MINIONS: MarINe Internet Of Nodes Specification

Prepared by:

Luke Metcalfe

MTCLUK002

Department of Electrical Engineering
University of Cape Town

Prepared for:

Dr Jane Wyngaard

Department of Electrical Engineering
University of Cape Town

14 October 2019

Submitted to the Department of Electrical Engineering at the University of Cape Town in partial fulfilment of the academic requirements for a Bachelor of Science degree in Mechatronics.

Key Words: IoT, marine-monitoring, LoRa, specification, standards

Declaration

1. I know that plagiarism is wrong. Plagiarism is to use another's work and pretend that it is one's own.
2. I have used the IEEE convention for citation and referencing. Each contribution to, and quotation in, this final year project report from the work(s) of other people, has been attributed and has been cited and referenced.
3. This final year project report is my own work.
4. I have not allowed, and will not allow, anyone to copy my work with the intention of passing it off as their own work or part thereof

Name: **Luke Metcalfe**

Signature: **LM** **Date:** 14 October 2019

Abstract

The Internet of Things (IoT) is seen as a promising new technology with ability to drastically change the ways in which we live, work and interact. As IoT technologies have advanced, and awareness of the current global environmental crisis has grown, the value of such technologies to the field of marine environment monitoring has become more apparent and scientists have begun using them extensively.

However, these technologies and the platforms, designs and standards used in the development of marine-monitoring networks are widely varied, making it difficult for systems to cooperate, and for designers to share designs and build on the works of different groups. These difficulties, along with the challenges of deploying such technologies in the harsh marine environment, have led to a call for standardisation in the sector to facilitate cooperation and accelerate scientific gains.

This project aims to address the lack of interoperability between IoT marine monitoring networks, and the difficulties of collaboration due to the heterogeneity of designs and standards used. The project aimed to define a common electrical, mechanical, operational and communication specification for IoT marine monitoring devices.

Once the specification had been defined, two prototypes were designed and built according to the specification. The prototypes were tested and used the refine and validate the specification.

The system was successfully able to gather, transmit, display and store ocean data. This occurred from a node out at sea, to a node on land, and to a user interface, where it could be viewed on any device within the local network. The system was able to give regular updates, and a degree of remote control was achieved.

The success of the system validated the specification, which aimed bring uniformity and interoperability to IoT marine monitoring networks. The specification introduced standards based on known standards and industry best practices and will aid the sharing of designs, reuse of modules and cooperation between networks. This will ensure devices are suited for deployment at sea and will accelerate development and advances in the field.

Table of Contents

1. Introduction

1.1 Background to the study

The Internet of Things (IoT) refers to a system of devices with sensing and/or actuating capabilities connected to the internet. This is seen as a promising new technology with ability to drastically change the ways in which we live, work and interact. As awareness of the current global environmental crisis has grown, so too has interest in the field of marine environment monitoring. As IoT technologies have advanced in recent years, the value of such technologies to the field of marine environment monitoring has become more and more apparent, and scientists have begun using them extensively.

However, these technologies and the platforms, designs and standards used in the development of marine-monitoring networks are widely varied. This makes it difficult for systems to cooperate, and for designers to share designs and build on the works of different groups. These difficulties, along with the challenges of deploying such technologies in the harsh marine environment, have led to a call for standardisation in the sector to facilitate cooperation and accelerate scientific gains.

1.2 Objectives of this study

1.2.1 Problems to be investigated

The project aims to address the lack of interoperability between IoT marine monitoring networks, and the difficulties of sharing designs and building on the works of others due to the heterogeneity of platforms, designs and standards used. The project aims to define a common electrical, mechanical, operational and communication specification for IoT marine monitoring devices. The specification will be open and based on known standards and best practices in the field. It will also consider the operational and deployment requirements of marine environments, as well as community needs and use cases. Once the specification has been defined, two prototypes will be designed and built according to the specification. The prototypes will be tested and used the refine and validate the specification.

1.2.2 Purpose of the study

The project seeks to identify requirements of IoT marine monitoring devices and to define standards and specifications to bring uniformity to marine monitoring networks (while still allowing innovation) that will ensure their suitability to the environment, aid their cooperation and integration, accelerate their development, and accelerate advances in the field.

1.3 Scope and Limitations

1.3.1 Scope

The scope of work included:

1. A summary review of current practices
2. A written draft specification publicly available for community comment and input under an open source license.
3. Fully documented implementations of at least two prototype modules providing different functionalities implementing the published specification.
4. Laboratory evidence for the viability of the modules operating in the target operational environment over anticipated deployment time periods.
5. Documentation of the designs

For nodes deployed at sea, it was decided that the scope of the specification would be limited to floating nodes and that specifications regarding submerged nodes would form future work.

1.3.2 Limitations

The project had a budget of R1500 for components and materials not supplied by the university, and the timeline was limited to 12 weeks. The university supplied limited mechanical and electrical supplies, free laser cutting and 3D printing services and use of the university's workshops and laboratories.

As the initial proposed network of two nodes at sea interacting with a "receiver" node on land was not feasible due to the budget restrictions, it was decided that a network consisting of one node at sea and one node on land would suffice.

1.4 Plan of development

This report follows the development of a specification for IoT marine monitoring devices, and the detailed development of two nodes according to the specification. The report begins with a detailed literature review to develop an understanding of the current applications of IoT marine monitoring devices, the requirements of such devices and relevant standards.

An interview was conducted with an industry professional, so the insight and advice gained on the development of the specification and the prototypes is discussed. Motivations for some of the requirements outlined in the specification is given, followed by the specification. Based on the specification and literature review, user requirements and technical specifications are defined, which are carried into the design process.

A detailed design process is then followed, concerning electrical, software and mechanical design. The final implementation of the system is discussed, followed by outlines of the testing procedures. The results of the tests are discussed, conclusions are drawn, and recommendations for future work are made.

2. Literature Review

The Internet of Things (IoT) refers to the extension of the internet into the physical realm by means of the widespread deployment of spatially distributed devices with embedded identification, sensing and/or actuation capabilities [1]. Many predict that the density of sensing and actuating IoT devices will steadily increase in the near future, creating a *smart world*, where machines and smart objects are able to communicate, compute and coordinate. This will drastically change the ways in which we live, work and interact and will offer new services, applications and entertainment [2].

In recent years, Wireless Sensor Networks (WSNs), as a subset of IoT, have been deployed in a variety of applications and services, including smart home, smart building, smart transportation, smart industrial automation, smart healthcare, smart grids, and smart cities [3]. These devices will generate massive amounts of data with the potential to improve life across the globe and could play an invaluable role in the monitoring and protection of marine environments [3].

However, deployment of such devices in harsh marine environments has proven to be particularly challenging. Several papers have identified requirements of IoT networks [1] and the need for industry standards [3][4]. [3] reviews the applications of IoT sensor networks in marine environments and both [3] and [5] identify system requirements specific to IoT networks deployed in harsh marine environments. [3] identified the following requirements:

1. **Energy autonomy** – the system should implement software for optimal power management and should have energy harvesting devices in order to operate for long periods without human intervention.

2. **Robustness and fault tolerance** – the system must have the means to ensure there is no loss of data in the event of failures of communication or power supply. In the event of a communication failure, the system should be able to continue collecting data until connection is re-established. Communications subsystems should be robust and reliable and guarantee communication between sensor nodes even in adverse weather conditions.

3. **Stability of Radio Signal** – The curvature of the earth, waves and the placement of devices at sea-level often result in an unstable line-of-sight between transmitters. Bad weather can compound these issues, so systems should take steps to ensure the stability radio signals.

4. **High water resistance** – Typically, such systems would be deployed on the ocean surface or at depth, so sensor and actuator nodes need to be appropriately water and pressure resistant.

5. **Other issues -** Devices should be highly reliable due to the difficulties of deployment and maintenance, sensor network coverage needs to be carefully calculated, and equipment should be designed against possible acts of vandalism.

[4] outlines several issues with current practices in the collection, storage, processing and particularly sharing of ocean data. It calls for the "development, rationalisation and uptake of standards" and investigates "how technologies, scoped by standards, best practice and communities of practice, can be deployed to change the way that ocean data is accessed, utilized, augmented and transformed into information and knowledge". The following requirement can be drawn from their suggestions:

6. **Persistent Identifiers -** sensors, data sets, models, and products should have standard persistent identifiers. These uniquely identify a single entity or a class of entities within a specific context and ensure the correct composition and operation of the system [6].

The FAIR Data Principles [7] suggest that data should be Findable, Accessible, Interoperable and Reusable. The Principles define characteristics that contemporary data resources, tools, vocabularies and infrastructures should exhibit to assist discovery and reuse by third-parties. [4] suggests bringing new technologies and frameworks, such as IoT, into the standards process, and to continue efforts toward building and disseminating ideas around best practice and implementing the FAIR principles for data access and use, leading to a second requirement for data generated by IoT devices.

7. **FAIR Data** – data must be Findable, Accessible, Interoperable and Reusable

[5] developed a low-cost sensor buoy network for possible use in shallow, near-shore marine environments. The system aimed to be small enough for easy deployment, ensure stability of the system in shifting environments, and to provide total autonomy of power supply and data recording. [5] implements the requirements listed above in the following ways:

Requirement	**How requirement is implemented**
1. Energy autonomy	Ultra-Low-Power MSP430F2618 microcontroller, 5000mAh battery supplemented by a solar panel.
2. Robustness and fault tolerance	Flash memory to store data in the event of loss of connection, RTC to generate real-time timestamps for samples.
3. Stability of Radio Signal	Transmitter mounted on tall mast, mechanical structure designed to ensure floatability and stability, weight installed to provide further stability.
4. High water resistance	Watertight enclosure.
5. a) Reliability	Buoy tested in lagoon for reliability.
b) Sensor Network Coverage	Survey carried out in deployment area, maximum range of coverage was determined to calculate minimum number of buoys required to cover area. Star topology employed.
c) Protection against vandalism	Not implemented. Ease of theft due to ease of accessibility noted in discussion.
6. Persistent Identifiers	Each sample identified with a timestamp from RTC. Small mention of identifiers for sensor nodes.
7. FAIR Data	No mention of data being publicly available.

Table 1: Ways in which the system developed in [5] fulfils the requirements identified in [3].

The authors of [5] identified several requirements of the system for it to be effective in its environment. The requirements for *energy autonomy* and *robustness and fault tolerance* identified in [3] were corroborated in [5], but the authors extended the requirements as follows:

8. **Flexibility** – systems should facilitate different configurations of parameters to be sensed as well as timing of sampling and storage. Physically, the system should be flexible and customisable to accommodate a wide variety of devices and communication technologies.

9. **Resource optimisation** – the system should make efficient use of resources and reduce the costs of manufacturing, deployment, operation and maintenance.

10. **Scalability** – the system should be designed to function effectively regardless of the network topology and the number of buoys that will be deployed in the monitored area.

11. **Mechanical design** – components should guarantee an appropriate level of insulation and corrosion-resistance. Connectors are especially sensitive to corrosion and prone to fault and should be kept to a minimum. The physical design should facilitate quick and easy access to components for maintenance, modification and dismantling. The design should also minimise the system's environmental impact, and considerations must be made toward sea traffic.

These additional requirements were implemented in the following ways:

Requirement	How requirement is implemented
8. Flexibility	Adjustable sample period (5-720min), simple design to accommodate several configurations.
9. Resource optimisation	Cost minimized to €180 per device, simple design optimizes resources and allows reuse of system with multiple sensors.
10. Scalability	Variety of network topologies and any number of buoys may be used.
11. Mechanical design	System designed to be robust and durable, remote management implemented to reduce need to access. If access is required, design facilitates ready access to components. Design guarantees insulation and corrosion-proofing.

Table 2: Ways in which the system developed in [5] fulfils the extended requirements identified in [5]

Several good examples of IoT marine monitoring sensor networks, similar in vision to MINIONs, were found [5][8][9], and these will serve as inspiration when developing the specification and designing prototypes of the specification.

[8] proposes "A Flexible Arduino-Based Logging Platform for Long-Term Monitoring in Harsh Environments". They aimed to build a low-cost data-logging platform to provide long-term operation in remote or submerged environments. This paper goes into quite excellent detail of the development of the sensor nodes and prioritises low cost, ease of use and development and flexibility above all else. They also go into great detail about power optimisation and the development of a robust underwater housing. Their power optimisation techniques in particular will be of great value in designing MINIONs. This project lists several benefits of using Arduino microcontrollers. While Arduino microcontrollers may not be used in MINIONs, it provides a case for selecting microcontrollers that are easy to use and allow designers to share code.

Figure 1: Deployment of the sensor nodes developed in [8]

[9] developed an anchored buoy to provide real-time seismometer data. This project is designed for use in deep, powerful seas. Several constraints dictate that MINIONs will most likely be limited to coastal applications, so this project extends quite far beyond the scope of MINIONs. However, the detail in mechanical design and simulation of this system is quite excellent, so will be referred to when developing a flexible platform for MINIONs. The use of a toroid-shaped buoy as a platform for the system will be investigated for use in MINIONs.

Figure 2:The large toroid platform developed in [9]

[3] discusses several different network topologies, referring to the connection topology of the nodes in the network. As discussed in [5], each of these network topologies can be connected as a wired buoy network, wireless buoy network linked by radio, GPRS or both, or an underwater buoy network connected by acoustic waves. Four typical network topologies are illustrated below:

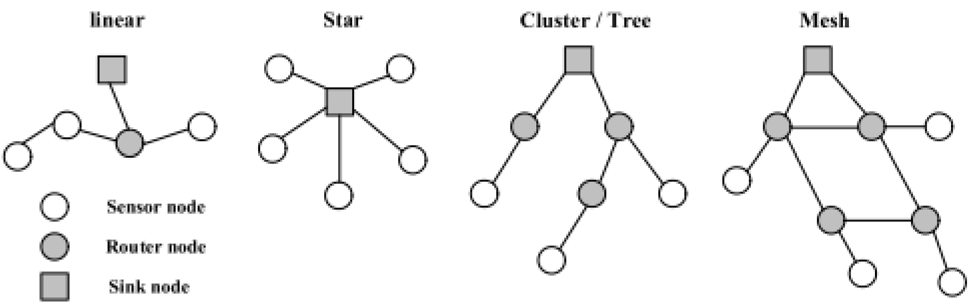

Figure 3: Illustration of the network topologies listed in [3]

The choice of network topology and node density depends on the application and environment in which the network is deployed [10], but other factors to consider include the volume of data and frequency at which it is transmitted, the distance of transmission, energy requirements and availability, maintenance, and the mobility of the sensor nodes [11]. When designing a sensor network, a higher node density may enhance data availability and accuracy, but it may increase energy consumption data collisions, and interferences [3].

[3] illustrates the general architecture of a marine monitoring sensor node, shown in *Figure 5* below. The node typically has a buoy device to keep the system afloat and protect against water, and consists of a sensing module, a microcontroller, a wireless transceiver module, and a power supply module [12].

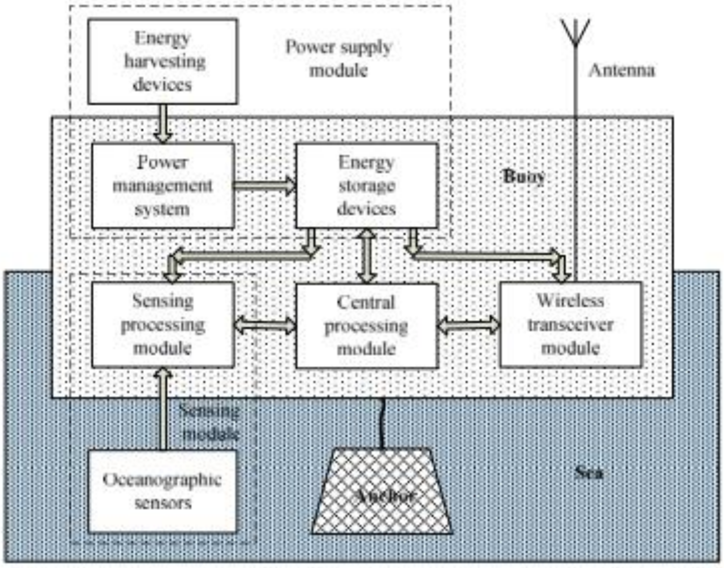

Figure 4: The node architecture illustrated in [3]

The sensing module typically consists of either physical sensors or chemical sensors. Physical sensors measure physical parameters such as temperature, humidity, pressure, wind speed, and wind direction. Chemical sensors measure chemical parameters such as salinity, turbidity, pH, and dissolved oxygen (DO) [3].

The communications module requires some sort of wireless transceiver. The fact that these devices operate in the harsh marine environment requires that the wireless communications network is reliable and energy efficient, as defined in system requirements 1 and 3. Various wireless communication technologies such as WiFi, ZigBee, Bluetooth, GPRS, GSM and WiMax are already well-established and widely used in IoT [3]. Recent developments in IoT have strong requirements of coverage, power consumption and scalability, but none of the technologies employed thus far have managed to simultaneously cope with these requirements [13].

WiFi and Bluetooth are Local Area Network (LAN) technologies which are easy to use and currently dominate the IoT market. These have high data rates but very poor range and very high power consumption, which make them ill-suited for most sensor networks and especially marine monitoring applications. Wireless technologies using the Industrial, Scientific and Medical (ISM) band have been widely used [3][14], but these too have limited range and average power consumption. A novel solution known as Low-Power Wide Area Network (LP-WAN) has emerged as a promising alternative, providing low-cost, low-power and long-range communication technologies [13]. Perhaps the most promising technology to emerge out of this is the Long-Range Wide Area Network (LoRaWAN) technology, or LoRa. LoRa is a spread spectrum modulation technique derived from chirp spread spectrum (CSS) technology. LoRa devices operate across the ISM band at frequencies such as 433 MHz, 868 MHz and 915 MHz, and offer features such as long-range transmission, low power consumption and secure data transmission [15].

[3] lists a large number of IoT marine-monitoring sensor networks. A wide variety of microcontrollers was used in each of the projects, depending on the requirements of the node and the competencies and preferences of the designers. Various requirements of the microcontroller such as memory, clock speed, power usage and ability to optimise power usage, bus interfaces, number of ADC, DAC, PWM-capable and GPIO pins, size and ease of use must be considered when selecting a suitable microcontroller. Many of these used Arduino microcontrollers because of their ease of use, and several projects were able to optimise their power usage [16][8].

[8] made a strong case for using Arduino-based microcontrollers due to their affordability, ease of use, and its well-developed Integrated Development Environment (IDE). However, most Arduinos have relatively large form factors and are not optimised for reducing power consumption. The proliferation of alternative Arduino-compatible boards is providing a new range of hardware options with significantly better power performance in data logging applications [8]. One such alternative is the ESP32 microcontroller, developed by Espressif. The ESP32 is a low-cost, ultra-low-power microcontroller with WiFi and Bluetooth functionality. Important specifications are listed below [17]:

Operating Voltage	3.3V
Maximum Operating Frequency	240 MHz
Analogue Input Pins (ADC)	8
Analogue Output Pins (DAC)	2
Digital I/O pins	28
Flash Memory	4 MB
SRAM	520 kB
Communication	SPI (4), I2C (2), I2S (2), CAN, UART (3) WiFi Bluetooth Radio (2.4 GHz)

Table 3: Specifications of the ESP32 microcontroller

Other notable features include a Real-Time Clock (RTC), Micro-USB port, and configurable power modes which can be optimised to conserve power. The power modes are summarised below [17]:

Power mode	Description			Power consumption
Active (RF working)	Wi-Fi Tx packet			Please refer to Table 15 for details.
	Wi-Fi/BT Tx packet			
	Wi-Fi/BT Rx and listening			
Modem-sleep	The CPU is powered on.	240 MHz *	Dual-core chip(s)	30 mA ~ 68 mA
			Single-core chip(s)	N/A
		160 MHz *	Dual-core chip(s)	27 mA ~ 44 mA
			Single-core chip(s)	27 mA ~ 34 mA
		Normal speed: 80 MHz	Dual-core chip(s)	20 mA ~ 31 mA
			Single-core chip(s)	20 mA ~ 25 mA
Light-sleep	-			0.8 mA
Deep-sleep	The ULP co-processor is powered on.			150 µA
	ULP sensor-monitored pattern			100 µA @1% duty
	RTC timer + RTC memory			10 µA
Hibernation	RTC timer only			5 µA
Power off	CHIP_PU is set to low level, the chip is powered off.			0.1 µA

Table 4: Summary of the configurable power modes of the ESP32 microcontroller [17]

These features make the ESP32 an incredibly powerful microcontroller with very low power consumption at an attractive price point.

Several marine specifications and standards exist. However, these mostly concern large steel buoys [18][19], large steel ships [20] and yachts. While these may not be directly relevant to smaller IoT-based marine-monitoring devices, they contain several sections that may be adapted for this use. For instance, [18] and [19] give insight into the structural requirements of marine buoys. [20] provides standards on thermoplastics for use at sea. This mainly concerns plastic piping on board large steel ships, but the standards may bear relevance to plastic enclosures. This will be investigated further and may form part of the material specification for the MINIONs node enclosure.

Several United States Military standards (e.g. MIL-STD, MIL-DTL) exist on electrical connections and cable connectors, such as MIL-DTL-38999M [21]. These standards will be investigated further to select a connector for the specification, to facilitate wired connections between devices within a network and between multiple networks, perhaps owned by different people.

[22] recommends HDPE and LDPE plastics for use at sea due to their low water absorption and high UV resistance, in terms of both material degradation an aesthetics. [23], [24] and [25] use ABS plastic in attempts to produce 3D-printed waterproof enclosures, while [26] and [27] recommend PETG plastic for this purpose. Waterproofing a 3D printed enclosure poses a major challenge, as most prints are not inherently waterproof due to the micro-gaps between printed layers. However, [24], [23] and [25] use acetone vapour on ABS plastics to smooth and finish the print, filling in any micro-gaps. A similar process can be followed with PETG using ethyl acetate [28].

PETG has good water resistance, durability and UV resistance [26][27][29]. ABS is also durable and has good water resistance, but has poor UV resistance [30], so is not well-suited for this purpose. The material properties of the four plastics mentioned thus far are summarised in Table 5 below [31]:

Plastic	Water absorption rate (%)	UV resistance	Durability	Tensile Strength (psi)	Modulus of Elasticity
HDPE	0.1	High	High	4000	200,000
LDPE	0.1	High	High	1400	30,000
ABS	0.3	Poor	High	4100	304,000
PETG	0.2	High	High	7700	310,000

Table 5: Material properties of HDPE, LDPE, ABS and PETG plastics

HDPE and LDPE stand out as firm favourites for injection moulded products for use at sea, but ABS and PETG have been explored as alternatives to produce 3D printed waterproof enclosures. ABS seems to have been explored mainly for its ability to be smoothed but is found to be severely lacking in its resistance to UV.

PETG's tensile strength (7700 psi) is relatively large, and its modulus of elasticity (310,000) is quite high meaning that it is quite stiff, which might not be an ideal property for use in the constantly-moving marine environment. However, the enclosure is unlikely to bear a great load, so these properties may not be critical. Some of PETG's physical properties make it ideal for this application and this material will certainly be explored in an attempt to 3D print a waterproof enclosure. This would serve the MINIONS specification well, as it would provide an enclosure that is easy to replicate and uniform across all MINIONS projects.

[32] provides information on a variety of metals and recommends certain alloys for use at sea. Type 316 stainless steel is widely used in marine environments, as it is more resistant to atmospheric and general corrosive conditions than any of the other standard stainless steel alloys and has a very high tensile strength [32]. The 5052 aluminium alloy has good formability, workability and weldability, fair machinability, excellent corrosion resistance and medium to high strength [33]. It is used primarily in marine applications and in can be used in precise CNC machining [33][34]. Due to its workability, 5052 aluminium will be used for structural elements of the buoy, while type 316 stainless steel will be used for fasteners due to its high tensile strength.

[35] conducted a survey on 60 launched, and 44 to-be-launched CubeSat satellites to determine the reliability of their bus interfaces. They found that the I2C data bus shows many bus lockup issues. I2C was proven to have caused a catastrophic failure for one mission, and it was hypothesised that it was the cause of catastrophic failures in two other CubeSat missions. Electrical Power Subsystems (EPS's) were found to be a major source of in-orbit failures, but most of the failures were not related to the power bus. The paper made several recommendations on the physical interface to the power unit, but with regard to data busses, suggested only that "the data bus should have a continuous nominal behaviour, without major risk for bus lockups." However, another paper [36] involving one of the authors of [35] went on to design a data bus architecture for CubeSats, recommending the USB, CAN and RS485 bus protocols. However, it identified RS485 as the preferred choice due to its relatively low power consumption, high achievable data rate, simplicity and reliability. This will be used to define the specification for wired connections between nodes.

Ingress Protection (IP) Ratings, as defined in the international standard IEC60529, classify the degrees of protection provided against the intrusion of solid objects, dust, accidental contact, and water in electrical enclosures. IP68 and IP69K, the highest IP ratings, both specify that the enclosure is

protected from total dust ingress. Additionally, IP68 specifies that the enclosure is protected from long-term immersion “up to a specified pressure” (specified by the manufacturer), while IP69K specifies that the enclosure is protected from steam-jet cleaning [37]. The IP ratings are quite limited for specifying the waterproof rating of devices for use in deep-sea environments. Further research will be done to include a clearer, well-established standard concerning waterproofing into the MINIONs specification. If this cannot be found, IP68 will be adopted and a depth requirement will be specified.
Several projects have aimed to develop low-cost, waterproof sensor enclosures that are easy to replicate [8][23][24]. [8] based their system on cheap and readily available PVC piping and fittings. PVC is waterproof and chemically resistant but has poor UV resistance. To get around this, they applied a protective coating of water-based exterior latex paint. This solution is not ideal, as it is effectively applying a semi-permanent seal that would have to be broken and reapplied for maintenance. This contradicts the requirement set out in [5] that the system should “facilitate access to components for easy maintenance, modification and dismantling”.

