UNIVERSITY OF CALIFORNIA,
IRVINE

The Use of Bluetooth Low Energy for Continuous Monitoring of Body Sensor Networks

THESIS

submitted in partial satisfaction of the requirements
for the degree of

MASTER OF SCIENCE

in Electrical and Computer Engineering

by

Michael Atef Ayoub

Thesis Committee:
Professor Ahmed Eltawil, Chair
Professor Ender Ayanoglu
Professor Ozdal Boyraz

2019

DEDICATION

To

My dear wife Monica and my parents Samia and Atef

For their continuous support and encouragement

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

ACKNOWLEDGMENTS

A thesis does not come into being by only one person writing in a vacuum, but is strongly dependent on the moral, practical, and emotional support of many people. I would like to take this chance to express my thanks to them all. First and foremost, I thank God.

Special thanks are due to Professor Ahmed El-Tawil, my committee chair, for his valuable guidance and support throughout my research. I would also like to thank the committee members who took the time to read my thesis; Professor Ender Ayanoglu and Professor Ozdal Boyraz.

I extend my gratitude to ThirdWayv Inc. and in particular to Dr. Nabil Wasily. His practical support, advice and encouragement have been invaluably helpful throughout this process.

I also owe Dr. Mahmoud Ismail and Dr. Mirette Sadek for their trust in me and their valuable recommendations, giving me the chance to enter in this esteemed program.

I am also very grateful to my whole family, especially my wife (and proof-reader) Monica, my parents Samia and Atef, my sister Lydia and her family (Joseph, Mark and Dany), my in-laws Mervat and Sami and siblings-in-law Marina and Daniel for their wonderful support throughout this journey.

This work would not have been possible without several dear friends. I am deeply grateful to Ibrahim Alquaydheb, Ahmed Alzughaibi, Mohamed Fouda, Ahmed Ibrahim, Ahmed Khorshid and Sergey Shaboyan, for their presence, encouragement, practical support and advice.

Thank you all again for being there for me. I can never thank you enough.

ABSTRACT OF THE THESIS

The Use of Bluetooth Low Energy for Continuous Monitoring of Body Sensor Networks

By

Michael Atef Ayoub

Master of Science in Electrical and Computer Engineering

University of California, Irvine, 2019

Professor Ahmed Eltawil, Chair

Wireless communications enable remote monitoring and controlling of Body Sensor Networks (BSN), thus playing a key role in the development of numerous medical and fitness applications and providing various advantages in terms of cost and user's convenience. However, several issues have been brought to the surface regarding the sensor's size and lifetime.

Bluetooth Low Energy (BLE) is a booming wireless technology that targets low-power, low-complexity and low-throughput applications and thus an excellent candidate for BSN connectivity. Additionally, BLE-powered smartphones can act as user controllers and internet gateways for BSN devices, at no extra cost of network deployment.

This thesis conducts a study on the use of BLE technology in continuous monitoring of BSNs in terms of the required throughput, power consumption and latency. We compare the performance of different versions of the Bluetooth core specification using a theoretical model and an experimental setup based on nRF52840 chip by Nordic Semiconductor. We focus on Electrocardiography (EKG) and give the current consumption and battery lifetime estimation of an EKG gateway and an EKG node for different BLE versions and configurations.

1 Introduction

1.1 Body Sensor Networks

The field of wireless Body Sensor Networks (BSNs) has become a hot topic in both academia and the industry in the past few years and its importance will continue to increase as long as wireless devices continue to play greater roles in our lives.

A couple of decades ago, cell phones were only used to make phone calls. A few years later, people started to use their smartphones to check email, access social media, browse the internet, take photos and videos, navigate using maps, play games, buy and sell stuff, manage banking accounts and many other applications. Currently, connectivity is not only limited to smartphones and tablets. Every day more devices become connected, from vehicles to electrical appliances, light bulbs, access gates and various sensor devices.

