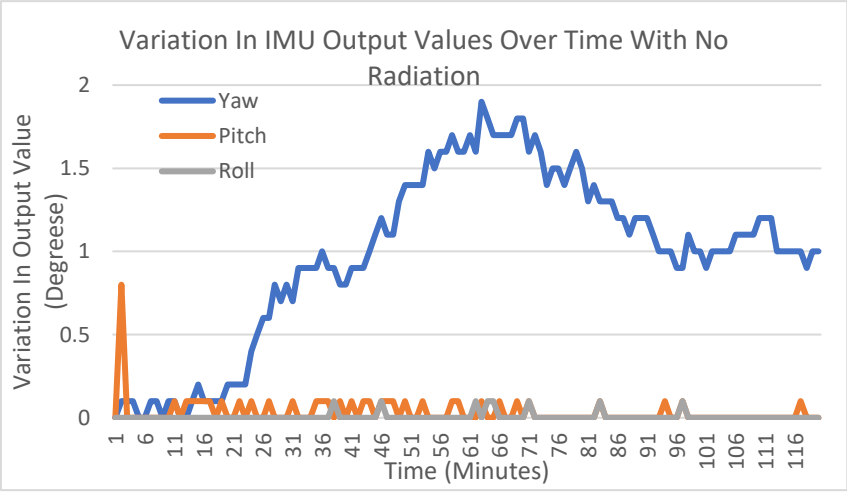

Figure 31: Variation In IMU Output Values Over Time with No Radiation

A time slot in the GM59 laboratory has been booked in the near future to run the test procedure detailed in Appendix D to find out the effects of ionising gamma radiation on the IMU. This experiment will mirror the control test above, and a comparison will be made between the two for the variation of the IMU output values.

To test the tether encoder, the first task was to determine how the force that rollers clamp onto the tether affects the pull-through force of the tether for the PIBAIR modules. If the clamping force of the rollers is increased, this will reduce the slipping of the tether over the rollers, making the encoder readings more accurate. However, the clamping force cannot be too high because it will stop the PIBAIR modules from pulling the tether through. Therefore, a balance must be found for a clamping force that gives a high enough encoder reading accuracy but does not resist the PIBAIR module's movement. A pull-through force of 3 newtons was set as the system's target as this is the same as the pulling force of one PIBAIR module. As there are two PIBAIR modules, this would leave the other module responsible for towing the weight of the tether.

The experimental setup shown in Appendix D, Figure 44, was used to find the roller clamping force effects on the pull-through force. 100g weights were added and hung vertically using a roller which would cause them to apply a constant force on the tether due to acceleration from gravity. This would mean that approximately 300g would be equivalent to 3 newtons of pulling force on the tether. As it would be challenging to measure the clamping force of the rollers, the M3 Casing Bolts in Figure 20 were adjusted till the weights began to move the tether through the rollers. To begin with, the bolts were tightened all the way down. Then the number of counterclockwise (loosening) rotations for the front two bolts was recorded until the weights moved the tether through the rollers. It was found that six counterclockwise rotations were required for the weight to move. This number of rotations will be used for every setup of the tether encoder to have consistent results.

Now the clamping force is set, an experiment was run to find the deviation in encoder counts as the tether was pulled further into the pipe. The investigation would find the average number of encoder counts per meter of the tether, which can then be used to find the system's accuracy. The tether was pulled through the rollers ten times. Markings were put at 5, 10, 15 and 30 meters along the tether, and the total number of encoder counts were recorded at each marking. The average number of encoder counts per meter was calculated for each distance to determine the deviation in the distance measured over the ten tests. The standard deviation in meters for each distance measured was calculated and is shown in Figure 32.

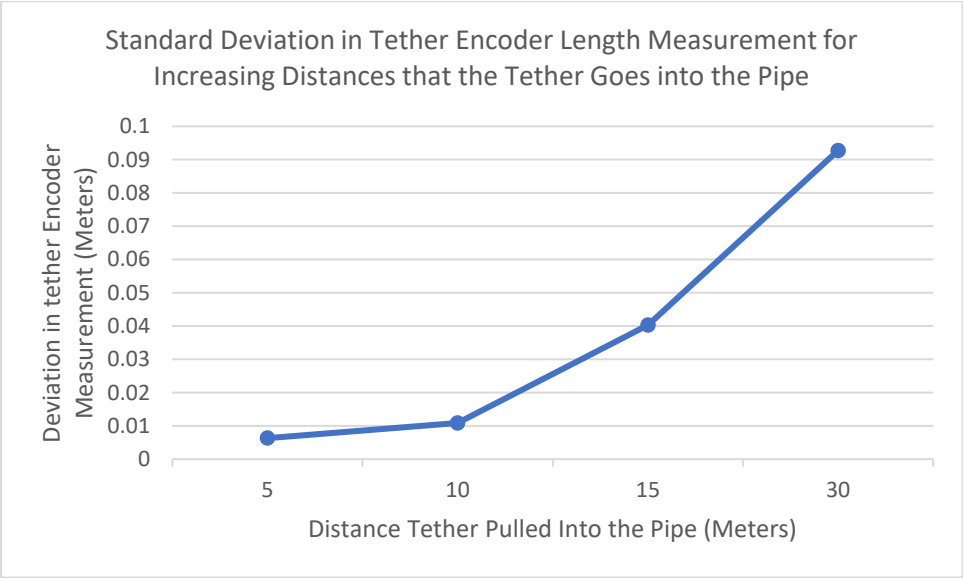

Figure 32: Standard deviation in tether encoder length measurement for different distances that the tether goes into the pipe

The results in Figure 32 show that as the tether is pulled further in the pipe the deviation of the measured length of the tether by the encoder increases. This is expected as the tether will inconstantly slip through the rollers as it is pulled through. Therefore, the longer distance that the tether is pulled through, the more this inconsistency will cause encoder measurement to vary. With the tether five meters into the pipe, the deviation of the length measured by the encoder was only ± 0.006 meters over the ten tests. This deviation is much lower than the target accuracy set in the requirements. As the tether goes further into the pipe, at 30m the deviation increases to 0.093 meters.