Figure 5: The PVC enclosure developed in [8]

[23] and [24] endeavoured to 3D print a waterproof enclosure. Both provided good insight, advice and lessons learnt in their endeavours. They both used ABS plastic and mentioned the use of acetone to smooth the outer surface and fill in any small gaps. [23] did not manage to achieve the objective but made good recommendations for future work. [24] made beautiful, naturally-shaped, waterproof enclosures in order to gather data on the forces experienced by mobile river sediment grains. Their system consists of an internal, pill-shaped enclosure to house the sensors. This inner case was designed to hold the electronics firmly in place. This was then enclosed in a larger case that mimicked the form of natural sediment. Each external case was designed to enclose the same internal case, meaning that the external case of a sensor could be changed at will. They used clever design to interlock two halves of each case, but information on the seals they created is unfortunately quite vague, although they do mention the use of O-rings. All three of these projects will be considered when designing an enclosure for the MINIONS Specification. [38] provides an excellent guide to making waterproof enclosures using static O-ring and gasket seals. This guide also mentions the use of 3D printing to make enclosures and recommends using high-resolution printing materials such as ABS in order to seal against O-rings and gaskets.

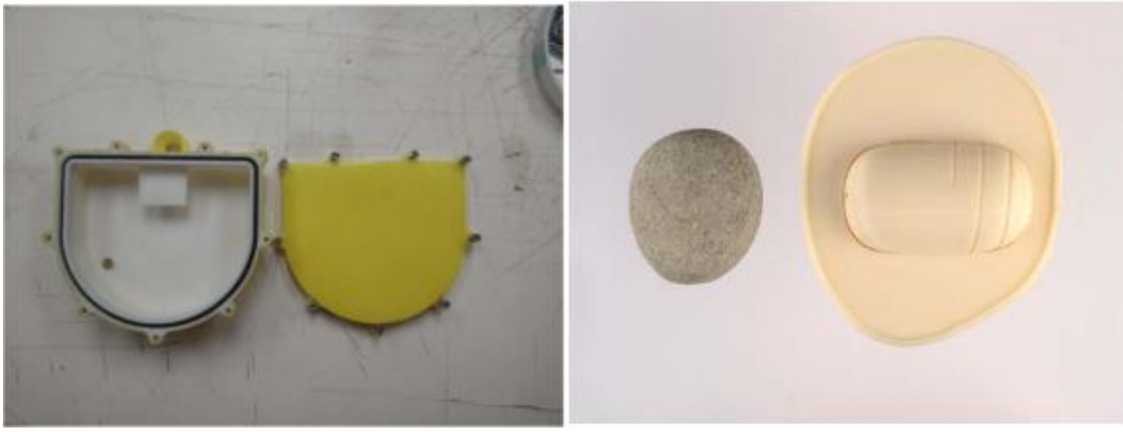

Figure 6: 3D printed enclosures developed in [23] (left) and [24] (right)

3. Methodology

3.1 Overview

The approach taken for this project was to gain an understanding of what was required and what already existed in the field before embarking on a design process. As the project advisor, Dr Jane Wyngaard was consulted to define the project scope. Once the scope and project requirements were understood, research and a literature review were conducted. An interview was conducted with Mr Rick Harding, a retired marine biologist with a wealth of experience designing and building marine-monitoring devices. The interview provided a wealth of insight and recommendations for both the MINIONS Specification and the two prototypes that were to be built.

The MINIONS Specification was then drafted, setting in place the requirements of the prototypes and the standards and specifications they would have to follow. The Specification was not set in stone. While the Specification had a large influence on the design of the prototypes, the prototypes helped refine the Specification and reveal issues that had not yet been considered. The literature review, along with the MINIONS Specification were used to develop User Requirements and Technical Specifications, which were taken into the design phase. The design phase of the project included the designing, building and testing of two prototypes according to the Specification.

3.2 Interview with Mr. Rick Harding

Mr. Rick Harding is a retired Marine Biologist with vast experience in designing moored buoys and drifters and gathering data on board research vessels such as the S.A. Agulhas and Africana. He made several suggestions for the mechanical and electrical design of MINIONS, and also provided insight into the wealth of knowledge he has gained throughout his career. He recalled how one of his projects mysteriously made its way several kilometres inland of Mozambique, and how they had to negotiate with the South African Naval Academy in Gordon’s Bay to test one of their devices within their harbour to prevent it being stolen. He told these stories light-heartedly but emphasised the need to make serious considerations regarding security, as he has experienced several incidents of theft and vandalism. These considerations are less of a concern when operating far out at sea but become more important in coastal applications. This corroborates the need identified in [3] to design against possible acts of vandalism.

He offered great insight into mooring systems and suggested a simple system for MINIONS. One of the key takeaways was to keep the mechanical design as simple and structurally solid as possible. He suggested to avoid underwater cabling and complicated rigging systems, as he has seen many of these tangled and destroyed by the intense storms and swells in the Western Cape.

3.3 Drafting the MINIONS Specification

The MINIONS Specification drew inspiration from the CubeSat specifications developed by California Polytechnic State University (Cal Poly) and Stanford University in 1999 [39]. The CubeSat specification covers general, mechanical, electrical and operational requirements, which the MINIONS Specification tried to emulate, but adaptations and additional provisions were made to tailor the Specification to the marine environment.

The CubeSat specification had a requirement stipulating that all parts shall remain attached to the CubeSats during launch and that no additional debris shall be created. This was applicable to the marine environment, given the current pollution crisis and the hazard it could pose to sea traffic and sea users, so was included in the MINIONS Specification. It also stipulated that total stored chemical energy may not exceed 100 Watt-hours. Attempts were made to find a justification for this requirement, and it was found that the United States Federal Aviation Authority imposes restrictions on the maximum capacity of Lithium batteries on flights [40]. As such, any designer hoping to travel with their MINIONS-compliant device could face difficulties, so this requirement was included in the specification.

The literature review determined that IP68 and NEMA 6P were the most suitable ratings of their respective rating systems for use in the MINIONS Specification. However, deep sea conditions far exceed those covered in both rating systems, so an accommodation was made for customised waterproof requirements in the MINIONS Specification. A variety of materials was discussed in the literature review, and those suitable for use at sea were included in the Specification. However, provision was made for other materials, as long as they proved to have acceptable material properties.

The literature review discussed reasons for standardising RS485 as the external bus protocol, as other protocols had led to failures in several CubeSat missions [35]. This paper also discussed how power unit failures accounted for a large majority of CubeSat failures. Thus, the CubeSat specification places an emphasis on ensuring the safety and reliability of power units and requires that battery protection circuitry be incorporated to avoid unbalanced cell conditions. This requirement was relevant to the MINIONS Specification, so was included.

Several restrictions are placed on use of the ISM band in a particular country, and each country has its own regulations and frequency bands. Non-compliance can wreak havoc with other systems and could bear legal consequences for the offender. As such, a requirement for compliance with the user's regional licence agreements was introduced.

The requirements on mechanical form and dimensions were left out of the Specification until the mechanical system was developed during the design phase. The mechanical design was influenced by several of the already-drafted specifications, such as those on approved materials. Once the system was designed, built and tested, attention was turned to refining the Specification. It was hoped that the knowledge and experience gained by building the two prototypes would reveal the true practicality of the Specification and bring to light new requirements which had not yet been considered.

The final MINIONS Specification can be found in the Git repository link in Appendix C.

3.4 User Requirements and Technical Specification

3.4.1 *User Requirements*

Based on the requirements of IoT devices in marine monitoring environments, identified in the literature review, as well as the requirements detailed in the MINIONS Specification, the following User Requirements were created:

i. General System Requirements

Requirement Number	User Requirement
GR1	Build and develop a marine monitoring device according to the MINIONS specification.
GR2	The system must be flexible to allow for multiple configurations of sensors and communication technologies to be used.
GR3	The system should be scalable to allow for different network topologies and to cover different areas.
GR4	The system's environmental impact should be kept to a minimum
GR5	The system must not obstruct sea traffic.
GR6	The network coverage and an appropriate network topology must be carefully determined.

Table 6: Table of General User Requirements developed for the MINIONS prototype

ii. Mechanical Requirements

Requirement Number	User Requirement
MR1	The system must be durable and last beyond the length of the planned deployment.
MR2	The system must be waterproof.
MR3	The system must be stable to aid wireless communications.
MR4	The design should facilitate easy access for easy maintenance, modification and dismantling.
MR5	The system must be designed to protect against acts of vandalism.
MR6	The system must be reliable to minimise maintenance.
MR7	All components, electrical and mechanical, must be corrosion-resistant or protected against the effects of corrosion.
MR8	Connectors must be kept to a minimum.

Table 7: Table of Mechanical User Requirements developed for the MINIONS prototype

iii. Software Requirements

Requirement Number	User Requirement
SR1	The system must give accurate updates of the device's state.
SR2	A degree of remote-control must be achieved.

SR3	It must be possible to change certain parameters such as sample period remotely.
SR4	The system should implement software for optimal power management.
SR5	The system must log and store all data for analysis.
SR6	The system must upload data to a user interface for public viewing.
SR7	Persistent identifiers must be used for devices as well as data samples.
SR8	The system must be fault tolerant and allow sampling to continue in the event of a communication failure.

Table 8: Table of Software User Requirements developed for the MINIONS prototype

iv. Electrical Requirements

Requirement Number	**User Requirement**
ER1	The system must sense one physical parameter and one chemical parameter.
ER2	The system should log samples and transmit the data and other important information to a receiver on land.
ER3	All electronics should be well-insulated.
ER4	The system should have some level of energy autonomy.
ER5	Electronics should be optimised for low energy consumption and optimal power use.
ER6	A stable power supply must be provided
ER7	Protections must be in place to prevent over-charge and over-discharge of the battery.

Table 9: Table of Electrical User Requirements developed for the MINIONS prototype

3.4.2 Technical Specifications

The User Requirements were used to draft Technical Specifications for the system, which drove the design phase of the project.

Technical Specification Number	**Relevant User Requirement(s)**	**Technical Specification**
TS1	GR6	A linear network topology will be employed.
TS2	GR5	The buoy will be placed in an area where sea traffic is not frequent.
TS3	GR2	The buoy platform will allow a maximum of 8 modules to be attached.
TS4	GR2	The enclosures must be customisable.
TS5	MR4	The enclosure lid will have a screw-top to allow easy access.
TS6	MR5	The buoy will be painted to camouflage and avoid attention

		from vandals.
TS7	ER1	The node will sense temperature (physical) and turbidity (chemical).
TS8	GR6, ER2, ER5	Long-range wireless communications will use LoRa technologies at a frequency of 868 MHz.
TS9	SR6	Short-range wireless communications will use WiFi.
TS10	GR6, ER2	The system must be able to transmit data at least 1km.
TS11	ER5	The power module will consist of a 4.2V Lithium-ion battery, a battery protection and charging board, and a voltage regulator.
TS12	ER5	LoRa radio modules and an ESP32 microcontroller will be used for their low-power capabilities.
TS13	MR3, GR2, GR3	The device will consist of a stable floatation device with a platform on which a variety of modules may be placed in a variety of configurations.
TS14	MR2	The enclosures for the modules must be IP68 and NEMA 6P rated.
TS15	GR2, MR4	The enclosures will be cylindrical, measuring 110mm in both diameter and height. The pitch of the holes to bolt the enclosure to the platform must form a 127.5x127.5mm square. The exact design will follow that of the MINIONS Specification.
TS16	GR1, MR1	All exposed parts will be made of materials approved by the MINIONS specification.
TS17	SR1	During each period, the node will transmit its state (Wake or Sleep), sensor data, sample period and an estimation of remaining battery percentage.
TS18	SR1, SR6	A user interface will display all received data.
TS19	SR2, SR3	The user interface will allow the user to change the sample period and restart the node if necessary.
TS20	SR4, ER5	The node will enter “Deep Sleep Mode” after each transmission, for a length of time determined by a “sample period” parameter.
TS21	SR5	All received data will be stored in CSV format.
TS22	SR7	The sensor network is identified with a network ID, all devices in the network are identified with an address, and all data samples are identified with a Unix timestamp. If the network has more than one node, it is possible for samples to contain an identifier for the device from which they were recorded.
		Battery protection circuitry will be used to prevent over-charge and over-discharge. Additionally, a voltage monitor circuit will be implemented so the user is able to receive live updates on the battery’s state. Software will shut the node down and inform the user if the battery voltage goes below a certain threshold.

Table 10: Table of Technical Specifications developed for the MINIONS prototype

3.5 Design

3.5.1 Design Approach

The approach taken was to get an understanding of what was required of the system as a whole, and then to identify nodes which would make up the system. It was decided that the system would require one node at sea to record and transmit data, another node on land to receive data, and some way of interacting with the system via the internet (server). For naming purposes, the node at sea was referred to as the Transmitter Node, while the node on land was referred to as the Receiver Node. Once these were identified, the Transmitter and Receiver nodes were broken up into three subsystems: electrical, software and mechanical.

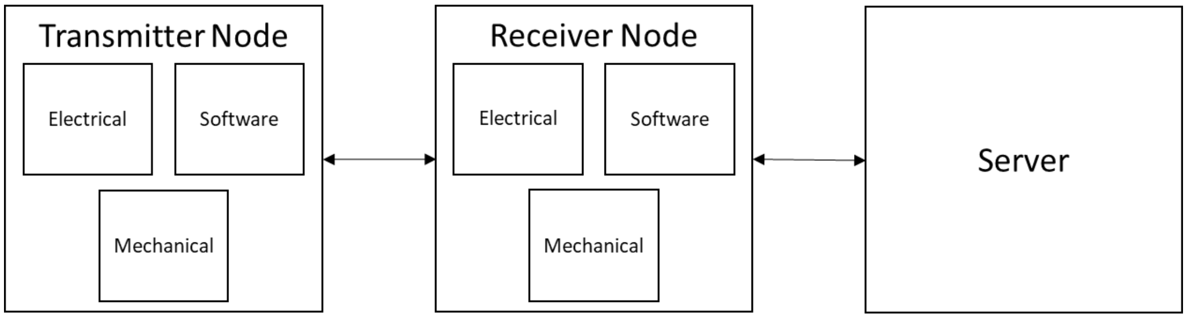

Figure 7: Diagram illustrating the division of the MINIONS system into Transmitter Node, Receiver Node and Server

The electrical subsystems of both nodes were focused on first, while the mechanical subsystem was focused on last. As the nodes would be physically separated yet communicating wirelessly, the interactions and interfaces between them became the focus point, so the electrical design of each was addressed first. Research was conducted to find components for each node that were fit-for-purpose, and components were chosen to ensure the nodes were electronically compatible. Consideration was made to software to ensure that each node's software would be able to interact with that of the other, although no software design had yet taken place. Once the electrical design was complete, the focus was turned towards designing software that would use the electrical subsystems to achieve the required system functionality and would interact with the server to create a user interface. The electrical and software sections of each node were integrated, and the interaction between the two nodes, the server and the user interface was tested. The focus was then turned to designing a mechanical structure for the Transmitter Node that would be able to accommodate the electronics to and allow the node to operate at sea in order to satisfy the User Requirements. The mechanical system was tested independently before being integrated with the electrical and software subsystems. Once the complete system had been tested in a safe environment, it was deployed at sea to conduct its final testing.

3.5.2 *Electrical Design*

i. ***Transmitter node***

Sensor Module

The system architecture of transmitter node was chosen to be that shown in Figure 4, and consists of a sensing module, a microcontroller, a wireless transceiver module, and a power supply module. After conducting the literature review and listening to the suggestions of Rick Harding, it was determined in the technical specifications that the sensing module would sense temperature and turbidity.

[3] identified that sensors fall under two categories – chemical or physical. It was therefore decided to use one sensor from each category. The paper also identified commonly-sensed parameters, and after consulting with Rick Harding, and conducting research to identify which sensors were available, it was decided that the system would sense temperature (a physical parameter) and turbidity (a chemical parameter).

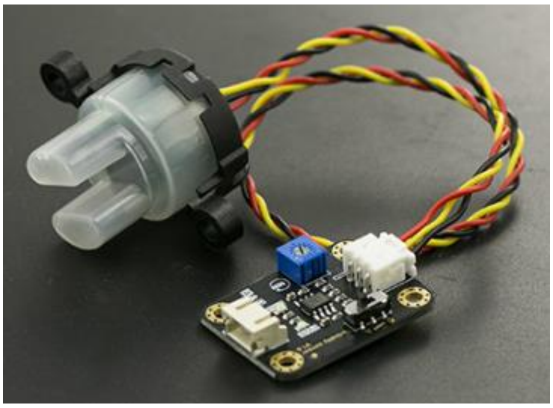

Figure 8: "Gravity: Analog Turbidity Sensor For Arduino", available at DF Robot [41].

DF Robot's "Gravity: Analog Turbidity Sensor for Arduino" was chosen due to its ease of use and versatility. A simple switch allows the user to choose between analogue or digital output, and the sensor only requires connections to 3.3V, GND, and either an analogue or digital pin on the microcontroller. This avoids having to use more complicated bus protocols. Additionally, the sensor's clear plastic housing, while not waterproof, appeared as though it would be possible to modify to create a waterproof housing which would allow the sensor to protrude from the main enclosure.

Figure 9: "Waterproof DS18B20 Digital Temperature Sensor for Arduino", available at DF Robot [42].

DF Robot’s “Waterproof DS18B20 Digital Temperature Sensor for Arduino” was chosen for its accuracy and ease of use. The sensor only requires connections to 3.3V, GND, and a digital pin on the microcontroller. Once again, this avoids having to use more complicated bus protocols. The sensor is apparently waterproof, although no IP or NEMA rating is given, so an enclosure which allowed the sensor to protrude from the main enclosure would have to be created for this sensor too.

Wireless communication module

LoRa was chosen as the wireless communication technology for long-range transmissions due to its low power consumption and long range. The REYAX RYLR896 868 MHz LoRa module, consisting of a Semtech SX1276 LoRa chip, was chosen because it came with a built-in coil antenna. Several other LoRa modules exist on the market but require an antenna to be installed. These can cost as much as the modules themselves, so the RYLR896 module was chosen to significantly reduce costs. A simple wire antenna could have been used with other LoRa modules, but the reliability and range of such a setup was doubted.

The module is rated for a typical range of 4.5km, and maximum range of 15km. Other benefits include high interference immunity and high sensitivity, which means the module is able to demodulate signals at low power levels. The module is controlled by AT commands, which simplifies configuration, transmission and reception significantly, and features AES-128 [43] data encryption. The device has configurable “Network ID” and “Address” parameters, which allows the user to select which network, and which device in the network to communicate with.

Figure 10: REYAX RYLR896 868 MHz LoRa Transceiver [44]

Microcontroller

Having conducted the literature review, the ESP32 microcontroller, produced by Espressif Systems, was chosen as the microcontroller for both the transmitter and receiver nodes. The ESP32’s specifications are shown in Table 3, which illustrates its power and versatility. The board has a large number of both analogue and digital input and output pins and supports a variety of bus protocols. This was important, as the microcontroller would need to interface with a number of sensors and peripherals, and perhaps other microcontrollers via a wired connection. To connect with the server, the microcontroller needed to support WiFi. ESP32 supports both WiFi and Bluetooth, which created the opportunity for several more applications.

The ESP32’s configurable power modes were a significant factor in its choice, as the requirements listed in the literature review emphasised the need for power optimisation, both electrically and in software. The board has a Real-Time Clock (RTC), Micro-USB port, and can be configured to run in the Arduino Integrated Development Environment, which has well-written libraries to support the ESP32 interfaces.

Figure 11: ESP32 Dev Kit-C [45]

Power Module

Lithium-ion cells were chosen due to their relative power density. These are commonly used in portable electronics and electric cars, so they are well-documented in literature, and many components exist to support these cells. They are small, have a nominal voltage of 3.7V, nominal capacity of 2600mAh, and an operational range of −20 to 60°C [46].

Figure 12: Samsung 3.7V, 2600mAh 18650 Lithium-ion Battery [47]

The MINIONS Specification requires battery protection measures, so the TP4056 Lithium-ion battery charging board was chosen to protect against over-charge, over-discharge and overcurrent conditions.

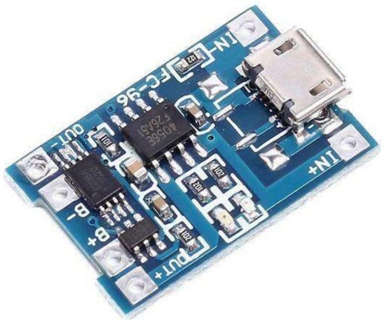

Figure 13: TP4056 Lithium-ion Battery Charging Board [48]

This connects to the battery and allows it to be charged either by another voltage source or a USB input. The board provides a protected output which can be connected to the rest of the circuit.

At full charge, the battery has a voltage of 4.2V. This voltage is too high for the other components in the circuit but also drops as the battery discharges and so a voltage regulator was required. The MCP1700 is a family of CMOS low dropout (LDO) voltage regulators that can deliver up to 250 mA of current while consuming only 1.6μA of quiescent current [49]. The low dropout voltage and low quiescent current are crucial in maximising the power supply module's efficiency. The fixed 3.3V variant was chosen, as all of the components selected thus far require a 3.3V power supply. As per the datasheet, $1\mu F$ capacitors were connected from V_{in} and V_{out} to ground.

To monitor the voltage of the battery, a simple voltage monitor circuit had to be implemented. The ESP32 is able to read analogue voltages up to a maximum of 3.3V. As the battery can reach 4.2V at full charge, a voltage divider was connected across the battery so that the output was limited to a maximum of 3.3V.

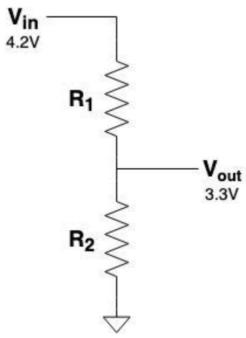

Figure 14: Voltage divider used in the Battery Voltage Monitor circuit

The equation for the voltage divider is as follows:

$$V_{out} = \frac{R_2}{R_1 + R_2} \cdot V_{in}$$

To reduce the current draw by the voltage monitor circuit, high resistances were chosen. By choosing $R_2 = 100\ k\Omega$, R_1 was calculated to be $27\ k\Omega$, and the current was limited to $33\mu A$. The microcontroller was then able to read the voltage and implement a mapping in software to convert the value to a percentage to indicate the remaining capacity.

Each of the electrical modules were prototyped tested on a breadboard. Once all of the modules were tested and verified, they were connected to one another. The schematic of the assembled circuit is shown below:

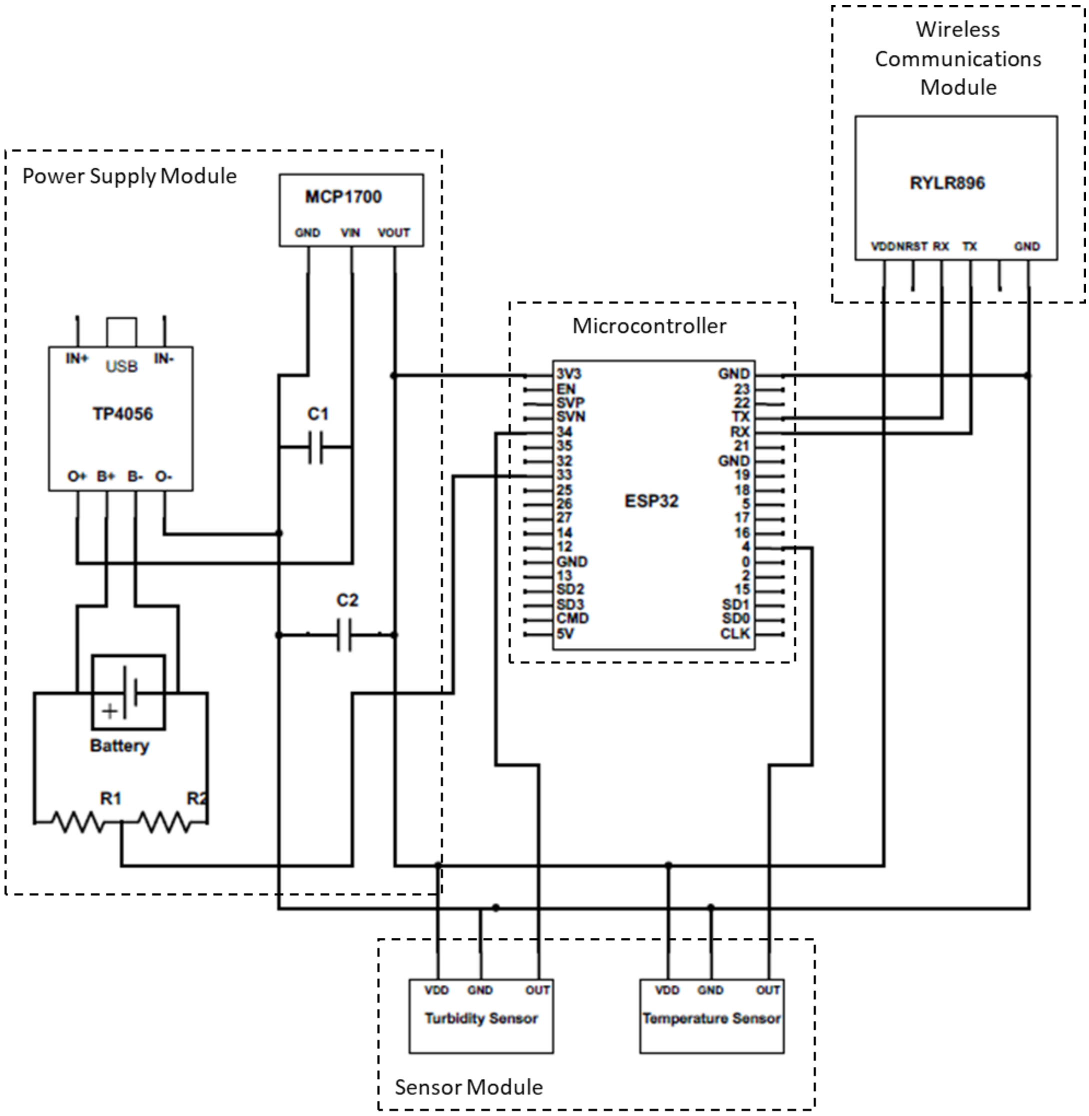

Figure 15: Transmitter Node circuit diagram

This assembled circuit for the Transmitter Node was tested and verified, so all that was needed was to design Veroboard circuits that would fit into the mechanical design. Placing the enclosures as far away from the water as possible, and especially avoiding them being submerged, reduced the risk of a system failure due to a leak. However, the sensing module needed to be submerged. As such, it was decided to split the circuit into two sections and contain each section in a separate enclosure:

- The **upper enclosure**, containing the microcontroller, wireless communications module and power supply module, would be placed on top of the platform.

