Evaluating the Operational Efficacy of LoRaWAN Class C ESP32 Animal Tag Devices During Firmware Updates Over-the-Air

Benjamin McKitterick

MSci Hons Computer Science w/ Industrial Experience

June 2021

I certify that the material contained in this dissertation is my own work and does not contain unreferenced or unacknowledged material. I also warrant that the above statement applies to the implementation of the project and all associated documentation. Regarding the electronically submitted work, I consent to this being stored electronically and copied for assessment purposes, including the School's use of plagiarism detection systems in order to check the integrity of assessed work. I agree to my dissertation being placed in the public domain, with my name explicitly included as the author of the work.

Name: Benjamin McKitterick

Date: 04/06/2021

Abstract

Upland hill harming is a key pastoral farming practice integral to the prosperity of the Lake District World Heritage Site. An ESP32 based animal tag device developed by Milliamp Technologies aims to help maintain the occupancy, which in turn sustains the cultural landscape. This paper sets out to evaluate the efficacy of the LoRaWAN Class C model operational mode during a FUOTA procedure while the animal tag device is in operation. The study found that the device, when tested within hilly outdoor terrain, was significantly prone to interference from obstacles such as vegetation and water; especially when transmitting using lower data rates. Interviews with local farmers were conducted to inquire into general agricultural routine to better understand the contextual needs of the animal tag for the FUOTA procedure. After a collective analysis, we find possible farming procedures that complement the same timing requirements for firmware updates, while reducing distance and power use.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Milliamp Technologies Ltd is an independent electronics contract design and manufacturing enterprise centred around innovative cutting-edge technology. In order to provide a sustainable technology that eases and begins to autonomize agriculture, Milliamp has ideated project "MTL Animal Tag" to assist in livestock monitoring. Farmers will have access to a platform that allows them to swiftly determine animal health status, thereby permitting prioritised care to animals identified as sick. The project seeks to fill a growing digital urban-rural divide in the UK [1] whilst driving a competitive rural economic growth that will increase standards of living.

Firmware updates over-the-air (FUOTA) are an essential component for large-scale deployment of such a project yet constitute a great portion of a device's energy expenditure; the backbone of the company's project lies within the longevity of the device. Working in partnership with Milliamp, this paper aims to explore the manner in which remote firmware updates are applied on constrained ESP32 animal tag devices; henceforth, identified as 'tags'. The tags require both long range communication and low power consumption for deployment to be operationally feasible and marketable. To facilitate such prerequisites, a state-of-the-art low-power wide-area network (LPWAN) technology known as LoRaWAN has been selected by the company. Therefore, an evaluation of a LoRaWAN class C [2] model implementation during administration of firmware updates would highlight model drawbacks and allow for refinement of the projects FUOTA process within its applicational context.

1.1 Lake District Hill Farming

The significance of pastoral farming to the natural prosperity of the Lake District National Park is unquestionable. As noted by D. Harvey et al. [3, p. 1] "Protection of this candidate World Heritage site depends on the continuance of hill farming in the lake district". The report finds that the majority of the study sample farms had a capital return of less than 0% from tenant investment, calling attention to the financial unsustainability of the practice. This project could provide a new cost-effective technological solution to augment traditional hill farming. Adoption of this product could ameliorate the agricultural practice and aid farmers, improving efficiency and rural development, maximising production, and minimising costs. For instance, collecting data using advanced AI and ML algorithms to predict deviations and abnormalities would allow farmers to swiftly identify, predict and even prevent disease outbreaks. A past outbreak of foot-and-mouth disease in the UK in 2001 caused a tremendous calamity where over six million sheep and cows had to be slaughtered to halt the illnesses spread [4]. Additionally, this would indirectly support the diverse ecosystem of surrounding flora and fauna, limit soil erosion, and protect significant upland habitats from declining. Ensuring the beloved site is protected and its 16 million public visitors continue to reap physical and mental benefits from its natural wonders, as well as maintaining the site's wealth of history and culture.

1.2 Project Aims

This research project aims to discover if the LoRaWAN Class C device configuration proposed by Milliamp is suitable for their project framework while additionally providing an insightful analysis for operators to develop more intelligent LoRaWAN applications. With this considered, the following research question is intended to guide this study:

"Based on power consumption, relative performance, and update time, does the LoRaWAN Class C model perform adequately enough for its required applicational needs?"

A partially developed LoRaWAN FUOTA infrastructure will act as a foundation for this research in order to focus more on the development of the FUOTA tag mechanisms. As a result, the research question can be divided into six main aims:

- Research and configure any remaining LoRaWAN infrastructure required to implement the FUOTA process for the tag.
- Design and implement the FUOTA multicast deployment mechanisms.
- Conduct an assessment and investigation of the tag, measuring power consumption, and update time metrics.
- Analyse network performance by measuring RF strength during a FUOTA operation to the tag in a natural outdoor environment.
- Evaluate the tag's performance, update efficiency, and power consumption to determine if the model can effectively operate for its application context.
- Discuss the results and deliberate over the possibility of a LoRaWAN Class B model use case.

1.3 Chapter Overview

Firstly, the background chapter of this report will cover the constraints, limitations, and challenges faced during the work placement for Milliamp. It will then move onto the historical difficulties of firmware update processes and then analyse existing systems' assets and liabilities. The design chapter will detail plans for the system's technological infrastructure and architecture. Ideas for message flow between network and tag are then proposed, alongside arranging a schematic for measuring the power consumption of said devices. The implementation section covers the main algorithms, data structures, and procedures used to develop a 'ping-pong' testing tool and the aforementioned power measuring device. Next, a system in operation chapter goes over the various system components and functions in a pipeline fashion to provide the reader with a greater depth of overall understanding. This is followed by the testing and evaluation section that details 'on-the-bench' system tests with a proceeding field study. The data is then analysed and evaluated, alongside a review of the study itself. Finally, an overview of the project and its potential future development is discussed in a concluding chapter.

Chapter 2

Background

As we usher into an era of ubiquitous and pervasive computing, the 'internet of things' (IoT) applications are fuelling a burgeoning demand for more efficient, low-cost solutions to device energy consumption [5]. Recent advancements in long-range energy-efficient networking designed to wirelessly connect IoT devices can be encapsulated by the term low-power wide-area networks (LPWANs). According to reports, LPWAN market size is expected to grow at a compound annual growth rate of over 60% between 2021 and 2027 [6]. Examples of the leading technologies in this market include LoRaWAN, SigFox, and NB-IoT—with the LoRaWAN protocol recently receiving much academic and industrial attention [7, 8, 9]. FUOTA are one of the latest renovations to the LoRaWAN specification [2] and a required technological mechanic needed to update fleets of deployed products. Over-the-air (OTA) firmware updates are a critical component of an IoT system and past mistakes, such as the Mirai-botnet Distributed Denial of Service (DDoS) attacks [10], have taught us the damaging effects of having out-of-date firmware—discussed further in section 2.2.1. At the same time, it is important to investigate the most appropriate methods to perform these updates and to recognise the successes and constraints of existing systems before any implementation for this project can be designed.

2.1 Project MTL Animal Tag

Prior to this research, a ten-week placement was undertaken with Milliamp in which a proof-of-concept investigation was completed to determine the possibilities of the novel MTL Animal Tag project being feasible. Due to the project's large scope Milliamp forked the objectives into two main polarised branches—artificial intelligence and IoT networking. The later package remains the focus of this paper and is rooted mainly in the ESP32 series microcontroller, LoRaWAN technology, and FUOTA. The package was divided into the following three deliverables:

- D1 – Transmitting a payload to and from the animal tag.
- D2 – Designing and developing the device firmware update mechanisms.
- D3 – Configuring an architecture that allows 'updating on the fly'.

All deliverables were accomplished, and an architectural foundation was designed and implemented for the company. As concepts were successfully proven, it was clear that the concept had potential and that further research and expansion were required. As such, this research intends to build itself of the existing technological infrastructure to move forwards and produce a functional and refined FUOTA process.

2.1.1 TTN & Infrastructural Challenges

A significant barrier emerged during the early days of the project's development. The companies preferred decentralised IoT network provider was 'The Things Network' (TTN) and it was instructed that applications should be developed using this particular technology. However, after further research, a major complication was found—the TTN fair access policy [11]. The policy limits the data that each end-device can send and so only ten downlink payloads per 24 hours could be transmitted over this network service. For FUOTA such a limitation is unfeasible and so it was acknowledged that a private LoRaWAN network would have to be designed and implemented to proceed any further.

Later on in the project, the same oversight had significant repercussions. A LoRaWAN concentrator or gateway is the intermediary that allows the sensor devices to transmit and receive data from the LoRaWAN server stack [12]. Milliamp provided a gateway model known as 'The Things Indoor Gateway' (TTIG), which, unfortunately, was hardcoded to work only with TTN and could not be reconfigured with the private LoRa network servers being developed. After discussion with Milliamp, it was decided that efforts would be best redirected, for the meantime, towards the application top-end rather than the network back-end work.

2.1.2 Resource-Constrained Tags

Throughout the placement, many resource constraints had to be carefully taken into consideration while developing software. In one instance, while developing the firmware patch tool critical consideration of memory management was paramount. The patch software required $n+m+O(1)$ bytes of memory, where n is the size of the old file and m the size of the new file. With only 520KB of internal SRAM available on the tag and a machine learning model requiring a lot of space, stack overflows were a common occurrence. However, alterations to the machine learning model size eventually provided enough space to load both the old and new patch file into main memory and perform binary diffing to apply the firmware patches.

Another persistently constraining factor throughout the project's development was device energy use. The problematic trade-off between power consumption and performance always had to be regarded. For instance, when writing the firmware update mechanisms for the ESP32 architecture, reading an application image onto the task stack using a write buffer of equal size to the ESP32 application images padded boundary operates but isn't efficient. Instead, dynamically allocating memory to the write buffer based on the ESP32 application images metadata increases algorithmic complexity yet lowers read and writes—reducing energy usage.

2.2 Related Work

As briefly touched at the beginning of this chapter, firmware updates are essential throughout a device's field deployment for a multitude of reasons, including performance enhancement, deployment of bug fixes, and new security feature implementations [13, 14]. Nevertheless, past events have already shown the aftermath that can be caused by incorrectly configured firmware update systems or by the inexistence of such systems [10, 15]. Thus, in order to obtain a crucial viewpoint for understanding, analysing broadly related systems and events from the past may ameliorate one's perspective and reveal patterns that might otherwise be invisible in the present.

2.2.1 Historical Firmware Update Failures

History has highlighted that minor events can trigger system outages during firmware updates, such as the 1990 AT&T collapse [16] where technicians upgraded switch software to speed processing of certain message types. A single one-line bug was inadvertently propagated throughout the network which caused a cascading reset that incapacitated the entire system. Over 50 million calls were blocked over the nine hours it took to reconfigure the network. As P. Neumann stated [15, p. 64–99] “such risks grow with both the complexity of interconnected systems and with the attempts to optimise performance of the whole by increasing the coupling of the part”. Accordingly, concern should be given towards continuing system operability during the presence of reliability collapses and security attacks. A more sturdy, fault-tolerant software system that could tackle such bugs without resetting would have lessened impacts significantly [16]. Moreover, as the software was coded and compiled in C rather than using a more structured language with debugging exceptions and a stricter compiler like C++, the errors remained undetected. Such a notable and rare event provides much food for thought.

In October 2016, cybercriminals discovered a way to administer one of the largest distributed denial-of-service (DDoS) attacks in history. Mirai—meaning “the future” in Japanese—and its various forms took advantage of insecure IoT devices and managed to amass a conglomerate of bots as foot soldiers. The Mirai botnet DDoS attack targeted the French web host and cloud service provider OVH [17] peaking at 1.1 TBps of traffic. Kolias et al. [10] deduced that the most prominent factor for the success of the attack was that default credentials were left unchanged and that provided firmware updates were inadequate. S. Nappo [18] wisely identifies that “The Internet of Things devoid of comprehensive security management is tantamount to the Internet of Threats”, elucidating the value of having more reliable and secure firmware update mechanisms for IoT devices. Reflecting on the above, an IoT deployments architectural configuration needs to be carefully considered during design and development. Some have suggested the use of hardened border routers or gateways; others advocate the use of security agents to recognize anomalies within networks; while others have proposed IoT network segregation [19, 20].

2.2.2 Existing LPWAN Systems in IoT

In furtherance of achieving a more diversified perspective, it is necessary to examine the state of existing homogenous LPWAN systems within the field of IoT. Analysing similar existing implementations could inspire and spark ideas of innovation for the planning and application of this project. As such, in this section, two existing system examples are inspected—LoRaWAN and Symphony Link—due to their pertinence and proficiency for FUOTA. A comparative study by J. P. Queralta et al. [21] found that LoRaWAN was best suited for small-scale public deployments, while Symphony Link provided a robust and more large-scale deployment for private industrial environments. Inspecting these polar opposite systems will provide a broader overview of the state-of-the-art and their applicational values and limitations.

2.2.2.1 LoRaWAN

As mentioned at the beginning of this chapter, LoRaWAN has placed itself in a top position of popularity for IoT communication over available unlicensed ISM bands. It is the only technology to this date that has a publicised working and demonstrated detailed prototype of a firmware update application over an LPWAN. The prototype is an ARM Mbed OS 5 based firmware update over LoRaWAN example application [22] developed by J. Jongboom and J. Stokking—both active LoRa Alliance members—alongside a proposal to properly standardise device updates over LPWANs [23]. The application implements multicast firmware updates over LoRaWAN using an end-device Class C configuration by implementing remote multicast setup, fragmented data block transport, and application layer clock synchronization [2]. Class C is known as the ‘continuous’ reception mode as the receive windows stay persistently open. Figure 1 shows how the open-ended receive window can only close when the device sends a transmission back to the server [24, pp. 59-60]. Due to this the mode has low latency yet must compromise its receiver's power expenditure—making this mode best suited for applications where continuous power is available. However, mode switching can be used for intermittent tasks like firmware updates over-the-air as shown in J. Jongboom’s use case.

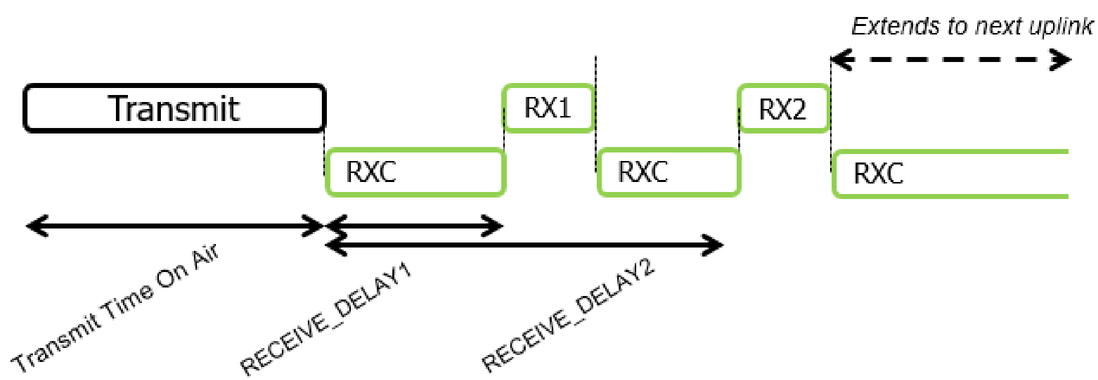

Figure 1: Class C reception window slot timing.

A benefit of this system is that due to its asynchronous nature, congestion at the gateway is significantly reduced as devices do not need to synchronise between receiver and transmitter. Another substantial feature is the security safeguards in place designed to mitigate firmware modification attacks. Precautions have been taken using a configuration of public and private session keys [24] to sign firmware and verify if it came from a trusted party and if it was meant for that specific device, thus, protecting against malicious firmware modification injections.

Conversely, a downside to this system is the cost of the microcontroller Multi-tech xDot [25] hardware and other additional components used. Although relatively cheap, it is still a considerable and unaffordable cost for some business and applicational settings. A more expendable and replaceable product could be better suited in the case of deterioration or wreckage due to unforeseen circumstances. Another complication is the devices seem to have only been tested using fixed transmission parameters and in [26] M. Bor et al. discuss how dynamic configuration of communication settings in LoRa networks can have a profound impact on network scalability. Consequently, such flaws and limited close-proximity indoor testing, illuminate possible drawbacks that may arise within an industrial large-scale scenario—as supported by [21].

2.2.2.2 Symphony Link

Recently, competitive proprietary solutions have been designed to combat the constraints of LoRaWAN. A protocol called Symphony Link [27] maintains a plethora of advantageous features, for instance, solving difficulties with duty cycle limits and supporting the use of repeaters to increase deployed gateway range. Unfortunately, as the technology is relatively new and only mentioned by a few researchers [21, 28, 29], no technical existing firmware update example applications have been found. A single experimental study by D. Patel and M. Won [29] investigates how LPWAN technology has effects on mobility in both indoor and outdoor environments. They designed a functional Symphony Link test platform to conduct an experimental evaluation of performance under various mobility settings, using a GUI designed by Link Labs called Prelude [30]. The study found that LPWAN performance is easily impacted by even minor mobility and that impact is significantly escalated as the distance between the end-device and gateway increased.