The results above show that the mapping system successfully meets the accuracy objectives for driving along a straight section of pipe. However, the real-world performance of the mapping system will vary as the robot makes more turns in a pipe network. This is because the tether will bend unpredictably as it goes round pipe geometry, which can cause the distance that the robot has moved into the pipe to deviate further.

4.5.2 Performance Qualification

To test the mapping system's real-world performance, the test rig in Appendix D, Figure 45 was built. This contains a t-section and a 90-degree bend over two meters of pipe. Testing has begun on the whole system together to validate its real-world performance and make any improvements to the design if issues are found.

The initial testing done so far on the system has successfully shown that it can map pipes when going in a straight line. The 3D map in Figure 33 was produced by driving the robot along the top section of the pipe over the t-section bend. In the GUI, black lines are printed every 10cm and white lines are printed every 1cm on the 3D map to allow the operator to measure how far the robot is in the pipe. The systems mapping accuracy was found to be within 1cm of the robot’s actual position for this initial test.

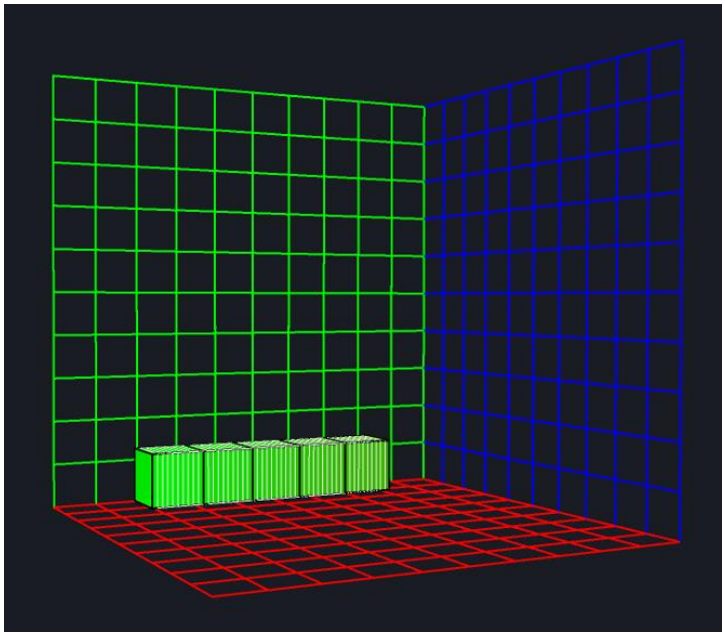

Figure 33: Real-World Test GUI 3D Map

The initial real-world testing of the system has already exposed some issues with the mechanical design of the PIBAIR modules from the wider research project. One problem is that the articulation spring in the middle of the new PIBAIR module is too stiff, making it difficult to go around bends. Different springs with varying stiffness will be tested to find the optimal one for going around bends. Once these updates have been made to the PIBAIR system, further testing will be conducted to determine how going round bends affect the accuracy of the mapping system. However, real-world testing has already validated that the GUI mapping system is fully functional and allows full control of both PIBAIR modules simultaneously.

4.6 Future developments

As the work conducted above is part of a research project, testing will continue in the Real Robotics laboratory until the system is fully optimised and further data is collected for the Pipebots journal paper. The following future developments for this part of the project will take in the coming months in preparation for the research projects demonstration:

1. **Ionising radiation testing**: The IMUs resistance to ionising radiation will be tested in the GM59 laboratory.

2. **Improve second PIBAIR module integration:** The system currently requires the operator to control the retraction of the legs for the second PIBAIR module when going round t-section bends. Therefore, the legs of the second module need to be programmed to automatically contract as the operator cannot see them in enclosed pipes. As the distance is known between the two PIBAIR modules, once the first PIBIAR modules have travelled this set distance after a turn, the second module can contract its legs and then be pulled around the bend by the first module.

3. **Digital touch screen controls**: Further touch controls will be added to the control tablet to remove the need for an Xbox controller.

4. **Enclose external electronics**: The power electronics and tether encoder microcontroller must be integrated into a single 3D printed box.

Once the above has taken place, the PIBAIR system will be taken to the University of Bristol's Fenswood facility, which has a large pipe network. There is access to the GR1-A radiation sensor at this facility, so it can be tested with the current system. Further real-world testing will take place to see if the system can find radioactive sources in the pipe network. The maximum range that the system can travel with two PIBAIR modules will be tested to validate the overall design. As more data is gathered, the mapping systems settings can be tuned further in an attempt to increase the accuracy over longer distances and more bends.