- The **lower enclosure**, containing only the sensing module, would be placed on the bottom of the platform, submerged.

Once this decision was made and the complete circuit was tested and verified, Veroboard circuits were designed and built to be used in each of the enclosures.

Upper enclosure

Due to space limitations of the enclosure, it was decided to split the Veroboard design into two levels. To facilitate easy access, as per user requirement MR4, the battery needed to be placed on the upper level, and so it was decided that the entire power supply module would be placed on the upper level. As the Veroboard has metal tracks running along its length, placing the wireless communications module on the lower level would likely block signals, so it was decided to place it on the upper level as well. This meant that all that was left to be placed on the lower level was the microcontroller. The Veroboards were cut into circular shapes to fit the enclosure, and the circuits were tested using common lab equipment.

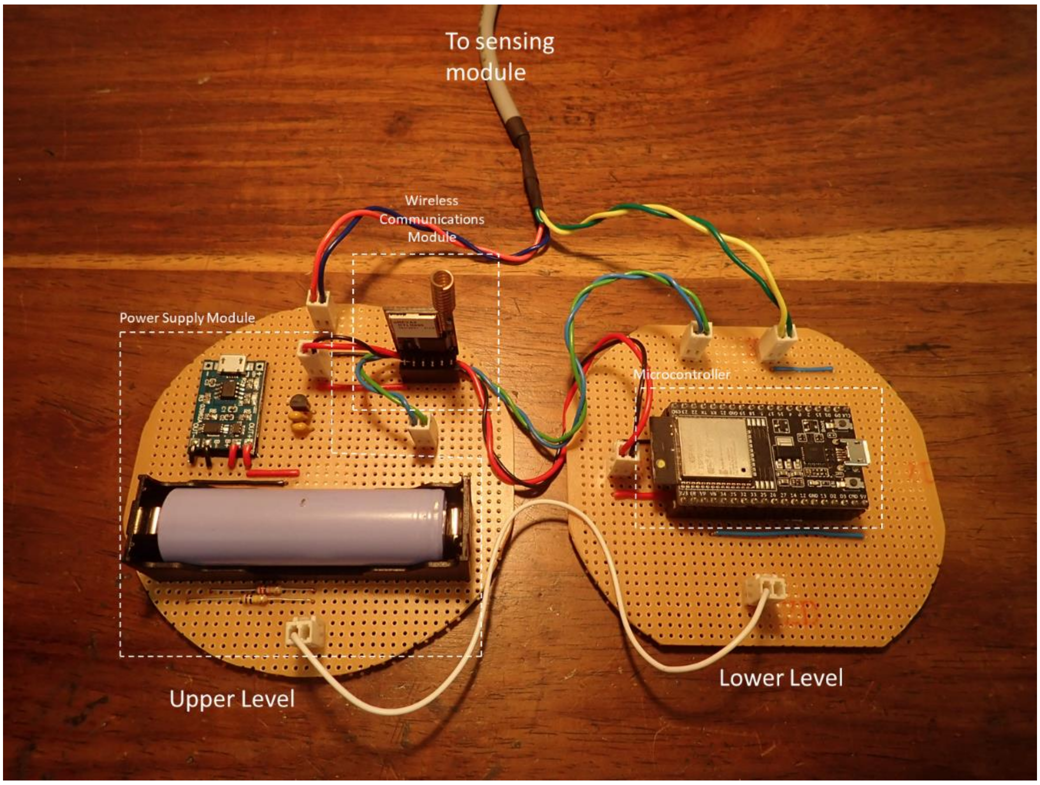

Figure 16: Circuity installed in the upper enclosure

Lower enclosure

Both sensors are able to share a common VDD and common GND, which reduced the number of cables required for the sensing module from six to four. A cable with four internal cables was used to reduce the number of cables needed to connect the two enclosures, thus reducing the number of connectors as per user requirement MR8. A small Veroboard with 3 Molex connectors allowed the cable to be broken out. The two sensors plugged into to this board, thus receiving the VDD and GND they required, and allowed their respective outputs to be connected to the microcontroller.

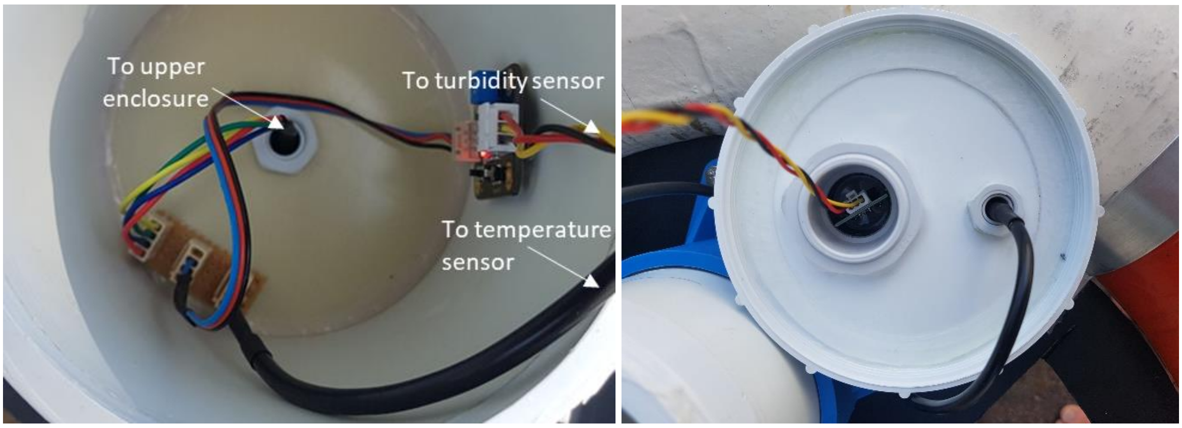

Figure 17: Circuity installed in the lower enclosure

ii. ***Receiver Node***

Electrically, the Receiver Node needed only a microcontroller, wireless communications module and a user interface. For the same reasons listed in the electrical design, the ESP32 was chosen as the microcontroller for the Receiver Node too. The most critical features of the ESP32 for use at the Receiver Node were its high processing power, and its WiFi connectivity. As will be discussed in the Software Design section, this was crucial as the Receiver would have to run constantly, process a lot of information, and upload to the internet.

The REYAX RYLR896 LoRa modules had been chosen for the wireless communication module of the Transmitter Node for its performance, ease of use, affordability and encryption. For the same reasons, and to simplify the process of ordering components, the same module was chosen for the Receiver Node.

The Receiver Node is a component of the base station, so the requirement to optimise power consumption was relaxed, as the node could be powered from mains. In fact, using the ESP32's power optimisation software would have led to mistimings in the transmission and reception of data, thus resulting in lost packets. This resulted in a significant power consumption by the receiver node, and so the ESP32 was powered via a USB cable connected directly to mains.

Besides the web interface that was to be developed, it was intended that the Receiver Node also have some sort of user interface to give the operator a better idea of the system's functioning during deployment. For this reason, LED's indicated the state of the Transmitter Node and the successful reception of a message, and an OLED screen displayed the most recent message along with other important information from the Transmitter Node.

Figure 18: ACM I2C 1.3in OLED screen

A schematic of the complete system is shown below:

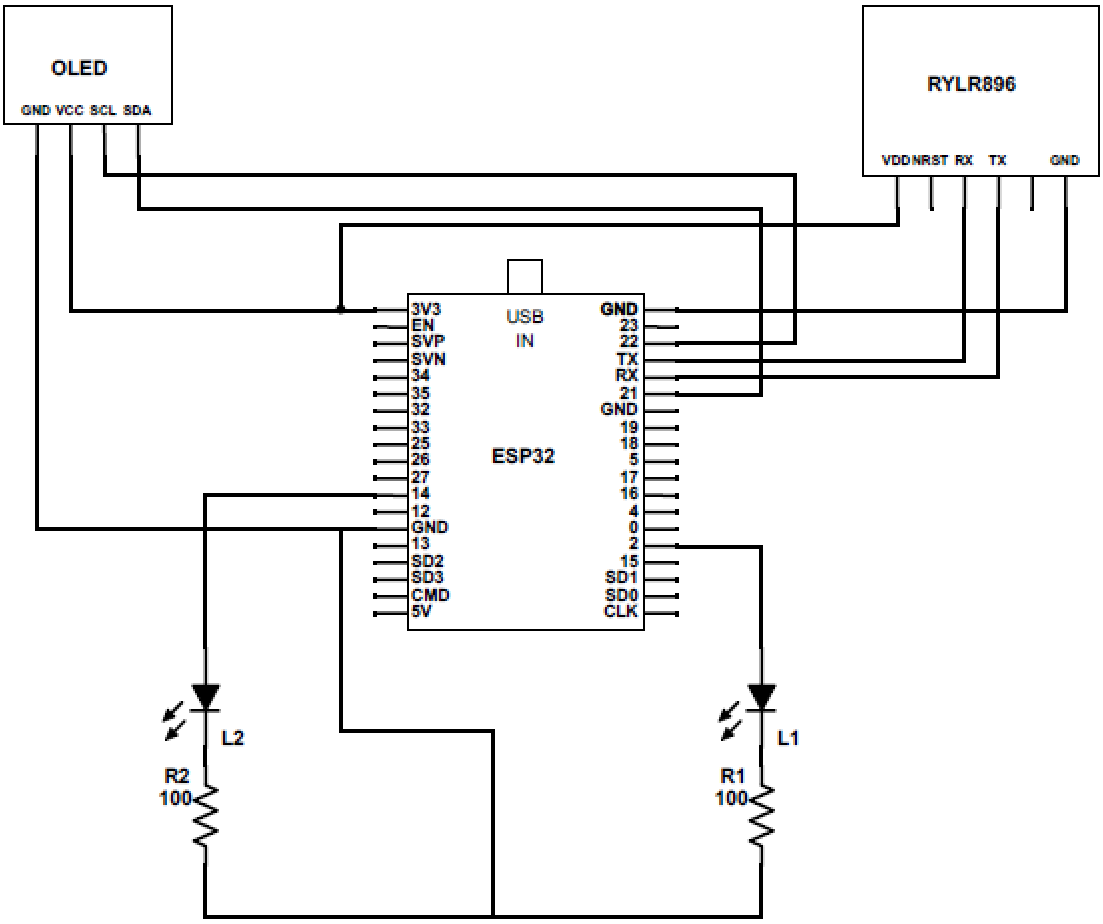

Figure 19: Receiver Node circuit diagram

Due to time constraints, the circuit was implemented only on a breadboard. Future work will involve building a Veroboard circuit for the Receiver Node and 3D printing a simple enclosure. The circuit is shown below:

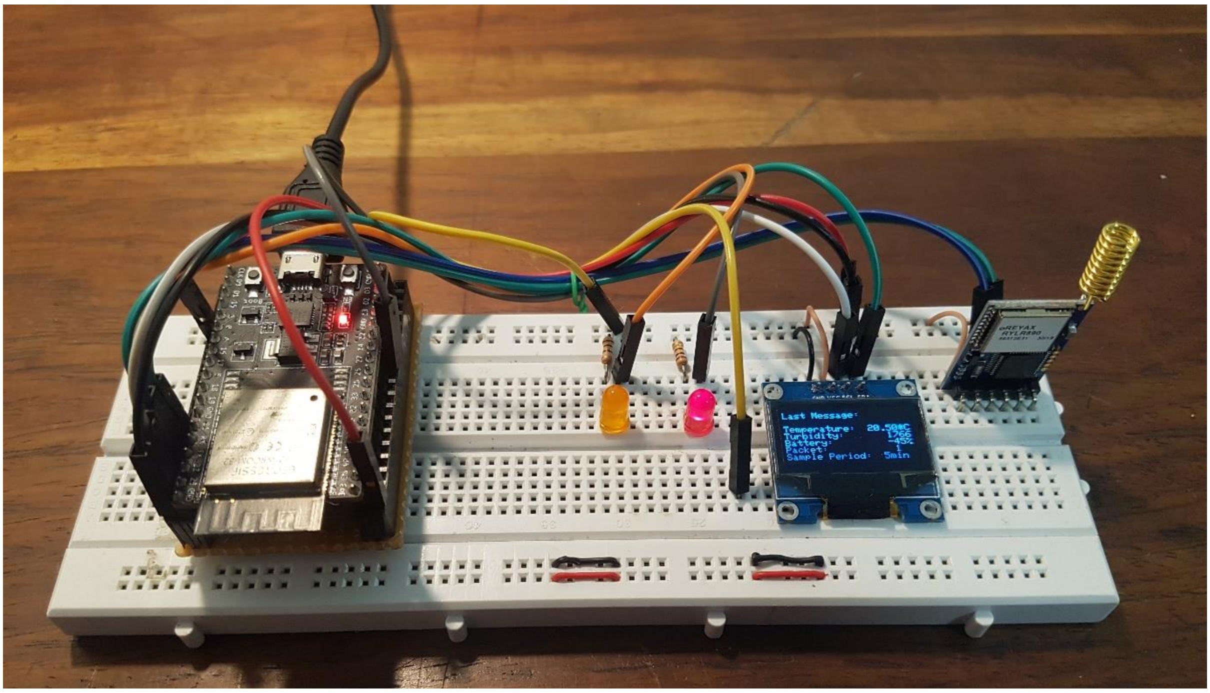

Figure 20: Receiver Node

3.5.3 *Software Design*

i. *Design*

Figure 21 below illustrates the interaction between subsystems:

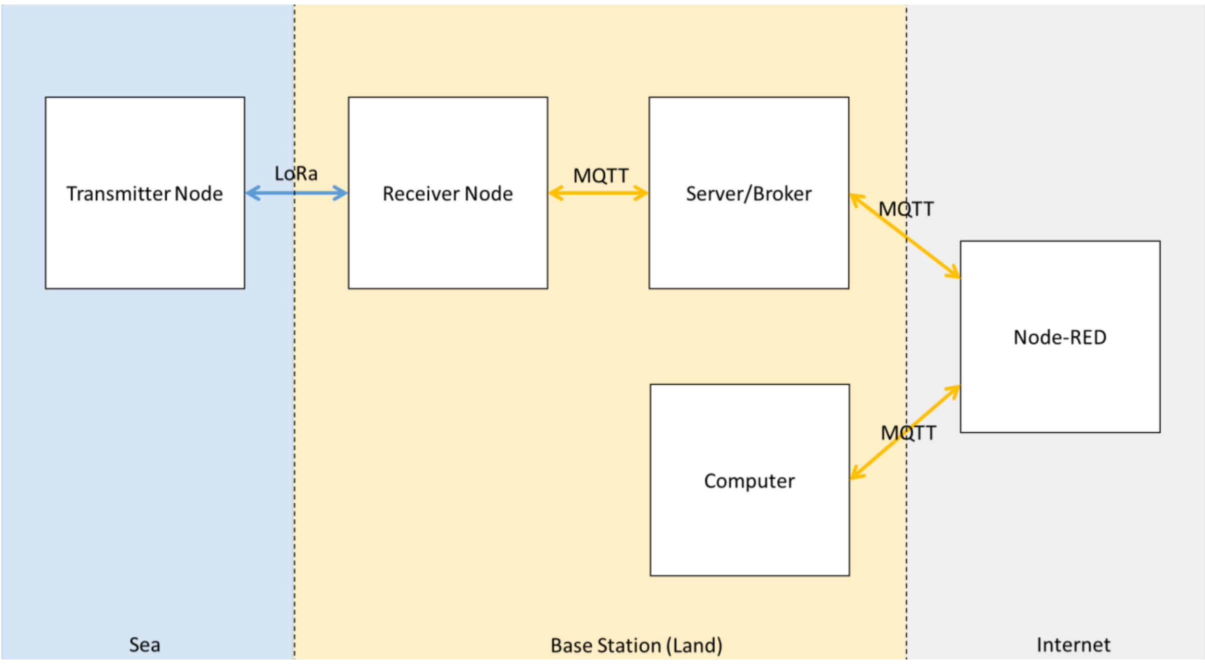

Figure 21: Diagram illustrating the location and interactions between MINIONS nodes, the server, Node-RED and other devices

Development of the software for both of the nodes as well as the system and user interface, was a slow, iterative process. Once the circuit design was complete, code was developed for each subsystem of the node. Once the code for a subsystem was tested, it was integrated into the complete system. New functionality was added in steps until the complete system software came to fruition.

To start, a simple script was written to a single ESP32 to setup and read the temperature sensor and print the reading to the Serial Monitor. A similar program was then written for the turbidity sensor. The code became quite modular, and once a certain functionality was successfully implemented, it became a block of code to integrate into the overall system.

A program was written to transmit "Hello" from one ESP32 to another using the LoRa modules, and LED's were used to indicate the successful transmission and reception of a message. The code to read the temperature sensor was then integrated with the simple LoRa transmission code, which made it possible to transmit temperature readings from one node and receive and display them on the other. The turbidity sensor was neglected for the time being, as it made the code slightly more complex and offered little benefit. A counter variable was introduced at the transmitter. This was transmitted with the other variables, which made it possible to determine if any packets were lost. At this point, additional code was written to extract parameters such the Received Signal Strength Intensity (RSSI) and the Signal to Noise Ratio (SNR) and Packet Number. This system was used in the Range Test discussed later.

Code was developed to display text on the OLED screen. Once this was achieved, it was integrated into the rest of the system, which allowed the receiver to display the most recent message from the transmitter. At this point, communication occurred only from the transmitter to the receiver, so code from the transmitter was implemented on the receiver and vice versa to enable two-way communication.

The next step was to introduce a degree of remote control. A pseudo “TxState” variable was introduced, and code was developed to enable the user to provide input at the receiver which could be sent to the transmitter. Code on the transmitter could then extract the necessary information and update the “TxState” variable. Code was then implemented so that the transmitter was able to confirm the change with the receiver. Code was also implemented so that the receiver was able to request the state of this variable at any time.

To implement remote control via a user interface, the next phase of the program development focused on the implementation of MQTT. MQTT (Message Queuing Telemetry Transport) is a machine-to-machine/Internet of Things connectivity protocol (defined in ISO/IEC 20922:2016 [50]) that uses a publish-subscribe model to transport messages between devices. An MQTT network consists of clients communicating with a server, often called a *broker*. Information, in the form of *messages*, is published to *topics*, to which clients can either publish or subscribe. When a message is published on a topic, the broker distributes the message to all clients that are subscribed to the topic.

Node-RED is a visual programming development tool used to connect IoT devices. It essentially acts as a client subscribed to all topics and makes use of blocks called *nodes*, which when stringed together create a *flow*. The Node-RED Dashboard allows the developers to create interfaces through which they can interact with their devices and display things such as charts and recent messages and have inputs to their network such as switches and text fields. These inputs and outputs are all connected to clients through the *flow* and can be used to develop powerful IoT applications.

A private broker was created by installing the Eclipse Mosquitto broker software on a Raspberry Pi. The PubSubClient library was installed in the Arduino IDE on the ESP32, which handled all of the publish and subscribe functionality seamlessly. For this phase, LoRa transmission was ignored and the system was reduced to one node. The temperature sensor circuit from before was used to take temperature readings and upload them to Node-RED. A simple flow and a MINIONS Dashboard were created. The result was a simple network with a dashboard which allowed the user to view real-time temperature updates from the node. The readings were displayed in a chart. To test the backwards functionality of the system, another flow was created to allow the user to toggle an LED at the node via a switch on the dashboard. This feature opened the door to implementing a variety of new functionality.

The circuit which used two nodes to transmit temperature readings via LoRa was used in the next stage of development. The code which enabled interactions with the broker was implemented on the receiver, and so the system was able to transmit temperature readings from the transmitter to the receiver via LoRa, and the readings were published to the MINIONS dashboard via Node-RED. This system was close to what was required but needed several more features to be implemented. A text input field was added to the dashboard to enable the user to change the transmitter’s sampling period. This was used in conjunction with previously-implemented code to produce the required functionality. A button was added to the dashboard which enabled the user to restart the transmitter, adding a further degree of remote control and some recourse in the event of a malfunction. At this stage, the turbidity sensor was integrated, and the dashboard updated to display the new information.

User requirement SR4 stipulated that the power optimisation software be implemented. As mentioned in the literature review, a significant advantage of the ESP32 is its configurable power modes. The board has 6 configurable power modes, described in Table 4, which offer a variety of features and have current draws ranging from 68mA down to 0.1μA. Deep-sleep mode keeps the RTC timer and RTC memory active while disabling everything else, which makes it possible to use the timer to wake the board up and store a limited amount of memory during Deep-sleep. The program needed to store a counter variable in the RTC memory during sleep, and so Deep-sleep was favoured over Hibernation, which deactivates the RTC memory.

Up until this point, the code for the transmitter ran on a loop, and used a delay to change the sample period. However, when a program implements Deep-sleep mode, the program structure needs to be changed, as all of the functionality is carried out in the "setup" section of the program, and the "loop" section is never reached. As such, the program was restructured on both the transmitter and the receiver. This revealed flaws in the program, and so several improvements were made.

Errors occurred when trying to implement two-way communication with the new structure, so a new communication protocol was introduced. The transmitter would wake up, inform the receiver that it was awake, transmit its data, send a "clear to send" instruction to the receiver, then run through a loop where it would do nothing but listen for messages from the receiver. If the receiver had any messages, it would only transmit in this window, which ensured that no messages were lost. The transmitter would then carry out the instructions of the messages, inform the receiver that it was entering Deep-sleep mode, then go to sleep for the length of the sample period. The receiver ran on a constant loop listening for messages from the receiver. Messages received on MQTT topics to which the receiver was subscribed triggered an interrupt which would carry out any necessary instructions. Certain topics triggered the receiver to send messages to the transmitter (such as a period change or instruction to restart), and so these messages were placed in a queue and were sent only when the "awake" and "clear to send" messages received.
To protect the battery, a "CheckBattery()" function was implemented. This would read the output of the Voltage Monitor circuit via an analogue pin. The battery's cutoff voltage was listed as 2.75V in the device's datasheet [46]. A very conservative approach was taken to protect the battery, so the software ensured that the battery did not go below 3.3V. This was decided because the voltage regulator would no longer be able to output the required 3.3V output, so the circuit would no longer function. Another reason is that the circuit still consumes $0.1\mu A$ in *Power Off* mode, and the battery suffers from self-discharge, so if the Transmitter Node was unable to be retrieved for a long period of time, the voltage may reach the cutoff voltage. An equation mapped the voltages to a percentage, placing 0% at 3.3V and 100% at 4.2V. The "CheckBattery()" function monitored the percentage, and notified the Receiver Node and placed the Transmitter Node into and indefinite Deep-sleep if the percentage reached 0%. The ESP32 could only be woken up by a physical reset, so this essentially marked the end of the deployment.

All of the software development for the transmitter, receiver and Node-RED was done in conjunction. In the image below, the transmitter with the temperature sensor is shown at left and the receiver at right. Both were programmed in the Arduino IDE via USB cables connected to the laptop, while the Node-RED flow and dashboard were configured in the browser.

