Enabling Bidirectional Communication on Batteryless Devices For Sustainable IoT

Ashish Kumar Sultania

Supervisor **Prof. Jeroen Famaey**

Thesis submitted in fulfilment of the requirements for the degree of Doctor in Science: Computer Science

Faculty of Science — Antwerpen, 2022

University of Antwerp

University
of Antwerp

Faculty of Science

Enabling Bidirectional Communication on Batteryless Devices For Sustainable IoT

Thesis submitted in fulfilment of the requirements for the degree of
Doctor in Science: Computer Science
at University of Antwerp

Ashish Kumar Sultania

Supervisor
Prof. Jeroen Famaey

Antwerpen, 2022

Jury

Chairman
Prof. Chris Blondia, University of Antwerp, Belgium

Supervisor
Prof. Jeroen Famaey, University of Antwerp, Belgium

Members
Prof. Carles Gomez, Universitat Politècnica de Catalunya, Spain
Prof. Danny Hughes, Katholieke Universiteit Leuven, Belgium
Prof. Jeroen Hoebeke, Ghent University, Belgium
Prof. Maarten Weyn, University of Antwerp, Belgium

Contact
Ashish Kumar Sultania

University of Antwerp
Faculty of Science
IDLab
Sint-Pietersvliet 7, B-2000 Antwerp, Belgium

Abstract

The Internet of Things (IoT) envisions billions of smart sensors and actuators to be connected to the Internet. Many of these devices require wireless connectivity. The IoT imposes three main requirements on such wireless networks: wide transmission range, low cost, and low power consumption. The last requirement is crucial for the devices to have long autonomy, thus reducing the need for device maintenance. Low-power networks, such as Low Power Wide Area Network (LPWAN) for long-range transmission and Wireless Personal Area Networks (WPAN) for short-range transmission, promise to fulfil the stated IoT requirements. The LPWAN Narrowband Internet of things (NB-IoT) and WPAN Bluetooth Low Energy (BLE) radio technologies are popular and enable low-cost and low-power devices. Such communication technologies optimize energy consumption by introducing a sleep or deep sleep state, allowing the device to spend most time in a low-power state. This potentially enables IoT devices to operate for several years, powered by small coin cell batteries. Nevertheless, while the device is in this sleep state, it is unreachable and cannot immediately receive any downlink (DL) data. Thus, saving the battery life comes at the cost of DL data latency. The uplink (UL) data communication is mostly controlled by the device and is therefore less impacted, as the device can wake-up when it needs to transmit data. Despite being energy-efficient technologies, most IoT devices still require frequent manual battery replacement. This increases operational costs and limits the feasibility to deploy in remote areas, and also affects the environment due to the harmful chemicals that the discarded batteries can leak into the soil. Therefore, the interest in energy harvesting (EH) from ambient sources and usage of a small capacitor to store the energy has been rising, enabling the Internet of Batteryless Things (IoBT). Renewable energy sources such as solar, wind, thermal, and radio frequency energy are popular options for harvesting. A key challenge in ensuring sustainable energy harvesting is dealing with the time-varying energy supply and changing device power demand. The much smaller energy density of the capacitor and unpredictable availability of harvested energy result in intermittent on-off behaviour of the device. This means a batteryless device can turn on and off frequently, and that can further impact its data latency.

This thesis studies the trade-off between energy consumption and communication latency for Bluetooth Mesh and NB-IoT, two popular wireless IoT network technologies. I first study BLE, a WPAN technology for short-range communication. An analysis is done on how the friendship feature can be leveraged to minimize DL data latency of Bluetooth Mesh low-power nodes (LPNs) that spend most of their time in a deep sleep state or a turn-off state. Also, the impact on the data latency by introducing a secondary ultra-low-power wake-up radio (WuR) to a Bluetooth Mesh LPN is explored. Next, the focus switches to the study of NB-IoT, an LPWAN technology for long-range communication. NB-IoT proposes Power Saving Mode (PSM) and Extended Discontinuous Reception (eDRX) features to save power consumption. These features are configured using various timers to switch the device's states. Therefore, a model is designed to investigate the impact of each timer on energy consumption and latency. Moreover, it can help to deduce the optimal configuration according to the IoT use case requirements.

Additionally, batteryless prototypes for Bluetooth Mesh and NB-IoT devices based on indoor light are also developed by leveraging the Bluetooth Mesh friendship and NB-IoT eDRX/PSM features. I evaluate the suitability of using Bluetooth Mesh and NB-IoT in the context of intermittently powered devices by analyzing their energy requirements in different radio states. The intermittent operation means these devices frequently lose power altogether, causing task execution to be halted prematurely and loss of executing contexts. It means on a restart, they need to start from the beginning. This intermittent operation of a batteryless device can be avoided by programming it to be aware of its capacitor voltage. The energy-aware device executes the tasks only if it has the required energy stored in the capacitor to execute the task. On that account, the energy-aware device delays the task's execution time until its capacitor recharges up to a required threshold voltage. The thesis compares energy-aware and energy-unaware solutions in terms of data latency and packet delivery ratio by developing a Bluetooth Mesh prototype that enables Bluetooth Mesh batteryless LPN to communicate uni-directional downlink only or bi-directional for different capacitor sizes at different light-harvesting powers.

Acknowledgements

I would like to express my utmost gratitude to the Almighty, who is my eternal source of strength, courage and inspiration.

I want to thank Prof. dr. Jeroen Famaey for providing me with valuable guidance, feedback, freedom and the opportunity to deliver quality work during my research journey. He has been of great influence to me both professionally and personally.

I worked on different projects such as MAGICIaN, BLUESS, COMBAT, and IoBaLeT. On that account, I would like to acknowledge all the sponsors, University of Antwerp, FWO, and imec, for funding these projects that provides me with this coveted opportunity to finish my degree in an excellent research lab (IDLab). I would also like to thank my co-supervisor of the BLUESS and COMBAT project, Dr. Carmen Delgado, for helping me technically, reviewing my work and providing detailed feedback which greatly improved the quality of this work.

I have enjoyed working with Prof. Dr. Chris Blondia. He patiently helped and supported me to design the mathematical models. His zestful and jovial personality always cheered me up and made working with him a fun experience.

I am grateful to everyone I have collaborated with; Adnan Sabovic, Farouk Mahfoudhi, Subho Basu, Dragan Subotic, Priyesh Puluckul, Pouria Zand and Martina Capuzzo. Thank you all for your support, input and collaboration. Also, thanks to everyone I have met through the IDLab for your friendship especially Serena, Raja, Esteban, Filip, Glenn, Le Tian, Ritesh, and Akash. Your support has added a lot to my life. It provided me with a space to take a break and enjoy talks that have certainly improved my work. I have already published two papers with Farouk, those ideas were discussed during the breaks or fun events.

I am always thankful to my family for their support especially my father, mother and wife who have believed in me and supported my pursuit of education. Many sacrifices have been made over the years that have gone a long way towards helping me advance and achieve success in my career. I am also blessed with the love of my daughter Tanvi during the course of this PhD .

"No one who does good work will ever come to a bad end"
"You are what you believe in. You become that which you believe you can become"
— Bhagavad Gita

Contents

List of Figures

List of Tables

Acronyms

API Application Programming Interfaces.

AT Arrival Time.

BLE Bluetooth Low Energy.

BW Bandwidth.

C-DRX Connected-mode Discontinuous Reception.

CE Coverage Enhancement.

CP Control Plane Cellular IoT.

CS Capacitor Size.

CyP Cyclic Prefix.

DCI Downlink Control Information.

DL Downlink.

DLI DL Data Interval.

DRX Discontinuous Reception.

EC Energy Check.

ECI Energy Check Interval.

EDLC Electrochemical Double-layer Capacitors.

eDRX Extended Discontinuous Reception.

EH Energy Harvesting.

eNB eNodeB.

ESR Equivalent Series Resistance.

FM Frequency Modulation.

FN Friend Node.

FP Friend Poll.

FQ Friend Queue.

FSN Friend Sequence Number.

FU Friend Update.

GSM Global System for Mobile Communication.

HP Harvesting Power.

I-DRX Idle-mode Discontinuous Reception.

IAT Inter-Arrival Time.

IoBT Internet of Batteryless Things.

IoT Internet of Things.

LPN Low Power Node.

LPWAN Low Power Wide Area Networks.

MCU Micro-Controller Unit.

MD More Data.

MIB-NB Narrowband Master Information Block.

MPP Maximum Power Point.

MPPT Maximum Power Point Tracking.

NB-IoT Narrowband Internet of things.

NCS Modulation and Coding Scheme.

NPBCH Narrowband Physical Broadcast Channel Narrowband.

NPDCCH Narrowband Physical Downlink Control Channel.

NPDSCH Narrowband Physical Downlink Shared Channel.

NPRACH Narrowband Physical Random Access Channel.

NPSS Narrowband Primary Synchronization Signal.

NPUSCH Narrowband Physical Uplink Shared Channel.

NRS Narrowband Reference Signal.

NSSS Narrow-band Secondary Synchronization Signal.

OFDMA Orthogonal Frequency Division Multiple Access.

PASTA Poisson Arrivals See Time Averages.

PDR Packet Delivery Ratio.

PDU Protocol Data Unit.

PE Poll Expiration.

PI Polling Interval.

PMB Power Management Board.

PO Paging Occasion.

PPK-II Nordic-Power Profiler Kit-II.

PRB Physical Resource Blocks.

PSM Power Saving Mode.

PT PollTimeout.

PTW Paging Time Window.

PV Photovoltaics.

QPSK Quadrature Phase Shift Keying.

RA Random Access.

RACH Random Access Channel.

RAI Release Assistance Indication.

RD ReceiveDelay.

RE Eesource Element.

RF Radio Frequency.

RRC Radio Resource Control.

RU Resource Unit.

RW ReceiveWindow.

Rx Receive.

SC-FDMA Single Carrier Frequency Division Multiple Access.

SDK Software Development Kit.

SF Subframe.

SI Signal Interval.

SIB1-NB Narrowband System Information Block.

SINR Signal-to-Noise Ratio.

TAU Tracking Area Update.

TBS Transport Block Size.

TEG Thermoelectric Generator.

Tx Transmit.

UE User Equipment.

UL Uplink.

ULI UL Data Interval.

UP User Plane Cellular IoT.

WPAN Wireless Personal Area Networks.

WSN Wireless Sensor Network.

WuR Wake-up Radio.

WuS Wake up Signal.

Part I

Introduction

Chapter 1

Research Introduction

1.1 Context

Internet of Things (IoT) devices are typically composed of a microcontroller, memory, interfaces to communicate with peripherals (sensors, radio) and batteries. They can potentially be installed anywhere and sometimes in a hard-to-reach area. Also, they are usually power constrained [1]. Therefore, communication technologies that target low-power consumption are usually selected for IoT. Multiple technologies have been developed and adapted for the IoT requirements. IoT communication standards include Low Power Wide Area Networks (LPWAN) for long-range transmission and Wireless Personal Area Networks (WPAN) for short-range transmission to support applications in domains such as traffic management, health care systems, environmental monitoring, and smart buildings. Such technologies include Narrowband Internet of things (NB-IoT), LoRa, Sigfox as long-range communication technologies and Bluetooth Low Energy (BLE), ZigBee, Z-Wave as short-range technologies (shown in Figure 1.1).

With the benefits of batteries being cheap and a source of constant power, nowadays, IoT devices, being low-powered, can operate for several months on a non-rechargeable battery [2]. To save the battery life and keep the devices to be operational for a long time, almost all the technologies support switching the devices to SLEEP or DEEP SLEEP state. In this state, the devices consume the lowest possible power by shutting off their communication and internal components. To study the power consumption and power-saving schemes of IoT technologies, Bluetooth Mesh and NB-IoT are chosen in this thesis by considering their wide adoption in the market [3–5]. The BLE mesh specification [6] provides a friendship feature (which operates on top of BLE radio), and NB-IoT proposes two power-saving schemes: Power Saving Mode (PSM) and Extended Discontinuous Reception (eDRX) [7] that enable their nodes to save energy by keeping themselves in SLEEP or DEEP SLEEP mode (by switching off most of their circuitry while staying registered to the network) for most of the time. They awaken only (or turn on) to transmit or receive data packets. To send uplink (UL) data packets, these nodes can send them at any time by switching from SLEEP to ACTIVE mode. On the other hand, it can switch to ACTIVE mode only periodically to receive downlink (DL) packets that are temporarily buffered at the network. As the devices (BLE or NB-IoT) can wake up at any time to send UL data, they

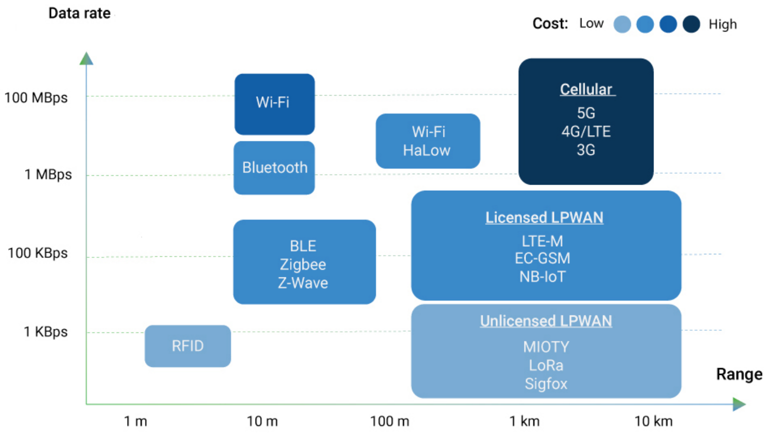


Figure 1.1: Wireless communication technologies for IoT [13].

can experience low UL latency. In contrast, the DL latency is affected because in SLEEP or DEEP SLEEP state the devices are inactive and can only receive DL packets when they explicitly wake up (either scheduled or with receiver-initiated polling). A low DL latency is also important in various IoT use cases e.g., to update the device configuration, in control loops, to poll for data, or to update firmware.

Even though the devices consume low power, they have a limited lifetime and need the replacement of the used batteries. As IoT supports numerous applications, it is expected that billions of connected devices will be deployed in the future [2]. However, deploying so many devices comes with the challenge of powering these devices sustainably. When we look at the problem of replacing billions of batteries every year, it not only increases the cost of maintenance but also creates damaging environmental waste [8]. Moreover, this operation is not instantaneous, and sometimes the devices are non-accessible if installed in hard-to-reach areas. Battery replacement is time-consuming, complicated, and expensive, even when the device itself is of low cost. It means battery replacement for such a large number of IoT devices is not only impractical but also impacts the environment due to the harmful chemicals that discarded batteries can leak into the soil. To solve this problem, environment-friendly capacitors and energy harvesters can replace the batteries. Capacitors support a vast number of charging cycles and thus have a much longer lifetime than batteries [9].

Therefore, to enable sustainable operation of IoT devices without any periodic maintenance, the use of energy harvesting from ambient sources and power management techniques can be considered. Many energy sources such as light, wind, thermal, radio frequency (RF) and vibrations can be harvested with solar panels, wind turbines, thermoelectric generators (TEGs), RF harvesters, and piezoelectric elements, respectively [10]. Figure 1.2 shows the amount of energy that can be harvested using different harvesting techniques. Light energy (e.g., from indoor our outdoor sunlight, or artificial light bulbs) harvesting is attractive in terms of the amount of harvested power and its availability.

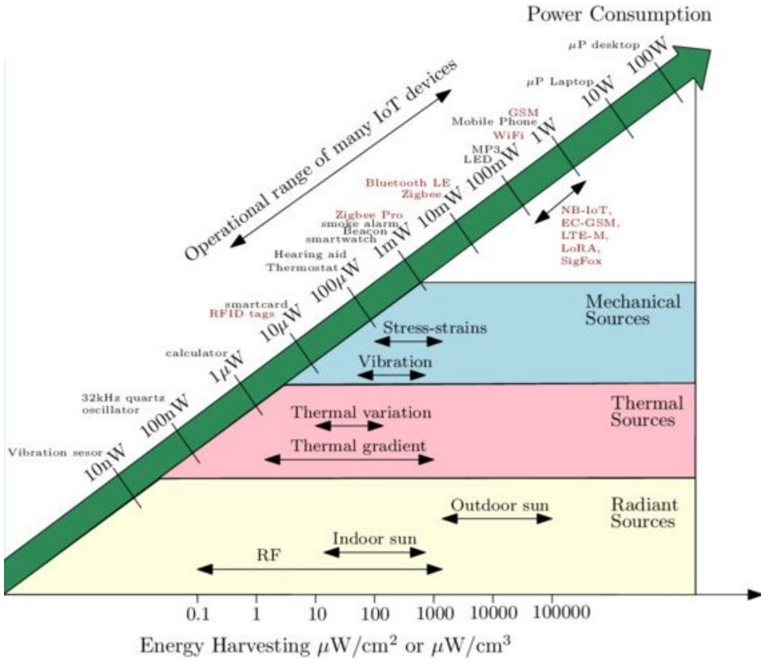


Figure 1.2: Power consumption for various applications and energy harvesting density for various energy sources [14].

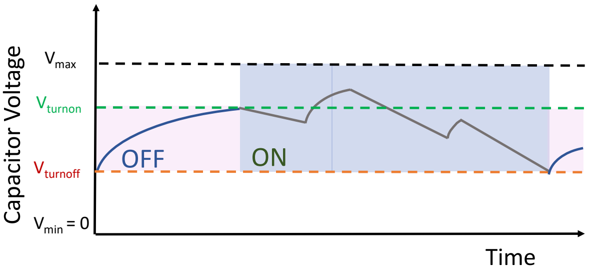


Figure 1.3: Intermittent behaviour of capacitor enabled device harvesting continuously.

Even though the amount of harvested energy is limited and unpredictable, solar panels and other harvesters can be a practical and cost-effective solution to provide energy to an IoT device. This results in batteryless devices to be intermittently powered [11] as shown in Figure 1.3. When the capacitor voltage drops below the minimum operating voltage ($V_{turnoff}$), the device will turn off. To mitigate the intermittent behaviour, the system and its software can be developed to be energy-aware to maximise the network's operational lifetime and reduce the data latency. An energy-aware IoT device can delay the execution of its scheduled task until it collects the required energy level in the capacitor [12]. This can help the device to avoid unexpectedly turning off. Therefore, in comparison to a harvest-store-use architecture with an unpredictable energy source, an energy-aware solution could ensure a more reliable operation of the batteryless IoT device.

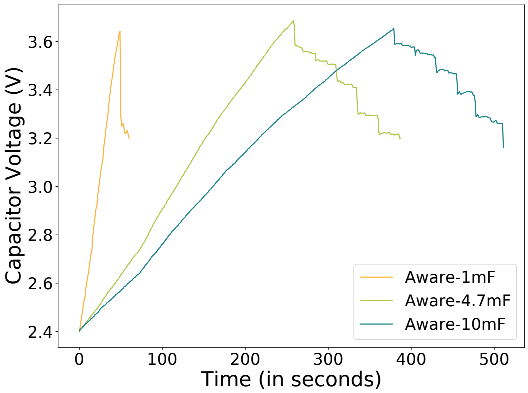


Figure 1.4: Charging and discharging behaviour of capacitors at fixed harvesting power of 250 μW sending Bluetooth Mesh packets at data interval of 10s once the capacitor reaches 3.67 V.

The charging and discharging behaviour of the capacitor is heavily influenced by its capacity. Specifically, a large capacitor can store more energy while it takes longer to charge itself, as shown in Figure 1.4. Therefore, the selection of the optimal capacitor depends on various factors such as the power consumption of the communication protocols, energy harvesting, etc. Generally, this varies in range from tens of microwatts to hundreds of milliwatts for different radio states.

1.2 Problem Statement

IoT devices save their energy by switching off the radio and other peripherals when not performing any tasks. They switch to SLEEP or DEEP SLEEP state and can not receive any DL messages. Also, when these devices are powered by a capacitor and energy harvesters, they can experience intermittent power failures, which can further aggravate the DL latency. Therefore, it is important to study the trade-off between energy-saving and DL data latency in both battery-powered and energy harvesting IoT devices. To move the research on these topics forward, the research questions of this thesis are as follows:

- **How is the power consumption of an IoT device affected by different power saving scheme's configurations?** An analysis that allows researchers to characterize the energy consumption of the IoT devices quickly and can help in finding the battery life of the installed device.

- **How can an IoT device be configured optimally to achieve low-latency communication at the lowest possible power consumption?** Minimizing the communication delay while maintaining low-power consumption is an important IoT requirement that needs to be researched. Moreover, can introducing an extra wake-up radio (WuR) help to improve the performance.

- **Can LPWAN and WPAN IoT devices be powered without batteries, and what are the challenges?** Batteries provide constant power, but replacing them with ambient energy harvesters and capacitors makes the system work intermittent. Therefore, the challenges in the batteryless system are different.

- **How can the intermittent behaviour of batteryless devices be avoided, and are there any benefits?** Due to the unpredictability of energy harvesting and the use of a small capacitor, the device power can become intermittent. This causes the device to frequently switch between the ON and OFF states as it is unaware of its available energy while trying to perform scheduled tasks. Therefore, can avoiding this intermittent behaviour improve the system performance?

- **Can batteryless devices support both uni- and bi-directional communications and their performance?** Normally, batteryless communication is not controlled and performed whenever the device has energy to do transmission and therefore targets mostly uni-directional uplink communication. However, the challenge is to enable not only uni-directional uplink but also uni-directional downlink and bi-directional communication.

1.3 Research Contributions

The motivation toward using less or no-batteries requires not only smart but green solutions for IoT infrastructure. This inspires research towards batteryless IoT solutions with energy harvesting techniques and power management. Therefore, in this thesis, two technologies, short-range Bluetooth and long-range NB-IoT, are chosen to evaluate the feasibility of a batteryless solution by answering the research questions posed in the previous section. The list of contributions is following:

- Starting from the short range technology because of its low power consumption, I started the feasibility research of the batteryless solution of Bluetooth Mesh LPN. An analytical model is devised based on the capacitor behaviour, Bluetooth Mesh friend node queuing (FQ) and Bluetooth Mesh LPN polling mechanism. The performance is evaluated in terms of packet delivery ratio and DL data latency. **Chapter 3**

- As a batteryless device, its radio is usually in DEEP SLEEP to save energy, only waking up intermittently to transmit or receive data. Therefore, the device consumes less power, but the data latency can massively increase. To improve the reaction time to receive the data packets maintaining low-energy consumption, a separate radio called WuR can be used. WuR, as an addition to the main radio, helps in switching the device components completely off, while it has an always-on capability for on-demand communication with power consumption in the uW range. As soon as a request signal with a unique identifier is received, it wakes up the main processor and other peripheral devices. This WuR can improve the DL latency in batteryless devices. Therefore, this work proposes a model and simulation to study the WuR impact and usefulness in **Chapter 4**.

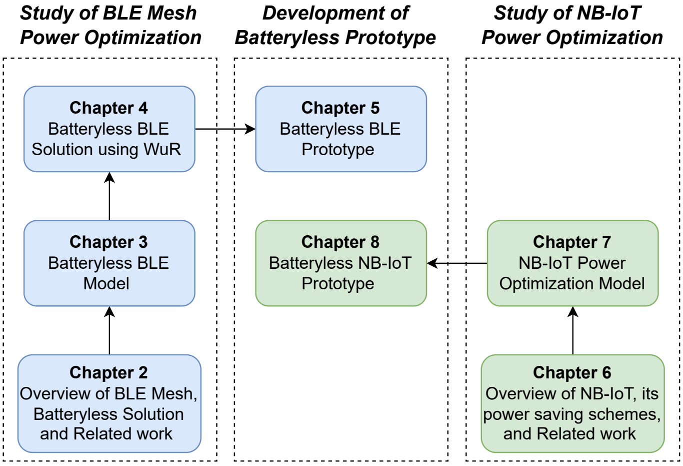

Study of BLE Mesh Power Optimization
Development of Batteryless Prototype
Study of NB-IoT Power Optimization
Chapter 4 Batteryless BLE Solution using WuR
Chapter 5 Batteryless BLE Prototype
Chapter 3 Batteryless BLE Model
Chapter 8 Batteryless NB-IoT Prototype
Chapter 7 NB-IoT Power Optimization Model
Chapter 2 Overview of BLE Mesh, Batteryless Solution and Related work
Chapter 6 Overview of NB-IoT, its power saving schemes, and Related work

Figure 1.5: Schematic overview of the structure of this dissertation.

- In addition to the theoretical study, this work aims to analyse the techniques experimentally using commercial off-the-shelf devices showing the batteryless design. **Chapter 5** for the Bluetooth mesh network focuses on the prototype focusing on the performance of bidirectional communication using solar-powered Bluetooth Mesh devices. Therefore, the development of energy-efficient solutions for such batteryless devices is a crucial issue. In this thesis, an energy-aware system model to operate Bluetooth Mesh batteryless nodes is designed and implemented. Specifically, it helps in avoiding the intermittent behaviour of the batteryless node. **Chapter 5** presents the experimental analysis of the energy-aware and energy unaware solution.

- Next, the study of the NB-IoT is performed focusing on the analysis of tradeoff between power consumption and DL latency for long-range NB-IoT. This chapter proposes an analytical system model to analyse the power consumption and the DL data latency for an NB-IoT device. This model helps to find the optimal configuration to achieve the best DL data latency, spending the lowest possible energy. **(Chapter 7)**

- The prototype of a batteryless NB-IoT device and its performance for different capacitor sizes and harvesting powers is developed. Due to the high power consumption for network acquisition by the NB-IoT devices, a batteryless NB-IoT device requires large size capacitor and high harvesting power. This is presented in **Chapter 8**.

1.4 Thesis Outline

This thesis is divided into two parts as illustrated in Figure 1.5. **Part 1**, composed of Chapters 2 to 5, focuses on a batteryless and the energy-aware solution for **BLE** nodes. Chapter 2 introduces the Bluetooth Mesh protocol, batteryless solution and related work. The power consumption model of a batteryless low power node (LPN) is presented in Chapter 3. Chapter 4 focuses on the analysis of introducing the benefits of WuR in a batteryless Bluetooth Mesh LPN. Lastly, Chapter 5 details the practical experiments using Nordic hardware for the batteryless solution, principally focusing on energy-aware and unaware LPN. **Part 2**, which comprises Chapters 6, 7 and 8, concentrates on the **NB-IoT** device. Chapter 6 introduces the NB-IoT technology, its power saving scheme, and contains literature review. Chapter 7 focuses on the optimal configurations to gain low DL data latency and power consumption. Moreover, Chapter 8 describes the design of the power management system for energy harvesting NB-IoT devices.

1.5 Publications

All the work presented in this thesis has been published or submitted in seven journals and five international conferences and workshops. The list of publications follows:

1.5.1 A1: Publications in international journals indexed by the Web of Science

J1 **A. K. Sultania**, C. Delgado, and J. Famaey, "Enabling Low-Latency Bluetooth Low Energy on Energy Harvesting Batteryless Devices Using Wake-Up Radios," Sensors, vol. 20, no. 18, p. 5196, Sep. 2020 [Online]. Available: `http://dx.doi.org/10.3390/s20185196`. [Impact Factor: 3.567]

J2 **A. K. Sultania**, F. Mahfoudhi and J. Famaey, "Real-Time Demand Response Using NB-IoT," in IEEE Internet of Things Journal, vol. 7, no. 12, pp. 11863-11872, Dec. 2020, doi: 10.1109/JIOT.2020.3004390. [Impact Factor: 9.471]

J3 **A. K. Sultania**, C. Blondia and J. Famaey, "Optimising the Energy-Latency Tradeoff in NB-IoT With PSM and eDRX," in IEEE Internet of Things Journal, vol. 8, no. 15, pp. 12436-12454, 1 Aug.1, 2021, doi: 10.1109/JIOT.2021.3063435. [Impact Factor: 9.471]

J4 **A. K. Sultania**, and J. Famaey, "Batteryless Bluetooth Low Energy Prototype with Energy-Aware Bidirectional Communication Powered by Ambient Light," in IEEE Sensors Journal, 2022, doi: 10.1109/JSEN.2022.3153097. [Impact Factor: 3.301]

J5 **A. K. Sultania**, C. Delgado, C. Blondia and J. Famaey, "Downlink Performance Modeling and Evaluation of Batteryless Low Power BLE Node," in MDPI Sensors, 2022. [Impact Factor: 3.576]

1.5.2 Publications submitted in international journals indexed by the Web of Science

J6 **A. K. Sultania**, and J. Famaey, "Batteryless NB-IoT Prototype for Bidirectional Communication Powered by Ambient Light," in Ad Hoc Networks, 2022. [Impact Factor: 4.816]

J7 F. Mahfoudhi, **A. K. Sultania**, and J. Famaey, “Firmware Update for Constrained IoT Devices Over NB-IoT,” in MDPI Sensors, 2022.

1.5.3 Published in Conference proceedings indexed by the Web of Science

C1 **A. K. Sultania**, P. Zand, C. Blondia and J. Famaey, “Energy Modeling and Evaluation of NB-IoT with PSM and eDRX,” 2018 IEEE Globecom Workshops (GC Wkshps), 2018, pp. 1-7, doi: 10.1109/GLOCOMW.2018.8644074.

C2 **A. K. Sultania**, C Delgado, and J. Famaey, “Implementation of NB-IoT Power Saving Schemes in ns-3,” In Proceedings of the 2019 Workshop on Next-Generation Wireless with ns-3, WNGW 2019. Association for Computing Machinery, New York, NY, USA, 2019, pp. 5–8. DOI:https://doi.org/10.1145/3337941.3337944.

C3 S. S. Basu, **A. K. Sultania**, J. Famaey and J. Hoebeke, “Experimental Performance Evaluation of NB-IoT,” 2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2019, pp. 1-6, doi: 10.1109/WiMOB.2019.8923221.

C4 **A. K. Sultania**, and J. Famaey, “Demo Abstract: Energy-Aware Battery-Less Bluetooth Low Energy Device Prototype Powered By Ambient Light,” In Proceedings of The 19th ACM International Conference on Embedded Networked Sensor Systems (SenSys), Nov 15-17, 2021, Coimbra, Portugal, 2021, pp. 1–2.

C5 M. Capuzzo, C. Delgado, **A. K. Sultania**, J. Famaey, A. Zanella, “Enabling Green IoT: Energy-Aware Communication Protocols for Battery-less LoRaWAN Devices,” MSWiM ’21: Proceedings of the 24th International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, Nov 2021, pp. 95–98.

Chapter 3 describes the model presented in journal J5. Chapter 4 presents the analysis and performance evaluation of journal J1. Then, Chapter 5 demonstrates the working prototype of batteryless Bluetooth Mesh and its results of publications J4 and C4. Next, in the NB-IoT part, the analysis of the optimal configuration of an NB-IoT device is presented in Chapter 7 are published in J3 and C4. Finally, Chapter 8 describes the batteryless NB-IoT implementation as presented in the journal publication J6.

Some publications listed above were not directly part of the research described in this dissertation. As such, they were not included in this thesis. These works are J2, J7, C1, C3, and C5.

Part II

Bluetooth Low Energy Mesh

Chapter 2

Introduction to Bluetooth Mesh

The content of this chapter is based on:

- *A. K. Sultania, C. Delgado, C. Blondia and J. Famaey, "Downlink Performance Modeling and Evaluation of Batteryless Low Power BLE Node," in MDPI Sensors, 2022. [Impact Factor: 3.576]*
- *A. K. Sultania, C. Delgado, and J. Famaey, "Enabling Low-Latency Bluetooth Low Energy on Energy Harvesting Batteryless Devices Using Wake-Up Radios," Sensors, vol. 20, no. 18, p. 5196, Sep. 2020. [Impact Factor: 3.567]*
- *A. K. Sultania, and J. Famaey, "Batteryless Bluetooth Low Energy Prototype with Energy-Aware Bidirectional Communication Powered by Ambient Light," in IEEE Sensors Journal, 2022. [Impact Factor: 3.301]*

This chapter provides an overview of the Bluetooth mesh network and its friendship feature. The concept of Wakeup Radio (WuR) that can be applied to a Bluetooth Mesh LPN is also introduced here. Moreover, the batteryless concepts such as energy-aware and unaware solution for a Bluetooth Mesh LPN is also summarized. Finally, a literature review on the Bluetooth Mesh batteryless works and the WuR solution is presented.

2.1 Introduction of BLE and BLE Mesh

Bluetooth technology is focused on short-range wireless connectivity. In 2010, the Bluetooth Special Interest Group (SIG) released a new version of Bluetooth (Bluetooth 4.0) which is known as Bluetooth Low-Energy (BLE) or Bluetooth Smart. The main aim was to target the connectivity of battery powered IoT devices. One of the uses is a broadcast-type sensor (e.g., temperature, humidity, etc.) that enables the possibility to send one message to a large number of receivers simultaneously.

In 2016, additional enhancements were released to improve speed and distance. This produced the development of a new connectivity model for BLE known as the Mesh Profile.

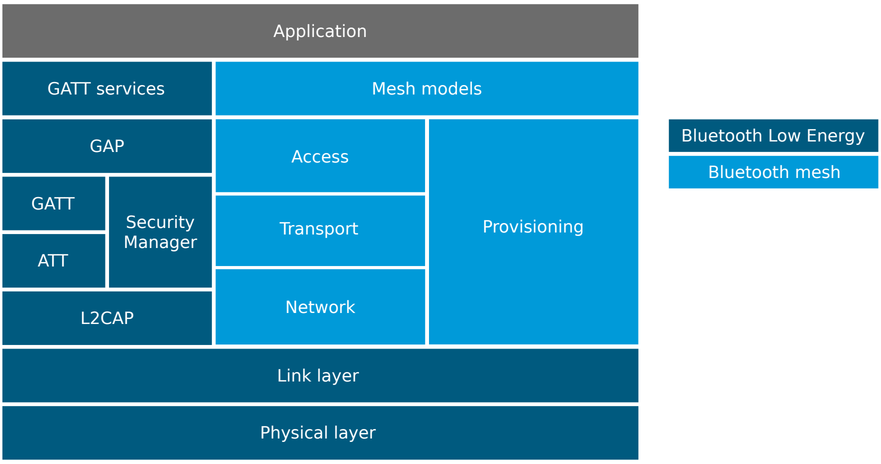


Figure 2.1: Relationship between Bluetooth mesh and Bluetooth LE specifications [15]

Bluetooth mesh specifies a completely new host stack.

Some concepts are shared with the lowest layers of BLE Smart as shown in Figure 2.1. BLE Mesh is a many-to-many style of communication with provisions for message relay from one device to another to create a flexible mesh topology with multiple potential paths between nodes to ensure message delivery.

2.2 BLE Mesh Nodes

The devices that join a Bluetooth mesh network are called nodes. They follow a publish-subscribe communication pattern. A node publishes messages to send, and receivers can subscribe to the sender's address to receive them. The nodes can possess optional additional features based on their capabilities in the network. These features categorize the nodes as a relay, proxy, friend, or low power node as shown in Figure 2.2. The **relay nodes** support the re-transmission of data packets that are broadcast by other nodes. They help in extending the range of the entire network. The **proxy nodes** help the non-mesh-supported BLE devices to communicate via the mesh network. The **Friend Node** (FN) and **Low Power Node** (LPN) have a friendship relationship where the FN receives and stores the DL data packets intended for the associated LPN while the LPN sleeps or temporarily shuts down. The FN maintains multiple friend queues (FQs), one for each connected LPN, to store all the incoming data packets [16]. The maximum size of an FQ containing 16 bytes of lower transport protocol data units (PDUs) that the LPN can request is 128 packets. The data can be retrieved later by the LPNs using a polling mechanism. This provides the LPNs with the flexibility to remain mostly in the lowest power state. A node in the network can enable or disable its responsibility to support these four features, as they come with an additional overhead while connected to the network.

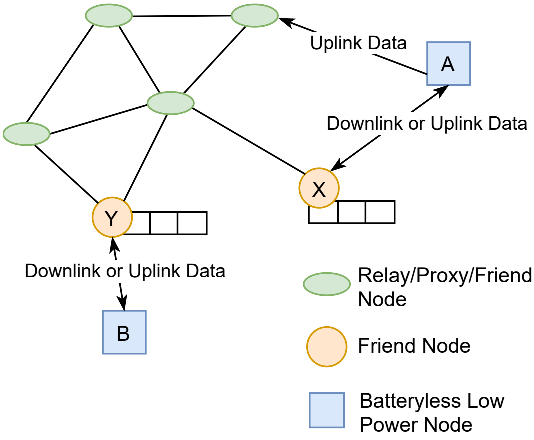


Figure 2.2: BLE mesh network.

2.2.1 BLE Friendship Feature

According to the Bluetooth mesh specification, the LPN initiates the request to establish a friendship relationship. The neighbouring nodes (within a single hop) can respond for the acceptance with their capabilities to become friends. Subsequently, the LPN accepts one of the best capable nodes to be its friend. A node cannot have the low power feature enabled unless a neighbouring node agrees to be a friend. Additionally, an FN needs a sustainable power supply to keep it awake consistently. An FN can support friendship with a maximum of seven LPNs simultaneously. Whereas an LPN can be a friend with only one FN. Figure 2.3 shows an example of the message exchange between the LPN and FN after establishing the friendship. The LPN sends a friend subscription list message to the FN, which contains all its subscribed addresses. This list enables the FN to identify which messages to buffer for the LPN. The LPN periodically sends a friend poll (FP) message to the FN to get any stored data and to keep the connection alive. The FP messages are sent in all the three BLE advertising channels (37–39). After receiving the FP message, the FN replies with the oldest buffered data packet. It discards the packet from the FQ once the LPN acknowledges its reception. The acknowledgement consists of a single bit and is referred to as the friend sequence number (FSN). The LPN toggles the FSN each time it successfully receives a packet. Therefore, the FN sends another entry of the FQ if it receives an FP message with a different FSN field value as in the previously received FP message. If the FSN is the same; it retransmits the previous message (if it has not been discarded in the meantime). The FN returns a friend update (FU) message if the FQ is empty or when the security parameters for the network have changed. The FN signals the FQ occupancy to the LPN via the 1-byte More Data (MD) flag in the FU message. If MD equals 0, it denotes the FQ is empty, 1 that it is not. The values 2 to 255 are reserved for future use.

The standard Bluetooth Mesh friendship protocol defines three timing parameters: ReceiveDelay (RD), ReceiveWindow (RW), and PollTimeout (PT), that are fixed for a session of the relationship. Their corresponding times can be represented as T_{RD}, T_{RW}, and T_{PT}, respectively. These timers are negotiated during the process of the friendship establishment as shown in Figure 2.4 that illustrates the messages exchanged during the friendship es-

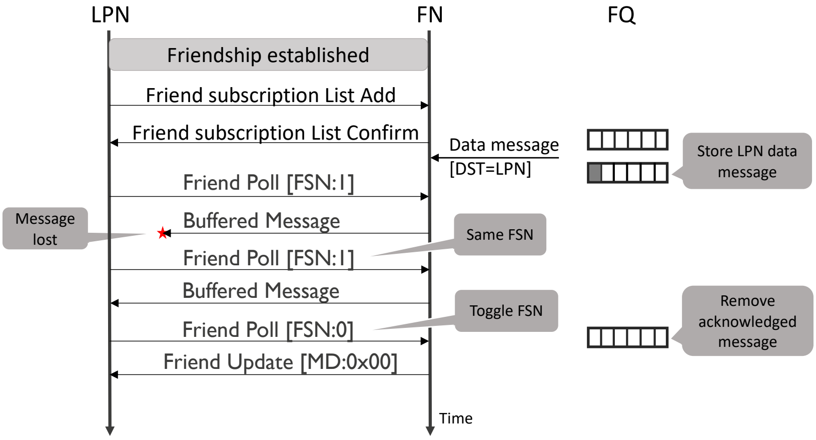


Figure 2.3: Network messages between the LPN and FN.

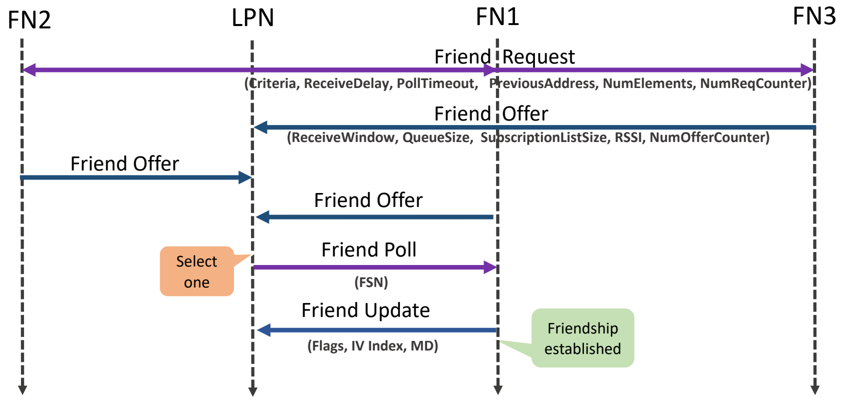

FN2
LPN
FN1
FN3
Friend Request
(Criteria, ReceiveDelay, PollTimeout, PreviousAddress, NumElements, NumReqCounter)
Friend Offer
(ReceiveWindow, QueueSize, SubscriptionListSize, RSSI, NumOfferCounter)
Friend Offer
Friend Offer
Select one
Friend Poll
(FSN)
Friend Update
(Flags, IV Index, MD)
Friendship established

Figure 2.4: Friendship establishment procedure of BLE.

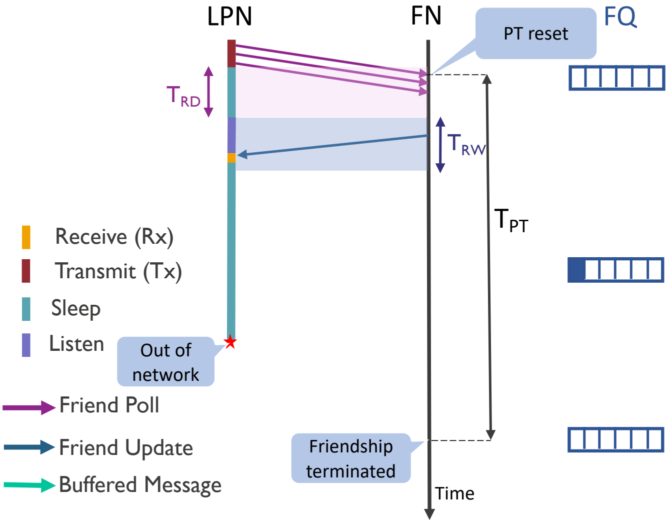


Figure 2.5: Friendship termination by the FN.

tablishment procedure. The LPN presents the timer values for RD and PT in the friend request message, whereas the FN proposes the RW in the friend offer message. The RD and RW can be configured with a maximum value of 255 ms and the PT with a maximum of 96 hours. The usage of these timers can be seen in Figure 2.5. RD is the delay between the LPN sending the FP message and when it starts listening to a response from the FN. During the RD, the LPN can turn off its radio and switch to SLEEP mode. During RW, the LPN expects the data and listens actively for it. The LPN uses more energy due to active listening during RW, and therefore, a short RW should be configured if possible. The PT is used as a timeout to ensure that its friendship relationship is not kept alive indefinitely when an LPN leaves the network. If the FN does not receive any poll request from an LPN before the PT timer expires (since the last poll), it terminates the friendship with that LPN and removes the corresponding FQ. The friendship can also be terminated if the LPN does not receive any response from the FN after sending the FP for a certain number of attempts. Moreover, Bluetooth mesh provides infrastructure-less connectivity without any overheads of time synchronization [17]. As the nodes (except LPNs) are assumed to be always listening, so there is no need to synchronize their listening timeslots. Moreover, the LPN always initiates communication with the FN itself (i.e., receiver-initiated communication), so the LPN does not need to synchronize explicitly for receiving data either [18]. The timing of the RW is done relative to the FP message sent by the LPN to its FN.

2.3 Wake up Radio in Bluetooth Mesh LPN

As mentioned above, the data communication is initiated by the LPN based on its polling mechanism and independent on the arrival of DL data packets at the FN. As such, the DL latency can become very high if the LPN sleeps longer without sending FP (which is often desirable to minimize energy consumption). To solve this, adding a WuR to an LPN provides the FN with the ability to notify it in advance about the incoming data packet before the LPN initiates polling. The LPN enabled with a WuR can continuously listen to the incoming Wake up Signal (WuS), as WuR idle listening power consumption is orders of magnitude lower than that of the main radio. The FN sends a WuS to the LPN whenever it receives a network packet corresponding to that LPN in an empty FQ. The LPN (having sufficient energy) polls until it receives an FU message indicating there are no messages buffered in the FQ. Upon receiving the WuS by the LPN's WuR, it subsequently triggers the main radio to initiate sending the FP message. This reduces the need to send frequent periodic FP messages and polls only when data is buffered in the FQ. Thus, it not only reduces the DL latency but also reduces the energy wastage due to polling without receiving buffered data. Let us now see how the WuR enable LPN is different that the traditional Poll-based LPN.