If the PIBAIR system is successful at Fenswood, work will be conducted to try to commercialise PIBIAR to gain further funding for the research project. The modules will be printed out with tougher resins that are more resistant to nitric acid. The system will also be tested in contaminated pipes to simulate the conditions at Sellafield. Further radiation resistance testing with stronger sources will take place at the University of Bristol’s Hot Robotics facility. The sources available in the GM59 laboratory were not strong enough to test the electronics to failure. At the Hot Robotics facility, there will be access to strong ionising sources that can be used to simulate the environment at Sellafield more accurately. This will help identify if radiation-hardened electronics need to be integrated into the motor control module. If the system were to be sold as a commercial product, the external power electronics should be integrated into the control tablet to finish the product and make it feel more polished.

5 Conclusion

The previous PIBAIR platform has been updated to be able to map and navigate a pipe network. Hardware has been designed and selected to enable the system to travel 30 meters into a pipe network and accurately map radiation up to ± 0.6 cm.

The motor control electronics have been miniaturised into control modules which are connected via USB. Multiple PIBAIR robots can work collaboratively to pull the radiation sensor through a pipe network, and the whole system can be controlled from a touch screen tablet. A GUI has been developed to display video, 3D mapping and radiation intensity data. 3D printing has been used extensively throughout the project to prototype design ideas and manufacture protective casings.

The effects of ionising radiation on the system's electronics have been evaluated. Further testing with a radiation sensor in a large pipe network is planned to take place at the University of Bristol's Fenswood facility. Radiation resistance testing will also be conducted at the University of Bristol's Hot Robotics facility, which will help move the PIBAIR system towards commercialisation for use at the Sellafield nuclear power station.

During this project, skills have been developed in PCB design, CAD modelling, 3D printing and embedded programming. Systems engineering has been used throughout to validate that the design of the final robot meets the project objectives and requirements. It was found that developing a custom mapping system and GUI using the Processing library was much faster than using ROS. A radiation permit has been obtained by completing nuclear radiation handling training to allow future testing with ionising sources in the GM59 laboratory. Additional work has been conducted to build a Linux based control tablet that will enable real-time mapping of nuclear contaminants.

Initial real-world testing has validated that the mapping system and GUI are fully functional. Further testing will be conducted to determine how the mapping system's accuracy is affected as more turns are made in a pipe network.

6 References

[1] H. Bayram. (2020, Aug 5). Development of pipe inspection robot. [Online]. Available: https://rainhub.org.uk

[2] B. Evans, "Nuclear Pipe Inspection Robot," January 2022.

[3] G. H. Mills, A. E. Jackson, and R. C. Richardson, "Advances in the Inspection of Unpiggable Pipelines," *Robotics,* vol. 6, no. 4, 2017, doi: 10.3390/robotics6040036.

[4] J. Norton, "The Design and Development of a Mobile Colonoscopy Robot," Ph.D. thesis, Dept. Mech. Eng., Leeds Univ., 2017.

[5] S. Kazeminasab, N. Sadeghi, V. Janfaza, M. Razavi, S. Ziyadidegan, and M. K. Banks, "Localisation, Mapping, Navigation, and Inspection Methods in In-Pipe Robots: A Review," *IEEE Access,* vol. 9, pp. 162035-162058, 2021, doi: 10.1109/ACCESS.2021.3130233.

[6] A. Kakogawa, T. Nishimura, and S. Ma, "Development of a screw drive in-pipe robot for passing through bent and branch pipes," in *IEEE ISR 2013*, 24-26 Oct. 2013 2013, pp. 1-6, doi: 10.1109/ISR.2013.6695638.

[7] Y. Kwon, B. Lee, I. Whang, W. Kim, and B. Yi, "A flat pipeline inspection robot with two wheel chains," in *2011 IEEE International Conference on Robotics and Automation*, 9-13 May 2011 2011, pp. 5141-5146, doi: 10.1109/ICRA.2011.5979712.

[8] Liam Brown, Joaquin Carrasco, Simon Watson, and B. Lennox, "FURO: Pipe Inspection Robot for Radiological Characterisation.," in *UK-RAS Conference: 'Robots Working For & Among Us' Proceedings*, 2018, pp. 56-58.

[9] Y. Bando *et al.*, "Sound-based online localisation for an in-pipe snake robot," in *2016 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 23-27 Oct. 2016 2016, pp. 207-213, doi: 10.1109/SSRR.2016.7784300.

[10] C. Rizzo, T. Seco, J. Espelosín, F. Lera, and J. Villarroel, "An alternative approach for robot localisation inside pipes using RF spatial fadings," *Robotics and Autonomous Systems,* vol. 136, p. 103702, 02/01 2021, doi: 10.1016/j.robot.2020.103702.

[11] T. B. Schön, J. D. Hol, and M. Kok, "Using Inertial Sensors for Position and Orientation Estimation," *Foundations and Trends® in Signal Processing,* vol. 11, no. 1-2, 2017, doi: 10.1561/2000000094.

[12] A. C. Murtra and J. M. M. Tur, "IMU and cable encoder data fusion for in-pipe mobile robot localisation," in *2013 IEEE Conference on Technologies for Practical Robot Applications (TePRA)*, 22-23 April 2013 2013, pp. 1-6, doi: 10.1109/TePRA.2013.6556377.