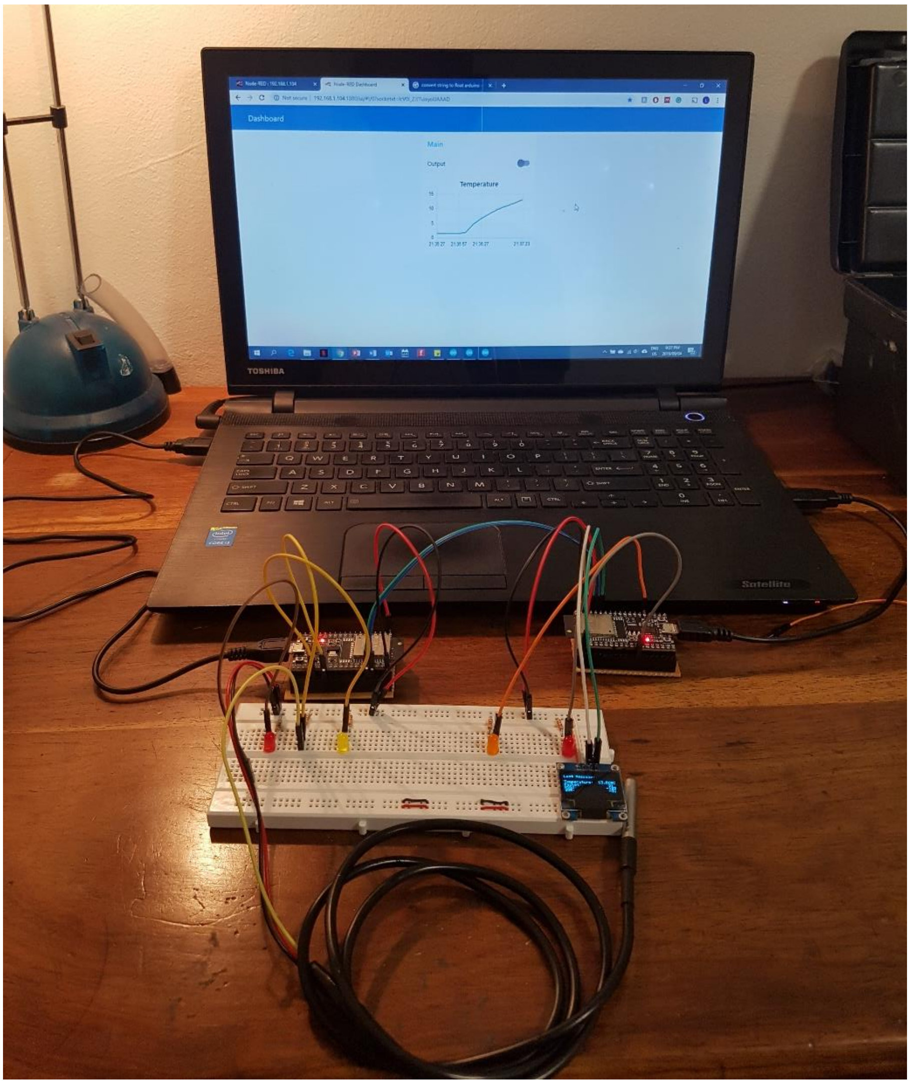

Figure 22: Image of the software development process, showing the Transmitter and Receiver Nodes connected to a laptop, displaying the MINIONS Dashboard

ii. *Implementation*

Flow charts of each of the programs are shown below. A separate flowchart was drawn to illustrate the interrupts generated by Node-RED.

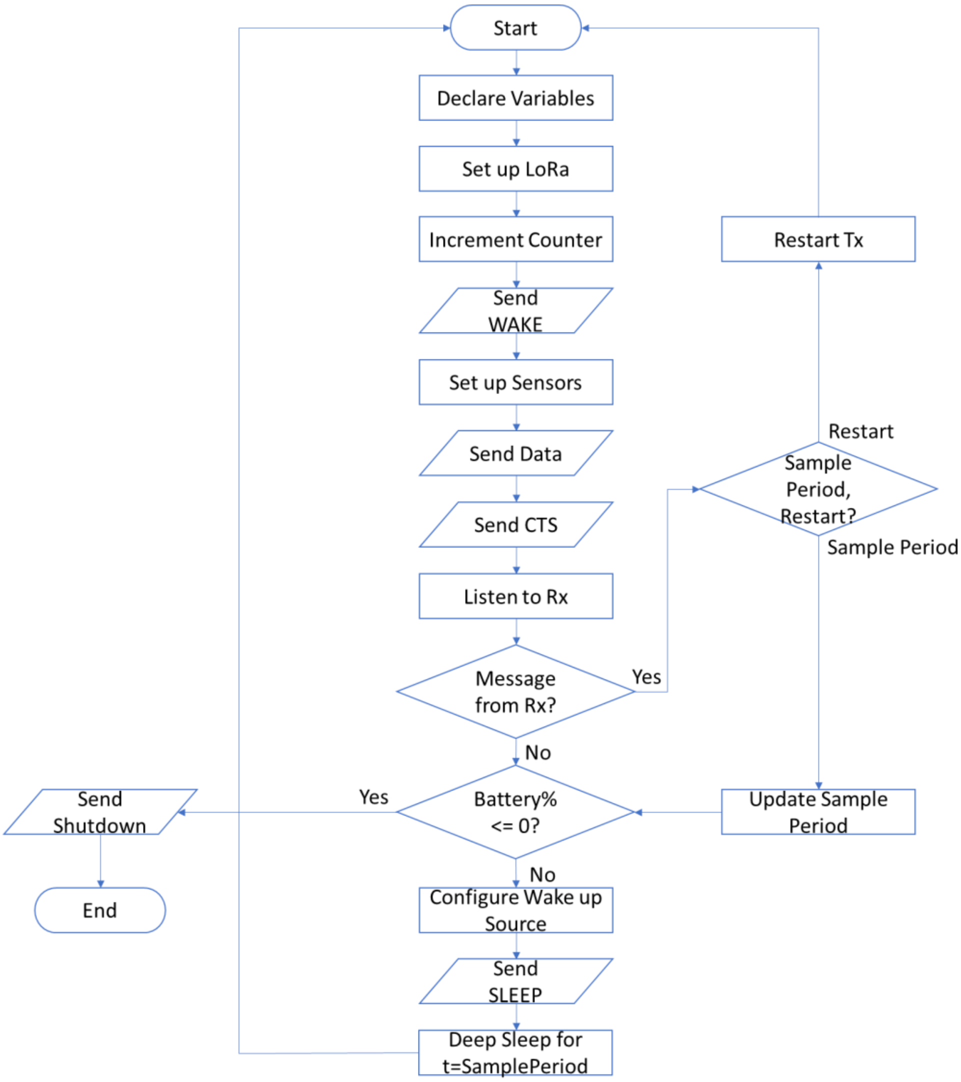

Figure 23: Software flowchart of the Transmitter Node

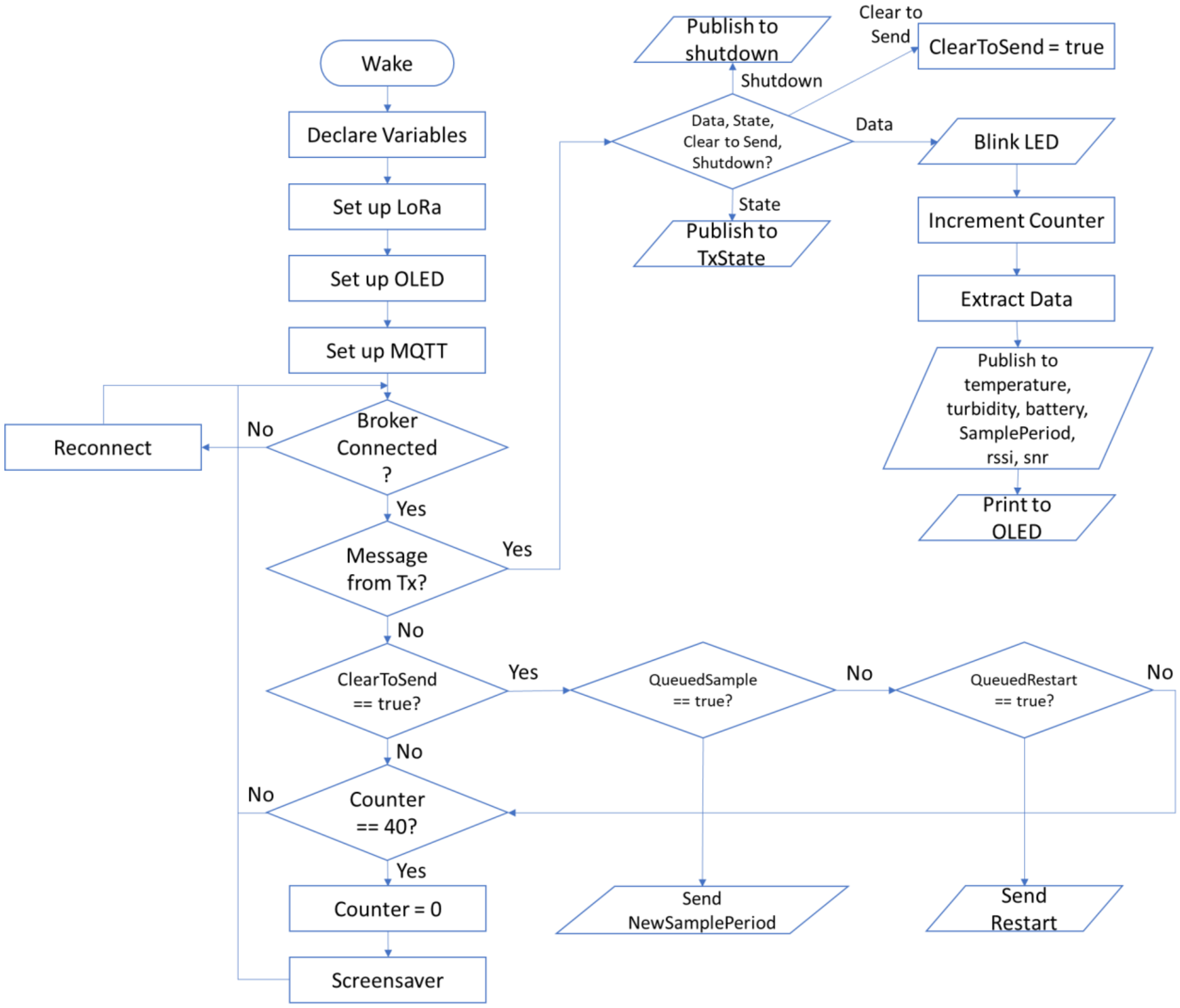

Figure 24: Software flowchart of the Receiver Node

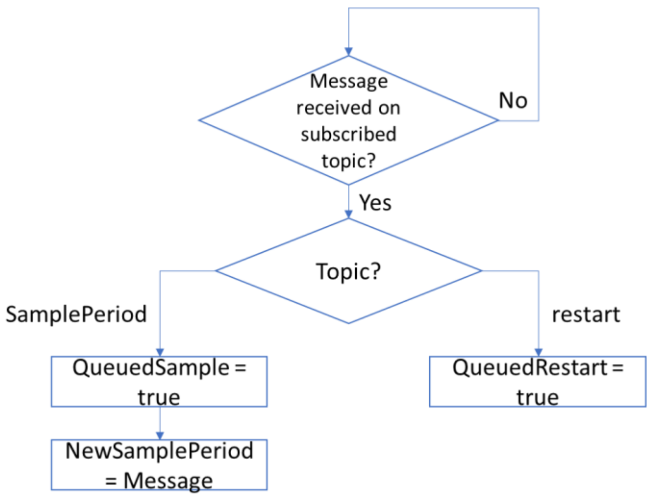

Figure 25: Flowchart illustrating the functioning of the interrupts generated by the MINIONS Dashboard

The Node-RED flow and Dashboard were configured as follows:
To display data on the MINIONS Dashboard, *MQTT input* nodes subscribed the system to the following topics:

- esp32/temperature
- esp32/turbidity

- esp32/battery
- esp32/rssi
- esp32/snr

Each of these nodes were connected to a *chart* node, when a message was received on each of the topic, the value was added to a chart on the MINIONS Dashboard. An *inject* node was configured to send an empty string ("[]") to each of the charts, which would clear them of all previous values, for display purposes.

A *text input* node displayed a text input field on the MINIONS Dashboard. This node was connected to an *MQTT output* node so that when a text input was added, it was published to the "esp32/SamplePeriod" topic.

An *MQTT* input node subscribed to the "esp32/TxState" topic was connected to a custom *template* node. This contained HTML code with placed an LED-like object on the MINIONS dashboard, which would appear green if "WAKE" was published to the topic, and grey if "SLEEP" was published.

Another *MQTT* input node subscribed to the "esp32/TxSamplePeriod" topic, connected to a *text output* node displayed the transmitter's sample period on the dashboard.

A *button* node connected to an *MQTT output* node displayed a button on the dashboard. This would publish "Restart" to the "esp32/restart" topic if the button was pressed. Code was implemented on the Receiver Node to transmit a restart command to the Transmitter Node, a feature which was explained above.

An *MQTT* input node subscribed to the "esp32/shutdown" topic was connected to a *show dialogue* node. When the Transmitter Node's battery was depleted, it would notify the Receiver Node, which would publish "Shutdown" to the "esp32/shutdown" topic. This would then open a dialogue box on the dashboard, informing the operator that the battery was depleted and that the Transmitter Node had entered its shutdown state.

Some of the *MQTT input* nodes were connected to *Modify Payload*, *csv* and *file* nodes. The *file* nodes were configured to "append to file" and the filename was created. This would save any messages received on the topic to a CSV file with a Unix timestamp, the message, and the topic on which the message was received. This was necessary to conduct data analysis at the end of the experiment.

An *inject* node was connected to several more *file* nodes, configured to "delete file", with the same filenames as before. This would delete the files that were previously created, so that the experiment had a clean slate.

Unfortunately, the MINIONS Dashboard was only visible within the local network. The Node-RED flow and its code are attached in Appendix B.

3.5.4 *Mechanical Design*

i. ***Enclosures***

For the electronics to function at sea, they needed to be placed in a secure, waterproof enclosure. In order to have an enclosure that could be included in the MINIONS Specification and be easily-replicable and uniform across all MINIONS projects, it was endeavoured to develop an enclosure that could be 3D printed. The literature review determined that PETG would be the best 3D-printable material for this purpose.

To save costs and enable rapid prototyping, an experimental setup was developed. This consisted of a small 3D-printed "test piece", and a system of pipes and valves (the "test rig") into which the test piece could be inserted. The test piece was printed with 100% infill to ensure the walls were solid. The test rig could be connected to; and pressurised by; a bicycle pump with a pressure gauge. The test piece was placed in a bucket of water while the rig was pressurised so that any escaping air could be seen. This would determine the pressure at which the piece failed. Using $P = \rho g h$, the depth in seawater at which the piece would fail could be calculated. This did, however, neglect the effects of prolonged immersion at pressure, but still gave an indication of the piece's waterproofness.

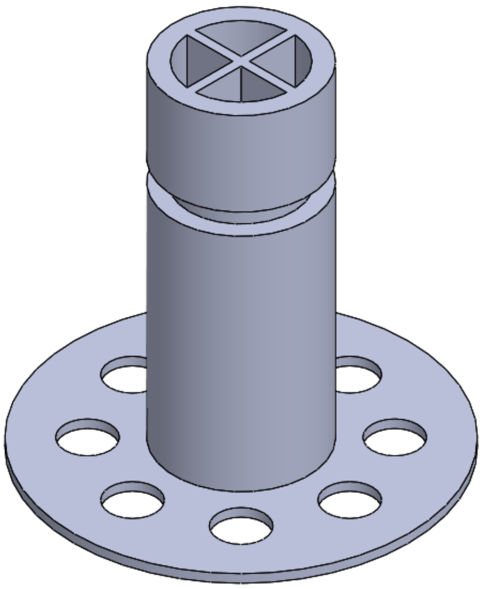

Figure 26: SolidWorks drawing of the "test piece" to be 3D printed

Figure 27: "Test rig" used during pressure tests

The test piece had a groove below the opening to allow an O-ring to be inserted, ensuring that any leaks came through the material itself and not the seal between the test piece and the test rig. As the piece is relatively tall and narrow, when the printer printed the upper layers of the piece, it would pull on it, and the greater moment about the base caused it to dislodge from the bed and the print to fail. A wide circle was added to the bottom of the piece to increase adhesion to the printing bed, holes were cut into it to save on material, and the piece was printed with 100% infill.

A control test was run on an untreated test piece. This allowed the inherent waterproofness of a 3D printed part to be determined. The piece was inserted into the rig with an O-ring, and a hose clamp was tightened around the piece to further enhance the seal and ensure that the pressure did not push the piece out of the rig.

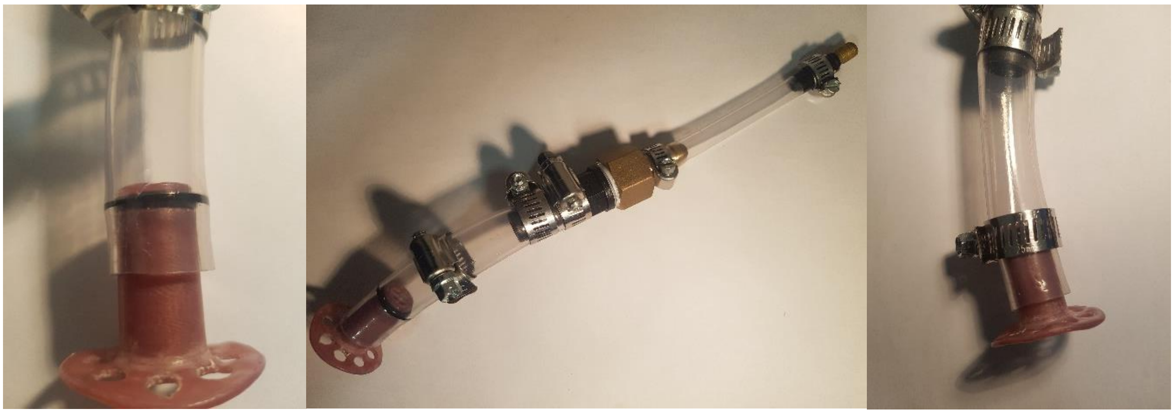

Figure 28: Images of the test piece being inserted into the test rig during the pressure test

The piece could not be pressurised, as air rushed straight through the walls of the test piece, proving that the untreated 3D print was not waterproof. As the printer prints each layer, the previous layer has already cooled, and so the bond between the layers is not perfect, resulting in slight gaps between them.

To improve this, two methods were used in an attempt seal these gaps. The first method involved heating ethyl acetate, a strong solvent, to its boiling point of 77.1°C. The 3D print can then be placed in the ethyl acetate vapour for about 15min, which lightly melts the outer layers of the print, causing them to bond. This process needs to be done in a well-ventilated area and the temperature needs to be carefully controlled, so a gas camping stove, together with a heat diffuser, was used outside while a temperature probe from a digital multimeter was used. The use of a gas stove is dangerous with such a flammable chemical, and it is difficult to control the temperature, so future recommendations will include the use of an electric stove with finer temperature control. The ethyl acetate was heated in a partially-sealed tin, and once it began to evaporate, the test piece was suspended inside to expose it to the vapour.

Figure 29: Depiction of the process of smoothing a PETG print using ethyl acetate

The print came out significantly smoother than before. The piece felt rubbery, yet stiff, and all of the fine grooves created by the 3D printing process were filled in. The visual appearance of the material changed slightly, and the thin, wide base warped and delaminated in areas.

Figure 30: 3D printed PETG test piece after ethyl acetate smoothing has occurred

The second method used a heat gun in slow passes over the piece to melt the outer layers and achieve a similar effect to the first method.

Figure 31: 3D printed PETG test piece after smoothing by means of a heat gun has occurred

With the heat gun, it difficult to heat the piece slowly and evenly, so the effect is difficult to achieve. The result is not ideal, as it leaves bubbles in the melted plastic and causes the piece to swell and deform. If this were used on a full-size enclosure, this may result in pieces not fitting together properly, leaving gaps and compromising its waterproofness. The heating also caused the plastic to become brittle, causing the wide base to snap off.

With the deadline fast-approaching, it was decided to abandon this endeavour in favour of a tried and tested construction technique. At this point in the project, the absence of a reliable enclosure was holding the project back, and there was simply not enough time to conduct thorough tests, iterate, try new techniques, and develop new prototypes until a final design could be chosen based on the 3D printing method.

Instead, it was decided to create a waterproof enclosure out of PVC piping and fittings, based on the designs shown in [8]. Two enclosures were constructed using a short length of 110mm PVC pipe, an *end cap*, and a *stopend access*. The stopend access was chosen because it acted as a screw-top lid to ensure easy access to the enclosure as per user requirement MR4. The fittings were bonded using PVC weld.

Figure 32: The enclosure constructed using PVC fittings

Each of the enclosure lids needed to be customised to suit their purposes. The main enclosure needed to have one cable exiting it, so a cable gland was installed in the lid. The sensor enclosure needed to have the same cable entering it, as well as a way for each of the sensors to be submerged.

Both enclosures needed to have an exit and entry point for the cable that connected the two, and so IP68-rated cable glands were installed into the lid of the main enclosure and the underside of the sensor enclosure. The sensor enclosure needed to allow the sensors to be submerged in the water. The temperature sensor probe was a 6mm metal rod. A 6mm cable gland was installed in the lid, through which the temperature probe was inserted, which ensured a waterproof seal and allowed the probe to be submerged.

Figure 33: External view of the temperature sensor waterproof housing, constructed using a 6mm cable gland

The turbidity sensor works by shining infrared light from a receiver to a transmitter and measuring the amount of light received. Clear water will allow most of the light to pass through, while murky water will absorb a lot of the light, so the sensor is thus able to give an indication of water clarity. This poses a problem, as the transmitter and receiver need to be practically submerged in the water yet protected from it.

The sensor came in a clear casing which was open at the back, so would only provide protection in roughly 20mm of perfectly still water. As such, a waterproof housing for the sensor needed to be created. While installing the other cable glands, the idea came to mind to use a large cable gland able to accommodate a cable the size of the sensor case. The gland works by compressing a thick rubber ring on the cable to create a seal, but unfortunately the gland was fractionally too small to fit both the rubber ring and the sensor case. The first iteration of the sensor housing used a small section of a discarded bicycle tube as a slightly thinner substitute for the thick rubber ring. The gland assembly is shown in Figure 34 below.

Figure 34: Deconstructed view of a 32mm cable gland

The new housing was tested in a pool. The test vaguely followed the requirement set out in ISO20653:2018 [51] for an IP68-rated enclosure, and both the main and sensor enclosures were submerged at a depth of 1m for one hour. The initial test was not successful, as the enclosures were completely waterlogged at the end of the test.

It was suspected that the enclosure failed at multiple points, and so several changes were made. The next iteration of the sensor enclosure included an additional rubber ring at the opening of the gland. The screw-top lid of the enclosure was only designed to act as an access point to piping and was not designed to hold pressure, so a bead of marine silicone was placed on the inside of the lid to form a seal as the lid was screwed shut.

Figure 35: Ring of silicone added to the inside of the lid to improve waterproofing

As extra precautions, thread tape (commonly used by plumbers to seal threaded connections) was placed on the screw thread and Vaseline was placed on the rim of the lid. This is an old sailor's trick to waterproofing and served the additional purpose of ensuring that the silicone ring was not ripped off of the lid as it was screwed down.

This iteration also failed the test but had far more promising results. The enclosure was no longer waterlogged, and it was suspected that the leak was coming from the housing of the turbidity sensor. In the third iteration of the sensor housing, the additional ring of the second iteration was discarded, and a layer of marine silicone was added to each surface of the gland as it was assembled. A final thick layer was added before the compression cap was screwed on, ensuring silicone was squeezed into every crevice.

Figure 36: External view of the turbidity sensor waterproof housing, constructed using a 32mm cable gland

This iteration was tested again and was partially successful. There were small droplets of water around the cable glands through which the cable connecting the two enclosures passed. This was due to the fact that the cable had air gaps between each of its internal cables. When the cable gland got compressed around the cable, it stretched and deformed the outer casing of the cable. Through the multiple tests that were conducted, this issue worsened to the point that there was no longer a perfect seal. It was too late in the project to find a substitute cable, so hot glue was used to seal the glands from the inside. The system was tested again and had no leaks. The enclosures were thus capable of withstanding greater pressures than they were planned to be deployed in.

The enclosures needed a way to be fastened to the buoy platform, and so a special bracket was designed in SolidWorks and 3D printed. The bracket was designed to fit around the enclosure and had 4 tabs to insert bolts to fasten the bracket and the enclosure.

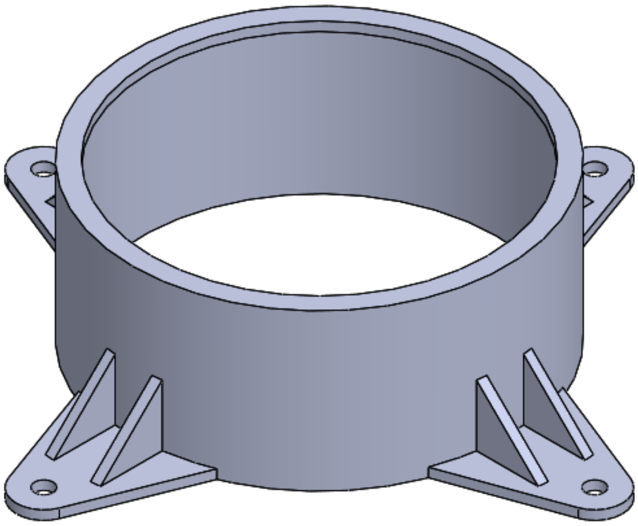

Figure 37: SolidWorks drawing of the enclosure bracket to be 3D printed

The bracket was printed to fit the enclosure exactly. The shrinkage and tolerances of 3D printing were not accounted for, and so the brackets did not fit. To save time and material, the brackets were adapted by cutting a single slit, which allowed it to expand slightly and fit around the enclosure.