The Symphony Link protocol used for this application uses a more flexible duty cycle strategy of frequency hopping combined with a dynamic frequency agility band to allow more packets to be sent at a given time [31]. The protocol also supports repeaters that can extend a gateway's range, meaning overall infrastructural costs can be decreased by employing single gateway base stations and sets of repeaters to expand coverage over a wider area. For an agricultural livestock tracking application in an upland geographical location, network performance will be affected by dense vegetation and hilly terrain. The characteristics of this technology allow such difficulties to be combatted with careful repeater positioning to scale up the network without impacting latency [32]. An experimental study by K. Mikhaylov et al. [33] analyses the performance of a multi-gateway LoRaWAN deployment and found that many assumptions regarding communication within LoRaWAN networks do not hold tight in practice. It is possible that similar findings may hold true for Symphony Link, so theoretical postulations must be challenged with empirical findings to highlight any discrepancies.

A limitation of this protocol is its requirement that all devices must operate under the same conditions, providing no flexibility for class configurations to suit the needs of dissimilar applicational conditions. Moreover, having no straightforward options for asynchronous abilities means that both the sender and receiver are required to perform more complex time synchronization rather than using embedded input clocks. Resultingly, this inflates operational costs and creates a barrier for more simple cost-effective systems that don't require faster data transfer or intricate and efficient mechanisms. Arguably, another drawback is that the protocol is proprietary and closed source, so it caps any external configurability and adaptation to an application's environment. It is impossible to know, for instance, if the data packet transfer design holds any redundant data that could be removed to reduce power consumption—like how the Cayenne Low Power Payload (LPP) transmission library functions for other LPWAN networks [34].

2.2.3 Reflection

After studying the aforementioned systems, it is simpler to conceptualise what problems will be faced as this research moves forward. Integral system elements that have hindered advancements and the functionality of other existing systems have illuminated key issues that need to be addressed for a project like Milliamp's to succeed.

Reflecting on core issues underlined by the LoRaWAN system, being able to integrate dynamic frequency configurations throughout the data exchange to maintain the most favourable radio frequency (RF) settings during device mobility would be an essential requirement. Not only would this help maximise battery life, as shown by N. Benkahla et al. [35], it would also improve performance and the success of the FUOTA process in arduous hilly terrain [30]. Examining various configuration of transmission parameters over a LoRaWAN Class C implementation could help characterize the system better. Although the Symphony Link protocol is beyond the bounds of this project, it has helped to compare a similar LPWAN system against the LoRaWAN protocol to highlight its benefits and flaws. For instance, the comparison of systems illuminated possibilities of multi-gateway deployments using repeaters for more cost-effective, extended, and efficient transmission to end-devices in outdoor terrain. Yet as discussed, other studies showed that such claims may not hold true for real-life deployments, thus, emphasizing the need for ground-truth experimentation to be conducted.

Finally, musing upon the historical events discussed in section 2.2.1 has called attention to the careful consideration of development language and platforms used. It has accentuated the importance of error detection and the role that a language and its compiler play in finding and resolving bugs or defects. Additionally, the more pertinent happenings of 2016 [10] demanded improved deliberation of IoT architectural design, stimulating thought of better system security planning using hardened gateways and subnets. All of these chronicled incidents have made it critically clear that improved and effective firmware update mechanisms are essential. This has provided further motivation and inspiration to attain revealing research of how a functional, power efficient FUOTA process over LoRaWAN can be obtained.

2.2.4 Discussion

Now that existing system shortcomings have been revealed, discussion can be had on how to amend them. To address the issues reflected in section 2.2.3, a mechanism of dynamic frequency configurations throughout data exchange using slotted synchronous downlinks could provide improvements to data transfer reliability, power consumption, and mobility effects. LoRaWAN Class B 'beaconing' mode provides frequently scheduled receive windows alongside device class A type propensity [2]. The device mode time-synchronises the network using periodic beaconing broadcasts via gateways, as illustrated below in figure 2.

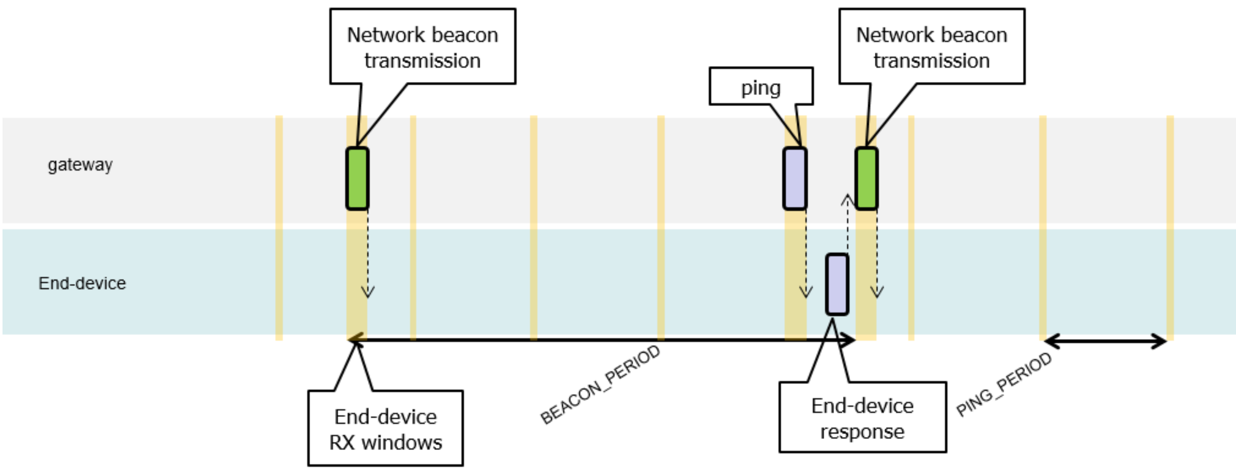

Figure 2: Class B beacon reception and ping slots

Such a configuration would allow the gateway to inform nodes of a change in communication parameters on the next available receive window to then change accordingly for subsequent transmissions. If these communication parameters can be correctly optimised, the sleeping times of the end-devices can be increased, thereby increasing the devices battery life. Additionally, this facilitates device adaptation for varying transmission ranges which may allow the device to be more mobile.

As of yet, it does not appear that an implementation of a system of this kind for firmware updates exists. As the technology is relatively new, many have only tested this operational mode through simulations and LoRa-based testbed infrastructures [36, 37, 38, 39], however, none have created a working prototype to examine and test within a real-world environment. The findings in [33] further emphasise the need for empirical evidence—specifically for LoRaWAN networks—as common assumptions in regards to communication have been demonstrated to be inaccurate. In respect to this, this research aims to gather objective evidence of the premise and evaluate and measure the Class C FUOTA system under real-life environmental conditions to then develop a use case further for the Class B FUOTA system.

Chapter 3

Design

This chapter aims to deliberate over the project's overall system architecture and technological infrastructure. Designs of the FUOTA process information flow from end-device to network are established and discussed. An approach to measuring low power consumption of the end-device is explored to later lead into the testing and evaluation of the designed and implemented FUOTA system. It should be noted that half of the work covered in section 3.2 was already completed during the work placement for Milliamp.

3.1 System Infrastructure

Before the FUOTA process can be empirically analysed, a working implementation is required for constructs to be measured. Several technological characteristics must be selected to procure a functioning system. The proceeding sub-chapters aim to discuss decisions involved when choosing these infrastructural fundamentals.

3.1.1 ChirpStack Platform

It is extremely important to work with an appropriate and effective framework that facilitates all the required factors for a project of this kind. As mentioned in section 2.1.1, TTN was an unsuitable platform for this application due to its limiting fair access downlink policies. ChirpStack [40] on the other hand is an open-source network server stack that has no downlink restraints, it conforms to a modular structure while being able to neatly integrate with other infrastructures. The platform allows you to set up a fully scalable, secure stack through MQTT and TLS, along with provided API integration via gRPC and REST for external services. A plethora of other IoT platforms for LoRaWAN exist, yet none match the calibre of ChirpStack's fine-grained feature configurability, which allows greater control throughout the entire system framework.

3.1.2 Language

Corresponding to what was discussed earlier in chapter 2.2.1, using a more structured language like C++ which possesses debugging exceptions and a stricter compiler would be better suited for a high-level application on the end-device side. For network backend services like FUOTA deployments, a language like Go—sometimes referred to as 'Golang'—would be more appropriate [41]. Go is a scripting language that is ideal for rapid network development with Docker—which itself is written in Go. It is an extremely powerful and efficient language like C/C++, handling parallelisms like Java, and is simple to read, pick up, and learn like Python.

3.1.3 LoRaMAC-node Library

LoRaMAC-node [42] is an end-device LoRaWAN stack application library that works in accordance to the LoRaWAN Specification v1.0.3 [2]. The concept of its API follows the idea of primitives—Request-Confirm and Indication-Response—of the IEEE Standard for local and metropolitan area networks [43]. The LoRaMAC layer provides MAC Common Part Sublayer (MCPS) services, MAC Layer Management Entity (MLME) services, and a MAC Information Base (MIB). To provide a brief overview, the LoRaMAC layer makes use of the MCPS services to transmit and receive data, the MLME service manages the LoRaWAN network, and the MIB holds configuration data for the LoRaMAC layer along with runtime information [44]. The library unfortunately doesn't support the ESP32 series, though it does provide porting guide documentation. It is the only major LoRaWAN library for end-nodes that provides the up-to-date functionality required for FUOTA.

3.2 System Architecture

The overall system framework consists of animal tags that communicate with nearby gateways that are geographically distributed to cover a selected area. Figure 3 below, shows a simplistic overview of the interoperability between the various components that constitute this project's architecture. Each of these system elements will now be briefly discussed:

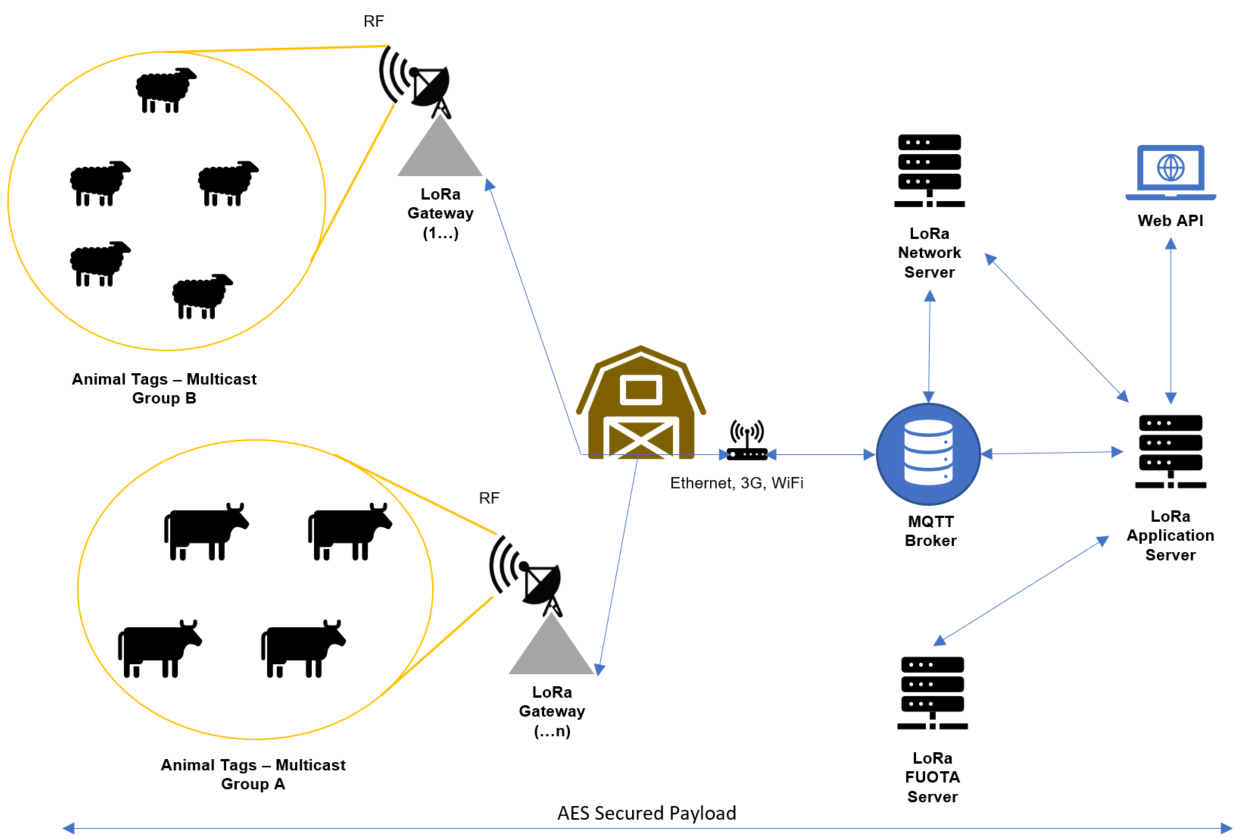

Figure 3: Basic overview of system architecture network communication

- **Animal Tag** –An ESP32 series end-device and the target for firmware updates. Their main operation involves the utilisation of a machine-learning algorithm to detect livestock ailments. When a negative animal health status is detected, the device will send an uplink alert to the gateway which will alert a farmer's dashboard. The devices can be clustered into multicast groups for firmware updates, meaning the server will treat all the devices as one and apply the same update to them all over multicast.
- **LoRa Gateway** – This component is equipped with a LoRa concentrator that allows it to act as a medium between the animal tags and the LoRa network. As LoRaWAN networks are ALOHA based, the gateways can forward communications from any animal tag within range. Figure 4 shows a more comprehensive breakdown of this component's internal parts and operations. As illustrated, the gateway connects to the network server via an MQTT broker over TCP transmission rather than UDP due to the ChirpStack Gateway bridge component—making the connection more reliable in case packet loss is common. Private credentials can be configured for each gateway so that only those with valid certification have the ability to ingest data into the network.
- **LoRa Network Server** – This server holds the responsibility of de-duplicating LoRaWAN frames, scheduling and queueing downlinks, communicating with the application server, translating LoRaWAN MAC layer commands, and finally ensuring device authentication for the security and reliability of data routing through the network.
- **LoRa Application Server –** This server handles the LoRaWAN application layer and holds the responsibility of retaining the device inventory section of the entire LoRaWAN infrastructure. This involves handling join requests, the encryption of application payloads, integrating the network with external services using gRPC and RESTful API, and offering a web interface.
- **Web API** – Offered by the application server, this can be used to manage users, applications, organisations, gateways, devices, and more.
- **LoRa FUOTA Server** – This server integrates itself with the application server using HTTP to receive uplinks and the server API to enqueue firmware downlink payloads—acting as a gRPC endpoint. It handles the LoRaWAN application layer information flow on the server-side to manage firmware patch deployments to device multicast groups.
- **MQTT Broker** – A server that distributes and filters received MQTT messages from clients based on topics to then dispense them to subscribers**.** This broker supports SSL and TLS meaning a secure connection can be established without malicious entities intercepting or tampering with data.

A higher fidelity architectural diagram is provided in figure 4, showing a more elaborate overview of system communication over various network layers. Each system element's internal components are detailed and the interoperability between them is shown. Examples of potential subscribed and published MQTT topics, along with gRPC service messages are illustrated.

