Università degli Studi di Padova

Dipartimento di Ingegneria dell'Informazione

University of Padova

Department of Information Engineering

Bachelor thesis in
Computer Engineering

An LPWAN MAC protocol for agricutural applications

Supervisor:
Prof. Stefano Tomasin

Candidate:
Giulio Codutti
2008795

Year 2022/2023

Abstract

Low Power Wide Area Networks (LPWANs) are getting very popular these days in Internet of Things (IoT) applications. This technology is a milestone of the 4^{th} industrial era thanks to the value that it brings to the automation processes.
This thesis has the objective to propose a Media Access Control (MAC) protocol for LPWANs called Bacco. It is built upon the LoRa physical layer protocol and it is targeted specifically at agricultural applications, where achieving both high power efficiency and long-distance communication is crucial due to the lack of reliable power sources and Internet coverage.
As well as describing the protocol in detail, Bacco is also compared to the general purpose and industry-standard LoRaWAN.

Sommario

Le reti Low Power Area Network (LPWAN) stanno prendendo piede oggigiorno nel mondo dell'Internet of Things (IoT). Questa tecnologia è un caposaldo dell'industria di quarta generazione, grazie al grande contributo che porta ai processi di automazione.
Questa tesi ha l'obiettivo di proporre un protocollo Media Access Control (MAC) per reti LPWAN, chiamato Bacco. Quest'ultimo è costruito basandosi sul protocollo a livello fisico LoRa e si rivolge principalmente ad applicazioni in ambito agricolo, dove è cruciale raggiungere un'alta efficienza energetica e un ampio raggio di comunicazione, data la mancanza di fonti energetiche affidabili e di copertura da parte della rete Internet.
Oltre a una descrizione dettagliata di Bacco, esso viene anche comparato al noto LoRaWAN, che è proposto al mercato come protocollo standard per le applicazioni che fanno uso della tecnologia LoRa.

Glossary

GSM global system for mobile communications, see [1] for more information.

LTE long term evolution, 4th generation mobile communication standard, see [2] for more information.

FTP file transfer protocol, see [3] for more information.

VHF very high frequency, it refers to the radio frequency band between 30 and 300 MHz.

TTN the things network, an open community of LoRaWAN gateways.

LoRaWAN lora wide area network, it refers to an open protocol built on top of LoRa.

IoT internet of things.

PHY physical layer protocol.

MAC media access control.

ISO/OSI open system interconnection.

LPWAN low power wide area network.

RSSI receive signal strenght indicator.

SNR signal to noise ratio.

CAD channel activity detection.

CSMA carrier sense multiple access.

SYN sync message.

SYN/ACK sync acknowledge message.

ACK acknowledge message.

ACKACK double acknowledge message.

MCU micro controller unit.

IQ in phase/quadrature.

ToA time on air.

ERP effective radiated power.

CRC cyclic redundancy check.

Contents

Introduction

Agricultural technologies have been a source of innovation since the dawn of humanity. The quest for more efficient food production has driven significant research, and above all, automation stands out as a paramount achievement in the modern world. One of the tools used to improve the automation process is internet of things (IoT), a paradigm that refers to the interconnection of physical devices in a diffused network in which each device collects valuable and localized information that is used to improve efficiency, productivity, and decision making. Many technologies implement the concepts described by IoT using disparate approaches, furthermore, various architectures describe the interaction between them. The most basic model proposes a stack composed of three layers:

- a Perception layer, that collects data from sensors;
- a Network layer, that connects devices to falcilitate the exchange of information;
- an Application layer, that processes the data and makes it available.

The Network layer can be split further according to the open system interconnection (ISO/OSI) model, in order to distinguish the purposes of the communication protocols. Figure 1 gives a representation of the stack and its components. The lowest layer of the stack is called physical layer, it takes care of electrical, mechanical and procedural interfaces that directly concern the transmission medium. There resides LoRa, a protocol that has proven itself to be one of the most fitting to construct low power wide area networks (LPWANs) that follow the IoT paradigm. This fact is validated by extensive research, see [4] [5], and by market analyses, in fact LoRa based IoT is estimated at USD 5.6 billion in 2023 and is projected at USD 25.5 billion by 2028 [6]. The areas of application of this technology are health care, logistics, transportation, smart homes, and many others. As stated above, the thesis will focus on applications in the field of agriculture,

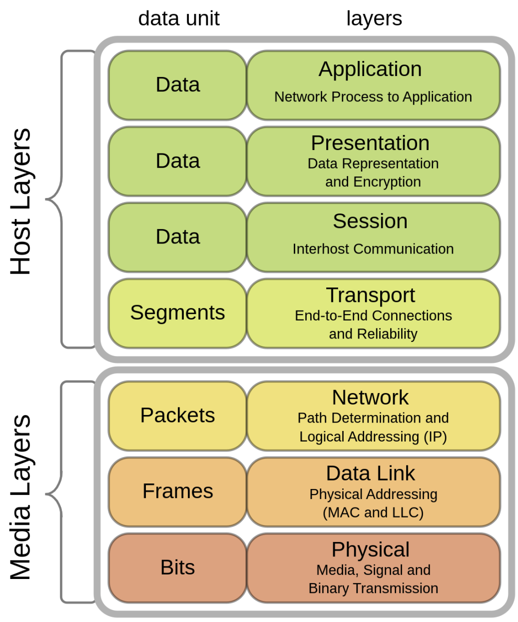

Figure 1: ISO/OSI networking stack.

by presenting a media access control (MAC) layer protocol called Bacco; it makes use of LoRa physical layer protocol (PHY) to construct LPWANs used for exchanging data collected by localized sensors placed on-site. In order to do so, the thesis is organized as follows:

- Chapter 1 provides an overview of the advantages and prerequisites of implementing IoT in agriculture. Furthermore, it introduces Bacco and LoRa wide area network (LoRaWAN), the industry-standard MAC protocol used for implementing LoRa-based LPWANs;
- Chapter 2 describes the protocol and its specifics in depth;
- Chapter 3 discusses and compares the physical performance of Bacco and LoRaWAN with calculations and laboratory tests.

Chapter 1

Benefits, Requirements And Use Case Scenario

The purpose of this chapter is to delineate the advantages and prerequisites of implementing IoT in agriculture, and in particular its implementations that make use of LPWANs. It also aims to introduce Bacco: the protocol that constitutes the fundamental core of this thesis, along with an alternative approach. Specifically, we will provide a concise overview of the LoRaWAN protocol, emphasizing certain illustrative use-case scenarios.

1.1 Benefits

If IoT is integrated into agricultural contexts, it can bring lots of benefits to production:

- Precision farming: IoT devices enable farmers to collect real-time data on environmental variables. This data helps to optimize irrigation, fertilization, and pest control, leading to higher crop yields and reduced resource waste;
- Decision making: the generated data provides farmers with insights into crop health, growth patterns, and yield predictions. This allows them to make informed decisions regarding planting, harvesting, and resource allocation, resulting in better outcomes;
- Remote monitoring and management: farmers can remotely monitor their fields through IoT devices, reducing their physical presence. This is especially valuable for managing large or distant farms;

- Automation and labor savings: automation can handle tasks such as irrigation and pests treatments. This not only reduces labor costs but also frees up farmers to focus on more strategic aspects of their operations;

- Knowledge sharing and collaboration: IoT platforms can facilitate the exchange of best practices, data, and insights among farmers, researchers, and agricultural experts;

- Sustainability and environmental impact: trough optimized resource use, IoT contributes to more sustainable and environmentally friendly agricultural practices;

1.2 Requirements

When integrating IoT into agricultural contexts, a set of unique and challenging requirements come into play:

- Power management: many remote agricultural locations lack reliable power sources. Devices need to be designed with energy-efficient technology paired with high-energy batteries and/or alternative power sources like solar panels to ensure uninterrupted operation;

- Connectivity and coverage: many agricultural areas lack reliable Internet connectivity, which can hinder transmissions between IoT devices and central systems. Even when using alternative solutions such as LPWANs like LoRaWAN or Bacco, long distances and physical barriers are problems that need to be faced;

- Safety on system fails: human operation is not always convenient or even possible in some cases, so having a fail-safe system is crucial;

- Physical damage: constant exposure to weather conditions undermines the integrity of the devices used in the open field, so it is required to use materials that are resistant to such circumstances;

- Environmental impact: some ecosystems may be influenced by the introduction of alien objects or electromagnetic radiation;

- Cost: initial setup costs for devices, sensors, and infrastructure can be high, which may discourage small farm business from adopting IoT;

- Regulations and compliance: different regions may have varying regulations concerning the electromagnetic spectrum usage, environmental monitoring, and technology deployment. Adhering to these regulations can be complex when implementing solutions across different areas;
- User-friendly interfaces: the user interfaces need to be intuitive, especially for farmers who might not have extensive technical knowledge.