Unfortunately, the filament in the department's 3D printer was changed from PETG to PLA, so the bracket does not comply with the MINIONS Specification and is not ideally suited for use at sea. The assembled enclosure is shown below:

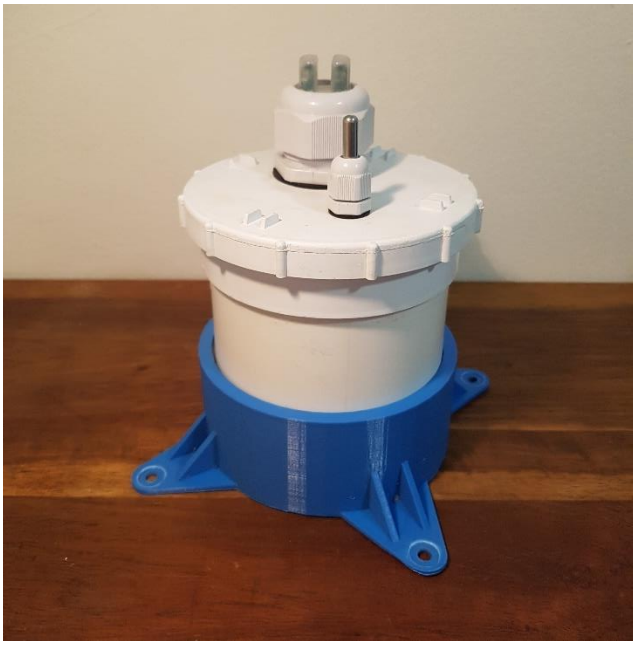

Figure 38: The assembled waterproof enclosure, ready for use at sea

ii. *Floating Platform*

The sensing node needs a stable, sufficiently buoyant platform to keep it afloat, as per user requirement MR3. The literature review identified toroid-shaped buoys as stable and popular platforms for marine-monitoring devices. A lifebuoy was thus chosen as the floatation device as these are wide, highly buoyant and very commonly found. Their shape and size is standardised in the SOLAS Treaty [52], making it suitable to include as a standard floatation device in the MINIONS Specification. A discarded lifebuoy was found at a local boat yard, shown in Figure 39 below.

Figure 39: Discarded lifebuoy to be reused as the MINIONS floatation device

The wide opening in the middle of the buoy made it possible to mount a flat plate onto which the enclosures could be mounted. The mounting plate was made out a 6mm sheet of HDPE plastic. The sheet was cut to fit the inner diameter of the buoy, holes were drilled to mount the enclosures, and unused space was cut away so that the buoy did not form a partial vacuum with the surface as it rose and fell with the waves.

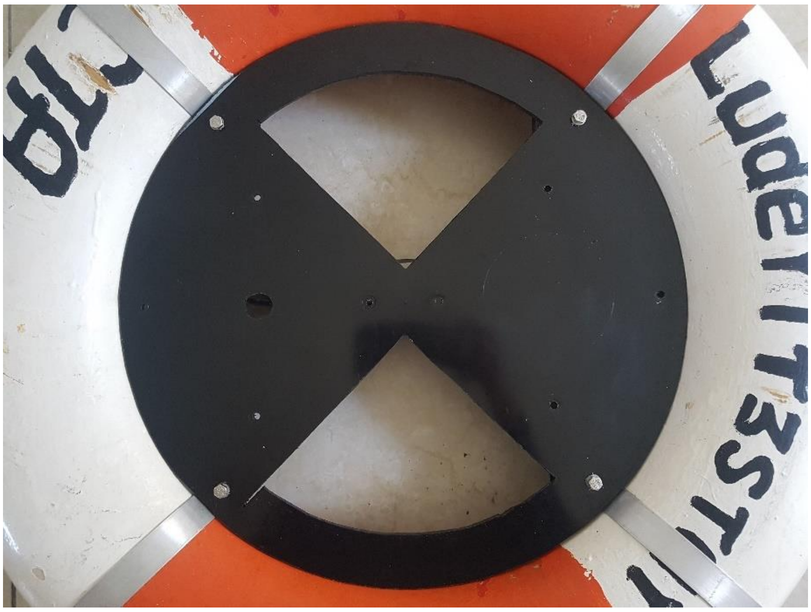

Figure 40: HDPE plastic sheet cut into shape to form the mounting platform, installed in the buoy

The plate was secured to the buoy with four brackets. These were made out of 1.6x20mm aluminium flat bar, bent into shape to fit the contours of the buoy. The bar wrapped around the outside of the buoy and met in the middle, where the ends where bent together to create an attachment point for the plate. Holes were drilled at this point, through which the plate could be fastened to the bracket by bolts. The bolts tighten the bracket around the buoy, and friction ensures that the plate and brackets stay secure.

Figure 41: Aluminium brackets made to secure the mounting platform to the buoy

M6, Type 316 stainless steel nuts, bolts and washers were used to fasten the enclosures to the platform and to secure the platform to the aluminium brackets. The assembly of the buoy, brackets and plate is shown below:

Figure 42: Complete assembly of the mechanical components of the Transmitter Node, consisting of brackets, plate and waterproof enclosures

Mechanical drawings of the MINIONS enclosure and mounting platform are attached in Appendix D.

iii. ***Mooring***

The platform needed to be anchored to ensure it did not drift out to sea. The buoy had attachment points for rope to be threaded through to create a handle for the user. The rope was removed, leaving sturdy attachment points to which the rest of the mooring system could be attached.

Figure 43: View of the buoy's attachment points

M10 bolts were threaded through the holes in order to attach a chain harness. The harness consisted of four lengths of chain, attached to the four attachment points, and held together in the middle by a D-shackle.

Figure 44 (left): View of the attachment point, with chain attached by an M10 bolt

Figure 45 (right): Complete view of the chain harness attached to the buoy

The shackle provided a central attachment point for the anchor so that the platform remained flat and stable in the water. When designing the mooring system, it was recommended to use an anchor, connected to a length of chain that would run along the sea floor for some distance, and then a length of nylon rope going up to the surface to connect to the buoy, as illustrated in Figure 46 below. This relieves tension in the system, allows for local tidal range (maximum 2.0m) and ensures that the buoy is not pulled underwater as waves pass by.

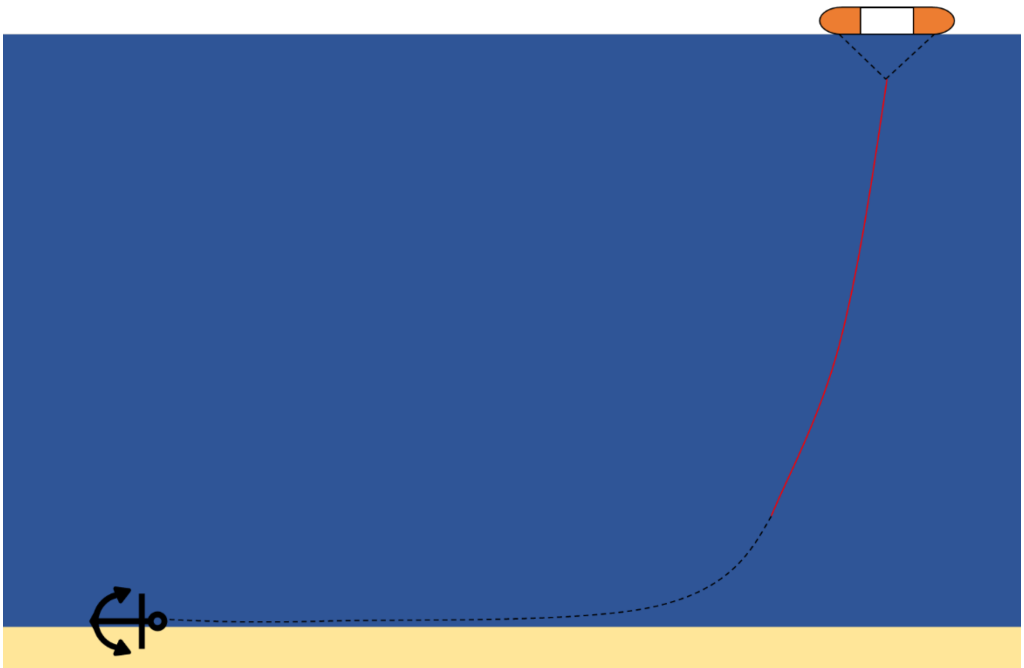

Figure 46: Illustration of the mooring system consisting of an anchor, chain and nylon rope, as suggested by Rick Harding

This creates the buoy's *watch circle*, which is the maximum radius in which the buoy is able to travel. This is important to calculate, as a large watch circle poses a hazard to sea traffic and could get the buoy caught in nearby obstructions. A local sailor estimated the depth at the proposed location to be between 6 to 8 metres. A 10kg anchor and 8m of chain was borrowed from a local lifesaving club. A 5m length of nylon rope was used to connect the chain to the buoy's harness, and so the total line length was 13m. The watch circle was calculated as follows:

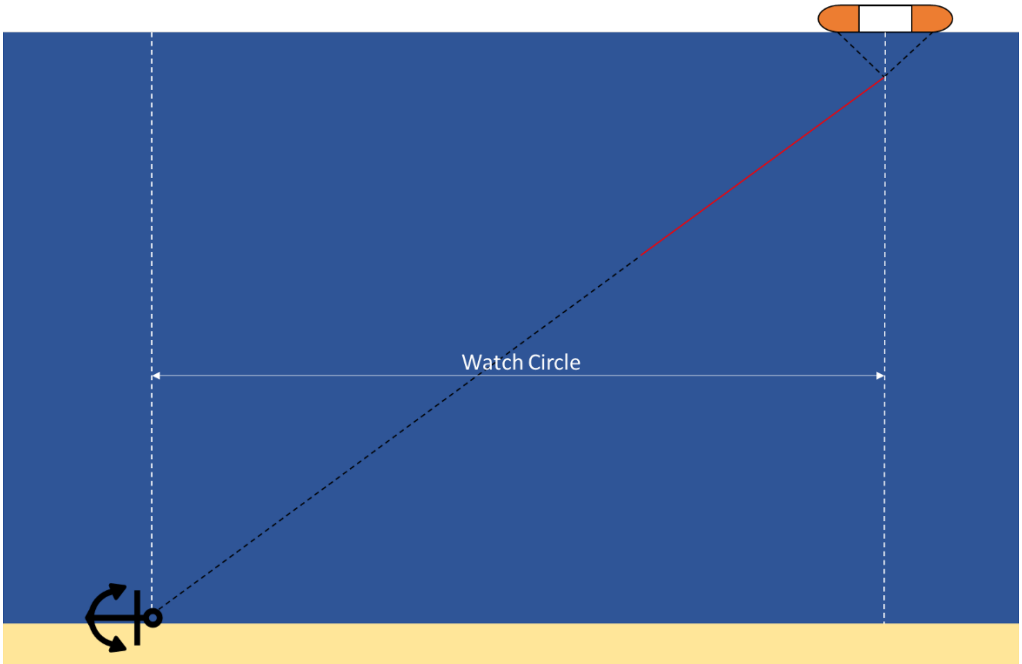

Figure 47: Illustration of the calculation of the watch circle

$$w_r = \sqrt{line\ length^2 - depth^2}$$
$$\therefore w_r = \sqrt{13^2 - 7^2}$$
$$\therefore w_r = 10.95m$$

Therefore, the buoy had to be placed at least 11 meters from the nearest obstruction. The chain, anchor, a small float attached to the anchor in order to locate it if it gets detached, and the nylon rope are shown below:

Figure 48: The chain, anchor, float and nylon rope used in the buoy's mooring system

3.6 Testing

3.6.1 *Range Test*

A test was conducted to investigate the range of the modules. The program mentioned earlier in Section X extracted the Packet Number, RSSI and SNR and printed it to the Serial Monitor. The experiment aimed to investigate the relationship between transmission distance and signal quality (measured by RSSI and SNR), and to see if the range claimed in the data sheet was true.

The test required a transmitter circuit, a receiver circuit, a laptop connected to the receiver to view messages printed to the serial monitor, and a car. The two circuits would be separated physically by known distances by driving along a predetermined route. Transmissions would occur constantly, and RSSI and SNR would be measured at designated points along the route, chosen at 500m intervals. The route and selected points are shown in Figure 49 below [53].

Figure 49: Illustration of the route followed during the Range Test, and the selected points of interest [53]

The transmitter ran on a constant loop, which transmitted and increased a “packet number” variable each loop and was left at the point labelled “Home”. The receiver was driven from point “Home” to point H and listened constantly for messages from the transmitter and printed the packet number, RSSI and SNR to the serial monitor. This helped identify whether any packets got lost and gave an indication of the signal quality of each received message. Point X is the approximate location of the final deployment, so while points C to H are all above sea-level, the test would additionally give an indication of the signal quality to be expected from the deployment site.

LED’s indicated the successful transmission and reception of a message, and a button allowed points of interest to be logged at the chosen points. The laptop stored the logged data and gave the driver assurance that the system was running properly. The experimental setup is shown below:

Figure 50: Images of the experimental setup inside the car used during the Range Test

The receiver was driven from the Base Station to a maximum distance of 4km (Point H in Figure 49). Data was collected driving both towards and away from point H, so all other points had two datasets.

3.6.2 Deployment Test

The deployment site was chosen to be the eastern side of Hout Bay, Cape Town. The site is approximately 100m from the beach and the nearest piece of coast. As seen in Figure 49 above, Hout Bay is very sheltered from the open-ocean, as swell has to bend into the bay and has thus dissipated most of its energy by the time it reaches the beach. The western side of the bay is even more sheltered than the eastern side, which is perhaps the reason for the location of Hout Bay Harbour. Unfortunately, this means that the western side has significantly more sea traffic and would also make the device easier to access by potential vandals. Thus, the eastern side of the bay was chosen to minimise the hazard to sea traffic. The Transmitter Node was left at the location marked by position X in Figure 51 below. The distance from the Receiver Node, marked by the position labelled "Base Station", is approximately 1.46km [53].

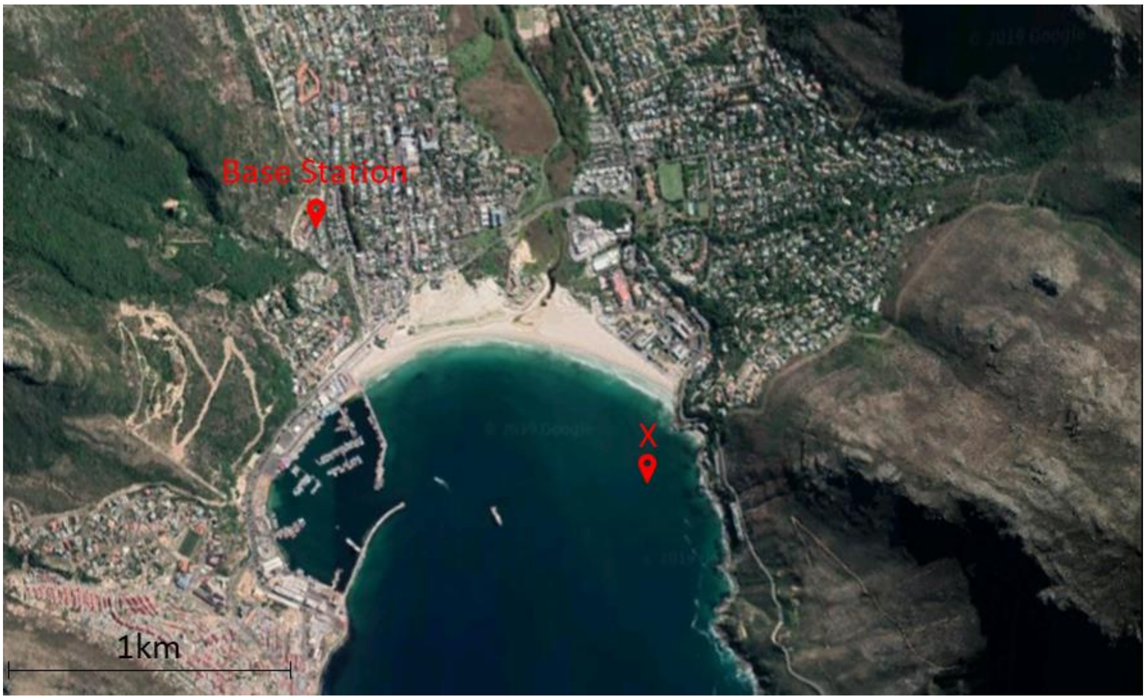

Figure 51: Satellite view of the locations of the two nodes during the Deployment Test [53]

The Transmitter Node was transported to the deployment site on a kayak. Once the site was reached, the anchor was dropped and set. Final checks were conducted, and the node was left at sea at approximately 17:30 on 7 October 2019.

This day was chosen to deploy as it had the smallest swell in the period, with the swell height ranging between 2.7 and 3.6 meters [54]. The node was retrieved at approximately 13:00 the following day, as the swell was forecast to increase to 5.8 meters overnight, which posed a threat to the node.

Data was logged throughout the deployment, the MINIONS Dashboard was monitored to ensure that data was being received successfully and the functionalities of the sample period change and restart features were tested. Measures of the system's performance and success included RSSI, SNR, its ability to perform the required functionality, and the number of corrupted packets.

4. Results

4.1 Range Test

Recordings of RSSI, SNR and Packet Number were logged at each of the points of interest. Each point of interest had multiple datasets on both legs of the trip, so average values were calculated for each point. The results of the Range Test are shown in Figure 52 and Figure 53 below:

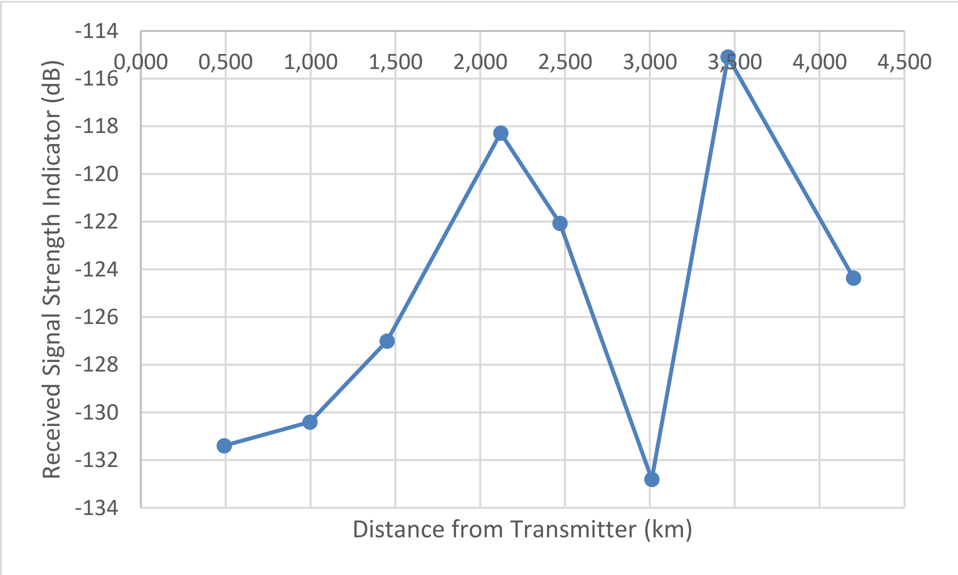

Figure 52: Received Signal Strength Indicator measured by the receiver at various distances from the transmitter during the Range Test

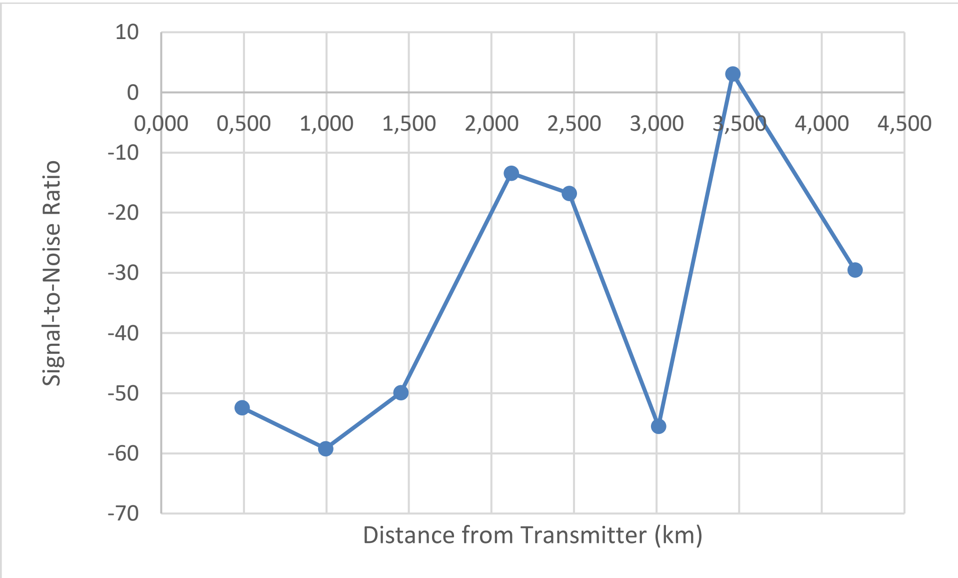

Figure 53: Signal-to-Noise Ratio measured by the receiver at various distances from the transmitter during the Range Test

4.2 Deployment Test

4.2.1 *Received Data*

Temperature, turbidity, battery percentage and sample period were recorded by the Transmitter Node and transmitted to the Receiver Node. The Receiver Node recorder RSSI and SNR and uploaded all six variables to the Node-RED Dashboard for the duration of the deployment. The values were recorded in CSV format and the results are shown in Figure 54 toFigure 59 below.

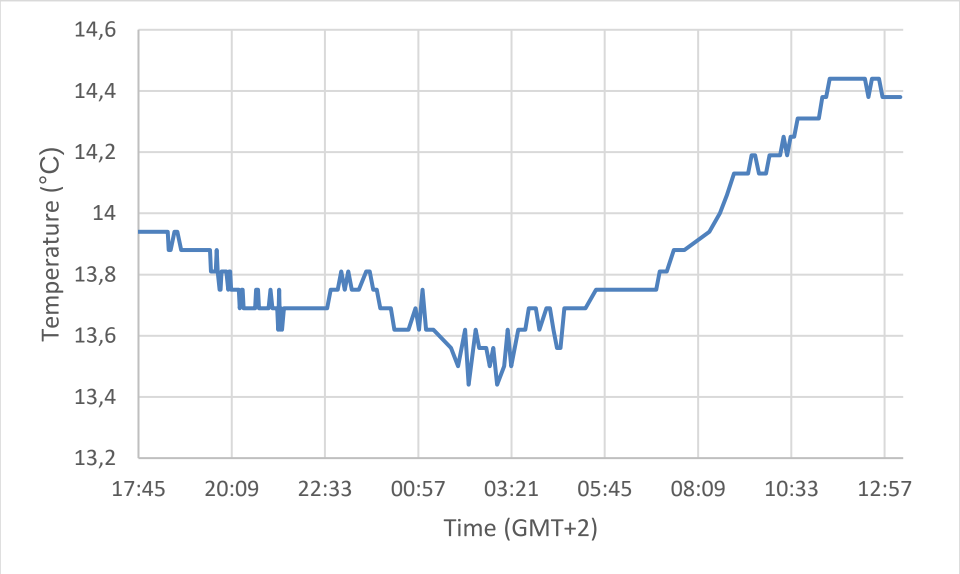

Figure 54: Water temperature as measured at the Transmitter Node during Deployment Test, 17:45 SAST 07-10-19 through 12:57 SAST 08-10-19

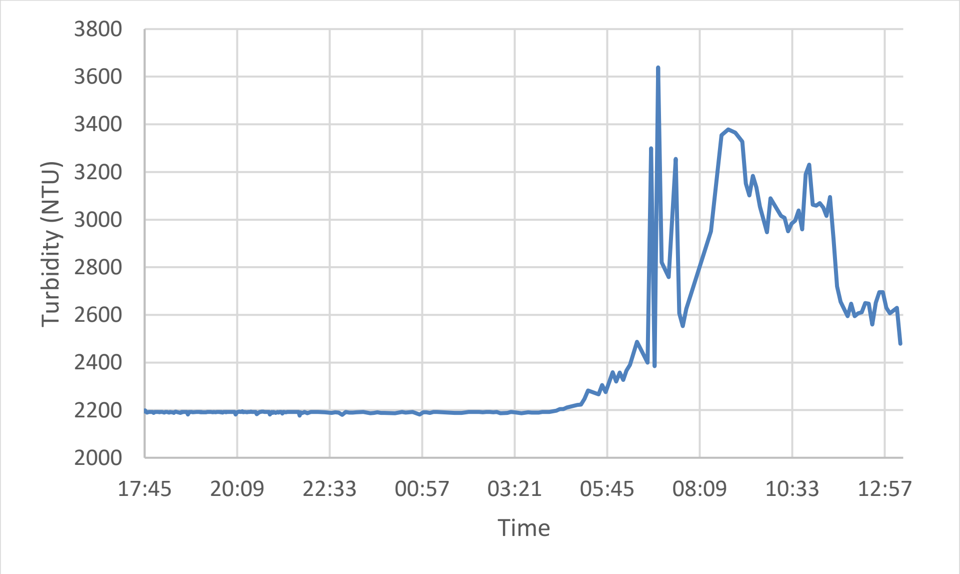

Figure 55: Water turbidity as measured at the Transmitter Node during Deployment Test, 17:45 SAST 07-10-19 through 12:57 SAST 08-10-19