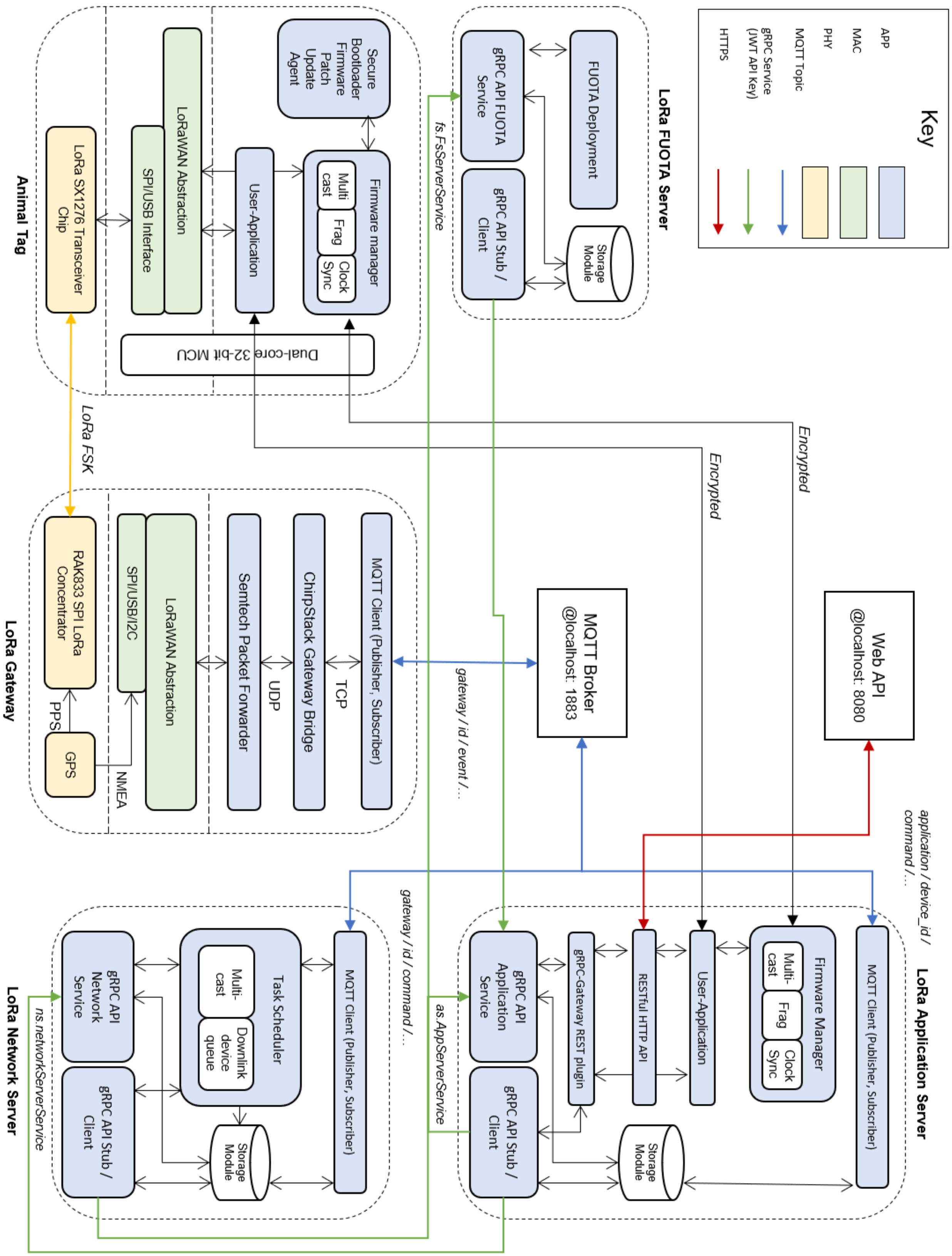

Figure 4: Comprehensive LoRaWAN network architecture overview for FUOTA deployments

3.3 Network/end-device Interworking

In 2018 the LoRa Alliance announced the public release of three new specifications to standardise and assist the FUOTA process for LoRaWAN. The specifications are of application layer clock synchronisation [45], remote multicast setup [46], and fragmented data block transportation [47]. In accordance with said specifications, this section shows designs for message flow between the end-device and application server during a Class C FUOTA deployment, while utilizing the LoRaMAC-node library primitives and call-backs.

3.3.1 Time Synchronization

Before a FUOTA deployment can be configured, the end-device must synchronise its real-time clock (RTC) to the networks GPS clock with second precision. This way, all end-devices in a multicast group can switch operational modes to Class C temporarily and synchronously at the beginning of a firmware transmission RX slot. This can be done at either the application layer using *AppTimeReq* or at the MAC layer using *DeviceTimeReq* as shown below in figure 5.

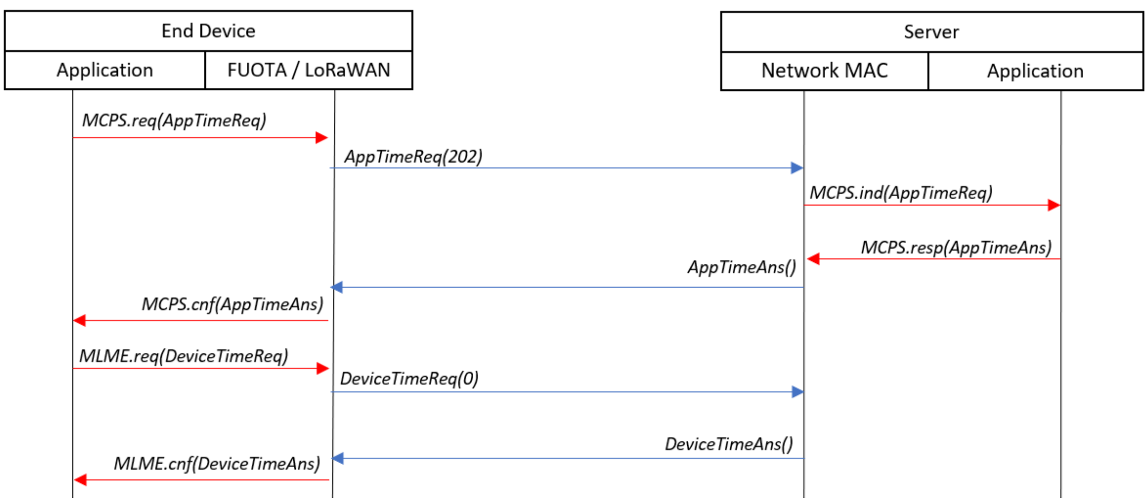

Figure 3: Time synchronization at application and MAC layer before setting up FUOTA session

3.3.2 Multicast Setup

Another requirement for the FUOTA process is to program a multicast distribution window into a selected group of end-devices. This will allow control over the initial group switch to Class C temporarily and then the later reversion back to ordinary Class A operation. A multicast group context is defined by a multicast group ID, address, key, and frame counter. An initial *McGroupSetupReq* message from the server-side is what instantiates the multicast setup as shown in figure 6.

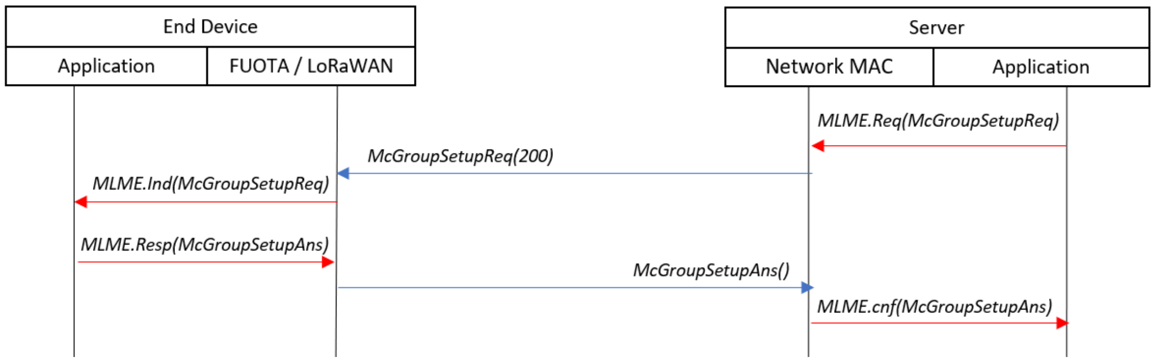

Figure 4: Multicast session setup at application and MAC layer

3.3.3 Firmware Fragmentation Setup

In order to safely transport an entire firmware image without having to repeat each fragment multiple times, a fragmented data block transport procedure makes use of forward error correction (FEC). The firmware to be sent is fragmented into n equal-length fragments, where each fragments fits into a LoRaWAN transmitted payload. A *FragSessionSetupReq* command—shown in figure 7—is sent from the server to be acknowledged by all devices within a multicast group. The command uses parameters such as session ID, fragment size, number of fragments, padding, and a file descriptor.

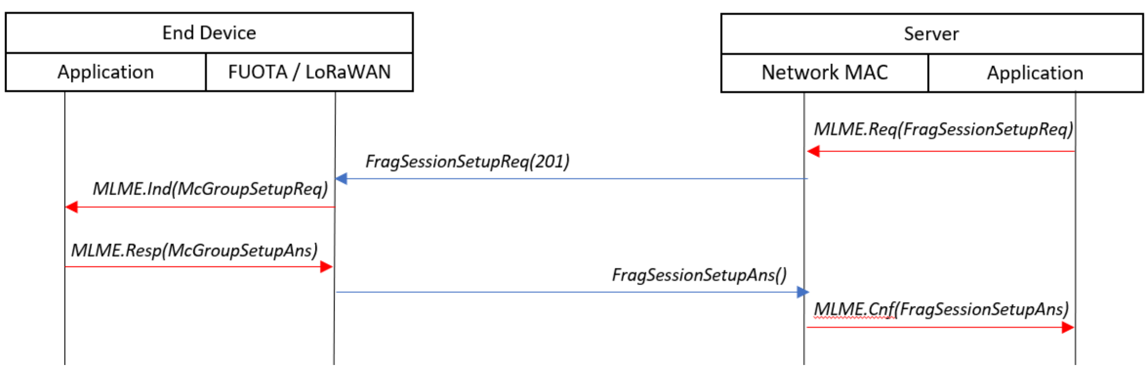

Figure 5: Fragmented data block transport session initialisation for firmware transmission

3.3.4 Class C Session Setup

Now that the devices RTC's are synchronized, all the devices must be programmed to open their receive windows simultaneously, while using the same radio channel, data rate, and other parameters. To transport and configure the required parameters, the remote multicast setup package allows the definition of a Class C session setup. The setup is initialised by the FUOTA server using the *McClassCSessionReq* command and helps define the start time, duration, and radio parameters that each device will require to receive the firmware update.

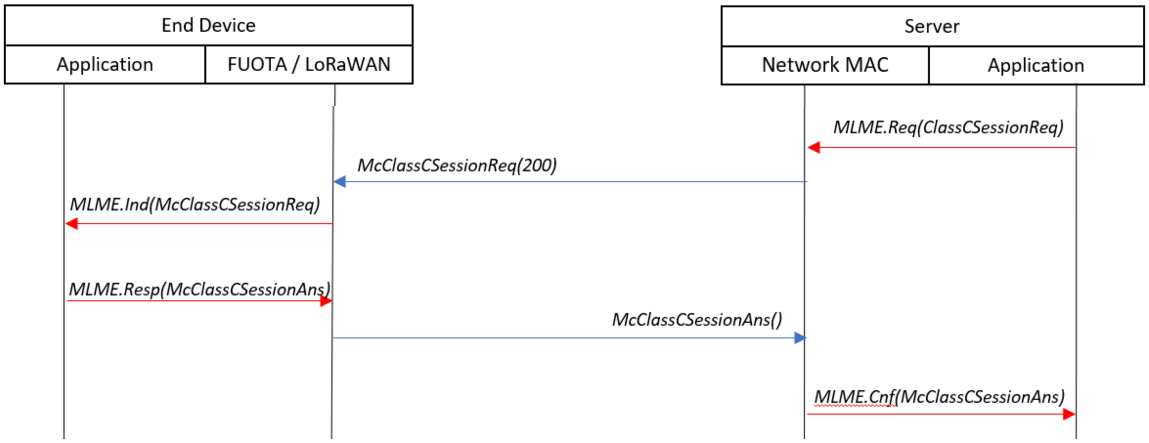

Figure 6: Class C session initialization before RX window can be opened for the FUOTA process

3.3 Monitoring Low Power Consumption

To better understand the manner in which energy is consumed throughout the FUOTA process a small test circuit has been outlined. The Heltec Wi-Fi LoRa 32 IoT dev-board consumes 800µA when operating at its lowest capacity—during deep-sleep mode—and when producing a LoRa 20bB output it can consume up to 130mA. Because of this, a VCT Monitor Click [48] module has been selected to accurately monitor low current use. The board incorporates an LTC2990 that holds a 14-bit ADC and is designed to operate with 3.3v logic voltage levels to use the I^2C

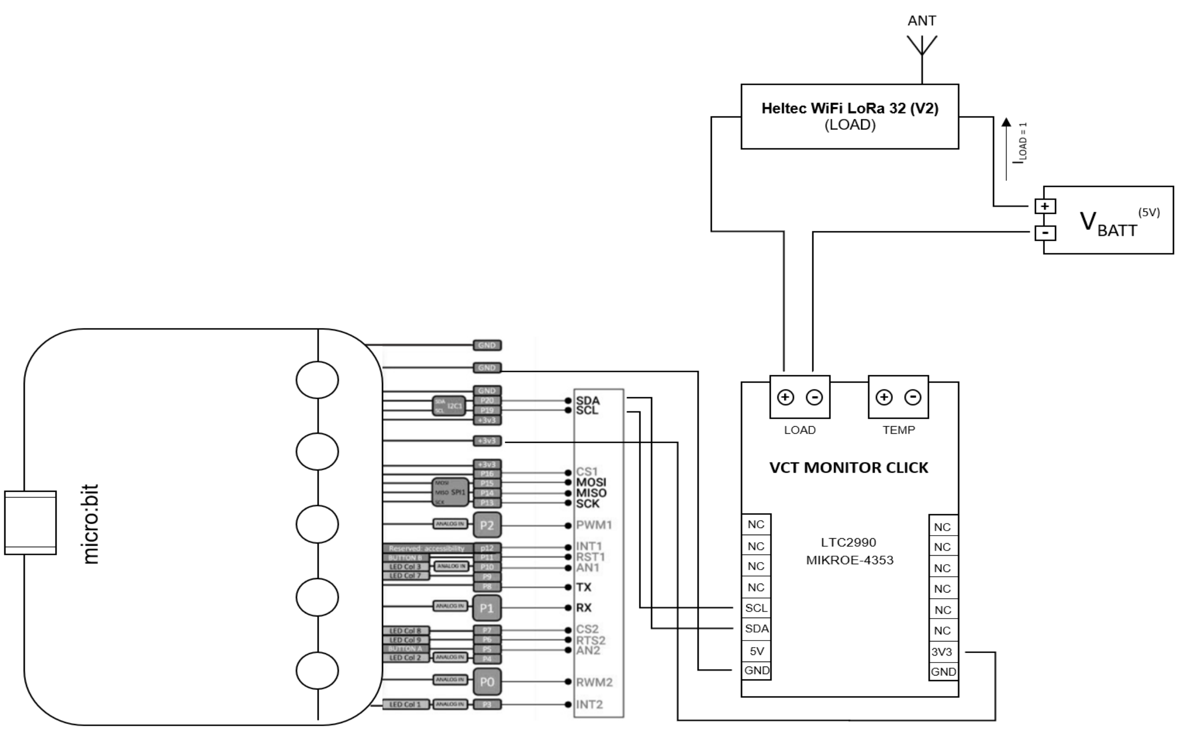

Figure 7: Circuit schematic for measuring current draw of the Heltec ESP32 LoRa node over I^2C

communication lines. The advantage of this is that there is no dependence on the voltage being perfect at the micro:bit; when observing small currents, this could make all the difference. To prevent confounding voltage readings, it will be necessary to use a power supply to provide a stable output in case of any dips in voltage.

Monitoring power consumption in a time-series fashion during the FUOTA process could reveal patterns and characteristics for a more detailed, in-depth analysis of the applicational behaviour—as opposed to measuring average current draw. This setup could be later applied on an implementation of a Class B model undergoing a firmware update to gather contrasting data of the behaviour of a slotted, synchronous multicast group's power use.

Chapter 4

Implementation

This chapter intends to cover the main algorithms, data structures and procedures used throughout the implementation so that the reader can gain a detailed understanding of how the various applications operate internally. This will include development of both the animal tag 'ping-pong' application and the created power monitoring tool. At the end of this chapter a short discussion is had to explain implementation shortcomings of the main FUOTA application.

4.1 Tag Ping-Pong Development

To determine how effectively the animal tag operates within its applicational environment a basic uplink/downlink 'ping-pong' application has been developed to test the RF signal using a received signal strength indicator (RSSI) measurement. Figure 8 below shows the basic flow of operation involved of which are determined in the software by setting the device state: initialise, join, cycle time, transmit, deep-sleep, and display.

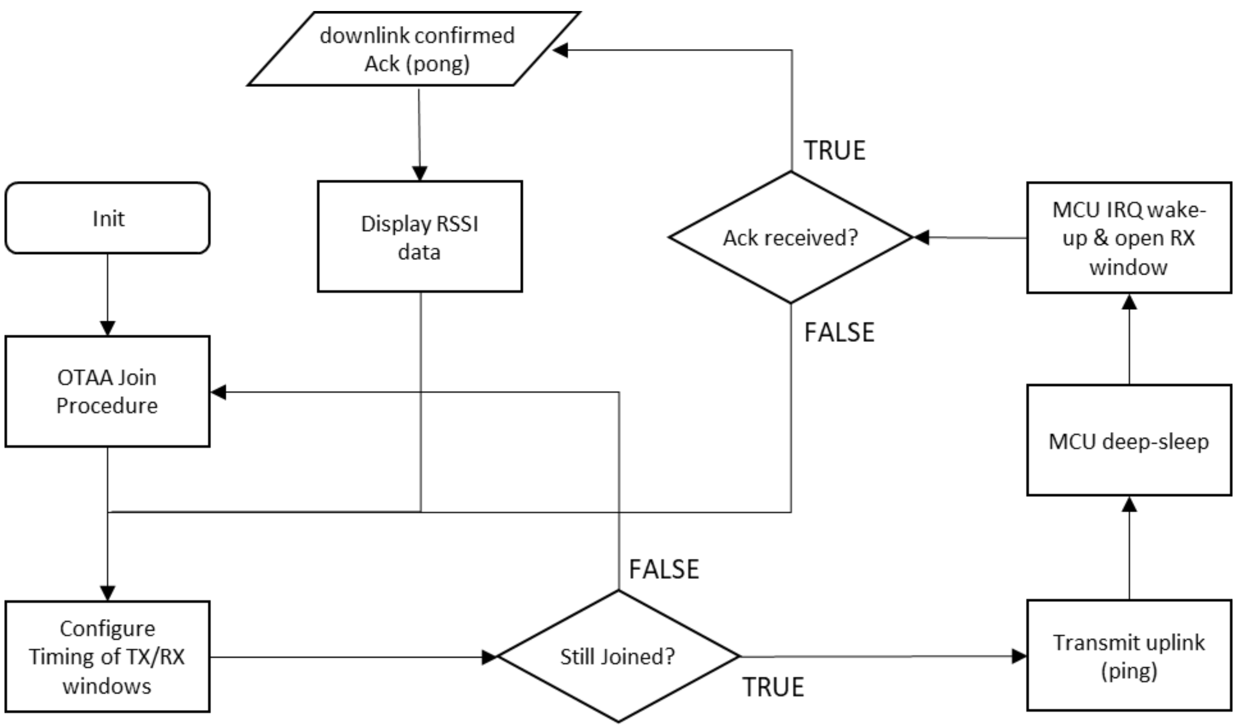

Figure 8: Flow diagram of ping-pong application logic

Initialise - The initialisation state configures the required LoRaMAC layer primitives and call-backs needed by the application layer. The *LoRaMacPrimitives_t* and *LoRaMacCallback_t* data structures both use function pointers to link procedure functionality. The primitive's structure provides indication and confirm functions, while the call-back structure provides functions to fetch information about the device. Required time settings are configured alongside regional parameters.