This is not where the story ends; this is just the beginning. Connectivity capabilities have eventually been added to various wearable and implanted devices, interacting with the human body itself. Examples of such devices include smart watches, fitness trackers, insulin pumps, continuous glucose monitors and pulse oximeters, as well as sensors that measure respiratory rate, blood pressure, body temperature, Electromyography (EMG), Electroencephalogram (EEG) and Electrocardiography (EKG). This led to the evolution of the concept of a BSN. BSNs have various useful applications that can provide health services more effectively, at a lower cost, usable from home and user convenient. Ultimately BSNs can save lives.

An important aspect of a BSN system is the wireless communication protocol that enables continuous monitoring of body activities. Nodes may be required to talk to each other

and/or to a controller device such as a smartphone. Patient data may also be instantaneously uploaded and processed by the cloud and reviewed by the health provider. In such case, the user and/or the health provider can be alerted when the case requires an emergency action. Thus a complete eHealthcare system can be provided as shown in figure 1.1.

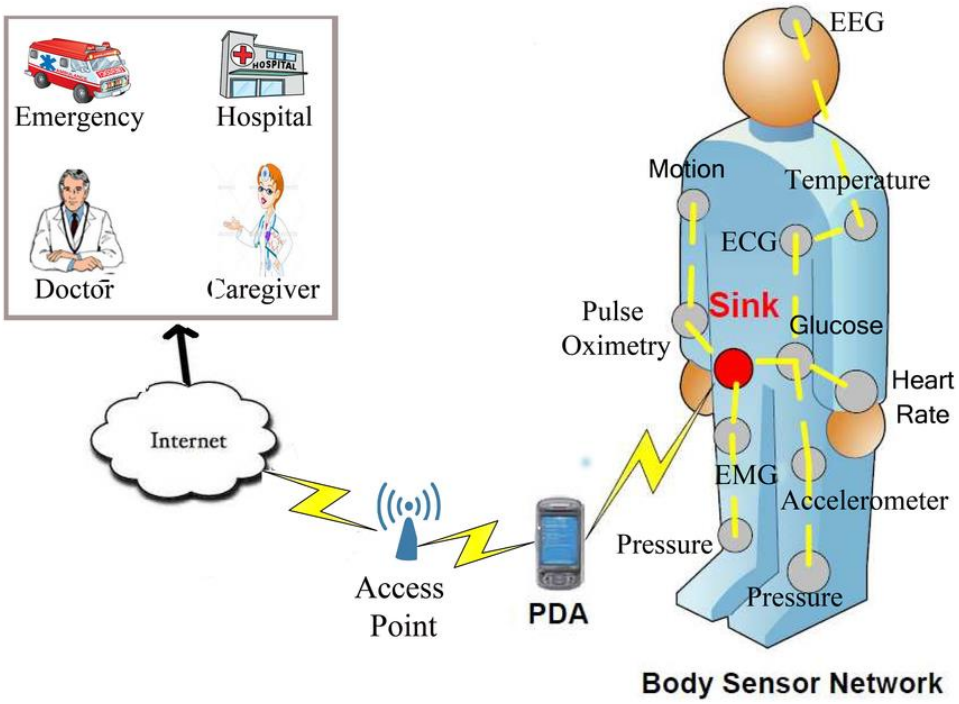

Figure 1-1: BSN as a part of an eHealthcare system. Retrieved from [1]

When it comes to wireless communications in BSNs, there are always two big problems. The first is power consumption. There are three main contributors to the power consumption of a wireless sensor; sensing, data processing and wireless communication. Compared to sensing and data processing, wireless communication typically consumes a significant amount of energy [2]. Wireless protocols are usually power-hungry, while sensor devices typically use small coin cell batteries and are required to operate for long time. The second problem with wireless communications is related to security. All wireless communications have the broadcast property by definition, so a device cannot prevent an attacker from receiving the same packets it receives

or impersonating the device by sending packets on the device's behalf. One solution is to use proprietary radio frequencies but this requires special hardware, does not allow users to control their devices using their smartphones and forces them to carry more and more controller devices, which is not convenient. Ultimately, a radio protocol security layer and an application-level security layer are needed, especially in medical applications that may affect human health.