2.3.1 Poll-Based LPN

In the default Bluetooth Mesh friendship mechanism, the LPN initiates the communication by sending an FP message. Figure 2.6 shows a sequence diagram representing the communication steps. As the friendship is established, the LPN can now switch to SLEEP mode, and all the incoming DL packets of the LPN will be buffered in the FQ. The LPN can retrieve them from the FN by periodically waking up and sending an FP message in all three broadcast channels (37 to 39). On reception of the FP, the FN gets a time window of RW (T_{RW}) seconds, starting (T_{RD}) seconds after receiving the FP to send a response [19]. The arrival time (AT or T_{AT}) is defined as the total listening time until the LPN starts receiving the message. While sending the FP, the LPN needs to toggle the FSN value to indicate the acknowledgement of the previous message. If the FN receives the same FSN field value as the last FP message received from the FN, it responds with the same message it has previously sent unless the FQ head is updated. This can be updated by any new incoming packet(s) whenever it arrives at a full FQ. In this case, the oldest packet (FQ head) would be discarded (dequeued), and the newly arrived packet would be added in the queue tail (queued). This strategy is called 'push-out'. Therefore, if the previously sent message has been discarded (on the arrival of new packets), then the oldest entry in the FQ shall be sent. And if the FQ is empty, an FU is sent with MD equal to zero. To avoid receiving frequent, continuous FU messages, a timer poll interval timer (PI or T_{PI}) can be considered. On receiving an FU message, an LPN needs to wait until the PI expires to send the next FP message. As a batteryless LPN, it may happen that it is unable to send the FP exactly at the expiration of PI due to energy constraints.

2.3.2 WuR-Based LPN

The sequence diagram shown in Figure 2.7 presents the communication system to receive DL data packets by a WuR-based LPN. The WuR remains turned-on operating at orders of magnitude lower power consumption than regular radio while listening for a Wake-up Signal (WuS) [20]. This state where the LPN actively listens using the WuR, keeping the main radio in DEEP SLEEP mode, is named as wake-up state. When the FN receives a message for an LPN in an empty FQ, it initiates a communication event by sending a WuS.

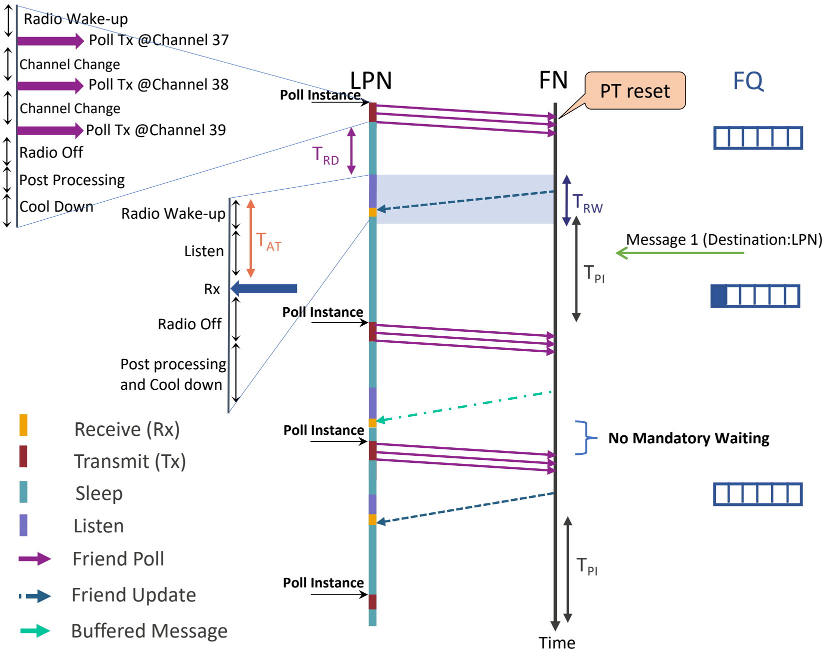


Figure 2.6: Sequence diagram to receive downlink (DL) data packets using poll-based communication.

Upon receiving the WuS, the WuR can interrupt the main radio of the LPN to start the process to request the message.

2.4 Introduction to Batteryless Solution

The sustainability of the IoT nodes can be improved by replacing the use of batteries with eco-friendly devices to power them. Even the adoption of rechargeable batteries could not offer a truly sustainable solution, as these batteries also support only a limited number of charge-discharge cycles with a reduction in performance and energy storage density. Thus, rechargeable batteries also need to be replaced over time, which is an undesirable and tedious task. Therefore, one possible heading for this problem is to make wireless sensor nodes batteryless. It means the nodes harvest the needed energy from ambient sources.

There are two commercial designs for energy harvesting batteryless systems: direct-usage-based and energy storage-based. In the direct-usage-based design, the harvester's output directly powers the load (IoT devices). However, this design either does not power the device when the harvesting power is less than required or wastes energy when the harvestable energy is more than the required energy to power the device successfully. On the other hand, the energy-storage-based design in the presence of capacitor(s) can be charged for future use whenever the harvester harvests energy. Therefore, in this thesis, the energy storage-based design for the batteryless solution is considered.

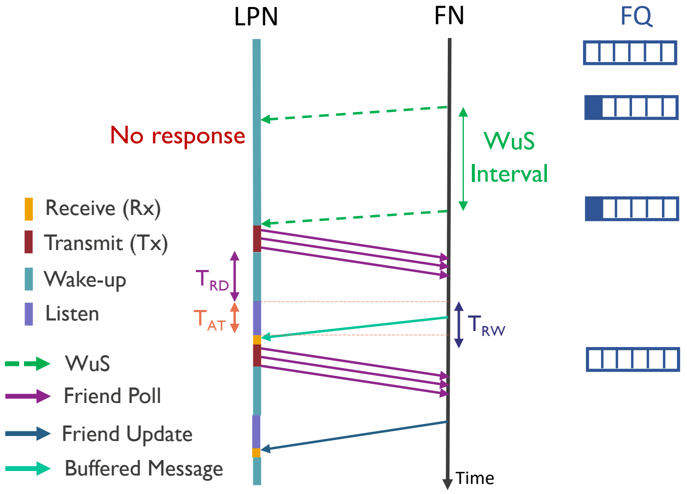
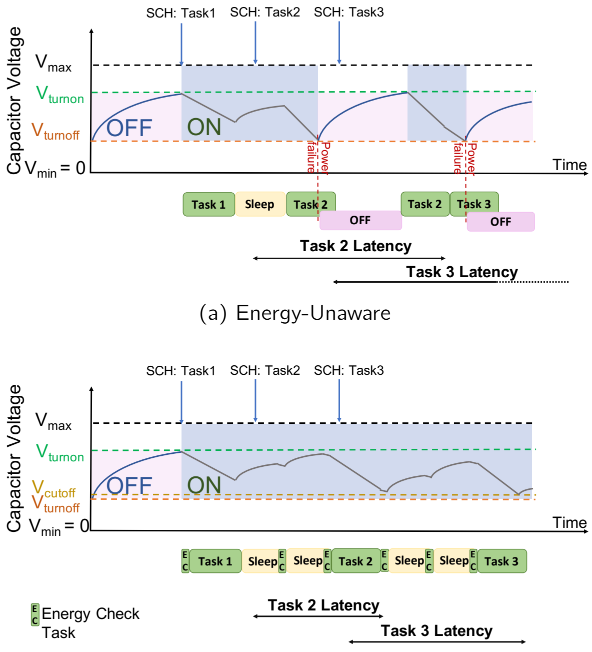


Figure 2.7: Sequence diagram to receive DL data packets using WuR-based communication.

The power requirement of most IoT nodes for short- or long-range wireless communication protocols is that they consume tens of μW in DEEP SLEEP mode and a few tens to hundreds of mA during data transmission and reception. As practically, the IoT nodes remain in a DEEP SLEEP state for most of the time and spend most of the energy only during the communication; the required QoS for IoT applications can be achieved. Depending on the ambient source availability, the batteryless devices are powered intermittently. It means these devices frequently lose power altogether, causing task execution to be halted prematurely and the loss of all the threads and data stored in volatile memory. Moreover, on a restart, they need to start their application from the beginning.

2.4.1 Energy-Unaware and -Aware LPN

Considering storage-based design, when the capacitor voltage drops below a turn off voltage ($V_{turnoff}$), the node turns off as shown in Figure 2.8, as such, the LPN can work being either unaware of the available voltage at the capacitor or being aware of it. This intermittent behaviour can be observed more in an energy-unaware solution because the batteryless device will try to execute the tasks as soon they are scheduled. As shown in Figure 2.8a, during the execution of Task 2 and Task 3, the capacitor's voltage drops below $V_{turnoff}$, and the device loses power. This forces the device to reschedule these tasks upon the next wake-up time (i.e., the task execution was delayed until the capacitor again reaches V_{turnon} voltage). However, the device can avoid this with an energy-aware solution by checking the capacitor voltage before executing any task. The batteryless device should start the task only if its capacitor has acquired a sufficient threshold voltage ($V_{threshold}^{Task_i}$) corresponding to that task ($Task_i$) and should not drop its voltage below a cut-off voltage V_{cutoff} after the execution. The cut-off voltage should be selected above $V_{turnoff}$ to ensure the LPN avoids turning off. However, if its voltage is below the required threshold value, it can periodically check the available voltage at a fixed energy check interval (ECI) or alternatively, it can also use a predictor to predict when the voltage will reach a threshold. Figure 2.8b



(a) Energy-Unaware

(b) Energy-aware

Figure 2.8: Comparing Energy-Unaware and Aware solution

shows such behaviour, where to execute Task 3, the device wakes up twice to check the energy. Intrinsically, energy-aware solutions can have a great potential to improve network performance. Comparing both the solutions in Figure 2.8, it can be possible that the energy-aware solution executes more tasks within the same time interval. However, it needs to spend a small amount of energy to check the voltage of the capacitor. Therefore, the ECI should be selected optimally so that the device does not end up spending energy only to check the voltage rather than to execute the scheduled tasks.

The energy-aware LPN can use an ultra-low-power comparator with power consumption in the order of pico-watts [21] to determine if it has reached $V_{threshold}$. The comparator can be configured to generate events every time the LPN voltage reaches $V_{threshold}$ or V_{off}. An UP event is generated whenever the LPN's voltage reaches $V_{threshold}$ and a DOWN event whenever it falls below V_{off} [22]. The LPN's MCU can take appropriate actions based on these generated events. Using such an ultra-low-power comparator would not significantly impact the LPN voltage. Alternatively, it can use voltage-divider circuit. An Analog-to-digital Converter (ADC) pin of the IoT device can be leveraged to read the capacitor voltage. However, the capacitor can not be connected directly to the board pin due to two reasons. First, the capacitor can supply a burst of current to this pin that could damage the device and second, the IoT device can read inputs only up to the maximum operating voltage. Therefore, the solution is to have a voltage divider that can reduce both the current and voltage. The schematic diagram of the voltage divider circuit [23] is shown in Figure 2.9. The resistors need to be selected such that the voltage entered from $R1$ should be in the range permissible by the operating device. The output voltage is based on Ohm's law and calculated as Equation 2.1.

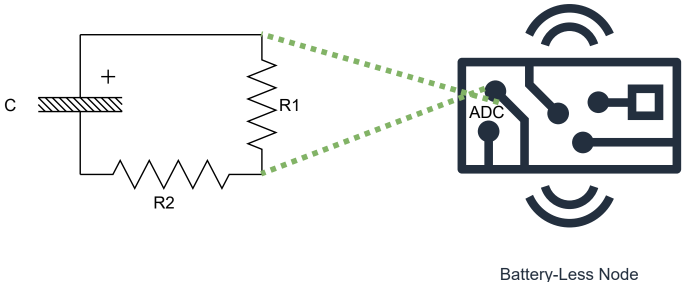


Figure 2.9: Voltage divider circuit

$$V_{R1} = \frac{V_c \times R_1}{R_1 + R_2}, \tag{2.1}$$

where, V_{R1} is the input voltage to the board, V_c is the actual capacitor voltage, and R_1 and R_2 are the resistance of the resistors.

Now, depending on the bit size of the ADC, the measured ADC value can be converted to the actual voltage of the capacitor. For an N bit ADC, working at a standard operating voltage of V_{op}, the conversion formula is mentioned in Equation 2.2.

$$V_c = \frac{V_{op} \times N_{adc}}{2^N}, \tag{2.2}$$

where, N_{adc} is the digital value on the ADC pin.

Moreover, to reduce the continuous current consumption due to the added resistors, MOSFETS are used as a circuit switch. An N-channel MOSFET can close the circuit when a positive voltage is applied to its gate from a GPIO pin of the device.

2.4.2 Batteryless Poll-Based LPN

Using a batteryless LPN, this communication becomes more complicated because the LPN might not always have sufficient stored energy to perform polling at a predefined polling interval, or immediately after receiving a buffered message. As the harvesting power can influence the polling interval, instead of a fixed predefined polling interval, a batteryless LPN uses it as a minimum interval and waits longer to poll if not enough energy is available. Moreover, an energy aware LPN will not poll immediately after receiving a buffered message, but only as soon as it has harvested enough power. During the process of receiving a buffered data packet, the LPN could experience multiple shutdown events as the required capacitor voltage to successfully receive a buffered data packet could be higher than its voltage at which it starts the communication. In simplicity, the LPN should start sending the FP only if it has acquired a sufficient threshold voltage ($V_{threshold}$). Initiating the poll from $V_{threshold}$, it can receive at least one buffered data packet successfully without reaching below the device turn-off voltage V_{off}. Thus, the overall DL latency for a batteryless LPN could increase (wrt. battery-powered) as the packets need to wait in the FQ until the LPN acquires the voltage $V_{threshold}$ and the poll interval expires.

2.4.3 Batteryless WuR-Based LPN

Similar to the poll-based LPN, to prevent the LPN from getting shut down during the communication, it wakes up the main radio to send the FP message only if it has sufficient voltage ($V_{threshold}$). Thereafter, the LPN follows the same procedure as mentioned for the direct Poll scheme (cf., Section 2.3.1) by sending the FP messages in the three advertisement channels. To account for the potential loss of a WuS (e.g., because the LPN does not have enough energy to receive it or is temporarily shut down), the WuS is retransmitted by the FN if no FP is received within a pre-configured WuS interval timer. While sleeping, the current consumption of a WuR-based LPN does not only include the sum of the MCU sleep current $I_{sleepMCU}$ and main radio sleep current $I_{sleepMR}$, but also the WuR listening current $I_{listenWuR}$.

2.5 Related work

2.5.1 Batteryless Bluetooth Mesh Performance

Energy harvesting has been extensively explored to support the sustainable operations of IoT systems. Various types of ambient energy have been utilized, such as RF, solar and wind energy. An overview of the energy harvesting technologies for various applications is presented in [24–26]. There are already some solutions that explore energy harvesting to enable batteryless Bluetooth Mesh solutions.

Silva *et al.* [27] presented a prototype where an implantable batteryless glucose monitor node is capable of transmitting data to a mobile phone when supplied power wirelessly using an RF energy source. However, their design does not consider the Bluetooth mesh friendship feature and sends only UL data. The batteryless Bluetooth Mesh prototype, presented by Radhika et al. [28], is based on the ambient Frequency Modulation (FM) band and the dedicated BLE RF source. The solution enabled the UL data packet communication at an interval of 1,200 s harvesting from the FM band and at 90 s with the dedicated RF source. However, using a BLE device equipped with a 50 mF capacitor harvesting from a 5 m apart Global System for Mobile Communication (GSM) mobile, Sanislav et al. [29] were able to support a data interval of 30 s. Liu *et al.* recently proposed another RF-power based prototype for indoor BLE beaconing [30]. They test the device for both Uplink (UL) and DL data. However, they do not analyze the effect of the capacitor size nor the performance in terms of DL latency.

Fraternali et al. [31] explored the design space of batteryless sensor nodes using commercial off-the-shelf components harvesting from ambient light. They study the BLE node lifetime, quality of service and energy availability for different capacitor sizes under varying lighting conditions. However, their focus was also only on the UL data. Nilsson *et al.* also conducted indoor and outdoor experiments to evaluate power consumption, range, throughput, and UL latency of a Bluetooth mesh network [32]. A control loop system is proposed where the nodes can request the energy source to provide energy, enabling them to replenish their storage (capacitor or battery) [33]. This method can ensure a certain level of quality-of-service for an IoT application. They also present a prototype using BLE nodes equipped with a photovoltaic energy harvester, which communicate and request a recharge from an indoor smart lighting system. Meli et al. [34, 35] demonstrated the ability to execute the BLE protocol using a batteryless node. They showed that the batteryless BLE devices powered by a small solar cell in an indoor environment could sense data such as temperature and humidity, which can later be sent to other nodes in the network. Wu et al. [36] also evaluate the performance of a subcutaneous solar energy harvester using a batteryless BLE prototype with a temperature sensor. Jeon et al. [37] proposed design principles for an ambient light energy harvesting BLE beacon capable of perpetual operation in an indoor environment based on the Nordic-nRF51822. Jang [38] investigated the output of the power management board (PMB) for a BLE node that harvests power from light using small capacitors of 200 μF and 420 μF. He found that by increasing the advertising interval or the capacitance, the number of charge-discharge cycles decreases. Nevertheless, he considers a prototype where the PMB starts charging the capacitor only when the solar panel output is higher than 1.55 V. He also conducted a similar study based on piezoelectric harvesting technology [39].

A scalable solution is designed by Witham et al. [40] to detect water leakage using a batteryless BLE beacon powered by a customized sensor. Using a small capacitor of 3.9 mF and harvesting peak short-circuit harvesting current of 8.1 mA, the design was able to advertise data packets at an interval of 100 ms. Brunecker et al. achieved up to 32 mW harvested power using a 6 dBi gain transmitter antenna placed at 5 cm from the harvesting receiver and up to 1.5 mW at 40 cm distance [41]. Zhong et al. [42] implemented their design of an implantable batteryless bladder pressure monitor system that monitors bladder storage in real-time and transmits the feedback signal to the external receiver through BLE. They use a four-coil wireless energy transmission method, which supports a power transmission range of up to 7 cm.

In summary, most research on batteryless BLE nodes is performed considering UL data without considering the friendship feature or DL communications. It means whenever the device harvests energy, it turns on and sends the data periodically until it runs out of energy and turns off again. However, there are two papers that considered the friendship feature. While they considered battery-powered LPNs, rather than batteryless ones. The first paper is by Alvarez *et al.* [43]. They performed an in-depth analysis of security of the friendship concept. They demonstrated that denial-of-friendship attacks could reduce the LPN's battery life by 70-fold. The second paper is by Darroudi *et al.* [44]. They presented an analytical model to study the energy performance parameters of a battery-operated Bluetooth Mesh LPN. These parameters include current consumption, lifetime, and energy consumed per delivered bit for different PT and RW values. They calculate the evaluation parameters based on the average value of time and current consumption for all the radio states and fixed UL data rate interval, in contrast to DL traffic does not require the polling feature and friend queue.

2.5.2 Wake-up Radio with Bluetooth Mesh node

The detailed surveys on WuR hardware and protocols are discussed in [45, 46]. Many researchers presented the design to couple a low power WuR with BLE to explore its potential. Many WuRs are also implemented to be triggered using BLE packets [47–49].

Giovanelli et al. [50] evaluated the possible benefits of integrating a WuR in the BLE protocol stack. They observed that the use of a WuR reduces energy consumption and DL latency. The WuR decreases the DL latency by up to 40% in the case of connection-oriented communication when the number of devices is large (100+), while with few devices, the traditional approach performs better. Mikhaylov et al. [51] demonstrated that the WuR-based BLE could outperform the classic BLE solution (without WuR) if the maximum latency for data delivery is tolerable by the application does not exceed 2.1 s. Sanchez [52] also evaluated that WuR-based BLE performs better than the classic BLE for low frequency data rates. They performed the tests for battery-powered nodes, whereas our work focuses on batteryless nodes. This is expected to impact the results and conclusions significantly. Specifically, the low harvesting power density combined with the added power consumption of a WuR can worsen the intermittent behaviour of a batteryless device. Other works that have integrated the WuR capabilities in BLE are reported in [53–55]. These works focused on hardware design aspects of WuR-integration, while our work looks at protocol aspects instead. Liu et al. [56] presented an RF-based passive WuR enabled batteryless node. They observed that the energy harvested within 100 ms at a distance of 1 m from the RF energy transmitter is sufficient to transmit and receive 40 B long beacon messages in a range of 3 m. Whereas they target a fixed type of harvesting technology without considering the impact of the capacitor, this thesis investigates the optimal size of a capacitor for different ranges of harvested power.

2.5.3 Thesis Contribution

This thesis is the first to study the DL performance of batteryless Bluetooth Mesh LPNs. The unique contribution is the inclusion of the DL packet arrivals, which are first queued at the FN, and so the proposed model studies the queueing performance based on the Markov chain with Poisson distributed data arrivals. The model provides the opportunity to analyse the batteryless LPN powered by different capacitor sizes receiving different harvested powers for various Bluetooth mesh network parameters such as Friend queue size, packet size, or receive delay. The model is also compared to our developed simulator as presented in [57]. Moreover, it also considers combining the batteryless device with a WuR to optimize the DL latency further. The thesis also demonstrates the working prototype using commercial off-the-shelf components. Summarizing, the contributions made in this thesis include 1) investigation of hardware components; 2) design principles for selecting hardware components subject to varying environmental conditions and application requirements; 3) prototyping to prove its practicality; and 4) analyzing and comparing the energy-unaware and aware solution.

Chapter 3

Batteryless Bluetooth Mesh Model

The content of this chapter is based on:

- *A. K. Sultania, C. Delgado, C. Blondia and J. Famaey, "Downlink Performance Modeling and Evaluation of Batteryless Low Power BLE Node," in MDPI Sensors, 2022. [Impact Factor: 3.576]*

This chapter presents an analytical model to characterize the performance of a batteryless LPN that is powered by different harvesters and capacitor sizes. The DL data latency and packet delivery ratio (PDR) are evaluated and compared with the results of the simulator. The proposed model would help to optimally choose the correct configuration of the batteryless LPN for its network deployment.

3.1 Batteryless Bluetooth Mesh LPN

Let the LPN have already established the friendship with a neighbouring FN. So, to receive DL packets, the radio of an LPN can be in 15 different states as shown in Figure 2.6 (Radio Wake-up to Post-processing and Cool down). The voltage (energy) of a capacitor as a function of time, supporting various states of the radio as used is described next in this section.

3.1.1 Capacitor Voltage

A batteryless LPN is equipped with an energy harvester, a capacitor, a micro-controller unit (MCU), a main radio and an optional WuR. The batteryless circuit is introduced by Delgado et al. [58] to calculate the voltage of a batteryless device at a specific time, as shown in Figure 3.1. Assume the device maximum operating voltage is E (volt) and the harvester provides a power of P_h (watts) modelled as a real voltage source having an internal resistance r_i (ohm). In order to limit the power of the harvester, this internal resistance r_i is defined as E^2/P_h. The energy-consuming components are the MCU, main radio, WuR,

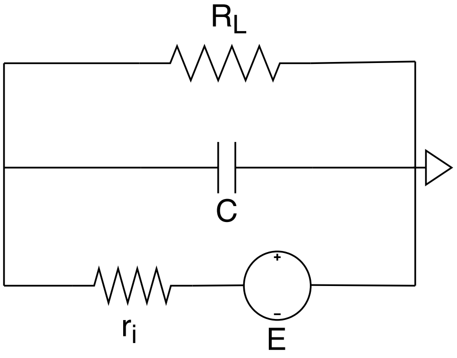


Figure 3.1: Electrical circuit model of a batteryless IoT device.

or any other peripherals that are modelled as a load resistance. The current consumption of these components varies with their operating states (e.g., sleep, active). Let the total current consumption of the LPN load at a time instant t be $I_L(t)$ (ampere) then according to Ohm's law, the load resistance of the LPN equals $R_L(t) = E/I_L(t)$. As the capacitor, the harvester and the LPN components have a parallel connection, the equivalent resistance of the LPN is calculated as Equation (3.1).

$$R_{eq}(t) = \frac{1}{\frac{1}{r_i} + \frac{1}{R_L(t)}} = \frac{R_L(t) \cdot r_i}{R_L(t) + r_i} = \frac{E^2}{P_h + E \cdot I_L(t)}. \tag{3.1}$$

By applying Kirchoff's voltage law to the circuit, the voltage across the load V(t+Δt) over a time period Δt starting at time t, given its voltage $V(t)$ at a time t, having a capacitor of C (farad) and a fixed resistance $R_{eq}(\Delta t)$ during the time interval Δt can be calculated as Equation (3.2) [58].

$$V(t+\Delta t) = E \cdot \frac{R_{eq}(\Delta t)}{r_i}\left(1 - e^{\frac{-\Delta t}{Req(\Delta t) \cdot C}}\right) + V(t) \cdot e^{\frac{-\Delta t}{Req(\Delta t) \cdot C}}, \tag{3.2}$$

After substituting the value of r_i and $R_{eq}(\Delta t)$ (Equation (3.1)) in Equation (3.2), the final voltage is derived as Equation (3.3).

$$V(t+\Delta t) = \frac{E \cdot P_h}{P_h + E \cdot I_L(\Delta t)}\left(1 - e^{\frac{-\Delta t \cdot (P_h + E \cdot I_L(\Delta t))}{E^2 \cdot C}}\right) + V(t) \cdot e^{\frac{-\Delta t \cdot (P_h + E \cdot I_L(\Delta t))}{E^2 \cdot C}}. \tag{3.3}$$

The formula mentioned in Equation (3.3) calculates the voltage change of an LPN while its state (and thus the current consumption) remains the same during a time interval Δt. It needs to be recalculated every time the LPN's state changes.

The model assumes an energy-aware LPN. As stated before, the friendship can be terminated if no response is received either after sending $\dot{N}$ FPs or if the time between two consecutive FPs is larger than T_{PT}. To avoid this, the P_h is considered such that it allows the capacitor to recharge itself from V_{min} to a threshold voltage level that is sufficient to perform a poll request-response cycle in less than T_{PT}. As T_{PT} can be chosen freely (up to X seconds/minutes/days), this assumption is realistic even when the harvested power is low. The time to send the FP when the batteryless LPN needs to have a required threshold voltage of V_{th}^{FP} to start the FP and the remaining voltage of the capacitor after an FP is calculated in Section 3.1.2.

3.1.2 Time to Next Poll

The time at which the next FP can take place depends on whether the FQ is empty or not and the available energy after the previous poll.

3.1.2.1 FQ is not empty

The voltage of the capacitor after all the polling stages can be calculated according to Equation (3.3) starting from an initial voltage (V_0) and considering 'Radio wake-up' as an initial state. Let V_f be the final voltage after all the polling radio stages (as shown in Figure 2.5) of the previous poll are executed. Therefore, there are two scenarios:

- $V_f \geq V_{th}^{FP}$: The next FP can be sent immediately and the time between two FPs (T_{FP}) is the sum of all the state timings as defined in Table 3.1 and is calculated as Equation (3.4). The capacitor voltage at the start of the next poll instance will be V_f.

  $$T_{FP} = \sum_{i=AllStates} T_i \tag{3.4}$$

- $V_f < V_{th}^{FP}$: As V_f is less than the required voltage to poll, the device needs to switch to SLEEP mode to charge the capacitor and avoid its voltage from reaching V_{cutoff}. Let the node sleep for the time T_{sleep}. Therefore, the time between two FPs has an extra wait time of T_{sleep} in addition to all the FP state's timings and is calculated in Equation (3.5).

  $$T_{FP} = T_{sleep} + \sum_{i=AllStates} T_i \tag{3.5}$$

  The capacitor voltage at the start of the next poll instance will be V_{th}^{FP} (after the capacitor is charged while the LPN is in SLEEP mode).

3.1.2.2 FQ is empty

If the FQ is empty, the LPN needs to wait for the time PI (T_{PI}) to poll, as it received an FU after the previous FP message. Therefore, in this case, the voltage V_f would be calculated after a sleep time of T_{PI}. Now again, there are two cases.

- $V_f \geq V_{th}^{FP}$: The time to poll has an extra delay of T_{PI} in addition to all the FP state's timings.

  $$T_{FP} = T_{PI} + \sum_{i=AllStates} T_i \tag{3.6}$$

The capacitor voltage at the start of the next poll instance will be V_f.

- $V_f < V_{th}^{FP}$: Similar to above, the node needs to switch to SLEEP mode to provide enough time to charge the capacitor. Therefore, the time between two FPs has an additional time T_{sleep} and T_{PI} in addition to all the FP state's timings.

$$T_{FP} = T_{PI} + T_{sleep} + \sum_{i=AllStates} T_i \tag{3.7}$$

The capacitor voltage at the start of the next poll instance is V_{th}^{FP}.

The threshold voltage of the FP (V_{th}^{FP}) can be calculated by taking the inverse of the function as mentioned in Equation (3.3) that means calculating V(t) knowing the value of $V(t+\Delta t)$ for all the states of the poll. The considered initial voltage should be V_{cutoff} and the starting state as 'Post-processing and Cool down' as mentioned in Table 3.1.

3.2 System Model

This section presents the analytical model of the system described in Section 3.1. The model is based on the Markov chain which is used to analyse a randomly changing system where it is assumed that future states do not depend on past states. That means the transition of states of a BLE node and each packet arrivals are independent. In the considered network also, the packet generation times are uncertain because they depend on many unknown factors such as user behaviour. Hence, a Markov chain enables us to model a dynamical system, which is defined as one that changes its state over time. In our case, it would be the radio state of a batteryless node. The Markov model considers all possible states as well as the transitions, rate of transitions and probabilities between them. Therefore, the model can accurately determine DL latency, FQ occupancy and the Packet Delivery Ratio (PDR).

3.2.1 The Queuing Model

The time interval between two FPs, when the FQ is not empty at the start of the interval is referred to as a *service*. The time interval between two FPs during which an FU is transmitted from the FN to the LPN (i.e., the FQ at the start of the interval is empty) is called a *vacation*. The system can be modelled as a finite capacity queue where the server takes repeated vacations. The duration of a service and a vacation depends on the LPN's capacitor voltage. Let the packets arrive at the FN according to a Poisson distribution with data arrival rate λ and the FN temporarily buffers them in a buffer (FQ) of capacity N. The vector $\mathbf{E}$ represents the discretized voltages that a capacitor can take between 1 and V_{max}. Hence, the duration of a service when the voltage equals $i \in \mathbf{E}$ can be denoted by $\mathbb{S}_i$ with distribution function $\mathbb{S}_i(t)$ and Laplace Transform (LST) $\mathbb{S}_i^*(\nu)$. Similarly, for the vacation, $\mathbb{W}_i$, $\mathbb{W}_i(t)$ and $\mathbb{W}_i^*(\nu)$ can represent the duration, the distribution function and its LST respectively. The duration of a service consists of all the radio states during polling and possibly followed by a sleep period to reach the threshold index V_{th}^{FP}. The radio states during the polling can be assumed to have a constant duration for a fixed size of the DL packet. Hence, $\mathbb{S}_i$ can be computed considering FQ is not empty and $\mathbb{W}_i$ considering it is empty as discussed in section 3.1.2. If at the start of a service, the voltage is equal to i, then the voltage at the next poll instance is denoted by $\mathbb{V}_i^{\mathbb{S}}$ and calculated using Equation (3.3) for all the polling radio states. Similarly, if the voltage is equal to i at the start of a vacation, its values at the next inspection instant denoted by $\mathbb{V}_i^{\mathbb{W}}$ also includes the SLEEP state for T_{PI}.

The probability $\mathbb{P}_{k,i}^{\mathbb{S}}$ and $\mathbb{P}_{k,i}^{\mathbb{W}}$ that k packets arrive during the service time and the vacation time respectively, starting from $\mathbf{E}$ equal to i can be calculated as Equation (3.8).

$$\begin{aligned}\mathbb{P}_{k,i}^{\mathbb{S}} &= \int_0^{\infty} \frac{(\lambda t)^k}{k!} e^{-\lambda t} d\mathbb{S}_i(t) \\ \mathbb{P}_{k,i}^{\mathbb{W}} &= \int_0^{\infty} \frac{(\lambda t)^k}{k!} e^{-\lambda t} d\mathbb{W}_i(t)\end{aligned} \tag{3.8}$$

The column vectors of the averages of the service times and the vacation times are given by Equation (3.9).

$$\begin{aligned}\overline{\mathbf{E}[\mathbb{S}]} &= (\mathbf{E}[\mathbb{S}_1], \mathbf{E}[\mathbb{S}_2], ..., \mathbf{E}[\mathbb{S}_{V_{max}}])' \\ \overline{\mathbf{E}[\mathbb{W}]} &= (\mathbf{E}[\mathbb{W}_1], \mathbf{E}[\mathbb{W}_2], ..., \mathbf{E}[\mathbb{W}_{V_{max}}])'\end{aligned} \tag{3.9}$$

3.2.2 Embedded Markov Chain

The system is observed at the poll instants (t_n, n=1,2, ..,∞). These poll instants are referred as *inspection instants*. Let Q_n be the number of packets in the FQ, and V_n be the available capacitor voltage at time instant t_n. This should be noted that V_n will never take values below $V_{startpoll}$ which is the minimum voltage available at a poll instant that allows all functions of the radio to receive a packet. Hence, the stochastic process (Q_n, V_n) at t_n is a discrete-time finite Markov Chain with state-space size equal to $(N+1) \cdot V_{max} - V_{startpoll} + 1$. The limiting probability distribution function of this Markov Chain is denoted by $\overline{p} = (\overline{p}_0, \overline{p}_1, ..., \overline{p_N})$. Let $(\overline{p}_k)_i$ denote the joint probability distribution of the voltage and the queue length as shown in Equation (3.10).

$$(\overline{p}_k)_i = \lim_{n \to \infty} Prob\Big(Q_n = k, V_n = i\Big), \tag{3.10}$$

where, $0 \leq k \leq N, i = V_{startpoll} .... V_{max}$ The transition matrix of this Markov Chain is given by Equation (3.11).

$$\mathbf{P} = \begin{bmatrix} B_0 & B_1 & B_2 & ... & B_{N-2} & B_{N-1} & \sum_{n=N}^{\infty} B_n \\ A_0 & A_1 & A_2 & ... & A_{N-2} & \sum_{n=N-1}^{\infty} A_n & 0 \\ 0 & A_0 & A_1 & ... & A_{N-3} & \sum_{n=N-2}^{\infty} A_n & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & .... & A_0 & \sum_{n=1}^{\infty} A_n & 0 \\ 0 & 0 & 0 & .... & 0 & \sum_{n=0}^{\infty} A_n & 0 \end{bmatrix} \tag{3.11}$$

where, B_n represents the transitions starting from an empty system and ending up with n packets in the FQ at the next inspection point. Whereas, A_n represents the transitions starting from a non-empty system. The $(V_{max} - V_{startpoll} + 1) \cdot (V_{max} - V_{startpoll} + 1)$ size-matrices A_n are defined by Equation (3.12).

$$(A_n)_{i,j} = \begin{cases} \mathbb{P}^{\mathbb{S}}_{n,i}, & \text{for i=}[V_{startpoll}, V_{max}], \text{ j=}\mathbb{V}^{\mathbb{S}}_i \\ 0, & \text{elsewhere} \end{cases} \tag{3.12}$$

Similarly, B_n can also be calculated as Equation (3.13).

$$(B_n)_{i,j} = \begin{cases} \mathbb{P}^{\mathbb{W}}_{n,i}, & \text{for i=}[V_{startpoll}, V_{max}], \text{ j=}\mathbb{V}^{\mathbb{W}}_i \\ 0, \text{elsewhere} \end{cases} \tag{3.13}$$

The probability distribution satisfies Equation (3.14).

$$\begin{aligned} \overline{p} &= \overline{p} \cdot P \\ \overline{p} \cdot \overline{e} &= 1, \end{aligned} \tag{3.14}$$

where, $\overline{e}$ is the (N+1)$\cdot(V_{max} - V_{startpoll} + 1)$ unit vector. The steady-state probability of the system occupancy at arbitrary time instants has been derived in [59]. Let T_{FP} be the average time between two consecutive inspection instants given by Equation (3.15).

$$T_{FP} = \sum_{n=1}^{N} \overline{p}_n \cdot \overline{\mathbf{E}[\mathbb{S}]} + \overline{p}_0 \cdot \overline{\mathbf{E}[\mathbb{W}]}. \tag{3.15}$$

The queue occupancy distribution i.e. the probability that there are n packets in the FQ is then given by Equation (3.16) [59].

$$\begin{aligned} &O_n = \frac{1}{\lambda \cdot T_{FP}} \cdot \left[\overline{p}_n \cdot \overline{e} - \overline{p}_0 \cdot B_n \cdot \overline{e} \right], \qquad \text{where, n=[0,N-1]} \\ &\text{And} \\ &O_N = 1 - \frac{1}{\lambda \cdot T_{FP}} \cdot \left[1 - \overline{p}_0 \cdot \overline{e} \right]. \end{aligned} \tag{3.16}$$

Therefore, the total percentage of packet loss is equal to $100 \cdot O_N$.

3.2.3 Performance Metrics

In this section, the metrics, average FQ occupancy and average DL latency are computed.

- *Average FQ occupancy*: The FN uses a *push-out* strategy to manage its FQ, which means when a packet arrives at a full FQ, the oldest packet in the FQ would be dropped. However, the number of packets in the *push-out* system is the same even if the arriving packet is dropped (*drop-tail*). Since the probability that an arriving packet finds a full FQ is equal to O_N, the average system occupancy in a 'drop-tail' system is given by Equation (3.17).

$$\mathbf{O}_{DT} = \sum_{n=1}^{N} n \cdot O_n. \tag{3.17}$$

Now, the average system occupancy in a 'push-out' system ($\mathbf{O}_{PO}$) is equal to the $\mathbf{O}_{DT}$.

- *Average DL latency*: The actual arrival rate in a drop-tail system is given by $(1-O_N)\cdot\lambda$ and applying Little's well-known formula [60], the average response time in a drop-tail system is given by Equation (3.18).

$$\mathbf{T}_{DT} = \frac{\mathbf{O}_{DT}}{(1-O_N)\cdot\lambda}. \tag{3.18}$$

However, the waiting time of a packet in the system using push-out is quite different from the waiting time in a system using drop-tail. In a drop-tail system, the packets behind a tagged packet do not impact the delay of that packet. However, in a push-out system, packets arriving after the tagged packet may, in case they arrive at a full queue, push out the first packet in the queue, and hence influence the delay experienced by the tagged packet. The waiting time in the push-out system consists of two parts: firstly, the time interval between the packet arrival instant (at FQ) and the next inspection instant (i.e., until the end of the vacation if it arrives during a vacation or until the end of a service if it arrives during a service), and secondly, the time interval between this inspection instant and the start of the transmission of the packet.

  - *Between arrival instant and next inspection instant*: There are two scenarios, either the packet arrives during the vacation period or the service period.

    * **An arbitrary time instant falls in a vacation**: Let $P_n^{\omega}(t)$ be the probability that an arbitrary time instant falls in a vacation period, that there are n packets (maximum N) present in the system at time t and that the remaining vacation time is T_v that satisfies $t \leq T_v < t+dt$. The LST of this probability has already been determined by Blondia [59]. Hence, the inverse LST can provide us the value of $P_n^{\omega}(t)$ and $P_N^{\omega}(t)$ as defined in Equation (3.19).

$$P_n^{\omega}(t) = \frac{1}{\lambda\cdot T_{FP}} \cdot \Bigg[\overline{p}_0 \cdot \overline{e} \cdot \mu_{\mathrm{E}[\mathbb{W}]}(t) \cdot (-1)^{n+1} \cdot e^{\lambda\cdot(t-\mathrm{E}[\mathbb{W}])} \cdot \frac{(t-\mathbf{E}[\mathbb{W}])^n}{n!} - \sum_{l=0}^{n} \big(\overline{p}_0 \cdot B_l \cdot \overline{e} \cdot \lambda^{n-l+1} \cdot (-1)^{n-l+1} \cdot e^{\lambda\cdot t} \cdot \frac{t^{n-l}}{(n-l)!}\big)\Bigg].$$

And

$$P_N^{\omega}(t) = \frac{\overline{p}_0 \cdot \overline{e}}{T_{FP}} \cdot \left[1-\mu_{\mathrm{E}[\mathbb{W}]}(t)\right] - \sum_{n=0}^{N-1} P_n^{\omega}(t), \tag{3.19}$$

where, n = [0,N-1] and $\mu_{\mathrm{E}[\mathbb{W}]}$(t) is a Heaviside function that outputs (0) if t $\leq \mathbf{E}[\mathbb{W}]$ and (1) if t $> \mathbf{E}[\mathbb{W}]$.