This thesis will not comprehensively cover all the aspects mentioned above. Instead, it will focus on the communication protocol utilized by the network. While all efforts have been directed towards meeting the requirements as a whole, it is important to note that mechanical, environmental, economic, and legal considerations are left to future discussions.

1.3 An Already Available Technology: LoRaWAN

One of the most promising technologies in IoT is LoRaWAN, an open protocol built on top of Semtech's LoRa modulation. It provides all the necessary software components to build a network that is reliable, power efficient and scalable, according to extensive research and testing. See [7] and [8] for reference.
A LoRaWAN network consists of sender nodes and gateways. When a sender node broadcasts a LoRa packet, it can be received by one or more gateways, that can store the information or forward it to a web server through the Internet. These kind of messages are called uplinks. Gateways might need to transmit data to the sender nodes and these kind of messages are called downlinks. Gateways can be self-owned with a custom configuration or can be part of an existing community such as The Things Network (TTN) or Helium.
The gateway nodes are always listening to transmissions, so senders can uplink at any time. On the other hand, downlinks need to be scheduled as sender nodes do not require to listen continuously. Depending on the power budget, LoRaWAN defines 3 classes of devices: class A, class B, and class C. We will only describe class A, because of its superior power efficiency compared to the other two. This fact makes it the most suitable for agricultural contexts. In this mode, a downlink can only be sent during precise time slots during which the sender listens for transmissions. 2 slots are open after every uplink, at pre-defined delays. Figure 1.1 shows a schematic version of the time slot management.
Due to the precise time slot management, the LoRaWAN standard does not tolerate additional delays coming from the use of repeaters. Successful research has

been done to overcome this limitation, see [9], however, the required additional custom hardware and the lack of official support from the standard made the solution not appealing to production environments.

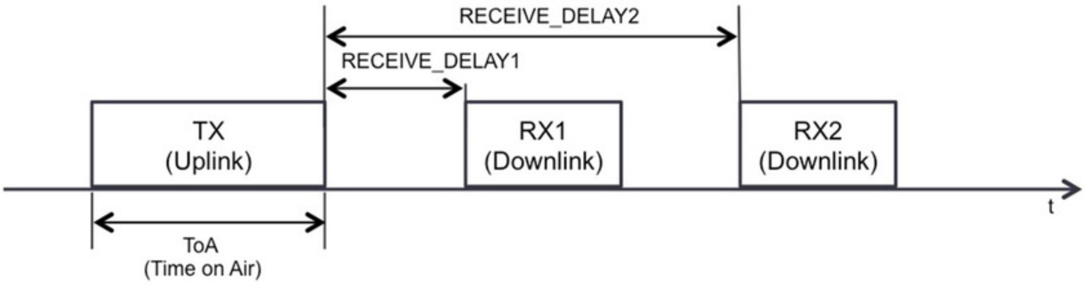

Figure 1.1: LoRaWAN class A device operation.

1.4 Real Word Use Case Scenario

To better understand the limitations and the strengths of LoRaWAN and Bacco, we will now describe 3 real-world scenarios in which IoT can be integrated to achieve the benefits described in Section 1.1. All of them require to collect data on a vineyard, however, the positions and the surface areas of the vineyards are different for the 3 cases. A brief description of each of them follows:

1. A small vineyard with an area of 1 hectare [1] which is fully covered by the signal of one or more hotspots registered on a public LoRaWAN network. The surface of the terrain is almost flat;

2. A small vineyard with an area of 1 hectare which is located far from any urban area and is not in reach from any public hotspot. The surface of the terrain is almost flat;

3. A large vineyard with an area of 100 hectares which is located far from any urban area and is not in reach from any public hotspot. The vineyard is spread across multiple hills.

1.4.1 LoRaWAN Setup Using A Public Gateway

We will now analyze an IoT network that makes use of public infrastructures such as TTN or Helium. Each sender node can be easily registered on the platform

[1] 1 hectare is equivalent to 10^4 m^2

and can transmit independently.
When trying to apply this solution to each of the described scenarios, we would observe the following behavior:

1. Since the vineyard is covered by one or more public hotspots, it is possible to build the desired system. Depending on the distance from the hotspot, the sender nodes may be required to transmit at a high power in order for the signal to be detected by the hotspot. Any fault of nearby hotspots will result in an unusable system;

2. Since the vineyard is not fully covered by any public LoRaWAN network, it is not possible to build the desired system;

3. Since the vineyard is not fully covered by any public LoRaWAN network, it is not possible to build the desired system.

This is the simplest and cheapest proposed solution to implement, as no additional hardware is needed other than the sender nodes, however, it does not satisfy the requirements for 2 out of the 3 proposed scenarios.

1.4.2 LoRaWAN Setup Using An Owned Gateway

We will now analyze an IoT network that makes use of owned/private LoRaWAN hotspots. Each sender node can be connected to it and can transmit independently.
When trying to apply this solution to each of the described scenarios, we would observe the following behavior:

1. Since the vineyard is small and the underlying surface is mostly flat, the hotspot can fully cover it. The system satisfies the requirements;

2. Since the vineyard is small and the underlying surface is mostly flat, the hotspot can fully cover it. The system satisfies the requirements;

3. Since the vineyard is large and spread across multiple hills, it may not be possible for a single hotspot to cover the entirety of it because of physical barriers, so multiple hotspots would be needed to satisfy the requirements.

The solution satisfies the requirements for all the scenarios, however, in the 3^{rd} case, it is necessary to use multiple gateways to achieve full coverage. This makes the system expensive and difficult to configure.

1.4.3 Bacco Setup

Bacco is proposed and described in this thesis as an alternative to the discussed existing solutions. It has the goal to satisfy the requirements for the scenarios and to provide a competitive system to LoRaWAN in the field of agriculture. The next chapters will describe its implementation in detail and compare its performance to LoRaWAN's.

Chapter 2

Bacco Protocol

The goal of this chapter is to give a detailed description of the Bacco protocol and to discuss the implementation choices that were made to deploy it. This is achieved using a top-down ordering for the level of detail, meaning that the overview is presented before the specifics.

2.1 Overview

We will now describe a simple network that makes use of Bacco to better understand its operating principle. The network is built upon 4 categories of devices:

- Sender node - collects data and sends it to a Gateway or Repeater using Bacco over LoRa modulation;
- Repeater node - listens to incoming Bacco messages from Senders and forwards them to a Gateway;
- Gateway node - collects data coming from Sender or Repeater nodes and sends it to the web server. In the example shown in Figure 2.1, this is achieved using file transfer protocol (FTP) over a global system for mobile communications (GSM) or long term evolution (LTE) mobile network. This node has the role of coordinating and synchronizing Sender nodes. It can be optionally configured to perform pre-processing operations (e.g. filtering, smoothing, interpolation ...) on the incoming data;
- Web server - receives data coming from the Gateways, elaborates it, and makes it available to the user. Note that the scheme of communication involving this device is not covered by Bacco.

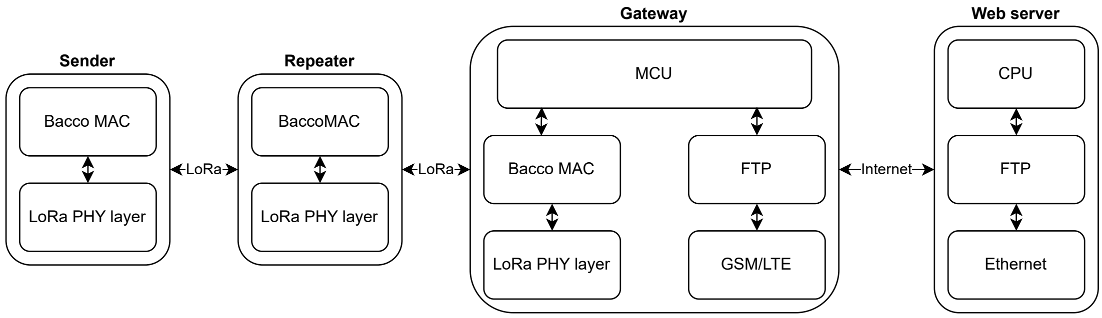

Figure 2.1: Schematic representation of an example network using Bacco.