In this thesis we mainly focus on the low-power and throughput requirements for continuous data monitoring of BSNs and leave the security problem as an opportunity for future research.

1.2 BSN radio protocols

A survey on BSN radio protocols by the authors of [2] included Bluetooth Low Energy (BLE), Zigbee, classic Bluetooth and ANT as well as less-often used protocols like RuBee, Sensium, Zarlink, Z-Wave, Insteon, Wavenis, BodyLAN, Dash7, ONE-NET, EnOcean and emerging Intra-Body Communication (IBC) technologies. Among this list, the authors found BLE and Zigbee as the most prominent. Compared to classic Bluetooth, BLE uses only 1%-50% of the power consumption of classic Bluetooth depending on the use case [3]. One important advantage of BLE over classic Bluetooth is that BLE utilizes fewer channels during the pairing process and consumes considerably less time (few milliseconds) for device discovery and synchronization compared to seconds for classic Bluetooth [2]. When it comes to the comparison between BLE, Zigbee and ANT, it highly depends on the use case. A Power consumption analysis of Bluetooth Low Energy, ZigBee and ANT sensor nodes in a cyclic sleep scenario is carried out in [4], concluding that BLE achieved the lowest power consumption, followed by ZigBee and ANT, where BLE won mainly because of the reconnection time.

Other than the prominent BLE performance, the wide penetration of BLE in smartphones, tablets, laptops and smart watches makes it very adequate to be used in BSNs. Instead of having multiple special-purpose controllers for different sensors, it is very convenient if the user's smartphone for example can serve as a central controller and data collector for the whole BSN. Moreover, since smartphones are usually connected to 3G or LTE networks, such devices act as gateways for the data collected over BLE to reach the cloud, without the need of additional infrastructure.

In the field of BSNs, one challenge that faces BLE and other RF protocols in general is the RF signal absorption by the human body. The human body is not-homogeneous, featuring various elements and organs with different dielectric constants, thickness and characteristic impedance values and thus it is not a good medium for RF propagation. Emerging technologies like IBC and Near Field Magnetic Induction (NFMI) try to provide an alternative to RF protocols for BSNs. However, even such technologies still need an RF technology like BLE to offer a gateway for the data collected by the BSN.

1.3 BSN requirements

BSN applications vary in their requirements. For instance, the required throughput can be as low as few bytes every few minutes or as high as tens or even hundreds of kbps of continuous data transfer. Table 1.1, retrieved from [5], shows typical throughput for some medical applications. Among this list we are only interested in applications that fit in BSNs, as the rest of the list requires special setup at a hospital or clinic.

Table 1-1: Typical Throughput for Some Medical Applications. Retrieved from [5]

Application	Required Data Rate
ECG (Electrocardiogram)	15kbps to 288 kbps
EMG (Electromyography)	320 kbps
EEG(Electroencephalography)	10kbps to 43.2kbps
Blood Saturation	16 bps
Glucose Monitoring	1600 bps
Temperature	120 bps
Motion Sensor	35 kbps
Cochlear Implant	100 kbps
Artificial Retina	50-700 kbps
Audio	4kbps -1Mbps
Voice	50-100 kbps
Heart Sound	120 kbps
Ultrasound, Cardiology, Radiology	2048 kbps
Magnetic Resonance Image	3072 kbps
Scanned X-Ray	14.4 Mbps
Digital Radiography	48 Mbps
Mammogram	192 Mbps
Compressed and Full Motion Video	348 kbps to 1.544 Mbps

Different topologies and techniques may fit different applications. For intermittent data transfer, it may be more power-efficient for communicating devices to disconnect and then reconnect whenever needed later on. This is because maintaining an ongoing wireless connection often requires periodic keep-alive and link-control packets. On the other hand, other applications require continuous data streaming with almost constant throughput and low latency. We focus on EKG as an ideal example of the latter case. The EKG signal is a representation of the heart muscle activity that has the following properties:

- The signal bandwidth is 0.05-100 Hz [6].