[13] H. Lim, J. Choi, Y. Kwon, E.-J. Jung, and B.-J. Yi, *SLAM in indoor pipelines with 15mm diameter*. 2008, pp. 4005-4011.

[14] *Universal Serial Bus, USB 2.0*, 2000.

[15] H. Chen, B. Gao, X. Zhang, and Z. Deng, "Drive Control System for Pipeline Crawl Robot Based on CAN Bus," in *Journal of Physics: Conference Series 48*, 2006, pp. 1233-1237.

[16] M. Nancekievill, "The radiation tolerance and development of robotic platforms for nuclear decommissioning," Ph.D. thesis, Dept. Electron. Eng., Manchester Univ., 2018.

[17] Formlabs. Solvent compatibility. [Online]. Available: https://formlabs.com

[18] S. Burge. (2009, Apr 30). Systems Engineering Tools & Techniques. [Online]. Available: https://www.burgehugheswalsh.co.uk

[19] S. Burge. (2011). "Context Diagram" The Systems Engineering Tool Box. [Online]. Available: https://www.burgehugheswalsh.co.uk

[20] S. Burge. (2011). "Function Means Analysis (FMA)" The Systems Engineering Tool Box. [Online]. Available: https://www.burgehugheswalsh.co.uk

[21] S. Burge. (2006). "Holistic Requirements Model (HRM)" The Systems Engineering Tool Box. [Online]. Available: https://www.burgehugheswalsh.co.uk

[22] GR1-A and GR1,12th ed, Kromek Group plc, Sedgefield , UK, 2016, pp. 4

[23] S. Burge. (2006). “Matrix Diagram” The Systems Thinking Tool Box. [Online]. Available: https://www.burgehugheswalsh.co.uk
[24] S. Burge. (2009). “Pugh Matrix” The Systems Engineering Tool Box. [Online]. Available: https://www.burgehugheswalsh.co.uk
[25] kvaser. (2001, Jan 31). The Can Bus Protocol Tutorial. [Online]. Available: https://www.kvaser.com
[26] *RS-485, TIA/EIA-485*, 1983.
[27] miciwan. (2021, May 19). USB Video Class on Teensy 4.1; Missing isochronous packets. [Online]. Available: https://forum.pjrc.com
[28] Guangzhou Putal Communication Technology, "PTC06 Serial Camera Specification ".
[29] mju-cam, "MuC204 (NTSC) Color Camera Module w/LED Control ". MuC204 datasheet
[30] CM Technology. 4.5MM HD 720P Medical Endoscope Camera Module with GalaxyCore GC1009 Sensor. [Online]. Available: http://www.camera-module.com
[31] InvenSense, "ICM-20948," DS-000189 datasheet, ICM-20948 datasheet, Dec. 2016 [Revised Feb. 2021].
[32] TDK-Invensense. (2018, Sep). Migrating from MPU-9250 to ICM-20948. [Online]. Available: https://invensense.tdk.com
[33] YUEQING YUMO ELECTRIC, "Rotary Encoders." B2-CWZ3E datasheet
[34] Top Electronic Parts, "8.9"WQXGA LCD." WQXGA datasheet, Nov. 2013
[35] Raspberry PI, "Raspberry Pi Ccompute Module 4." CM4 datasheet, Oct. 2020
[36] A. Vostrukhin. Raspberry Pi CM4 TV Stick. [Online]. Available: https://mbs-shop.online
[37] B. Evans, "The Design and Manufacture of an Autonomous Robot to Detect Radioactive Contamination on a Laboratory Floor," May 2021.
[38] B. Evans. (Dec 2021). PIBAIR. [Online]. Available: https://github.com/evanso931/PIBAIR
[39] O. oclyke. (Apr 2019). SparkFun_ICM-20948_ArduinoLibrary. [Online]. Available: https://github.com/sparkfun/SparkFun_ICM-20948_ArduinoLibrary

7 Appendices

7.1 Appendix A

Hardware Selection Calculations:

$$\text{Video data rate (360p, 1FPS, RGB)} = 360 \times 480 \times \frac{24}{8} = 518.4 \text{ kBps} \quad (1.0)$$

$$\text{IMU data rate (Yaw \& pitch every 0.1 seconds,)} = 2 \times 10 = 20 \text{ Bps} \quad (1.1)$$

$$\text{Radiation Intensity data rate (Counts per seconds)} = 4 \text{ Bps} \quad (1.2)$$

$$\text{PWM data rate (5 romobot modules every 0.1 seconds)} = 5 \times 8 \times 2 \times 10 = 800\ Bps \quad (1.3)$$

$$\text{Mininum Communication data rate} = 518400 + 20 + 4 + 800 = 519.2 \text{ kBps} \quad (1.4)$$

7.2 Appendix B

PIBAIR System Wiring Diagrams:

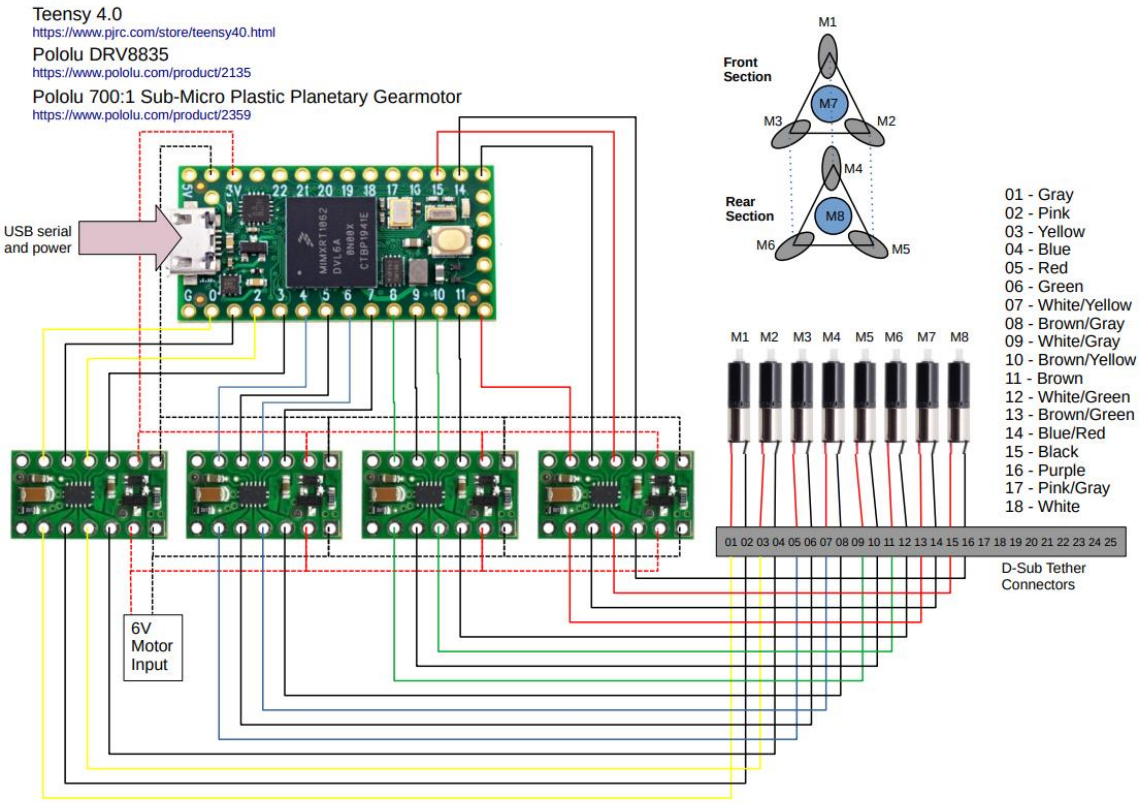

Figure 34: Motor Driver and Connection Schematic - Provided by Nicholas Castledine