Join – Here an MLME request is performed to solicit an over-the-air activation join procedure. Programmed keys—device EUI, application EUI, and an AES-128 application key—have to be provided to safely authenticate the device and permit the join-request. After consultation, the status of this operation is then reported back by the network server to validate the request by responding with a join-accept message.

Cycle Time - From the returned join-accept message a receive delay figure is provided to calculate the start time of the receive window. The window is scheduled to open according to the transmit time-on-air plus the receive delay time (+/- 1 second) after the end of the uplink modulation. Thus, in this state we configure the MCU's internal RTC to set the timer value and start time according to calculations.

Transmit – During the sending stage we initially re-confirm the join status to the network by querying if transmission is still possible at the LoRaMAC layer. If returned true, function *tx_data_frame(uint8_t port)* is called to prepare the transmission frame and send it as an MCPS request using a confirmed uplink request type. As soon as the transmit operation is completed the device switches to deep-sleep state.

Deep-sleep - Before switching device mode to deep-sleep this state sets the SX1276 transceiver radio IRQ process so that it's ready to activate. The embedded RTC controller has a built-in timer that can be used to activate an interrupt request after a predefined amount of time. The device will awaken ready for the next receive window to accept any incoming downlink messages.

Display – Once the uplink acknowledgment message is received the RSSI is displayed on the OLED display using a *U8x8 library*. This helps to determine how well the device can 'hear' the received signal from the gateway and allows experimental data to be recorded on network performance. The RSSI value is provided by the primitive *McpsIndication_t* data structure during downlink data handling. At the end of the receive cycle the logic recurses.

4.1.1 Porting the LoRaMac-node Library to ESP32

Porting over the required LoRaMac-node library functionality to the ESP32 hardware platform was labour-intensive and challenging. Due to the novel nature of FUOTA over LoRaWAN, the few existing libraries available only partly implemented functionality for operations of older specifications. This meant that generic abstraction layers had to be heavily adapted to operate following the newer guidelines. Standard peripheral files had to be added alongside drivers for SPI, RTC, GPIO, and SX1276 to board translation. Fortunately, most of the drivers could be taken from other projects and altered to function for the LoRaMac-node library operations. For instance, RTC board operations had to be translated to be compatible with the library's *systime.c* functions *SysTimeGet()* and *SysTimeSet()*. More difficulties were had with oddly nested conditional compilation directives for blocks being gated based on target chip architecture. This caused certain structure types to be incorrectly defined as platform-specific settings needed configuration. To debug such issues, *#warning* pre-processor directives were used to locate and understand compilation flow.

4.2 Current Draw Measurement Application

As discussed earlier in chapter 3.3, a power measurement application is required to characterise the system and to understand what features are most influential in the role of power consumption. This will be helpful in the later testing phases to determine what and how to test before field measurements are conducted.

4.3.1 Micro:bit Monitor

To monitor the voltage readings from the VCT monitor click module and to perform calculations, a micro:bit V1 microcontroller was employed. The micro:bit DAL runtime [49] was utilized to support the development of this voltage monitoring application. It operates simply by registering a *readVoltage()* function to be called on a *MICROBIT_BUTTON_EVT_CLICK* at the start of the application. Once the event is heard by clicking button B, the voltage recording begins—discussed in detail in section 4.3.2. The recorded voltage is then simply outputted to serial with commas to separate each read value. This data can then be copied to a .csv file to then later be plotted on a graph for visualisation using Microsoft Excel.

4.3.2 LTC2990

The LTC2990 [50] is an I2C configurable voltage, temperature, and current monitor featured on the VCT monitor click board. The only software support provided for this module is given in MIKROE package form for a MIKROE compiler, thus, a library for the micro:bit had to be developed independently. The library consists of an LTC2990 class that is initialised by parsing a I2C base address—following Table 1, as both ADR1 and ADR0 pins are pulled low the base address is 0x98—and an I2C bus. The class then features *init()*, *status()*, *trigger()*, *getVoltageFloat()*, and a few basic check functions for its operation.

Hex I2C Base Address	Binary I2C Base Address	ADR1	ADR0
98h	1001 100X*	0	0
9Ah	1001 101X*	0	1
9Ch	1001 110X*	1	0
9Eh	1001 111X*	1	1
EEh	1110 1110	Global Sync Address	

*Table 1: I2C Base Addresses (*X = R√W Bit)*

To initialise the chip, a byte is written to the control register *0x01* to select the measurement mode of the device. Measurements are activated via the trigger register *0x02* to allow a conversion to take place and then the status register *0x00* is queried to see if the voltage reading is ready to be sampled. After waiting for the selected voltage register to return a not busy indicator, the voltage on *V1 0x06* through to *V4 0x0D* can be read.

Voltage results are spread over two register bytes, an MSB and LSB register. The voltage MSB register provides a *data_valid* bit to show if the register contents have been accessed since the last write. The MSB register *b[5:0]* holds the two's compliment conversion result and the LSB register *b[7:0]* store the conversion bits. The sign value can be determined by bit six as shown in table 2. To calculate the current and determine how quickly power is being consumed we can use the Ohm's law formula: $current = b[14:0] \,.\, V_{SUPPLY} / R_{SENSE}$ where R_{SENSE} is the current sensing resistor on the VCT monitor board and V_{SUPPLY} is the supplied voltage. Essentially, what is being measured is a drop in voltage over the fixed shunt resistor R_{SENSE}.

Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DV*	Sign	D13	D12	D11	D10	D9	D8

Table 2: MSB register format for voltage/current measurement.

Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
D7	D6	D5	D4	D3	D2	D1	D0

Table 3: LSB register format for voltage/current measurement.

Finally, to determine how much power is being consumed with the two values of ampere (A) and voltage (V) we can use Watt's formula: *P(W) = I(A) x V(V).*

4.4 Discussion of Constraints

Unfortunately, due mainly to time constraints, the designs specified in section 3.3 could not be fully implemented. Not all the FUOTA operations have been suitably developed and tested in the LoRaMac-node library on account of the LoRaWAN FUOTA feature having only emerged rather recently. As the main FUOTA application could not be developed, the implementations discussed in this chapter were created with aims to evaluate shared aspects between them and the designed FOUTA application. The 'ping-pong' applicational tool similarly mimics certain operations detailed in the section 3.3 designs, ergo allowing characterisation of how those aspects operated and behaved. Empirical evidence of this system can now be gathered and reflected on the FUOTA designs to evaluate and predict how the model may behave. For instance, measuring the TX and RX power consumption of various frequency settings and then applying that data to the models messaging patterns to predict how much power could be consumed.

The workload for this project was extensive and such unforeseen difficulties—incomplete libraries and platform porting—massively impacted the project schedule. In a blind, thoughtless flurry of ambitious thinking, and with intentions to push and challenge oneself, boundaries were reached, and capabilities were overestimated given the time frame. However, on a positive note, dynamic adaptation to the challenging circumstances resulted in the development of solutions to overcome such hurdles, thereby even in the face of adversity, practical knowledge and a successful project outcome are still possible.

Chapter 5

System in Operation

To assist the reader in developing a more comprehensive understanding of how the system functions, this chapter will go over the various system components and their operations in a pipeline fashion from tag to application server. Note that usability of this system is not the principal aim of this project but rather evaluating the efficacy of device's performance.

5.1 Animal Tag

The animal tag testing apparatus shown below in figure 9 was developed in accordance with the designed circuit schematic specified in section 3.3. To provide an overview of the components the following labels are given:

1. Heltec ESP32 Wi-Fi LoRa tag device
2. 10000mAh external battery power bank
3. VCT monitor click module
4. Kitronik edge connector breakout board
5. Micro:bit V1

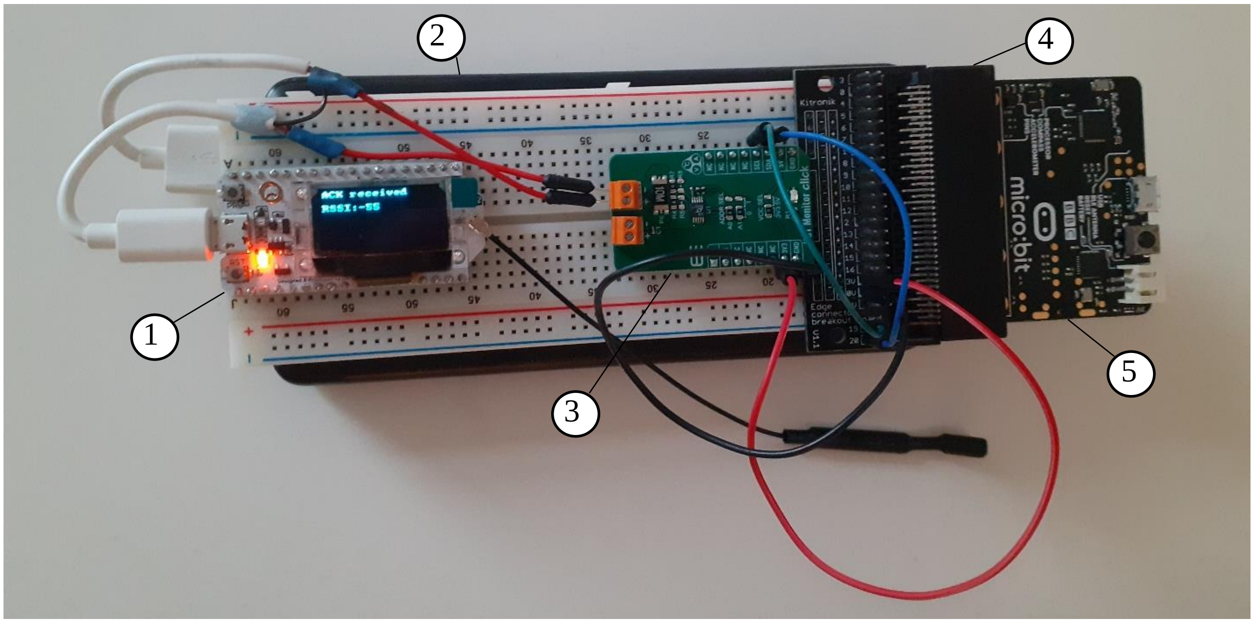

Figure 9: RSSI and power consumption testing apparatus wired to the Heltec ESP32 Wi-Fi LoRa animal tag microcontroller.

As soon as the tool powers on, the tag is programmed to attempt a join-request with the network server before transmitting confirmed uplink messages and opening its RX window to receive the server's responses. To measure any power consumed during this process, we simply plug the micro:bit into a computer and monitor the serial output—activated by pressing button B on the underside of the micro:bit.

Figure 10: Serial monitor terminal output for the animal tag ping-pong application.

The screenshot above in figure 10, shows the serial monitor output for the tag during the 'ping-pong' applications use. The OLED display outputs the RSSI for field measurements, however, a more detailed description of the frequency settings being used is outputted to serial. These settings can be monitored and adjusted later for more in-depth testing in the later chapters.

5.2 RAK833 Gateway

The next stage of the payload transmission is at the LoRa gateway. Below in figure 11, after executing the command '*sudo tcpdump -i lo -Auq udp 1700'* —to capture any UDP packets on port 1700—it is possible to observe the received and transmitted payloads forwarded by the

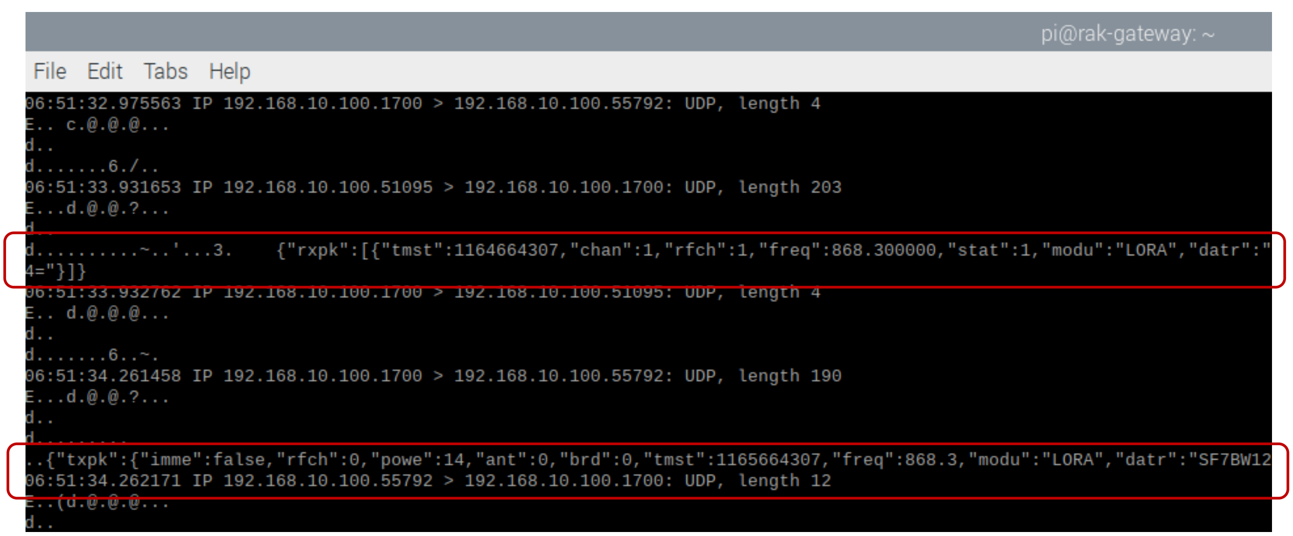

Figure 11: Terminal output of RX and TX payloads from monitoring UDP port 1700 on the RAK833 Raspberry Pi hosted gateway.

gateway's UDP packet forwarder to either server or device. A configured ChirpStack gateway bridge hosted on the Raspberry Pi then acts as a relay between the network server and the packet forwarder to translate communications more securely over TCP. This can be observed by executing *'sudo journalctl -u chirpstack-gateway-bridge -f -n 50'* as the server runs in the background as a system service—shown below in figure 12.

Figure 12: Terminal output of the ChirpStack gateway bridge system service showing uplink and downlink events.

5.3 ChirpStack Network Server

At the network server, downlink responses are scheduled, and uplinks are relayed to the application server which handles any join-requests. In figure 13 below, we can observe the confirmed data uplinks and downlink commands being subscribed and published to the MQTT broker.

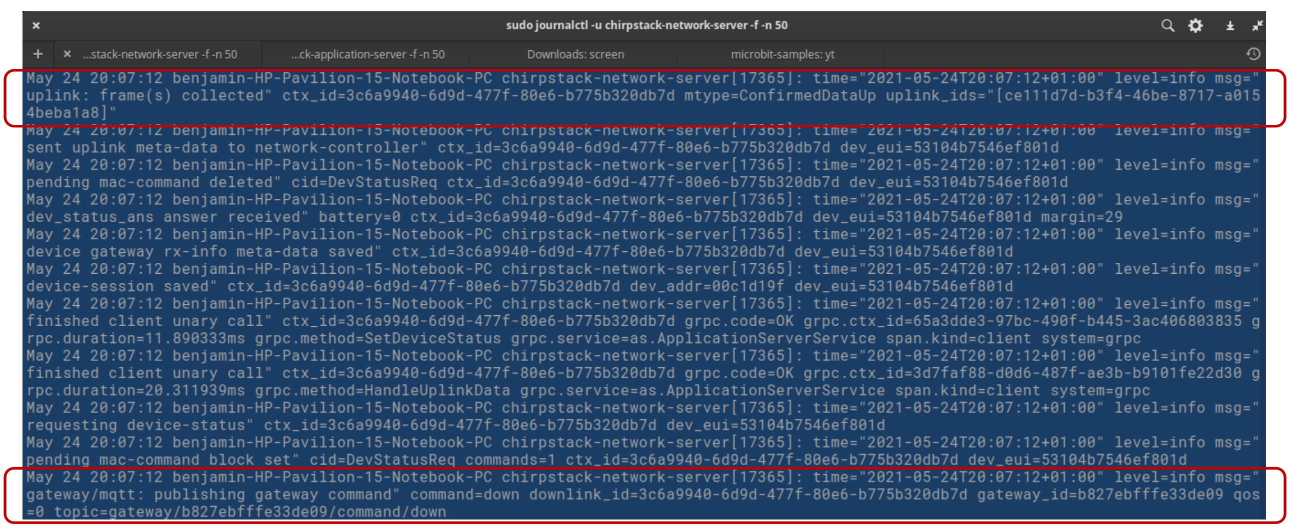

Figure 13: Terminal output of the ChirpStack network server routing data between the application server and RAK833 gateway.