2.2 Topology

The network has a star-of-stars topology, in which the zeroth level is occupied by the Web server, the first level by Gateways and Repeaters, and the second level contains the Senders. Figure 2.2 shows the types of devices that are involved and their communication scheme.
The structure is equivalent to a tree, hence we can define a hierarchy for the nodes. The root node is the central web server and its children nodes are the Gateways. All sender nodes are children of either a Gateway or a Repeater and have no children, so they correspond to the leaves of the tree.

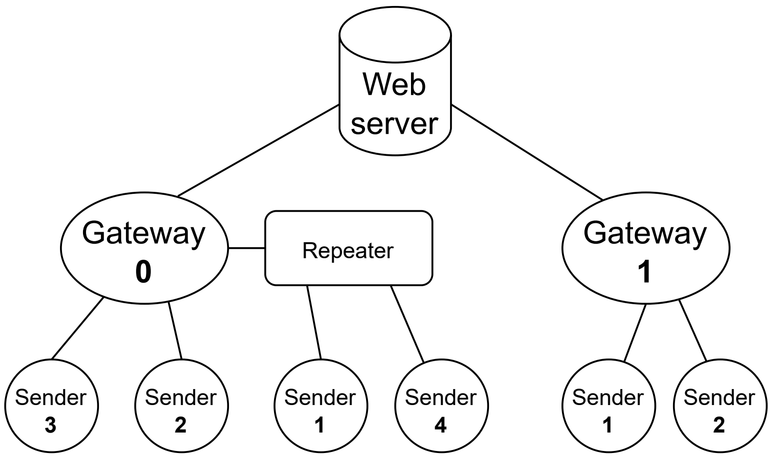

Figure 2.2: Bacco network topology

2.3 Addressing

It is crucial to identify each Sender node to contextualize the messages coming to the Gateway node. This is achieved by assigning them a unique identifier, represented by a natural number in the range $[1, 254]$. Address 0 is reserved for the Receiver and address 255 is used as a globally invalid address. This limits the number of Sender nodes connected to a single Gateway to 254 [1]. If necessary, the network can scale up by using additional Gateways. Note that since Repeaters do not produce messages themselves nor they modify the forwarded ones, they will not be given an identifier.

2.4 Interference Mitigation

The LoRa PHY protocol specification does not fully cover the matter of sharing the communication link between multiple devices, thus it is necessary to define methods for doing so, to minimize interference and achieve a reliable exchange of information. Different techniques are applied in the domain of both time and frequency.

2.4.1 Channel Activity Detection

Channel activity detection (CAD) is a feature available for most LoRa transceivers [10]. In this mode, the LoRa node listens for any transmission on a specific frequency and, if it detects a signal, an interrupt is returned to the micro controller unit (MCU). This presents a possible carrier sense multiple access (CSMA) mechanism.

Bacco does not make use of CAD for data packets, but it enables it in specific situations such as network discovery (discussed in Section 2.5).

2.4.2 IQ Inversion

In phase/quadrature (IQ) inversion is a LoRa primitive that makes it possible to have 2 types of transmissions on the same frequency and spreading factor, that are easily distinguishable for a receiver. The name IQ usually refers to signals that are out of phase from each other by $\frac{\pi}{4}$ rad. Despite that, LoRa uses the IQ acronym to describe signals with inverted chirp direction, namely up-chirp and down-chirp.

Bacco uses this technique to discriminate between uplink messages (i.e., from Sender to Gateway/Repeater or from Repeater to Gateway) and downlink messages (i.e., from Gateway to Sender/Repeater or from Repeater to Sender). This implies that Sender nodes and Gateway nodes are not able to communicate with

[1] This choice is influenced by the considerations presented in Section 3.1.1

other devices of the same category (e.g. a Sender would not detect any transmission coming from a Sender).

2.4.3 Subnetting

The LoRa protocol supports a wide range of carrier frequencies in the very high frequency (VHF) spectrum [2].
Bacco exploits this fact to build sub-networks that operate at different frequencies to achieve very low interference between them. The set of used frequencies is defined by Equation 2.1.

$$f = \{f_k : f_k = 868\text{MHz} + k \times 125\text{KHz}, k \in \{0...10\}\} \tag{2.1}$$

The main sub-network operates at the base frequency of 868.0 MHz, obtained by setting $k = 0$ in Equation 2.1; it is composed of the Gateway and all the Sender nodes connected to it. Every other sub-network operates at a different frequency that is obtained by choosing $k \in \{1...10\}$; they are composed of a Repeater and all the Sender nodes connected to it. To be able to communicate with the Gateway, Repeaters need to forward the messages at the base frequency of the Gateway, regardless of their operating frequency inside the sub-net.
A network using Bacco can have up to 10 Repeaters operating at different frequencies, however, if more coverage is needed, it is possible to have multiple Repeater nodes working at the same frequency, given that they are not in reach with each other. This is very important because of bouncing, a phenomenon that occurs when a message is sent back and forth between Repeaters.
Figure 2.3 shows a network consisting of 3 different sub-networks, represented with different colors; the main one is in white, the blue one has a value of $k = 2$, and the red one has a value of $k = 1$.

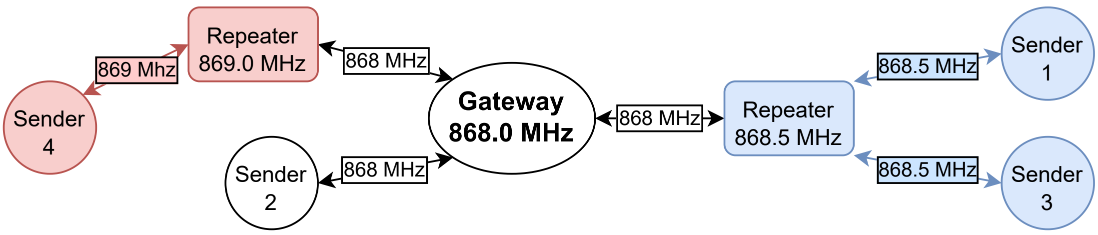

Figure 2.3: Example network with subnets in different colors.

[2]For reference, the SX1262[11] transceiver features a continuous frequency coverage from 150 MHz to 960 MHz

2.4.4 Distribution Of Transmission Activity

The Bacco protocol distributes activity on the radio channel over time with defined frames reserved for each Sender, using an approach that was first introduced by the AlohaNet [12] protocol. Frames are equally distributed between the maximum number of Senders that can be connected to a Gateway (i.e., $N_{max} = 254$), and the frame assignment is based on the identifier (e.g. Sender1 to slot 1, Sender2 to slot 2, etc.). A Sender can only transmit during its assigned frame, otherwise, the Gateway will send a time-correction message. (see Subsection 2.4.4 for a detailed explanation). The time delay between consecutive transmissions from the same Sender is a constant value and it is called C (cycle time). Between each frame, a time equivalent to $\frac{1}{3}$ of a Sender frame is left as tolerance and is called a radio silence frame. At the end of a cycle, a time equal to $\frac{C}{5}$ is reserved for the Gateway to upload the collected data. Figure 2.4 shows a schematic representation of the time management used by Bacco. The cycle time is a user-defined parameter, and all the other values are calculated based on it as shown in Table 2.1.

Table 2.1: Time parameters calculation.