$P_{n,k}^{\omega}(t)$ represents the probability that an arbitrary time instant falls in a vacation period, that there are n packets (maximum N) present in the system at time t, and there are k packet arrivals (maximum $N-1$) during the remaining vacation time T_v. This probability is given by Equation (3.20).

$$P_{n,k}^{\omega}(t) = e^{-\lambda\cdot t}\frac{(\lambda\cdot t)^k}{k!} \cdot P_n^{\omega}(t). \tag{3.20}$$

The probability that at an arbitrary time instant there are n packets (maximum $N-1$) in the system and that during the remaining time of the vacation k packets (maximum $N-1$) have arrived is given by Equation (3.21).

$$P^{\Omega}_{n,k} = \int_0^{\infty} P^{\omega}_{n,k}(t)dt = \frac{1}{\lambda T_{FP}} \sum_{l=0}^{n} \Big(\overline{p}_0 \cdot B_l \cdot \overline{e}\Big) \cdot \\ \left[\sum_{k=0}^{n-j} \frac{(\lambda \cdot \mathbf{E}[\mathbb{W}])^{n-l+k+1} \cdot (-1)^{n-l}}{(n-l)! \cdot k! \cdot (n-l+k+1)} \right] \tag{3.21}$$

And

$$P^{\Omega}_{N,k} = \frac{\overline{p}_0 \cdot \overline{e}}{\lambda T_{FP}} \cdot \left[1 - e^{-\lambda \cdot \mathrm{E}[\mathbb{W}]} \cdot \sum_{i=0}^{k} \frac{(\lambda \cdot \mathbf{E}[\mathbb{W}])^{i}}{(i)!} \right] - \sum_{n=0}^{N-1} P^{\Omega}_{n,k}.$$

The average waiting time until the next inspection instant of a packet that arrives at an arbitrary time instant in a vacation period is then given by Equation (3.22).

$$T^{\Omega}_{n,k} = \int_0^{\infty} t \cdot P^{\omega}_{n,k}(t)dt = \frac{1}{\lambda T_{FP}} \sum_{l=0}^{n} \Big(\overline{p}_0 \cdot B_l \cdot \overline{e}\Big) \cdot \\ \left[\frac{(\lambda \cdot \mathbf{E}[\mathbb{W}])^{n-l+k+1} \cdot (-1)^{n-l} \cdot \mathbf{E}[\mathbb{W}]}{(n-l)! \cdot k! \cdot (n-l+k+2)} \right]$$

And

$$T^{\Omega}_{N,k} = \frac{\overline{p}_0 \cdot \overline{e} \cdot (k+1)}{\lambda^2 \cdot T_{FP}} \cdot \left[1 - e^{-\lambda \cdot \mathrm{E}[\mathbb{W}]} \cdot \sum_{i=0}^{k+1} \frac{(\lambda \cdot \mathbf{E}[\mathbb{W}])^{i}}{(i)!} \right] - \sum_{n=0}^{N-1} T^{\Omega}_{n,k}. \tag{3.22}$$

∗ *An arbitrary time instant falls in a service*: Let $P^{\pi}_{n}(t)$ be the probability that an arbitrary time instant falls in a service period, that there are n packets (maximum N) present in the system at time t and that the remaining service time T_s satisfies $t \leq T_s < t + dt$. Based on the Laplace transform [59] $P^{\pi}_{n}(t)$

can be calculated as Equation (3.23).

$$
\begin{aligned}
P_n^{\pi}(t) = {} & \frac{1}{\lambda \cdot T_{FP}} \cdot \Bigg[\sum_{l=1}^{n} \Big(\overline{p}_1 \cdot \overline{e} \cdot \lambda^{n-l} \cdot \mu_{\mathrm{E}[\mathbb{S}]}(t) \cdot (-1)^{n-l+1} \cdot e^{\lambda \cdot (t - \mathrm{E}[\mathbb{S}])} \cdot \\
& \frac{(t - \mathbf{E}[\mathbb{S}])^{n-l}}{(n-l)!} \Big) - \sum_{l=1}^{n} \Big(\overline{p}_{l-1} - \overline{p}_0 \cdot B_{l-1} \cdot \overline{e} \cdot \lambda^{n-l} \cdot (-1)^{n-l+1} \cdot \\
& e^{-\lambda \cdot t} \cdot \frac{t^{n-l}}{(n-l)!} \Big) \Bigg], \\
& \text{And} \\
& P_N^{\pi}(t) = \frac{\sum\limits_{l=1}^{N-1} \overline{p}_1 \cdot \overline{e}}{T_{FP}} \cdot \Big[1 - \mu_{\mathrm{E}[\mathbb{S}]}(t) \Big] - \sum_{n=1}^{N-1} P_n^{\pi}(t). \\
& \qquad \text{where, n=[0,..,N-1].}
\end{aligned}
\tag{3.23}
$$

Then, the probability that a packet arrives in a service period and n packets are present in the FQ at time t and k packets arrive in the remaining service time is given by Equation (3.24).

$$
P_{n,k}^{\pi}(t) = e^{-\lambda t} \frac{(\lambda t)^k}{k!} \cdot P_n^{\pi}(t), \tag{3.24}
$$

where, n=[0,N] and k=[0,N-1].
Similar to Equation (3.21), the probability ($P_{n,k}^{\Pi}$) that at an arbitrary time instant there are n packets in the system and that during the remaining time of the service k packets have arrived can be calculated as Equation (3.25).

$$
\begin{aligned}
& P_{n,k}^{\Pi} = \int_0^{\infty} P_{n,k}^{\pi}(t) dt \\
& = \frac{1}{\lambda T_{FP}} \sum_{l=1}^{n} \Big(\overline{p}_{l-1} - \overline{p}_0 \cdot B_{l-1} \cdot \overline{e} \Big) \cdot \Big[\frac{(\lambda \cdot \mathbf{E}[\mathbb{S}])^{n-l+k+1} \cdot (-1)^{n-l}}{(n-l)! \cdot k! \cdot (n-l+k+1)} \Big]. \\
& \text{And} \\
& P_{N,k}^{\Pi} = \frac{\sum_{l=1}^{N} \overline{p}_l \cdot \overline{e}}{\lambda T_{FP}} \cdot \Big[1 - e^{-\lambda \cdot \mathrm{E}[\mathbb{S}]} \cdot \sum_{i=0}^{k} \frac{(\lambda \cdot \mathbf{E}[\mathbb{S}])^i}{(i)!} \Big] - \sum_{n=1}^{N-1} P_{n,k}^{\pi}.
\end{aligned}
\tag{3.25}
$$

The average waiting time of a packet that arrives at an arbitrary time instant in a service period is then given by Equation (3.26).

$$T^{\Pi}_{n,k} = \int_0^{\infty} t \cdot P^{\pi}_{n,k}(t)dt = \frac{\sum\limits_{l=1}^{N}(\overline{p}_l \cdot \overline{e})}{\lambda T_{FP}} \cdot \left[\frac{(\lambda \cdot \mathbf{E}[\mathbb{S}])^{n-l+k+1} \cdot (-1)^{n-l} \cdot \mathbf{E}[\mathbb{S}]}{(n-l)! \cdot k! \cdot (n-l+k+2)}\right].$$

And

$$T^{\pi}_{N,k} = \frac{\sum_{l=1}^{N} \overline{p}_l \cdot \overline{e} \cdot (k+1)}{\lambda^2 \cdot T_{FP}} \cdot \left[1 - e^{-\lambda \cdot \mathrm{E}[\mathbb{S}]} \cdot \sum_{i=0}^{k+1} \frac{(\lambda \cdot \mathbf{E}[\mathbb{S}])^i}{(i)!}\right] - \sum_{n=0}^{N-1} T^{\pi}_{n,k}. \tag{3.26}$$

- **Between this inspection instant and the start of the transmission of the packet**:
  Let $T^s_{n,k}$ be the mean waiting time of a packet in the FQ at the *end* of a service that will be served and that has n packets (maximum $N-2$) ahead and k packets (maximum $N-n-2$) behind, it is given by Equation (3.27).

$$\begin{aligned}
T^s_{0,k} &= 0, \text{ for k=[0,N-2]} \\
T^s_{1,k} &= \sum_{j=0}^{N-k-2} \left(T^s_{0,k+j} + \mathbf{E}[\mathbb{S}] \cdot \frac{(\lambda \cdot \mathbf{E}[\mathbb{S}])^j}{j!} \cdot e^{-\lambda \cdot \mathrm{E}[\mathbb{S}]}\right), \text{ for k=[0,N-3]} \\
T^s_{n,k} &= \sum_{j=0}^{N-n-k-1} \left(T^s_{n-1,k+j} + \mathbf{E}[\mathbb{S}] \cdot \frac{(\lambda \cdot \mathbf{E}[\mathbb{S}])^j}{j!} \cdot e^{-\lambda \cdot \mathrm{E}[\mathbb{S}]}\right) + \\
&\sum_{j=N-n-k}^{N-k-2} \left(T^s_{N-k-2-j,k+j} + \mathbf{E}[\mathbb{S}] \cdot \frac{(\lambda \cdot \mathbf{E}[\mathbb{S}])^j}{j!} \cdot e^{-\lambda \cdot \mathrm{E}[\mathbb{S}]}\right), \text{ for k=[0,N-n-2]} \\
&\text{and n=[2,N-2].}
\end{aligned} \tag{3.27}$$

  Similar to Equation (3.27), the mean waiting time of a packet in the FQ at the end of a vacation period $T^v_{n,k}$ that will be served and has n packets ahead (maximum $N-1$) and k packets (maximum $N-n-1$) behind can be calculated.

The total waiting time of a packet arriving at an arbitrary instant at the FN or average DL latency that is served eventually in a system using the push-out strategy, is given by Equation (3.28).

$$\begin{aligned}
\mathbf{T}_{PO} = &\sum_{k=0}^{N-1} \left(T^{\Omega}_{0,k} + P^{\Omega}_{0,k} \cdot T^v_{0,k}\right) + \sum_{n=1}^{N-1} \left[\sum\nolimits_{k=0}^{N-n-1} \left(T^{\Omega}_{n,k} + P^{\Omega}_{n,k} \cdot T^v_{n,k}\right) + \right. \\
&\left.\sum\nolimits_{k=N-n}^{N-1} \left(T^{\Omega}_{N,k} + P^{\Omega}_{N,k} \cdot T^v_{N-k-1,k}\right)\right] + \sum_{k=0}^{N-2} \left(T^{\Pi}_{1,k} + P^{\Pi}_{1,k} \cdot T^s_{0,k}\right) + \sum_{n=2}^{N-1} \\
&\left[\sum\nolimits_{k=0}^{N-n-1} \left(T^{\Pi}_{n,k} + P^{\Pi}_{n,k} \cdot T^s_{n-1,k}\right) + \sum\nolimits_{k=N-n}^{N-2} \left(T^{\Pi}_{N,k} + P^{\Pi}_{N,k} \cdot T^s_{N-k-2,k}\right)\right]
\end{aligned} \tag{3.28}$$

3.3 Evaluation

This section presents the evaluation of the DL latency and PDR results using the analytical model and the simulator. Firstly, the description of the simulation setup is presented. This is followed by comparing the results of our analytical model and the simulator. The model is implemented in Matlab R2020b, while the simulation program is written in Python 3.9.6. Thereafter, these results are analyzed in detail based on parameters such as harvesting power, capacitor size, data arrival rate and poll interval timer of a batteryless Bluetooth Mesh LPN.

Remark that the execution time of the model in Matlab scales with the size of the transition matrix. Therefore, if it is known that at a specified harvesting power, the capacitor would be unable to recharge itself up to V_{max}, the matrix size can be reduced by assuming a lower value of V_{max}. This behaviour can be easily perceivable for low harvesting powers in combination with large capacitor sizes.

3.3.1 Simulation Setup

A Python-based simulator to imitate the friendship communication mechanism of the batteryless LPNs as shown in Figures 2.6 and 2.7 is implemented. The simulator follows an event-based packet scheduling and is capable of reproducing the Bluetooth Mesh radio activities such as sending FP messages and FQ buffered data packets. It also uses the same capacitor model to imitate the capacitor behaviour as mentioned in Section 3.1. The flow chart of the simulator is presented in Figure 3.2. Each experiment of the simulator is run until a total of 25,000 packets have been generated according to a Poisson arrival process. To request and receive a buffered data packet, the LPN follows the sequence of states listed in Table 3.1, in the order from top to bottom. The table also mentions the execution time and the current consumption of the corresponding states of the LPNs. These time and current consumption values can be used as Δt and $I_L(X)$ to calculate the capacitor's voltage at any time using Equation (3.3). The time and the current consumption was measured at 1.8V supply voltage on the Nordic nRF52840 Bluetooth Mesh devkit [61] using the Nordic-Power Profiler Kit-II (PPK-II) [62] while running the developed LPN application. All the states of the FP and receiving message mentioned in Table 3.1 can be seen in Figure 3.3.

The parameters listed in Table 3.2 are used to evaluate the model. As such, the values of V_{cutoff} and V_{max} are taken considering one of the configurations of a representative off-the-shelf power management unit [63]. The following performance metrics are considered in the comparison:

- Packet delivery ratio (PDR): The ratio of packets successfully received by the LPN compared to the total reaching the FQ.
- Downlink packet latency: The average latency to receive a packet by the LPN from the time it arrives at the FQ.

Table 3.1: State sequence for a poll request and response of an LPN.

Identifier	**Radio State**	**Duration [µs]**	**LPN Current cons.[mA]**	**Comments**
1	Radio wake-up	2940	0.587	Prepare for poll transmission
2	TX (at 4dBm)	471	9.09	Tx advertisement packet on channel 37
3	Change channel	29	7.78	Change radio frequency
4	TX (at 4dBm)	380	9.15	Tx advertisement packet on channel 38.
5	Change channel	29	7.57	Change again the radio frequency
6	TX (at 4dBm)	384	9.16	Tx advertisement packet on channel 39.
7	Radio off	34	6.60	Main radio in OFF state
8	Post processing	420	2.17	
9	Cool down	20820	0.00653	Prepare to switch to the SLEEP state
10	Sleep	RD	0.00896	Node in SLEEP mode
11	Wake-up prescan	1910	1.13	Wake-up for Rx
12	Listen	AT	8.68	Actively Listen for incoming DL packet (no data being received)
13	Scan message	900 (or 1160)	9.64	Rx 24 B FQ data (or 22 B FU)
14	Radio off	389	4.69	Main radio in OFF state
15	Post-processing and Cool down	23,410	0.00619	Set up the sleep timer for the next state and switch to SLEEP state

Table 3.2: Simulation parameters.

Parameters	**Symbol**	**Values/Range**
Poisson packet arrival rate	μ_p	[1, 10, 30, 60, 120] s
Harvesting power	P_h	[0.1, 0.15,0.25,0.3,0.35, 0.2,0.4,0.6,0.8,1] mW
Capacitor size	C	[0.1, 0.2,0.3,0.4,0.5,1] mF
Poll interval	T_{PI}	[1, 10, 20, 40, 60, 80, 100, 150, 200, 250, 300, 400, 500, 600, 800, 1000] s
Turn-off voltage	V_{cutoff}	2.8 V
Max operating voltage	V_{max}	4.5 V
Friend Queue size	N	[8,16] packets
Receive Delay	T_{RD}	255 ms
Arrival time (max up to T_{RW})	T_{AT}	0 ms
Tx power level		0 dBm

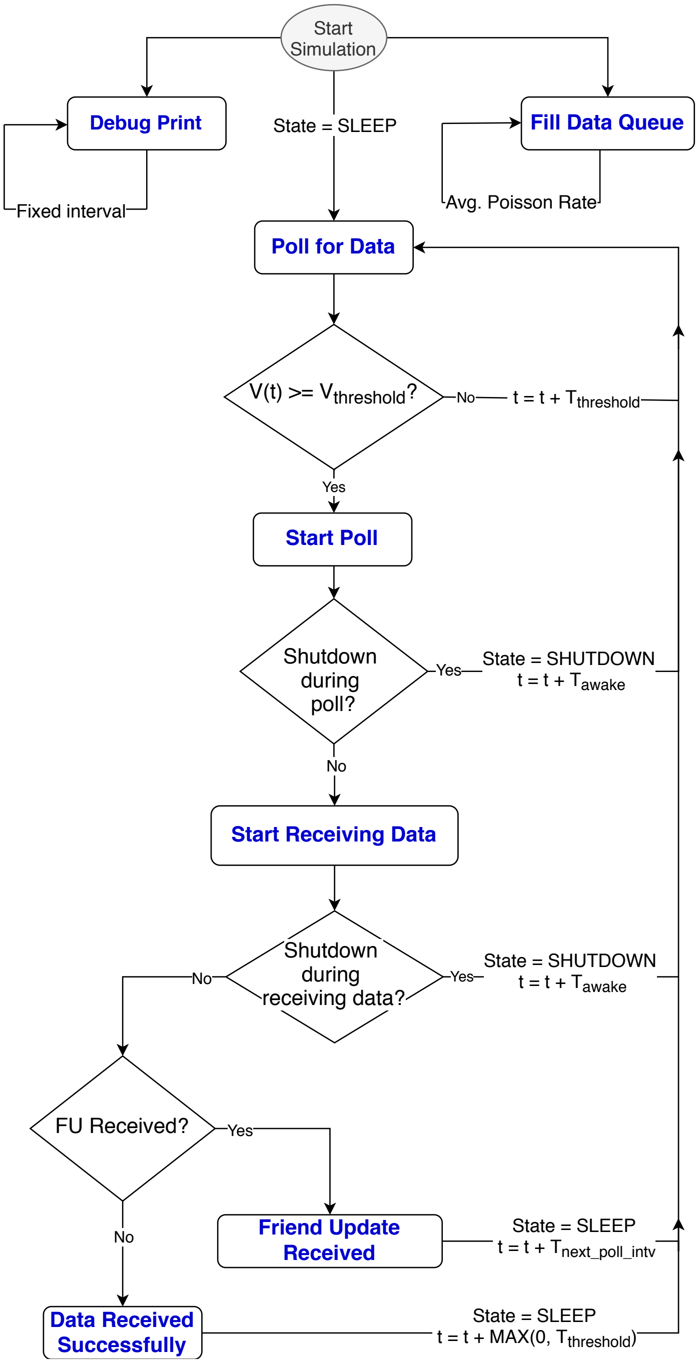
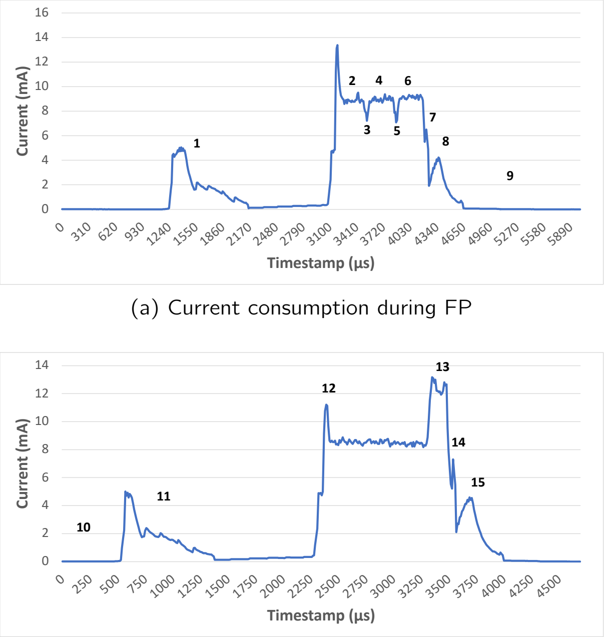


Figure 3.2: Flowchart of the simulator for batteryless LPN communication based on periodic polling. (V(t): Voltage of LPN at the time t, T_{awake}: The time after which LPN wakes up, $T_{threshold}$: The time to achieve $V_{threshold}$, $T_{next\_poll\_intv}$: The time of next polling activity.



(a) Current consumption during FP

(b) Current consumption during scan

Figure 3.3: Current consumption of nRF52840 with the state number as defined in Table3.1.

3.3.2 Model Validation

A total of 5640 test cases to compare the analytical model and the simulation results are executed. The test cases encompass all possible parameter combinations listed in Table 3.2. The harvesting power is considered for harvesting techniques which are low indoor light or human body temperature, that generate up to 1 mW. To avoid longer latency delay, the poll interval is considered only up to 10 times the data rate. These test cases analyze only DL data. The cumulative distribution shown in Figure 3.4 presents the relative deviation in PDR and DL latency between the simulation and the model results.

The mean difference across all test cases is 2.23% and 0.099% for DL latency and PDR, respectively. A high relative difference in PDR value ($\geq 2\%$) is observed when the PI is set to a relatively larger value than the data arrival rate. For these cases, the relative difference in latency is around 13%. Moreover, the large relative deviation in latency value ($\geq 15\%$) is also observed for the low P_h when the PDR is less than 100%. However, the deviation in latency is higher for the lowest FQ sizes (8 or less). For such cases, the deviation in PDR is only 0.58%.

There are a total of 92.73% and 100% test cases for DL latency and PDR, respectively, with a percentage difference less than 10%. By analyzing the results for the PDR and DL latency of the model and simulation, it is evident that the model's accuracy is high. However, solving the model is computationally complex for large FQ sizes (32 or more) and the relative mean difference in the DL latency test cases for the lowest FQ size (2 packets)

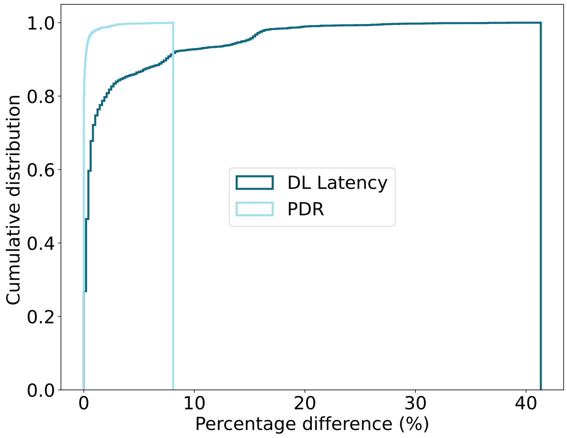
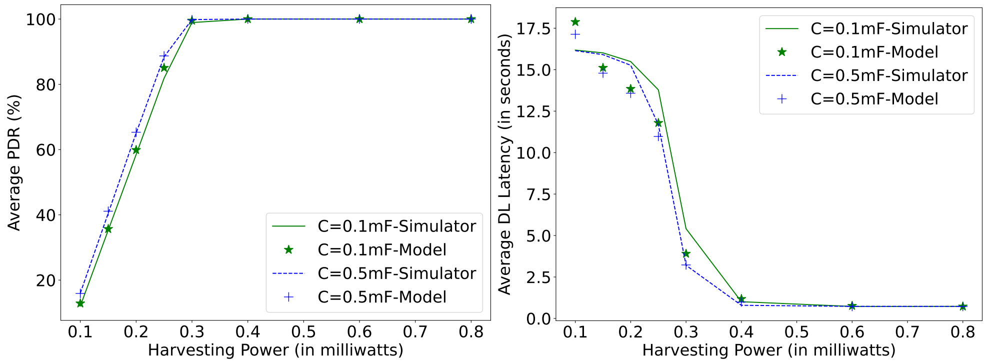


Figure 3.4: Cumulative distribution of difference in Model and Simulation results for both the DL latency and PDR metrics.

is higher compared to larger queue sizes.

3.3.3 Result Analysis

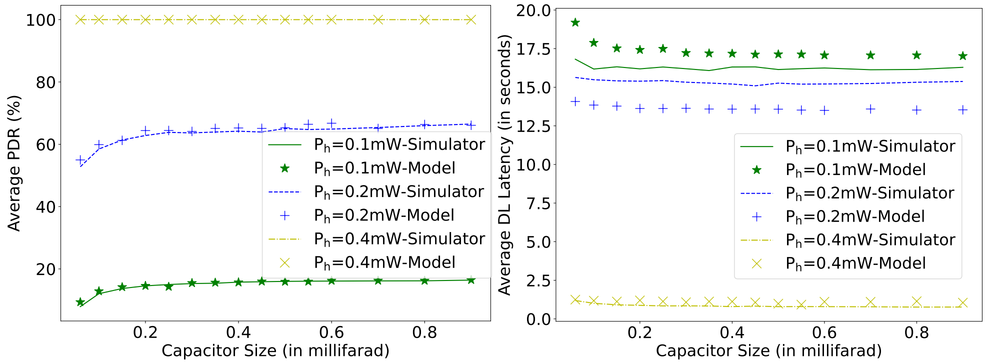
Figures 3.5, 3.6 and 3.7 plot the results of the analytical model and the simulation showing the effect of all the dependent parameters. It is clear that the analytical model and simulation follow the same behaviour, and the results match closely in all the plotted graphs. It can be observed from Figure 3.5 that as the harvesting power increases, the average PDR and DL latency improve until they reach their corresponding maximum achievable values. The improvement in the PDR and DL latency accelerate faster for low harvesting power until the PDR crosses 80% (0.25 mW in Figure 3.5a and Figure 3.5b). Further, an increase in the harvesting power improves the DL latency at a higher rate than the PDR because at such harvesting power; the capacitor can recharge faster again to V_{th}^{FP} to receive packets. As soon as the PDR and DL latency reach their corresponding best achievable values, an increase in the harvesting power does not affect the results because the capacitor voltage never drops below V_{th}^{FP} and therefore, the node does not need to go into SLEEP mode to recharge its capacitor. For an arrival rate of 1s, this harvesting power point is 0.6 mW as shown in Figure 3.5. With the decrease in arrival rate, the capacitor gets more time to recharge itself. Therefore, for an arrival rate of 10 s, the mentioned harvesting power point value reduces to 0.3 mW (not shown in figures). The size of the capacitor also plays a vital role to achieve the desired performance. As a larger capacitor can store more energy, the PDR for the 0.5 mF capacitor is higher than for 0.1 mF. However, the increase of the capacitor size also affects its charging time. Therefore, the performance does not improve much with the increase in the capacitor size after a certain optimal capacitor size for each specific harvesting power. There is an optimal capacitor size for each set of parameters. This can be observed from Figure 3.6b that at a harvesting power of 0.1 mW, the optimal capacitor size where the DL latency is lowest is 0.35 mF, and for 0.2 mW of harvested power, it is 0.45 mF. An increase in the capacitor size higher than 0.5 mF shows reduced growth in the PDR (harvesting power 0.1 and 0.2 mW). Also, when the PDR is 100%, the



(a) Average Packet delivery ratio

(b) DL Latency

Figure 3.5: Comparing Simulator and Model for different harvesting powers at μ_p=1 s, T_{PI}=1 s, N=16, T_{AT}=0 ms.



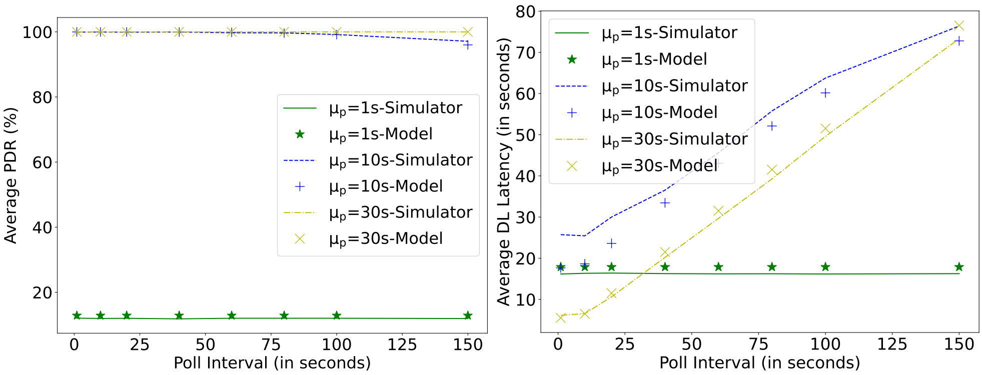
(a) Average Packet delivery ratio

(b) DL Latency

Figure 3.6: Comparing Simulator and Model for different capacitor sizes and harvesting power at μ_p=1 s, T_{PI}=1 s, N=16, T_{AT}= 0 ms.

increase in the capacitor size would improve the DL latency (yellow line at P_h 0.4 mW) but not substantially.

Figure 3.7 shows that for a fixed harvesting power (0.1 mW), with the decrease in Poisson arrival rate for downlink packets, the PDR can be improved to 100%. This is because the capacitor gets ample time to charge itself before the subsequent transmission. However, the achieved latency can increase with the increase in the poll interval timer. As the node has to wait for this PI once it receives the FU. Therefore, the higher the PI, the longer the arriving packets wait in the FQ to be polled. As such, when the data arrival rate is 10 s, some packets are also being dropped for large PI (150 s). Although frequent polling consumes additional energy, the considered harvesting power (0.1mW) is sufficient to support a PI as low as 1 s without negatively affecting PDR. Moreover, the DL latency at the arrival rate of 1 s is not affected by the increase in PI because there are always packets waiting in the



(a) Average Packet delivery ratio

(b) DL Latency

Figure 3.7: Comparing Simulator and Model for different Poll Intervals at C=0.1 mF, P_h=0.1 mW, N=16, T_{AT}=0 ms.

FQ to be polled, and the LPN never receives an FU message.

If the size of the FQ decreases, then the number of dropped packets increases for higher PI. This can be observed in Figure 3.8. The smaller the FQ, the lower the PDR. Another interesting parameter of the LPN is the AT, which also affects the power consumption and DL latency, as the LPN needs to listen during the configured time actively. With the increase in AT, the LPN wastes energy in idle listening on the incoming channels. Figure 3.9 shows its effect on DL latency and PDR. It can be observed that a slight increase in AT can drastically affect the PDR and so the DL latency. However, when the data arrival rate is low, the LPN has sufficient time to recharge its capacitor to maintain a PDR of 100%, but the increase in AT affects the DL latency exponentially. As seen for the Poisson arrival rate of 30 s, the latency increases 60-fold from 0.65 s at 0 s AT to 38.43 s at AT of 255 ms. At a high Poisson arrival rate (i.e. 1 s), the impact of the AT is higher because the LPN has less time to charge its capacitor and needs to receive more packets compared to lower Poisson data arrival rates. Increasing the AT further affects the performance marginally. Therefore, AT should be configured as small as possible to achieve high LPN performance. However, a small AT value provides less time to the FN to send a response to the LPN, which in turn affects the scalability (i.e., it causes missed response opportunities when multiple LPNs expect a response simultaneously).

Furthermore, from Figure 3.10 it can be observed that an LPN having a capacitor of 0.1 mF and a harvesting power of 0.1 mW can receive almost all the data packets when the arrival rate is 10 s for poll interval timers lower than 60 s. Further, decreasing the data arrival rate (from 7s to 30s) would improve the latency because fewer packets are entering into the FQ, and the capacitor gets more time to recharge itself. Similarly, it improves the PDR when the data arrival rate decreases from 1s to 7s. But the DL latency increases in this interval because with the decrease in data interval fewer packets are entered into the FQ and thus wait longer to be polled.

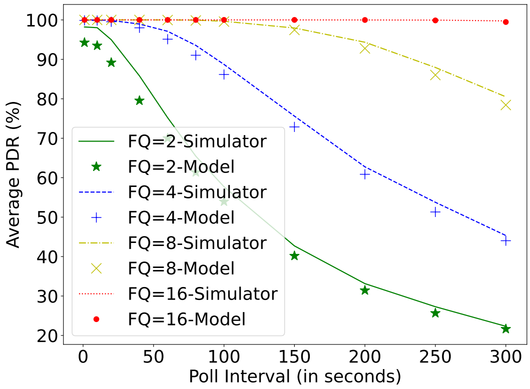
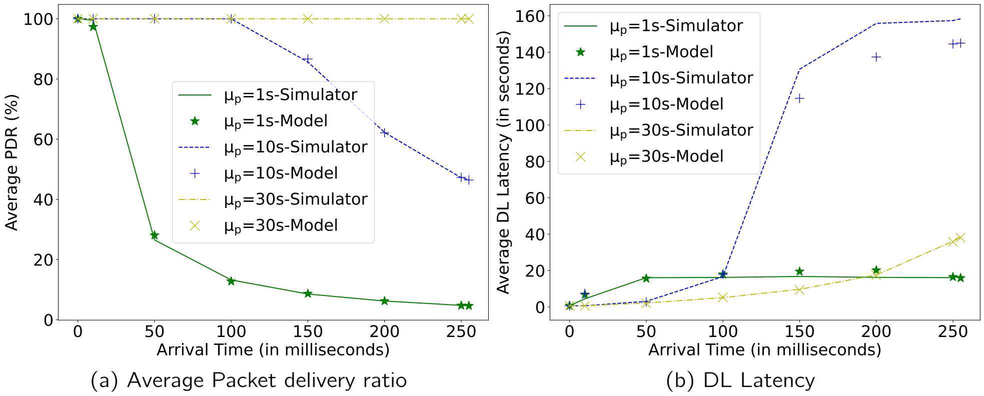


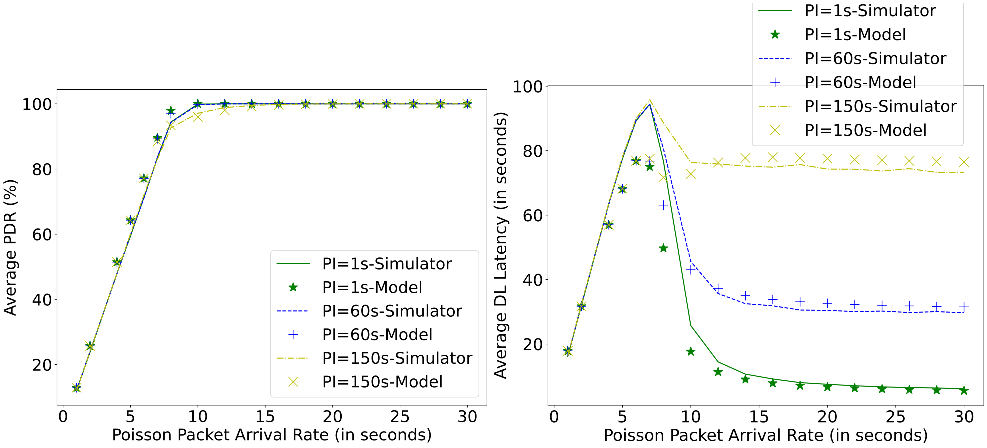
Figure 3.8: Comparing Simulator and Model for different Poll Intervals and FQ at C=0.1 mF, P_h=0.1 mW, μ_p=30 s, T_{AT}= 0 ms.



(a) Average Packet delivery ratio

(b) DL Latency

Figure 3.9: Comparing Simulator and Model for different Arrival Times (AT) at C=4 mF, P_h=1 mW, N=16, T_{PI}=1 s.



(a) Average Packet delivery ratio

(b) DL Latency

Figure 3.10: Comparing Simulator and Model for different Packet Arrival Rate at C=0.1 mF, P_h=0.1 mW, N=16.

3.4 Conclusion

An analytical model to calculate the DL latency and PDR for a batteryless Bluetooth Mesh LPN operating on power harvested from natural or artificial indoor light is presented. The model's accuracy was compared with Python-based simulation results for various parameters (i.e., capacitor size, harvesting power, data arrival rate, poll interval timer, arrival time, and friend queue size). The results derived from these two methods showed an average deviation of 2.23% in the results of DL latency and 0.09% in PDR. The graphs of different parameters showed diverse behaviour for DL latency and PDR. Whereas the model and simulation show the same pattern.

The results showed that a harvesting power of 0.6 mW can maintain a DL packet inter-arrival rate as low as 1s, and at 0.3 mW, the LPN can receive all the packets arriving at a rate of 10 s to an FQ of size 16. The smallest optimal capacitor size should be chosen depending on the available minimum harvesting power so that it can be recharged rapidly. The poll interval timer affects the DL latency linearly if the PDR is 100%. Therefore, it is important to choose the lowest possible poll interval timer that can be supported by the minimum harvesting power. If the harvesting power varies over time, the poll interval timer should be dynamically adapted. The external factors such as Friend Queue size also affect the performance of the batteryless LPN. A small queue size increases the packet loss. In our experiments, a queue size of 16 packets was enough to maintain 100% PDR for polling intervals as high as 300 seconds at a downlink packet arrival rate of 30 seconds. Moreover, the AT parameter should be configured to be as small as possible so that the LPN does not lose energy in idle listening on the incoming channels. A very low AT, however, may negatively affect scalability. For a harvesting power of 1mW, an AT up to 100 ms could be supported without decrease in PDR.

Chapter 4

Batteryless BLE Mesh Device using Wakeup Radio

The content of this chapter is based on:

- *A. K. Sultania, C. Delgado, and J. Famaey, "Enabling Low-Latency Bluetooth Low Energy on Energy Harvesting Batteryless Devices Using Wake-Up Radios," Sensors, vol. 20, no. 18, p. 5196, Sep. 2020. [Impact Factor: 3.567]*

This chapter evaluates the WuR-based batteryless Bluetooth Mesh LPN using the same capacitor model as described in Section 3.1.1 based on the previously introduced Python-based simulator. The WuR allows the FN to notify the LPN when new DL data is available by sending a WuS. This removes the need for frequent polling by the LPN and thus saves the little valuable energy available to the batteryless LPN. In this chapter, the standard Bluetooth Mesh duty-cycle based polling and WuR-based data communication between an FN and a batteryless energy-harvesting LPN are compared. This study allows optimising the LPN configuration (such as capacitor size, polling interval) based on the packet arrival rate, desired packet delivery ratio and DL latency at different harvesting powers.

4.1 Simulation Setup

The simulation setup is the same as defined in Section 3.3.1. However, the flow chart of the simulator to imitate WuR enabled LPN is presented in Figure 4.1. The time and the current consumption of the main radio and the MCU are based on Table 3.1 with the addition of WuR consumption. According to the data sheet of the AS3933 WuR [64], a WuR-based LPN consumes a current of 2.7 µA when one WuR channel actively listens to the incoming signals ($I_{listenWuR}$) and 12 µA while receiving them. Therefore, this additional current consumption of 2.7 µA is added to all the radio states of a WuR enabled LPN. When a WuS is received, the WuR spends 60 µs in the receive state, consuming 12 µA.

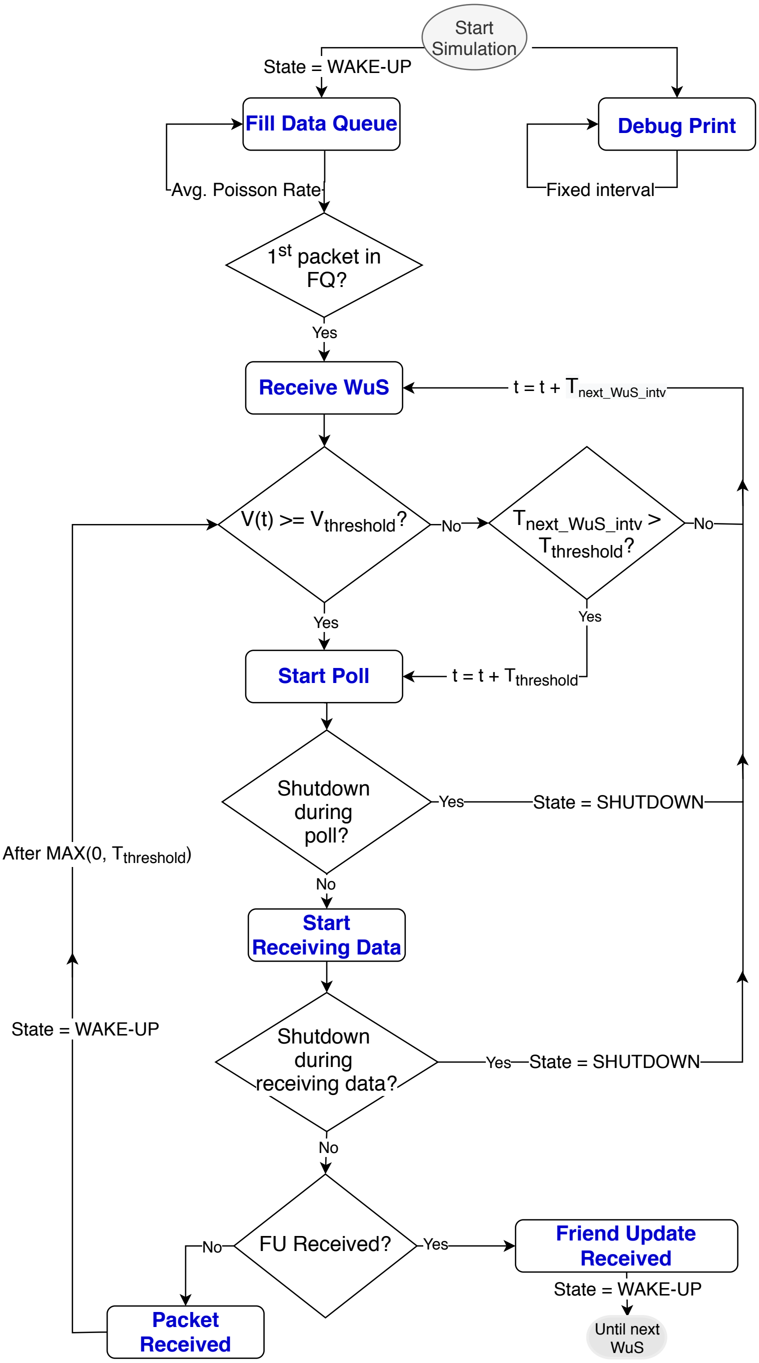


Figure 4.1: Flowchart of the simulator for batteryless LPN communication based on WuR. (V(t): Voltage of LPN at the time t, T_{awake}: The time after which LPN wakes up, $T_{threshold}$: The time to achieve $V_{threshold}$, $T_{next\_WuS\_intv}$: The time of next WuS).

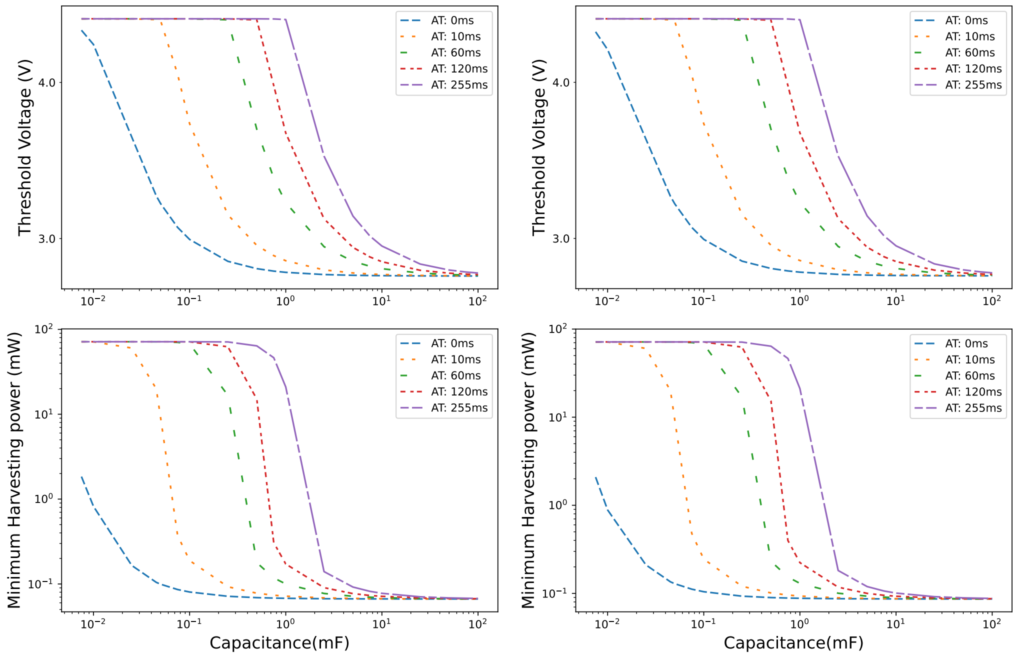
4.2 Results

This section compares the performance of both considered LPN communication approaches, i.e., direct Poll- and WuR-based.

4.2.1 Minimum Harvesting Power

As the LPN consists of many energy-consuming components, it is needed to know the minimum harvesting power at which the LPN can still charge its capacitor to the threshold voltage $V_{threshold}$ while in sleep mode (or WuR listening mode). Moreover, the harvested power needs to be enough to be able to complete at least one full polling cycle with a fully charged capacitor (i.e., $V_{threshold}$ needs to be lower than the maximum capacitor voltage), given a specific capacitance. The minimum harvesting power is calculated based on Equation (3.3), where the final voltage V(t + Δt) equals $V_{threshold}$ for the limit of Δt towards infinity at sleep state. The harvesting power becomes independent of the initial voltage and the capacitor size, as for an infinitely large Δt, the final voltage becomes $(E \cdot P_h)/(P_h + E \cdot I_L(\Delta t))$. However, to perform the Bluetooth Mesh friendship communication cycle (including sending a poll and receiving the response), different capacitor sized LPNs need a different threshold voltage. Accordingly, the required minimum harvesting power for different capacitance varies.

Considering the turn-off voltage V_{off} of 2.8 V, the threshold voltage and corresponding minimum harvesting power for different ATs are shown in Figure 4.2. The minimum harvesting power does not vary much for the capacitor of 50 mF and higher. It is around 66.6 µW for the LPN with only a main radio and around 86.4 µW for the LPN with a WuR along with the main radio for such large capacitors. Both the communication schemes presented for the LPN (with and without WuR) show similar behaviour for the change in the values of minimum harvesting power and threshold voltage for different capacitor sizes. As the LPN with WuR has higher current consumption, it requires more harvesting power, and the differences in the minimum harvesting power observed are up to 246.9 µW. Additionally, with the differences in the minimum harvesting power, the LPN with WuR has a lower threshold voltage with the difference compared to the LPN without WuR up to 0.014 V. In the WuR enabled LPN (cf. Figure 4.2b) at T_{AT} equal to 0 ms, the minimum harvesting power decreases exponentially from 2.08 to 0.0885 mW for a capacitor size between 7.5 and 750 µF. However, with the increase in AT, this exponential decrease shifts to larger capacitor sizes. As with the increase in AT, the listening time of the LPN increases, which increases its energy consumption. Therefore, high threshold voltages are required for smaller capacitor sizes resulting in the increase of the minimum harvesting power needed.



(a) Direct Poll-based communication

(b) WuR-based communication

Figure 4.2: Threshold voltage and minimum harvesting power for different arrival times (ATs) and capacitances.

4.2.2 Simulation Parameters

The harvesting powers can be grouped based on the capabilities of different harvesting techniques, which are small (0.075 to 0.099 mW), representing harvesting at the rate of electromagnetic or piezoelectric harvesting techniques, medium (0.1 to 1 mW) in line with indoor light, and large (1.1 to 100 mW) in line with techniques based on direct sunlight, mechanical movements, or thermal energy [65]. It is already observed from Figure 4.2 that the larger the capacitor used, the lower the minimum harvesting power required and the smaller the capacitor considered the higher threshold voltage required. Therefore, considering the exponential curve, a mid point is considered. A minimum capacitor size of 45 μF is considered for the experiment which requires a harvesting power greater than 0.14 μW. So, medium and large ranges are used for the result. The parameters defined in Table 4.1 are used to compare the LPN communication schemes (Poll based and WuR based). As such, V_{min} and V_{max} are taken considering the configuration of the Epeas AEM10941 power management board as 2.8 V and 4.5 V, respectively [63]. PDR and DL packet latency are considered in the comparison between the friendship communication mechanisms.

In each experiment, the values of the capacitor size and signal (WuS/Poll) interval that achieve the minimum DL latency while maintaining maximum PDR are calculated. For simplicity, continuous power harvesting is assumed. The packet losses are considered only due

Table 4.1: Simulation parameters.