5.2 ChirpStack Application Server

Finally, we have followed the join-request and proceeding confirmed uplink messages to the application server which offers a web-interface to interact and observe with the devices, gateways, and applications. Figure 14 shows the devices messages being received at the application server. The initial join request is followed by uplinks and status messages; configured by the network server using the *DevStatusReq* mac-commands.

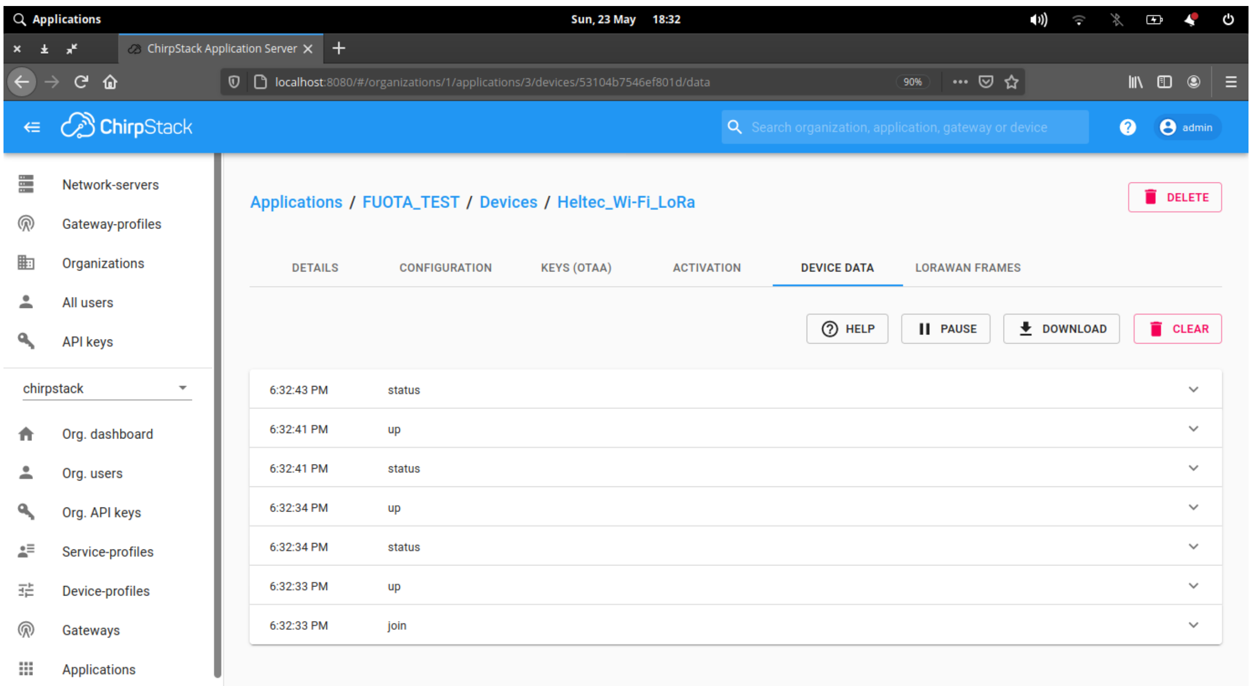

Figure 14: Screenshot of the live device data within the 'FUOTA_TEST' application during execution of the 'ping-pong' software.

Chapter 6

Testing & Evaluation

This chapter begins with system testing of data rate and transmission power settings to establish a baseline of what to measure during the proceeding field study. Next, a critical analysis and evaluation of the testing is conducted, followed by a discussion of potential solutions to problems found.

6.1 On-the-Bench System Testing

For the purpose of characterising the system, regulated 'on-the-bench' tests were conducted. Such examination allows for better awareness of how certain system variables behave before conducting further tests. Accordingly, this section examines various impactful frequency settings to observe their influence on energy use. Section 2.2.3 reflects on the significance of frequency parameter alteration during a FUOTA deployment, thus, exactly how such configurations affect the system are investigated.

6.1.1 Data Rate Energy Consumption

The first parameter we investigate is the data rate (DR) of the LoRa modulation. The data rate depends on the bandwidth and spreading factor used—this is dependent on regional regulation and frequency plans. LoRaWAN channels have usable bandwidth of either 125kHz, 250kHz, or 500kHz. The spreading factor (SF) value defines the number or raw bits that can be encoded by a symbol, so if the number of bits that can be encoded by a symbol is seven, then the spread factor is SF7. Each sweep signal (or symbol) can be divided into 2^{SF} chips and the symbol rate (SR) can be calculated by dividing the bandwidth (BW) by 2^{SF}. The final thing we need to know is the coding rate (CR) which refers to the proportion of bits transmitted that contain data. The data rate can then be calculated using the formula:

$$DR\ (bps) = SF \times SR \times \frac{4}{(4 + CR)}$$

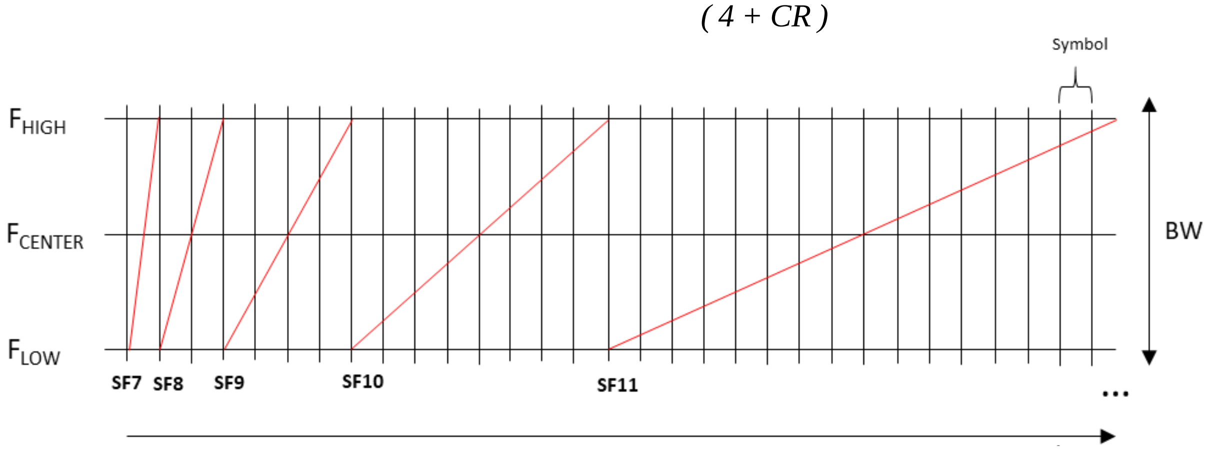

Figure 15: Overview of spreading factor with respect to symbol duration

From this we can deduce that if you increase bandwidth the data rate increases and if you increase the spreading factor the data rate decreases. Figure 15 shows that increasing the spreading factor actually reduces the data rate by half and that the message transmission time increases, which in theory is tantamount to an increase in communication range.

To test and compare the power use of varying data rates, the tag's current consumption was monitored using the application developed in section 4.2 during a confirmed uplink 'ping-pong'. This provided the results illustrated in figure 16 of which seemingly correspond to the message behaviour mentioned in figure 1. As expected, the findings agree with the spreading factor theory discussed above with transmission time doubling with each data rate decrease. Note that the data rates used for this test have a fixed bandwidth of 125kHz and a default error correction rate of 4/5. All transmissions use a fixed default TX power of 13 dBm.

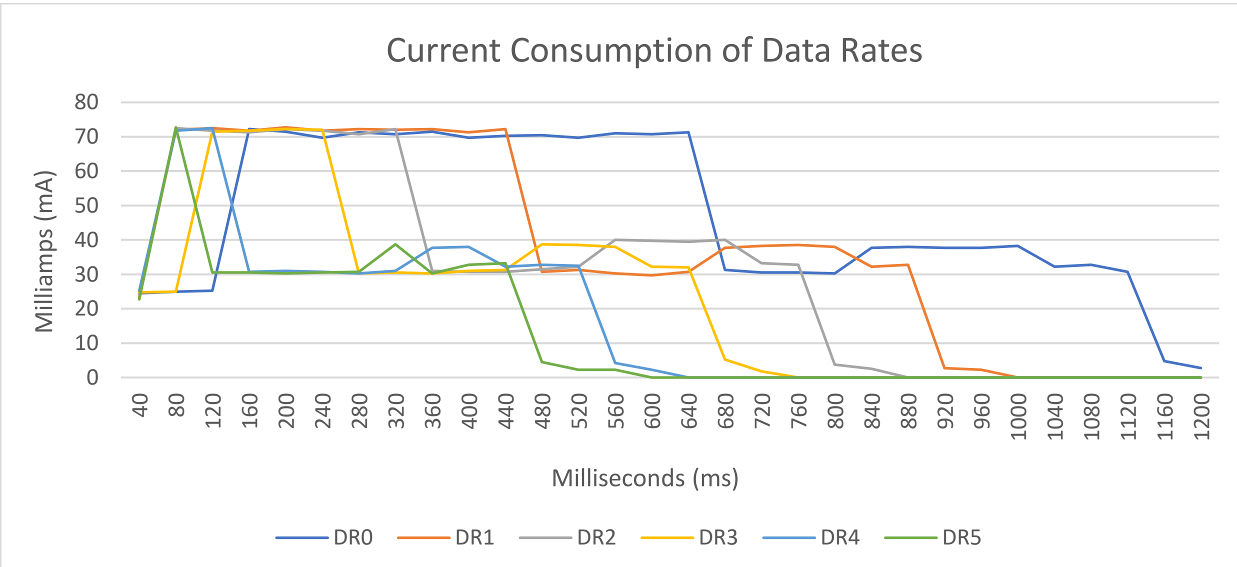

Figure 16: Current consumption characteristics of the Heltec ESP32 Wi-Fi Lora (V2) using varying data rates during a confirmed uplink 'ping-pong'

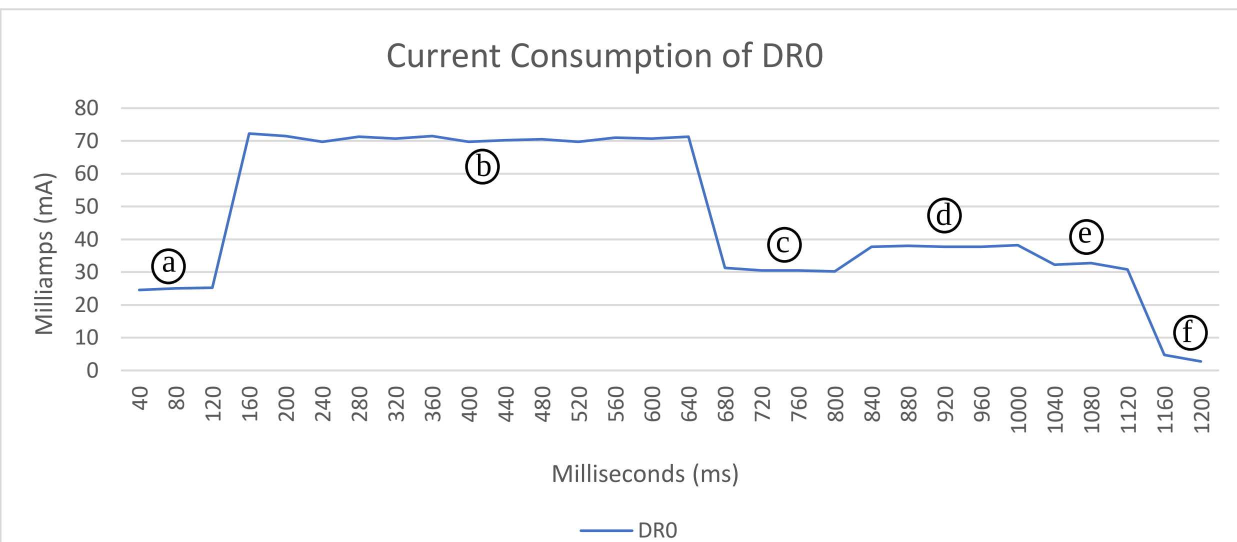

Figure 17: Labelled attributes of confirmed uplink behaviour during current measurement time periods.

Through observation and deduction, we can begin to attribute current use behaviour to the various periods of the confirmed uplink. For the sake of simplicity, figure 17 shows the current consumption readings for only *DR_0* as it more clearly shows the different stages of the transmission due to the lengthier nature of higher spread factors. To profile these labelled attributes and provide characteristics, they have been listed and described in table 4 below.

Letter	Attribute	Variable	Time (ms)	Current Consumption (mA)
a	Initialisation Period	V_{INIT}	160ms	24.9
b	Transmission	V_{TX}	520ms	70.8
c	Wait period (receive delay)	V_{WAIT}	120ms	30.6
d	Receive ACK window	V_{RX}	200ms	37.9
e	Post-processing ACK	V_{ACK}	160ms	31.9
f	Deep-sleep mode transition	V_{SLEEP}	40ms	3.75

Table 4: Attribute labels for various time periods of confirmed uplink DR0 transmission.

Another notable observation during these measurements was the difference in intervals between each confirmed uplink packet transmission for the various data rates. The mean interval time between uplinks exponentially grew as the data rate decreased—shown by the graph in figure 18. This can be brought down to the maximum duty-cycle limitation per sub-band. The time of emission and time-on-air duration of the transmission is recorded, and the following equation used to determine when the sub-band can be used again:

$$TimeOff_{sub\text{-}band} = (\ TimeOnAir\ /\ Duty\ Cycle_{sub\text{-}band}\ ) - TimeOnAir$$

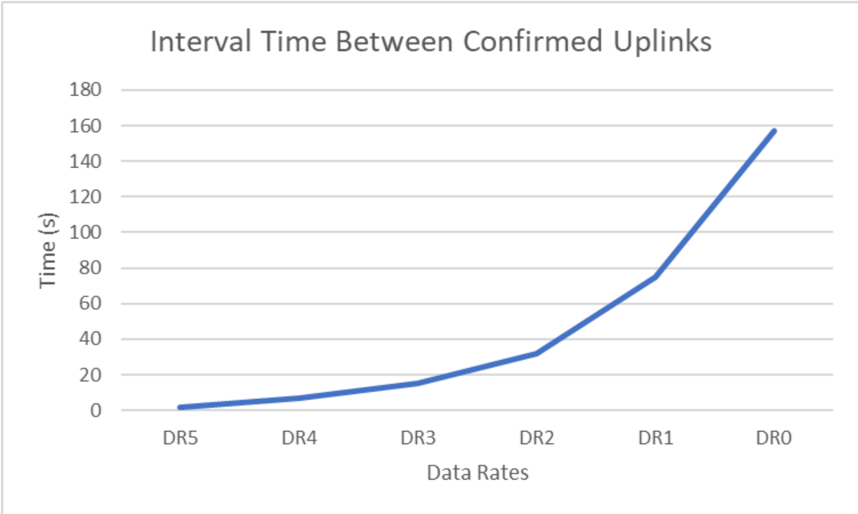

Figure 18: Graph showing exponential growth of mean interval time between confirmed uplinks

A higher spreading factor suggests fewer chirps per second and so less encoded data a second. When comparing this with the lower spread factors, transmitting the same amount of data with a higher spread factor means more time on-air. More time-on-air means that the node transmits for longer and uses more power.

During the data readings of the lower data rates—higher spread factors—it was observed that collisions between packets were much more common. Retransmissions were frequent on the lower data rates and at the higher data rates, no collisions were detected. Figure 19 shows a transmission packet acknowledgment error *COLLISION_PACKET* after an attempted confirmed uplink transmission.