Parameter	Value
Gateway frame time	$0.2 \cdot C$
Sender frame time	$\frac{0.6 \cdot C}{N_{max}}$
Silence frame time	$\frac{0.2 \cdot C}{N_{max}}$

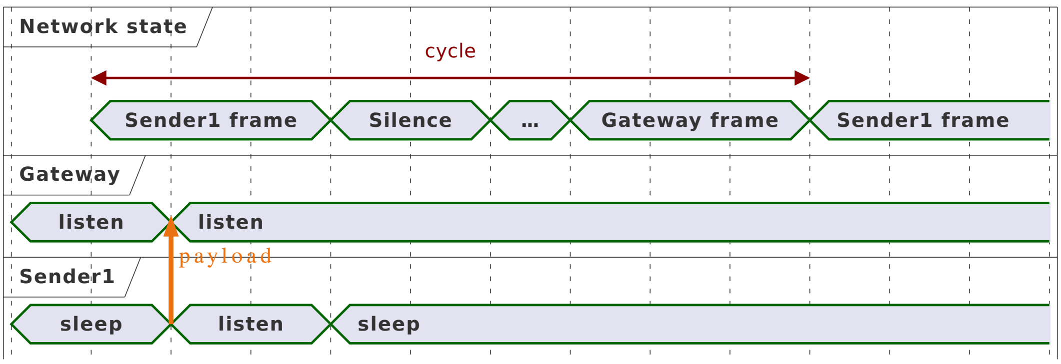

Figure 2.4: Timing diagram - Sender1 is in sync.

Clock drift compensation

All the Senders in the network need to transmit during their assigned frame; this means that all the clocks of the devices are required to be in sync. This is hard to achieve without dynamic recalibration because commercial oscillators can not provide a constant frequency source due to manufacturing imprecisions, temperature gradient, etc...

To deal with this problem, Bacco assigns the Gateway node the role of coordinating the network timings through the dispatch of downlink messages containing the network timestamp. Such a message is sent as soon as the Gateway receives an uplink message that exceeds its correct time frame. Figure 2.5 shows this specific case. In addition to that, the Gateway sends at least 1 downlink message every 10 uplink messages from the same Sender, to indicate that the connection is still active.

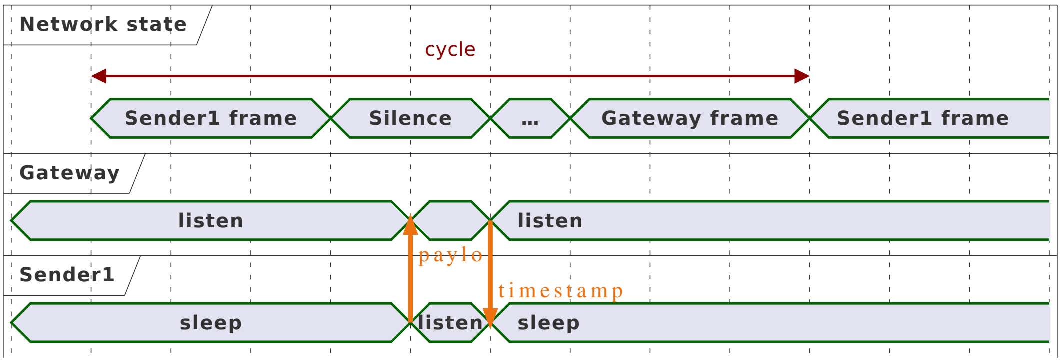

Figure 2.5: Timing diagram - Sender1 is out of sync.

2.5 Network Discovery

When a Sender node is first started, it needs to decide at what frequency to operate for minimizing the power needed to reach a Repeater or Gateway. To this end, Bacco introduces Algorithm 1 for scanning nearby devices and selecting the most suitable. The Sender tries to establish communication with Repeaters and Gateways throughout all the available frequencies by sending a particular type of message that triggers a sync acknowledge message (SYN/ACK) response. CAD will be used by the Sender to not interfere with ongoing communications; this is because the board does not yet have an allocated time frame and thus can not decide when to transmit otherwise.

Algorithm 1 Network discovery.

```
rssi_values ← [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
while all rssi_values are equal to 0 do
    for k from 0 to 10 do
        f_k ← 868 × 10^6 + k × 125 ∗ 10^3
        for i from 0 to 10 do
            do
                sleep for 1 s
                enter CAD mode at frequency f_k for 1.5 s
            while activity detected by CAD
            send sensing message
            enter receive mode for 3 s
            if received SYN/ACK then
                rssi_values[k] ← current rssi
                break
            end if
        end for
    end for
end while
return 868 × 10^6 + argmin(rssi_values) × 125 ∗ 10^3
```

2.6 Network Joining

When a Sender node needs to connect to the network for the first time, it does not yet have an identifier nor its clock is in sync. The procedure to achieve that will be called the joining process. Note that we assume that the Sender node has already selected the frequency of operation, as described in section 2.5.

We will ignore the act of forwarding made by any optional Repeater node, as in this case it does not affect the content of the messages, but note that delay and error rate would raise in that situation. All the messages sent from Sender and Receiver make use of CAD to make sure the channel is free before the actual transmission; this step will be omitted in the description for brevity. First, the Sender transmits a sync message (SYN) to the Gateway and waits for a SYN/ACK response for 3 s. If no message is received, another SYN is sent a maximum of 10 times. After that, the Gateway waits for 3 s for an acknowledge message (ACK) from the Receiver, and if no message is received it will try again for a maximum of 10 times. The SYN/ACK contains the timestamp of the network as well as the assigned identifier. If the maximum number of iterations is reached in any of the steps, the process starts again after 30 minutes. Figure 2.6 shows a schematic representation of the process.

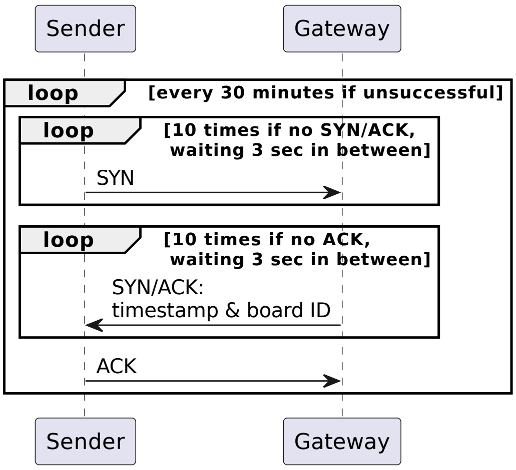

Figure 2.6: Network joining process.

2.7 Downlink Commands

In some situations such as transmission power adaption, it is required to be able to change the behavior of the network dynamically and reliably. Bacco achieves this by exchanging specialized packets that contain commands. Each message contains a command and needs to be acknowledged by the Sender to the Gateway/Repeater in a similar way as done during the network joining procedure. Figure 2.7 shows a graphical representation of the process.
The following commands are defined by the protocol, and each of them is associated with an opcode as shown in Table 2.2:

- Shutdown - If this command is sent and processed successfully, the Sender goes to deep sleep indefinitely until a manual reset is invoked by pressing a physical button;

- Enter sleep mode - If this command is processed successfully, the Sender stops sending data, but it keeps listening for incoming messages/commands during its time frame;

- Wakeup - If this command is processed successfully, the Sender enters normal/active mode and thus starts to transmit data;

- Increase transmission power - If this command is processed successfully, the Sender increases its transmission power by $P_{step} = 3$ dBm;

- Decrease transmission power - If this command is processed successfully, the Sender decreases its transmission power by $P_{step} = 3$ dBm.

Table 2.2: Table of opcodes.

Command	Opcode as 7 bit unsigned integer
Shutdown	0
Enter sleep mode	1
Wakeup	2
Increase transmission power	3
Decrease transmission power	4
Reserved for later use	[4, 50]
User-defined	[51, 127]

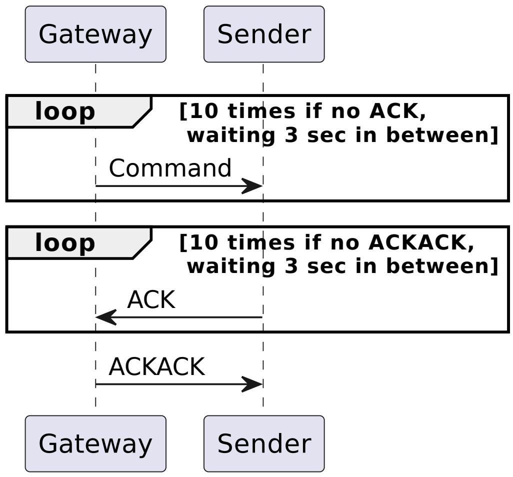

Figure 2.7: Command sending process.