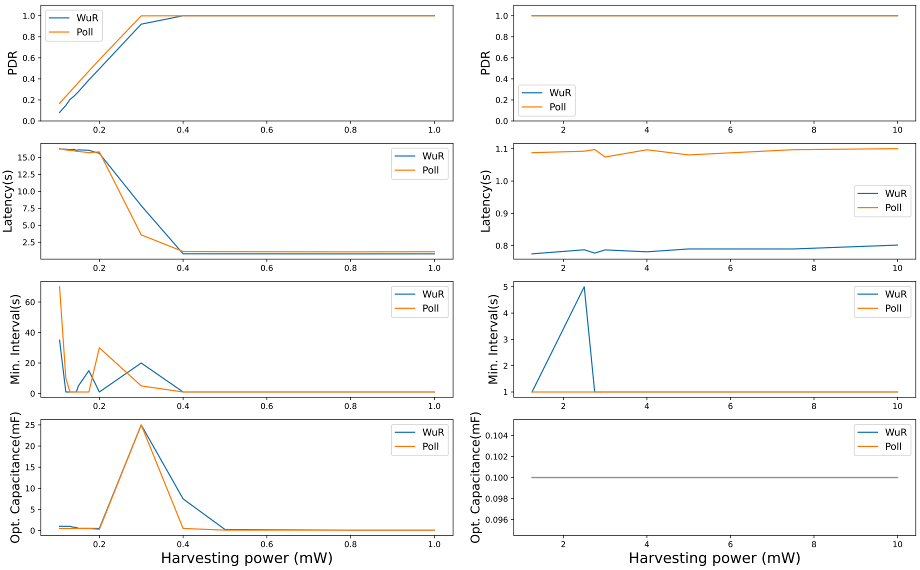
Parameters	**Symbol**	**Value**
Poisson packet arrival rate	μ_p	{1, 10, 60} s
Harvesting power	P_h	[0.14, 10] mW
Capacitor size	C	[0.045, 50] mF
Signal (WuS/Poll) interval	SI	[1, 400] s
Turn-off voltage	V_{off}	2.8 V
Max operating voltage	E	4.5 V
Friend Queue size	N_{fq}	16 packets
Receive Delay	T_{RD}	255 ms
Arrival time (max up to T_{RW})	T_{AT}	0 ms

to the LPN not having enough energy to receive the WuS or the packet, but not due to interference or collisions. There have been many studies evaluating the impact of interference on BLE under other technologies such as ZigBee, IEEE 802.11, and IEEE 802.15.4 [66–68]. Such interference causes the reception of erroneous packets, thereby affecting the DL latency. Collisions could happen when multiple LPNs are attached to an FN with a short advertisement interval or deployed near Bluetooth mesh nodes. The percentage of packet collisions with an advertising interval of 500 ms and having 7 Bluetooth Mesh nodes is less than 0.4% [69]. The batteryless LPNs (maximum 7) connected to an FN generally transmit or receive data at a much lower frequency than 500 ms. As a consequence, the PDR and DL latency of the LPNs would be negligibly affected by the presence of other nearby LPNs.

Generally, for a fixed harvesting power, with an increase in the signal (poll/WuS) interval (SI) values, the PDR decreases because the LPN polling frequency or receiving WuS notification frequency decreases. Furthermore, with the increase in the capacitance, the LPN can store more energy, and thus the PDR improves. Moreover, for a harvesting power, there exists a minimum optimal capacitance. More than that, further increasing the capacitor size above that value does not affect the PDR nor the DL latency. Therefore, for each value of harvesting power, this optimal capacitance and the minimum SI that can provide the highest PDR and lowest DL latency are calculated. All capacitance and SI combinations that deviate at most 5% of the maximum achievable PDR and lowest DL latency are also considered. This 5% deviation allows eliminating minor differences in the PDR and the DL latency that occurs for the optimal and higher capacitance values due to the randomness in Poisson packet arrivals at the FQ. By allowing this 5% deviation, the curves of optimal capacitance and interval can be smoothened. A value smaller than 5% did not provide the necessary smoothing effect, while a larger value would affect the optimality too much.

4.2.3 WuR-Based and Direct Poll-Based Friendship Protocol Performance

Figures 4.3–4.5 compare both the communication schemes for a Poisson packet arrival rate of 1, 10 and 60 s, respectively, showing the optimal capacitance and its corresponding PDR, DL latency value and minimum SI at different harvesting power values.

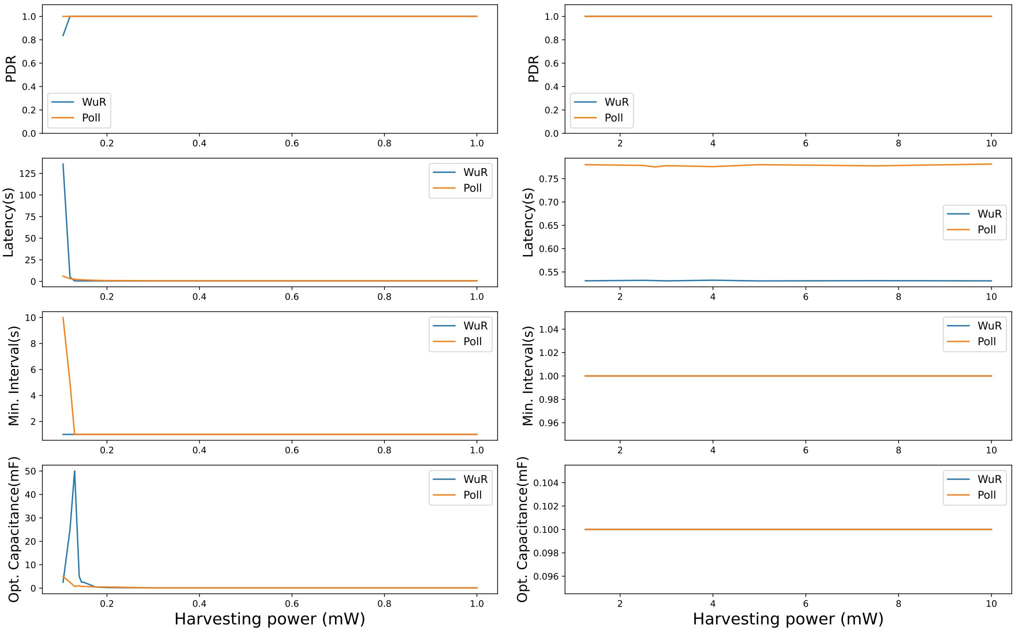


(a) Medium harvesting power

(b) Large harvesting power

Figure 4.3: Comparing WuR-based and direct poll-based communication schemes for a Poisson packet arrival rate = 1 s.

At the Poisson packet arrival rate of 1 s and for medium harvesting power values (cf. Figure 4.3a), with the increase in the harvesting power, the PDR improves, but the DL latency does not. This is because, with the increase in harvesting power, the LPN can reach the threshold voltage faster, and so it polls to receive the data more frequently. The DL latency values do not vary at these harvesting powers (up to 200μW) because most of the packets are dropped from the FQ due to the low PDR. Moreover, as with the increase in harvesting power, the capacitor charges faster, and therefore, the optimal capacitance decreases once the LPN starts achieving high PDR (more than 80%). The WuR-based LPN achieves lower PDR values because some of its energy is wasted in listening to the periodic WuS while remaining in sleep mode and this delays acquiring the voltage $V_{threshold}$ to start the communication. With the increase in the delay to poll, more packets are dropped from the FQ. The FN sends these WuSs assuming the previous WuS has not reached the LPN (WuS might be lost in transmission or the LPN might be shutdown) as it does not get any response from the LPN. Therefore, the optimal capacitance required for the WuR-based LPN is also a bit higher. Figure 4.3a shows that for low harvesting power (between 105 and 400 µW), direct Poll-based communication performs better, but neither approach achieves sufficiently high PDR. Whereas, above 400 µW both types of communication achieve the maximum PDR (cf. Figure 4.3a) and WuR-based communication starts to outperform direct Poll-based data communication in terms of DL latency. At the higher harvesting power values, when the PDR of around 80% is achieved (300 µW and above in Figure 4.3a), it is observed that with the increase in the harvesting power, the DL latency decreases and stabilizes to around 0.78 and 1.10 s for the WuR- and Poll-based approaches, respectively. With the increased value of the harvesting power, the LPN takes less time to achieve the threshold voltage. Thus, the DL latency is reduced with an increase in the harvesting power



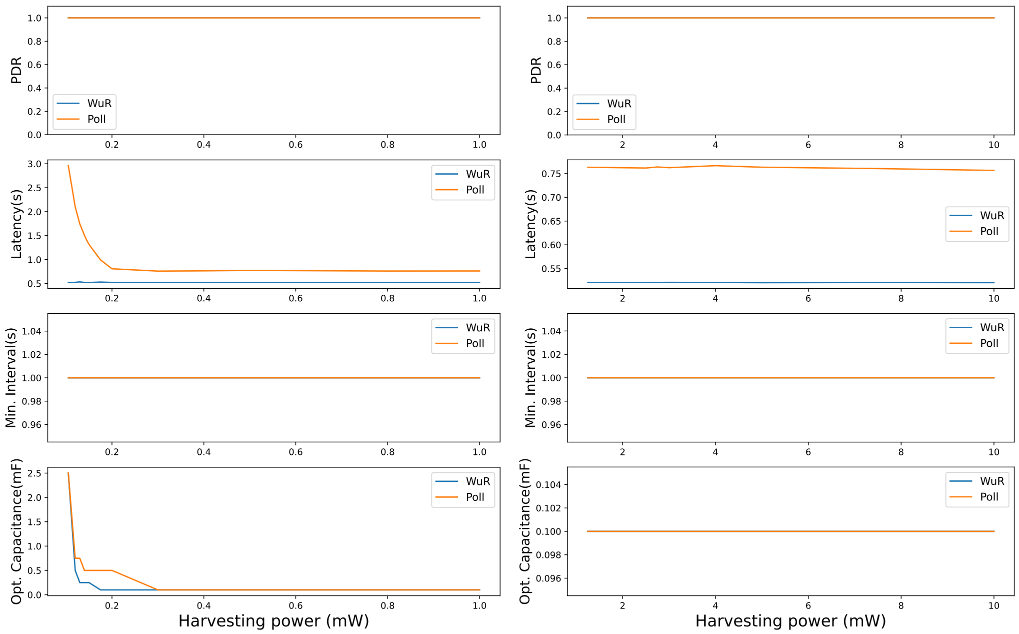
(a) Medium harvesting power

(b) Large harvesting power

Figure 4.4: Comparing WuR-based and direct poll-based communication schemes for a Poisson packet arrival rate = 10 s.

that it decreases drastically from 16.1 to 0.78 s for the WuR-based communication by increasing the harvesting power from 0.2 to 0.4 mW. The optimal capacitance is larger for the harvesting power at which the high PDR is achieved (such as at 300 μW in Figure 4.4a or 130 μW in Figure 4.5a), and the LPN can support frequent SIs. It means the LPN can harvest enough energy to successfully receive the WuS whenever they are sent and can frequently poll without letting the FN drop packets from the FQ.

At the Poisson packet arrival rate of 10 s for low harvesting powers, the PDR improves as compared to that of 1 s packet arrival rate, but the DL latency deteriorates (up to 135.8 s) as shown in Figure 4.4a. As for the low Poisson packet arrival rate (1 s) and low harvesting power, a greater number of packets enter the FQ without being polled by the LPN, and thus a greater number of packets are dropped due to the queue being full. However, as the Poisson packet arrival rate increases, a smaller number of packets are dropped, improving the PDR but also increasing the DL latency. With the higher Poisson packet arrival rate, the LPN receives the older packets that have waited longer, whereas the low Poisson packet arrival rate drops the older packets and the LPN receives the recently added FQ packets obtaining lower DL latency. Moreover, at the Poisson packet arrival rate of 10 s, when the PDR of around 80% is achieved (105 μW and above for WuR-based in Figure 4.4a), with the increase in the harvesting power, the DL latency starts decreasing drastically. For poll-based communication, it decreases from 6.1 to 2.5 s and for WuR-based communication, it decreases from 135.8 to 0.54 s by increasing the harvesting power from 105 to 130 μW. For higher harvesting power, as shown in Figure 4.4a the latency of the



(a) Medium harvesting power

(b) Large harvesting power

Figure 4.5: Comparing WuR-based and direct poll-based communication schemes for a Poisson packet arrival rate = 60 s.

WuR-based communication decreased lower than the poll-based communication.

Similar to the Poisson packet arrival rate of 1 and 10 s, at 60 s for WuR-based communication (cf. Figure 4.5a) shows a decrease in the DL latency from 2.95 to 0.80 s with the increase in harvesting power from 105 to 200 µW. WuR-based communication performs better in terms of DL latency for the harvesting power above 105 µW, where it achieves the DL latency of 0.52 s. For poll-based communication, the latency decreases to as low as 0.76 s at harvesting power of 300 µW. Moreover, for Poisson packet arrival rates higher than 60 s, the conclusions are similar to that of 60 s, showing improvements in PDR at lower harvesting power values for WuR-based communication.

Analysing the results for a specific harvesting power higher than 1 mW (graphs omitted), it is observed that the optimal capacitance for both types of communication becomes the same at a high harvesting power. The value of harvesting power at which the optimal capacitance becomes the same decreases with an increase in the Poisson packet arrival rate. Moreover, with high harvesting power values, the DL latency values for both scenarios remain almost constant, where the WuR-based communication continues to perform better than direct Polling. WuR-based communication obtains 78.08, 53.26 and 52.08 ms of DL latency for the Poisson packet arrival rate of 1, 10, and 60 s, respectively; whereas, poll-based communication obtains 107.44, 77.58, and 76.66 ms.

It can be concluded that the poll-based data communication performs better for low power

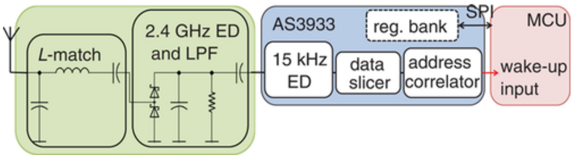


Figure 4.6: 2.4 GHz WuRx design

harvesting techniques such as for low irradiance indoor light (producing up to 400 µW power). In contrast, WuR-based data communication outperforms for large harvesting power techniques. For high harvesting powers (e.g., using thermal energy), a small capacitor of only 100 µF can support all data rates (1 s and above) at maximum PDR. For medium harvesting power, a capacitance of 50 mF is enough to support a packet arrival rate of 1 s or more.

4.3 Discussion

The considered WuR integrated circuit AS3933 works at frequencies between 15 and 150 kHz. However, it is practically easy if this WuR receiver can work directly with existing BLE communication to provide a very low-power channel monitoring. To support that, an extra 2.45 GHz antenna can be added to the batteryless receiving node to decode messages at the same frequency as BLE as shown in Figure 4.6. This is similar to the solution proposed by [70]. After impedance matching, the RF received signal is down-converted from the GHz to the KHz level by means of an envelope detector and low pass filter. With these extra components, an additional power consumption to WuR is also required which would be around 10 μW for sleep and Rx mode [70].

4.4 Conclusions

The optimal parameters to perform the communication between a friend node and a batteryless low power node in Bluetooth mesh networks are studied. The achievable PDR and latency of DL packets, considering different parameters (i.e., capacitance, energy harvesting power and Poisson packet arrival rate) are explored. The results have proven that a batteryless Bluetooth Mesh device can easily support DL communications by using the Bluetooth Mesh friendship feature, both using the traditional polling-based technique or by employing a WuR. Even with harvesting power ranges in the order of tens of micro-watts, a packet arrival rate of 10 s can be supported without any packet loss. The WuR-based approach is mainly beneficial in terms of DL latency when the packet arrival rate is very low (i.e., 1 s) or high (i.e., 60 s). In these scenarios, it provides a DL latency reduction of more than 25% compared to the polling-based technique, which is from 107.44 to 76.66 ms at 1 s packet arrival rate. In summary, this work can be used to know the minimum harvesting power and the optimal capacitor size, which can provide the desired PDR and DL latency for different configurations of the batteryless LPN and FN.

Chapter 5

Bluetooth Mesh Batteryless Prototype

The content of this chapter is based on:

- *A. K. Sultania, and J. Famaey, "Batteryless Bluetooth Low Energy Prototype with Energy-Aware Bidirectional Communication Powered by Ambient Light," in IEEE Sensors Journal, 2022. [Impact Factor: 3.301]*

Indoor light can be used as a new energy source to power micro- or milli-watts low consumption wireless sensor networks (WSNs). The challenge comes from the low irradiance level, and therefore, both artificial and natural light to design a batteryless prototype of a Bluetooth Mesh LPN are studied. This chapter shows a prototype using a mini photovoltaic solar panel for indoor light harvesting using sunlight or a light bulb to power the Bluetooth Mesh LPN. Due to the unpredictability of energy harvesting and the use of a small capacitor instead of a battery, the Bluetooth Mesh LPN's power can become intermittent. This causes the device to frequently switch between the ON and OFF states as it is unaware of its available energy while trying to perform scheduled tasks. In contrast, an energy-aware LPN can try to avoid the OFF state. With the knowledge of the capacitor voltage, it can proactively delay the execution of upcoming sensing or communication tasks and provide some time to recharge the capacitor while consuming a minimum amount of energy by switching itself to the SLEEP state. This chapter evaluates an energy-unaware and -aware batteryless Bluetooth Mesh LPN communicating unidirectionally (downlink only) or bidirectionally for different capacitor sizes at different light-harvesting powers.

5.1 Energy Harvesting Prototype Setup

This section describes the prototype setup, including the selection criteria of hardware. The sustainability of the solution is evaluated by firstly measuring the amount of energy that can be harvested by a solar panel and later the power consumption of the Bluetooth Mesh device executing an optimized LPN application. The power consumption of the application running on a Bluetooth Mesh device is measured using the off-the-shelf Nordic Power Profiler Kit-II (PPK-II), which has a resolution of 0.1 μA [71]. Moreover, the harvesting power at different indoor conditions is calculated by measuring the maximum power point (MPP) voltage V_{mpp}

of the panel (maximum voltage that the panel produces with a load) and the MPP current I_{mpp} of the panel (maximum current that the solar panel produces). This is done by using the digital multimeter (Keysight-U1281A). Now, the chosen commercial components and the reasons for each of them will be discussed.

5.1.1 Hardware Trade-off Analysis

5.1.1.1 Solar Panel:

The solar cells produce voltage and current when they are exposed to light. There are different types of solar cells available in the market, as classified by their materials, that can be used for IoT indoor applications. These materials are mono-crystalline, poly-crystalline, amorphous silicon, and concentrated photovoltaics (PV). A common way to compare solar cells is by their efficiency, which measures how much of the power of the incident light can be extracted while working at the MPP. The concentrated PV cells have the highest efficiency, but they need a cooling system to achieve that. Therefore, mono-crystalline solar panels can be chosen for indoor IoT solutions, considering their high efficiency and life. ANYSOLAR-SM141K06L 6-cell mono-crystalline panels are chosen which have a maximum efficiency of 25% at a reasonable cost. These solar panels are small and can supply up to 184 mW of power which is adequate to support Bluetooth Mesh activities.

5.1.1.2 Light source:

Two different rooms whose corresponding windows are faced towards east and west are chosen. The sunlight in the room can be blocked by walls or furniture and thus does not distribute equally in the room. Therefore, the windowsill is chosen to place the solar panel to receive maximum light from the outside sun. As the sunlight intensity is unpredictable, the artificial lights placed at some distance from the solar panel are chosen to evaluate the performance in a controllable setup. There are various light sources available in the market, such as LED, Fluorescent, and halogen. LED has better luminous efficacy than Fluorescent [72]. Therefore, a 6 W Warm White LED lamp (400 lm) from the Ikea-LED1521R6 series is selected. The experiment with an artificial light bulb is conducted in a dark room where only the bulb produces the light.

5.1.1.3 Energy Storage:

The capacitance of supercapacitors, also known as electrochemical double-layer capacitors (EDLCs), ranges from a few μF to over 1000 F to store energy. Although their energy density is lower than that of batteries, they support many charge and discharge cycles, which in turn provides long life. Yet, they experience ageing effects, as with time, the Equivalent Series Resistance (ESR) increases, and its capacitance decreases. Furthermore, a supercapacitor does not work ideally, as a high ESR limits the amount of useful energy that can be extracted once stored. Therefore, the capacitors with the lowest ESR available in the market at an inexpensive price are chosen. The ESR of the supercapacitor is usually

Table 5.1: Epeas-AEM10941 board configurations

Config. no.	**V**$_{\text{max}}$	**V**$_{\text{turnon}}$	**V**$_{\text{turnoff}}$	**V**$_{\text{HOUT}}$	**V**$_{\text{LOUT}}$
1	4.50 V	3.67 V	2.80 V	2.5 V	1.8 V
2	4.50 V	3.92 V	3.60 V	3.3 V	1.8 V
3	4.12 V	3.67 V	3.01 V	2.5 V	1.8 V
4	4.12 V	4.04 V	3.60 V	3.3 V	1.8 V
5	4.12 V	3.67 V	3.60 V	3.3 V	1.8 V
6	3.63 V	3.10 V	2.80 V	2.5 V	1.8 V
7	2.70 V	2.30 V	2.20 V	1.8 V	1.2 V

decreased with the increase in capacitance. The capacitors chosen have an ESR of 0.47Ω (for 0.47 mF), 0.22 Ω (for 1 mF) and 0.30 Ω (for 10 mF).

5.1.1.4 Power Management Board:

PMBs have multiple roles in the circuit. They are not limited to charging the storage element, regulating the output voltage to the load, and extracting maximum power from the solar cell. Usually, due to the non-continuous and low ambient light source energy harvested by the solar panel, there is a need to use a PMB to manage this incoming power. The PMB can use a Maximum power point tracking (MPPT) circuit to collect as much energy as possible from the energy harvesters. It uses a boost converter to step-up the incoming DC voltage and a buck converter to step-down to a lower voltage. The Epeas-AEM10941 board [63] is capable of both stepping up and stepping down a DC voltage. The stored energy is then converted to a stable voltage to operate a Bluetooth Mesh board. The AEM10941 can be configured to supply different stable voltages (1.2, 1.8, 2.5 or 3.3 V) as V_{HOUT} and V_{LOUT}. The cold-start voltage, i.e., the voltage needed at the start the capacitor charging, is by default at its minimum value of 380 mV with an input power of only 3 μW.

The PMB can charge up the capacitor up to a certain voltage level (V_{max}) and can release the stored energy whenever required by the end node. The PMB also manages the lower threshold of the capacitor by cutting off the output to the end-node once its voltage drops to $V_{turnoff}$ and can release output when the voltage of the capacitor increases to V_{turnon}. The Epeas AEM10941 board already supports seven different configurations as defined in Table 5.1. The user can also define its configuration but needs to add extra resistors to the board.

5.1.1.5 Bluetooth Mesh board:

There are many boards available commercially to build Bluetooth Mesh applications, and their energy consumption comparison can be found in [73]. With the lower energy consumption, and the support of Software Development Kits (SDKs) (containing all the Bluetooth software, structure and Mesh protocol), example applications (to test the device) and online community, Nordic boards are chosen to develop batteryless applications. The Nordic nRF52840 DK boards [74] are chosen. The boards can be programmed to behave like a node in a Bluetooth Mesh network and act as a relay node, FN or LPN. It can be powered by the Epeas-AEM10941 V_{HOUT} or V_{LOUT} providing a voltage of at least 1.8 V.

5.1.2 LPN Application Implementation

The application development is based on the Nordic SDK for Mesh v.4.2.0 and correspondingSoftDevice v7.0.1 with the SDK version 16.0.0. Firstly, to develop the application for a batteryless LPN node, a vendor-specific mesh model to send string messages is added. This helps to communicate any message, such as temperature, event detection, humidity, etc., on the same destination group address using a single model. The receiver end can decode the messages based on the first character in the string message. Secondly, being a batteryless node, it can shut down unpredictably due to the unavailability of energy. Thus, the SDK's Application Programming Interfaces (APIs) are modified so that the LPN saves the context of the associated FN during the friend establishment phase in the free pages of the flash memory. On the restart, the LPN reads the flash data, and if the FN context is already saved in a flash, the LPN restores the local friendship context without reassociating. Furthermore, the usage of the PT timer is divided into polling interval timer (T_{PI}) and poll expiration timer (T_{PE}). According to the specification, the PT timer is the time between two consecutive requests sent by the LPN, and if the FN receives no FPs before the expiration of the PT timer, then it assumes that the LPN is disconnected from the network. Thus, the FN can consider the termination of the friendship. The batteryless LPN can stay in a turnoff state for a long time and should stay connected to the FN without such restriction. Accordingly, the LPN should try to poll on every expiration of T_{PI} and the FN should consider the termination only on the expiration of T_{PE}, which is set to a much higher value.

Moreover, the *lpn* [75] and *light_switch* [76] examples, provided in the Nordic SDK are extended. The *lpn* example is optimized such that it consumes as low energy as possible. The DC/DC regulators are enabled to maximize the system's power efficiency. The *light_switch* example is modified to respond to the incoming messages in its vendor model. A USB powered Nordic device runs the *light_switch* example to act as a friend node, whereas the PMB supplies power to the battery-less node running the modified *lpn* example.

5.1.3 Computing Bluetooth Mesh LPN Power Consumption

To be able to calculate the voltage threshold of each application task X using Equation 3.3, it is thus needed to know the load current $I_L(X)$. The Nordic PPK-II is used by connecting it to the Nordic board in source meter mode as shown in Figure 5.1 to measure the current consumption of each application task. The Nordic board executes in `nRF Only` mode, which reduces the current consumption by shutting the power supply to several external components such as external memories, LEDs and buttons. The Nordic PPK-II interface provides the graphs in real-time, and the data can be exported to a CSV file. It is observed that the lower the supply voltage, the lower the power consumption. Moreover, as Epeas V_{LOUT} pin can be configured to 1.8 V, and that is in the range of the operating voltage of the Nordic

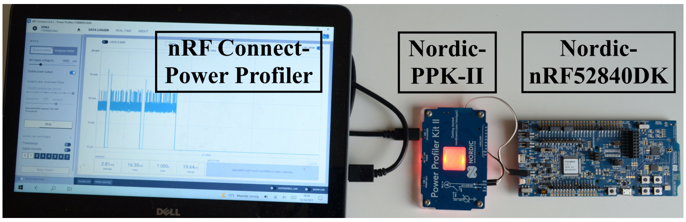


Figure 5.1: Nordic PPK-II connection

Table 5.2: Power consumption at $V_{op} = 1.8$ V for Energy-unaware LPN

State	**Average Current Con. (mA)**	**Time (ms)**	**Average Energy Con. (μJ)**
Initialize	2.25	5.431	21.99
Flash Read (Check FN Context)	5.01	25.61	230.94
Flash Write	4.54	7.155	58.46
Rx Data	4.15	4.207	31.42
Rx FU	4.02	3.669	26.53
Rx FO	4.80	3.495	30.186
Tx Data (0dB)	3.18	4.01	22.95
Tx FP	3.5	3.841	24.19
Tx FR	3.61	4.098	26.62
Tx FP + Rx Data	0.315	104.7	59.36
Tx FP + Rx FU	0.297	104.5	55.85
Tx FR + Rx FO + Flash Write	0.577	110.2	114.44
Sleep State	0.00627	N/A	N/A

board, 1.8 V is used to measure the current consumption. Based on the measurements, the power consumption for different states of the LPN is presented in Tables 5.2 and 5.3. The voltage divider circuit with MOSFET and a large resistor are always connected to the Nordic board, which incurs extra current consumption in the energy-aware LPN. Moreover, the reported numbers are averaged over ten different executions. These numbers vary, with around 8% of deviation on different executions due to the erratic behaviour of hardware. It can be noted from Tables 5.2 and 5.3 that flash activities are power-consuming tasks and therefore should be avoided as much as possible. In our application, flash write is performed only when the friendship is established and flash read when the LPN is rebooted to restore the FN context. To summarize, the unaware LPN performs more flash reads, as it often reboots, while the energy-aware LPN incurs more power consumption due to maintaining the sleep state and using the voltage divider and MOSFETs.

5.1.4 Computing Harvesting Power

In order to evaluate if indoor light harvesting can sufficiently power the LPN activities such as friendship establishment, message exchange, sensing, and reading/writing into flash memory, preliminary measurements to calculate the harvested power are performed. The Epeas board

Table 5.3: Power consumption at $V_{op} = 1.8$ V for Energy-Aware LPN

State	**Average Current Con. (mA)**	**Time (ms)**	**Average Energy Con. (μJ)**
Initialize	2.25	5.431	21.99
Flash Read	5.41	24.66	240.14
Flash Write	4.53	3.314	27.01
Energy Check (EC)	4.86	2.583	22.59
EC + Tx Data (0dB)	0.601	35.18	38.07
EC + Tx FP	0.695	29.33	36.68
EC + Tx FP + Rx Data	0.412	107.6	79.79
EC + Tx FP + Rx FU	0.379	107	73
EC + Tx FR + Rx FO + Flash Write	0.596	113.6	121.87
Sleep State	0.0062	N/A	N/A

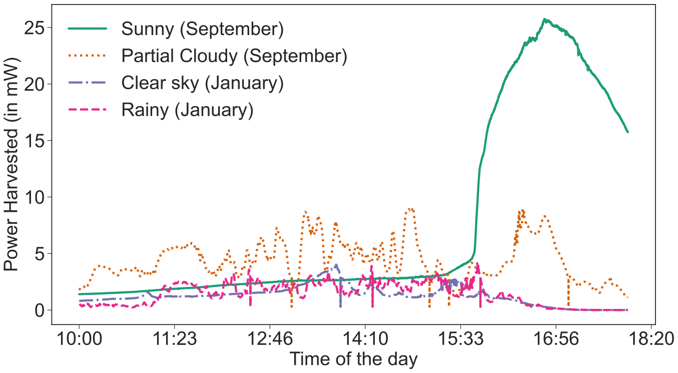


Figure 5.2: Harvesting power using sunlight at west-side windowsill

harvests power at MPP (P_{mpp}), which can be calculated by measuring MPP voltage (V_{mpp}) and MPP current (I_{mpp}). P_{mpp} is computed for different times of the day with sunlight and differently distanced light bulbs (in a room without sunlight). The calculated P_{mpp} for an entire day using the solar panel placed at a west-side windowsill is shown in Figure 5.2. It can be observed that during most of the time, the panel can easily harvest around 2 mW in January and around 3 mW in September. On a sunny day, when sunlight falls on the solar panel, the harvested power can be increased eightfold. Whereas, with artificial light (Figure 5.3), the harvested energy is more predictable and varies from 50 mW (10 cm) to 60 μW (210 cm).

It is also observed that the harvesting power is lowest during the winter months (October to February), and the average is usually less than 6 mW. This is equivalent to harvesting the energy from a LED bulb placed 20 to 30 cm away from the panel. It should be noted that the source light spectrum was not analyzed. When sunlight shines through the windows of a building, it also loses some of its energy because glass windows typically block UV radiation [77]. Therefore, based on the type of window glass used, the results can deviate.

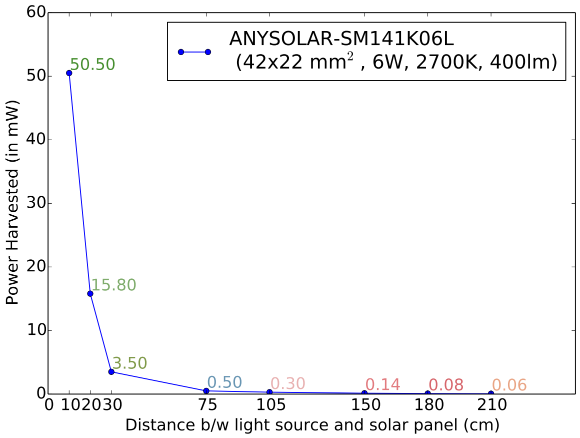


Figure 5.3: Harvesting power using light bulb

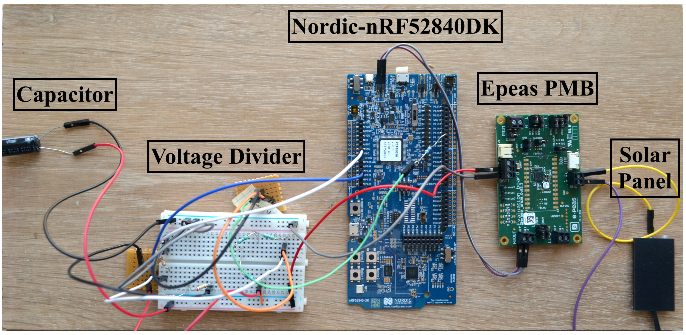


Figure 5.4: Energy-aware prototype setup

5.1.5 Final Prototype Design

The setup of the batteryless prototype for an energy-aware solution is shown in Figure 5.4. Whereas, for an energy-unaware solution, the voltage divider is removed. The Epeas-AEM10941 evaluation board is attached to the solar panel and a supercapacitor. One of the output pins of the board provides power to the Nordic nrf52840 DK. An energy-unaware solution works intermittently due to unpredictable energy sources and storage for the scheduled tasks. Thus, upon power failure, the volatile memory is lost. Therefore, some data (e.g., the context of the associated FN) needs to be retrieved from non-volatile memory after regaining power. This retrieval for the Nordic board is done by reading the FN context from the flash memory, which itself is a power-hungry task. This lowers the overall usable energy to perform Bluetooth Mesh computations. Thus, the energy-aware solution can benefit as it avoids restarting by making intelligent decisions to delay the tasks' execution. Also, large size capacitors can store more energy, but their charging time is higher. An LPN with a large capacitor harvesting at a low HP can take hours to charge the capacitor from $V_{turnoff}$ to V_{turnon} (for unaware LPN) or from V_{cutoff} to $V_{threshold}^{Task_i}$ (for aware LPN). During this time, the LPN is unavailable and therefore, there is a need to choose an optimal minimum

Table 5.4: Threshold voltage for configuration 1 (V_{max} = 4.5 V, V_{turnon} = 3.67 V, $V_{turnoff}$ = 2.8 V), HP = 140 μW, V_{cutoff} = 2.83 V

State	**C= 0.22mF**	**C= 0.47mF**	**C= 1mF**	**C= 10mF**
Initialize	2.87	2.85	2.84	2.84
Flash Read	3.23	3.01	2.92	2.84
Flash Write	2.88	2.85	2.84	2.84
EC	2.87	2.85	2.84	2.84
EC + Tx Data (0dB)	2.88	2.86	2.84	2.84
EC + Tx FP	2.88	2.86	2.85	2.84
EC + Tx FP + Rx Data	2.92	2.87	2.85	2.84
EC + Tx FP + Rx FU	2.83	2.83	2.83	2.84
EC + Tx FR + Rx FO + Flash Write	2.98	2.9	2.87	2.84

size of the capacitor based on the required data interval (DI) and minimum available HR.

As an unaware batteryless LPN can only operate during the time the capacitor discharges from V_{turnon} to $V_{turnoff}$; the configuration number 1 of the Epeas board from Table 5.1 is chosen, which provides the maximum difference in V_{turnon} and $V_{turnoff}$. In comparison, the aware LPN executes tasks once it attains $V_{threshold}^{Task_i}$which needs to be calculated for each task. Based on the measured current consumption of each task (Table 5.3) and the configured voltages, the corresponding threshold voltages are calculated using Equation 3.3 as presented in Table 5.4 for the selected Epeas configuration. The HP is considered to be low between 140 and 280 μW. The V_{cutoff} (below which the energy-aware LPN would stop its tasks' execution) as $V_{turnoff}$ plus 0.03 V. By considering the response time of the Epeas PMB board to decide to shut down the voltage supply, it is experimentally found that selecting V_{cutoff} of 2.83 V can avoid restarting the energy-aware LPN with the smallest capacitor size. These threshold values are in the range between V_{max} and $V_{turnoff}$; therefore, the aware LPN would be able to execute the tasks. The aware LPN can be configured to execute each task when the corresponding capacitor threshold voltage is reached based on the size of the attached capacitor. The threshold voltage of the small capacitor is higher than that of the large capacitor. Thus, for simplicity, the threshold of the lowest capacitor (0.22mF) is used to execute the aware LPN application attached to any capacitor size of 0.22mF or higher. While this is suboptimal for larger capacitor sizes, it allows us to use a single task scheduling configuration, independent of the used capacitor size and the Epeas configuration.

5.2 Comparison of Energy-Unaware and -Aware LPN

Two experimental use-cases are set up to compare the energy-unaware and -aware LPN as follows:

- *Uni-directional DL only*: The LPN is expected only to receive data from other nodes in the mesh network that is being transmitted at a fixed interval.

- *Bi-directional communication*: The LPN is expected to receive data from other nodes in the mesh network, and upon receiving a data packet, it sends back a UL packet.

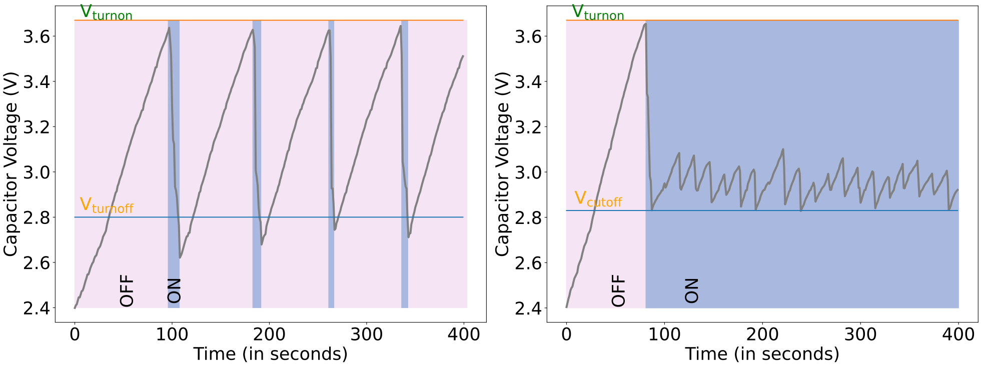
Each data packet is considered to be unsegmented with a fixed payload length of 8 bytes. Also, after the data communication, the LPN returns to sleep mode, consuming a minimum amount of energy (6.2 μA). Our work is focused on evaluating the friendship feature specifically, the uni-directional UL only use case is omitted.

The DI of 5s is considered. It is observed that when the data is transferred at a fixed DI of 5s, the energy harvested with the light bulb at a distance of 105 cm (i.e., 300 μW) is sufficient for the operation of the LPN. Therefore, the sunlight in the daytime is also more than sufficient to support all the use-cases (Uni- and Bi-directional communications). At this HR, the voltage of the small capacitor (0.47 mF) never drops below, $V_{turnoff}$ and so the results in unaware and aware solutions are similar. As such, the batteryless LPN harvesting at rates lower than 300 μW are analysed in the remainder of this section.

5.2.1 Capacitor Voltage Behaviour

The charging and discharging behaviour of a batteryless LPN are computed for the energy-unaware and aware application. Figures 5.5-5.6 show, for a 0.47 and 1 mF capacitor, respectively, the variation in the capacitor voltage running the LPN application with bidirectional communication on a batteryless node harvesting at 80 μW for the Epeas configuration 1. It can be observed that the unaware LPN's capacitor voltage continuously goes below 2.8 V ($V_{turnoff}$) while doing the LPN's activities (flash read/write, polling, data communication), leading the PMB to turn off the LPN and later turning it on again at 3.67 V. The energy-aware LPN never goes below 2.8 V and recharges the capacitor up to the required threshold voltage to perform the LPN's tasks. The drop of the capacitor voltage does not stop strictly at 2.8 V because it depends on the action sensitivity of the Epeas PMB to stop supplying energy from the capacitor.

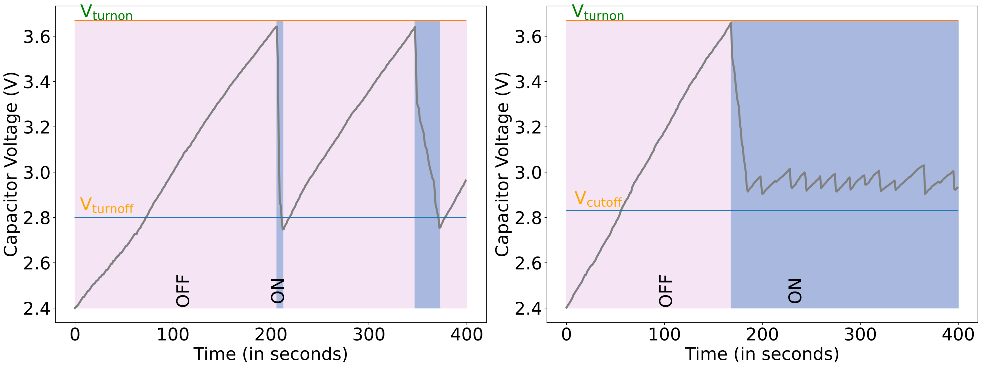
The LPN's outage time depends on the capacitor size and the active-time voltage gap. The larger the capacitor, the longer it takes to charge and discharge. By comparing Figures 5.5 and 5.6, it can be noted that the smaller capacitor causes more frequent charge-discharge cycles of the capacitor. The charging time from $V_{turnoff}$ to V_{turnon} for an unaware LPN that is supported by a 0.47mF capacitor takes around 64s, and stays on 10s while performing communication. With a 1mF capacitor, the active communication time increases to 25s, but now the charging time also increases to 135s. In contrast, once the aware LPN becomes active, it does not execute any tasks for the time to recharge the capacitor from V_{cutoff} to $V_{threshold}^{Task}$. When first turning on at a voltage of 3.67 V, the LPN spends energy in reading the flash and establishing a friendship with an FN. Later, there are mostly two activities scheduled, which are receiving DL by polling and optionally sending UL data. The average waiting time for an aware LPN with a 0.47 mF capacitor is 15.7s which then receives and sends one data packet. However, with 1mF, it only takes 12.7 s to recharge from its corresponding dropped voltage value (near V_{cutoff}) to $V_{threshold}^{Task}$. This is because of two



(a) Unaware

(b) Aware

Figure 5.5: Variation in 0.47 mF capacitor voltage supporting bi-directional communication at 80 μW for Epeas Configuration 1, ECI = 15s



(a) Unaware

(b) Aware

Figure 5.6: Variation in 1 mF capacitor voltage supporting bi-directional communication at 80 μW for Epeas Configuration 1, ECI=15s

reasons. First, voltage drop after task execution for a 0.47 mF capacitor is higher than that of a 1 mF capacitor. Secondly, for the first reason, the number of the executed energy check (EC) tasks of 0.47 mF capacitor is more than that of a 1 mF.

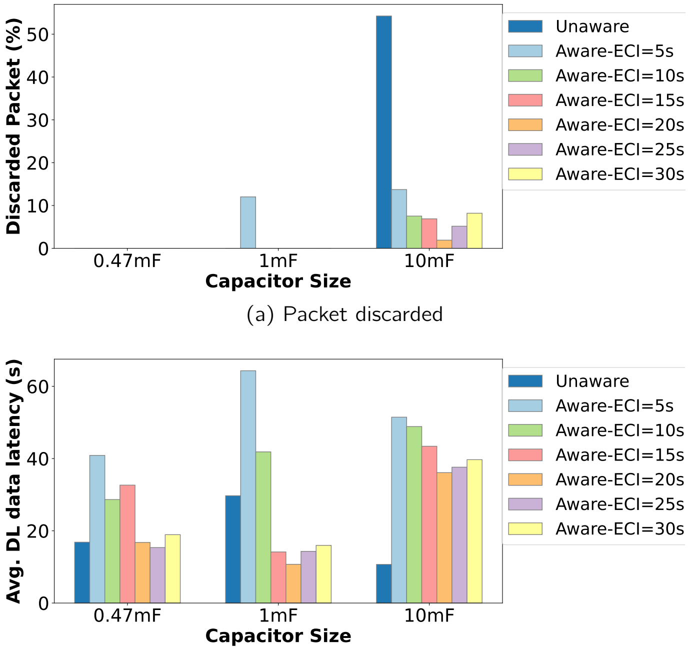
5.2.2 Result Analysis

The Epeas PMB is configured with configuration 1 to analyze the energy-unaware and aware LPN. The following performance metrics are considered in the comparison between the friendship communication mechanisms:

- *Average DL Data Latency*: The average time gap between receiving a packet by the LPN from the time it arrives at the FQ.
- *Average DL IAT*: The average time difference between the arrival of two consecutive DL packets received by the LPN.
- *Average UL Data IAT*: The average time difference between the arrival of two consecutive UL packets received by the FN.
- *Discarded Packet Percentage*: The percentage of packets that are discarded from the FQ due to queue overflow.
- *Restart Count*: The number of restarts by the LPN.
- *Outage Time*: The time during which the LPN is in the OFF state.

5.2.2.1 Uni-directional DL Only

Figure 5.7 shows the results for different ECIs for the LPN receiving data at 5s data interval and harvesting at 140 μW. It can be observed that as the capacitor size increases, more packet drops can be observed. This is because the larger the capacitor, the more time it takes to charge. At an ECI of 20s with a 1mF capacitor, the energy-aware LPN shows the best DL data latency. With the same capacitor for the lower ECIs (less than 15s), the LPN wastes energy checking the capacitor voltage frequently rather than initiating an FP. As a result, it delays the capacitor voltage in reaching $V_{threshold}^{DL}$. Therefore, the latency is higher than when using an ECI equal to 15s. Whereas, increasing the ECI beyond a certain value (20s with 1mF or 10 mF and 25s with 0.47 mF) increases the delay as the device does not immediately detect the voltage threshold ($V_{threshold}^{DL}$) being reached. Therefore, the DL data latency and packet discard percentage increases for those large ECIs (can be observed with 10 mF). This can be confirmed by checking the capacitor charge time, which in these cases is equal to or close to the corresponding ECI. The capacitor charge time of the unaware LPN is high, but it can send more than 1 packet during its ON state. The best result of the unaware LPN is at 0.47 mF capacitor with no packet loss. As the developer can choose both the ECI and capacitor size, the optimal ECI/capacitor combination of the energy-aware solution with the optimal capacitor size of the energy-unaware solution will be compared in the remainder of this section. Table 5.5 shows all the optimal configurations for different traffic types, FQ sizes, and harvesting powers.