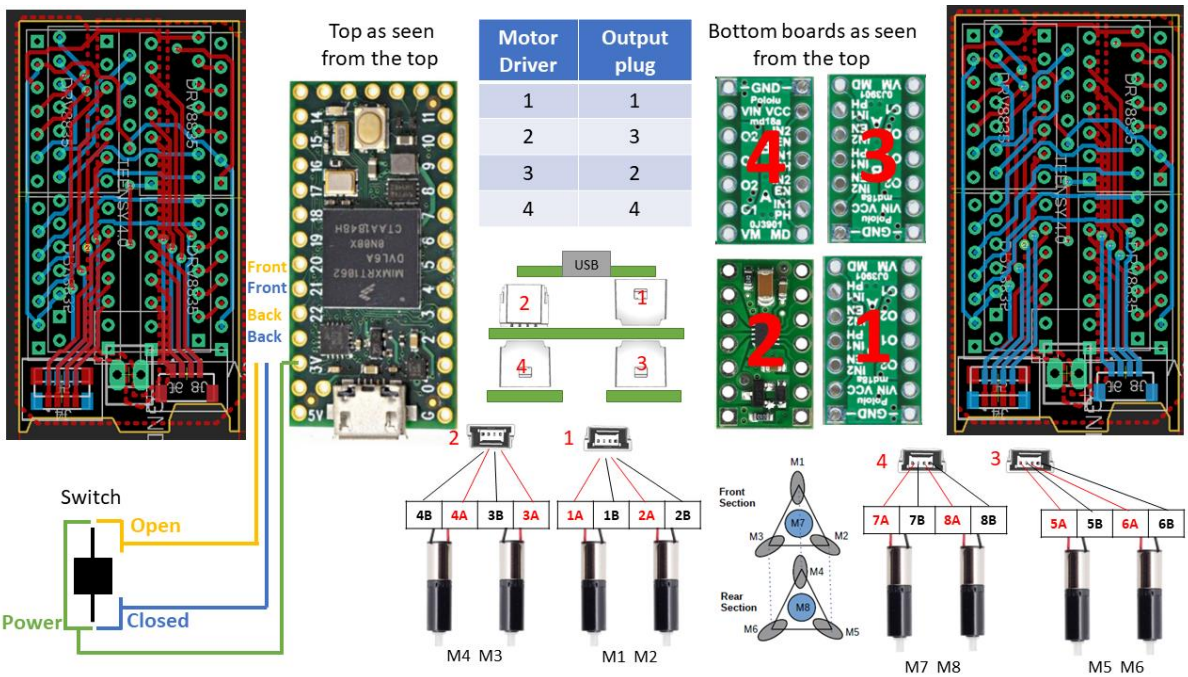

Figure 35: Final Motor and Switch Wiring Schematic - Provided by Rory Turnbull

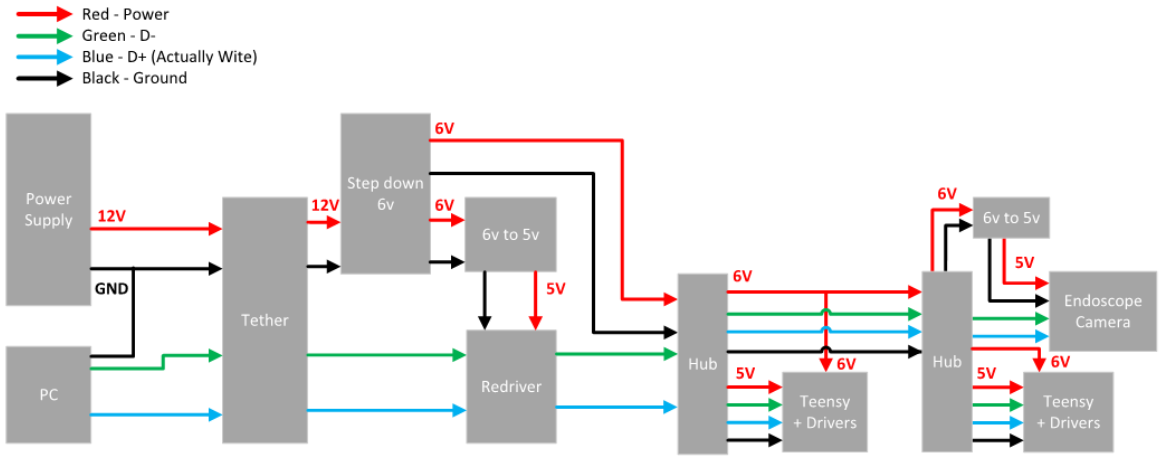

Figure 36: PIBAIR Voltage Flow Schematic - Provided by Rory Turnbull

7.3 Appendix C

PCB Designs:

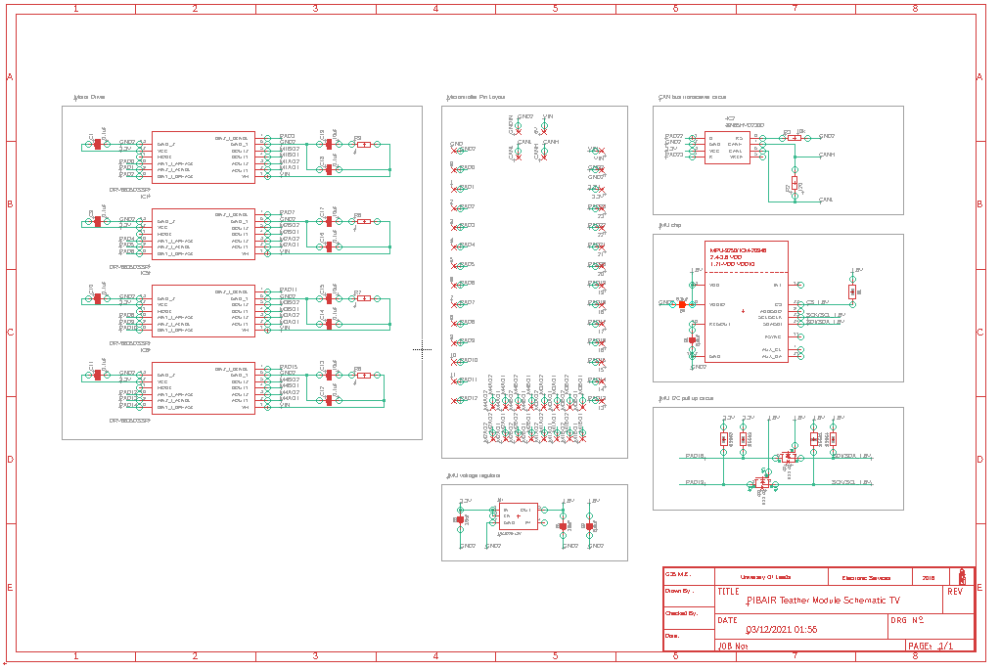

Figure 37: Large Test Board Schematic [2]

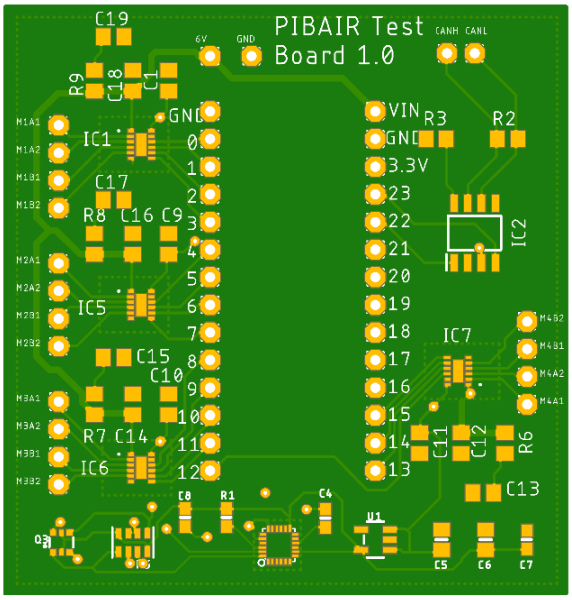

Figure 38: Large Test PCB Manufactured Preview [2]