2.8 Transmission Power Adaption

Since Senders can be placed at different distances from a Gateway or Repeater, it is useful to optimize the power used by the node to transmit. The default value for the transmission power is equal to P_0; starting from that, the network will automatically drift towards a more suitable value according to the following triggers and Table 2.3:

- If a Sender has not received any downlink messages during the last 20 frames, then its transmission power will be increased by P_{step} up to a maximum of P_{max}

- If 10 out of the last 10 messages received by a Repeater or a Gateway from the same Sender satisfy RSSI > RSSI_{high} and SNR > SNR_{high}, then a command is sent telling to decrease the transmission power by P_{step} down to a minimum of P_{min}

- If 8 out of the last 10 messages received by a Repeater or a Gateway from the same Sender satisfy RSSI < RSSI_{low} or have not been received, then a command is sent telling to increase the transmission power by P_{step}.

Table 2.3: Parameters for transmit power adaption algorithm.

Parameter	Value
$P_0 = P_{max}$	14 dBm
P_{min}	5 dBm
P_{step}	3 dBm
RSSI_{low}	-115 dBm
RSSI_{high}	-60 dBm
SNR_{low}	-7 dBm

Gateways and Repeaters always operate at P_0, since power efficiency is less critical than ensuring the highest possible delivery rate.

2.9 Packet Format

In this section, the bit format of the messages is shown. The analysis will be split between uplink packets and downlink packets.

2.9.1 Uplink Packet Format

Uplink messages are sent from a Sender to a Gateway/Repeater or from a Repeater to a Gateway. It has a variable length and it is transmitted using up-chirps, i.e., with IQ inversion disabled. The least significant byte contains the Sender's identifier represented as an 8-bit unsigned integer. The second least significant byte contains the size of the payload in bytes as an 8-bit unsigned integer. The rest of the message contains the payload and has a length defined by the previous field. Figure 2.8 shows the packet format.

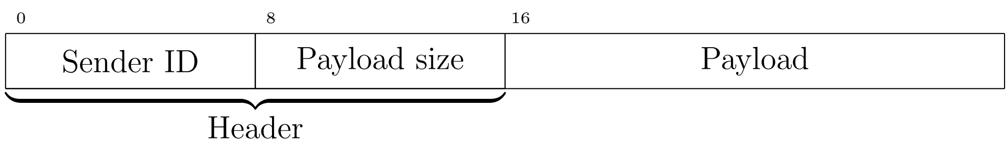

Figure 2.8: Uplink packet format.

2.9.2 Downlink Packet Format

Downlink messages are sent from a Gateway to a Sender/Repeater or from a Repeater to a Sender. They have a fixed length of 5 bytes and are is transmitted using down-chirps, i.e., with IQ inversion enabled. Since the size of each packet is pre-defined, LoRa's implicit header mode is used, that means that no physical layer header is added to the MAC layer packet. The least significant byte contains the identifier of the Sender for which the message is directed as an 8-bit unsigned integer. The following bit contains the type of the message: 0 represents a time sync message whereas 1 represents a command message. The content of the following bits depend on the type of message: in the case of a time sync message, the remaining 31 bits contain the timestamp, whereas in the case of a command message, the first 7 bits contain an opcode and the remaining 14 bits are left for padding and can be later used by future revisions of the protocol. Figure 2.9, 2.10, 2.11 show the general format, the timestamp format, and the command format respectively.

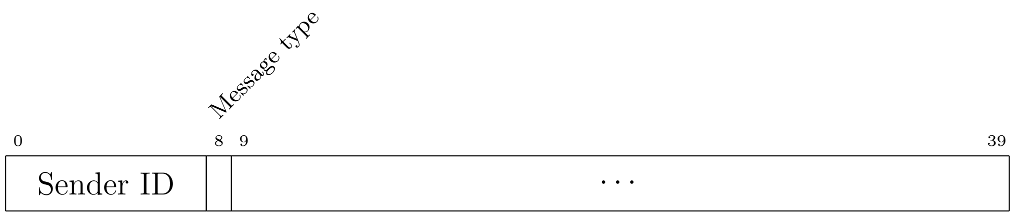

Figure 2.9: Downlink packet format.

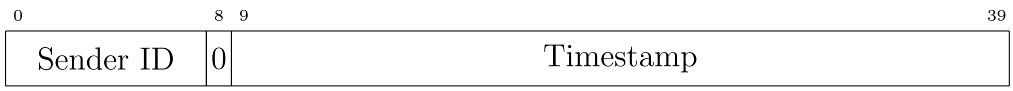

Figure 2.10: Downlink packet format for timestamps.

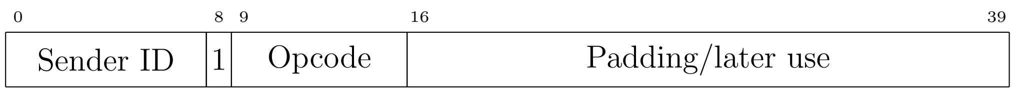

Figure 2.11: Downlink packet format for commands.

2.9.3 Comparing Bacco And LoRaWAN Packet Formats

We will now compare Bacco's packet format to LoRaWAN's. The latter is described thoroughly by Semtech's official documentation [13], a schematic reference can be found in Figure 2.12 and 2.13.
The focus of the comparison will be on the total size of the packets rather than the function of each field, to analyze the overhead associated with each protocol.

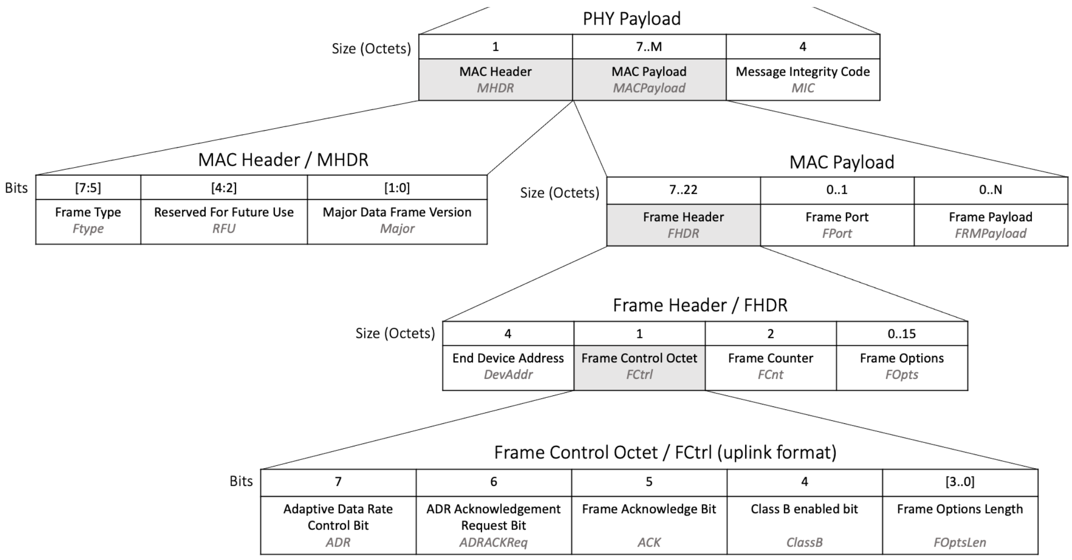

Figure 2.12: LoRaWAN uplink packet format.

Uplink Packets

Considering two uplink packets containing the same payload and transmitted with the two protocols under analysis, we can observe that Bacco uplink packets feature a fixed header size of 2 bytes as described in Section 2.9.1, while LoRaWAN uses a MAC header size that varies between 7 and 23 bytes.
To show that, we can consider an empty *Frame Payload* and sum all the other fields' sizes to get the overall header size. LoRaWAN supports a range of sizes for the uplink packets; the lower bound is given by Frame Header: 7 + Frame Port: 0; the upper bound is given by Frame Header: 22 + Frame Port: 1. This means that, at a fixed payload size, Bacco uplink packets are always shorter by a constant factor between 5 and 21 bytes. Figure 2.14 shows a plot of the total size of an uplink packet compared to its payload length, using both Bacco and LoRaWAN.

Downlink Packets

Considering two uplink packets containing the same payload, but trasmitted using the two protocols under analysis, we can observe that Bacco downlink packets feature a fixed overall size of 5 bytes as discussed in Section 2.9.2, while LoRaWAN uses packets with a header of size between 7 and 23 bytes (the calculation is analogous to the one of the uplink case, discussed in Section 2.9.3).