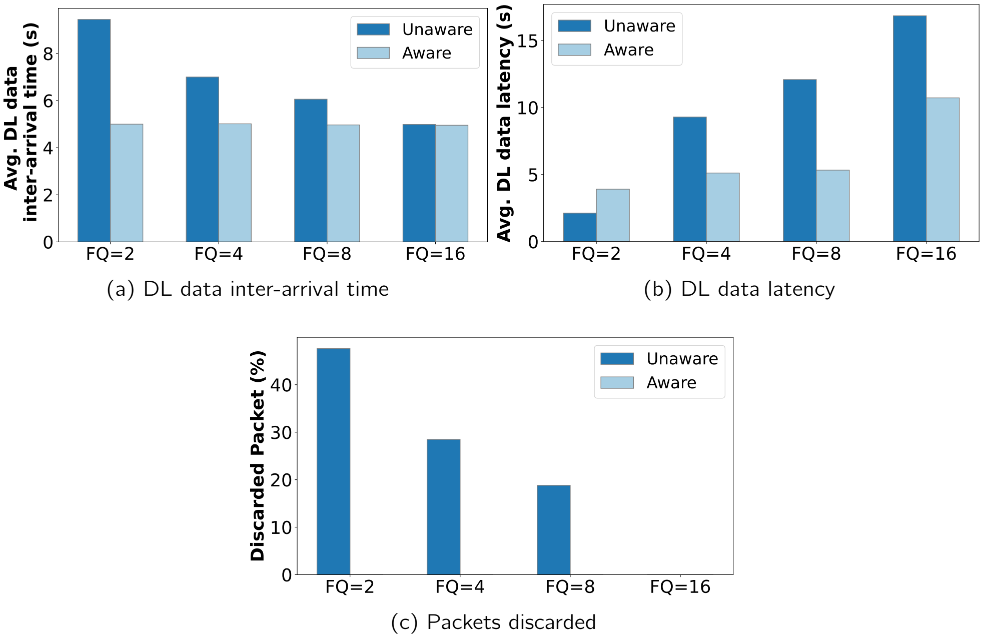
(a) Packet discarded

(b) DL data latency

Figure 5.7: Uni-directional communicating LPN's performance for different ECI at HP = 140 μW and FQ = 16

Table 5.5: Optimal configuration for Energy-aware LPN for DR=5s

Traffic type	**HP**	**FQ**	**Best Configuration(s) (CS,ECI)**
Uni-directional	140 μW	16	(1mF, 20s)
	210 μW	16	(1mF, 5s), (0.47mF, 3s)
	280 μW	16	(1mF, 3s), (0.47mF, 3s)
	140 μW	8	(1mF, 5s), (0.47mF, 5s)
	140 μW	4	(1mF, 4s), (0.47mF, 5s)
	140 μW	2	(1mF, 4s), (0.47mF, 5s)
Bi-directional	140 μW	16	(1mF, 20s)
	210 μW	16	(0.47mF, 3s)
	280 μW	16	(1mF, 3s), (0.47mF, 3s)
	140 μW	8	(1mF, 30s),(0.47mF, 30s)
	210 μW	8	(1mF, 10s)
	280 μW	8	(0.47mF, 3s)



(a) DL data inter-arrival time

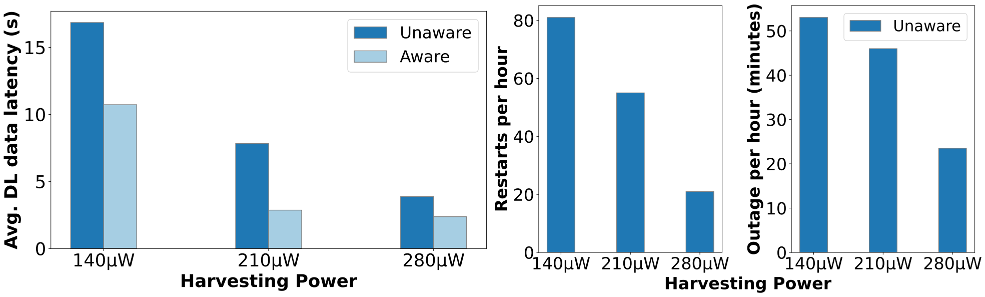
(b) DL data latency

(c) Packets discarded

Figure 5.8: Uni-directional communicating LPN's performance for different FQ at HP = 140 μW, Optimal ECI

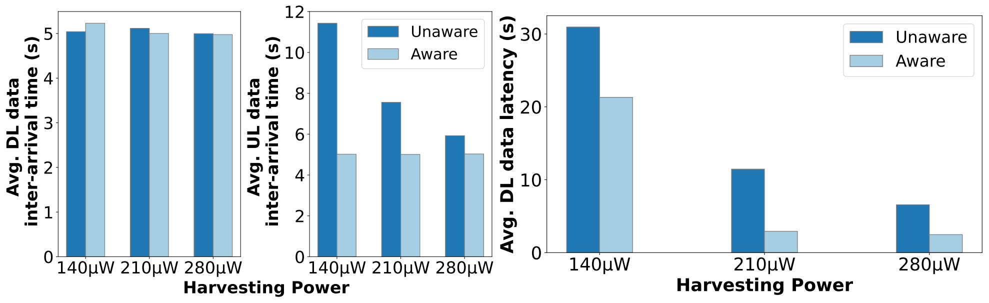
It can be observed from Figure 5.8 that the larger the FQ size, the better the energy-unaware LPN performs in terms of IAT and discarded packets. As with the increase in FQ, more packets start being queued; the unaware LPN receives more packets per discharge cycle. Thus, the packet discard percentage is reduced. Also, several packets need to wait longer to be polled by an LPN, which increases the overall DL data latency. It is the same for the aware LPN, which increases its DL data latency. However, the energy-aware LPN polls whenever it gains the threshold voltage to receive a DL packet ($V_{threshold}^{DL}$), which takes less than 5s, and therefore it never receives an FU. Therefore, the discarded packet percentage and the IAT does not change. As, at FQ equal to 16, the energy-unaware LPN shows zero discarded packets; this FQ size is considered to compare it with aware LPN results.

Further, an optimal ECI configuration is considered to send data at 5s intervals, harvesting at different rates (140 to 280 μW) and having an FQ equal to 16. The results are shown in Figure 5.9. It is observed that the DL data IAT is 5 seconds for both the LPNs (energy-aware and unaware) for all the HPs, while there is no packet loss. However, as the harvested power increases, the capacitor charging time decreases, which decreases the DL data latency and also the number of restarts and outage time per hour. An energy-aware LPN gains the benefit of lower data latency, which is up to 63% at HP of 210 μW. Also, the energy-unaware LPN improves its performance for higher HP but still have DL data latency 38% higher than the energy-aware LPN. It also experiences outages due to restarts, which is 23.5 minutes per hour.

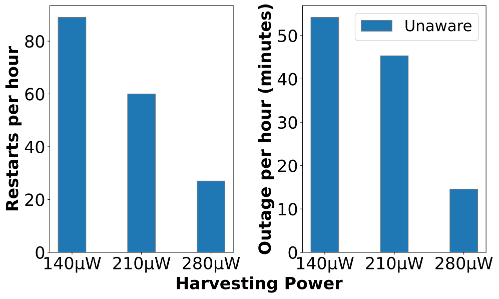


(a) DL data latency (b) Energy-unaware LPN outage

Figure 5.9: Uni-directional communicating LPN's performance at different HP at FQ=16, Optimal ECI

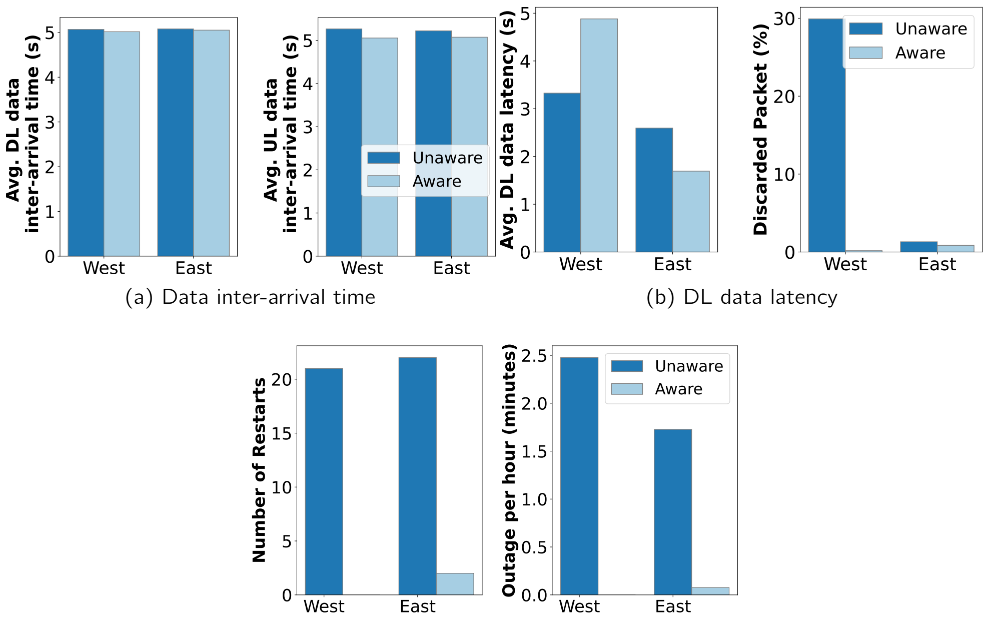


(a) Data inter-arrival time (b) DL data latency time



(c) Energy-unaware LPN Outage

Figure 5.10: Bi-directional communicating LPN's performance at different HP at FQ=16, Optimal ECI



(a) Data inter-arrival time

(b) DL data latency

(c) LPN outage

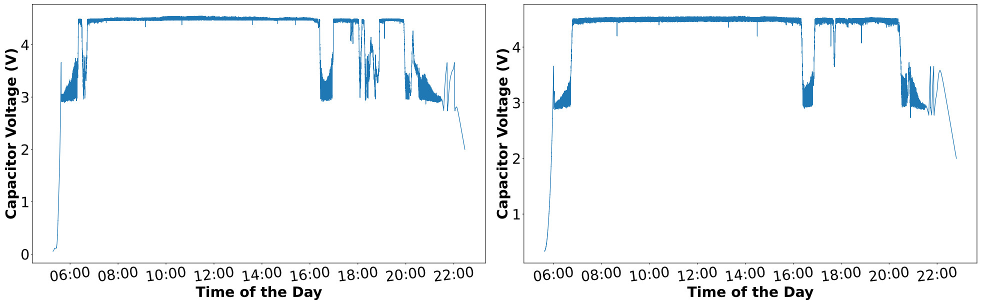
Figure 5.11: Bi-directional communicating LPN's performance by placing solar panel at the east and west side windows

5.2.2.2 Bi-directional communication

The pattern of the bi-directional communication results is the same as uni-directional communication. When both UL and DL data are communicated, the LPN consumes a bit more energy in each cycle than uni-directional communication; therefore, the DL data IAT is increased. It can be observed from Figure 5.10a) that the DL data IAT is increased about 6% for an aware LPN harvesting at 140 μW. However, the UL data IAT for the aware LPN is better than unaware (more than half) because, for UL, there is no buffer to store the packets temporarily. The DL data latency for the aware LPN is always lower than the unaware LPN. At the HP of 210 μW, the energy-aware LPN shows maximum improvement (up to 75%) on DL data latency over its counterpart. Furthermore, compared to uni-directional communication, the DL data latency for bi-directional communication is higher because of the power consumption due to the additional UL tasks. Moreover, for the same reason, the restart count and the outages increase a bit. The DL data discard percentage for the LPNs is zero (graph not shown).

5.2.2.3 Natural Light Experiments

The experiments using natural light are performed to evaluate the performance of the system under dynamic harvesting power that changes over time. The setup is placed at the windowsills of two separate rooms in the east and west direction in Antwerp, Belgium, where



(a) East-Side LPN

(b) West-Side LPN

Figure 5.12: Variation in energy-aware LPN capacitor (1 mF) voltage over time with solar panel at the east and west side windows

the indoor sunlight is coming from the double-layered glass windows. The energy-aware LPN is equipped with a 1mF capacitor, and the ECI is configured to 3s (the optimal configuration at 280 μW). On 21st Sept, the day was sunny, and the sun rose and set at 07:27 and 19:43, respectively [78]. The performance of both the LPNs placed by the east and west windowsills is presented in Figure 5.11. It can be observed that the IAT of the energy-aware LPN is better than the unaware LPN. The DL data latency for the unaware LPN is better on the west side but experiences around 30 % of packet loss. The packets that wait longer in the queue are lost, and so the latency drops.However, when the discard percentage for both the LPNs are nearly equal (East side), it can be observed that the energy-aware LPN performs better in terms of DL latency that is 34% low. The energy-aware LPN experiences two restarts that occur at 07:34 and 19:40 when the light illumination is low. The same setup with a 0.47 mF capacitor is tested, and the results were similar but showed more restarts for the energy-unaware LPN, up to 115 on the east side. Being a sunny day, the capacitor voltage is maintained in the range between V_{max} and 4.45 V for almost the entire day, except during the sunrise and sunset. The variation in the capacitor voltage over the entire day of 10th July 2021 is presented, which was mostly cloudy and rainy. The sun rose and set at 05:39 AM, and 09:56 PM [78]. The change in voltage for the energy-aware LPN's capacitor of 1mF placed on both sides is measured for the whole day as shown in Figure 5.12. It can be observed that the east-side LPN starts communication 22 min earlier than the west side (06:00) because it receives more light beforehand. Whereas during the sunset, the east side LPN turns off at 22:08, which is 21 minutes earlier than the west side, which was getting more light during the sunset. This also provides the west-side LPN with a bit of power to reboot 3 times before turning off. Moreover, a sharp voltage down is observed after 16:20, which is due to heavy rainfall [79]. Apart from that, the capacitors maintain the V_{max} for almost the entire day.

However, when increasing the capacitor size to 220mF in the east side LPN, it can be observed from Figure 5.13 that the capacitor voltage never drops below $V_{turnoff}$. It means the energy stored in a 220mF capacitor during the daytime is sufficient to keep the communication active during the nighttime after a sunny day, maintaining a DL latency of 1.47s.

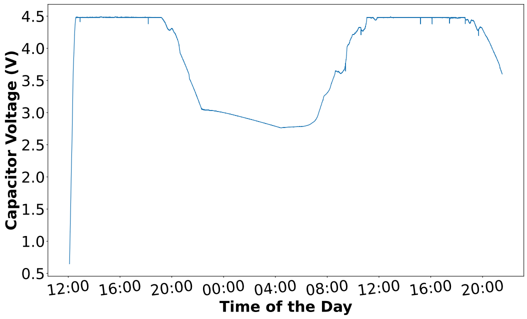


Figure 5.13: Variation in energy-aware LPN capacitor (220 mF) voltage over two days with solar panel at the east side window

5.3 Conclusion

A working prototype of the batteryless Bluetooth Mesh LPN is presented, which operates by harvesting power from indoor light and is sufficient to support many IoT use-cases. The energy-unaware and aware LPN are compared for different capacitor sizes and harvesting powers. It can be concluded that the aware LPN performs better when configured by optimally choosing the energy check interval (ECI) values. This is because the ECI defines the time gap for the energy-aware LPN to recheck the available voltage if its capacitor voltage exceeds the required threshold value to execute a task successfully. The energy-aware LPN provides good performance with different friend queue sizes and different harvesting powers, communicating uni-or bi-directional data. At the harvesting power 140 μW and the friend queue size of 2, the energy-aware LPN avoids packet loss, which was around 47% for the energy-unaware LPN. The energy-aware LPN also shows improvement in DL data latency by 63% when receiving uni-directional DL data, and this improvement is increased to 74% when it performs bi-directional communication. With bi-directional communication, both the LPNs consumes a bit more energy to send the UL data and therefore, the overall DL latency for the corresponding harvesting power increases. The aware LPN also shows better performance results when it harvests from indoor artificial or sunlight (in all weathers).

Part III

Narrowband Internet of Things

Chapter 6

Introduction to NB-IoT Power Saving Features

The content of this chapter is based on:

- *A. K. Sultania, C. Blondia and J. Famaey, "Optimising the Energy-Latency Tradeoff in NB-IoT With PSM and eDRX," in IEEE Internet of Things Journal, vol. 8, no. 15, pp. 12436-12454, 1 Aug.1, 2021. [Impact Factor: 9.471]*
- *A. K. Sultania, and J. Famaey, "Batteryless NB-IoT Prototype for Bidirectional Communication Powered by Ambient Light," in Ad Hoc Networks, 2022. [Impact Factor: 4.111]*

This chapter introduces the NB-IoT device network access procedure and its power-saving schemes. It also presents the literature review of power consumption and latency optimization and batteryless NB-IoT prototype.

6.1 Overview of NB-IoT

NB-IoT is an LPWAN version based on Long-Term Evolution (LTE). It is designed to address the IoT device requirements of low cost, low power, sparse transmission, good coverage, and long-range. It was introduced in the 3GPP cellular standards of Release 13 onwards. It works in a narrow channel bandwidth of just 180 kHz (1 resource block of an LTE channel, with 12 sub-carriers and each sub-carrier being separated by 15 kHz). Functioning in the licensed spectrum, it is less susceptible to channel interference and thus able to deliver high reliability. Working on a single Physical Resource Blocks (PRBs) (180 kHz) with much-reduced complexity and power consumption, many of the legacy LTE control channels have been redefined to fit into a single PRB of NB-IoT. So, essentially when a User Equipment (UE) attaches to a network, it searches for an NB-IoT channel assigned by the network operator. The network also differentiates between the NB-IoT UE and other legacy LTE UEs as they have different architectures for PRB and treat the corresponding signals

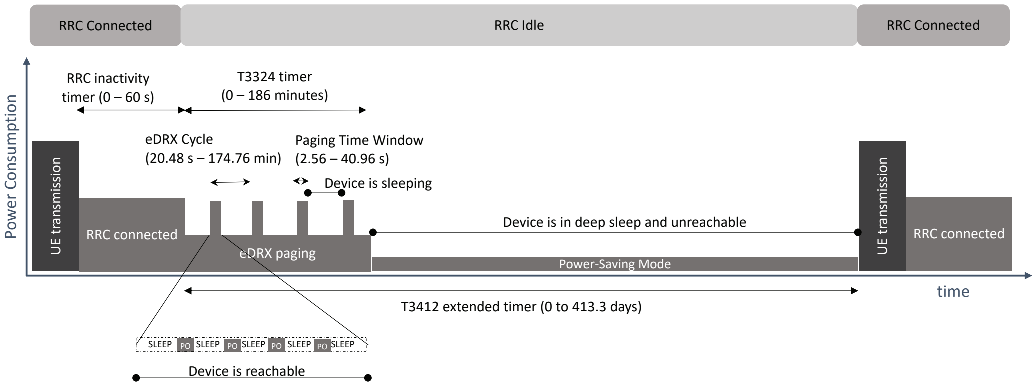

RRC Connected
RRC Idle
RRC Connected
Power Consumption
RRC inactivity timer (0 – 60 s)
T3324 timer (0 – 186 minutes)
eDRX Cycle (20.48 s – 174.76 min)
Paging Time Window (2.56 – 40.96 s)
Device is sleeping
UE transmission
RRC connected
eDRX paging
Device is in deep sleep and unreachable
Power-Saving Mode
UE transmission
RRC connected
time
T3412 extended timer (0 to 413.3 days)
SLEEP PO SLEEP PO SLEEP PO SLEEP PO SLEEP
Device is reachable

Figure 6.1: Overview of the NB-IoT UE duty cycle states

in different ways. But even though NB-IoT has a re-designed architecture, the essential elements which constitute the slot, subframe, radio frame, resource elements (REs), and their timing and scheduling remain the same. Therefore, NB-IoT can reuse the raw essentials of legacy LTE by updating the RAN at the eNodeBs. It uses the same Orthogonal Frequency Division Multiple Access (OFDMA) and Single Carrier Frequency Division Multiple Access (SC-FDMA) principles as LTE for downlink and uplink communication respectively. NB-IoT also reuses the basic features of LTE such as robust authentication, encryption, and other security mechanisms and thus can be targeted towards mission-critical IoT applications.

6.2 Overview of NB-IoT Radio States

When the device is powered on, it fetches the frequency, synchronizing timings, detects signal quality, and some important configurations from the network. It decides to camp on to the best cell and starts the Random Access Channel (RACH) process. The RACH helps the device to obtain the resources for the Radio Resource Control (RRC) connection request. This RRC connection setup is an important step because only after that, the device and the network exchange data. When the RRC connection is established, the device is said to be in RRC Connected state. In NB-IoT, there are two RRC states for devices, namely, *RRC Connected* and *RRC Idle* as shown in Figure 6.1. When the device releases its active RRC connection, it moves to the RRC Idle state. The device in RRC Connected state consumes more energy as it gets dedicated bearers established to begin the data transmission and monitors the DL channel in all the subframes (SFs) except for the SFs for UL transmission. The control channel it monitors is called the Narrowband Physical Downlink Control Channel (NPDCCH), which is required to receive the DL data notification or UL data grant from the eNodeB (eNB). These notifiers are known as paging indicators, and the procedure of indicating to the device that data is available is called paging. The time instance of paging is known as a paging occasion (PO). The device receives the data from the network over the Narrowband Physical Downlink Shared Channel (NPDSCH) and transmits it over the Narrowband Physical Uplink Shared Channel (NPUSCH). There is a network defined timer known as *RRC Inactivity Timer*, whose expiration makes the device transition from the

RRC Connected to RRC Idle state. The eNB initiates the RRC Release message on the expiration of the RRC Inactivity timer. Instead of waiting for the RRC release message from the network, the UE may also use Release Assistance Indication (RAI) to indicate that no further uplink or downlink data transmissions are expected. This can help the network to decide if the connection can be released and thus reduce the period the UE spends in the RRC connected state. Hence, it conserves even more energy. However, the RAI feature has been introduced in Release 14 and is not supported by all the NB-IoT device categories. Therefore, its study is omitted from this work.

In the RRC Idle state, NB-IoT defines two power-saving schemes, i.e., Extended Discontinuous Reception (eDRX) and Power Saving Mode (PSM). In this state, the device saves a lot of energy as it stops continuously monitoring the DL channels. This, however, has a significant impact on the latency of DL transmissions, as POs are either less frequent (in eDRX) or non-existent (in PSM). As such, DL data is buffered by the NB-IoT network until the next PO occurs. These power-saving schemes are described in the remainder of this section.

6.2.1 RRC Idle State

In RRC idle state, the UE does not have an established physical connection to the eNB. However, the network has its identity and location based on the tracking area update (TAU). This section discusses the two power-saving schemes in this state.

6.2.1.1 Extended Discontinuous Reception

It is similar to Discontinuous Reception (DRX) used in LTE systems but with longer timer values to achieve further improvement in energy consumption. The eDRX is designed to allow periodic DL data while minimizing energy consumption. The scheme works in cycles where each cycle consists of a short period "*On Duration*" during which the device monitors the DL control channel and a sleeping period during which the device saves its battery and stops monitoring the control channel. This feature can be used while the device is in either of the RRC states. In RRC connected state, it is named connected-mode DRX (C-DRX), and in the RRC Idle state, it is named idle-mode DRX (I-DRX). In IoT use cases, the RRC Connected period should be short, and therefore eDRX during RRC Idle makes a bigger contribution to the battery-saving compared to eDRX during RRC Connected. As such, in this thesis, only eDRX during RRC Idle state is considered, as shown in Figure 6.1. A timer T_{3324} (also known as Active timer) is defined as the eDRX state time of the device during which a device can still be periodically reachable by the network. For NB-IoT, it varies from 0 to 186 minutes [80]. During this period, the device monitors the channel for paging messages at the interval of the eDRX cycle, which can be configured up to 174.76 minutes [81]. An eDRX cycle consists of a Paging Time Window (PTW) time (between 2.56 and 40.96 seconds) followed by a sleep time [82]. The device monitors the channels during a few SFs, the POs within the PTW. PTW thus involves cycles that alternate between periods of active listening and sleep. During paging, if the device receives a DL data notification from the eNB, it switches to the Connected state, and the eNB resets the RRC inactivity timer. If a DL packet arrives at the eNB in between paging events, the data is temporarily buffered by the network. This periodic DL reception during eDRX incurs additional latency. However, the UL latency is not affected, as the UE can directly move to the RRC Connected state as soon as UL data needs to be sent. As such, the UL latency depends only on synchronization, broadcast information fetching, random access, resource allocation, data transmission, and feedback response delay [7]. Upon expiration of T_{3324} without any activity, the device can switch to PSM or restart the eDRX state if PSM is disabled.

6.2.1.2 Power Saving Mode

The main purpose of PSM is to minimize energy consumption while the device does not transmit or receive anything. The device is in a dormant state during the PSM and cannot receive DL data. It consumes energy similar to the switch-off state while it stays registered to the network. The PSM timer for NB-IoT is up to 413.3 days and is represented as $T_{3412\ extended}$ [80]. Upon expiration of T_{3412}, the UE monitors the channel for paging messages or performs a TAU for synchronization. Therefore, it can receive DL data only when PSM ends, which happens when the PSM timer expires or the UE switches to the Connected state (e.g., when it needs to send UL data). If a device in the PSM state generates a UL packet, it switches to the Paging state to monitor the control channel for the UL grant. The device will switch to the Connected state if it receives the grant, so the UL latency is only slightly affected. If the grant is rejected, the device switches back to the PSM state. All the incoming DL data during the PSM cycle is buffered by the network and sent to the device after PSM ends.

In comparison to PSM, eDRX saves less energy but provides better DL communication latency. Therefore, both the PSM and eDRX mechanisms can be used to adapt to different IoT scenarios. The state change depends on network traffic and device behaviour. Hence, it becomes crucial to select an optimal eDRX configuration and paging period for various types of UL and DL traffic in order to balance the power-saving ratio and DL communication latency.

6.2.2 RRC Connected State

The UE switches to the RRC Connected state to transmit or receive some data. Therefore, this section presents an overview of both the DL and UL data transmission mechanisms.

6.2.2.1 Downlink

The NB-IoT DL frame consists of three channels, namely Narrowband Physical Broadcast Channel Narrowband (NPBCH), NPDCCH, and NPDSCH, as presented in Figure 6.2. The DL frame also consists of three signals that are generated at the physical layer for synchronization and channel estimation functions. These signals are named as Narrowband Reference Signal (NRS), Narrowband Primary Synchronization Signal (NPSS), and Narrowband Secondary Synchronization Signal (NSSS). NPBCH carries the Narrowband Master

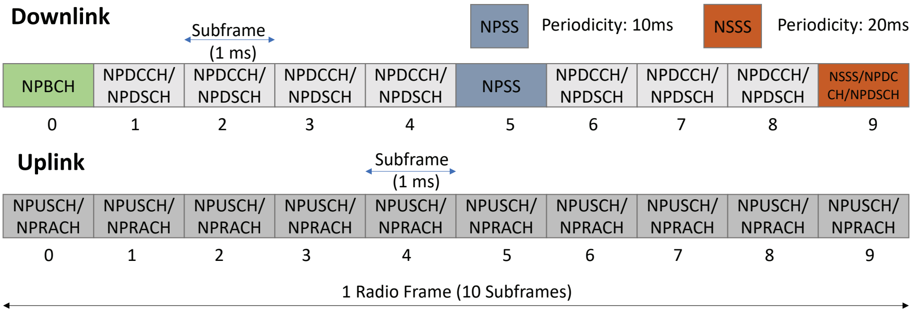


Figure 6.2: NB-IoT uplink and downlink frame structure

Information Block (MIB-NB) at SF-0 and is transmitted over a time period of 640ms. MIB-NB carries some high-level information such as system timing and System Information Block (SIB1-NB) scheduling configurations. The UE uses the NPSS and the NSSS for time and frequency synchronization and cell identity detection. The NPSS is transmitted at SF-5 of every radio frame, and NSSS at SF-9 of every even radio frame. The NRS is used to provide phase reference for the demodulation of the DL channels [7]. The NRS is transmitted in the SFs that carry NPBCH, NPDCCH and NPDSCH using 8 resource elements (REs). Other remaining SFs can be assigned to NPDCCH or NPDSCH. The NPDCCH indicates for which UE there is data in the NPDSCH, SFs containing NPDSCH and its repetition count. The NPDCCH carries the Downlink Control Information (DCI), which, depending on its functionality, has three different formats:

- DCI Format N0: It contains the information related to UL scheduling grants.
- DCI Format N1: It is used for NPDSCH and Narrowband Physical Random Access Channel (NPRACH) scheduling.
- DCI Format N2: It is used for paging and direct indication, such as informing the UE about parameter modifications or issuing warning messages.

The DL scheduling information carries parameters such as modulation and coding scheme (MCS), SF assignment, NPDSCH repetition count (N_{Rep}^{DL}), and scheduling delay (k_{N1}). The MCS and the assigned number of SFs help to select transport block size (TBS) using Table 16.4.1.5.1-1 mentioned in [83]. The TBS size varies from 2 to 85 bytes. For DL, NB-IoT uses only one modulation scheme that is quadrature phase-shift keying. After the reception of the DL data, the UE acknowledges it using Narrowband Physical Uplink Shared Channel (NPUSCH).

There are up to three coverage enhancement (CE) levels (0,1, and 2) to tackle different radio conditions. To enhance the transmission reliability for different CE levels, the data and the associated control signaling have to be repeated several times. Additionally, in FDD, NB-IoT chooses half-duplex, meaning the UE can either receive or transmit and cannot perform both operations simultaneously. Therefore, some guard SFs are needed in between every switch from transmission mode (Tx) to receive mode (Rx) or vice versa to provide the time to the UE to switch the radio activity. This scheduling delay is notified by the DCIs.

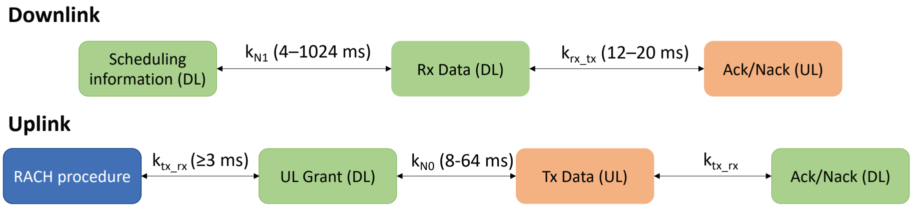


Figure 6.3: Scheduling delay and data transmission in NB-IoT

Figure 6.3 mentions these delay values between the channels. There needs to be at least 3 SFs gap when switching from Tx to Rx (k_{tx_rx}). The minimum scheduling delay between the end of Rx scheduling info and Rx data (k_{N1}) is 4 SFs. Whereas the minimum scheduling delay increases to 8 ms (k_{N0}) and 12 ms (k_{rx_tx}) when the UE needs to switch from Rx to Tx. This allows it to decode the received data, switch the radio mode and prepare for the Tx. However, by introducing these scheduling delays, the data rate decreases.

6.2.2.2 Uplink

The NB-IoT UL frame consists of two channels, namely NPRACH and NPUSCH. The NPRACH is used by the UE to access the network and to request radio resources to transmit its data. Figure 6.2 represents the time multiplexing of the NB-IoT UL physical channels, which shows that both the channels can occupy any SFs. Therefore, except for the NPRACH transmission, the UE data can be sent over the NPUSCH. For NPRACH, based on the Cyclic Prefix (CyP) length, two preamble formats are defined, format 0 and format 1. Each symbol group has a CyP followed by 5 symbols. The length of the five symbols is 1.333 ms. Each preamble is composed of 4 symbol groups transmitted without gaps which makes the complete preamble 5.6 and 6.4 ms for format 0 and 1, respectively [84]. The preamble format is broadcast in the system information. The frequency hopping is applied to the symbol group so that they can be transmitted on a different subcarrier. Depending on the different CE levels, the network can configure the NPRACH parameters such as NPRACH resource periodicity, number of preamble repetition count, number of subcarriers, and starting time of NPRACH resource.

The smallest unit that can be transmitted in UL transmissions is done using the resource unit (RU). The NPUSCH has two different formats, NPUSCH Format-1 (NPUSCH F1) and NPUSCH Format 2 (NPUSCH F2). NPUSCH F1 is used to carry UL data, and NPUSCH F2 is used to transmit acknowledgements of DL data. UL works either with a 3.75 or 15 kHz subcarrier spacing which is decided by the eNB. As the symbol duration for 3.75 kHz subcarrier spacing is four times higher compared to the 15 kHz spacing, it results in a slot length of 2 ms. The slot length of 15 kHz is 0.5 ms. For NPUSCH F1 and 15 kHz subcarrier spacing, the RU duration is up to 8 ms, depending on the number of slots. Whereas for 3.75 kHz, RU consists of one subcarrier in the frequency range and 16 slots in the time range; therefore, it has a duration of 32 ms. For NPUSCH F2, the RU has one subcarrier with 4 slots. Consequently, for the 15 kHz subcarrier spacing, the RU has a 2 ms duration and the 3.75 kHz subcarrier spacing 8 ms. Depending on the coverage level, the eNB indicates the NPUSCH repetition count that is at most up to 128.

Table 6.1: NB-IoT settings

Parameter name	**Symbol**	**Possible Settings**
Scheduled bandwidth	BW	{3.75, 15, 45, 90, 180} kHz
RU length of NPUSCH-F2	T_{RU}^{ack}	2 ms (if BW=15 kHz); 8 ms (if BW=3.75 kHz)
RU length of NPUSCH-F1	T_{RU}	8 ms (if BW=15 kHz); 32 ms (if BW=3.75 kHz)
Number of RU	N_{RU}	{1, 2, 3, 4, 5, 6, 8, 10}
Number of NPDSCH SFs	N_{SF}	{1, 2, 3, 4, 5, 6, 8, 10}
MCS index	MCS	{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
Maximum repetition of NPDCCH	R_{max}	2048
N/Ack repetition count (NPUSCH-F2)	N_{Rep}^{ack}	{1, 2, 4, 8, 16, 32, 64, 128}
Msg3 PUSCH repetition count	N_{Rep}^{Msg3}	{4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048}
NPUSCH repetition count	N_{Rep}^{NPUSCH}	{1, 2, 4, 8, 16, 32, 64, or 128}
NPDSCH repetition count	N_{Rep}^{NPDSCH}	{1, 2, 4, 8, 16, 32, 64, 128, 192, 256, 384, 512, 768, 1024, 1536, 2048}
NPDCCH repetition count	N_{Rep}^{NPDCCH}	{1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048}
Preamble repetition count	N_{Rep}^{pmbl}	{1, 2, 4, 8, 16, 32, 64, 128}
RACH preamble latency	T_{pmbl}	5.6 ms (format 0); 6.4 ms (format 1)
Subframes occupied by NPRACH	N_{NPRACH}	$T_{pmbl} \cdot N_{Rep}^{pmbl}$
NPRACH resource periodicity	$P_{NPRACH} = N_{frame}^{UL}$	{40, 80, 160, 240, 320, 640, 1280, 2560} ms
NPUSCH-F1 scheduling delay	k_{N0}	{8, 16, 32, 64} ms
NPDSCH scheduling delay	k_{N1}	{0, 4, 8, 12, 16, 32, 64, 128} ms (if R_{max} <128) {0, 16, 32, 64, 128, 256, 512, 1024} ms (if R_{max} ≥128)
Scheduling delay between Rx data and Tx N/ACK	k_{rx_tx}	{12, 14, 16, 17} ms (if BW=15 kHz) {12, 20} ms (if BW=3.75kHz)

The allowed values of some parameters needed to configure the UE to successfully receive or transmit data are defined in Table 6.1.

6.3 Related Work

6.3.1 Optimization of Power Consumption and Downlink Latency

NB-IoT has been commercially recognized as one of the most promising LPWA technologies. Many operators are offering commercial NB-IoT services in many countries. However still, the majority of research analysis are performed considering only UL traffic. This might be because most of the IoT use cases consider sensor-based monitoring, where the majority of traffic is expected to be UL. The UL latency generally consists of broadcast latency, random access (RA) latency, and data transmission latency. Maldonado et al. [85] evaluated the power consumption of an NB-IoT device considering different coverage levels and different UL IATs by reducing signaling using the mechanisms known as Control Plane Cellular IoT (CP) optimization and User Plane Cellular IoT (UP) optimization. Recently, they also defined an analytical model based on Markov chains to calculate the power consumption of UEs [86]. Their model estimates the power consumption and latency only for a device sending periodic UL data using CP optimization. In 2017, Lee at al. [87] proposed a prediction-based energy-saving mechanism for UL transmission. They allocate the resources in advance based on response time for each previous transmission and achieve up to 34 % of battery saving.

Bello et al. [88] developed a semi-Markov chain to evaluate power consumption and delay performance under periodic UL data transmission. Furthermore, they also introduced an optimization model of the PSM timers that minimizes energy consumption and the average delay. Liu et al. [89] propose a Markov chain to analyze PSM with DL and UL traffic.

They also use a genetic algorithm to obtain the power saving factor that is the fraction of time the UE spends in the PSM mode. However, the paper does not study the trade-off between DL latency and power consumption. Other works such as [90], [91], and [92] solely focused on performance models to maximize the RACH success probability concerning data transmission and RA latency.

Nevertheless, DL data transmission is also important in some use cases such as over-the-air device configuration, control loops, voice calls, and polling-based data retrieval. In these scenarios, to receive DL data or notifications, the device needs to continuously monitor the DL channel, which reduces the battery life. Oh et al. [93] modeled and evaluated the battery consumption rate for DL data reception only. This research lacks the analysis of UL transmissions and latency. Several works have evaluated the effect of eDRX parameters on system performance [94, 95].

An initial work on the development of an NB-IoT simulation platform based on OPNET is presented in [96] and validated for the low-rate data transmission of NB-IoT, focusing on its physical layer characteristics. Soussi et al. [97] presented an alternative implementation of the NB-IoT physical layer in ns-3, further building upon the ns-3 LTE module. This work is extended with support for various data link layer timers and features, such as the Radio Resource Control (RRC) connection and idle state, as well as eDRX and PSM [98]. Furthermore, Lauridsen et al. [99] presented empirical power consumption measurements of two NB-IoT UEs.

However, the works mentioned above have not provided a complete analysis of IoT applications with both UL and DL transmissions simultaneously. Also, the energy-latency trade-off when using NB-IoT with eDRX and PSM has not been studied. In contrast, this thesis presents the complete picture of the NB-IoT energy saving schemes.

6.3.2 Batteryless NB-IoT Prototype

Batteryless solutions based on energy harvesting from ambient sources for NB-IoT have not been widely studied. There are only a few works in literature that talk about this theme. One of the reasons could be that the solutions have to depend on the operator to accept the configured timers of the NB-IoT UE. Some challenges and opportunities for energy harvesting solutions for NB-IoT UEs are pointed out by Haridas et al. [100]. They considered smart home use cases powered by ambient light, both indoors (devices on window sills and book shelves) and outdoors. However, they aim to extend the battery lifetime and without considering the capacitor-enabled NB-IoT devices. Other than this, much literature only targets the analysis of its power consumption, such as [101, 102]. Gabelle, in his thesis [103], worked with the Nordic NB-IoT chip (nRF9160) to measure the power consumption. This is the same chip also used in the thesis. Gabelle used the Nordic DK board to do the measurement and measured only the radio power consumption. However, in this thesis, as it considers a batteryless prototype, the power consumption of a complete board needs to be considered. As it s difficult to disable superfluous peripherals from the Nordic DK board; this work uses an in-house designed mini-module (without any peripherals) with an nRF9160 chip.

Moreover, there are many NB-IoT devices available such as Zolitron Z-node sensor [104] that comprise temperature, move, light, GPS, and magnetic fields, street light controller [105], liquid level sensor [106], asset tracker [107], movement tracker [108], air quality sensor [109], leakage sensor [110] and many more. Still, these products are powered by batteries. A solar-powered Asset Tracker based on NB-IoT is also designed by SODAQ [111], but this device is also powered by a 2400 mAh rechargeable Lithium battery. They also have a hybrid solution using a supercapacitor, and battery [112]. The only completely battery-free NB-IoT products found are the vibration monitor and temperature monitor designed by AEInnova [113, 114] which are powered by waste heat. AEInnova also designed the vibration monitor for ventilation fan monitoring in infrastructure tunnels powered by wind energy [115]. Xnor [116] has developed a camera prototype that is powered by solar energy. The device includes connectivity using NB-IoT to transmit data. Other companies such as 8power [117], and Perpetuum [118] have future motivation toward using batteryless NB-IoT powered by vibration energy. The feasibility of using vibration energy for another long-range technology, LoRa, is studied by Orfei et al. [119] to monitor road traffic.

Chapter 7

Power Optimization

The content of this chapter is based on:

- *A. K. Sultania, C. Blondia and J. Famaey, "Optimising the Energy-Latency Tradeoff in NB-IoT With PSM and eDRX," in IEEE Internet of Things Journal, vol. 8, no. 15, pp. 12436-12454, 1 Aug.1, 2021. [Impact Factor: 9.471]*

The embedded software has to control the different components in order to manage the power consumed by the device. This role is particularly important in energy harvesting systems, as it enables the node to increase its battery life and avoid battery depletion. This chapter presents a Markov chain model to evaluate the power consumption and latency of NB-IoT devices using PSM and eDRX. By exploiting the characteristics of the steady-state distribution of the Markov chain, the probabilities in steady-state can be obtained explicitly. Based on these probabilities, the system downlink (DL) latency is calculated as a function of different timers of these power-saving features. The model is also compared to simulation results obtained from the ns-3 event-based network simulator, to determine its accuracy.

7.1 PSM and eDRX NB-IoT Model

In this section, the analytical system model is proposed to analyze energy consumption and latency. It is considered that an eNB serves multiple UEs in an NB-IoT system. For ease of exposition, the following assumptions are made.

- The eNB and UEs have a finite buffer to store UL and DL traffic.
- The random access contention is not modeled, assuming an average value instead, as its effect on latency and energy consumption is negligible compared to that of PSM and eDRX.
- The three main states of a UE are considered to be PSM, eDRX, and RRC connected.
- The data packets arrive at the eNB or UE according to a Poisson process.

Table 7.1: Notations

Symbol	**Parameter name**
λ_{UL}	Poisson arrival rate of UL packet
λ_{DL}	Poisson arrival rate of DL packet
$\lambda_{DL} + \lambda_{UL}$	Total poisson arrival rate, λ_{tot}
E_{UL}	Energy per unit time to transmit UL packet
E_{DL}	Energy per unit time to transmit DL packet
E_P	Energy needed to perform a paging action
E_C	Energy per unit time in Connected state
E_{PSM}	Energy per unit time in PSM state
E_{eDRX}	Energy per unit time in eDRX state
T_{RRC}	RRC inactivity Timer
$T_{eDRX} = T_{3324}$	eDRX state/Active Timer
T_{cycle}	eDRX cycle Timer
$T_{PSM} = T_{3412} - T_{3324}$	Deep sleep Time (PSM)
$\eta = T_{eDRX}/T_{cycle}$	Number of eDRX cycles
$\lambda_{tot} = \lambda_{UL} + \lambda_{DL}$	Total packet generation rate
N_{DL}	eNB buffer size
N_{UL}	UE buffer size
T_{DL}	Downlink Packet transmission time
T_{UL}	Uplink Packet transmission time

- The transmission time (UL or DL) for a packet is considered for fixed packet size, MCS, number of tones, tone spacing, number of slots and repetition count. However, the model can be configured with different values of these parameters, and the outputs can be used to determine the average power consumption and DL latency of the device.

Some important notations and parameters used in the model are listed in Table 7.1. Based on the assumptions above, a system to start from any of the states is considered.

7.1.1 Introduction of State Changes

The considered UE states are PSM, eDRX, and RRC Connected. The system behaves as follows based on the UE state.

- *PSM state*: This state starts with empty buffers on both sides. The maximal duration of the PSM state is fixed and denoted by T_{PSM}. This is the time a UE is in a DEEP SLEEP and equal to $T_{3412} - T_{3324}$ (as shown in Figure 6.1). During the PSM state, as long as no UL packets are generated, DL packets generated are stored at the eNB buffer. When a UL packet is generated, the system switches to the RRC connected state. On the expiration of the PSM timer, if there exist one or more DL packets in the DL buffer, the system switches to the RRC connected state, otherwise to the eDRX state.

- *RRC Connected state*: It is assumed that the generated packets at the eNB or the UE are transmitted immediately. There are two possibilities at the start of RRC Connected state, either the UL buffer is empty with some ready to transfer k_{DL} DL packets, where $1 \leq k_{DL} \leq N_{DL}$ or one UL packet and k_{DL} DL packets, where $0 \leq k_{DL} \leq N_{DL}$.