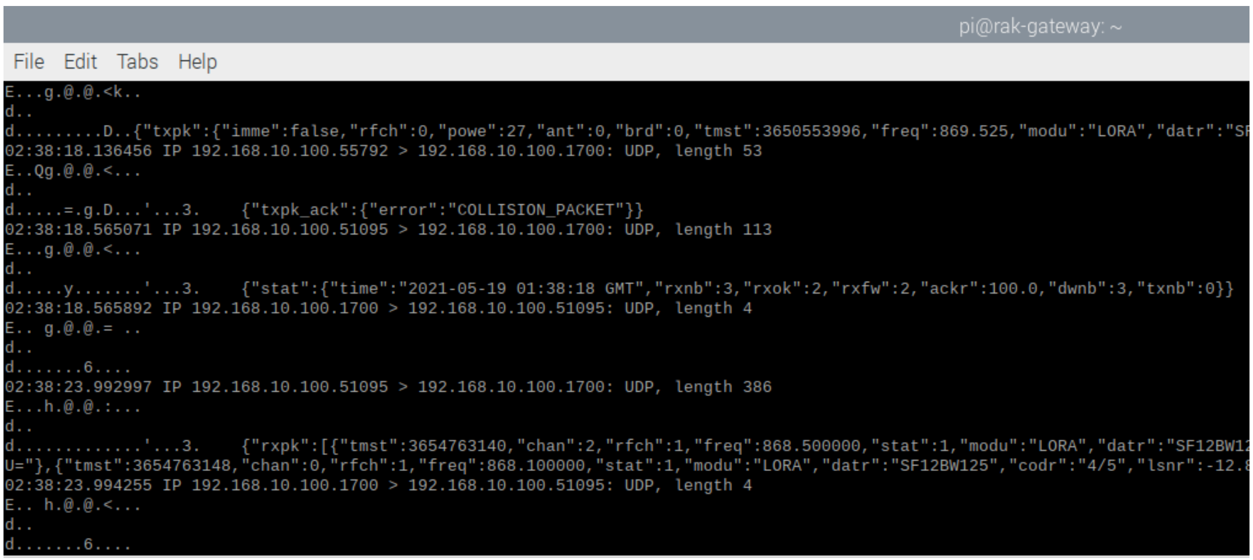

Figure 19: Terminal screenshot of observed collisions of packet transmissions using DR0.

6.1.2 Comparing TX Power & DR Energy Consumption

The transmission power in LoRaWAN networks can be appropriately configured, yet in Europe when using the ISM band frequencies (863MHz – 870MHz) the max transmission power is constrained to 14dBm. Reducing transmission power will save power, however, in turn, the signal range will be lowered. To better understand how much power is drained when altering this value, we can document the tag's current consumption during a confirmed uplink. Figure 20 illustrates the recorded relationship between power consumption and transmission power alteration between 14dBm and 1dBm. The results show an expected linear decrease in power consumption as the transmission power value is lowered. Figure 21 shows the mean current draw of the various data rate adjustments during the same confirmed uplink operation. Similarly, the graph shows another linear drop in power consumption as the data rate increases, however, the gradient of the decrease is almost double that of the opposing TX power adjustments. The data shows that altering the data rates has a more significant impact on power consumption than altering the transmission power. For this study, we halt any further investigation of transmission power alterations as it is less impactful for the Class C FUOTA process. According to the designs in section 3.3, only a few initial uplink transmissions are required to configure and begin the process before the lengthier RX window opens to receive the fragmented firmware.

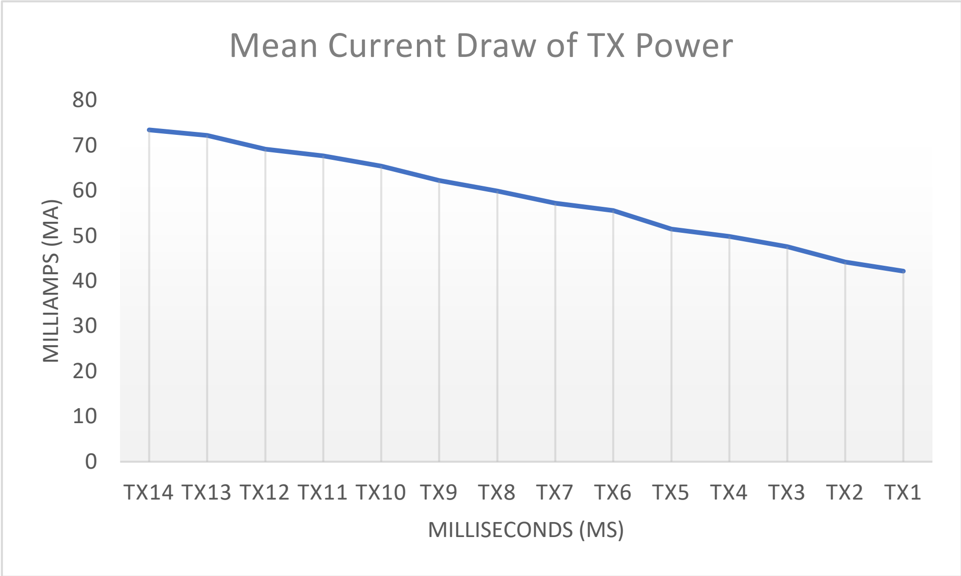

Figure 20: Average current draw of transmission power alterations between 14dBm – 1dBm.

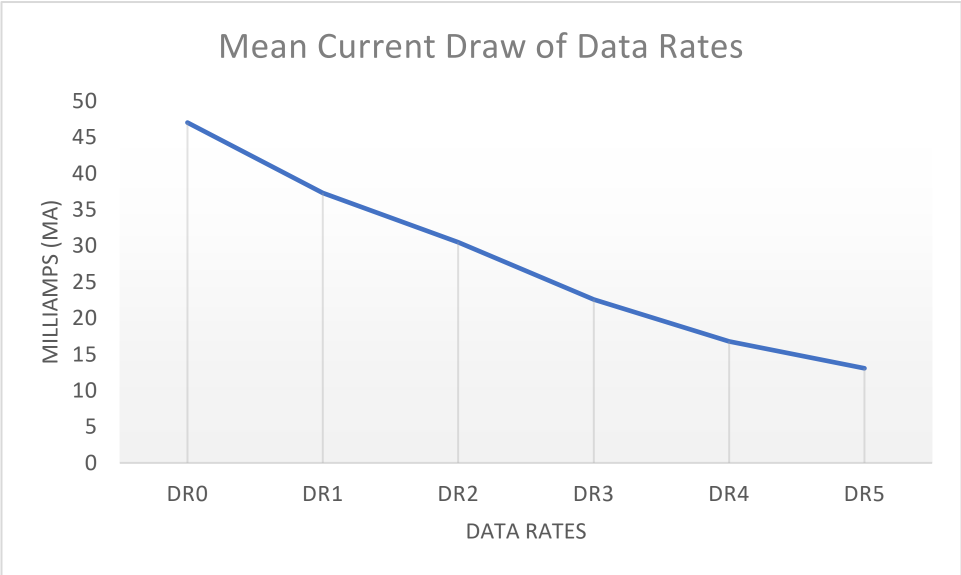

Figure 21: Average current draw of data rate adjustments between DR0 and DR5 (SF12/BW125 – SF7/BW125)

6.1.3 Summary of Frequency Parameter Testing

Using the data gathered in table 4, we can now work out a few further details of how the tag would fair when undergoing certain stages of the firmware update. In a hypothetical scenario, where the device has a battery pack with a capacity of 16000mAh, we know that the open RX window variable V_{RX} consumes 37.9mA and so we can use the following formula to determine how long the tag would be able to run during this state:

$$\textit{Operation Run Time}_{(hours)} = \frac{\textit{Battery Pack Capacity}_{(mAh)}}{\textit{Operation Current Consumption}_{(mA)}}$$

By using the values identified above and placing them within the given formula we can determine that the animal tag would be able to open its receive window for a total of 422.2 hours before draining the battery pack. Likewise, we can apply the same formula to the value of variable V_{SLEEP} to determine how long the device could operate when only running in deep-sleep mode. Once again, using the above formula and the current value of V_{SLEEP} we can calculate that the device would last for 4266 hours, which equates to 178 days. To increase the device battery life, either the battery pack capacity must be increased or the deep-sleep current consumption needs to be minimised further—in this case possibly by removing certain parts such as the OLED display or other unnecessary components that are draining power.

On a separate note, as witnessed during the 'on-the-bench' testing phase, a higher spreading factor means collisions are made more probable. This must be considered carefully as simply selecting the optimal spreading factor that yields the lowest energy cost will not necessarily be the highest or lowest, yet more likely a parameter between. If transmissions are lost and have to be excessively retransmitted—when confirmations of messages are required—the higher data rate energy savings of a quicker transmission will be removed by the need for retransmitted messages. As a result, this also uses up extra duty-cycle time of that sub-band and extends the time it would take to transmit all fragmented segments of the firmware.

To conclude, total device power consumption cannot be so simply optimized by altering a single parameter like spread factor, many other variables must be considered and investigated. Due to the limiting time constraints of this study, we are restricted to only investigate a select few. However, now that the implemented system has been initially tested and characterised, we can further our investigation of spread factor alteration and compare this against reception in a real-world scenario. LoRa claims to be able to effectively avoid interference as it has the ability to receive signals below the noise floor [51], however, this must be put to the test by investigating such claims within a natural environment with real-world obstacles and obstructions like trees and bodies of water.

6.2 Field Study

As the company aspires to administer firmware updates to the animal tags while the farmer's cattle roam around the fells, the tag must be trialled in its intended environment. Comparing the lab-tested frequency settings against how well data is received in-field will help to determine the existence of barriers and possible ways to address them. Testing the system under realistic conditions will assist in proving or disproving theoretical postulation of how the LoRa modulation is expected to behave within an outdoor environment. Additionally, empirical data of the reception of the product under certain conditions will be obtained, providing further in-depth knowledge of how device operations fair.

The field study took place just outside of Kendal—often described as the Southern gateway to the Lake District—and was conducted over an $800m^2$ area of hilly farmland. After gaining the permission of a local farmer to record measurements in the surrounding fields, a topographic map of the area was plotted to mark the permitted and unpermitted zones. The map can be found in Figure 22, which makes clear any successful and unsuccessful points of measurement, alongside obstacles such as trees, becks, and buildings—a key is provided for further detail. The gateway was set up in a family home at the highest point of the house next to an open window to limit interference and improve antenna gain. Three repeated measurements were recorded at every location and averaged for each of the five data rates; the positions are marked on the map as red circles and those with crosses received no signal on any data rate. The measurements will be an RSSI estimated power level that the RF client animal tag receives from the gateway; the signal is expected to get weaker over larger distances which leads to lower data throughput. Note that the weather on this day was clear and sunny—though it had rained an hour before—and the start time was 1:50 pm.

6.2.1 Notable Observations

There were many significant environmental effects observed during the study that are worth taking note of:

Vegetation – Trees, hedges, and other foliage are prolific in the rural countryside and it seemed to have a great effect on radio signal strength. The vegetation was significantly saturated from heavy rainfall an hour before the experiment was conducted. Perhaps the leaves covered with rain absorbed and scattered most of the signals as the behaviour of the received acknowledgements were odd and unpredictable at times. The most obvious and prominent example of this from the figure 22 map are all the failed reading points northeast of the gateway. A very dense and tall row of trees within sector *P1* seemed to completely block any signal being received past that area.

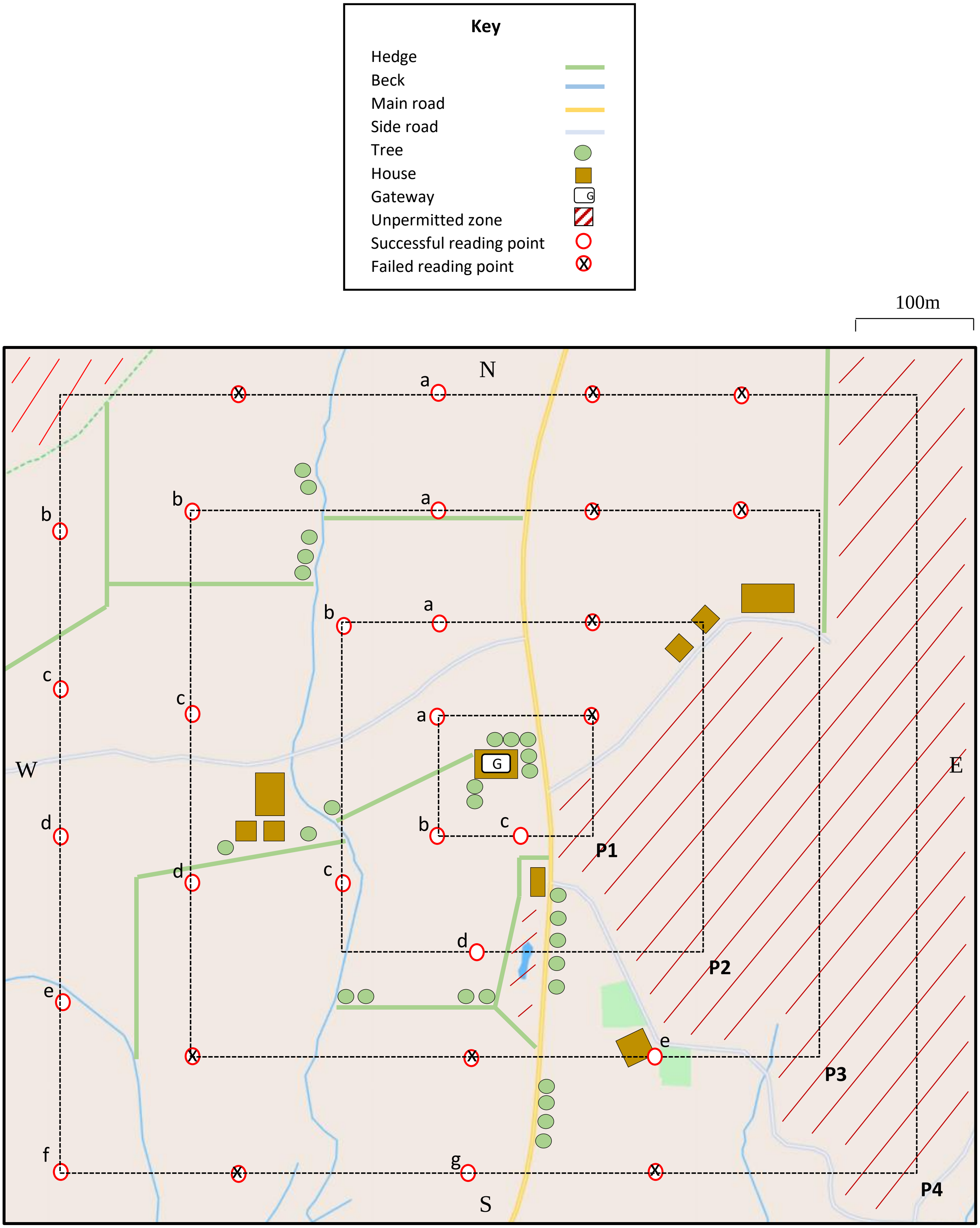

Figure 22: Topographic map of the field study environment with marked positions of measurement and various obstructions/obstacles.

Line-of-sight – As the map shows no levels of hill elevation it makes the failed points of reading seem odd and nonsensical. To alleviate this confusion, a lot of the failed reading points marked down were due to the positions being at low altitudes and the absence of a clear path between the device antenna and gateway for radio transmission. For instance, in sector *P4* at measuring point *g* there is a large rising hill where the signal is easily received, whereas the measuring point 100 metres directly in front of that was within a declining hillslope and was unable to receive any radio signals. This can be easily explained by the 'Fresnel zone' [52], which is an ellipsoidal region of space between transmitter and receiver that propagated transmissions can travel through.

Bodies of water – The received signal strength around becks—specifically point *e* in sector *P4* and point *c* in sector *P2*—fluctuated within a range of +/-10 RSSI. This created anomalous readings at those two points in comparison to other locations; this is reflected in the results tables that can be found in the Appendix in section 4. Unfortunately, this location didn't have many bodies of water to test next to and the only water body shown in the map (30 metres right of sector *P2* point *d*) was inaccessible due to it being within the bounds of a neighbour's property.

Power lines – Though the effect was not significant, the power lines through the field, south of the gateway, between zone *P1* and *P2*, seemed to cause minor radio signal attenuation. This was more noticeable when taking measurements of the higher data rates, however, we cannot be completely certain and further investigation would be required to determine the truth of this.

6.2.2 Results

Table 5 below shows the average RSSI values for each data rate in each zone of every positional point of measurement recorded for that area. Note that the distances between the gateways and points of measurement for these areas are not completely equal.

	P1	**P2**	**P3**	**P4**
DR5	-60.5	-81.5	-93.4	-119.7
DR4	-64.5	-87.75	-97	-116.3
DR3	-69	-89.25	-107	-117.4
DR2	-71	-98.25	-109.8	-118.9
DR1	-71	-100.5	-112.4	-120.3
DR0	-73	-103	-113.2	-122.4

Table 5: Mean values of received radio signal strength for each sector and data rate.

All the RSSI data rate readings seem to gradually and linearly decrease in correlation with the distance increasing as expected. However, the average reading of *DR5* in section *P4* was distinguishably higher than anticipated. At *P4* the other readings gradually increase in line with the data rate also increasing, yet the *DR5* reading anomalously spikes. It is possible that this was caused by the data rate beginning to reach its maximum travelling distance and so the radio signal strength began to diminish more than some of its counterpart data rates. Figure 23 illustrates the relationship between data rate and distance with signal strength more clearly. The linear *P4* trendline makes the *DR5* observation more prominent through its lack of gradient in comparison to the other trendlines.