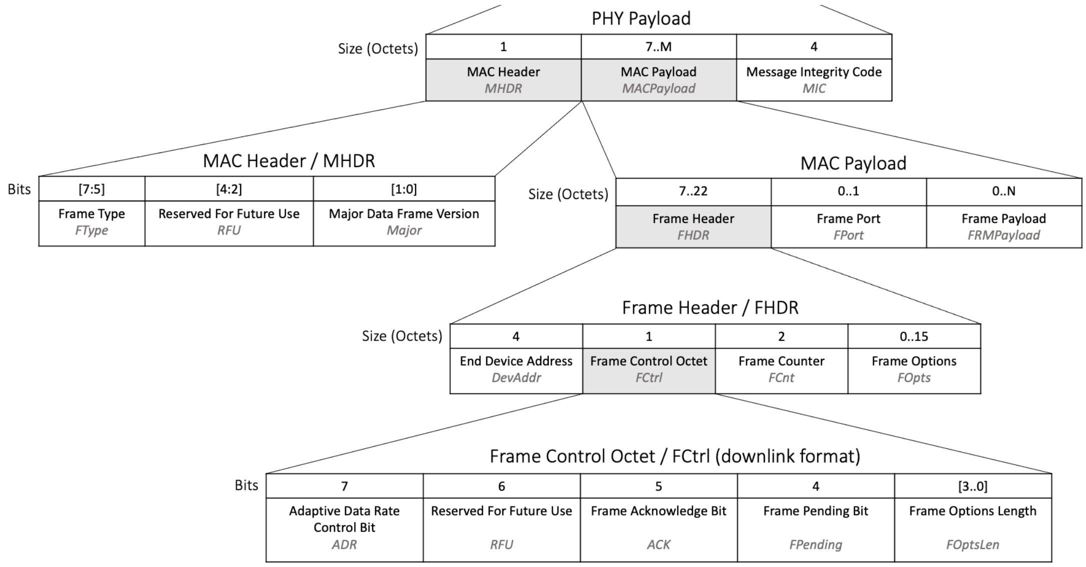

Figure 2.13: LoRaWAN downlink packet format.

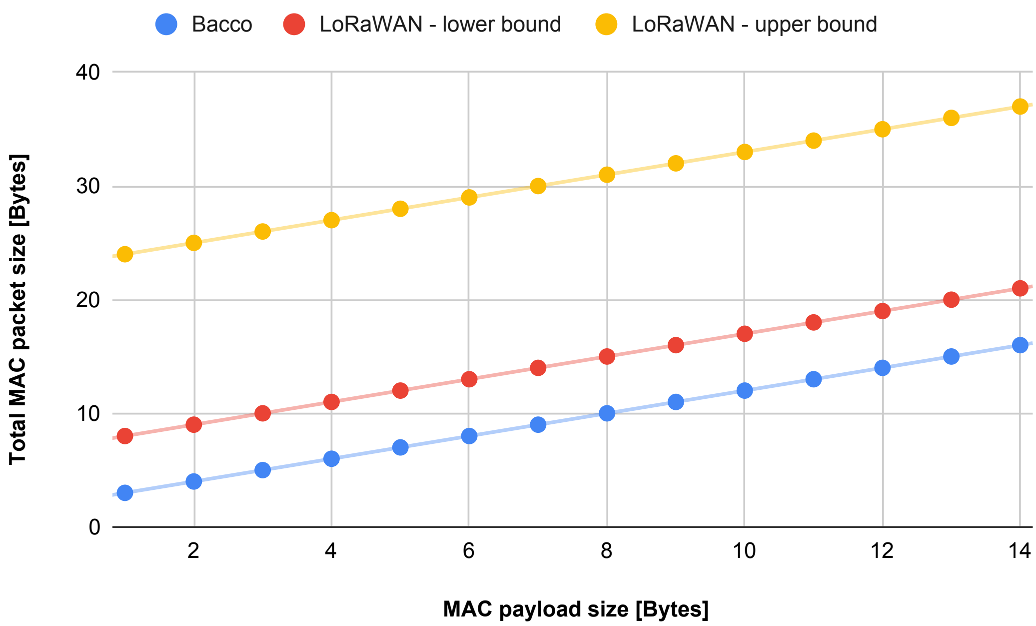

Figure 2.14: LoRa and Bacco total packet size with respect to payload size

Chapter 3

Performance

The goal of this chapter is to discuss the performance of Bacco, using a variety of parameters, methods, and laboratory tests. The results will also be compared to the ones obtained by applying the same procedures to devices using the LoRaWAN protocol.

3.1 Time On Air And Duty Cycle

Time on air (ToA) is a crucial metric to take into consideration when measuring the performance of a network protocol. It is defined as the time that a device takes to transmit a packet on the channel. At the same conditions, a shorter ToA results in improved energy efficiency and a smaller probability of interference with packets transmitted from other devices.
ToA is correlated to the packet length in symbols (a longer packet will require a longer transmission time). Referencing the official LoRa documentation, see [11], we can find that ToA is discussed in section 6.1.4; in particular the exact formula for ToA is shown here by Equation 3.1.

$$ToA = \frac{2^{SF}}{BW} * N_{symbols} \tag{3.1}$$

where

- SF is the Spreading Factor (from 5 to 12);
- BW is the Bandwidth (in kHz);
- $N_{symbols}$ is the number of symbols in the packet.

$N_{symbols}$ can be calculated knowing the number of bytes that compose the packet

($N_{bytes_{payload}}$) by using Equation 3.2.

$$N_{symbols} = N_{symbols_{preamble}} + 12.25 + max\left(ceil\left(\frac{8*N_{bytes_{MAC}} - 4*SF + 24 + 20*H}{4*SF}\right), 0\right) * \frac{4}{CR} \quad (3.2)$$

where

- $N_{symbols_{preamble}}$ is equal to the number of symbols in the preamble sync message. This parameter can be set arbitrarily, but it is often given a value of 8.
- H denotes the presence of an explicit physical layer header. It assumes the value of 1 when the header is present and 0 when it is not present;
- CR is equal to the coding rate. It can be set to $\frac{4}{5}$, $\frac{4}{6}$, $\frac{4}{7}$ or $\frac{4}{8}$.
- $N_{bytes_{MAC}}$ is equal to the total number of bytes of the MAC packet.

The duty cycle is the fraction of time in which the channel is busy, it can be calculated using Equation 3.3.

$$d = \frac{\tau}{T} \quad (3.3)$$

where

- d is the duty cycle;
- τ is the ToA;
- T is the transmission period.

3.1.1 Regulations

To ensure a fair distribution of the transmission activity, local, national and international governments define limits on the usage of the radio channel. The limits vary by the frequency band in which the devices operate. The two parameters that are often used as an upper bound not to be crossed are effective radiated power (ERP)[1] and duty cycle.
Using the current Italian regulations at the time of writing as an example, we can observe that the spectrum is split into frequency bands that can be either free, reserved, or restricted; each of them has three parameters associated with it: the maximum ERP, the minimum channel bandwidth and the maximum duty cycle. The official document that discusses this matter in detail can be found

[1]ERP is the measure of the power effectively radiated by an antenna system in a specific direction, accounting for both transmitter output and antenna characteristic. It is commonly expressed in watts or decibels and it is used for determining the coverage area and range of a radio.

in [14]. The bands that LoRa devices use in Europe are $[868.0, 868.6]$ MHz and $[868.7, 869.2]$ MHz, they both feature a maximum ERP of 25 mW or 14 dBm and do not have any restriction on channel bandwidth, however, the first band has a maximum duty cycle of 1%, while for the other the limit is set to 0.1%.
An important fact to point out is that the limits apply to a physical person or organization and not to on a device basis, so to calculate the effective duty cycle, we have to consider the transmission activity generated by the whole network.

3.1.2 Maximum Number Of Devices In A Network

We can now calculate the maximum number of devices that can be operational at the same time. To achieve that, we can derive Equation 3.4 from Equation 3.1 and then maximize the number of devices ($N_{devices_{max}}$), getting Equation 3.5.

$$\frac{\tau * N_{devices}}{T} \leq d_{max} \tag{3.4}$$

$$N_{devices_{max}} = floor\left(\frac{d_{max} * T}{\tau}\right) \tag{3.5}$$

where

- T is the transmission period and can be set arbitrarily;
- τ can be computed using Equation 3.1;
- d_{max} is the maximum duty cycle, it has to be smaller or equal to 1 and can be set according to the local regulations or specific needs.