- *eDRX state*: Similar to the PSM state, the eDRX state starts with empty buffers. When T_{eDRX} expires, the system switches to the PSM state. However, if a UL packet is generated, the system switches immediately to the RRC connected state. The UE periodically performs paging to detect DL packets. The number of consecutive eDRX cycles is limited to a fixed value denoted by η (defined in Table 7.1). If no UL packet is generated during an eDRX cycle and DL packets have been detected at paging, the system switches to the RRC connected state, transmitting the packets. Otherwise, the next eDRX cycle starts.

The energy consumption efficiency and its trade-off with latency can be derived after calculating the transition probability and the time duration of the different states that together form a Markov chain.

7.1.2 RRC Connected State

The highest modulation of NB-IoT is Quadrature Phase Shift Keying (QPSK). In LTE, with the improvement in SINR, the nodes are capable of using higher modulation schemes, resulting in improved spectrum efficiency. However, in NB-IoT with improved SINR, the repetition factor reduces [120]. The transmission schemes and repetition counts are different for different channels of DL and UL. Therefore, both the packet transmissions are modelled separately.

7.1.2.1 Downlink

As shown in Figure 6.2, at every 20 SFs, only 15 SFs, can be used for NPDSCH or NPDCCH due to the presence of NPBCH, NPSS and NSSS. Therefore, based on the data transmitting SF position, it might need to wait up to 3 ms before transmitting. When the transmission collides with NSSS, it can take 3 ms due to waiting for NSSS and NPBCH to complete. Whereas, if it collides with NPBCH or NPSS, the transmission can be completed in 2 ms (during the next available SF). For the other 15 SFs, the transmission can happen in that SF spending 1 ms. Therefore, the effective average data transmission time of 1 SF in the presence of NPSS, NSSS and NPBCH is given by Equation 7.1.

$$\begin{aligned} T_{DL}^{SF} = (1\ ms) \cdot \frac{N_{frame}^{DL} - (N_{NPBCH} + N_{NPSS} + N_{NSSS})}{N_{frame}^{DL}} \\ + (2\ ms) \cdot \frac{N_{NPBCH} + N_{NPSS}}{N_{frame}^{DL}} + (3\ ms) \cdot \frac{N_{NSSS}}{N_{frame}^{DL}}, \end{aligned} \tag{7.1}$$

where, N_{frame}^{DL} is the total DL frame size. N_{NPSS}, N_{NSSS} and N_{NPBCH} are the number of NPSS, NSSS and NPBCH packets, respectively, during the N_{frame}^{DL}. Therefore, considering N_{frame}^{DL} of 20 ms, the value of T_{DL}^{SF} is 1.3ms. The application data is appended to protocols headers. Generally, PDCP performs robust header compression to reduce the header size. The reduction and the compressed size of headers depend on the traffic type as defined in [121]. Assuming N_{data}^{DL} as the total size of application data and N_{header}^{DL} as the size of the headers including UDP, IP, PDCP, RLC, MAC, etc and TBS_{NPDSCH} as the transport block size for the NPDSCH resulting from the selection of MCS and number of SFs (N_{SF}) as defined in Table 16.4.1.5.1-1 in [83], the number of packet segments can be calculated. The packets are segmented at the RLC layer. Therefore, the data segment does not include the header size of the MAC layer (H_{mac}), which is appended after the segmentation. As such, the number of packet segments (MAC layer packets) is given by Equation 7.2.

$$N_{seg}^{DL} = \left\lceil \frac{N_{data}^{DL} + N_{header}^{DL} - H_{mac}}{TBS_{NPDSCH}(MCS, N_{SF}) - H_{mac}} \right\rceil . \tag{7.2}$$

The header size for the MAC layer in NB-IoT is 2 bytes [122]. To avoid the blockage by the DL resources, some transmission gaps (T_{DL}^{G}) are introduced on continuous DL transmission for T_{rx}^{C} [86]. Therefore, the time taken for each data segment transmission on the NPDSCH is given by Equation 7.3.

$$T_{NPDSCH} = T_{DL}^{SF} \cdot N_{SF} \cdot N_{Rep}^{NPDSCH} \cdot (1 \; + \; \frac{T_{DL}^{G}}{T_{rx}^{C}}), \tag{7.3}$$

where, N_{Rep}^{NPDSCH} is the number of repetitions for NPDSCH selected depending on the coverage area. Hence, the total transmission time of DL data (T_{DL}) for all the segments is calculated by adding the time of each activity as shown in Figure 6.3. That is for receiving scheduling information, receiving DL data, sending the corresponding acknowledgement and all the activity scheduling delays. Assuming the static TBS, this is given by Equation 7.4.

$$\begin{aligned} T_{DL} = N_{seg}^{DL} \cdot (T_{DL}^{SF} \cdot 1 \cdot N_{Rep}^{NPDCCH} \; + k_{N1} \; + \; T_{NPDSCH} + \; k_{rx\_tx} + \; T_{RU}^{ack} \cdot N_{RU} \cdot N_{Rep}^{ack}) \\ + \; (N_{seg}^{DL} - 1) \cdot k_{next}^{DL}, \end{aligned} \tag{7.4}$$

where, k_{N1} and k_{rx_tx} are the scheduling delay required to change the channels as represented in Figure 6.3. Now, the energy consumption for DL data (E_{DL}^{C}) in RRC connected state is calculated as defined in Equation 7.5.

$$\begin{aligned} E_{DL}^{C} = N_{seg}^{DL} \cdot (T_{DL}^{SF} \cdot 1 \cdot N_{Rep}^{NPDCCH} \cdot E_{DL} \; + \; k_{N1} \cdot E_{C} \; + \; T_{NPDSCH} \cdot E_{DL} \; + \\ k_{rx\_tx} \cdot E_{C} \; + \; T_{RU}^{ack} \cdot N_{RU} \cdot N_{Rep}^{ack} \cdot E_{UL}) \; + \; (N_{seg}^{DL} - 1) \cdot k_{next}^{sch} \cdot E_{C} . \end{aligned} \tag{7.5}$$

7.1.2.2 Uplink

The UL data time in RRC connected state includes the time spent for different activities, including the Random access (RA) procedure to send scheduling requests, receiving a UL grant, sending UL data and receiving an acknowledgment, as shown in Figure 6.3. Assuming N_{data}^{UL} as the total size of the UL application data, the number of UL packet segments (N_{seg}^{UL}) can be calculated similar to Equation 7.2 by replacing $TBS_{NPDSCH}(MCS, N_{SF})$ with $TBS_{NPUSCHF1}(MCS, N_{RU})$ which is defined as Table 16.5.1.2-2 in [83]. However, to maintain resynchronization with the DL reference signals, a certain gap (T_{UL}^{G}) is needed at a continuous NPUSCH transmission for T_{tx}^{C}. Whereas, for NPRACH, a 40 ms gap is introduced

Table 7.2: State transitions

From ($State$, k_{UL}, k_{DL}, tl_{eDRX})	**To** ($State$, k_{UL}, k_{DL}, tl_{eDRX})	**Probability**
$(PSM, 0, 0, 0)$	$(RRC, 0, k_{DL}, 0), 0 < k_{DL} < N_{DL}$	$Pn(\lambda_{DL}, k_{DL}, T_{PSM}) \cdot Pn(\lambda_{UL}, 0, T_{PSM})$
$(PSM, 0, 0, 0)$	$(RRC, 0, N_{DL}, 0)$	$RPn(\lambda_{DL}, N_{DL}, T_{PSM}) \cdot Pn(\lambda_{UL}, 0, T_{PSM})$
$(PSM, 0, 0, 0)$	$(RRC, 1, k_{DL}, 0), 0 \leq k_{DL} < N_{DL}$	$A(k_{DL}, 1, T_{PSM})$
$(PSM, 0, 0, 0)$	$(RRC, 1, N_{DL}, 0)$	1-$Pn(\lambda_{UL}, 0, T_{PSM}) -$ $\sum_{k_{DL}=0}^{N_{DL}-1} A(k_{DL}, 1, T_{PSM})$
$(PSM, 0, 0, 0)$	$(eDRX, 0, 0, 1)$	$Pn(\lambda_{tot}, 0, T_{PSM})$
$(RRC, 0, k_{DL}, 0), 0 < k_{DL} \leq N_{DL}$	$(eDRX, 0, 0, 1)$	$Pn(\lambda_{tot}, 0, T_{RRC})$
$(RRC, 0, k_{DL}, 0), 0 < k_{DL} \leq N_{DL}$	$(RRC, 1, 0, 0)$	$(\lambda_{UL}/\lambda_{tot}) \cdot (1 - Pn(\lambda_{tot}, 0, T_{RRC}))$
$(RRC, 0, k_{DL}, 0), 0 < k_{DL} \leq N_{DL}$	$(RRC, 0, 1, 0)$	$(\lambda_{DL}/\lambda_{tot}) \cdot (1 - Pn(\lambda_{tot}, 0, T_{RRC}))$
$(RRC, 1, k_{DL}, 0), 0 \leq k_{DL} \leq N_{DL}$	$(eDRX, 0, 0, 1)$	$Pn(\lambda_{tot}, 0, T_{RRC})$
$(RRC, 1, k_{DL}, 0), 0 \leq k_{DL} \leq N_{DL}$	$(RRC, 1, 0, 0)$	$(\lambda_{UL}/\lambda_{tot}) \cdot (1 - Pn(\lambda_{tot}, 0, T_{RRC}))$
$(RRC, 1, k_{DL}, 0), 0 \leq k_{DL} \leq N_{DL}$	$(RRC, 0, 1, 0)$	$(\lambda_{DL}/\lambda_{tot}) \cdot (1 - Pn(\lambda_{tot}, 0, T_{RRC}))$
$(eDRX, 0, 0, tl_{eDRX}), 1 \leq tl_{eDRX} \leq \eta$	$(eDRX, 0, 0, tl_{eDRX} + 1)$	$Pn\left(\lambda_{tot}, 0, T_{cycle}\right)$
$(eDRX, 0, 0, \eta)$	$(PSM, 0, 0, 0)$	$Pn\left(\lambda_{tot}, 0, T_{cycle}\right)$
$(eDRX, 0, 0, tl_{eDRX}), 1 \leq tl_{eDRX} \leq \eta$	$(RRC, 0, k_{DL}, 0), 0 < k_{DL} < N_{DL}$	$Pn\left(\lambda_{DL}, k_{DL}, T_{cycle}\right) \cdot Pn\left(\lambda_{UL}, 0, T_{cycle}\right)$
$(eDRX, 0, 0, tl_{eDRX}), 1 \leq tl_{eDRX} \leq \eta$	$(RRC, 0, N_{DL}, 0)$	$RPn\left(\lambda_{DL}, N_{DL}, T_{cycle}\right) \cdot$ $Pn\left(\lambda_{UL}, 0, T_{cycle}\right)$
$(eDRX, 0, 0, tl_{eDRX}), 1 \leq tl_{eDRX} \leq \eta$	$(RRC, 1, k_{DL}, 0), 0 < k_{DL} < N_{DL}$	$A\ \left(k_{DL}, 1, T_{cycle}\right)$
$(eDRX, 0, 0, tl_{eDRX}), 1 \leq tl_{eDRX} \leq \eta$	$(RRC, 1, N_{DL}, 0)$	1- $Pn\left(\lambda_{UL}, 0, T_{cycle}\right) -$ $\sum_{k_{DL}=0}^{N_{DL}-1} A\left(k_{DL}, 1, T_{cycle}\right)$

at every 64 preambles [123]. The calculation of the average NPUSCH transmission time of each UL data segment is given by Equation 7.6.

$$T_{NPUSCH} = T_{RU} \cdot N_{RU} \cdot N_{Rep}^{NPUSCH} \cdot \frac{N_{frame}^{UL}}{N_{frame}^{UL} - N_{NPRACH}} \cdot (1 + \frac{T_{UL}^{G}}{T_{tx}^{C}}), \tag{7.6}$$

where, N_{frame}^{UL} and N_{NPRACH} are the total UL frame size and number of NPRACH frames during the considered total UL frame.

The RA procedure consists of messages that occupy only one SF in the DL and one RU in the UL. There exists a four-step message transaction (Msg1, Msg2, Msg3, and Msg4) between the UE and the eNB. The time required to send Msg1 that transmits the NPRACH preamble and its timing is given by Equation 7.7.

$$T_{msg1} = T_{pmbl} \cdot 1 \cdot N_{Rep}^{pmbl}. \tag{7.7}$$

Then the control and data channels are transferred in DL as Msg2. There average transmission time is given by Equation 7.8.

$$T_{msg2} = T_{DL}^{SF} \cdot 1 \cdot N_{Rep}^{NPDCCH} + T_{DL}^{SF} \cdot 1 \cdot N_{Rep}^{NPDSCH}. \tag{7.8}$$

Msg3, where the RRC connection request and the acknowledgement of Msg2 are sent as UL transmission. Their transmission times (T_{msg3}) and (T_{UL}^{ack}) can be calculated using Equation 7.6 for their repetition count N_{Rep}^{Msg3} and N_{Rep}^{ack}, respectively. Also, Msg4 is similar to Msg2, that is, an NPDCCH followed by the NPDSCH, so its transmission time (T_{msg4}) is the same as that of Msg2.

The total time of the RA (T_{RA}) procedure can be calculated as the sum of all the four

RACH messages and scheduling delays between them. This is given by Equation 7.9.

$$\begin{aligned} T_{RA} = {} & T_{msg1} \ + \ k_{tx\_rx} \ + (k_{N1} \ + T_{msg2}) \ + \ k_{rx\_tx} \ + \ T_{msg3} \ + \ k_{tx\_rx} \ + \\ & (k_{N1} \ + \ T_{msg4}) \ + \ k_{rx\_tx} \ + \ T_{UL}^{ack} \, . \end{aligned} \tag{7.9}$$

Therefore, the energy of the RA procedure is given by Equation 7.10.

$$\begin{aligned} E_{RA} = {} & T_{msg1} \cdot E_{UL} + k_{tx\_rx} \cdot E_C + k_{N1} \cdot E_C + T_{msg2} \cdot E_{DL} + k_{rx\_tx} \cdot E_C \\ & + T_{msg3} \cdot E_{UL} + k_{tx\_rx} \cdot E_C + k_{N1} \cdot E_C + T_{msg4} \cdot E_{DL} + k_{rx\_tx} \cdot E_C + T_{UL}^{ack} \cdot E_{UL} \, . \end{aligned} \tag{7.10}$$

The time to receive the UL grant or acknowledgement in the NPDCCH (T_{NPDCCH}) is calculated as given in Equation 7.11.

$$T_{NPDCCH} = T_{DL}^{SF} \cdot 1 \cdot N_{Rep}^{NPDCCH} \, , \tag{7.11}$$

So, the total transmission time to transmit UL data is given by Equation 7.12.

$$\begin{aligned} T_{UL} = {} & N_{seg}^{UL} \cdot (T_{RA} + k_{tx\_rx} + T_{NPDCCH} + k_{N0} + T_{NPUSCH} + k_{tx\_rx} + T_{NPDCCH}) \ + \\ & (N_{seg}^{UL} - 1) \cdot k_{next}^{sch} \, , \end{aligned} \tag{7.12}$$

where, k_{next}^{sch} are the scheduling delay for next UL grant transmission. Therefore, the energy to transmit UL data (E_{UL}^{C}) in RRC connected state is given by Equation 7.13.

$$\begin{aligned} E_{UL}^{C} = {} & N_{seg}^{UL} \cdot (E_{RA} \ + \ k_{tx\_rx} \cdot E_C \ + \ T_{NPDCCH} \cdot E_{DL} \ + \ k_{N0} \cdot E_C \ + \\ & T_{NPUSCH} \cdot E_{UL} + k_{tx\_rx} \cdot E_C \ + \ T_{NPDCCH} \cdot E_{DL}) \ + \ (N_{seg}^{UL} - 1) \cdot k_{next}^{sch} \cdot E_C \, . \end{aligned} \tag{7.13}$$

7.1.3 eDRX and PSM State

This section discusses the model to calculate the latency of DL data traffic. The Markov stochastic process is described by means of the following variables:

- *State*: It represents the UE states which encompass PSM, eDRX, or RRC connected.
- *DL buffer occupancy*: It represents the number of DL packets stored in the eNB transmission queue. New DL packets arriving at a full buffer are lost and do not contribute to the DL packet delay. This is represented by k_{DL}, such that, $0 \leq k_{DL} \leq N_{DL}$.
- *UL buffer occupancy*: Similar to DL buffer occupancy, it represents the number of UL packets stored in the UE transmission queue and is denoted by k_{UL}, such that, $0 \leq k_{UL} \leq N_{UL}$.
- *Counter eDRX intervals*: It represents the sequence number of the eDRX interval, denoted by tI_{eDRX}, such that, $0 \leq tI_{eDRX} \leq \eta$. It is zero for the states PSM and RRC Connected.

The transitions between the states, with their corresponding probabilities, are given in Table 7.2. The total number of states is given by $dim = 2 \times (1 + N_{DL}) + \eta$. The probability of k or more arrivals in an interval T and rate λ is given as Equation 7.14.

$$Pn(\lambda, k, T) = e^{-\lambda \cdot T} \cdot (\lambda \cdot T)^k / k! \, . \tag{7.14}$$

And the probability of k or more arrivals not in an interval T is given as Equation 7.15.

$$RPn(\lambda, N, T) = 1 - \sum_{k=0}^{N-1} Pn(\lambda, k, T) \, . \tag{7.15}$$

The probability that a UL arrival occurs after k_{DL} DL arrivals, but before a time interval of length T expires is given by Equation 7.16.

$$\begin{aligned} A(k_{DL}, 1, T) &= \int_0^T \lambda_{UL} \cdot e^{-\lambda_{UL} \cdot t} \cdot Pn(\lambda_{DL}, k_{DL}, t) \, dt \\ &= \frac{\lambda_{UL}}{\lambda_{DL}} \cdot \left(\frac{\lambda_{DL}}{\lambda_{tot}} \right)^{k_{DL}+1} \cdot RPn(\lambda_{tot}, k_{DL}+1, T) \, , \end{aligned} \tag{7.16}$$

where, $Pn(\lambda, k, T)$ is the probability of k arrivals in an interval T and rate λ and $RPn\,(\lambda, k, T)$ is the probability of k arrivals do not happen in an interval T.

The holding time of different states, which represents the average time the system remains in a state before transitioning to another state, is calculated below.

- *Holding time of (PSM,0,0,0):* There are two possibilities either there are no UL arrivals during the PSM state, or there are. The probability of the first case is $e^{-\lambda_{UL} \cdot T_{PSM}}$ with the holding time of the total duration, which is T_{PSM}. Therefore the probability of a UL arrival is $1 - e^{-\lambda_{UL} \cdot T_{PSM}}$ with the holding time as given by Equation 7.17.

$$\begin{aligned} HT_{3412} &= \int_0^{T_{PSM}} t \cdot \lambda_{UL} \cdot e^{-\lambda_{UL} \cdot t} dt \\ &= \frac{1}{\lambda_{UL}} - \frac{e^{-\lambda_{UL} \cdot T_{PSM}}}{\lambda_{UL}} - T_{PSM} \cdot e^{-\lambda_{UL} \cdot T_{PSM}} \, . \end{aligned} \tag{7.17}$$

  Hence, the average PSM Holding time, denoted by $H_{(PSM,0,0,0)}$ is given by Equation 7.18.

$$H_{(PSM,0,0,0)} = e^{-\lambda_{UL} \cdot T_{PSM}} \cdot T_{PSM} \; + \; (1 - e^{-\lambda_{UL} \cdot T_{PSM}}) \cdot HT_{3412} \, . \tag{7.18}$$

- *Holding time of* $(RRC, k_{UL}, k_{DL}, 0)$*:* Assuming at the start of an RRC state interval, there are k_{DL} DL and k_{UL} UL packets present in the buffer such that, $0 \leq k_{DL} \leq N_{DL}$ and $k_{UL} \in \{0, 1\}$. Then each of these $k_{DL} + k_{UL}$ packets needs to be transmitted, as well as the new packets that are generated during the transmission time of these packets. Therefore, the time needed to empty the buffers is called the busy period starting with k_{DL} DL and k_{UL} UL packets. The UL and DL buffers are dimensioned in such a way that the probability of queue drops is negligible. The time needed to empty the buffers is obtained by applying the formula for the average length of a busy period of the M/G/1 queue starting with $k_{DL} + k_{UL}$ packets is given by Equation 7.19.

$$BP_{M/G/1}(k_{DL} + k_{UL}) = \frac{k_{DL} + k_{UL}}{(1-\lambda_{tot}\cdot T_p)} \cdot T_p, \tag{7.19}$$

where, the average service time is given by $T_p = (\lambda_{UL}\cdot T_{UL} + \lambda_{DL}\cdot T_{DL})/\lambda_{tot})$.

At the end of this busy period, there are three possibilities.

- *No UL nor a DL arrival occurs during RRC Connected* $[0, T_{RRC}]$: This happens with the probability $e^{-(\lambda_{UL} + \lambda_{DL})\cdot T_{RRC}}$ and the holding time, in this case, is T_{RRC}.
- *An UL arrival occurs first during* $[0, T_{RRC}]$: Its probability is given by Equation 7.20 and the holding time is given by Equation 7.21.

$$P1 = \int_0^{T_{RRC}} \lambda_{UL}\cdot e^{-\lambda_{UL}\cdot t}\cdot e^{-\lambda_{DL}\cdot t} dt = \frac{\lambda_{UL}}{\lambda_{tot}}(1 - e^{\lambda_{tot}\cdot T_{RRC}}). \tag{7.20}$$

$$HT1 = \int_0^{T_{RRC}} t\cdot\lambda_{UL}\cdot e^{-\lambda_{UL}\cdot t} dt = \frac{1}{\lambda_{UL}} - \frac{e^{-\lambda_{UL}\cdot T_{RRC}}}{\lambda_{UL}} - T_{RRC}\cdot e^{-\lambda_{UL}\cdot T_{RRC}}. \tag{7.21}$$

- *A DL arrival occurs first during* $[0, T_{RRC}]$: Its probability is given by Equation 7.22 and the holding time is given by Equation 7.23.

$$P2 = \int_0^{T_{RRC}} \lambda_{DL}\cdot e^{-\lambda_{UL}\cdot t}\cdot e^{-\lambda_{DL}\cdot t} dt = \frac{\lambda_{DL}}{\lambda_{tot}}(1 - e^{\lambda_{tot}\cdot T_{RRC}}). \tag{7.22}$$

$$HT2 = \int_0^{T_{RRC}} t\cdot\lambda_{DL}\cdot e^{-\lambda_{DL}\cdot t} dt = \frac{1}{\lambda_{DL}} - \frac{e^{-\lambda_{DL}\cdot T_{RRC}}}{\lambda_{DL}} - T_{RRC}\cdot e^{-\lambda_{DL}\cdot T_{RRC}}. \tag{7.23}$$

The holding time of the state $(RRC, k_{UL}, k_{DL}, 0)$ is given by Equation 7.24

$$H_{(RRC,k_{UL},k_{DL},0)} = e^{-(\lambda_{UL}+\lambda_{DL})\cdot T_{PSM}}\cdot T_{RRC} + (P1\cdot HT1) + (P2\cdot HT2). \tag{7.24}$$

- *Holding time of* $(eDRX, 0, 0, tl_{eDRX})$*:* The computation of the holding time of (eDRX,0,0,tl$_{\text{eDRX}}$) is similar to that of (PSM,0,0,0). Hence, this holding time is given by Equation 7.25.

$$\begin{aligned} H_{(eDRX,0,0,tl_{eDRX})} = e^{-\lambda_{UL}\cdot T_{cycle}}\cdot T_{cycle} + \left(1 - e^{-\lambda_{UL}\cdot T_{cycle}}\right)\cdot\bigg\{\frac{1}{\lambda_{UL}} - \\ \frac{e^{-\lambda_{UL}\cdot T_{cycle}}}{\lambda_{UL}} - T_{cycle}\cdot e^{-\lambda_{UL}\cdot T_{cycle}}\bigg\}. \end{aligned} \tag{7.25}$$

The results of the holding times can help in determining the probability that a random instant falls in an interval that starts with the state $(S, k_{UL}, k_{DL}, tl_{eDRX})$, denoted by

$\nu_{(S,k_{UL},k_{DL},tl_{eDRX})}$. Let the parameter D be the average time between state transition instants as defined by Equation 7.26.

$$\begin{aligned} D = H_{(PSM,0,0,0)} \cdot \pi_{(PSM,0,0,0)} \; + \sum_{n=1}^{N_{DL}} H_{(RRC,0,n,0)} \cdot \pi_{(RRC,0,n,0)} \\ + \sum_{n=1}^{N_{DL}} H_{(RRC,1,n,0)} \cdot \pi_{(RRC,1,n,0)} + \sum_{n=1}^{\eta} H_{(eDRX,0,0,n)} \cdot \pi_{(eDRX,0,0,n)}\,, \end{aligned} \tag{7.26}$$

where, π is the steady-state vector of the embedded Markov chain S calculated as the left eigenvector corresponding to the eigenvalue 1. The holding times are defined by Equation 7.27-7.30 and the probabilities that a random instant falls in one of the states PSM, RRC or eDRX by Equation 7.31-7.33.

$$\nu_{(PSM,0,0,0)} = H_{(PSM,0,0,0)} \cdot \pi_{(PSM,0,0)}/D\,. \tag{7.27}$$

$$\begin{gathered} \nu_{(RRC,0,k_{DL},0)} = H_{(RRC,0,k_{DL},0)} \cdot \pi_{(RRC,0,k_{DL},0)}/D\,, \\ where,\; 1 \leq k_{DL} \leq N_{DL}\,. \end{gathered} \tag{7.28}$$

$$\begin{gathered} \nu_{(RRC,1,k_{DL},0)} = H_{(RRC,1,k_{DL},0)} \cdot \pi_{(RRC,1,k_{DL},0)}/D\,, \\ where,\; 0 \leq k_{DL} \leq N_{DL}\,. \end{gathered} \tag{7.29}$$

$$\begin{gathered} \nu_{(eDRX,0,tl_{eDRX})} = H_{(eDRX,tl_{eDRX})} \cdot \pi_{(eDRX,0,0,tl_{eDRX})}/D\,, \\ where,\; 1 \leq tl_{eDRX} \leq \eta\,. \end{gathered} \tag{7.30}$$

$$P_{PSM} = \nu_{(PSM,0,0,0)}\,. \tag{7.31}$$

$$P_{RRC} = \sum_{k_{DL}=1}^{N_{DL}} \nu_{(RRC,0,k_{DL},0)} + \sum_{k_{DL}=0}^{N_{DL}} \nu_{(RRC,1,k_{DL},0)}\,. \tag{7.32}$$

$$P_{eDRX} = \sum_{tl_{eDRX}=1}^{\eta} \nu_{(eDRX,0,0,tl_{eDRX})}\,. \tag{7.33}$$

7.1.4 Downlink Data Delay Analysis

As the arrival process of DL packets is assumed to be Poisson distributed, the Poisson Arrivals See Time Averages (PASTA) property can be applied to compute the delay of an arriving DL packet when it arrives at a random time instant in any of the states.

7.1.4.1 In the PSM state

The waiting time of a packet consists of the remaining time of the PSM cycle and the transmission time of all DL or UL packets that were present in the queue upon its arrival. There are two possible scenarios in the PSM state.

- *No UL arrival occurs during the PSM interval:* As there is no UL; therefore only DL transmissions can happen. If the arrival instant time of a packet is t, which is relative to the start of the PSM state, then the remaining time of the PSM cycle is given by $T_{PSM} - t$. Also, if k packets are waiting in the queue, then an additional waiting time of $k \cdot T_{DL}$ needs to be considered. The probability that this occurs is given by Equation 7.34.

$$P_{wDL} = \frac{1}{T_{PSM}} \cdot \frac{(\lambda_{DL} \cdot t)^k}{k!} \cdot e^{-\lambda_{DL} \cdot t}. \tag{7.34}$$

Hence, assuming an infinite capacity buffer, the average waiting time is given by Equation 7.35.

$$\begin{aligned} WT_{3412} &= \sum_{k=0}^{\infty} \int_0^{T_{PSM}} (T_{PSM} - t) + k \cdot T_{DL} \cdot P_{wDL} \\ &= (T_{PSM} \; + \; T_{DL} \cdot \lambda_{DL} \cdot T_{PSM})/2. \end{aligned} \tag{7.35}$$

The two components of the average waiting time include the average remaining time of the PSM cycle ($T_{PSM}/2$) and the transmission time of the packets that have arrived during this time. Therefore, the PSM delay component is then given by Equation 7.36.

$$Delay^1_{PSM} = WT_{3412} \; + \; T_{DL}. \tag{7.36}$$

And the probability that there is no UL arrival during the PSM time is given by Equation 7.37.

$$P_{NoUL} = e^{-\lambda_{UL} \cdot T_{PSM}}. \tag{7.37}$$

- *A UL arrival occurs during the PSM interval:* The probability that there is a UL arrival during the PSM time is $1 - P_{NoUL}$. To calculate the latency of DL packets, assume that the DL packet arrives at a random time instant between the start of the PSM cycle and a UL arrival. When the UL data is ready to be sent, the UE exits the PSM state. The average latency can be calculated using the time of a renewal process with the first moment or mean μ and variance σ^2 which is $X = \mu^2 \; + \; \sigma^2/2\mu$ where μ is equal to HT_{3412} and σ^2 is given by Equation 7.38.

$$\begin{aligned} \sigma^2 &= \int_0^{T_{PSM}} t^2 \cdot \lambda_{UL} \cdot e^{-\lambda_{UL} \cdot t} dt, \\ &= \frac{2}{(\lambda_{UL})^2} \cdot (1 - e^{-\lambda_{UL} \cdot T_{PSM}}) \\ &- T^2_{PSM} \cdot e^{-\lambda_{UL} \cdot T_{PSM}} - \frac{2}{\lambda_{UL}} \cdot T_{PSM} \cdot e^{-\lambda_{UL} \cdot T_{PSM}}. \end{aligned} \tag{7.38}$$

Therefore the average delay when there is a UL arrival during PSM is given by Equation 7.39.

$$Delay^2_{PSM} = X \; + \; T_{DL} \cdot (\lambda_{DL} X \; + \; 1). \tag{7.39}$$

The average delay a DL packet that arrives during a PSM interval is then given by Equation 7.40.

$$D_{(PSM,0,0,0)} = P_{NoUL} \cdot Delay^1_{PSM} \; + \; (1 - P_{NoUL}) \cdot Delay^2_{PSM}. \tag{7.40}$$

7.1.4.2 In the eDRX state

Similar to the PSM scenario, the eDRX state delay is given by Equation 7.41.

$$\begin{aligned} D_{(eDRX0,0,tl_{eDRX})} = e^{-\lambda_{UL}\cdot T_{cycle}} \cdot [T_{cycle}/2 + T_{DL} \cdot (\lambda_{DL} \cdot T_{cycle}/2+1)] + \\ (1-e^{-\lambda_{UL}\cdot T_{cycle}}) \cdot Y + T_{DL} \cdot (\lambda_{DL} Y + 1)\,, \end{aligned} \tag{7.41}$$

where, Y is the mean time interval between the DL arrival at a random instant and the UL arrival that ends the eDRX interval. Y is computed in the same way as X.

7.1.4.3 In the RRC connected state

There are two possibilities. Firstly, the packet arrives during the busy period when the $k_{UL} + k_{DL}$ packets are being transmitted, as well as those generated during these transmissions, or it arrives after that busy period, but before the RRC timer of length T_{RRC} expires. Using the result for the length of a busy period in the M/G/1 queue starting with $k_{DL} + k_{UL}$ packets, the probability that the packet arrives during the busy period is given by Equation 7.42.

$$P_{DuringBP} = \frac{BP_{M/G/1}(k_{DL} + k_{UL})}{BP_{M/G/1}(k_{DL} + k_{UL}) + HT2}\,. \tag{7.42}$$

And the probability that the packet arrives after the busy period is given by Equation 7.43.

$$P_{AfterBP} = \frac{HT2}{BP_{M/G/1}(k_{DL} + k_{UL}) + HT2}\,. \tag{7.43}$$

The mean residual lifetime of a renewal process with the first moment μ and variance σ^2 is given by $(\mu^2+\sigma^2)/2\mu$ where moment and variance are defined as in Equation 7.44.

$$\begin{aligned} \mu &= (k_{DL} + k_{UL}) \cdot T_p/(1-\lambda_{tot}T_p)\,. \\ \sigma^2 &= (k_{DL} + k_{UL}) \cdot \lambda_{tot} \cdot T_p^3/(1-\lambda_{tot}T_p)^3\,. \end{aligned} \tag{7.44}$$

Hence, when a packet arrives during the busy period that started with $k_{DL} + k_{UL}$ packets, the residual time of the busy period is given by Equation 7.45.

$$\begin{aligned} RBP(k_{DL}+k_{UL}) &= \mu^2 + \sigma^2/2\mu \\ &= \frac{(k_{DL} + k_{UL}) \cdot (1-\lambda_{tot}T_p) + \lambda_{tot} \cdot T_p}{2(1-\lambda_{tot}T_p)^2} \cdot T_p\,. \end{aligned} \tag{7.45}$$

And the RRC delay component of an RRC interval that starts with k_{DL} packets in the DL buffer, $1 \leq k_{DL} \leq N_{DL}$, is given by Equation 7.46.

$$\begin{aligned} D_{RRC}(k_{DL} + k_{UL}) &= P_{DuringBP} \cdot \big(RBP(k_{DL} + k_{UL}) + T_p\big) + P_{AfterBP} \cdot T_p \\ &= P_{DuringBP} \cdot RBP(k_{DL} + k_{UL}) + T_p\,. \end{aligned} \tag{7.46}$$

As such, the average delay an arriving DL packet experiences is given by Equation 7.47.

$$\begin{aligned}Delay_{tot} = P_{PSM} \cdot D_{(PSM,0,0,0)} \ + \sum_{k_{DL}=1}^{N_{DL}} \nu_{(RRC,0,k_{DL},0)} \cdot D_{RRC}(k_{DL}+0) \ + \\ \sum_{k_{DL}=0}^{N_{DL}} \nu_{(RRC,1,k_{DL},0)} \cdot D_{RRC}(k_{DL}+1) \ + \ P_{eDRX} \cdot D_{(eDRX,0,0,tl_{eDRX})} \cdot\end{aligned} \tag{7.47}$$

7.1.5 Energy Consumption Analysis

While in the PSM state, the system consumes $EC_{(PSM,0,0,0)} = E_{PSM}$ energy per time unit whereas in an eDRX interval, it consumes $EC_{(eDRX,0,0,tl_{eDRX})} = (E_{eDRX} \ + \ E_P/H_{(eDRX,0,0,tl_{eDRX})})$ per time unit. The energy consumption during the RRC connected state consists of the consumption in emptying the UL and DL buffers which are given by Equation 7.48.

$$\begin{aligned}E_{RRC1} = k_{UL} \cdot E^C_{UL} \ + \ k_{DL} \cdot E^C_{DL} \ + \ BP_{M/G/1}(k_{DL} \ + \ k_{UL}) \cdot \lambda_{UL} \cdot E^C_{UL} \ + \\ BP_{M/G/1}(k_{DL} \ + \ k_{UL}) \cdot \lambda_{DL} \cdot E^C_{DL} \,.\end{aligned} \tag{7.48}$$

The total energy consumption per time unit, during an RRC interval that starts in $(RRC, k_{UL}, k_{DL}, 0)$ is given by Equation 7.49.

$$EC_{(RRC,k_{UL},k_{DL},0)} = \frac{E_{RRC1} + P1 \cdot E^C_{DL} + P2 \cdot E^C_{UL}}{H_{(RRC,k_{UL},k_{DL},0)}} \ + \ E_C \,. \tag{7.49}$$

Hence, the total energy consumption per time unit of the NB-IoT network is given by Equation 7.50.

$$\begin{aligned}E_{tot} = P_{PSM} \cdot EC_{(PSM,0,0,0)} \ + \sum_{k_{DL}=1}^{N_{DL}} \nu_{(RRC,0,k_{DL},0)} \cdot EC_{(RRC,0,k_{DL},0)} \ + \\ \sum_{k_{DL}=0}^{N_{DL}} \nu_{(RRC,1,k_{DL},0)} \cdot EC_{(RRC,1,k_{DL},0)} \ + \ P_{eDRX} \cdot EC_{(eDRX,0,0,tl_{eDRX})} \cdot\end{aligned} \tag{7.50}$$

7.2 Numerical Results and Validation

This section evaluates the energy consumption and DL latency results of the analytical model and the Pareto optimized data points. Firstly, the description of the simulation setup is presented. This is followed by comparing the results of the analytical model and the simulation. Thereafter, these results are analyzed in detail based on the energy consumption and DL latency of a device. Finally, the optimized parameters of the power-saving schemes obtained using Pareto front analysis are discussed.

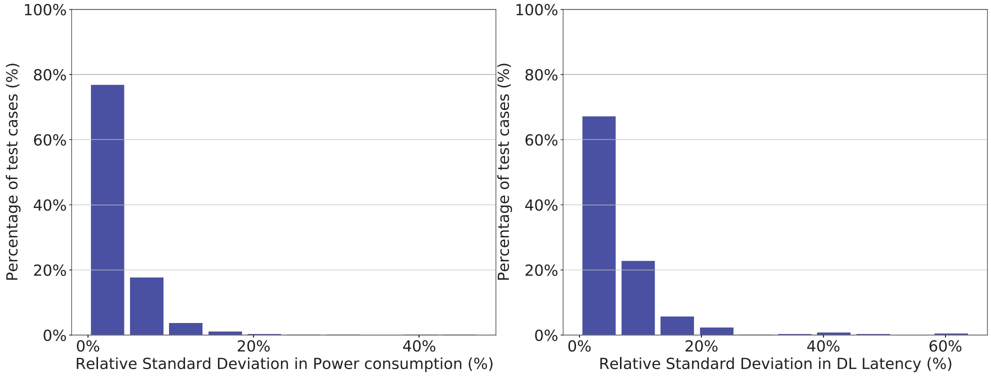
Table 7.3: Simulation and Model parameters

Parameters	**Symbol**	**Value**
Operating voltage	V_{op}	3.8
PSM state power consumption	E_{PSM}	$0.000003 \times V_{op} = 0.0000114\ W$
eDRX state power consumption	E_{eDRX}	$0.0008 \times V_{op} = 0.00304\ W$
Connected state power consumption	E_C	$0.006 \times V_{op} = 0.0228\ W$
Rx power consumption	E^C_{DL}	$0.023 \times V_{op} \times T_p = 0.0874 \times T_p\ J$
Tx power consumption	E^C_{UL}	$0.275 \times V_{op} \times T_p = 1.048 \times T_p\ J$
Paging state energy consumption per RB	E_P	$0.023 \times V_{op} \times T_{cch} = 0.0874\ \times T_{cch}\ J$
Tx time on control channel per RB	T_{cch}	0.214285 *ms*
Tx time on data channel per RB	T_{dch}	0.92857 *ms*
Modulation scheme	MCS	9
Application data size	N^{DL}_{data}	64 bytes
Number of DL/UL packet segments	N^{DL}_{seg} or N^{UL}_{seg}	7
Packet transmission time	T_p	$T_{dch} \times N^{DL}_{seg} = 6.499$ ms
RRC inactivity timer	T_{RRC}	[1,5,10,20,30,40,50,60] s
eDRX state/Active timer	T_{eDRX}	[0, 20.48, 40,96, 81.92, 163,84, 327.68, 655.36, 1310.72, 2621.44, 5242.88, 10485.76] s
eDRX cycle timer	T_{cycle}	[0, 20.48, 40,96, 81.92, 163,84, 327.68, 655.36, 1310.72, 2621.44, 5242.88, 10485.76] s
PSM timer	T_{PSM}	[0, 136.219, 272.438, 544.877, 1089.755, 2179.511, 4359.023, 8718.046, 17436.093, 34872.187] s
Number of eDRX cycles	η	[1,2,4,8,16,32]

7.2.1 Simulation Setup

The ns-3 simulator is used to analyze the latency and power consumption of an NB-IoT UE using eDRX and PSM. The ns-3 simulator is one of the most popular computer network simulators, where some NB-IoT features have been implemented on top of the LTE code [124]. The code defines several new features, such as limiting the number of LTE physical resource blocks to one in the frequency domain, modifying the physical error model to adopt lower MCS, separating the SFs for control and data channels, and including cross SF delays for both channels. The RRC idle mode features (PSM and eDRX), energy calculation modules, and scripts to calculate latency are implemented. The complete description of the code implementation is described in another paper [98].

The simulation results under various parameter values are compared with the analytical model to evaluate its accuracy. The analytical model is solved using MATLAB. All of the simulation parameters are shown in Table 7.3, which are selected according to the NB-IoT specification. The reference power values are taken from the u-blox SARA-N3 NB-IoT radio module datasheet [125]. As mentioned in the datasheet, the current consumption in the PSM state of the SARA module is 3 μA at an operating voltage of 3.8 V. The power consumption of this state is calculated as 0.000003 × 3.8 that is 0.0114 mW. Similarly, the power consumption values of the other states are calculated and mentioned in Table 7.3. The transmission time on the control and data channel is evaluated from the ns-3 results. For simplicity, the experiments are performed for a single paging occasion using one device and one eNB. Many IoT use cases have data intervals ranging from weeks to months; therefore, multiple paging occasions in each eDRX cycle waste the energy in monitoring them. Furthermore, the repetition count only increases the duration of the RRC connected state. Therefore, to focus the effect of the PSM and eDRX feature on the DL latency and power consumption, the repetition of control and data channels is ignored. However, it



(a) Power consumption

(b) DL latency

Figure 7.1: Histogram of percentage relative standard deviation between Model and Simulation results

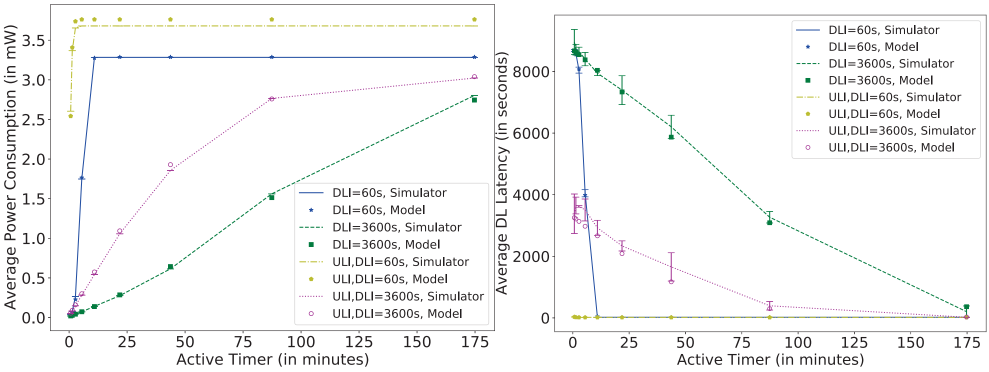
should be noted that the ns-3 mathematical model supports an arbitrary number of devices, paging occasions per eDRX cycle and repetitions.

7.2.2 Model Validation

A total of 1826 test cases are run to compare the result of the analytical model and simulation. The test cases are created from the parameters mentioned in Table 7.3. These test cases analyze only DL data as well as a combination of UL and DL data. The data interval of 1 minute and 60 minutes are considered. The histograms shown in Figure 7.1 present the relative standard deviation in power consumption and DL latency between the simulation and the model results. There is a relative standard deviation of 3.16% in power consumption and 6.37% in DL latency found in a run of all the test cases. The mean difference is around 4.45% and 8.73% in power consumption and DL latency, respectively. The relative difference in power consumption value is high in the cases where there is a high probability of a UE to remain in the PSM state. Around 0.49% of test cases have a deviation in power consumption value of more than 20%. But most of these cases have very small (less than 1 mW) absolute power consumption value. Therefore, a small value difference results in a large relative deviation.

The large relative deviation in latency value ($\geq$ 40%) was also observed for large values of the PSM timer with frequent DL data, but T_{eDRX} (Active timer) is 655.36 s. In 4.3% of test cases, there is a relative standard deviation above 20%. Mostly these cases have the eDRX cycle time of 40.96 s with large PSM and Active timer values, having DL data interval (DLI) or/and UL data interval (ULI) of 60 minutes. There are no test cases where the relative standard deviation for both (the latency and the power consumption) is higher than 15%.