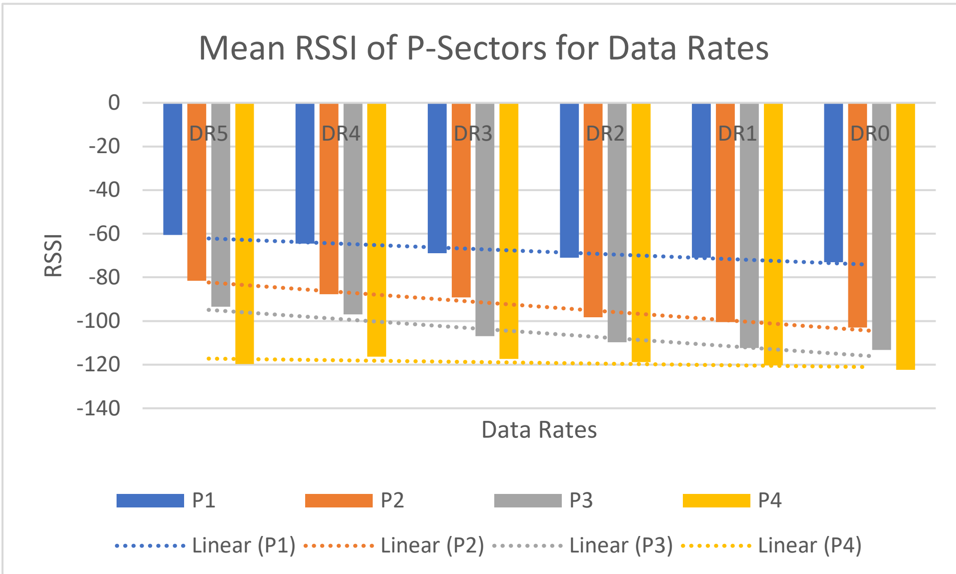

Figure 23: Trendlines highlighting the relational change in RSSI over the varying data rates and p-sectors.

6.2.3 Summary

From this study, we have found that the animal tag can be sensitive to the presence of various objects and radio signal reflectors despite the claims that the chirp spread spectrum (CSS) technology it uses is more robust against noise and signal interference [53]. The results tables listed in the Appendix (section 4) also show that there were more frequent failed readings for lower data rates which correlates with findings of the ‘on-the-bench’ testing. Seemingly, the lower data rates are more prone to interference and signal interruption by the obstacles discussed in section 6.2.1, which is a substantial cause for concern due to the fact that the product's intended environment will be much hillier and will have more vegetation and lakes to interrupt the signals. On top of this, the map in figure 22 marked a worrying 10 blind spots out of a total of 30 points of measurements, and so in this case, one-third of locations were inaccessible to the radio—mainly due to not being within the Fresnel zone’s line of sight. If this product were to be deployed in the hilly Lake District terrain then it would need an abundance of repeaters or gateways in high altitude, premeditated locations to provide signal coverage, and even then, the dense vegetation will cause further radio signal hindrances.

6.3 Testing Evaluation

To begin this section, semi-structured interviews with farmers local to the lake district are conducted and discussed to gather information about routine and the contextual needs of the animal tag. From this, we discover opportunities for addressing such product requirements and continue to evaluate the Class C model and other possible use cases and solutions.

6.3.1 Semi-Structured Farmer Interviews

Improving understanding of agricultural routine will help to better inform business and applicational decisions to be catered around already existing farming procedures. To achieve this, several interviews with farmers were orchestrated to extract qualitative data to aid this evaluation. The semi-structured interview format allows the farmer to openly express any concerns they have while allowing the conversation to be diverted to accommodate inquiries of varying subject matter. This technique allows the interviewer to delve into the reasoning behind answers to open-ended questions and encourages two-way communication. Idealistically, the collected agricultural practice data could propel this evaluation forward and provide solutions to the most power and time-efficient manner in which to apply firmware updates. The interview questions contain additional queries intended to help inform Milliamp of other farming scenarios that could benefit from their product. The interview questions and answers can be found in the Appendix from sections 2 - 3.

After organising three separate interviews with local farmers—all of which turned out to be pastoral farmers—the following takeaway points in relation to firmware updates, cattle location, and extended periods in which to apply updates, were made most prominent:

Farmer A importantly mentioned that "Cows are held within an acre-sized field from May to September". Additionally, they pointed out that activities like milking and sheering didn't take more than a few hours to complete. This farmer also discussed how their cattle tended to "flock around and underneath trees to protect themselves from the sun and rain".

Farmer B showed concern about the product's prosperity in regard to reception, they said that – "The Lake District is notorious for its terrible reception so the product would need good range and coverage for all areas if it is to work effectively." They also mentioned that milking activities took three to four hours to complete which correlates with what Farmer A stated.

Farmer C informed us of the practice of 'set stocking' in which farmers leave animals within a field for a long period of time—two to three months in the summer—which matches with Farmer A's comment.

From these comments, we can extrapolate that for pastoral farming there are prolonged periods of time in which livestock are held within a single area during ordinary agricultural routine. Perhaps remotely administering firmware updates through the fells is not the best approach and instead taking advantage of an already existing routine could provide a better-suited location and time to apply the updates. Farmers A's comments of cattle using trees for shelter, alongside the field study results that confirm rain and trees significantly impact radio signal, further exacerbate the difficulties of the remote update approach.

6.3.2 Evaluating the Class C Model Approach

Although a Class C FUOTA application was not developed for the animal tag in time, expectations of how the model operates based on theory and documentation can be used to apply the collected data towards. To recapitulate, the data rate parameter testing phase showed that the higher spread factors of the lower data rates caused packet collisions to be more frequent, and this loss of data resulted in additional consumption of duty-cycle time. On top of this, the field study results reinforced these findings with its field data showing an inability to receive confirmation messages when next to certain interfering obstructions. The field testing's problematic findings can be encapsulated by three main factors: vegetation, water, and line-of-sight.

From the data this study has unearthed, it is hard to see an effective, working Class C FUOTA operation being successful when applying updates while the livestock are roaming the fells. The Class C model opens its receive window endlessly while waiting to receive fragmented firmware as shown in figure 1. At higher spreading factors, the current draw—due to the duration of the V_{RX}'s operations—was substantial, the scenario given in section 6.1.3 showed that at *DR0* the device would only be able to receive data for 422.2 hours before being completely drained of power. Testing showed that all data rates drew the same amount of current, yet for different durations of time. Furthermore, the lower data rate collision possibilities would only extend that duration of power consumption for longer.

As such, the best and most feasible solution to reduce power consumption and functionality would be to update the devices on the highest possible data rate for the shortest period of time during the receiving stages of the update. Although this is not the manner of operation desired by the company, it is the most effective way to apply updates using the Class C model. The additional research gained from interviewing the farmers, however, does provide hope for a possible effective compromise. Agricultural routine provides periods of opportunity in which to apply updates while certain activities are being conducted. Depending on how long an update takes, shearing and milking activities do provide three to four hours in which to apply them. Additionally, if even more time is needed, the 'set-stocking' procedure ensures that livestock are held within an area of six to twenty acres for a few months. By collaborating with farmers to ensure livestock are within range, and providing good coverage over a selected field, updates could be swiftly and effectively applied on the more preferable data rates of *DR5* to *DR2*—as collisions are less common at these data rates and the received radio signal is stronger. However, vegetation and weather would still cause significant interference and so if possible, the most appropriate and power-friendly option would be to apply the update during ordinary farming operations like shearing and milking.

6.3.3 Evaluating the Potential Use Case of a Class B Model Solution

In section 2.2.4, another Class B model approach was discussed based on information acquired from related literature. This operational mode removes the Class C endless V_{RX} receive window and instead uses time synchronised periodic uplinks for each device in a multicast group. As a result, the V_{SLEEP} duration of the animal tags—using correctly optimised frequency parameters—has the potential to be massively increased. Nevertheless, one must take into consideration the additional cost of V_{TX} transmission power, which was significantly greater than the V_{RX} power consumption. The tests recorded a 70.8 average current draw while the device was transmitting a confirmed uplink. Contrariwise, depending on the periodicity of required synchronous uplinks, the transmissions may use more power overall than each device having a constant receive window listening. To determine if this is true or false, further investigation of the models, their power use, and behaviour should be conducted in the same manner as this study. Applying the same tests to each model would help to reveal characteristics and behaviour of a FUOTA update procedure within multicast groups.

Another factor to take into consideration is that the synchronous update model would greatly increase overall update time for the devices as a group. Potentially, this could create conflict with agricultural routine times mentioned by the farmers depending on the overall time update time taken. However, as before, the 'set-stocking' practice maintains its potential position as a conduit for updates if more time is required. Additionally, the model's synchronous nature and periodic communication present an opportunity for further device data processing, which in return could be used to optimise how data is being sent to the tag. Moreover, problems of mobility mentioned in N. Benkahla's study [35] could be addressed by the Class B model by adjusting frequency parameters and even predicting device movement based on retrieved GPS data through periodic uplinks. The Class B model clearly has potential for certain applicational use cases, however, more needs to be studied and understood before its application can be accurately considered.

6.3.4 Summary of Overall Findings

To briefly summarise, the main findings of this study were that CSS modulation is more prone to outdoor interference than theoretically advertised, especially when listening for lower data rate transmissions. The main difficulties and obstacles of outdoor LoRa radio transmission, specifically, in rural hilly terrain, have been highlighted. To solve this, possible farming procedure collaboration has been discussed that could potentially complement the same timing requirements for firmware updates, while reducing distance and power use. In addition to this, an empirical power consumption profile in relation to data rate alterations has been developed for the animal tag hardware.

Chapter 7

Conclusion

To conclude the study, this section reviews the project aims set in the introductory chapter to discuss their successes or shortcomings. The project itself is reviewed and revisions are suggested, followed by a deliberation of future work, lessons learned, and a final closing statement.

7.1 Review of Project Aims

Research and configure any remaining LoRaWAN infrastructure required to implement the FUOTA process for the tag.

Figure 4 in chapter 3.2 illustrates an extensive and fully constructed LoRaWAN infrastructure. This was already partially complete at the beginning of this study; however, the gateway component was missing. The system in operation chapter shows example terminal output of the configured RAK833 gateway which evidences completion of this objective.

Design and implement the FUOTA multicast deployment mechanisms.

This objective was only partly completed. Section 3.3 specifies designs according to three separate LoRaWAN specification sheets and depicts the nature of the message flow between the animal tag and the application server for a FUOTA multicast operation. However, due to time constraints, the implementation of the FUOTA application was not completed for the tag. Better management of time and more in-depth prior research into available ported libraries could have improved the likelihood of success for implementing the FUOTA multicast set up on the animal tag.

Conduct an assessment and investigation of the tag, measuring power consumption, and update time metrics.

Chapter 6.1.1 provides a detailed power consumption profile of the tag during the 'on-the-bench' system tests for a confirmed uplink procedure. For update time, as the FUOTA model was not implemented, this metric could only be discussed rather than empirically characterised. However, a scenario is provided, and update time capacity is predicted for updates based on the V_{RX} receive window current draw. Though the objective was not completed in the desired way, it was dynamically assessed from an alternate angle, showing a more versatile approach to the problem at hand.

Analyse network performance by measuring RF strength during a FUOTA operation to the tag in a natural outdoor environment.

In section 6.2 a field study was conducted within a farmer's fields on the outskirts of the Lake District. Although the radio signal strength was not tested during a FUOTA operation, the testing tools developed were proficient enough to simulate receiving packets in a similar fashion. Field notes provided recordings of notable observations and the data results correlated with the previous tests, improving the overall validity of the findings and fulfilling the aims of this objective.

Evaluate the tag's performance, update efficiency, and power consumption to determine if the model can effectively operate for its application context.

In section 6.3.1, semi-structured interviews with farmers are conducted to discover opportunities and a greater understanding of how agricultural routine and the contextual needs of the animal tag could symbiotically operate. Throughout the entirety of the chapter 6.3 testing evaluation, reference is persistently made to the developed power consumption profile of the tag and applied in the discussion. Predicted update efficiency based on altering parameters for transmission is discussed and the tag RF performance from the field results in section 6.2.3 is deliberated over to reflect on issues found. During the evaluation, potential solutions are contemplated and suggested, which with all things considered, concludes an overall positive result for this project's aim.

Discuss the results and deliberate over the possibility of a LoRaWAN Class B model use case.

Finally, in chapter 2.2.4 a possible solution of the LoRaWAN Class B model to address problems discovered in the relevant literature is discussed. In section 6.3.3, its potential application to the project is debated and reflected on based on the newly acquired data. We conclude that for certain use cases, it has the potential to reduce power consumption, yet increase update time, and conclude that further detailed inquiry is necessary to determine its applicability to this project.

To briefly conclude, throughout the entirety of this study, the research question has been kept in sight by persistently evaluating and testing the main metrics of power consumption, relative performance, and update time. Though the FUOTA model was not implemented in time, I still deem this project a huge success. A great amount of information has been acquired that will help towards building, improving, and boosting the potential of Milliamps animal tag product, which in turn may help to sustain the Lake District World Heritage Site.

7.2 Suggested Project Revisions

There is a fair amount of revision that could be made to this project, had I the time or resources to do so. The most obvious revision would be testing an actual implementation of the Class C FUOTA operation on the animal tag. Regarding the field testing, to further validate the findings, it would have been beneficial to complete the experiment on different days, with different weather, temperature, and wind speeds, to observe their impact on the animal tag's reception. Moreover, it would have been better to test the device over a much larger area to see the cut-off ranges of varying data rates.

For the interviews, the main drawback was that the farmers were all dairy cattle farmers rather than upland hill farmers. It would have been much more insightful to have interviewed and spoken to farmers of the products targeted profession. Social desirability and reporting biases may have been present during the interviews due to some of the sensitivity of the questions being asked. For instance, interviewees may have preferred not to reveal if their cattle had been commonly sick or their business needed technological assistance. Factoring out such confounds by sample randomisation or by more carefully catering the questions would assist in enhancing the project's data and overall outcome.

7.3 Future Work

As the research area of FUOTA over LoRaWAN is relatively new, there exists a plethora of exploratory avenues yet to explore as future work. The most pronounced future development would be the implementation of both a Class C and Class B animal tag FUOTA application to then test under the same environmental conditions. It would also be interesting and beneficial to examine alternate configurations of different transmission parameters—like bandwidth—over said implementations to help characterise the system further.

Beyond this, improved forms of time synchronisation between the animal tags and the application server could be explored to see if it can further optimise the system or reduce power use. Studies have explored distributed self-stabilising clock synchronisation through biomimetics of Asian fireflies [54], which has been shown to enhance the lifetime of nodes and could be a potential point of interest for a product of this kind. In addition, GPS-free geolocation over LoRaWAN using gateways for triangulation, rather than having a GPS module could help to reduce power use and lower hardware costs. Other technologies could be used to further improve the capabilities of such a system, like machine learning techniques like Support Vector Machine, Decision Tree, or Naïve Bayes to combine signal time differences of arrival to the gateway and RSSI measurements to improve the accuracy of geolocating nodes.

7.4 Lessons Learned

In reflection, the difficulties and challenges encountered over the course of this project have helped to develop technical knowledge and character. In terms of technical development, I have gained experience with managing docker containers and orchestrating an entire network of interconnected systems. On top of this, I was introduced to a new programming language—Golang—and learned a great deal about IoT, LoRaWAN, and the ESP32 series architecture.

Most importantly, I have been taught a vital lesson in time management, coping with an incredible workload, understanding how much I can accomplish within a given time frame, and admitting when I am stuck and need to ask for help. Communication is key to progression and without it, advancement is slow and often misdirected.

7.5 Final Comments

Overall, throughout this project, I have learned and developed an abundance of new skills, both interpersonal and technical. It has been incredibly challenging and frustrating at times, yet through trial and tribulation, I have persevered and overcome any difficulties faced. Although I did not manage to complete every objective, a lot of data was found, and progress was made towards the company’s project and its goals. It has been delightful to have played a part in the development of a product that has the potential to greatly impact the revival of culture and maintenance of a World Heritage Site and its natural wonders.

References

[1] L. Philip, C. Cottrill, J. Farrington, F. Williams and F. Ashmore, “The digital divide: Patterns, policy and scenarios for connecting the‘final few’in rural communities across Great Britain,” Elsevier, Aberdeen, 2017.

[2] L. A. T. Committee, “LoRaWAN(tm) 1.0.3 Specification,” 2018. [Online]. Available: https://lora-alliance.org/wp-content/uploads/2020/11/lorawan1.0.3.pdf. [Accessed 14 March 2021].

[3] D. Harvey, N. Thompson, C. Scott and C. Hubbard, “Farming & Farm Forestry in the Lake District,” A Report for the Lake District National Park Partnership, Farming & Forestry Task Force, Newcastle, 2013.

[4] T. Knight-Jones and J. Rushton, “The economic impacts of foot and mouth disease - What are they, how big are they and where do they occur?,” *Preventive Veterinary Medicine,* vol. 112, no. 3-4, pp. 161-173, 2013.