3.1.3 Duty Cycle Calculation

To demonstrate the use of the Equation 3.3 to calculate the duty cycle, we will now consider a network with the following properties:

- the electromagnetic interference is negligible and each packet gets delivered with an error rate of 0;
- each Sender node has a transmission period of $T = 10$ min $= 360$ s;
- each Sender node transmits a packet with a payload that has a length in bytes equal to $N_{bytes_{MAC\_payload}} = 15$;
- the devices in the network operate at a frequency of 868.0 MHz on the Italian territory, and thus have to respect the duty cycle limit of $d_{max} = 1\% = 0.01$;

- the devices in the network operate at a spreading factor equal to $SF = 7$;
- the devices in the network operate at a bandwidth equal to $BW = 125$ kHz;
- all the packets feature a coding rate equal to $CR = \frac{4}{5}$;
- all the packets are transmitted with a preamble length equal to $N_{symbols_{preamble}} = 8$.

Bacco

As discussed in Section 2.9, the overhead associated with each MAC packet in uplink mode is equal to $N_{bytes_{MAC\_header\_uplink}} = 2$, so the total MAC packet length is equal to

$$N_{bytes_{uplink}} = N_{bytes_{MAC\_header\_uplink}} + N_{bytes_{MAC\_payload}} = 2 + 15 = 17 \text{ bytes}$$

Now, using Equation 3.2 we can calculate the total number of symbols used to encode an uplink packet

$$mathitN_{symbols_{uplink}} = 8 + 12.25 + max\left(ceil\left(\frac{8*17-4*7+24+20*1}{4*7}\right), 0\right) * \frac{4}{4/5} = 50.25 \text{ symbols}$$

Now, using Equation 3.1 we can calculate the ToA of an uplink packet

$$ToA_{uplink} = \frac{2^7}{125000} * 50.25 = 51.456 \text{ ms}$$

As discussed in Section 2.9, the overall size of a downlink packet is equal to $N_{bytes_{dowlink}} = 5$ bytes Now, using Equation 3.2 we can calculate the total number of symbols used to encode a downlink packet

$$N_{symbols_{uplink}} = 8 + 12.25 + max\left(ceil\left(\frac{8*5-4*7+24}{4*7}\right), 0\right) * \frac{4}{4/5} = 30.25 \text{ symbols}$$

Now, using Equation 3.1 we can calculate the ToA of a downlink packet

$$ToA_{downlink} = \frac{2^7}{125000} * 30.25 = 30.976 \text{ ms}$$

Since the packet error rate is equal to 0, only one downlink message is sent every 10 uplink messages as discussed in 2.4.4, so the average ToA per period is equal to

$$ToA_{avg} = ToA_{uplink} + ToA_{downlink} * \frac{1}{10} = 54.5536 \text{ ms}$$

Now, using Equation 3.5 we can calculate the maximum number of devices that can be connected to the network

$$N_{devices_{max}} = floor\left(\frac{0.01 * 360}{54.5536 * 10^{-3}}\right) = 65$$

LoRaWAN

As discussed in Section 2.9.3, both LoRaWAN uplink and downlink packets have a MAC header that is greater or equal to $N_{bytes_{MAC\_header\_uplink}} = N_{bytes_{MAC\_header\_downlink}} =$ 12 bytes. We will suppose that that's the header size never exceeds that value, thus the total size of the uplink and downlink packets are

$$N_{bytes_{uplink}} = N_{bytes_{MAC\_header\_uplink}} + N_{bytes_{MAC\_payload}} = 13 + 15 = 28 \text{ bytes}$$

$$N_{bytes_{downlink}} = N_{bytes_{MAC\_header\_downlink}} = 13 \text{ bytes}$$

Now, using Equation 3.2 and 3.1 we can calculate the total ToA per period

$$ToA_{uplink} = 66.816 \text{ ms}$$

$$ToA_{downlink} = 46.336 \text{ ms}$$

Since a downlink packet is sent as an ACK for every uplink, the average ToA per period is equal to

$$ToA_{avg} = ToA_{uplink} + ToA_{downlink} = 448.816 \text{ ms}$$

Now, using Equation 3.5 we can calculate the maximum number of devices that can be connected to the network

$$N_{devices_{max}} = floor\left(\frac{0.01 * 360}{448.816 * 10^{-3}}\right) = 8$$

3.2 Laboratory Tests

This section will present laboratory tests concerning the performance of devices running Bacco. Specifically, the emphasis will be on power draw and packet error rate. The obtained results will be compared to those obtained using LoRaWAN.

3.2.1 Transmission Power Draw

This section will present a test that aims to measure the power drawn during the transmission of a packet. Here follows a list of the conditions under which the experiment was conducted:

- **Transmitter device**: Heltec AB01 MCU, equipped with ARM Cortex M0+ Core and Semtech SX1262 LoRa module, see [15]
- **Measuring device**: Keysight DSOX1204G digital oscilloscope.
- **Power source**: 3.3V stabilized voltage source.
- **Measuring circuit**: 1.18 ohm shunt resistor connected in series to the voltage source and in parallel to the oscilloscope probes, see Figure 3.1.
- **LoRa configuration**: uplink mode, MAC payload with a size of 15 bytes, spreading factor set to 7, output power set to 14 dBm, bandwidth set to 125 kHz.

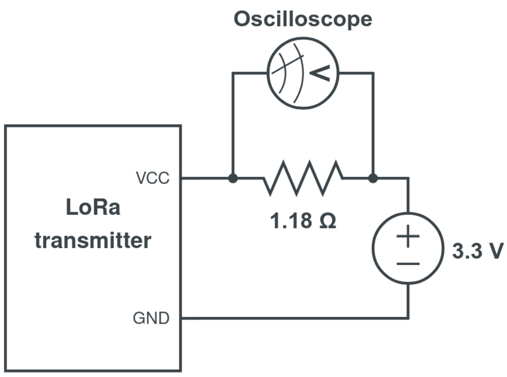

Figure 3.1: Electrical diagram of the measuring setup for power draw

The tests were conducted by transmitting a single packet multiple times at identical conditions. It has been observed that the experiment yielded exceptionally consistent results across the different transmissions. The procedure was applied to Bacco packets and LoRaWAN packets. By analyzing the data logs collected by the oscilloscope reported in Figure 3.2, it is possible to measure the ToA, equivalent to the period of time in which the board consumed a power greater than 350 mA, and the transmission energy, equivalent to the discrete integral of the power draw over the ToA period. The numerical results are shown in Table 3.1.

Table 3.1: Single packet transmission numerical results.

Parameter	Bacco	LoRaWAN
ToA	51.6 ms	71.8 ms
Transmission energy	21.3 mJ	30.8 mJ

3.2.2 Packet Error Rate

This section will present a test that aims to measure the packet error rate of a device transmitting messages using Bacco. Here follows a list of the conditions under which the experiment was conducted:

- **Sender nodes**: 3 x Heltec AB01 MCU, equipped with ARM Cortex M0+ Core and Semtech SX1262 LoRa module, see [15]
- **Gateway node**: Espressif ESP32 MCU, equipped with Semtech SX1276 LoRa module
- **LoRa configuration**: Uplink MAC payloads with size varying between 7 and 23 bytes, spreading factor set to 7, output power set to 5 dBm, bandwidth set to 125 kHz.
- **Displacement**: The Senders were placed approximately 200 m away from the Gateway with no clear line of sight as they were placed in between vineyards.
- **Signal strength and noise**: The average receive signal strenght indicator (RSSI) was -102 dBm and the average signal to noise ratio (SNR) was 1 dBm.
- **Cycle time and total duration**: The transmission cycle was set to 1 hour and the experiment lasted a total of 2 weeks; the total number of transmitted packets was 1008.

The Gateway was programmed to count the number of successfully received packets. Table 3.2 shows the results of the experiment.

Table 3.2: Packet error rate test results.