A clear observation is that the deviation in the results (for both energy consumption and DL latency) of the model and simulation is higher, mostly when the PSM timer is large in



(a) Power consumption

(b) DL latency

Figure 7.2: Comparing Simulator and Model for different data intervals varying Active timer (T_{3324}) for parameters: RRC inactivity timer (T_{RRC}) = 1 s, eDRX cycle (T_{cycle}) = 40.96 s, and PSM timer (T_{3412}) = 4.84 hours

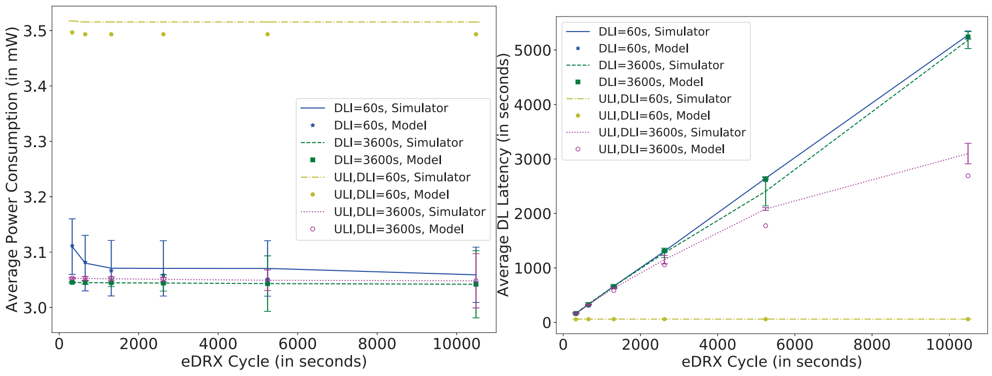
combination with certain other timer values. Also, the accuracy of the model is higher in terms of power consumption than in terms of DL latency.

Moreover, a different set of simulation results are also collected for five different seed values and ranges to generate various Poisson distributions in data transmission timings. The standard deviation observed in the simulation results is 0.18% and 4.72% for the power consumption and the DL latency, respectively. These observed variations improve the relative standard deviation between the test results of simulation and the analytical model.

7.2.3 Result Analysis

Figures 7.2 to 7.5 plot the results of the analytical model and the ns-3 simulation, for various transmission rates. It can be visibly seen that the analytical model and simulation follow the same behaviour, and the results match closely in all the plotted graphs. It can be observed that as the DLI increases, the power consumption decreases since fewer packets are being transferred per unit time. Moreover, the addition of UL data transmissions to DL data transmissions further increases the power consumption and decreases the DL latency because the UE gets an early opportunity to transition to the RRC connected state. These figures denote the effect on power consumption and DL data latency by varying different timers. The effect of the Active timer and eDRX cycle timer is analyzed in Figure 7.2 and 7.3 respectively. These parameters are important when IoT use case expects infrequent DL data. Whereas, when DL data is sent frequently, the RRC inactivity timer influences the DL latency. Its effect is shown in Figure 7.4. Finally, the PSM timer plays an important role in saving maximum energy, keeping the device in a DEEP SLEEP state. Figure 7.5 shows the variation of the PSM timer and its effect on both power consumption and DL latency.

In Figure 7.2, the Active timer is varied from 20.48 seconds to 2.91 hours. Since, in the RRC Idle state, the UE enters into eDRX state before PSM state, the PSM timer (T_{3412}) needs to be configured to a higher value than the Active timer (T_{3324}). Therefore, the PSM



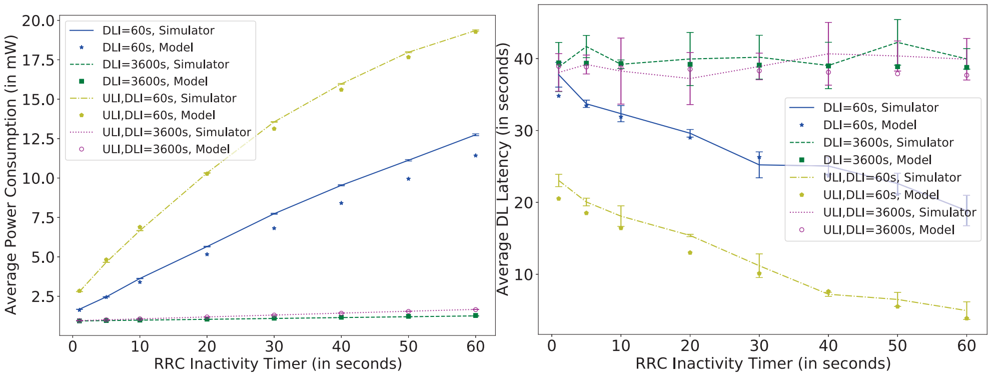
(a) Power consumption

(b) DL latency

Figure 7.3: Comparing Simulator and Model for different data intervals varying eDRX cycle timer (T_{cycle}) for parameters: $T_{3324} = 10485.76$ s, $T_{RRC} = 1$ s, and PSM Disabled

timer is configured to 4.84 hours while other timers are configured with small values. It can be observed from Figure 7.2a that the Active timer mostly impacts the power consumption linearly, except for frequent transmission intervals where large values of the Active timer have nearly no effect on power consumption and DL latency. This is because as the Active timer increases, the UE gets more time to remain in the eDRX state and do paging. During the paging (at every eDRX cycle time), the UE gets an opportunity to receive the incoming DL data keeping the DL latency low. It is observed from Figure 7.2b, the minimum latency is 20.24 s when only DL data is transferred (blue line) and 16.22 s for a combination of both DL and UL (yellow line). Also, the maximum latency is around 2.36 hours for a low value of the Active timer as in this case, the probability that the UE remains in the PSM state is high, which dominates the latency value (which is around half of the PSM timer). Therefore, use cases with infrequent UL traffic and that require DL latencies in the order of only tens of seconds should be configured with a long Active timer and short PSM deep-sleep time.

The graph plotted in Figure 7.3 shows the variation in DL data latency and power consumption for different eDRX cycle timer values by fixing the Active timer and RRC inactivity timer. The PSM feature is disabled. A large value of the Active timer is chosen to increase the probability of the UE to be in eDRX state so that the effect of the connected state is minimized. The interesting observation is that the power consumption does not vary much, but the latency increases linearly by increasing the eDRX cycle when only DL data is sent. The reason is that the UE sleeps, consuming low energy until the next paging occasion, and this paging occasion time affects the DL data latency. A large value of the eDRX cycle timer means that the paging occurrence is delayed, thereby increasing the latency without affecting the power consumption substantially. The variations in power consumption are observed due to the transition times from RRC connected to Idle state. When UL data is sent with DL data, the UE switches to the connected mode for sending UL data without waiting for the paging occasion. Therefore, the eDRX cycle timer has nearly no effect on DL latency when the ULI is close to the eDRX cycle time (as seen for the yellow line where ULI = DLI = 60s). The DL latency is nearly half of the eDRX cycle value. Considering the above observations, it can be concluded that for IoT use cases that require low latency, a



(a) Power consumption

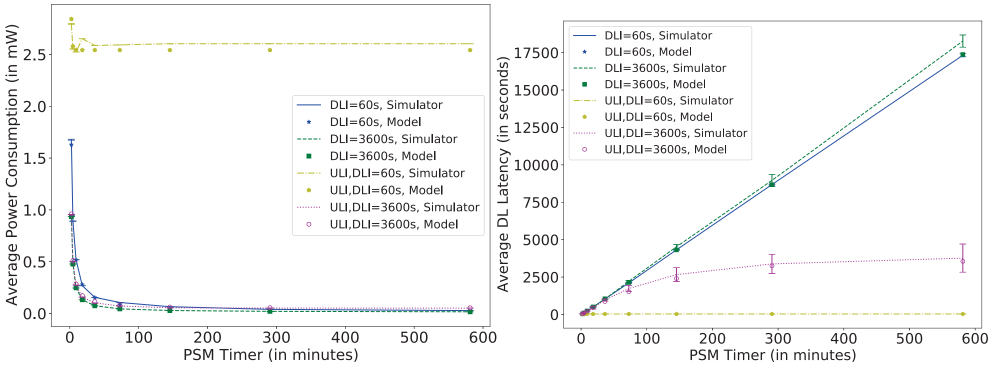
(b) DL latency

Figure 7.4: Comparing Simulator and Model for different data intervals varying RRC inactivity timer (T_{RRC}) for parameters: $T_{3324} = T_{cycle} = 40.96$ s, $T_{3412} = 136.219$ s

low value of the eDRX cycle value should be suggested, especially if the UL transmission interval is high. In Figure 7.3a, the standard deviation in simulation result for DL interval of 60s is visibly seems large, but the average deviation is only 0.47%.

A variation of the RRC inactivity timer is shown in Figure 7.4 with a small fixed value of the Active timer, eDRX cycle time, and PSM timer. Figure 7.4a shows that as the RRC inactivity timer increases, the power consumption increases because after each data transmission, the UE needs to stay in RRC connected state until the RRC inactivity timer expires. The RRC inactivity timer seems to have less effect on the power consumption for higher data interval scenarios because the UE spends more time in the RRC Idle state than in connected. It has an opposite effect on DL latency, i.e., it decreases with an increase in RRC inactivity timer (cf. Figure 7.4b). By analyzing the blue lines of Figure 7.4, it is observed that when DLI is 1 minute, the RRC inactivity timer affects the power consumption more than the DL latency. When the RRC inactivity timer increases from 1 second to 1 minute, it is observed that there is around 278% increase in power consumption and 50% decrease in DL latency. In Figure 7.4b, there is visible variation in the model and simulation results; it is not more than 10% that is when DLI and ULI are 60 s (yellow line) and sometimes this reduces when compared with the results of the simulator executed with different seed values. It can be concluded that the RRC inactivity timer should be configured carefully by the network operator as it augments the power consumption with comparatively little improvement on DL latency. This effect is most noticeable when there are frequent UL transmissions.

Finally, the test cases where the UE is configured such that it remains in the PSM state as much as possible are discussed. Figure 7.5 shows the power consumption and DL latency variation as a function of the PSM timer with small values of Active timer, eDRX cycle timer and RRC inactivity timer. It can be seen that the power consumption decreases exponentially with an increase in PSM timer. Whereas, if there is only downlink it is linear following the PSM timer, while if there is UL transmission, the latency is bounded by the UL interval. This occurs because the scheduled UL data stops the RRC idle state, and therefore, the PSM timer has minimal effect on latency and power consumption. The eDRX cycle timer



(a) Power consumption

(b) DL latency

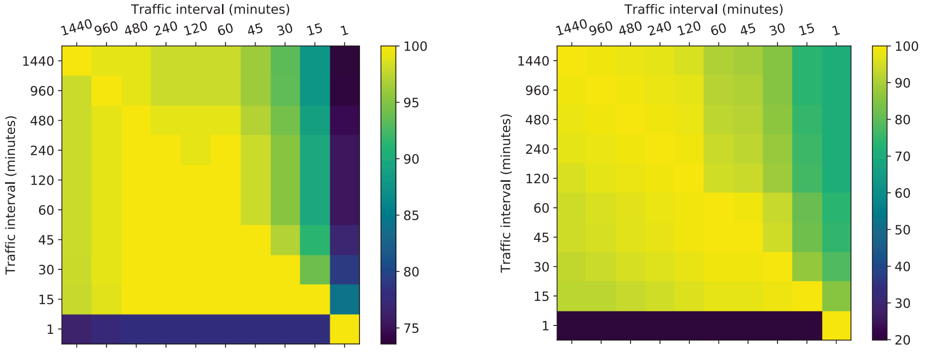
Figure 7.5: Comparing Simulator and model for different data intervals varying PSM timer (T_{3412}) for parameters: $T_{RRC} = 1$ s, $T_{3324} = T_{cycle} = 40.96$ s

and PSM timer follow a similar trend in the variation of DL latency. The DL latency is nearly half the PSM timer in a DL only scenario. It also shows that a large value of the PSM timer (>4.84 hours) does not substantially affect the power consumption since most of the DL packets are waiting in a queue for the expiration of the PSM timer. However, as some of the packets are waiting, the DL latency increases with the PSM timer. Therefore, when the DLI is infrequent, the PSM timer should be configured based on the maximum required latency of an application.

Analyzing the graphs of different timers, it can be concluded that the IoT use cases with frequent DL data requirements should consider small eDRX cycle timer values. This is because, with the increase in this timer, the DL latency increases with nearly no effect on power consumption. However, it is needed to consider the trade-off between power consumption and latency when selecting Active timer, PSM timer, and RRC inactivity timer. If the IoT use case has frequent UL data, small values of the eDRX cycle and RRC inactivity timer should be selected, and the optimal values of Active timer and PSM timer need to be calculated depending on the UL data frequency.

7.2.4 PSM and eDRX Parameter Optimization

The problem of obtaining optimal parameter values for different use cases is discussed here. A use case in terms of its UL and DL transmission rate is defined. It involves two objective functions as DL latency, and power consumption needs to be minimized. This objective using multi-objective optimization is solved using Python by determining the Pareto front to evaluate the latency and power consumption metrics for different values of all the timer combinations. We plot an approximate Pareto front obtained by solving the proposed Markov Model for 6560 test cases for the RRC inactivity timer (T_{RRC}) ranging from 1 to 60 s, Active timer (T_{3324}) and eDRX cycle from 0 to 10485.76 s, and PSM timer (T_{3412}) from 0 to 35709120 s. All 6560 test cases were run for different DLI and ULI (1 minute to 24 hours).



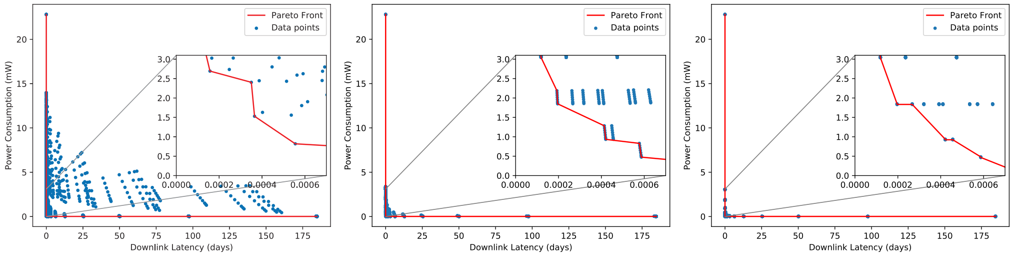
(a) DL only

(b) DL and UL

Figure 7.6: HeatMap of Pareto front matching percentage for different traffic intervals

Table 7.4: Number of Pareto front data points

Data Interval (#Minutes)	For DL only	For DL & UL
1440	106	96
960	107	96
480	107	96
240	106	96
120	106	96
60	106	92
45	104	91
30	101	87
15	95	77
1	102	332

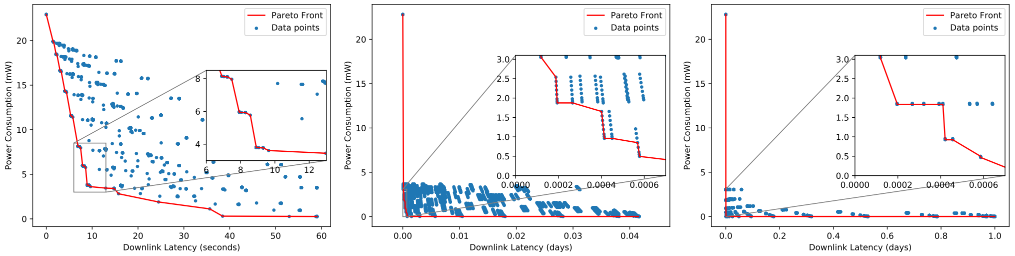


(a) DL 1 minute

(b) DL 60 minutes

(c) DL 1440 minutes

Figure 7.7: Power consumption and DL latency for all 6560 test cases, with the Pareto front marked by a line, for DL only scenarios



(a) UL & DL 1 minute

(b) UL & DL 60 minutes

(c) UL & DL 1440 minutes

Figure 7.8: Power consumption and DL latency for all 6560 test cases, with the Pareto front marked by a line, for DL and UL scenarios

Table 7.4 lists the number of Pareto optimal data points among the 6560 test cases for each transmission interval (DL only and UL-DL). It can be observed that the number of Pareto front elements decreases as the data interval decreases except in the case of a 1 minute interval. Moreover, the fronts obtained at lower data intervals are mostly the subset of the previous test case (having a larger data interval), which can be observed from the heatmap shown in Figure 7.6. The heatmap indicates the number of matching Pareto optimal data points represented as colours ranging from yellow to blue. Yellow entities match heavily, while blue ones overlap little or nothing. As such, in the DL only scenario, except the 1 and 60 minutes data intervals, all other intervals are a subset of its larger data intervals. However, in the DL and UL scenarios, only 1 minute shows a disparity in the behaviour.

Figures 7.7 and 7.8 show the results in terms of power consumption and DL latency and plot the Pareto front for 6 traffic scenarios. The points on the red line are the Pareto front, whereas the bold blue dots represent the points for all the test cases. The goal is to search the points close to the bottom left (i.e., the elbow), which means minimal DL latency consuming minimal energy. The plots show that the minimal DL latency is achieved when PSM and eDRX are disabled spending high power (22.8 mW) because the UE is always in RRC connected state (top left point on each figure). The minimum power is consumed when only PSM is enabled by disabling the eDRX feature, compromising latency optimization (bottom right). The pattern of the plots are similar for all cases (15 to 1440 minutes) but not when the data interval is 1 minute. The RRC inactivity timer impacts the power consumption more than DL latency, and this impact decreases with the increase in data frequency. Therefore, it can be visibly seen that the number of points present on the red line increases when the data interval increases from 1 to 60 minutes and later starts decreasing (which can be seen in Figure 7.7 or 7.8). It is due to the reduced effect of the RRC inactivity timer under more sporadic traffic (i.e., 1 hour and more) that the blue dots edge closer and closer to the red line of the Pareto front.

It is observed that most of the Pareto front data points obtained using a short eDRX cycle time of 20.48 seconds. Some relevant optimal configuration values are listed in Table 7.5. It can be observed that the pattern of optimal configurations is similar, and can be pointed out that the bottom left cases have the smallest values of all the timers. For each type of traffic frequency, a Pareto optimal configuration is marked with a good latency-energy trade-off (i.e., near the bottom left of the graph) with a grey background. This grey coloured configuration allows reducing the energy consumption to 4-10% of the max (i.e., from more than 22.8 down to less than 3 mW) while keeping the latency below 20 seconds. The change in data interval, from 60 to 1440 minutes, does not affect the latency and power consumption much in comparison to changing it from 1 to 60 minutes. This change in data interval from 1 to 60 minutes influences the latency and power consumption, with more than 55% when both DL and UL data is sent and more than 25% when only DL data is sent. Some optimal configurations have no effect on power consumption and latency when the Active timer range is between 20.48 and 655.36 s, provided the PSM feature is disabled. However, many configurations affect them by more than 50% as observed when the Active timer is reduced from 81.92 s to 40.96 s in case A, C or F. In the same way, the RRC inactivity timer affects them, but its effect decreases with an increase in data interval as noticed from case A and B. When the RRC inactivity timer is reduced from 60 to 20 seconds, the power consumption reduces and DL latency increases by more than 70% when DLI is 1 minute, but in case the data rate is increased to 60 minutes (case B), this effect is reduced to around 1%.

7.3 Conclusion

An analytical model is defined to calculate the power consumption and DL latency for NB-IoT with the PSM and eDRX power saving features. The model's accuracy was compared to ns-3 simulation results for various NB-IoT timers (i.e., Active timer, eDRX cycle timer, PSM timer, and RRC inactivity timer). The results derived from these two methods showed an average deviation of 6.33% in the results of power consumption and 8.73% in DL latency. The goal was to calculate the optimal configurations to obtain minimal power consumption and minimal DL latency. The graphs of different timers showed diverse behaviour on power consumption and DL latency for data frequency and transmission direction. Therefore, to analyze the trade-off between both objectives, Pareto front analysis is used. This provides the optimal values of NB-IoT timers for LPWA use cases. It was found that most optimal configurations had small timer values for various data intervals. Moreover, the impact of the RRC inactivity timer on power consumption and DL latency for optimal configurations decreases with increasing data frequency. For the LPWA applications where lower DL latency

Table 7.5: A subset of optimal Pareto front configurations

Test Case	**RRC inact. (s)**	**Active timer (s)**	**PSM Timer (s)**	**eDRX cycle (s)**	**DL Latency (s)**	**Power cn. (mW)**
A (DL 1 minute)	60	[20.48 - 655.36]	Disabled	20.48	4.59	13.96
	20	[20.48 - 655.36]	Disabled	20.48	7.89	7.58
	1	[20.48 - 655.36]	Disabled	20.48	10.1	3.32
	1	81.92	136.219	20.48	13.58	2.69
	1	40.96	136.219	20.48	31.54	1.52
	1	Disabled	136.219	Disabled	67.66	0.17
	1	Disabled	1089.755	Disabled	544.38	0.044
B (DL 60 minutes)	60	[20.48 - 655.36]	Disabled	20.48	10.08	3.36
	20	[20.48 - 655.36]	Disabled	20.48	10.19	3.15
	1	[20.48 - 655.36]	Disabled	20.48	10.24	3.05
	1	81.92	136.219	20.48	16.92	1.85
	1	20.48	136.219	20.48	50.66	0.48
	1	Disabled	136.219	Disabled	68.09	0.0178
C (DL 1440 minutes)	60	[20.48 - 655.36]	Disabled	20.48	10.239	3.054
	20	[20.48 - 655.36]	Disabled	20.48	10.244	3.045
	1	[20.48 - 655.36]	Disabled	20.48	10.246	3.041
	1	81.92	136.219	20.48	16.98	1.83
	1	40.96	136.219	20.48	36.38	0.92
	1	20.48	136.219	20.48	50.71	0.46
D (DL & UL 1 minute)	60	655.36	[1089.755 - 35709120]	20.48	1.48	19.87
	60	[20.48 - 655.36]	Disabled	20.48	1.48	19.87
	20	655.36	[1089.755 - 35709120]	20.48	5.33	11.56
	20	[20.48 - 655.36]	Disabled	20.48	5.33	11.56
	1	[20.48 - 655.36]	Disabled	20.48	8.91	3.80
	1	655.36	[1089.755 - 35709120]	20.48	8.91	3.80
	1	81.92	136.219	20.48	9.63	3.61
	1	20.48	136.219	20.48	24.48	1.89
	1	Disabled	1089.755	Disabled	59.01	0.26
E (DL & UL 60 minutes)	60	[20.48 - 655.36]	Disabled	20.48	9.89	3.68
	1	[20.48 - 655.36]	Disabled	20.48	10.22	3.05
	1	81.92	136.219	20.48	16.75	1.87
	1	20.48	136.219	20.48	49.93	0.49
	1	Disabled	136.219	Disabled	67.22	0.0179
F (DL & UL 1440 minutes)	60	[20.48 - 655.36]	Disabled	20.48	10.23	3.06
	1	[20.48 - 655.36]	Disabled	20.48	10.24	3.04
	1	81.92	136.219	20.48	16.97	1.83
	10	40.96	136.219	20.48	36.36	0.92
	1	Disabled	136.219	Disabled	68.07	0.01167
	1	Disabled	139488.75	Disabled	45671.45	0.01162

is preferable, a small value of the eDRX cycle and PSM timer should be considered. Whereas, for high UL frequency use cases, values of eDRX cycle and RRC inactivity timer should be small, as in these scenarios, increasing the timers has a little positive effect on DL latency, but does have a strong negative effect on energy consumption. Finally, if UL traffic is infrequent, the PSM timer should be set at most to double the maximum tolerable DL latency.

Chapter 8

Batteryless NB-IoT Prototype

The content of this chapter is based on:

- *A. K. Sultania, and J. Famaey, "Batteryless NB-IoT Prototype for Bidirectional Communication Powered by Ambient Light," in Ad Hoc Networks, 2022. [Impact Factor: 4.111]*

This chapter describes the batteryless prototype of the long-range NB-IoT technology. The NB-IoT UE is configured based on the optimal configuration derived in Chapter 7. The transmission power required for NB-IoT is higher compared to the short-range BLE. Therefore, the requirements of the capacitor and harvesting power are different compared to the batteryless BLE solution. We use the Nordic Semiconductor chip for the NB-IoT also.

8.1 Hardware Prototype

Nordic Semiconductor provides the nRF9160 development kit board, but it is pre-certified and they are in the process of optimizing its performance. Also, it is difficult to disable superfluous peripherals on the Nordic DK board. These peripherals consumes power in the range of mW which is undesirable for a batteryless device. Therefore, an in-house designed module based on the Nordic nRF9160 chip is used instead, as shown in Figure 8.1. It has a few interfaces such as a J-link probe (to flash the application), a serial port (to communicate externally mainly for debugging purposes) and an ISP adaptor which is leveraged to power the module.

Figure 8.2 shows the prototype used for the batteryless NB-IoT UE. The first piece of equipment is the solar panel that harvests voltage and current on light exposure. A monocrystalline solar panel is selected with a maximum of 400mW power [126], considering the maximum permitted power and voltage of the power management board (PMB), Epeas-AEM10941 [63] (the green rectangular board) that can handle input power up to 550mW and a maximum voltage up to 5 V. The Epeas board charges the capacitor and powers our nRF9160 board using another in-house developed Octa board (green octagonal-shaped board), which acts as a serial port connector to distribute power.

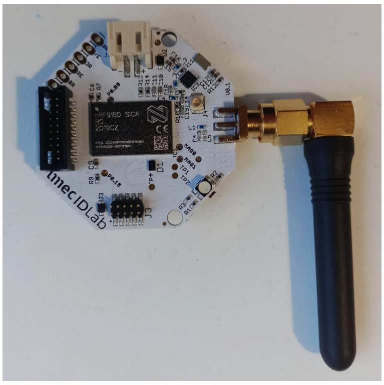


Figure 8.1: NB-IoT customized module

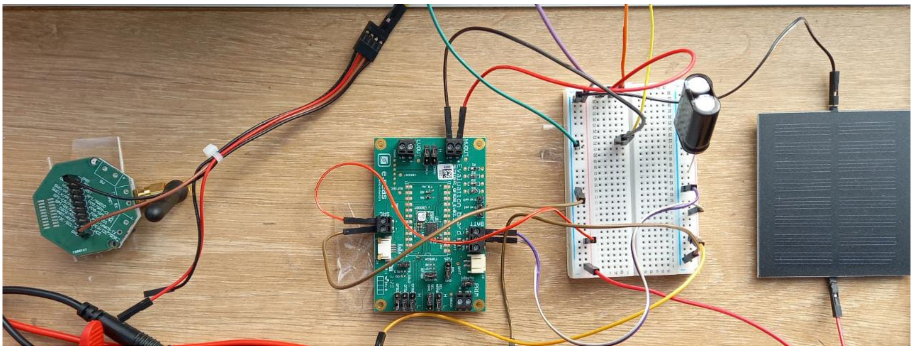


Figure 8.2: Batteryless NB-IoT prototype setup

We use an advanced PMB, Epeas-AEM10941 that equalizes the supply voltage, and only provides the device with power if the capacitor voltage is within certain minimum and maximum bounds. Whenever energy is harvested, the PMB is used to charge the connected capacitor to a maximum voltage point (V_{max}). The stored energy is then converted to a stable voltage to operate an NB-IoT module. The AEM10941 can be configured to supply different stable voltages (1.2, 1.8, 2.5 or 3.3V). The Nordic nRF9160 radio requires 3.3V [127]; therefore, using the default configuration, the Epeas board is configured such that the voltage thresholds as maximum, turn-on and turn-off voltages are set as 4.12V, 4.04V, and 3.6V, respectively. Once the capacitor charges up to the turn-on voltage (V_{turnon}), the device is powered by the supply voltage. Whereas, it is powered off when the capacitor voltage reaches turn-off voltage ($V_{turnoff}$). We chose the configuration which provides a maximum difference between turn-on and turn-off voltage because this provides the NB-IoT device to maximally use the energy available in the capacitor.

8.2 Software Implementation

The Nordic nRF9160 application is developed using the nrf Connect SDK which is based on the Zephyr real-time operating system. The Nordic UDP sample application is modified to support both unidirectional and bi-directional communication. An external server is configured to always listen to a UDP port. For bi-directional communication, the server sends a data packet on each data reception (from an NB-IoT device).

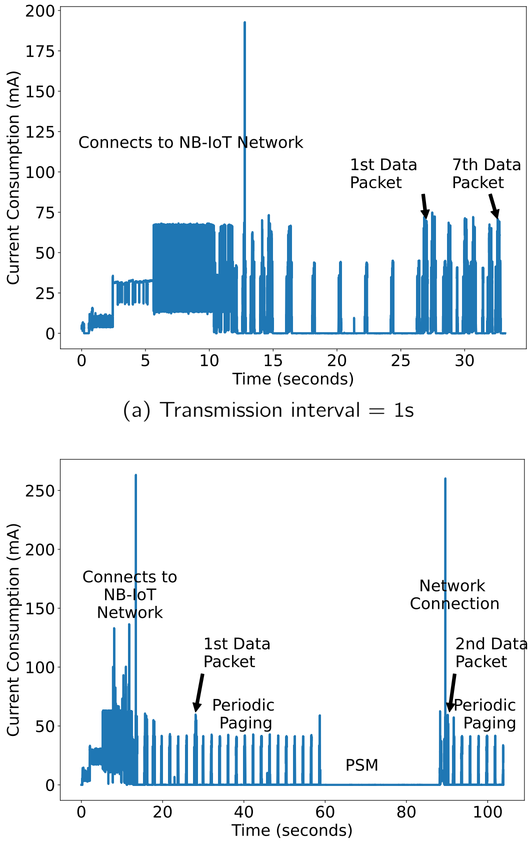
On reboot, the NB-IoT device first searches and attaches to the network. It then alternates between two states: RRC connected state and RRC idle state. In the RRC connected state, the NB-IoT device transfers data over a UDP socket to the server and in the RRC idle state, it saves power either by enabling the eDRX or PSM feature.

8.3 Experiments

In this section, we provide the overview of power consumption of the NB-IoT module in different radio states. The experimental setup and its parameters are also described. Finally, the results of unidirectional and bi-directional communicating batteryless NB-IoT UE are presented for both the artificial and natural light.

8.3.1 Power Consumption

The current profile of the device and its associated energy consumption are recorded using the off-the-shelf Nordic-Power Profiler Kit-II (PPK-II), which has a resolution of 0.1 μA [71]. We use the PPK-II connected with the NB-IoT module in source meter mode to measure the current consumption of each device state. The analysis is performed while the device is connected to the Belgian Orange NB-IoT network. All power-saving related network parameters were tested, and their effect on power consumption was evaluated. During this study, it is found that the Orange network does not support TAU-related T_{3412} *extended*



(a) Transmission interval = 1s

(b) Transmission interval = 60s

Figure 8.3: Current consumption of the UE as measured from PPK-II sending only uplink data

lower than 250 minutes, nor could an RRC inactivity time lower than 30s be configured. Therefore, considering the RRC inactivity timer of 30s, the best optimal configuration based on Section 7.2.4 for both the considered use-cases, uni- and bi-directional communication (when low power consumption is important) is to disable the eDRX timers (T_{3324}) and enable only the PSM timer (T_{3412} *extended*). Therefore, considering the network constraint, the UE is set to have T_{3412} *extended* equal to 250 minutes to switch to the deep-sleep state. However, it can try to switch to the RRC connected state to send a UL packet whenever they are scheduled by the application. Figure 8.3 shows the current consumption and the state changes of the UE. It can be observed that the UE takes around 26s to connect to the NB-IoT network, consuming the maximum amount of current. The average current consumption for associating with the network is 17 mA. After this step, the UE starts communication by consuming 27mA current for 463 ms. The RRC inactivity timer for the Orange network as observed from Figure 8.3b is 31s, during which the UE sends periodic paging in an interval of 2s. During this time, the UE consumes 1.45mA current on average. At a TI of 1s (Figure 8.3a), the UE never gets an opportunity to move to the PSM state and consumes around 28mA during 400ms to do bidirectional communication. Whereas, when the TI is 60s, it can switch to PSM state, which consumes only 6.93 μA.

8.3.2 Experimental Setup

In order to evaluate if indoor light harvesting can sufficiently power the NB-IoT device activities such as network acquisition, and message exchange, remain in RRC connected state or RRC Idle state, preliminary measurements to calculate the harvested power are performed. The Epeas board harvests power (P_{mpp}) at maximum power point (MPP), which can be calculated by measuring MPP voltage (V_{mpp}) and MPP current (I_{mpp}). P_{mpp} is computed from the solar panel directly facing a Philips smart bulb A67 (2700K) placed in a closed dark box. The intensity of the light is changed by the Android app provided by Philips. The voltage and current is measured using the digital multimeter (Keysight-U1281A) while device is in operation.

Two practical use-cases for NB-IoT UE are set up as follows:

- *Uni-directional UL only*: The UE is expected only to send data to the server at a fixed interval.
- *Bi-directional communication*: The UE is expected to receive data from the server, as upon sending a data packet, it also receives back a DL packet.

Each data packet is considered to be unsegmented with a fixed payload length of 8 bytes. Also, after the data communication, the UE returns to PSM mode consuming a minimum amount of energy (6.2 μA) after the RRC inactivity timer of 30s. The connected state significantly impacts the overall power consumption of any device operating on the network as the UE needs to be active for 30 s consuming around 30 mA during this period.

The following performance metrics are considered to evaluate the batteryless UE in an NB-IoT network:

- *Average Data Inter-arrival Time (IAT)*: The average time between the arrival of two consecutive DL or UL packets received by the server or UE.
- *Lost Packet Percentage*: The percentage of packets that are lost during communication.
- *Restart Count*: The number of times the UE restarts.
- *Outage Time*: The time during which the UE is in the OFF state after the first network acquisition.

Table 8.1 shows all the experimental parameters. The selection of the lowest capacitor size and harvesting power is determined experimentally. As shown in Figure 8.3, the network acquisition consumes most of the power which is the bottleneck in choosing the lowest capacitor size and harvesting power. In the experiments to determine the lowest supported capacitor size and harvesting power, we evaluated capacitors between 100mF and 5F, and values for harvesting power between 500μW and 50mW. It is noticed that a capacitor size of 1.5F with a harvesting power of 4mW was the lowest that could support the acquisition phase. Figure 8.4 shows the behaviour of the capacitor voltage when harvesting power is

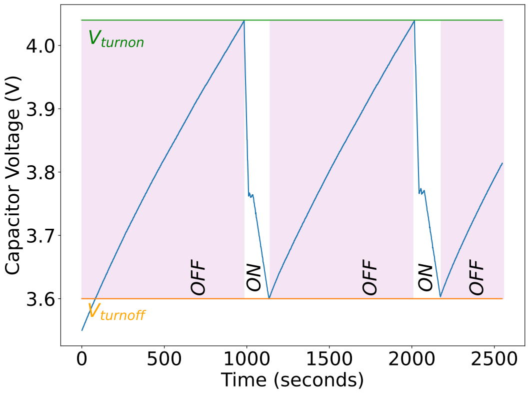


Figure 8.4: Variation in capacitor voltage of 2.5F harvesting at 6mW and the UE communicates unidirectional.

Table 8.1: Experimental parameters

Parameters	**Symbol**	**Value**
Operating voltage	V_{op}	3.3
Application data size	N_{data}	8 bytes
Transmission Interval	TI	{1, 60}s
Capacitor Size	CS	{1.5, 2.5, 3.5}F
Harvesting power	harvesting power	{4,6,8,18,30,50 }mW
RRC inactivity timer	T_{RRC}	30 s

6mW. It can be observed that the UE gets powered on when the capacitor reaches 4.04V, and its voltage decreases drastically for 26s (which is the network acquisition phase). Later, the UE starts communicating data packets until its voltage reaches the turnoff voltage. The time the UE remains in the ON state communicating data increases by increasing the capacitor size or the harvesting power or Transmission Interval (TI). Moreover, the same cycle continues repeatedly because the UE shuts down in RRC connected mode and on restart it needs to reacquire the network. Hence, a new IP address would be assigned. However, when the capacitor is less than 1.5F or harvesting power is less than 4mW, the ON state of the UE lasts less than 25s. Therefore, the UE restarts itself without communicating any data packet. Increasing the harvesting power, not only increases the ON state duration but also decreases the charging time.

8.3.3 Results

Once a batteryless device has started communicating, it will only move to the OFF state when its capacitor voltage goes below 3.6V ($V_{turnoff}$). Assuming that the device is continuously harvesting, it becomes active once the capacitor charges up to 4.04V (V_{turnon}). It means the time it takes to charge the capacitor from 3.6 to 4.04V contributes to the device outage time calculation. Figure 8.5 shows the charging time for different harvesting powers (4 to 30mW). It can be observed that the smaller the capacitor, the faster it recharges.

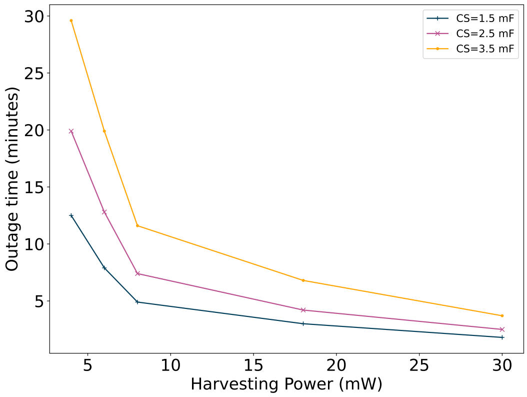


Figure 8.5: Capacitor charging time at different harvesting powers from 3.60 to 4.04V

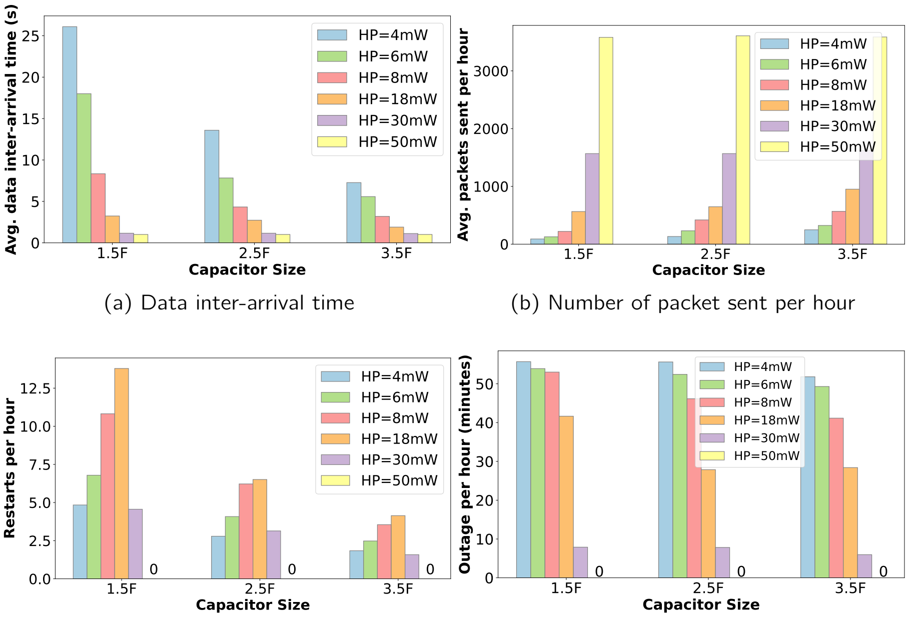
The capacitor of 1.5F at harvesting power of 4mW takes 12.5 minutes, whereas 3.5F takes 29.6 minutes. Moreover, as the harvesting power increases the charging time decreases exponentially.

8.3.3.1 Unidirectional UL only communication

Figure 8.6 shows the results for a TI of 1s. It can be observed that as the harvesting power increases, the IAT decreases and more packets are received by the server. By doubling the harvesting power from 4 to 8mW, it can be observed that the IAT is 3 times faster. The capacitor voltage of the UE never decreases when it harvests at least 50 mW. With an increase in harvesting power, the outage time decreases and therefore, the UE can send more packets (Figure 8.6d). However, as a consequence, it experiences more restarts per hour. Moreover, as the capacitor size increases, it can support the UE for more time to do communication. Therefore, with an increase in capacitor size, more packets are sent, and fewer restarts are observed. But the restart count also decreases at certain higher harvesting power (30mW in Figure 8.6c) because then it takes longer for a capacitor to discharge. With the growth in energy density with the increase in capacitor size, the outage time as shown in Figure 8.6d decreases as it can support UE to be in active state for long.

The results for TI of 60s are shown in Figure 8.7. Increasing the TI to 60s, it provides more time for the UE to recharge its capacitor between transmissions. It can be observed that at a harvesting power of 6mW, the capacitor voltage never lowers below the turn-off voltage as shown in Figure 8.8 for unidirectional communication. Moreover, the larger the capacitor, the better performance can be gained. The difference between 2.5F and 3.5F is smaller than compared to 1.5F. Compared to the result of TI 1s, it can be observed that the restarts per hour count for TI 60s decreases from 4.85 to 2.11 for 1.5F capacitor size and from 1.85 to 0.47 for 3.5F at harvesting power equal to 4mW.

However, at harvesting power equal to 4mW, when the TI is increased from 60s to 600s, the batteryless UE shows no restart count (not shown in the Figures). It means that this



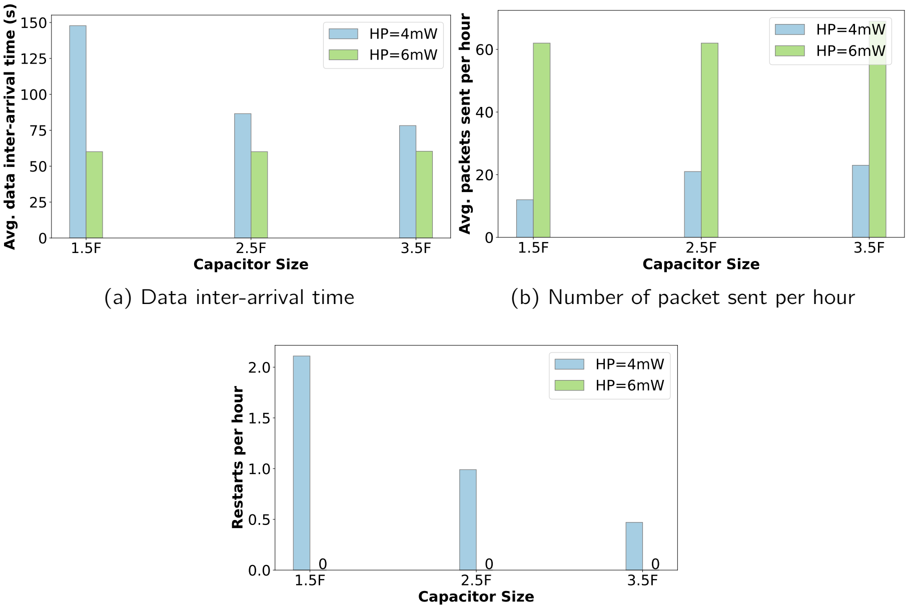
(a) Data inter-arrival time

(b) Number of packet sent per hour

(c) Number of restart per hour

(d) Outage per hour

Figure 8.6: Unidirectional communication at TI=1s



(a) Data inter-arrival time

(b) Number of packet sent per hour

(c) Number of restart per hour

Figure 8.7: Unidirectional communication at TI=60s

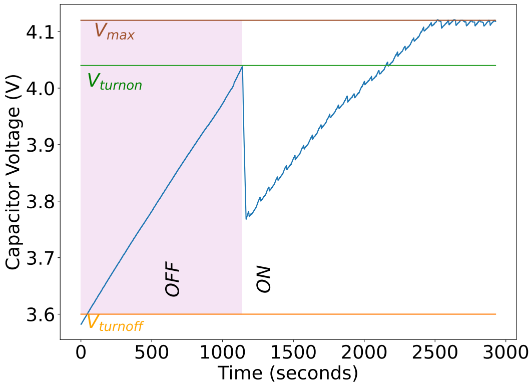


Figure 8.8: Variation in capacitor voltage of 3.5F harvesting at 6mW for Unidirectional communication TI=60s.