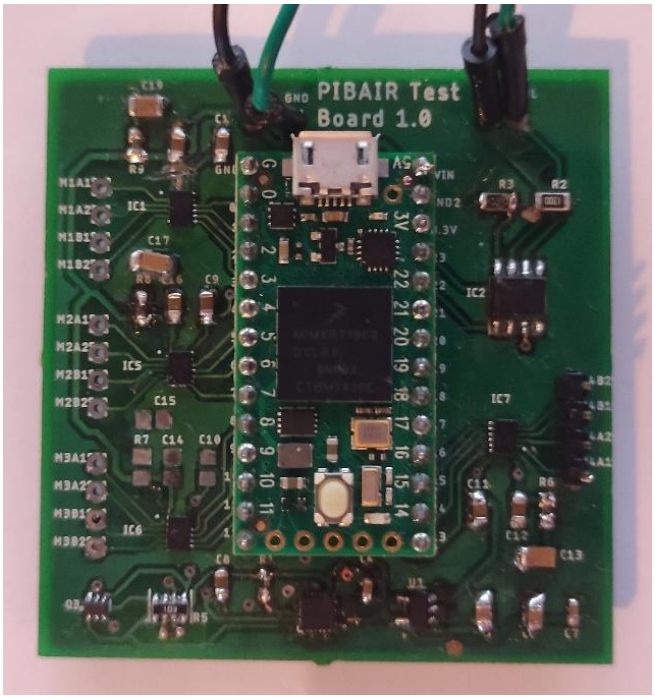

Figure 39: Large Test PCB Assembled [2]