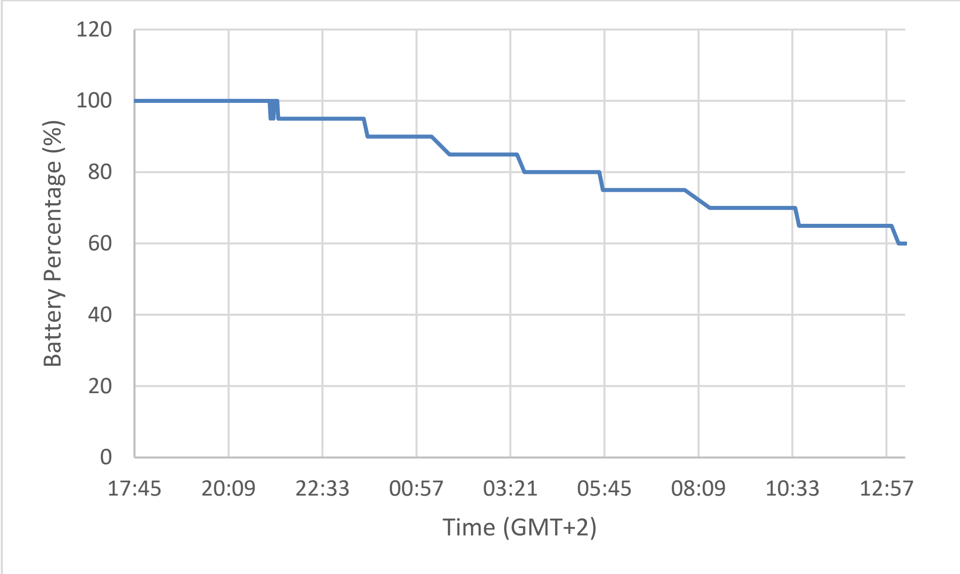

Figure 56: Battery Percentage as measured at the Transmitter Node during Deployment Test, 17:45 SAST 07-10-19 through 12:57 SAST 08-10-19

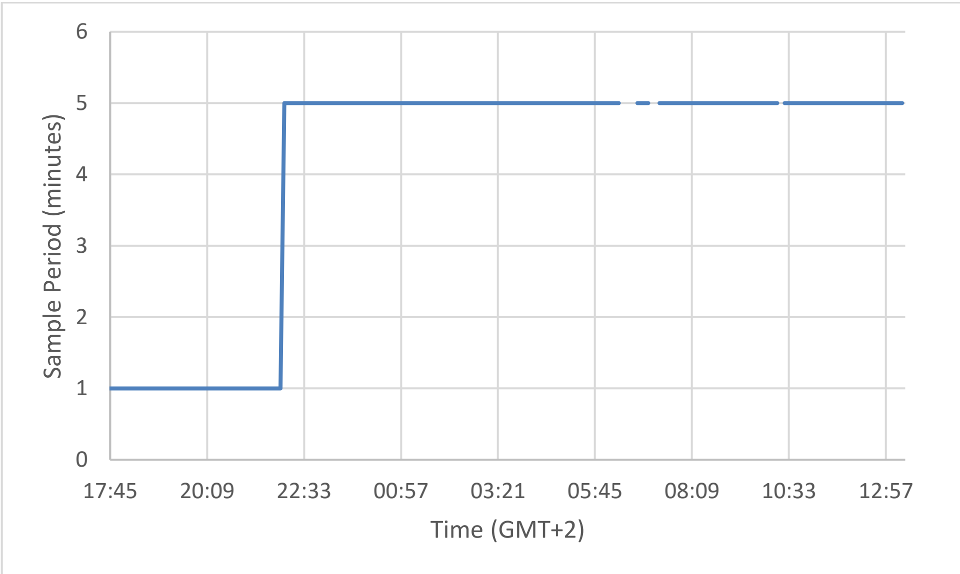

Figure 57: Sample Period of the Transmitter Node during Deployment Test, 17:45 SAST 07-10-19 through 12:57 SAST 08-10-19

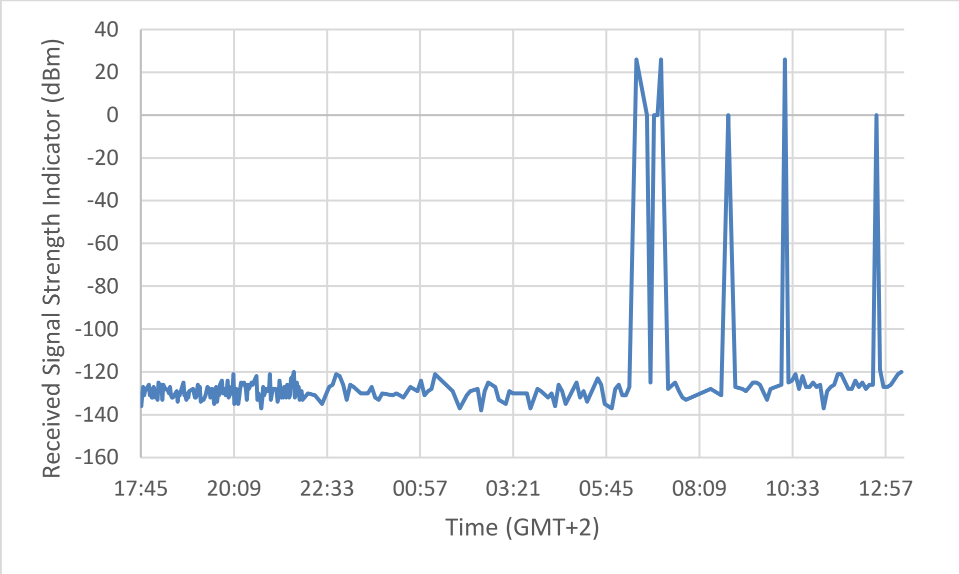

Figure 58: Received Signal Strength Indicator as measured at the Receiver Node during Deployment Test, 17:45 SAST 07-10-19 through 12:57 SAST 08-10-19

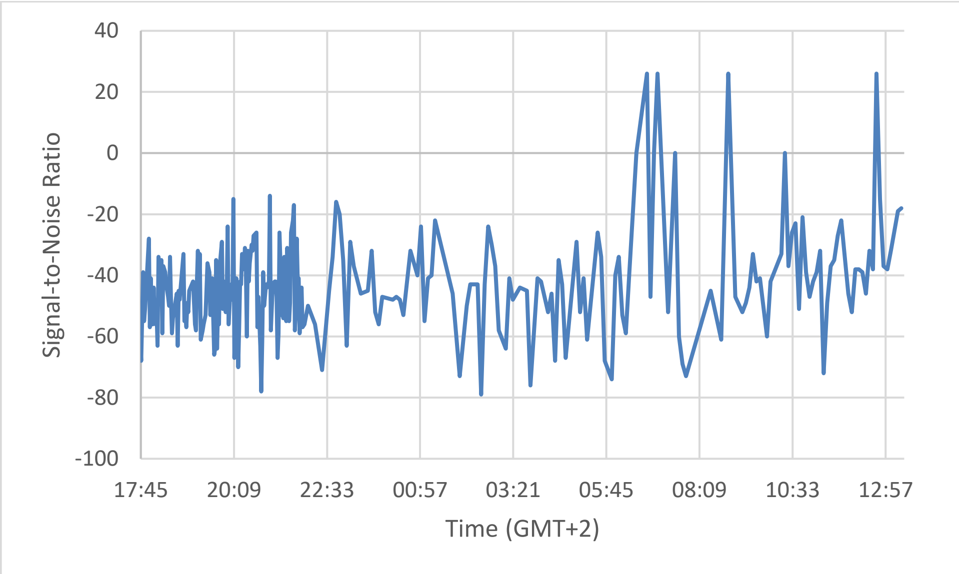

Figure 59: Signal-to-Noise Ratio as measured at the Receiver Node during Deployment Test, 17:45 SAST 07-10-19 through 12:57 SAST 08-10-19

4.2.2 User Interface

A screenshot of the MINIONS Dashboard during the deployment is shown in Figure 60 below. This was taken at 08:54 on the morning of the retrieval, so does not display all of the data illustrated in Figure 54 toFigure 59 above.

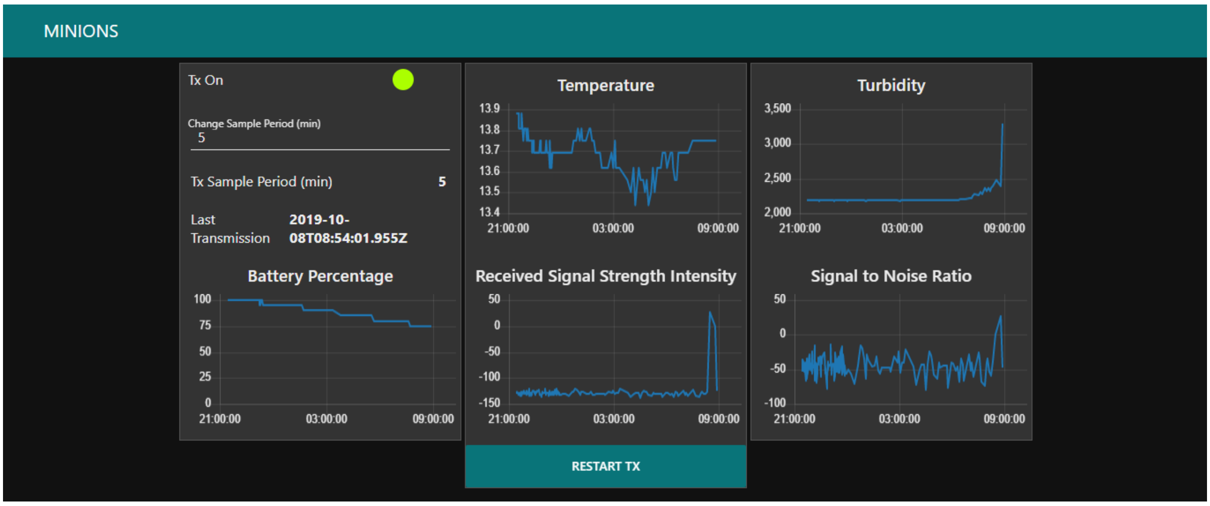

Figure 60: Screenshot of MINIONS Dashboard taken during Deployment Test, showing data recorded between approximately 21:00 07-10-2019 and 8:54 08-10-2019

The Transmitter Node's battery was depleted to a minimum of 60% during the deployment, so the shutdown feature was never activated. However, this was simulated after the deployment by connecting the GPIO pin dedicated to reading the output of the Voltage Monitor circuit to ground. The following response was generated in the MINIONS Dashboard:

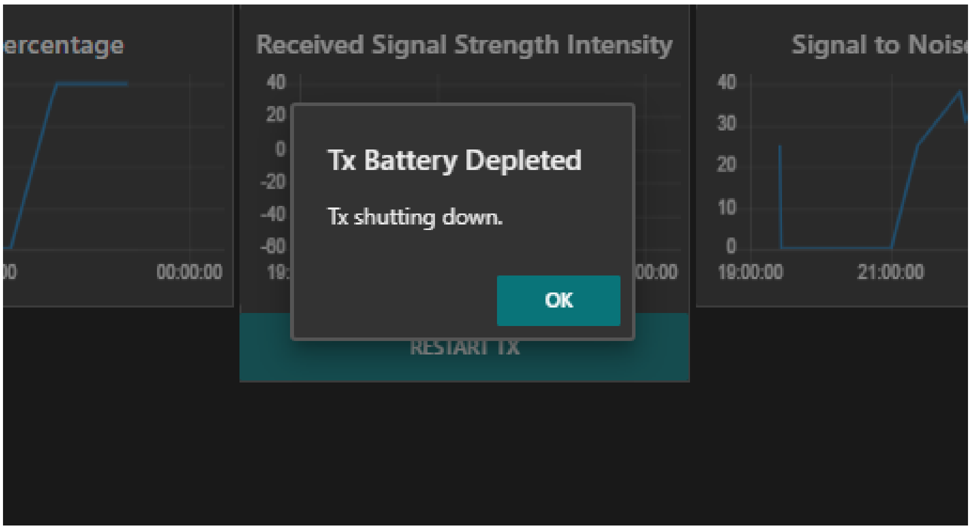

Figure 61: Screenshot of the MINIONS Dashboard taken during simulation of a depleted battery at the Transmitter Node

The only way to prove whether or not the system's restart functionality was successful was to observe whether the "Packet Number" variable had reset to 1. This variable is initialised when the node starts up for the first time and is incremented each time the node wakes up from Deep-sleep. Thus, restarting the node forces it to initialise the variable once again, setting it to 1. Unfortunately, the value of this variable was not recorded in CSV format over the course of the deployment. However, the Receiver

Node displays this variable to its OLED screen, so photographs of the screen were taken before and after a restart event to prove the functionality. These are shown in Figure 62 and Figure 63 below.

Figure 62: OLED screen at the Receiver Node displaying initial Packet Number before a restart event

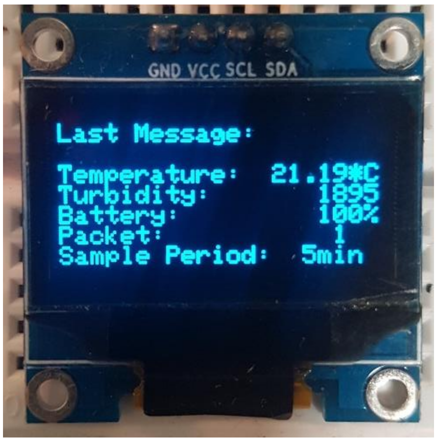

Figure 63: OLED screen at the Receiver Node displaying Packet Number reset to 1 after a restart event

4.2.3 *Mechanical System*

Upon retrieval, the enclosures were opened to inspect for leaks. The images shown in Figure 64 and Figure 65 below were taken at the time and show that there was no water inside either of the enclosures.

Figure 64: Internal view of the upper enclosure after deployment, showing no signs of leaks

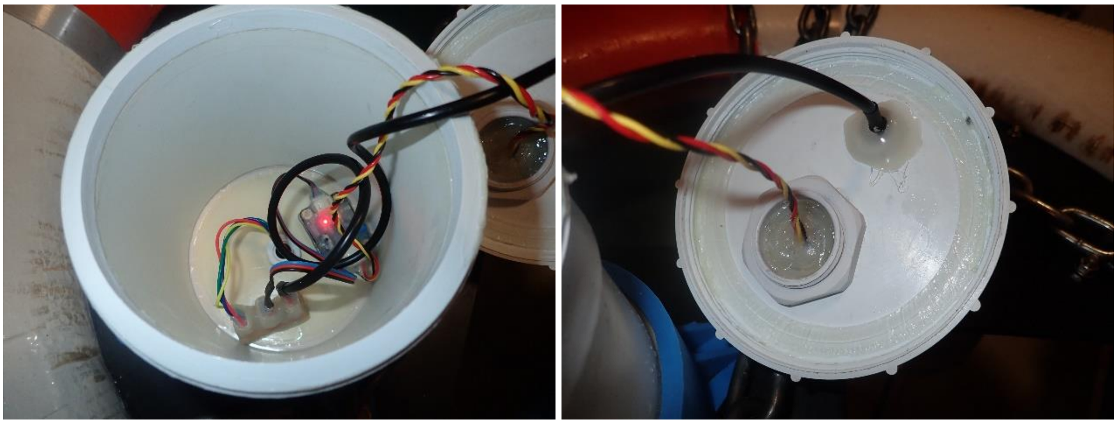

Figure 65:Internal view of the lower enclosure after deployment, showing no signs of leaks

While the node was being deployed, the lower enclosure was knocked, causing the 3D printed bracket to break at 3 of its 4 tabs. A length of rope was used to secure the enclosure to the platform for the experiment, as there was no time to postpone. The broken bracket is shown in Figure 66 below.

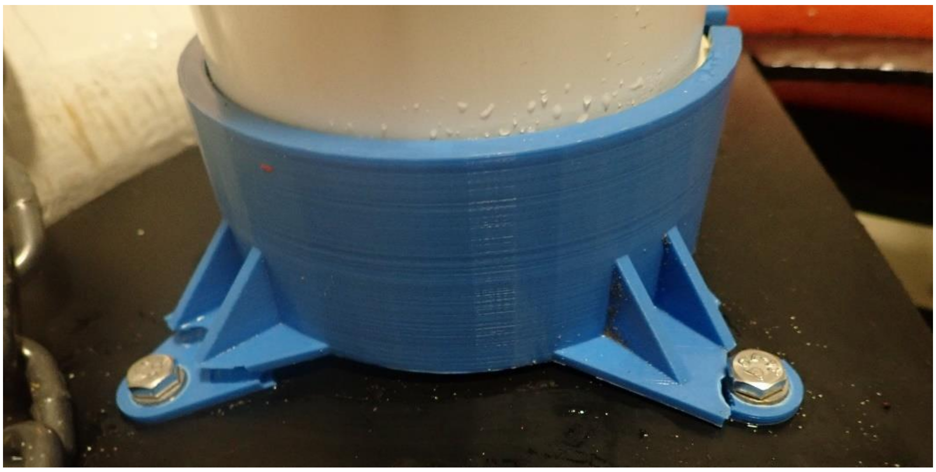

Figure 66: Photograph of the broken bracket of the lower enclosure

The chain used for the harness was shiny before the deployment, but after only 19 hours at sea, was visibly corroded. The Type 316 stainless steel fasteners and aluminium brackets were not corroded, although some of the large bolts had rust on their surface from being in contact with the chain.

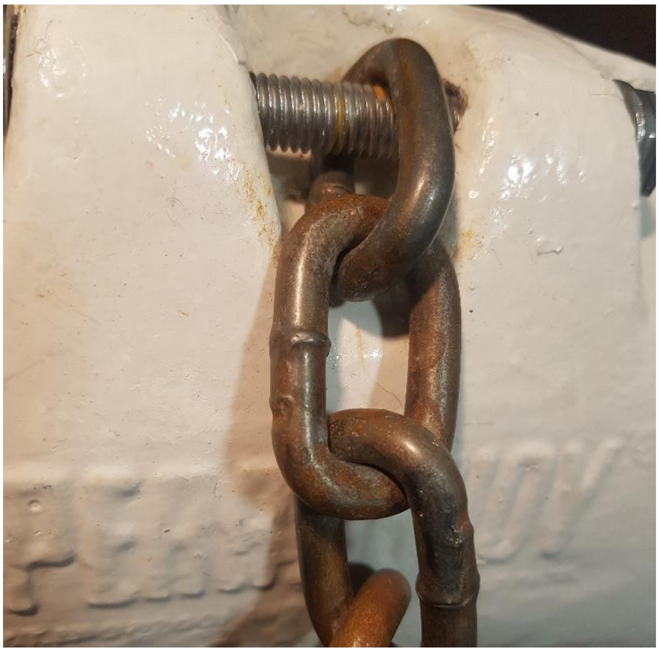

Figure 67: Corrosion on the chain harness after 19 hours at sea

5. Discussion

5.1 Range Test

The results show that the relationships between RSSI and distance, and SNR and distance are not linear. It can also be seen that RSSI and SNR are closely related. A high RSSI indicates that the strength of a signal is high. One would expect that the signal would be strongest at smaller distances, and thus that the RSSI would be higher. However, the results show that in the experiment, this was not the case. In fact, the strongest signals were received at approximately 2, 2.5 and 3.5km, while the weakest signals were received at 0.5, 1 and 3km. The strongest and second-weakest signals were received at consecutive points (Points F and G) in Figure 49.

The reason for the non-linear relationships is the degree of obstruction between the transmitter and receiver. Between 0.5-1.5km, both the transmitter and receiver were located less than 10m above sea level. There are many buildings between these points, and the fact that the circuits were at roughly the same, low altitude meant that the signal had to travel through multiple buildings to reach the receiver. Point F is located in a ravine and so is heavily obstructed by a mountain. Due to the severe degree of obstruction at these points, the RSSI and SNR is very low. At points G and H, there is almost direct line-of-sight due to the location and elevation, so the RSSI and SNR are significantly stronger than at other points.

Figure 68 below illustrates the “degree of obstruction” between the transmitter and receiver along the route. This was an estimated measure determined by the extent to which buildings and mountains obstructed a line-of-sight connection between the “Base Station” and the point.

Figure 68: Illustration of the degree of obstruction between transmitter and receiver along the route followed during the Range Test

5.2 Deployment Test

5.2.1 Received Data

A large period of consecutive, anomalous readings occurred between 06:32 and 07:32 on 08-10-2019, as illustrated in Table 11 below. An “X” indicates that an anomalous reading was recorded for the measurand at the listed time.

Time (GMT+2)	Turbidity	Sample Period	RSSI	SNR
06:32		X	X	X
06:48				X
06:54	X			
06:59			X	
07:04	X			X
07:10	X	X		
07:21	X			
07:32	X			X

Table 11: Anomalous readings recorded between 06:32 and 07:32 08-10-2019

Table 12 below illustrates other anomalous readings outside of the period mentioned above.

Time (GMT+2)	Turbidity	Sample Period	RSSI	SNR
08:54			X	X
10:21		X	X	X
12:43			X	X

Table 12: Other anomalous readings recording on 08-10-2019

This was an issue experienced regularly in design testing. The LoRa modules work by sending “AT” commands to the module. Messages are sent as strings containing send command, receiver address, payload length and the payload, via the serial port. For example, to transmit the word “Hello” to a receiver with Address 0, the microcontroller would write “AT+SEND=0,5,Hello”. The receiver module would then receive a string containing the receive command, transmitter address, payload length, payload, RSSI and SNR. In this example, the received string would be: “+RCV=1,5,Hello,-99,40”.

However, the last few characters of the string would often become scrambled, with messages such as: “+RCV=1,5,Hello,-9$%??”. The block of code used to extract variables from the received messages was not equipped to cope with this error, which resulted in erroneous readings. The root of the cause is unknown, and the presence of the error is difficult to predict. The structures of the transmitted and received messages is shown below:

Transmitted Message	Received Message
“AT+SEND=<Receiver Address>,<Payload Length>,<Temperature>+<Turbidity>+<Batt ery >+<Sample Period>+<Packet Number>”	“+RCV=<Transmitter Address>,<Payload Length>,<Temperature>+<Turbidity>+<Battery>+<Sa mple Period>+<Packet Number>,<RSSI>,<SNR>”

Table 13: Explanation of the transmitted and received messages using AT commands with the REYAX RYLR896 LoRa Modules

As the temperature reading was transmitted first, it was least likely to be scrambled. Thus, the temperature data had no erroneous readings. Meanwhile, sample period, RSSI and SNR recorded 3, 5 and 7 erroneous readings respectively. The error is a known occurrence, so it can be concluded with confidence that the readings are indeed erroneous. In total, 11 obviously-erroneous packets were recorded out of a total of 258, resulting in a percentage error of 4.26%.

i. ***Temperature***

The following sea surface temperature data was recorded by the Council for Scientific and Industrial Research's (CSIR) Wavenet system [55]. The data is gathered by a directional DataWell Waverider buoy, located 5.4km off the coast of Cape Point in 70m of water. This buoy is located 18.09km from the deployment site of the MINIONS Transmitter Node. Unfortunately, it not possible to get datapoints from the website, but a graph of the data gathered over the same period in which the MINIONS Transmitter Node was deployed is shown in Figure 69 below. The period of the deployment is outlined in red.

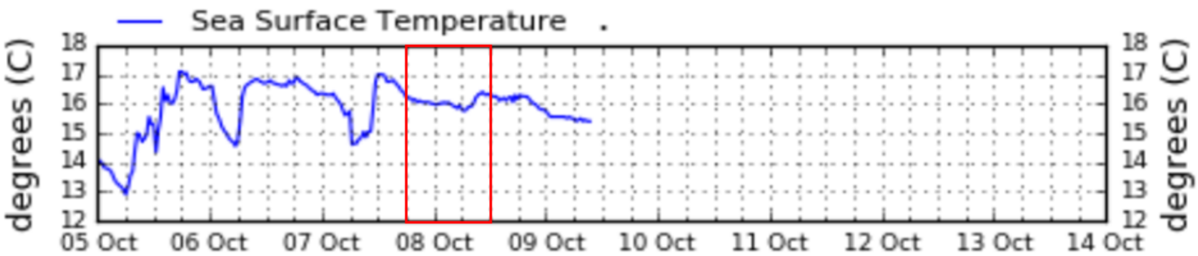

Figure 69: Sea surface temperatures recorded by Wavenet near Cape Point during the period 05-10-2019 and 09-10-2019 [55]

Figure 69 illustrates a general decline in temperature for the first twelve hours of the period, followed by a rise and subsequent decline in the last six hours of the period. Between 00:00 and 6:00, there is a slight rise in temperature, followed by a decline to the minimum temperature seen over the period.

The graph showing water temperature data gathered by the MINIONS Transmitter Node follows a similar shape to that of Wavenet. Figure 54 shows that the temperature reached a minimum of 13.44°C at 2:15, and a maximum of 14.44°C at 11:59. There is a general decline for the first nine hours of the period, with a slight rise, followed by a decline to the minimum temperature at 2:59. There is then a gradual rise to the maximum temperature at 11:49, followed by a small decline. However, the timing of each of the rises and falls differs by up to 3 hours between the two graphs, and the temperatures recorded by the Wavenet system are up to 2.4°C higher than those recorded by the MINIONS Transmitter Node.

ii. ***Turbidity***

Figure 55 shows that between 17:45 07-10-2019 and 04:38 08-10-2019, the turbidity readings stayed almost constant, with an average value of 2191.03 NTU, and a standard deviation of only 2.56883. A rise in the turbidity readings began at 04:38 on 08-10-2019. On this day, first light was recorded at

05:46 and sunrise at 06:12 for the greater Cape Town area [56]. The turbidity peaks at 3365 NTU recorded at 09:05 and begins to fall at approximately midday.