[5] Avnet, “Internet of Things: Low Power, Low Cost Connected Devices fuel demand for Microco,” Avnet, 25 February 2017. [Online]. Available: https://www.avnet.com/wps/portal/us/resources/article/internet-of-things-low-power-low-cost-connected-devices-fuel-demand-for-microco/. [Accessed 10 March 2021].

[6] P. Wadhwani and S. Loomba, “Low Power Wide Area Network Market Size by Component,” Global Market Insights, Selbyville, Delaware, 2021.

[7] V. D. Mea, M. H. Popescu, D. Gonano, T. Petaros, I. Emili and M. G. Fattori, “A Communication Infrastructure for the Health and Social Care Internet of Things: Proof-of-Concept Study,” JMIR Med Inform, 2020.

[8] K. Mekki, E. Bajic, F. Chaxel and F. Meyer, “A comparative study of LPWAN technologies for large-scale IoT deployment,” *ICT Express,* vol. 5, no. 1, pp. 1-7, 2019.

[9] R. Sanchez-Iborra, I. G. Liaño, C. Simoes, E. Couñago and A. F. Skarmeta, “Tracking and Monitoring System Based on LoRa Technology for Lightweight Boats,” *Electronics,* vol. 8, no. 1, p. 15, 2019.

[10] C. Kolias, G. Kambourakis, A. Stavrou and J. Voas, “DDoS in the IoT: Mirai and Other Botnets,” *Computer,* vol. 50, no. 7, pp. 80-84, 2017.

[11] T. T. Network, “LoRaWAN Duty Cycle,” The Things Network, 2021. [Online]. Available: https://www.thethingsnetwork.org/docs/lorawan/duty-cycle.html#maximum-duty-cycle. [Accessed 15 March 2021].

[12] T. T. Industries, “Gateways,” The Things Network, 2021. [Online]. Available: https://www.thethingsnetwork.org/docs/gateways/index.html. [Accessed 22 April 2021].

[13] M. Pule and A. M. Abu-Mahfouz, “Firmware Updates Over the Air Mechanisms for Low Power Wide Area Networks: A Review,” in *International Multidisciplinary Information Technology and Engineering Conference (IMITEC),* Vanderbijlpark, 2019.

[14] J. Jongboom and J. Stokking, “Enabling firmware updates over LPWANs,” in *Embedded World Conference,* Amsterdam, 2018.

[15] P. G. Neumann, Computer-Related Risks and the National Infrastructures, NY: ACM Press.Addison-Wesley Publishing Co., January 1995.

[16] D. Burke, “All Circuits are Busy Now: The 1990 AT&T Long Distance Network Collapse,” November 1995. [Online]. Available: https://users.csc.calpoly.edu/~jdalbey/SWE/Papers/att_collapse.html. [Accessed 26 April 2021].

[17] D. Goodin, “Record-breaking DDoS reportedly delivered by >145k hacked cameras,” arstechnica, 29 September 2016. [Online]. Available: https://arstechnica.com/information-technology/2016/09/botnet-of-145k-cameras-reportedly-deliver-internets-biggest-ddos-ever/. [Accessed 27 April 2021].

[18] S. Nappo, “The internet of things devoid of comprehensive security management,” Goodreads, 22 April 2020. [Online]. Available: https://www.goodreads.com/quotes/10043326-the-internet-of-things-iot-devoid-of-comprehensive-security-management. [Accessed 27 April 2021].

[19] A. Anastasiou, P. Christodoulou, K. Christodoulou, V. Vassiliou and Z. Zinonos, “IoT Device Firmware Update over LoRa: The Blockchain Solution,” in *2020 16th International Conference on Distributed Computing in Sensor Systems (DCOSS),* Neapolis, 2020.

[20] C. Ioannou and V. Vassiliou, “Security Agent Location in the Internet of Things,” *IEEE Access,* vol. 7, pp. 95844-95856, 2019.

[21] J. P. Queralta, T. N. Gia, Z. Zou, H. Tenhunen and T. Westerlund, “Comparative Study of LPWAN Technologies on Unlicensed Bands for M2M Communication in the IoT: beyond LoRa and LoRaWAN,” Elsevier, 2019.

[22] J. Stokking, “Firmware Updates over Low-Power Wide Area Networks,” The Things Network, 7 June 2017. [Online]. Available: https://www.thethingsnetwork.org/article/firmware-updates-over-low-power-wide-area-networks. [Accessed 27 April 2021].

[23] L. Dolan, “LoRa Alliance's New LoRaWAN Protocol Specifications to Support FUOTA,” Embedded Computing Design, 25 October 2018. [Online]. Available: https://www.embeddedcomputing.com/application/networking-5g/lora-alliance-s-new-lorawan-protocol-specifications-to-support-fuota. [Accessed 27 April 2021].

[24] T. T. Industries, “LoRaWAN Security,” The Things Network, 2021. [Online]. Available: https://www.thethingsnetwork.org/docs/lorawan/security/index.html. [Accessed 28 April 2021].

[25] M.-T. Systems, “MultiTech xDot® Long Range LoRa® Module Datasheet,” Multi-Techn Systems, 2019.

[26] B. Martin, U. Roedig, T. Voigt and J. M. Alonso, “Do LoRa Low-Power Wide-Area Networks Scale?,” ACM, Lancaster, 2016.

[27] Link-Labs, “Low Power Wide Area Network (LPWA),” Link-Labs, 28 June 2018. [Online]. Available: https://www.link-labs.com/blog/low-power-wide-area-network-lpwa. [Accessed 29 April 2021].

[28] G. Hassan and H. S. Hassanein, “MoT: A Deterministic Latency MAC Protocol for Mission-Critical IoT Applications,” in *14th International Wireless Communications & Mobile Computing Conference (IWCMC)*, Limassol, Cyprus, 2018.

[29] D. Patel and M. Won, “Experimental Study on Low Power Wide Area Networks (LPWAN) for Mobile Internet of Things,” arXiv, 2017.

[30] D. Patel, “Low Power Wide Area Networks (LPWAN): Technology Review and Experimental Study on Mobility Effect,” Electronic Theses and Dissertations, 2018.

[31] B. Singh and B. Kaur, “Comparative study of Internet of Things Infrastructure and Security,” Global Wireless Submit, Aarhus, Denmark, 2016.

[32] L. Labs, “LL-RPT Repeaters,” Link Labs, 2021. [Online]. Available: https://www.link-labs.com/documentation/ll-rpt-repeaters. [Accessed 1 May 2021].

[33] M. Konstantin, S. Martin, M. Pavel, F. Radek, M. Radek, A. Sergey and H. Jiri, “On the Performance of Multi-Gateway LoRaWAN Deployments: An Experimental Study,” in *2020 IEEE Wireless Communications and Networking Conference (WCNC)*, 2020.

[34] T. T. Network, “CayenneLPP,” The Things Network, 2021. [Online]. Available: https://www.thethingsnetwork.org/docs/devices/arduino/api/cayennelpp/index.html. [Accessed 2 May 2021].

[35] B. Norhane, T. Hajer, S. Ye-Qiong and F. Mounir, “Enhanced ADR for LoRaWAN networks with mobility,” in *15th International Wireless Communications & Mobile Computing Conference*, Tangier, Morocco, 2019.

[36] P. Marco, S. Emiliano, F. Paolo, R. Stefano, D. Alessandro, B. Paolo, D. G. Davide and F. Alessandra, “Evaluation of the Use of Class B LoRaWAN for the Coordination of Distributed Interface Protection Systems in Smart Grids,” *Journal of Sensor and Actuator Networks,* vol. 9, no. 1, p. 13, 2020.

[37] T.-F. J. S.-B. David, S.-P. Victor and P.-M. Ana, “Analysis of Bidirectional ADR-Enabled Class B LoRaWAN Networks in Industrial Scenarios,” *Applied Sciences,* vol. 10, no. 22, p. 7964, 2020.

[38] F. Joseph, B. Stephen and F. Ronan, “Evaluating the Scalability of LoRaWAN Gateways for Class B Communication in ns-3,” in *IEEE conference of Standards for Communications and Networking (CSCN),* Paris, France, 2018.

[39] T. Pie, M. Xiaoyuan, A. B. Carlo, L. Ye, Y. Fengxu, T. Xin, L. Dan and W. Jianmin, “ChirpBox: An Infrastructure-Less LoRa Testbed,” in *International Conference on Embedded Wireless Systems and Networks (EWSN),* Delft, Netherlands, 2021.

[40] O. Brocaar, “ChirpStack, open-source LoRaWAN Network Server Stack,” ChirpStack, 2021. [Online]. Available: https://www.chirpstack.io/. [Accessed 12 May 2021].

[41] “Golang,” Google, 2021. [Online]. Available: https://golang.org/. [Accessed 12 May 2021].

[42] S. Corporation, “Lora-net/LoRaMAC-node,” Github, 2021. [Online]. Available: https://github.com/Lora-net/LoRaMac-node. [Accessed 12 May 2021].

[43] IEEE, “EEE 802.15.4 - IEEE Standard for Local and metropolitan area networks--Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs),” IEEE, 2011.

[44] Semtech, “LoRaMac-doc-v4.5.1,” Semtech, 2021. [Online]. Available: http://stackforce.github.io/LoRaMac-doc/LoRaMac-doc-v4.5.1/index.html. [Accessed 12 May 2021].

[45] FUOTA Working Group of the LoRa Alliance Technical Committee, “LoRaWAN Application Layer Clock Synchronization Specification,” LoRa Alliance™, 2018.

[46] FUOTA Working Group of the LoRa Alliance Technical Committee, “LoRaWAN Remote Multicast Setup Specification,” LoRa Alliance™, 2018.

[47] F. W. G. o. t. L. A. T. Committee, “LoRaWAN Fragmented Data Block Transport Specification,” LoRa Alliance™ , 2018.

[48] MKROE, “Vct Monitor Click,” MKROE, 2021. [Online]. Available: https://www.mikroe.com/vct-monitor-click. [Accessed 13 May 2021].

[49] Lancaster University, “microbit-dal,” Github, October 2019. [Online]. Available: https://github.com/lancaster-university/microbit-dal. [Accessed 23 May 2021].

[50] Analog Devices, “LTC2990 - Quad I2C Voltage, Current and Temperature Monitor,” 2018. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/LTC2990.pdf. [Accessed 22 May 2021].

[51] T. Cooper, “The Immune System | Avoiding RF Interference with LoRa,” Semtech, 2021. [Online]. Available: https://lora-developers.semtech.com/library/tech-papers-and-guides/interference-immunity/#:~:text=While%20this%20is%20the%20worst,below%20interference%20(Figure%209).. [Accessed 30 May 2021].

[52] E. B., “LoRaWAN Quickguide,” Read the Docs, 2018. [Online]. Available: https://lora.readthedocs.io/en/latest/. [Accessed 30 May 2021].

[53] Semtech, “LoRa and LoRaWAN,” Semtech, 2021. [Online]. Available: https://lora-developers.semtech.com/library/tech-papers-and-guides/lora-and-lorawan/. [Accessed 31 May 2021].

[54] L. Robert and E. Wilfried, “Firefly Clock Synchronization in an 802.15.4 Wireless Network,” *EURASIP Journal on Embedded Systems,* p. 17, 2014.

Appendix

1 Code links

https://github.com/BenjaminMcKitterick/4th-Year-Project

2 Semi Structured interview questions

1. Please provide a brief description of the type of farming you carry out.

2. During your ordinary agricultural routines, how long is livestock held within a single area for? —no larger than a 250m radius. This could be for any kind of activity such as milking or shearing.

3. If livestock is held within a single area, how often does this occur?

4. Do you think it would be beneficial to your business to be able to detect and prevent any disease outbreaks before they occur?

5. Have you ever had an outbreak of a particular illness or disease that has resulted in the death of livestock?

6. If you answered yes to the above question, then how often do your livestock become sick?

7. If your farm does not focus specifically on pastoral hill farming, do you think a product of this kind could benefit your business? If so, please explain how.

8. What one thing would be of most value in making your farming operations more profitable?

9. Finally, do you have any suggestions or general comments that you would like to share that could improve how a product of this kind could be applied.

3 Interview Answers

3.1 Farmer A

Q1 – “Agriculture and diary, 21 years milking cows.”

Q2 – “Cows are held within an acre sized field from May to September. Don't do any shearing. Milking doesn't take that long, and then they get moved back to the pen again.”

Q3 – “All the time. They come inside from the outside during the summer months.”

Q4 – “Yes, predicting outbreaks would save a lot of livestock far in advance, but cows are often quite well looked after and always monitored. It’s not that common of an occurrence and on paper doesn't improve the outcome by very much. Livestock is also monitored by outside parties as well as the farmer himself who can also administers their own medical substance.”

Q5 – “Yes only with chickens, mites on chicken legs which causes roughness and for them to be affected, but this can be treated with oil and cream to suffocate the mites, might benefit if it can detect that happening before.”

Q6 – “Mainly the only problems that we got on the farm were a lot of cows having a calcium deficiency and not being able to get up again, but this is never a problem and they can always inject more calcium into a vein to get them back up.”

Q7 – “No.”

Q8 – “Keeping up with the latest tech on the farm, only in the sense that they know it will pay itself back overtime and won't be too expensive, basically just tech improvements to make the farm more profitable, but this could be anything from better farming equipment (Automation through tractors), to better dietary monitoring for cows or better ways of keeping the product fresh.”

Q9 – “Not so sure about listening to cows mooing to understand how they are feelings, normally cow’s sounds only change while they're distressed about their calf being taken.”

Additional comments:

- “On sunny days cows and sheep tend to flock around and underneath trees to protect themselves from the sun”

3.2 Farmer B

Q1 – “Dairy, beef & sheep farming. A mixture of livestock.”

Q2 – “3-4 hours for milking, shearing, dosing, freeze branding and scanning.”

Q3 – “Twice a day, 7 days a week for milking cows. Other mentioned activities occur every few weeks or months.”

Q4 – “Yes.”

Q5 – “No.”

Q6 – “N/A.”

Q7 – “I am a little uncertain if a product like this would benefit my business specifically.”

Q8 – “Fertility detection and monitoring livestock would be beneficial to my farm.”

Q9 – “The Lake District is notorious for its terrible reception so the product would need good range and coverage for all areas if it is to work effectively.”

3.3 Farmer C

Q1 – “Beef, sheep, and lamb farming. Store and fattening cattle.”

Q2 – “Stay within a field the size of six to twenty acres. It’s known as ‘set stocking’ where we leave a set number of animals in a field for a long period of time.”

Q3 – “This usually lasts for around two to three months during the summer periods.”

Q4 – “A lot of illness outbreaks happen when the cattle are kept inside over the winter months. It seems to be more common for larger farms with more cattle as it’s a scale issue. Highly bred breeds are also more vulnerable.”

Q5 – “Yes, during Spring we have had trouble with the Nematodirus battus worm in lambs.”

Q6 – “Every few years this can happen suddenly.”

Q7 – “Well, the government has recently been encouraging and subsidizing rewilding sustainability projects and I do think that your animal tag product could be beneficial for tracking and monitoring those over a large area of wild land.”

Q8 – “Just by increasing price for produce.”

Q9 – “The product will definitely be able to serve a useful and beneficial role in certain farming situations.”

Additional comments:

- "The tag casing needs to be designed well so that it doesn't get caught on fencing etc."
- "Perhaps the tag could be better located around the ankle. The tag being on the neck or ear will be more likely to get caught on something."
- "GPS would be useful for tracking down animals if they ever manage to escape and wonder off the farm."

4 Field Study Results Tables

RSSI readings of data rates for position one (P1).

Data Rate	**Position One (P1)**		
	a	**b**	**c**
DR0	-71	X	-75
DR1	-69	X	-73
DR2	-69	-90	-73
DR3	-68	-89	-70
DR4	-65	-73	-64
DR5	-60	-72	-61

RSSI readings of data rates for position two (P2).

Data Rate	**Position Two (P2)**			
	a	**b**	**c**	**d**
DR0	-98	-101	X	-110
DR1	-96	-99	-101	-106
DR2	-98	-98	-92	-105
DR3	-92	-95	-83	-87
DR4	-94	-81	-91	-85
DR5	-89	-75	-80	-82

RSSI readings of data rates for position three (P3).

Data Rate	**Position Three (P3)**				
	a	**b**	**c**	**d**	**e**
DR0	-112	-124	-106	-110	-114
DR1	-110	-121	-105	-111	-115
DR2	-104	-123	-102	-110	-110
DR3	-105	-120	-100	-103	-107
DR4	-89	-110	-92	-95	-99
DR5	-92	-101	-89	-92	-93

RSSI readings of data rates for position four (P4).

Data Rate	**Position Four (P4)**						
	a	**b**	**c**	**d**	**e**	**f**	**g**
DR0	-116	-125	-123	-120	-124	-126	-123
DR1	-114	-121	-119	-120	-123	-124	-121
DR2	-111	-120	-120	-119	-115	-125	-122

DR3	-110	-122	-120	-115	-110	-125	-120
DR4	-109	-115	-119	-118	-114	-119	-120
DR5	-117	-114	-120	-125	-123	-120	-119