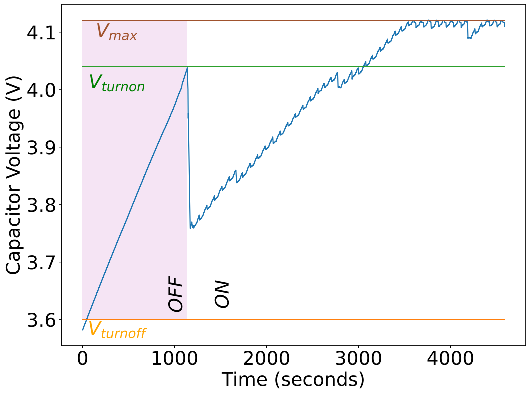
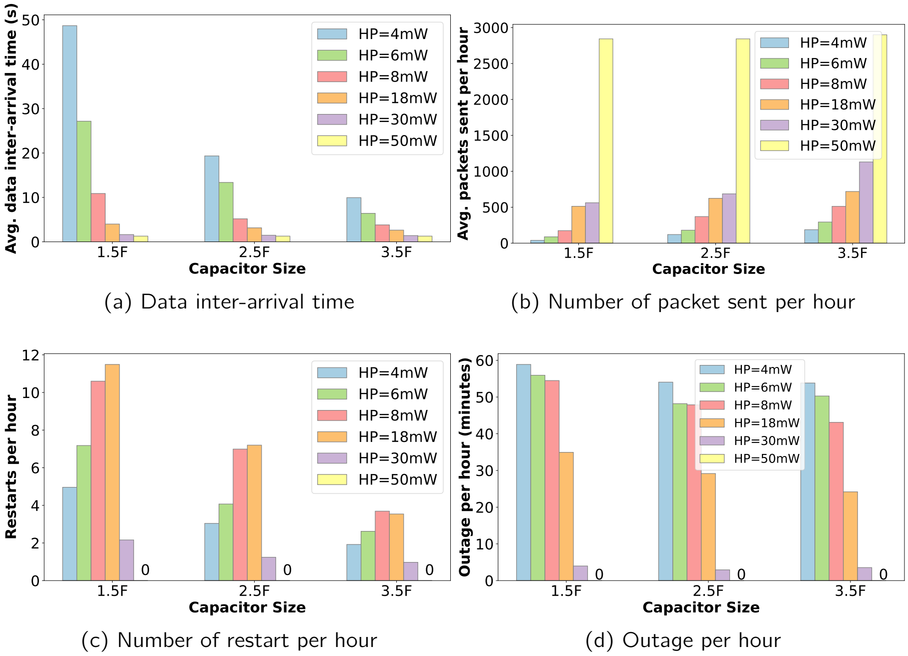


Figure 8.9: Variation in capacitor voltage of 3.5F harvesting at 6mW for Bidirectional communication TI=60s.

low harvesting power is enough to support the TI of 600s.

8.3.3.2 Bidirectional communication

As the capacitor voltage variation in bidirectional communication as shown in Figure 8.9 is similar to the unidirectional communication (Figure 8.8), the performance result for a TI of 1s for bidirectional communication as indicated in Figure 8.10 is also similar. Since, communicating bidirectional, the power consumption increases a bit compared to the unidirectional, the UE takes more time to maintain V_{max}. However, with increased power consumption, the IAT also increases compared to unidirectional communication. It can be observed that the IAT increases by 90% with 1.5F and by 25% with 3.5F for bidirectional communication at a TI of 1s compared to the unidirectional case. The UE needs to wait until it receives the DL packet after sending a UL packet. Therefore, the effective number of transmitted packets



(a) Data inter-arrival time

(b) Number of packet sent per hour

(c) Number of restart per hour

(d) Outage per hour

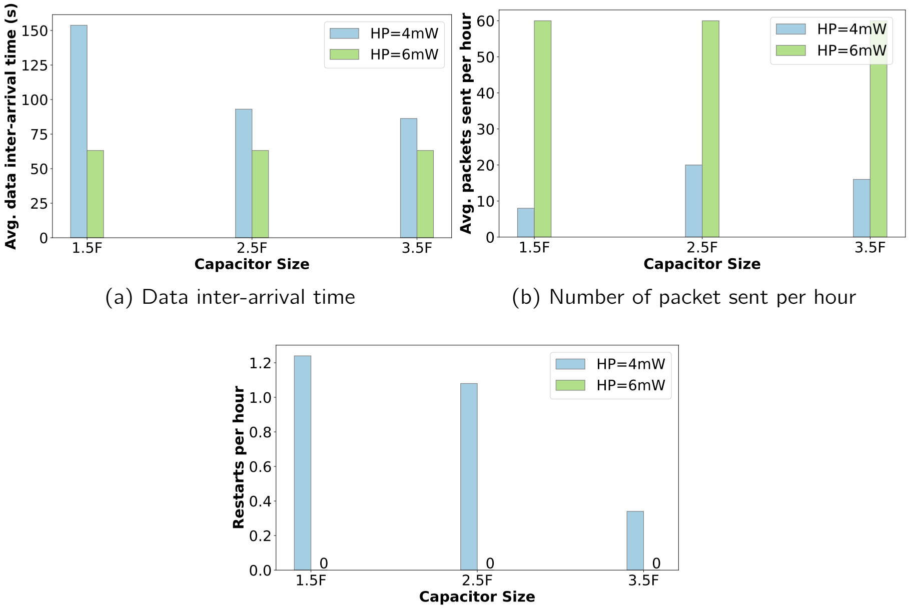
Figure 8.10: Bidirectional communication at TI=1s

per hour is lower in bidirectional communication. At 50mW, when there is no restart, it can be observed that the maximum number of packets sent is 2800 per hour (Figure 8.10b), whereas for unidirectional it is 3600 per hour (Fig 8.6). Therefore, the number of restarts per hour also decreases for all the capacitors compared to unidirectional communication. The outage time of the bi-directional communication increases for lower harvesting power (up to 8mW) and for higher harvesting powers it decreases when compared to the uni-directional results. This is because the higher harvesting powers can provide power to support DL communication such that the UE does not need to consume much from its capacitor. Moreover, it takes longer to perform one bi-directional packet transmission than uni-directional.

The result for a TI equal to 60s is shown in Figure 8.11, where it is also observed that the IAT increases compared to unidirectional for 4mW of harvesting power. However, the increase is less compared to a TI equal to 1s. With a 1.5F capacitor, the increase is only up to 8% and for 3.5F it is 4% compared to unidirectional communication.

8.3.3.3 Natural Light Experiments

The system is also tested using natural light to evaluate the performance of the system under dynamic harvesting power that changes over time. The batteryless UE and solar panel are placed on the west side window-sill with a 3.5F capacitor size in Antwerp, Belgium. The experiments are conducted in the fall season on cloudy and rainy days when the day length is around 12:45 hours [128]. The performance for both the DRs for bidirectional data communication is presented in Table 8.2. It is observed that at fast TI (1s) rain affects



(a) Data inter-arrival time

(b) Number of packet sent per hour

(c) Number of restart per hour

Figure 8.11: Bidirectional communication at TI=60s

the number of packets per hour by 48%, which is not the case when the TI is 60s. This is because, at 60s TI, the capacitor spends more time in the PSM state (after spending 30s in RRC connected state), while at 1s TI, the UE is always in RRC connected state, which requires much more power to remain active. It also means that during the rain, the power harvesting is lower than on a cloudy day. It is assumed that the summer months with sunny days show the best performance results that can easily provide a harvesting power higher than 30mW and therefore less inactive time. The number of restarts is also less when TI is 60s. IAT at 60s is around 83s for both cloudy and rainy weather, which is similar to the 4mW fixed harvesting power. However, more packets can be sent with natural lights compared to fixed harvesting power for both the DRs. The natural light during fall in Antwerp supports the UE to communicate for around 8 to 9 hours per day depending on external weather conditions.

Moreover, an energy-aware solution can maximize the energy efficiency and the operational lifetime of the UE by using harvesting energy during the day and use it for communication during the night. This can also reduce the number of restart count and thus save power in reacquiring the network.

8.4 Conclusion

In this paper, a working prototype of a batteryless NB-IoT UE is presented, which is powered by ambient indoor light. The UEs communicating uni- and bi-directional data are compared for different capacitor sizes and harvesting powers. It can be concluded that a TI higher than

Table 8.2: West-side Window-sill results using 3.5F capacitor for Bi-directional communication

Data rate	**Date and Weather**	**1st packet time**	**Last packet time**	**Total experiment time (minutes)**	**Data IAT including inactive time (s)**	**Data IAT (s)**	**Number of packets per hour**	**Number of restart count (s)**	**Average inactive time (minutes)**
1s	11 Oct 2021 (Cloudy)	09:26:15	18:04:33	518.3	15.27 ± 128.07	1.35 ± 0.38	235.94	30	15.75 ± 8.2
	12 Oct 2021 (Rainy after 13:00)	08:37:35	18:12:37	575.0	29.76 ± 245.8	1.42 ± 1.16	121.04	21	26.09 ± 16.46
60s	14 Oct 2021 (Cloudy)	09:33:42	17:32:04	478.4	83.9 ± 112.2	77.86 ± 5.38	43.01	1	35.85
	13 Oct 2021 (Rainy before 13:00)	09:22:40	17:53:29	510.8	82.6 ± 65.04	79.24 ± 4.77	43.69	1	22.14

60s can be easily supported in poor weather conditions or the availability of low intensity artificial light. However, the lowest capacitor size that supports NB-IoT acquisition on the evaluated Orange network is 1.5F. As the higher the harvesting power, the lower the time needed to recharge the capacitor. A harvesting power of 50mW is enough to keep the UE always on and communicating at a TI as low as 1s. At 50mW with 1.5 F, the UE can communicate 3580 packets per hour, whereas for bidirectional communication it is reduced to 2843 because it also needs to wait to receive DL packets. But as the harvesting power is reduced to 4mW, the average packet transfers per hour is reduced to 90 and 39 for unidirectional and bidirectional communication, respectively. Correspondingly, it affects the data IATs. Whereas, the number of restarts depends on power consumption and power harvested. Generally, when the harvesting power increases, the UE can perform more communication and thus loses more power. Therefore, the restart count increases with harvesting power, but at some higher optimal harvesting power point, the rate of decrease in capacitor voltage reduces and the UE can still communicate more without experiencing many restarts. At a TI of 1s for uni- and bidirectional communication, the tested optimal point is 30mW. Moreover, the solution also works for indoor natural light and the data IAT during the fall season is similar to the 4mW fixed harvesting power. It can be concluded that during the daytime and harvesting from natural lights, a batteryless NB-IoT UE can communicate data packets without being inactive if provided an opportunity to go into a PSM state.

Part IV

Conclusion

Chapter 9

Conclusion

Traditional sensor nodes are usually powered by batteries due to their cost and installation flexibility. Therefore, most research focuses on increasing the lifetime of these batteries to reduce the replacement cost. But the problem of batteries does not stop at the replacement but also their environmental pollution caused by battery production and disposal. Therefore, this thesis investigates if it is possible to completely replace the batteries with capacitors and charge them using ambient energy sources.

The thesis presents mathematical models to analyze the performance and help to optimally configure the IoT devices that support the short-range Bluetooth mesh technology and long-range NB-IoT technology. The optimal configuration is important to consider because there is always a trade-off between data latency and power consumption. One of the challenges for batteryless devices is to account for the variability that exists due to the harvesting source availability and power density. Therefore, this thesis also presents the feasibility of any IoT use case based on the considered power source. Moreover, the results are verified by an experimental prototype for both technologies. It can be observed that a batteryless Bluetooth Mesh LPN can support a data interval of 5 s (bi-directional) when it can harvest 300 μW power by placing a light bulb (6 W warm white LED of 400 lm) at a distance of 110 cm. Whereas, an NB-IoT UE due to its large power-consuming radio, needs 50 mW in combination with a capacitor of more than 1.5 F to communicate at a data rate of 1 s. The large capacitor sizes and high harvesting powers are mainly required because initial network association consumes around 170 mW of power. In order to harvest 50 mW power requires the light bulb to be placed 10 cm apart from the solar panel. However, placing the solar panel near the windowsill or outside can help the device to harvest this much power from natural light as well. Therefore, it can be concluded that the IoT use-cases where the device is placed outside such as smart agriculture, or smart city applications such as tracking city cycles, batteryless NB-IoT is a feasible solution during day time when the capacitor can be charged. This should also be noted that the capacitor charging can be hindered during the night or during heavy rain. Besides, we consider a data rate of 1 s (bi-directional) which is a frequent transmission, so if we decrease the data rate to 60 s, the harvesting power needed is only 6 mW which can be harvested using an indoor light bulb placed at a distance of 25 cm from the solar panel. Apart, most of the current IoT use-cases need a data frequency of 1 hour and only uni-directional communication, which further reduces the requirement of high harvesting power. Therefore, a batteryless NB-IoT device can be considered to deliver sustainable and environment-friendly IoT solutions. Yet, if an IoT use case considers much lower power density ambient sources such as Radio Frequency or vibration (up to 1 mW harvesting power), NB-IoT communication is difficult, still batteryless Bluetooth Mesh communication is viable. Yet, the prototypes based on different harvesting techniques (besides light energy) can be designed in the future to evaluate the batteryless solution. This thesis provides theoretical models which can help to calculate the required harvesting power and capacitor size to support certain data rates.

Moreover, other approaches are considered to reduce the DL latency such as introducing WuR and energy-aware solutions. It is concluded that adding an additional WuR circuit can improve the DL latency of a batteryless Bluetooth Mesh LPN by more than 25%. This improvement can be increased up to 74% by keeping the Bluetooth Mesh LPN always ON using an energy-aware solution by adding a voltage divider circuit to know the capacitor voltage. However, the circuit used can be optimised in future work so that the power consumption of the energy checking task can be reduced. Other than this, the performance improvement can be analysed for duty-cycled WuR or passive WuR, where receivers harvest energy to power themselves from the WuR signal transmitted by the sender (e.g. SWIPT) or the combination of both active and passive WuR.

In addition to focusing on data communication, in future work, the performance of batteryless devices can include sensor data collection and its computation. We performed a few experiments by placing the solar panel on the windowsill harvesting from sunlight. However, a lab setup can be created to emulate dynamic harvesting power which can help in evaluating different stages of a node separately, such as network acquisition stage, data collection, data processing, and data communication. Moreover, the current evaluation is performed by considering power management boards that have a single capacitor unit which can be changed in future to multi-capacitor boards. This can also help in evaluating different stages separately where a large capacitor can be used to support power demanding tasks or as a backup power source.

As with time the IoT technologies also improve their protocol to reduce their power consumption, in the future, this would improve the batteryless device performance as well. NB-IoT already released other power-saving schemes such as *Release Assistance Indication* that indicates the network when to release the connection, *Wake-up signals* that instruct the UE when to start decoding the control and data channels and *Early data transmission* that allows the UE to send or receive data during the random-access procedure. BLE also released new features such as *Extended Advertisement* and *Channel Classification Enhancement*. Once these features are included by the operators and device manufactures in Bluetooth Mesh, their impact can be explored in the future.

Bibliography

[1] Asma Haroon, Munam Ali Shah, Yousra Asim, Wajeeha Naeem, Muhammad Kamran and Qaisar Javaid, "Constraints in the IoT: The World in 2020 and Beyond" International Journal of Advanced Computer Science and Applications(IJACSA), 7(11), 2016. http://dx.doi.org/10.14569/IJACSA.2016.071133 3

[2] Up to 78 million batteries will be discarded daily by 2025, researchers warn. https://cordis.europa.eu/article/id/430457-up-to-78-million-batteries-will-be-discarded-daily-by-2025-researchers-warn (accessed on 20 January 2022) 3, 4

[3] 5 things to know about the LPWAN market in 2021 https://iotbusinessnews.com/2021/10/20/00352-5-things-to-know-about-the-lpwan-market-in-2021/(accessed on 20 January 2022) 3

[4] Bluetooth Low Energy Market Overview https://www.industryarc.com/Report/187/bluetooth-smart-market-forecast.html(accessed on 20 January 2022) 3

[5] ZigBee Market - Growth, Trends, COVID-19 Impact, And Forecasts (2021-2026). https://www.mordorintelligence.com/industry-reports/zigbee-market(accessed on 20 January 2022) 3

[6] Mesh Profile Bluetooth Specifications. https://www.bluetooth.com/specifications/mesh-specifications/(accessed on 20 August 2021). 3

[7] Y. E. Wang, X. Lin, A. Adhikary, A. Grovlen, Y. Sui, Y. Blankenship, J. Bergman and H. S. Razaghi, "A Primer on 3GPP Narrowband Internet of Things," *IEEE Communications Magazine*, vol. 55, no. 3, pp. 117-123, 2017. 3, 80, 81

[8] Opinion: Let's be honest, batteries are bad for the environment https://www.investmentmonitor.ai/insights/batteries-are-bad-for-the-environment (accessed on 05 Feb 2022). 4

[9] Supercapacitor vs Battery - Ultracapacitor Pros and Cons https://www.arrow.com/en/research-and-events/articles/supercapacitor-vs-battery-ultracapacitor-pros-and-cons (accessed on 05 Feb 2022). 4

[10] N. Garg and R. Garg, "Energy harvesting in IoT devices: A survey," 2017 International Conference on Intelligent Sustainable Systems (ICISS), 2017, pp. 127-131, doi: 10.1109/ISS1.2017.8389371. 4

[11] Josiah Hester and Jacob Sorber, "The Future of Sensing is Batteryless, Intermittent, and Awesome," In Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems (SenSys '17), Association for Computing Machinery, New York, NY, USA, Article 21, 2017, 1–6. DOI:https://doi.org/10.1145/3131672.3131699 5

[12] T. Ruan, Z. J. Chew, and M. Zhu, "Energy-aware approaches for energy harvesting powered wireless sensor nodes," *IEEE Sensors Journal*, 17(7), 2165–2173, **2017**. 5

[13] Best Uses of Wireless IoT Communication Technology `https://industrytoday.com/best-uses-of-wireless-iot-communication-technology/`(accessed on 20 August 2021) ix, 4

[14] M. Shirvanimoghaddam, K. Shirvanimoghaddam, M. M. Abolhasani, M. Farhangi, V. Z. Barsari, H. Liu, M. Dohler, and M. Naebe, "Paving the path to a green and self-powered internet of things," Electrical Engineering and Systems Science, 2018. [Online]. Available: http://arxiv.org/abs/1712.02277 ix, 5

[15] Bluetooth mesh concepts. `https://developer.nordicsemi.com/nRF_Connect_SDK/doc/1.6.0/nrf/ug_bt_mesh_concepts.html`(accessed on 24 July 2022). ix, 14

[16] Romero, E.A.D.L. "Experimental Performance Evaluation of the Bluetooth Mesh Protocol for Monitoring Applications," Ph.D. Thesis, Eindhoven University of Technology, Eindhoven, The Netherlands, 2019. 14

[17] A. Aijaz, "Infrastructure-less Wireless Connectivity for Mobile Robotic Systems in Logistics: Why Bluetooth Mesh Networking is Important?,"arXiv:2107.05563, pp. 1-8, Jul. 2021. 17

[18] D. Hortelano, T. Olivares, and M.C. Ruiz, "Reducing the energy consumption of the friendship mechanism in Bluetooth mesh," Computer Networks, vol. 195, pp. 1-16, May 2021. 17

[19] D. Hortelano, T. Olivares, and M. Carmen Ruiz, "Reducing the energy consumption of the friendship mechanism in Bluetooth mesh," Computer Networks, vol. 195, May. 2021. 18

[20] Fahmy, H.A. *Wireless Sensor Networks Energy Harvesting and Management for Research and Industry*; Springer: Berlin/Heidelberg, Germany, 2020. 18

[21] Gawhare, S.B.; Gaikwad, "A. Area and Power efficient High Speed Voltage Comparator," In Proceedings of the International Conference on Automatic Control and Dynamic Optimization Techniques, Pune, India, 9–10 September 2016; pp. 198–201. 21

[22] nRF52840 Product Specification. Available online: `https://infocenter.nordicsemi.com/pdf/nRF52840_PS_v1.0.pdf` (accessed on 20 August 2020). 21

[23] Voltage Divider Calculator. [Online]. Available `https://www.allaboutcircuits.com/tools/voltage-divider-calculator/` (accessed on 30 Sept 2021). 21

[24] Zeadallya, S.; Shaikh, F.K.; Talpur, A.; Sheng, Q.Z. "Design architectures for energy harvesting in the Internet of Things," *Renew. Sustain. Energy Rev.* **2020**, *128*, 1–22. 23

[25] Tang, X.; Wang, X.; Cattley, R.; Gu, F.; Ball, A.D. "Energy Harvesting Technologies for Achieving Self-Powered Wireless Sensor Networks in Machine Condition Monitoring: A Review," *Sensors* **2018**, *18*, 1–39. 23

[26] Adila, A. S.; Husam, A.; Husi, G. "Towards the self-powered Internet of Things (IoT) by energy harvesting: Trends and technologies for green IoT," In Proceedings of the 2nd International Symposium on Small-Scale Intelligent Manufacturing Systems, Cavan, Ireland, 16–18 April 2018; pp. 1–5. 23

[27] S. Silva, H. Martins, A. Valente, and S. Soares, "A Bluetooth Approach to Diabetes Sensing on Ambient Assisted Living systems," Procedia Computer Science, vol. 14, pp. 181–188, Dec. 2012. 23

[28] N. Radhika, P. Tandon, T.V. Prabhakar, and K.J. Vinoy, "RF Energy Harvesting For Self Powered Sensor Platform," *2018 16th IEEE International New Circuits and Systems Conference (NEWCAS)*, Montreal, QC, 24-27 June **2018**. 23

[29] Sanislav, T.; Zeadally, S.; Mois,G.D.; Folea, S.C. "Wireless energy harvesting: Empirical results and practical considerations for Internet of Things," *J. Netw. Comput. Appl.* **2018**, *121*, 149–158. 23

[30] Q. Liu, W. IJntema, A. Drif, P. Pawełczak, M. Zuniga and K. S. Yıldırım, "Perpetual Bluetooth Communications for the IoT," in IEEE Sensors Journal, vol. 21, no. 1, pp. 829-837, Jan. 2021. 23

[31] Fraternali, F.; Balaji, B.; Agarwal, Y.; Benini, L.; Gupta, R. "Pible: Battery-Free Mote for Perpetual Indoor BLE Applications." In Proceedings of the 5th Conference on Systems for Built Environments, Shenzen, China, 7–8 November 2018; pp. 168–171. 23

[32] M. Nilsson, and H. Deknache, "Investigation of Bluetooth Mesh and Long Range for IoT Wearables," Malmo Universitet/Teknik och Samhälle: Malmö, Sweden, 2018. 23

[33] Petäjäjärvi, J.; Kaleva, J.; Mikhaylov, K.; Pulkkinen, H.; Ahola, J.; Björkgren, M. "Automatic charging of an energy harvesting powered sensor node from controllable energy source," In Proceedings of the 2018 14th International Wireless Communications Mobile Computing Conference (IWCMC), Limassol, Cyprus, 25–29 June 2018; pp. 600–605. 23

[34] Meli, M.; Würth, M. "Advertising position with batteryless Bluetooth Low Energy," In Proceedings of the Embedded World Conference, Nuremberg, Germany, 28 February–1 March 2012; pp. 1–13. 24

[35] Meli, M.; Würth, M. "Indoor batteryless temperature and humidity sensor for Bluetooth Low Energy," In Proceedings of the Wireless Congress, Munich, Germany, 9–10 November 2011; pp. 1–9. 24

[36] T. Wu, J.-M. Redoute, and M. R. Yuce, "Subcutaneous solar energy harvesting for self-powered wireless implantable sensor systems," *Proc. 40th Annu. Int. Eng. Med. Biol. Conf.*, Honolulu, HI, USA, Jul. **2018**. 24

[37] K. E. Jeon, J. She, J. Xue, S. Kim, and S. Park, "LuXbeacon -a batteryless beacon for green IoT: Design, modeling, and field tests," *IEEE Internet Things J.*, pp. 1–1, **2019**. 24

[38] H.D. Jang, "Study on Design Method of Energy Harvesting System for BLE Beacon", *The Journal of Korean Institute of Communications and Information Sciences*, vol. 42, pp. 149-152 **2017**. 24

[39] H.D. Jang, "Study on the Design Method of the Energy Harvesting Smart Sensor for Implementing IoT Service", *Journal of the Korea Institute of Information, Electronics and Communication Technology (jkiiect)*, 18-02, Vol.11 No.1. 24

[40] Witham, O.; Johnston, L.N.; Xiao, M.; Feng, J.; Zhou, N.; Shaker, G. "Batteryless Wireless Water Leak Detection System," In Proceedings of the International Conference on Smart Applications, Communications and Networking, Sharm El Sheik, Egypt, 17–19 December 2019; pp. 1–4. 24

[41] Brunecker, O.; Magno, M., "TinyBird: An Energy Neutral Acoustic Bluetooth-Low-Energy Sensor Node with RF Energy Harvesting," In Proceedings of the 7th International Workshop on Energy Harvesting & Energy-Neutral Sensing Systems, New York, NY, USA, 10 November 2019; pp. 1–7. 24

[42] Y. Zhong, B. Qian, Y. Zhu, Z. Ren, J. Deng, J. Liu, Q. Bai, and X. Zhan, "Development of an Implantable Wireless and Batteryless Bladder Pressure Monitor System for Lower Urinary Tract Dysfunction," *IEEE J. Transl. Eng. Health Med.* **2019**, *8*, 1–7. 24

[43] F. Álvarez, L. Almon, A. Hahn, and M. Hollick, "Toxic Friends in Your Network: Breaking the Bluetooth Mesh Friendship Concept," Proceedings of the 5th ACM Workshop on Security Standardisation Research Workshop, pp. 1-12, Nov. 2019. 24

[44] S. Darroudi, R. Caldera-Sànchez, and C. Gomez, "Bluetooth Mesh Energy Consumption: A Model," Sensors, vol. 19, no. 5, Mar. 2019. 24

[45] Piyare, R.; Murphy, A.L.; Kiraly, C.; Tosato, P.; Brunelli, D., "Ultra Low Power Wake-Up Radios: A Hardware and Networking Survey," *IEEE Commun. Surv. Tutor.* **2017**, *19*, 2117–2157. 24

[46] Bello, H.; Xiaoping, Z.; Nordin, R.; Xin, J. "Advances and Opportunities in Passive Wake-Up Radios with Wireless Energy Harvesting for the Internet of Things Applications," *Sensors* **2019**, *19*, 3078. 24

[47] Gavrikov, P.; Verboket, P.E.; Ungan, T.; Müller, M.; Lai M.; Schindelhauer, C.; Reindl, L.M. ; Wendt, T. "Using Bluetooth Low Energy to trigger an ultra-low power FSK wake-up receiver," In Proceedings of the 25th IEEE International Conference on Electronics, Circuits and Systems, Bordeaux, France, 9–12 December 2018; pp. 781–784. 24

[48] Ding, M.; Zhang, P.; Lu, C.; Zhang, Y.; Traferro, S.; Schaik, G.J.; Liu, Y.H.; Huijts, J.; Bachmann, C.; Dolmans, G.; Philips, K. "A 2.4ghz ble-compliant fully-integrated wakeup receiver for latency-critical iot applications using a 2-dimensional wakeup pattern in 90nm cmo," In Proceedings of the IEEE Radio Frequency Integrated Circuits Symposium (RFIC), Honolulu, HI, USA, 4–6 June 2017; pp. 168–171. 24

[49] Roberts, N.E.; Craig, K.; Shrivastava, A.; Wooters, S.N.; Shakhsheer, Y.; Calhoun, B.H.; Wentzloff, D.D. "26.8 a 236nw -56.5dbm-sensitivity bluetooth low-energy

wakeup receiver with energy harvesting in 65nm cmos," In Proceedings of the IEEE International Solid-State Circuits Conference, San Francisco, CA, USA, 31 January–4 February 2016; pp. 450–451. 24

[50] Giovanelli, D.; Milosevic, B.; Brunelli, D.; Farella, E. "Enhancing Bluetooth Low Energy with wake-up radios for IoT applications," In Proceedings of the 2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC), Valencia, Spain, 26–30 June 2017; pp. 1–6. 25

[51] Mikhaylov, K.; Karvonen, H. "Wake-up radio enabled BLE wearables: Empirical and analytical evaluation of energy efficiency," In Proceedings of the 14th International Symposium on Medical Information Communication Technology, Nara, Japan, 20–22 May 2020; pp. 1–5. 25

[52] Sanchez, A.L. "Study of Adaptation Mechanisms of the Wireless Sensor Nodes to the Context for Ultra-low Power Consumption," Ph.D. Thesis, University of Grenoble, Grenoble, France, 2016; pp. 1–127. 25

[53] Abdelhamid, M.R.; Paidimarri, A.; Chandrakasan, A.P. "A -80dBm BLE-Compliant, FSK wake-up Receiver with System and Within-Bit Duty Cycling for Scalable Power and Latency," In Proceedings of the 2018 IEEE Custom Integrated Circuits Conference (CICC), San Diego, CA, USA, 8–11 April 2018; pp. 1–4. 25

[54] Ding, M.; Zhang, P.; Lu, C.; Zhang, Y.; Traferro, S.; Schaik, G.J.; Liu, Y.H.; Huijts, J.; Bachmann, C.; Dolmans, G.; Philips, K. "A 2.4GHz BLE-compliant fully-integrated wakeup receiver for latency-critical IoT applications using a 2-dimensional wakeup pattern in 90nm CMOS," In Proceedings of the IEEE Radio Frequency Integrated Circuits Symposium (RFIC), Honolulu, HI, USA, 4–6 June 2017; pp. 168–171. 25

[55] Wang, P.P.; Mercier, P.P. "A 220 μW -85 dBm sensitivity BLE compliant wake-up receiver achieving -60 dB SIR via single-die multichannel FBAR-based filtering and a 4-Dimentional wake-up signature," In Proceedings of the 2019 IEEE International Solid-State Circuits Conference-(ISSCC), San Francisco, CA, USA, 17–21 February 2019; pp. 440–442. 25

[56] Liu, Q.; IJntema, W.; Drif, A.; Pawelczak, P.; Zuniga, M. "BEH: Indoor Batteryless BLE Beacons using RF Energy Harvesting for Internet of Things," *arXiv* **2019**, arXiv:1911.03381. 25

[57] A. K. Sultania, C. Delgado, and J. Famaey, "Enabling Low-Latency Bluetooth Low Energy on Energy Harvesting Batteryless Devices Using Wake-Up Radios," Sensors, vol. 20, no. 18, p. 5196, Sep. 2020. 25

[58] Delgado, C.; Sanz, J.M.; Famaey, J. "On the Feasibility of Batteryless LoRaWAN Communications using Energy Harvesting," In Proceedings of the IEEE Global Communications Conference (GLOBECOM), Waikoloa, HI, USA, 9–13 December 2019; pp. 1–6. 27, 28

[59] C. Blondia, "A queueing model for a wireless sensor node using energy harvesting," Telecommun Syst 77, pp. 335–349, Feb. 2021. 32, 33, 34

[60] Stochastic, Karl Sigman [Online]. Available:http://www.columbia.edu/~ks20/stochastic-I/stochastic-I-LL.pdf, Accessed on: Aug. 04, 2021 33

[61] nRF52840 DK, Nordic Semiconductor [Online]. Available:https://www.nordicsemi.com/Software-and-tools/Development-Kits/nRF52840-DK, Accessed on: Jun. 22, 2021 37

[62] Nordic Semiconductor. Available online: https://devzone.nordicsemi.com/nordic/power/ (accessed on 20 August 2020). 37

[63] User Guide Evaluation board for AEM10941, epeas, [Online]. Available:https://e-peas.com/wp-content/uploads/2020/06/epeas_Solar_Energy_Harvesting_User_Guide_AEM10941.pdf, Accessed on: Jun. 22, 2021 37, 50, 59, 111

[64] Data Sheet AS3933. Available online: http://www1.futureelectronics.com/doc/AUSTRIAMICROSYSTEMS/AS3933-BQFT.pdf (accessed on 20 August 2020). 47

[65] Energy Sources and Power Management in IoT Sensors and Edge Devices. Available online: https://jaxenter.com/energy-sources-power-management-iot-sensors-edge-devices-145006.html (accessed on 8 August 2020). 50

[66] Karvonen, H.; Mikhaylov, K.; Acharya, D.; Rahman, M. "Performance Evaluation of Bluetooth Low Energy Technology Under Interference," In *EAI International Conference on Body Area Networks*; Springer: Cham, Switzerland, 2020; pp. 147–156. 51

[67] Natarajan, R.; Zand, P.; Nabi, M. "Analysis of Coexistence between IEEE 802.15.4, BLE and IEEE 802.11 in the 2.4 GHz ISM Band," In Proceedings of the IECON 2016—42nd Annual Conference of the IEEE Industrial Electronics Society, Florence, Italy, 23–26 October 2016; pp. 6025–6032. 51

[68] Silva, S.; Fernandes, T.; Valente, A.; Moreira, "A. Coexistence and Interference Tests on a Bluetooth Low Energy Front-End," In Proceedings of the IEEE Science and Information Conference, London, UK, 27–29 August 2014; pp. 1014–1018. 51

[69] Ghamari, M.; Villeneuve, E.; Soltanpur, C.; Khangosstar, J.; Janko, B.; Sherratt, R.S.; Harwin, W. "Detailed Examination of a Packet Collision Model for Bluetooth Low Energy Advertising Mode," *IEEE Access* **2018**, *6*, 46066–46073. 51

[70] Marinkovic, S.J., Popovici, E.M.: "Nano-power wireless wake-up receiver with serial peripheral interface," IEEE J. Sel. Areas Commun., 2011, 29, (8), pp. 1641–1647 (https://doi/org/10.1109/JSAC.2011.110913) 55

[71] Power Profiler Kit II. [Online]. Available https://www.nordicsemi.com/Products/Development-hardware/Power-Profiler-Kit-2 (accessed on 30 Sept 2021). 57, 113

[72] S. Bunjongjit, A. Ngaopitakkul, and M. Leelajindakrairerk , "Analysis of Harmonics in Indoor Lighting System with LED and Fluorescent Luminaire" , *2017 IEEE 3rd International Future Energy Electronics Conference and ECCE* Asia, Kaohsiung, Taiwan, 3-7 June **2017**. 58

[73] J. Bernegger, M. Meli, "Comparing the energy requirements of current Bluetooth Smart solutions," *Embedded World Conference*, Nuremberg, Germany, 25-27 February **2014**. 59

[74] nRF52840 DK. [Online]. Available `https://www.nordicsemi.com/Products/Development-hardware/nrf52840-dk` (accessed on 30 Sept 2021). 60

[75] Low Power node example. [Online]. Available `https://infocenter.nordicsemi.com/topic/com.nordic.infocenter.meshsdk.v4.2.0/md_examples_lpn_README.html` (accessed on 30 Sept 2021). 60

[76] Light switch example. [Online]. Available `https://infocenter.nordicsemi.com/topic/com.nordic.infocenter.meshsdk.v4.2.0/md_examples_light_switch_README.html` (accessed on 30 Sept 2021). 60

[77] T. Olofsson, "Energy Harvesting with Solar Cells for Wireless Alarm Nodes", *Faculty of Engineering, Lund University*, **2018**. 62

[78] Antwerp Historical Weather, BE. [Online]. Available `https://www.worldweatheronline.com/antwerp-weather-history/be.aspx` (accessed on 30 Sept 2021). 72

[79] Past Weather in Antwerp, Belgium. [Online]. Available `https://www.timeanddate.com/weather/belgium/antwerp/historic/` (accessed on 30 Sept 2021). 72

[80] *Cisco, 2018,Power Saving Mode (PSM) in UEs*. `https://www.cisco.com/c/en/us/td/docs/wireless/asr_5000/21/MME/b_21_MME_Admin/b_21_MME_Admin_chapter_0111010.pdf`(accessed on 20 January 2022). 79, 80

[81] F. Xie, 2018, *Power Consumption optimisation.* `https://cdn.rohde-schwarz.com/fr/general_37/local_webpages/2018_IoT_test_Day-Power_consumption_Optimisation.pdf`(accessed on 20 January 2022). 79

[82] Martínez, B., F. Adelantado, A. Bartoli, and X. Vilajosana. "Exploring the Performance Boundaries of NB-IoT," *Networking and Internet Architecture, arXiv*, 2018. 79

[83] Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures, v16.2.0, 3GPP Standard TS 36.213, 2020. 81, 90

[84] *Narrowband Internet of Things Whitepaper, Rohde-Schwarz*. `https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_application/application_notes/1ma266/1MA266_0e_NB_IoT.pdf`(accessed on 20 January 2022). 82

[85] P. A. Maldonado, P. Ameigeiras, J. P. Garzon, J. N. Ortiz and J. M. L. Soler, "NarrowBand IoT Data Transmission Procedures for Massive Machine Type Communications," *IEEE Networks Magazine*, vol. 31, pp. 8-15, 2017. 83

[86] P.A. Maldonado, M. Lauridsen, P. Ameigeiras, and J.M. Lopez-Soler, "Analytical Modeling and Experimental Validation of NB-IoT Device Energy Consumption," *IEEE Internet of Things Journal*, vol. 6, no. 3, 2019, pp. 5691-5701. 83, 90

[87] J. Lee and J. Lee, "Prediction-based energy saving mechanism in 3GPP NB-IoT networks," *Sensors (Basel)*, vol. 17, no. 9, 2017. 83

[88] H. Bello, X. Jian, Y. Wei, and M. Chen, "Energy-Delay Evaluation and optimisation for NB-IoT PSM With Periodic Uplink Reporting," *IEEE Access 2019*, vol. 7, pp. 3074-3081. 83

[89] K. Liu, G. Cui, Q. Li, S. Zhang, W. Wang, and X. Li, "An Optimal PSM Duration Dalculation Algorithm For NB-IoT," *2019 IEEE 5th International Conference on Computer and Communications (ICCC)*, Chengdu, China, Dec. 2019, pp. 447-453. 83

[90] R. Harwahyu, R.G. Cheng, C.H. Wei and R.F. Sari, "Optimisation of Random Access Channel in NB-IoT," *IEEE Internet of Things Journal*, Vol 5, no. 1, Feb 2018, pp 391-402. 84

[91] A. Azari, C. Stefanovic, P. Popovski and C. Cavdar, "On the Latency-Energy Performance of NB-IoT Systems in Providing Wide-Area IoT Connectivity," *IEEE Transactions on Green Communications and Networking*, 2019, pp. 1-22. 84

[92] H. Li, G. Chen, Y. Wang and W. Dong, "Accurate Performance Modeling of Uplink Transmission in NB-IoT," *IEEE 24th International Conference on Parallel and Distributed Systems*, 2018, pp. 910-917. 84

[93] S.-M. Oh, K.-R. Jung, M. S. Bae, and J. Shin, "Performance analysis for the battery consumption of the 3GPP NB-IoT device," *in Proc. ICTC*, Oct. 2017, pp. 981-983. 84

[94] S. Xu, Y. Liu, and W. Zhang, "Grouping-based discontinuous reception for massive narrowband Internet of Things systems," *IEEE Internet Things Journal*, vol. 5, no. 3, Jun 2018, pp. 1561-1571. 84

[95] C. C. Tseng, H. C. Wang, F. C. Kuo, K. C. Ting, H. H. Chen and G. Y. Chen, "Delay and Power Consumption in LTE/LTE-A DRX Mechanism with Mixed Short and Long Cycles," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 3, 2016, pp. 1721-1734. 84

[96] Y. Miao, W. Li, D. Tian, M. S. Hossain, and M. F. Alhamid, "Narrow band Internet of Things: Simulation and modelling," *IEEE Internet Things Journal.*, vol. 5, no. 4, Aug 2018, pp. 2304-2314. 84

[97] M. E. Soussi, P. Zand, F. Pasveer, and G. Dolmans, "Evaluating the performance of eMTC and NB-IoT for smart city applications," *IEEE Int. Conf. Commun. (ICC)*, Kansas City, MO, USA, May 2018, pp. 1-7. 84

[98] A.K. Sultania, C. Delgado and J. Famaey, "Implementation of NB-IoT Power Saving Schemes in ns-3",*Workshop on Next-Generation Wireless with ns-3*, pp. 5-8, 2019. 84, 99

[99] M. Lauridsen, R. Krigslund, M. Rohr, and G. Madueno, "An empirical NB-IoT power consumption model for battery lifetime estimation," *in Proc. IEEE 87th Veh. Technol. Conf. (VTC Spring), Porto, Portugal*, Jun. 2018, pp. 1-5. 84

[100] Haridas, A.; Rao, V.S.; Prasad, R.V.; Sarkar, C. "Opportunities and challenges in using energy-harvesting for NB-IoT," ACM SIGBED Rev. 2018, 15, 7–13. 84

[101] M. Lukic, S. Sobot, I. Mezei, D. Vukobratovic, and D. Danilovic, "In-depth real-world evaluation of nb-iot module energy consumption," in 2020 IEEE International Conference on Smart Internet of Things (SmartIoT), 2020, pp. 261–265. 84

[102] S. Paiva, S. Branco and J. Cabral, "Design and Power Consumption Analysis of a NB-IoT End Device for Monitoring Applications," IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society, 2020, pp. 2175-2182, doi: 10.1109/IECON43393.2020.9254374. 84

[103] F. Gabelle "Narrowband-IoT Power Saving Modes – A Comprehensive Study", Oulu University of Applied Sciences, 2021 https://www.theseus.fi/bitstream/handle/10024/493777/Gabelle_Florian.pdf?sequence=2 84

[104] Zolitron, "Z-Node:World's first and only autonomous cognitive sensor," https://zolitron.com/our-products/zolitron-z-node(accessed on 20 January 2022). 85

[105] Street light controller - LUMAWISE plug https://www.elkoep.com/street-light-controller---lumawise-plug---airslc-100nblwes_0-10(accessed on 20 January 2022) 85

[106] NDDS20 NB-IoT Waterproof LPWAN https://www.iot-store.com.au/collections/sensors/products/ndds20-nb-iot-lpwan-liquid-level-sensor(accessed on 20 January 2022) 85

[107] NB-IOT Outdoor – Indoor Asset Tracker (GPS, BLE) https://iotfactory.eu/products/iot-sensors/nb-iot-outdoor-indoor-asset-tracker-gps-wifi-ble/(accessed on 20 January 2022) 85

[108] NETOP PIR sensor https://market.thingpark.com/netop-nb-iot-pir-sensor.html(accessed on 20 January 2022) 85

[109] NB-IoT Room Sensor https://www.pikkerton.com/_objects/1/83.htm(accessed on 20 January 2022) 85

[110] NB-IoT/Cat-M1 Analog-/ Digital-In-Box https://www.pikkerton.com/_objects/1/75.htm(accessed on 20 January 2022) 85

[111] Discover 100% Solar-powered IoT Asset Tracker [SODAQ/e-peas] https://e-peas.com/news/discover-100-solar-powered-iot-asset-tracker/(accessed on 20 January 2022) 85

[112] IP69 SODAQ https://sodaq.com/products/ip69/ (accessed on 20 January 2022) 85

[113] INDU-EYE Vibro https://aeinnova.com/wp-content/uploads/2021/06/INDU-EYE-VIBRO-Brochure_NB-IoT_V1.pdf(accessed on 20 January 2022) 85

[114] INDU-EYE Termo https://aeinnova.com/wp-content/uploads/2021/06/INDU-EYE-TERMO-Brochure_NB_IoT_V1.pdf(accessed on 20 January 2022) 85

[115] INDU-EYE Vibro WIND LoRa / NB-IOT https://www.hannovermesse.de/product/indu-eye-vibro-wind-lora-nb-iot/267766/N1462207(accessed on 20 January 2022) 85

[116] Xnor's new solar-powered cameras present a telecom opportunity https://www.businessinsider.com/xnor-solar-cameras-telecom-opportunity-2019-2?r=US&IR=T(accessed on 20 January 2022) 85

[117] Vibration Energy Harvesting https://www.8power.com/(accessed on 20 January 2022) 85

[118] Perpetuum https://perpetuum.com/(accessed on 20 January 2022) 85

[119] F. Orfei, C. Benedetta Mezzetti, and F. Cottone, "Vibrations powered LoRa sensor: An electromechanical energy harvester working on a real bridge," in Proc. IEEE Sens., FL, USA, 2016, pp. 1–3. 85

[120] H. Malik, H. Pervaiz, M. M. Alam, Y. L. Moullec, A. Kuusik, and M. A. Imran, "Radio resource management scheme in NB-IoT systems," *IEEE Access*, vol. 6, pp. 15051–15064, 2018. 89

[121] *Solution for Cellular IoT*. http://www.effnet.com/solutions/cellular_iot/(accessed on 20 January 2022). 89

[122] Cellular System Support for Ultra Low Complexity and Low Throughput Internet of Things, document 3GPP TR 45.820 v13.0.0, 2015. 90

[123] D.L. González, "Theoretical and experimental analysis of NB-IoT technology," Bachelors Degree thesis, *Polytechnic University of Catalonia*, 2019. 91

[124] M. El Soussi, P. Zand, F. Pasveer and G. Dolmans, "Evaluating the Performance of eMTC and NB-IoT for Smart City Applications", *IEEE International Conference on Communications (ICC)*, 2018. 99

[125] *SARA-N2 series*. https://www.u-blox.com/en/docs/UBX-18066692(accessed on 20 January 2022). 99

[126] Solar Panel https://www.mikroe.com/solar-panel(accessed on 20 January 2022) 111

[127] nRF9160 https://www.tme.eu/Document/5586901f3019b07f5dcd6f1b87c61e45/nRF9160.pdf(accessed on 20 January 2022) 113

[128] Sunrise, Sunset, and Daylength https://www.timeanddate.com/sun/belgium/brussels?month=9(accessed on 20 January 2022) 120