The following weather observations were recorded at Cape Town international airport on 08-10-2019 [57]:

Time (GMT+2)	Condition
04:00	Mostly Cloudy
05:00	Fair
06:00	Partly Cloudy
06:28	Fair
06:57	Partly Cloudy
07:00	Partly Cloudy
08:00	Fair
08:26	Fair
09:00	Partly Cloudy
10:00	Mostly Cloudy
11:00	Partly Cloudy
12:00	Fair
13:00	Partly Cloudy

Table 14: Weather observations recorded at Cape Town International Airport between 04:00 and 13:00 on 08-10-2019 [57]

This weather station is located approximately 23km from the deployment site, so the data does not give an accurate indication of the cloud cover at the deployment site at the time. It does, however, suggest that the changes in turbidity may be partially attributed to the amount of cloud cover at the time. The peak in turbidity readings at 09:05 may be attributed to the period of sparse cloud cover between 07:00 and 09:00, while the decline in turbidity readings may be attributed to the increased cloud cover seen later in the morning.

The sensor uses an infrared transmitter and receiver to measure the light transmittance and scattering rate, which changes with the amount of Total Suspended Solids (TTS) in the water. It is therefore affected by the amount of sunlight falling on it, as this would increase the amount of infrared light reaching the receiver, which is the reason for the increase shortly before sunrise, and the rise and fall due to cloud cover, shown in Figure 55.

iii. ***RSSI and SNR***

If the erroneous readings are ignored, the RSSI had an average value of -129.401 dBm with a standard deviation of 3.461602. This means that the signal strength was consistently weak. However, -120 dBm is listed as a typical RSSI for LoRa devices [58], and the average value is well within the RYLR896's RF Sensitivity of -148 dBm.

The SNR had an average value of -46.2935 with a standard deviation 12.81126. LoRa is designed to operate below the noise floor, so a negative value is not alarming. However, the SNR is very large and negative, which indicates that the received signal had significantly less power than the noise floor.

5.2.2 *User Interface*

The results of the Deployment Test show that the Transmitter Node successfully transmitted data to the Receiver Node, and that this data was successfully uploaded to the MINIONS Dashboard. The system was able to indicate the state of the transmitter as either awake or asleep, indicated by a circle changing to either green or grey at "Tx On" in Figure 60. The Transmitter Node sent constant, accurate updates of its sample period, shown by the graph in Figure 57, as well as at "Tx Sample Period" in Figure 60. It also successfully notified the system of a depleted battery and shut itself down accordingly, as shown in Figure 61.

Figure 62 and Figure 63 demonstrate that it was possible to restart the Transmitter Node via the MINIONS Dashboard. As can be seen in Figure 57, the sample period was changed from 1 minute to 5 minutes at 22:15 on 07-10-2019. This demonstrates that it was possible to change the sample period of the Transmitter Node via the MINIONS Dashboard. These two facts prove that a degree of remote control was successfully implemented into the system. The dashboard displayed the time of the last successful transmission. In conjunction with knowledge of the Transmitter Node's sample period, this allowed the user to identify and missed packets or malfunctions.

5.2.3 *Mechanical System*

The buoy platform was strong, buoyant and stable. It weathered rough conditions and showed no signs of wear and tear. The mooring system performed well, as the buoy remained where it was deployed and did not come into contact with any obstructions. The materials fared well in the rough conditions and showed no signs of weathering and corrosion.

The enclosures were waterproof and were reasonably accessible. The addition of the silicone ring increased waterproofness but meant that screwing and unscrewing the lid of the enclosure required significant force. A screwdriver was used to tighten the lid but was also needed to loosen it. The fact that some of the cables and sensors went through the lids meant that care had to be taken so as not to twist the cables, potentially ripping them out of their connectors.

The enclosures were based off of the design used in [8]. PVC has poor UV-resistance and is thus poorly suited for use at sea. Thus, the authors used exterior latex paint to increase the UV-resistance. No such measures were taken with the enclosures used in the MINIONS Transmitter Node, so it is expected that they would show severe weathering from prolonged deployment. Additionally, the UV-resistance of the cable and cable glands used to connect the two enclosures was not considered, so this too may be a point of weakness for prolonged deployment. Importantly, the PVC of the enclosures did not block the signal of the LoRa modules.

The 3D printed bracket made to secure the enclosure to the platform performed poorly. The fact that it simply slipped over the enclosure meant that the enclosure was free to rotate inside the bracket, which made it difficult to tighten the lid. The bracket was printed in PLA, so was not suited for use at sea and UV-exposure and did not comply with Requirement 2.2.3 of the MINIONS Specification. The brittleness of the material meant that it shattered with very little force, which almost put the deployment in jeopardy. Additionally, fragments that had broken off had to be collected to comply with Requirement 1.2 of the MINIONS Specification, which stipulated that no additional debris be created. The bracket was printed with 50% infill, so when the bolts were tightened, the tabs were crushed, compromising the bracket's strength.

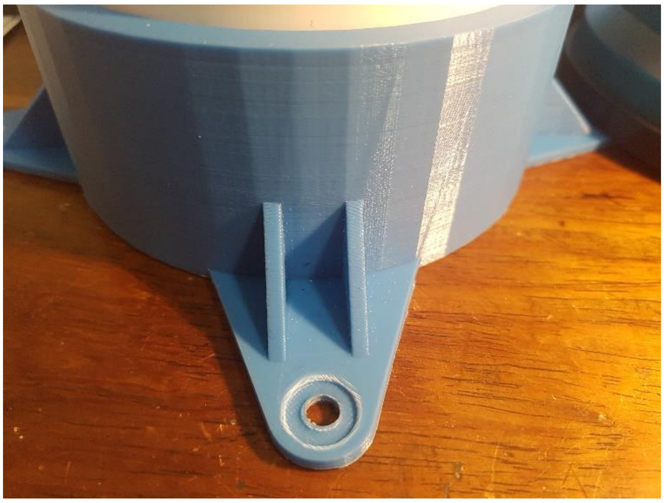

Figure 70: Crushing of the bracket tabs due to compression of the fasteners

The chain harness worked well and centred the load of the mooring system to keep the buoy stable in the water. However, the chain was not made of a marine-grade alloy and certainly not one included in the MINIONS Specification. As a result, it lost its shine and showed surface rust after only 19 hours at sea. The Type 316 stainless steel fittings and aluminium brackets, on the other hand, performed well and showed no signs of corrosion.

A true test of the system would require prolonged deployment at sea. This would, of course, require energy-harvesting devices, but would truly test the mooring system, waterproofness of the enclosures, and the suitability of the materials for use at sea.

6. Conclusions

6.1 Range Test

It can be concluded that signal quality is dependent on both distance and degree of obstruction. However, degree of obstruction seems to be the limiting factor, as a high degree of obstruction results in poor signal strength even at small distances, while a low degree of obstruction can result in good signal strength even at great distances.

Points D and E, and G and H in Figure 49 had similar degrees of obstruction between each pair as illustrated in Figure 68, but each pair of points was located 500m apart. The decrease in RSSI and SNR between each of these pairs illustrates a negative, somewhat linear relationship between distance and signal quality. One hypothesis is that if degree of obstruction is kept constant, a negative, linear relationship between distance and signal quality would exist within the device's rated range. Another hypothesis is that a similar relationship would exist between degree of obstruction and signal quality if distance was kept constant.

The results show that the REYAX RYLR896 LoRa modules are able to function at least 2.5km beyond the range of the planned deployment and suggest that they would be able to function well beyond the range reached in the experiment. However, they also show that degree of obstruction is a significant factor which must be considered when planning the sensor network coverage. At this point in time, no data had been gathered at the planned deployment site, so the signal quality could not yet be determined.

6.2 Deployment Test

The temperature data differed significantly in value from verified data recorded in the same period, although trends in the data seemed to correlate with those of the verified data. However, trends in the temperature data often led or lagged those seen in the verified data. The turbidity readings were closely correlated with the amount of sunlight falling on the sensor. Any trends in turbidity were heavily obscured by deviations in the readings due to sunlight, so no information regarding water clarity could be extracted.

Both sensors displayed a high degree of precision, but a low degree of accuracy. Neither sensor produced accurate and reliable data. However, this was not the purpose of the study, so the successful transmission and reception 4 datapoints demonstrated the capabilities of the system as a whole, and proved that the system worked as intended, meeting most of the requirements.

247 out of 258 packets of data were successfully received from the Transmitter Node, with an error rate of 4.26%. However, this error rate only accounts were messages that were received and known to be erroneous. It does not account for messages that were transmitted and not received at all, if there were any. The cause of the error was not established, and its presence could not be predicted.

In hindsight, excluding the Packet Number variable from the recorded variables was a massive oversight. The data was already available at the Receiver Node but was not published to Node-RED or stored. This would have greatly enhanced data analysis, would provide a more accurate error rate, may have helped determine the cause of the error, and would have provided concrete proof of the system's restart functionality.

The user interface successfully displayed all of the received data. It was able to indicate the state of the Transmitter Node, the time of the last successful transmission, and successfully notified the user when the battery was depleted. Crucially, it enabled remote control, as the user was able to change the Transmitter Node's sampling period and force a restart remotely. Unfortunately, it was not possible to view the user interface outside the local network, so the data was not findable and accessible as per the FAIR data principles outlined in the literature review.

The system successfully stored six different datasets in CSV format, which were successfully used for post-deployment data analysis. Each node was identified with a persistent identifier (Network ID and Address), and each sample was identified with a topic and Unix timestamp. The successful storage of data in conjunction with persistent identifiers greatly enhanced data analysis and would aid in the interoperability and reusability of data according to the FAIR data principles outlined in the literature review.

Power optimisation software was implemented at the Transmitter Node, which prolonged battery life. As a result, the battery still had approximately 60% capacity when the system was retrieved. The sharing of the battery percentage to the MINIONS dashboard was a power optimisation feature in its own right If the user observed the battery capacity declining, they could increase the sample period, which would significantly prolong battery life and extend the deployment.

The signal strength was consistent and weak throughout the deployment. However, LoRa is designed to operate under such conditions, which were well within the ratings of the modules used. The stability of the connection was illustrated by the low standard deviation in RSSI readings. While the SNR had a much greater standard deviation, it was still within the ratings of the device. However, the system was not designed to be fault tolerant. No software was included to ensure a secure connection during transmission or to store data onboard in the event of a poor connection. As a result, data transmitted during a poor connection would simply be lost. Again, recording Packet Number would have enabled further insight into this potential issue, but this was not done.

The buoy platform proved to be strong and stable, and weathered the rough sea conditions well. The floatation device, mounting platform and enclosures showed no signs of weathering, although the chain harness was slightly rusted.

The 3D printed bracket performed poorly, as it made it difficult to tighten the lid of the enclosure and shattered with very little force. It was printed in PLA, which meant that it did not comply with the MINIONS Specification and was poorly suited for use at sea. The inclusions of HDPE, Type 316 stainless steel and 5052 aluminium in the MINIONS Specification were justified as they fared significantly better in the Deployment test than the non-compliant materials used in the chain and enclosure bracket.

The mooring system performed well and proved to be a secure, reliable design. The waterproof enclosures proved to be sufficiently waterproof for the deployment test. Their long-term performance, however remains to be seen, as it is likely that the PVC fittings, as well as the cable glands and cables used to connect the enclosures, would likely fail with prolonged deployment at sea.

6.3 General Conclusions

The User Requirements were assessed to determine the extent to which they had been satisfied. Table 15 in Appendix A lists each of the user requirements and whether they had been Fully, Partially or Not Satisfied. Where a requirement was not fully satisfied, an explanation was provided in Table 16 in Appendix A.

In summary, 25 of the 29 User Requirements were fully satisfied, and only two were not satisfied. The system was successfully able to gather, transmit, display and store ocean data. This occurred from a node out at sea, to a node on land, and to a user interface, where it could be viewed on any device within the local network. The system was able to give regular updates, and a degree of remote control was achieved.

The success of the system validates the Specification, which will bring uniformity and interoperability to IoT marine monitoring networks. It has introduced standards based on known standards and industry best practices and will aid the sharing of designs, reuse of modules and cooperation between networks. This will ensure devices are suited for deployment at sea and will accelerate development and advances in the field.

7. Recommendations

Future work should include the development of a specification for underwater nodes. Future enclosures should not be made out of PVC. Investigations into the development of a 3D printed waterproof enclosure should continue, as this would provide a uniform and easily-replicable enclosure which designers could simply download and print. Developments in these regards could open up a host of opportunities for the Specification. When using ethyl acetate to smooth PETG prints, an electric heat source with fine temperature control should be used. When enclosures via cable, the cable should have a solid core to prevent compression of the cable, which could compromise the waterproof seal of a cable gland.

A Veroboard circuit should be designed and built for the Receiver Node, and an enclosure that is able to accommodate the OLED screen and LED lights should be 3D printed.

Future versions of the software should include storage of the Packet Number to aid data analysis, the identification of lost packets and confirmation of system functionality. Text extraction should include a way of identifying scrambled messages and notifying the user if scrambling occurs. Besides aiding data analysis, this could potentially aid in identifying the cause of the error and rectifying it. Functionality should be developed to enable the system to determine connection quality between transmitter and receiver before transmitting messages. If the connection is poor or cannot be established, the node should store samples until the connection can be established.

Timestamps were added to data once received in Node-RED. Instead, nodes should use an RTC to include a real-time timestamp at the transmitter, and this data should always be identified by that timestamp. In addition to the sample period, it should be possible to change the "transmission period" of the Transmitter Node. Transmission is one of the largest consumptions of power in the circuit, so if it is possible to store multiple samples and transmit them all at once, this could drastically reduce power consumption. This would, however, reduce the frequency of updates to the dashboard.

Revisions to the Specification should include detailed testing procedures to validate systems. Further adoption of the Specification will serve as a learning experience, so it should be expanded and refined accordingly.

8. List of References

[1] D. Miorandi, S. Sicari, F. De Pellegrini, and I. Chlamtac, "Ad Hoc Networks Internet of things : Vision , applications and research challenges," vol. 10, pp. 1497–1516, 2012.

[2] J. A. Stankovic and L. Fellow, "Research Directions for the Internet of Things," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 3–9, 2014.

[3] G. Xu, Y. Shi, X. Sun, and W. Shen, "Internet of things in marine environment monitoring: A review," *Sensors (Switzerland)*, vol. 19, no. 7, pp. 1–21, 2019.

[4] J. J. H. Buck *et al.*, "Ocean Data Product Integration Through Innovation-The Next Level of Data Interoperability," *Front. Mar. Sci.*, vol. 6, no. February, 2019.

[5] C. Albaladejo, F. Soto, R. Torres, P. Sánchez, and J. A. López, "A low-cost sensor buoy system for monitoring shallow marine environments," *Sensors (Switzerland)*, vol. 12, no. 7, pp. 9613–9634, 2012.

[6] Alliance for Internet Of Things Innovation, "Identifiers in Internet of Things (IoT)," no. February 2018, pp. 1–34.

[7] M. D. Wilkinson *et al.*, "Comment: The FAIR Guiding Principles for scientific data management and stewardship," *Sci. Data*, vol. 3, pp. 1–9, 2016.

[8] P. A. Beddows and E. K. Mallon, "Cave pearl data logger: A flexible arduino-based logging platform for long-term monitoring in harsh environments," *Sensors (Switzerland)*, vol. 18, no. 2, 2018.

[9] M. Carbonell, I. Massana, J. Prat, E. Trullols, and J. Del Rio, "Simulations and design of a sea moored buoy with a real time data undersea seismometer," in *OCEANS 2017 - Aberdeen*, 2017.

[10] M. Li and B. Yang, "A Survey on Topology issues in Wireless Sensor Network."

[11] A. Flammini, P. Ferrari, D. Marioli, E. Sisinni, and A. Taroni, "Wired and wireless sensor networks for industrial applications," *Microelectronics J.*, vol. 40, no. 9, pp. 1322–1336, 2009.

[12] G. Anastasi, M. Conti, M. Di, and A. Passarella, "Ad Hoc Networks Energy conservation in wireless sensor networks : A survey," *Ad Hoc Networks*, vol. 7, no. 3, pp. 537–568, 2009.

[13] M. C. Id and A. F. S. Id, "Performance Evaluation of LoRa Considering Scenario Conditions," no. i, 2018.

[14] X. Yang *et al.*, "Design of a Wireless Sensor Network for Long-term," pp. 455–472, 2002.

[15] Semtech, "What is LoRa?," 2019. [Online]. Available: https://www.semtech.com/lora/what-is-lora. [Accessed: 02-Oct-2019].

[16] J. Rabault, G. Sutherland, O. Gundersen, and A. Jensen, "An Open Source, Versatile, Affordable Waves in Ice Instrument for Remote Sensing in the Polar Regions," pp. 1–18, 2019.

[17] Espressif Systems, "ESP32 Series Datasheet," *Espr. Syst.*, pp. 1–61, 2019.

[18] United States Coast Guard Ocean Engineering Division, "Specification for Fabrication of Steel Ocean Buoys," pp. 1–39, 2013.

[19] A. Meindl, "Guide to moored buoys and other ocean data acquisition systems," *DBCP Tech. Doc.*, vol. 8, p. 40, 1996.

[20] Lloyd's Register, "Rules for the Manufacture, Testing and Certification of Materials," *Lloyd's Regist. Rules Regul. 2015 - Version 9.24*, no. July, 2015.

[21] United States Department of Defence, "MIL-DTL-38999M Detail Specification: Connectors, Electrical, Circular, Miniature, High Density, Quick Disconnect (Bayonet, Threaded or Breach Coupling), Environment Resistant with Crimp Removable Contacts or Hermetically Sealed with Fixed, Solderable Cont," 2017.

[22] Sealite Pty Ltd, "Benefits of Polyethylene Buoys in Harsh Conditions," 2015.

[23] B. T. Champion, M. Jamshidi, and M. A. Joordens, "3D printed underwater housing," *World Autom. Congr. Proc.*, vol. 2016-Octob, pp. 1–6, 2016.

[24] G. Maniatis, T. Hoey, and J. Sventek, "A New Method for Rapid Prototyping of Purpose-Specific Sensor Enclosures: Example Application and Implications for Data Coherence," *J. Sens. Actuator Networks*, vol. 2, no. 4, pp. 761–779, 2013.

[25] Y. Sun, S. Huang, S. Ma, and Q. Quan, "Development of a waterproof servo unit for amphibious robots," *2015 IEEE Int. Conf. Cyber Technol. Autom. Control Intell. Syst. IEEE-CYBER 2015*, pp. 923–928, 2015.

[26] Simplify3D, "Ultimate Materials Guide - Tips for 3D Printing with PETG," 2019. [Online]. Available: https://www.simplify3d.com/support/materials-guide/petg/. [Accessed: 16-Jul-2019].
[27] rigid.ink Ltd., "PETG Filament - Overview, Step-by-Step Settings & Problems Resolved." [Online]. Available: https://rigid.ink/blogs/news/175700615-petg-filament-heres-what-you-need-to-know. [Accessed: 16-Jul-2019].
[28] MatterHackers, "How To: Post-Processing PETG 3D Filament." [Online]. Available: https://www.matterhackers.com/articles/how-to-post-processing-petg-3d-filament. [Accessed: 03-Oct-2019].
[29] rigid.ink Ltd., "PETG Technical Data Sheet," 2019.
[30] rigid.ink Ltd., "ABS Technical Data Sheet," 2019.
[31] Curbell Plastics, "Curbell Plastics - Plastic Properties Table." [Online]. Available: https://www.curbellplastics.com/Research-Solutions/Plastic-Properties. [Accessed: 18-Jul-2019].
[32] M. S. I. . Inc., "Metal Reference Guide," 2017.
[33] M. S. I. . Inc., "7 Things to Consider When Choosing an Aluminum Grade," 2019. [Online]. Available: https://www.metalsupermarkets.com/7-things-consider-choosing-aluminum-grade/. [Accessed: 16-Jul-2019].
[34] AC Manufacturing, "5052 Aluminum : CNC precision machining." [Online]. Available: http://www.acmanufacturing.com/metal_06.htm. [Accessed: 16-Jul-2019].
[35] J. Bouwmeester, M. Langer, and E. Gill, "Survey on the implementation and reliability of CubeSat electrical bus interfaces," *CEAS Sp. J.*, vol. 9, no. 2, pp. 163–173, 2017.
[36] S. van der Linden, J. Bouwmeester, and A. Povolac, "Design and Validation of an Innovative Data Bus Architecture for CubeSats," *14th Reinventing Sp. Conf.*, 2016.
[37] DSMT, "IP Rating Chart," 2019. [Online]. Available: http://www.dsmt.com/resources/ip-rating-chart/. [Accessed: 25-Jun-2019].
[38] C. Brown, "Nothing Gets In: Waterproof Enclosure Design 101 (and IP68)," *Fictiv*, 2016. [Online]. Available: https://www.fictiv.com/blog/posts/nothing-gets-in-waterproof-enclosure-design-101-and-ip68. [Accessed: 27-Jun-2019].
[39] A. Mehrparvar, "CubeSat Design Specification," *CubeSat Program, CalPoly SLO*, Feb. 2014.
[40] "49 CFR Part 175 - CARRIAGE BY AIRCRAFT | CFR | US Law | LII / Legal Information Institute." [Online]. Available: https://www.law.cornell.edu/cfr/text/49/part-175. [Accessed: 12-Oct-2019].
[41] DF Robot, "Gravity: Analog Turbidity Sensor For Arduino." [Online]. Available: https://www.dfrobot.com/product-1394.html. [Accessed: 07-Oct-2019].
[42] DF Robot, "Waterproof DS18B20 Digital Temperature Sensor for Arduino." [Online]. Available: https://www.dfrobot.com/product-689.html.
[43] "Announcing the ADVANCED ENCRYPTION STANDARD (AES)," *Fed. Inf. Process. Stand. Publ. 197*, Nov. 2001.
[44] Reyax Technology Corporation Ltd., "RYLR896 UART Interface 868/915MHz LoRa Antenna Transceiver Module Datasheet," 2018.
[45] Core Electronics, "Espressif ESP32 Development Board - Developer Edition." [Online]. Available: https://core-electronics.com.au/espressif-esp32-development-board-developer-edition.html. [Accessed: 07-Oct-2019].
[46] S. S. C. Ltd., "Specification of Product for Lithium-ion Rechargeable Cell," 2009.
[47] "ICR18650-26F | Samsung 3.7V 18650 Lithium-ion Battery, 2600mAh | RS Components." [Online]. Available: https://za.rs-online.com/web/p/speciality-size-rechargeable-batteries/7887261/. [Accessed: 13-Oct-2019].
[48] "BSK LITHIUM CHARGR BOARD 5V 1A - Communica [Part No: BSK LITHIUM CHARGR BOARD 5V 1A]." [Online]. Available: https://www.communica.co.za/products/bsk-lithium-chargr-board-5v-1a. [Accessed: 13-Oct-2019].
[49] M. T. Inc., "MCP1700 Low Quiescent Current LDO."
[50] "ISO/IEC 20922:2016 Information technology -- Message Queuing Telemetry Transport (MQTT) v3.1.1," *iso.org*, Jun. 2016.
[51] "ISO - ISO 20653:2013 - Road vehicles — Degrees of protection (IP code) — Protection of electrical equipment against foreign objects, water and access." [Online]. Available:

https://www.iso.org/standard/58048.html. [Accessed: 12-Oct-2019].
[52] "International Convention for the Safety of Life at Sea (SOLAS), 1974." [Online]. Available: http://www.imo.org/en/About/Conventions/ListOfConventions/Pages/International-Convention-for-the-Safety-of-Life-at-Sea-(SOLAS),-1974.aspx. [Accessed: 12-Oct-2019].
[53] Google, "Hout Bay." [Online]. Available: https://goo.gl/maps/2pU8JFtt3NyEDXCKA. [Accessed: 13-Oct-2019].
[54] "Windguru - Cape Town." [Online]. Available: https://www.windguru.cz/91. [Accessed: 13-Oct-2019].
[55] "WaveNet: The online real time waves and weather for South Africa: Cape Town Waves." [Online]. Available: http://wavenet.csir.co.za/OnlineData/CapeTown/CapeTownwaveD.htm. [Accessed: 13-Oct-2019].
[56] "Cape Town Surf Report, Surf Forecast and Live Surf Webcams." [Online]. Available: https://magicseaweed.com/Cape-Town-Surf-Report/81/Historic/. [Accessed: 13-Oct-2019].
[57] "Matroosfontein, south africa History | Weather Underground." [Online]. Available: https://www.wunderground.com/history/daily/za/matroosfontein/FACT/date/2019-10-7. [Accessed: 13-Oct-2019].
[58] "Mobilefish.com - LoRa/LoRaWAN tutorial." [Online]. Available: https://www.mobilefish.com/developer/lorawan/lorawan_quickguide_tutorial.html. [Accessed: 13-Oct-2019].

9. Appendices

9.1 Appendix A

Validation of the User Requirements.