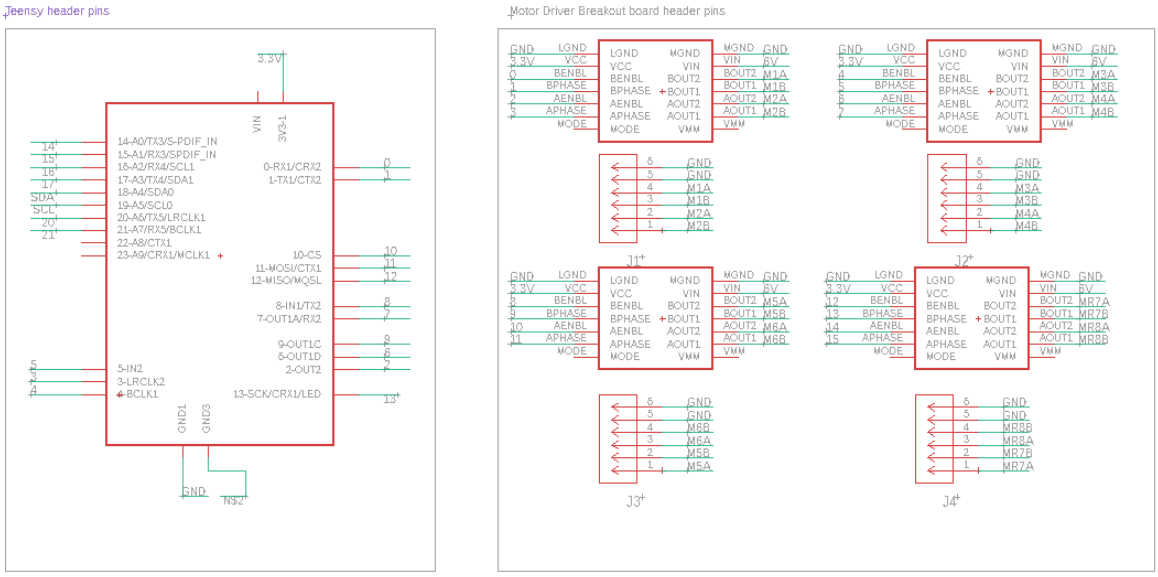

Figure 40: Miniaturised Motor Control Module Final PCB Schematic

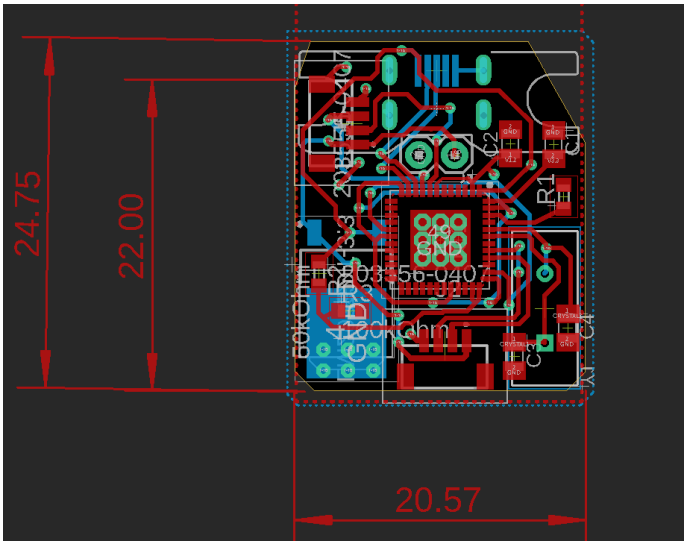

Figure 41: Motor Control Module USB Hub PCB - Rory Turnbull

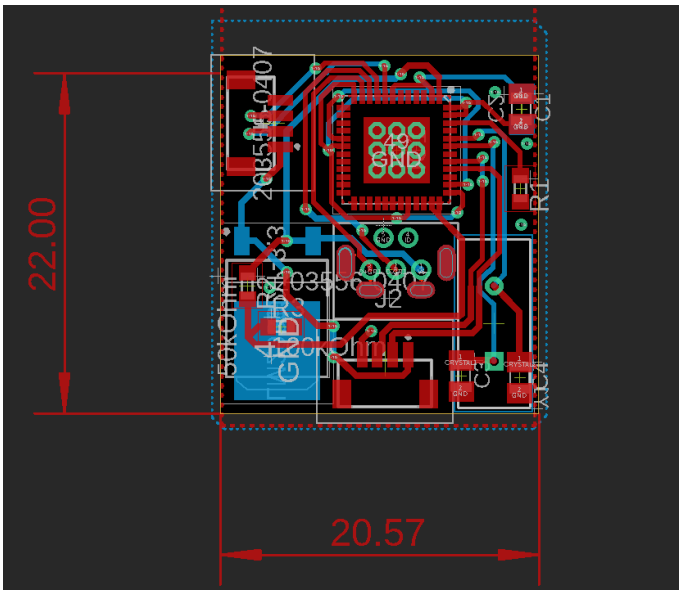

Figure 42: GR1-A Radiation Sensor USB Hub PCB – Rory Turnbull

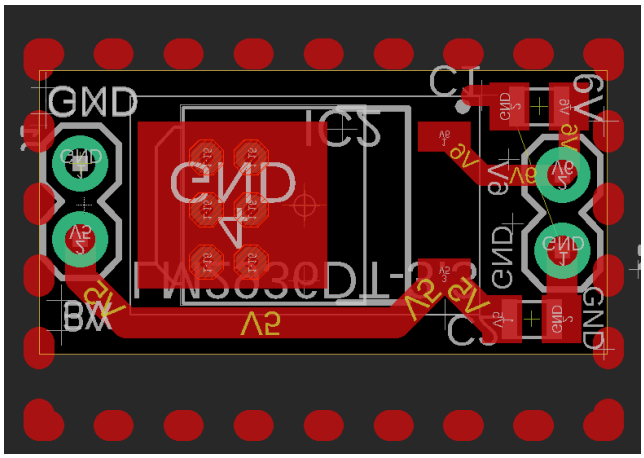

Figure 43: 6V to 5V Voltage Regulator PCB For Endoscope Camera - Rory Turnbull

7.4 Appendix D
Experimental Test Procedures

GM59: Exposure of Robotic Sensor components to radioactive sources test procedure

Risk assessment name; Robot sensor components exposure to radioactive sources

RIVO ID; …….

1. Scan the fume hood for background radiation
2. Set up experimental stage, maybe a clamp stand for sample and active source, potentially placed
3. Retrieve active source from locked cabinet
4. Place source on experimental stage (IMU Sensor Chip) for approximately 2 hours*
5. At the end of duration place source back in cabinet
6. Scan the component to ensure not active
7. Test component on robotics test bed

*Laptop connected to sensor via 3m USB cable to be away from source

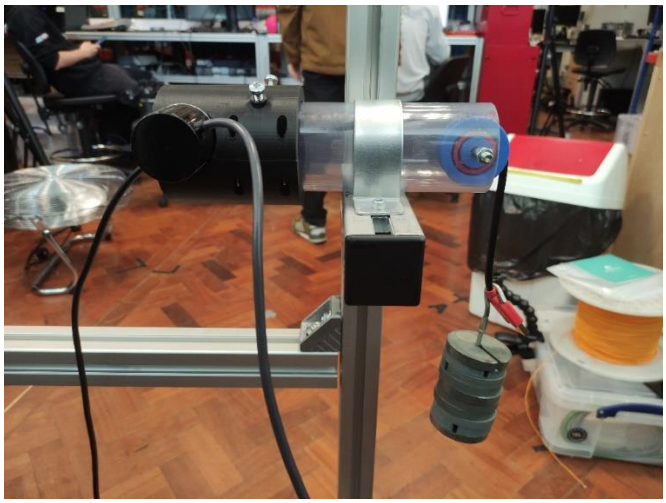

Figure 44: Tether Encoder Clamping Force Experimental Setup

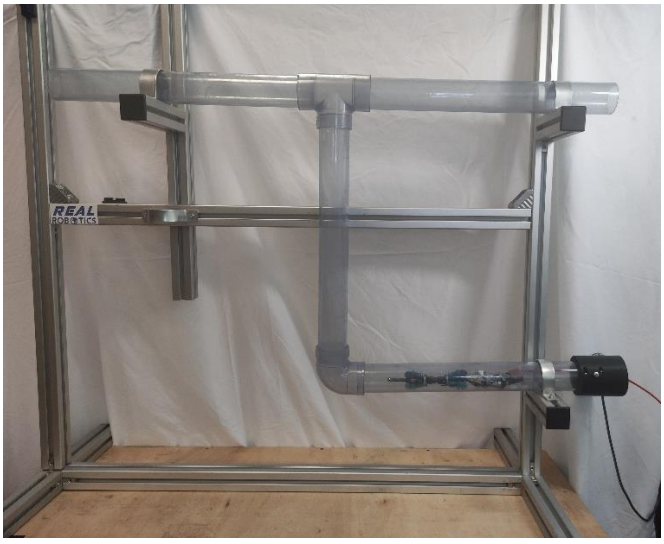

Figure 45: PIABIR Pipe Test Rig