Total packets sent	Total packets received	Error rate
1008	1004	0.4%

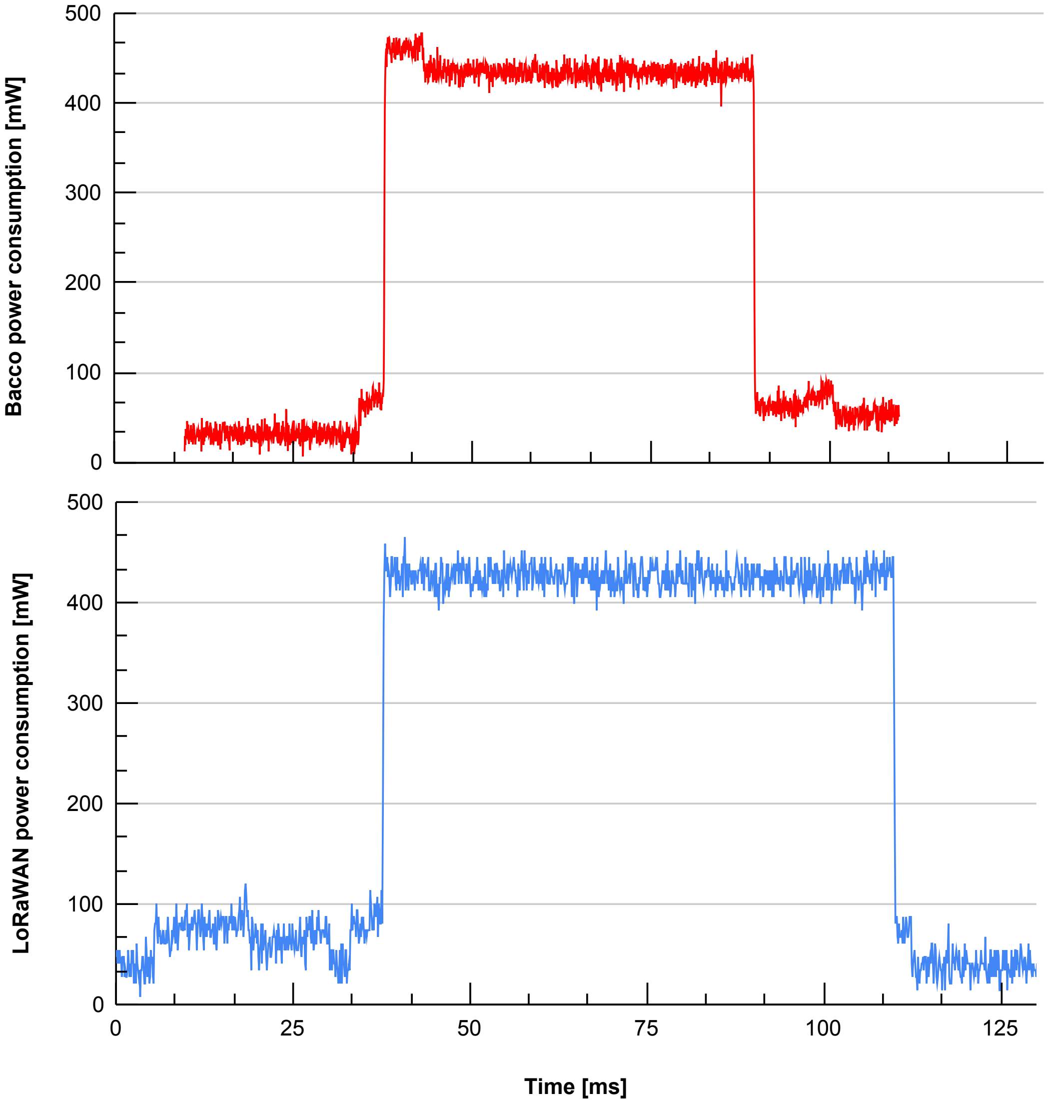

Figure 3.2: Power draw of Bacco (in red) and LoRaWAN (in blue) during the transmission of a packet with a payload of 15 bytes, using SF7, at 14dBm TX power and 125kHz bandwidth

Bibliography

[1] V. K. Garg and J. E. Wilkes, *Principles and Applications of GSM*. Prentice Hall PTR, 1998, ISBN: 0139491244. [Online]. Available: https://dl.acm.org/doi/abs/10.5555/552293.

[2] A. Ghosh, R. Ratasuk, B. Mondal, N. Mangalvedhe, and T. Thomas, "Lte-advanced: Next-generation wireless broadband technology [invited paper]", *IEEE Wireless Communications*, vol. 17, no. 3, 2010. DOI: 10.1109/MWC.2010.5490974.

[3] J. Postel and J. Reynolds, "File transfer protocol (ftp)", Network Working Group, RFC 959, 1985. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc959.

[4] M. Bor, J. E. Vidler, and U. Roedig, *LoRa for the Internet of Things*. Junction Publishing, 2017, ISBN: 9780994988607. [Online]. Available: https://eprints.lancs.ac.uk/id/eprint/77615.

[5] A. Augustin, J. Yi, T. Clausen, and W. M. Townsley, "A study of lora: Long range amp; low power networks for the internet of things", *Sensors*, vol. 16, no. 9, 2016, ISSN: 1424-8220. DOI: 10.3390/s16091466. [Online]. Available: https://www.mdpi.com/1424-8220/16/9/1466.

[6] "Lora and lorawan iot market by offering (hardware, platforms, and services), application (smart cities, industrial iot, smart healthcare), end user (retail, manufacturing, healthcare, energy utilities, residential) and region - global forecast to 2028", Research And Markets, Tech. Rep. [Online]. Available: https://www.researchandmarkets.com/reports/5821213/lora-lorawan-iot-market-offering-hardware?utm_source=GNE&utm_medium=PressRelease&utm_code=239mpc&utm_campaign=1868182+-+LoRa+and+LoRaWAN+IoT+Markets+-+Global+Forecast+to+2028+-+Increasing+Smart+City+Initiatives+%26+Rising+Adoption+of+Intelligent+Devices+Bolsters+Growth&utm_exec=chdo54prd.

[7] D. Davcev, K. Mitreski, S. Trajkovic, V. Nikolovski, and N. Koteli, "Iot agriculture system based on lorawan", in *2018 14th IEEE International Workshop on Factory Communication Systems (WFCS)*, 2018. DOI: 10.1109/WFCS.2018.8402368.

[8] P. Boonyopakorn and T. Thongna, "Environment monitoring system through lorawan for smart agriculture", in *2020 - 5th International Conference on Information Technology (InCIT)*, 2020. DOI: 10.1109/InCIT50588.2020.9310977.

[9] E. Sisinni, P. Ferrari, D. Fernandes Carvalho, *et al.*, "Lorawan range extender for industrial iot", *IEEE Transactions on Industrial Informatics*, vol. 16, no. 8, pp. 5607–5616, 2020. DOI: 10.1109/TII.2019.2958620.

[10] *Channel activity detection: Ensuring your lora packets are sent*, Semtech. [Online]. Available: https://lora-developers.semtech.com/documentation/tech-papers-and-guides/channel-activity-detection-ensuring-your-lora-packets-are-sent.

[11] *Sx1261/2 datasheet*, Rev. 2.1, Semtech, Dec. 2021. [Online]. Available: https://semtech.my.salesforce.com/sfc/p/E0000000JelG/a/2R000000Un7F/yT.fKdAr9ZAo3cJLc4F2cBdUsMftpT2vsOICP7NmvMo.

[12] N. Abramson, "Development of the alohanet", *IEEE Transactions on Information Theory*, vol. 31, no. 2, pp. 119–123, 1985. DOI: 10.1109/TIT.1985.1057021.

[13] *In-depth: Sending and receiving messages with lorawan*, Semtech. [Online]. Available: https://lora-developers.semtech.com/documentation/tech-papers-and-guides/sending-and-receiving-messages-with-lorawan.

[14] *Elenco delle interfacce per gli apparati radioelettrici - apparati a corto raggio utilizzati per applicazioni generiche (es. telemetria, telecomandi, allarmi e simili)*, Feb. 2005. [Online]. Available: https://www.gazzettaufficiale.it/atto/serie_generale/caricaArticolo?art.versione=1&art.idGruppo=0&art.flagTipoArticolo=1&art.codiceRedazionale=05A00705&art.idArticolo=1&art.idSottoArticolo=1&art.idSottoArticolo1=10&art.dataPubblicazioneGazzetta=2005-02-12&art.progressivo=0.

[15] *Ab01 product page and datasheet*, Heltec. [Online]. Available: https://heltec.org/project/htcc-ab01/.

[16] L. S. Vailshery, “Number of internet of things (iot) connected devices worldwide from 2019 to 2023, with forecasts from 2022 to 2030”, Statista, Tech. Rep. [Online]. Available: https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/.