Requirement Number	User Requirement	Satisfaction of Requirement
GR1	Build and develop a marine monitoring device according to the MINIONS specification.	Partially Satisfied
GR2	The system must be flexible to allow for multiple configurations of sensors and communication technologies to be used.	Fully Satisfied
GR3	The system should be scalable to allow for different network topologies and to cover different areas.	Fully Satisfied
GR4	The system's environmental impact should be kept to a minimum	Fully Satisfied
GR5	The system must not obstruct sea traffic, and proper approval must be acquired.	Fully Satisfied
GR6	The network coverage and an appropriate network topology must be carefully determined.	Fully Satisfied
MR1	The system must be durable and last beyond the length of the planned deployment.	Fully Satisfied
MR2	The system must be waterproof.	Fully Satisfied
MR3	The system must be stable to aid wireless communications.	Fully Satisfied
MR4	The design should facilitate easy access for easy maintenance, modification and dismantling.	Fully Satisfied
MR5	The system must be designed to protect against acts of vandalism.	Not Satisfied
MR6	The system must be reliable to reduce the need for maintenance.	Fully Satisfied
MR7	All components, electrical and mechanical, must be corrosion-resistant or protected against the effects of corrosion.	Fully Satisfied
MR8	Connectors must be kept to a minimum.	Fully Satisfied
SR1	The system must give accurate updates of the device's state.	Fully Satisfied
SR2	A degree of remote-control must be achieved.	Fully Satisfied
SR3	It must be possible to change certain parameters such as sample period remotely.	Fully Satisfied
SR4	The system should implement software for optimal power management.	Fully Satisfied
SR5	The system must log and store all data for analysis.	Partially Satisfied
SR6	The system must upload data to a user interface for public viewing.	Fully Satisfied
SR7	Persistent identifiers must be used for devices as well as data samples.	Fully Satisfied
SR8	The system must be fault tolerant and allow sampling to continue in the event of a communication failure.	Not Satisfied
ER1	The system must sense one physical parameter and one chemical	Fully Satisfied

	parameter.	
ER2	The system should log samples and transmit the data and other important information to a receiver on land.	Fully Satisfied
ER3	All electronics should be well-insulated.	Fully Satisfied
ER4	The system should have some level of energy autonomy.	Fully Satisfied
ER5	Electronics should be optimised for low energy consumption and optimal power use.	Fully Satisfied
ER6	A stable power supply must be provided	Fully Satisfied
ER7	Protections must be in place to prevent over-charge and over-discharge of the battery.	Fully Satisfied

Table 15: List of User Requirements and the extent to which they were satisfied

Requirement Number	Level of Satisfaction	Explanation
GR1	Partially Satisfied	Digressions from the specification occurred with the use of PLA and PVC plastic and non-marine grade chain. All other requirements were met.
MR5	Not Satisfied	It was intended to paint the buoy to blend into its environment, but this was not done. Instead, the white and orange paint made the buoy highly visible.
SR5	Partially Satisfied	All data except Packet Number was stored.
SR8	Not Satisfied	No measures were taken to ensure connection before transmission, and no samples were stored in the event of a loss of connection.

Table 16: Explanations for User Requirements which were not fully satisfied

9.2 Appendix B

Node-RED flow.

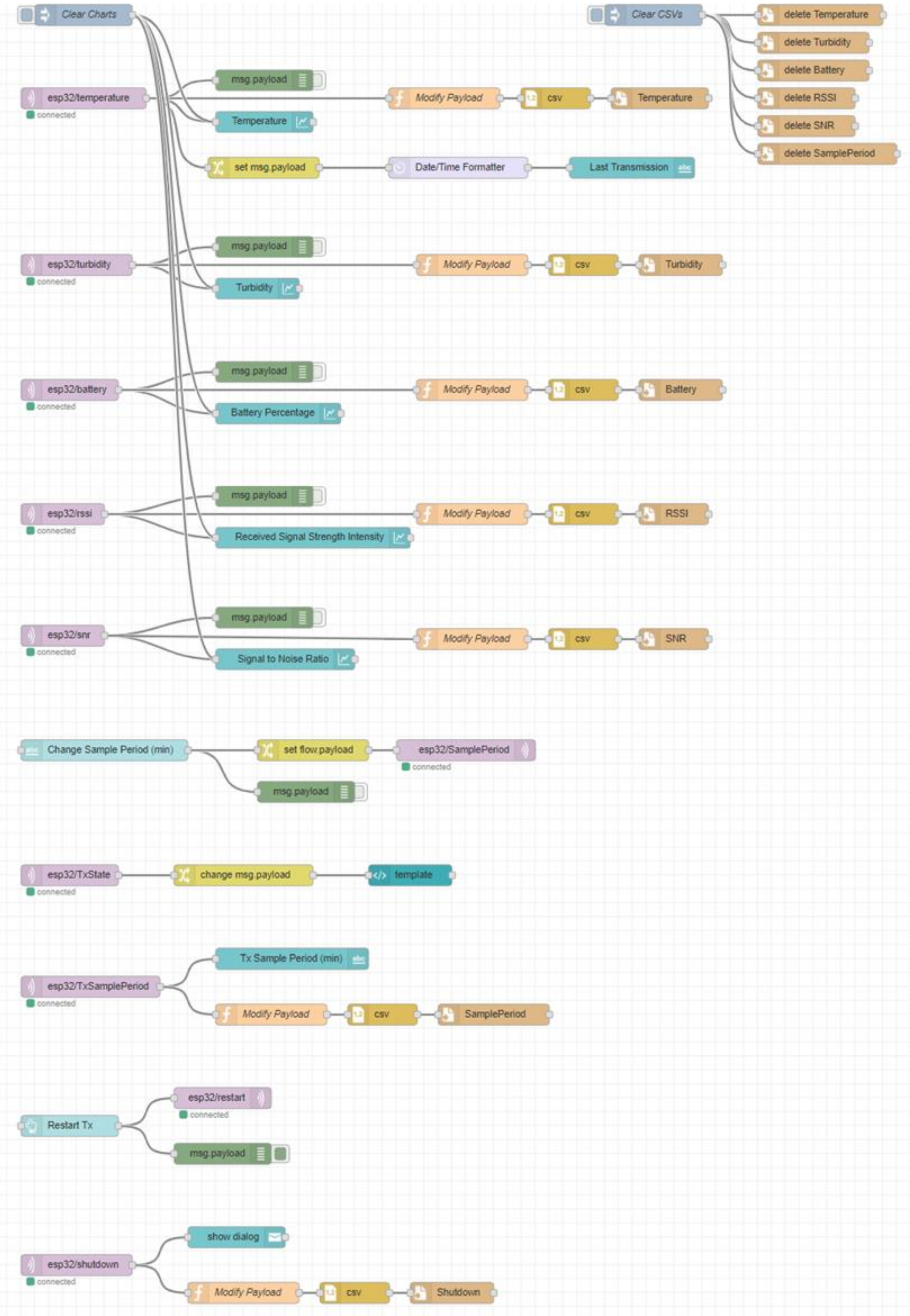

Figure 71: Node-RED flow used to create MINIONS Dashboard

9.3 Appendix C

https://github.com/LukeMet/MINIONS.git

The Git repository linked above contains the following:

- The MINIONS Specification
- Arduino code for both the Transmitter and Receiver Nodes
- Code for the Node-RED flow shown in Appendix B

9.4 Appendix D

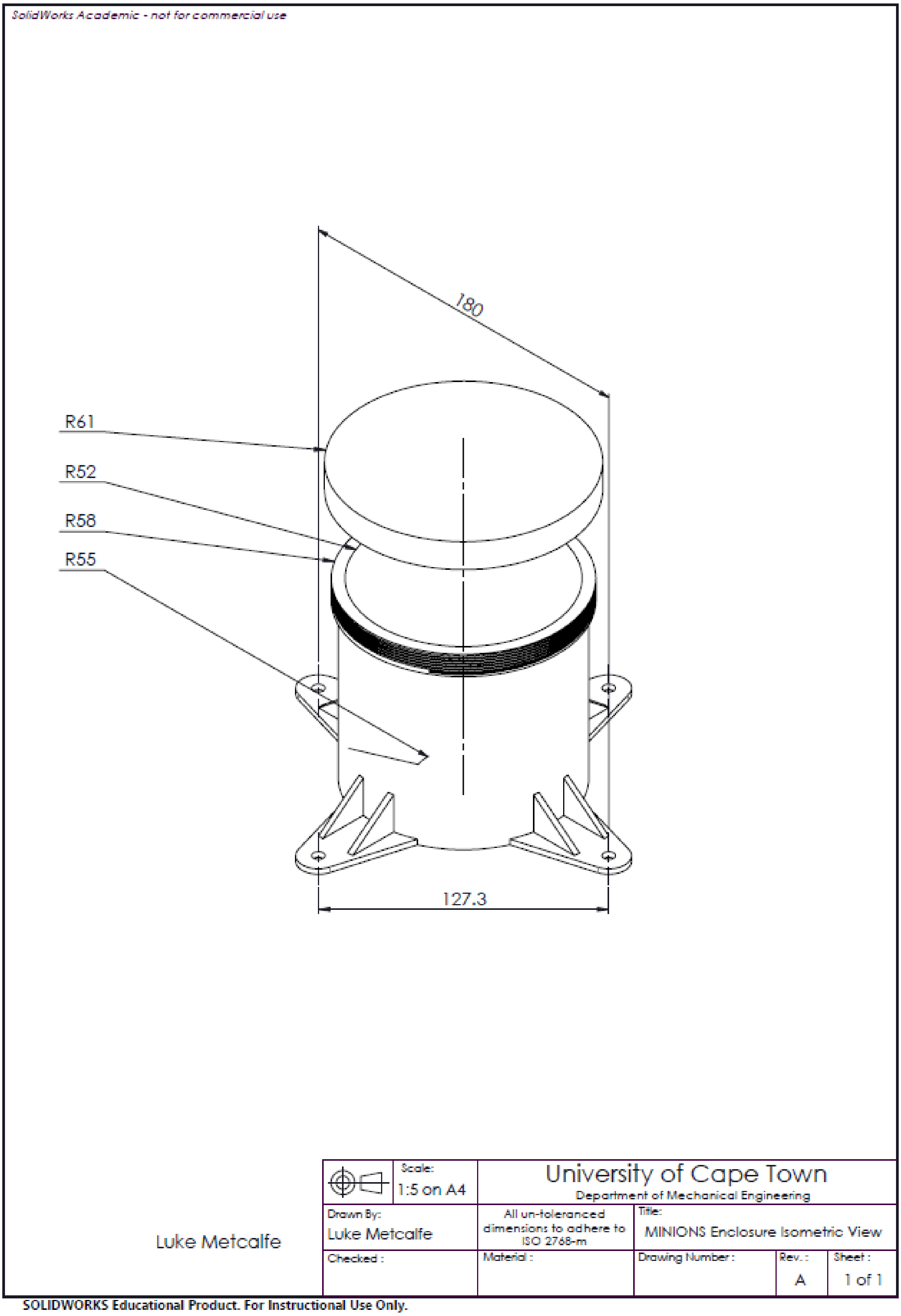

Figure 72: Isometric view of the MINIONS enclosure

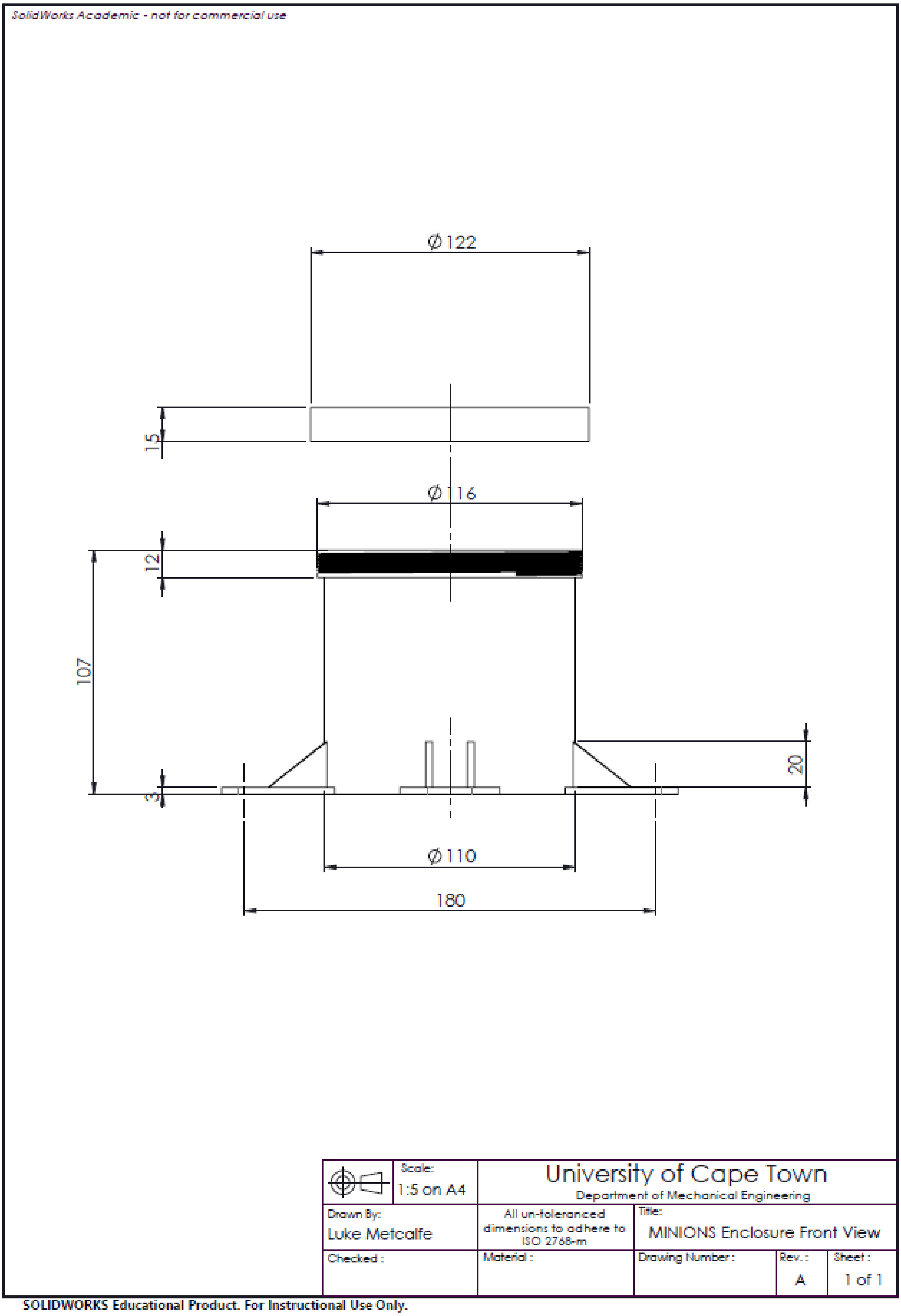

Figure 73: Front view of the MINIONS enclosure

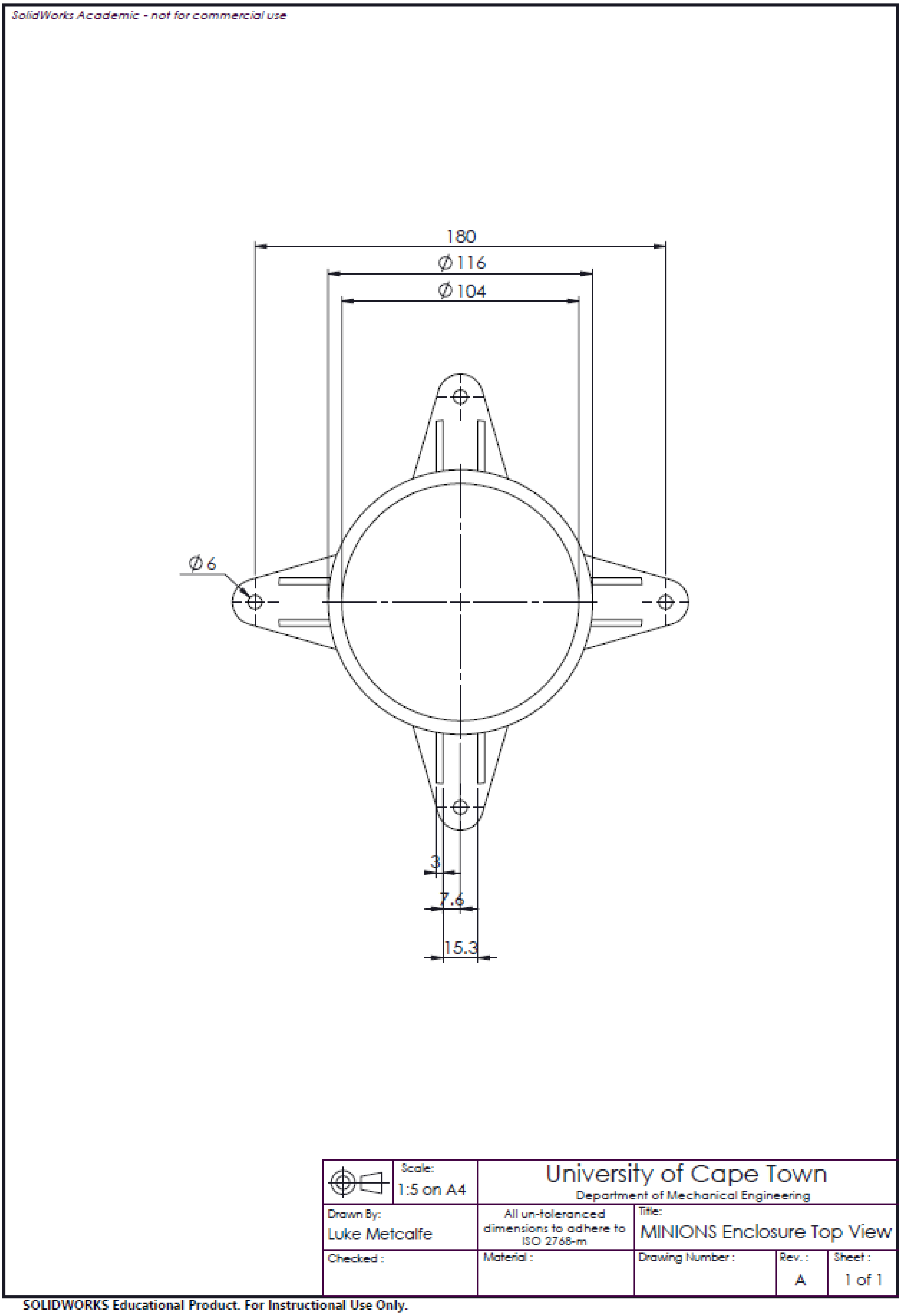

Figure 74: Top view of the MINIONS enclosure

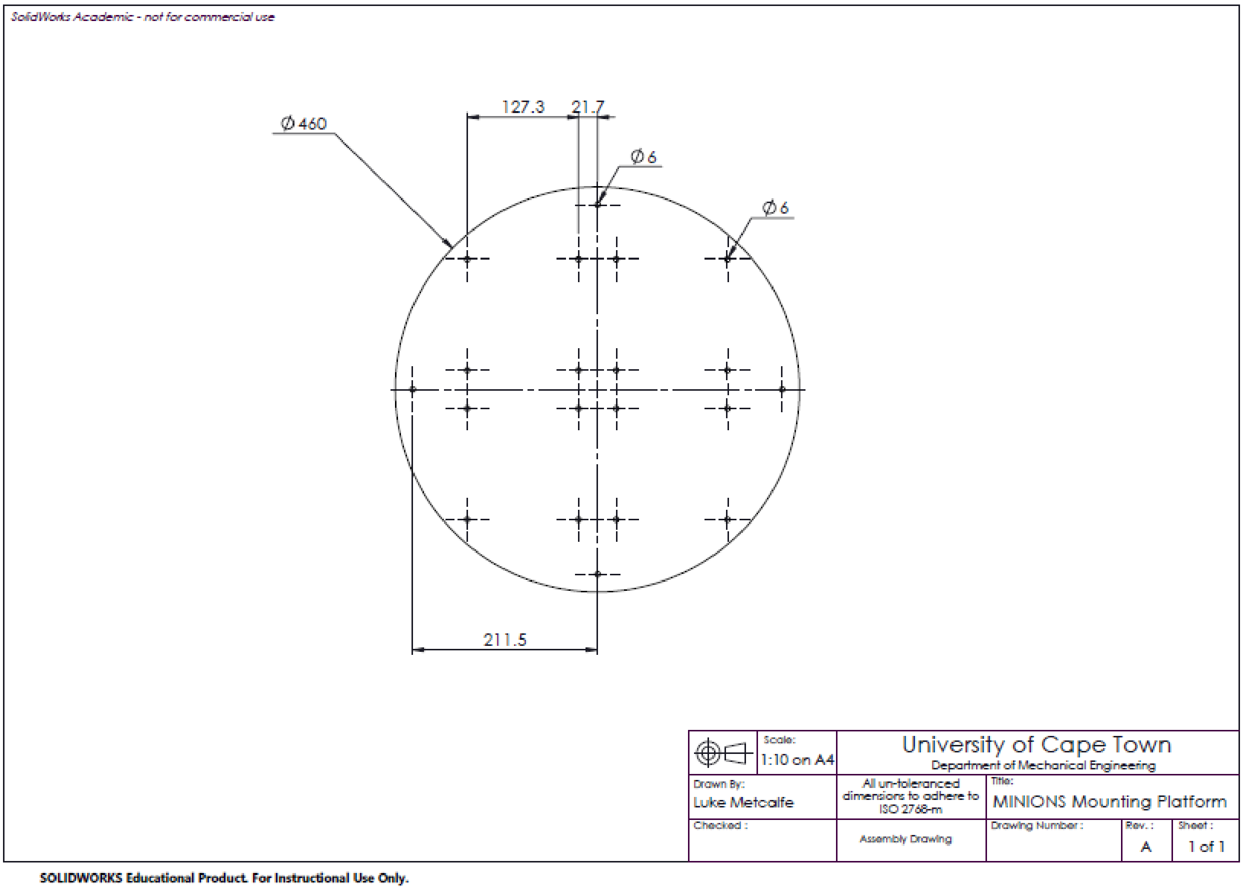

Figure 75: Top view of the MINIONS mounting platform

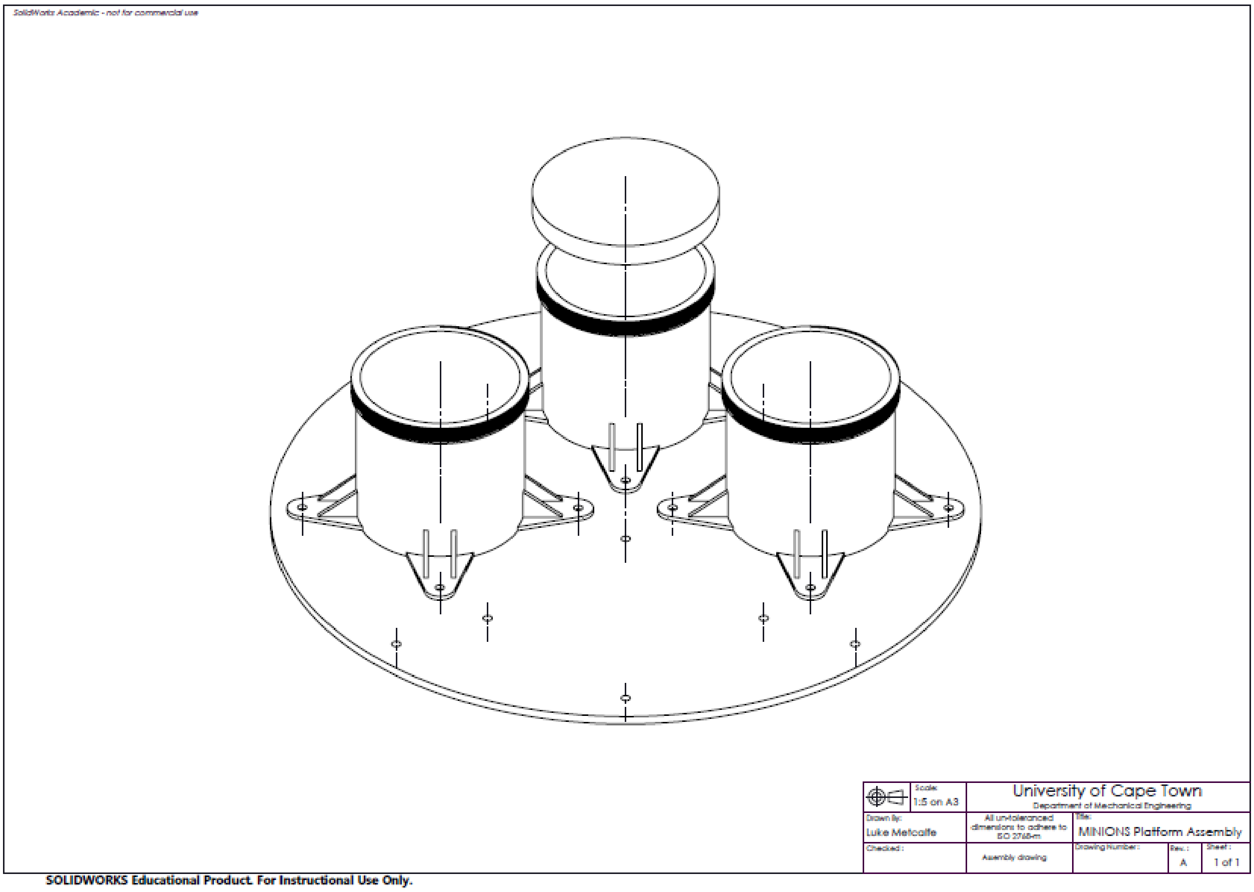

Figure 76: Isometric view of the assembly of the MINIONS mountint platform and three MINIONS enclosures