- As a result, the heart rate sampling frequency is typically chosen between 250 Hz and 500 Hz or higher [7].
- EKG values are commonly represented using 15-16 bits per sample [8].
- Additionally, low power consumption for the EKG sensor is an essential requirement to achieve a convenient duration of sensor's lifetime.

A single EKG measurement at one body position is called a single-lead EKG, which can be done by a commercial wearable patch that provides continuous monitoring such as iRhythm's Zio patch [9]. On the other hand, a 12-lead EKG is a conventional medical procedure that takes place using ten electrodes placed over the patient's limbs and chest. In [10], the total throughput of a 12-lead EKG is considered as 64 kbps.

Some medical applications may have restricted latency requirements as well. For example, a wireless EKG electrode defined by the IEEE 1073 group generates 4 kbps of data and the latency introduced by the packetization of the samples and the transmission delay shall remain below 500 ms [11].

The use cases previously mentioned, including the data transfer scheme, throughput, latency and power consumption makes BLE an excellent candidate for such applications. In this thesis, we test BLE performance in BSN applications that require continuous data streaming with almost constant throughput and low latency. We first give an overview of the BLE protocol in chapter 2. In chapter 3, we provide a summary of the related work. Chapter 4 discusses our experimental setup as well as the throughput and current consumption models that fit this setup. In chapter 5 we present our results. Finally, chapter 6 gives the conclusion of this thesis.

2 Bluetooth Low Energy (BLE) Background

2.1 Introduction to BLE technology

The BLE technology is part of the Bluetooth specification that is managed by the Bluetooth Special Interest Group (SIG). The idea behind BLE was first developed by Nokia in 2004. The first version of BLE was published in 2010, as part of the Bluetooth specification version 4.0. The current version of the specification is version 5, which was released in December 2016 [12].

BLE has several advantages over competing wireless standards [13]:

(1) Power Consumption: the low energy design translates into coin cell battery life of months to years.

(2) Low complexity: BLE provides a simple protocol to advertise services that the sensor supports and to connect to it. This results in low implementation cost.

(3) Wide adoption in the smartphone industry, making it very convenient to control many BLE sensors and devices using personal smartphones that users already have.

For these reasons, BLE is very suitable for many applications that require low throughput, complexity, cost and power consumption. It is worth mentioning that BLE does not provide backward compatibility with the earlier versions of the Bluetooth specification, which we will simply denote by “classical Bluetooth”.

2.2 BLE frequency band and channels

BLE uses the unlicensed 2.4-GHz Industrial, Scientific and Medical (ISM) radio band. The BLE spectrum is divided into 40 adjacent channels whose center frequencies are 2-MHz apart from each other. Three channels are used for connectionless communications and device discovery and are called advertising channels. As stated earlier, the device discovery in BLE using a reduced number of channels enables the pairing process to take only few milliseconds and is a main contributor towards the decreased complexity and power consumption of BLE.

The other 37 channels are used in the connection state and are called data channels. BLE 5 allows the use of data channels for some connectionless activities.

To minimize collisions with WiFi traffic which utilizes the same frequency band, the advertising channel locations are selected in such a way to avoid the most commonly used WiFi channels, as shown in figure 2.1, while adaptive frequency hopping is used in the data channels, enabling the exclusion of any channels that encounter WiFi collisions or high packet loss in general.

2.3 Overview of the BLE stack

The BLE protocol is a layered protocol consisting of host and controller subsystems, where the user's application resides on top of the host subsystem. Each subsystem includes several layers as shown in figure 2.2. A brief description of different layers of the BLE stack is given next.

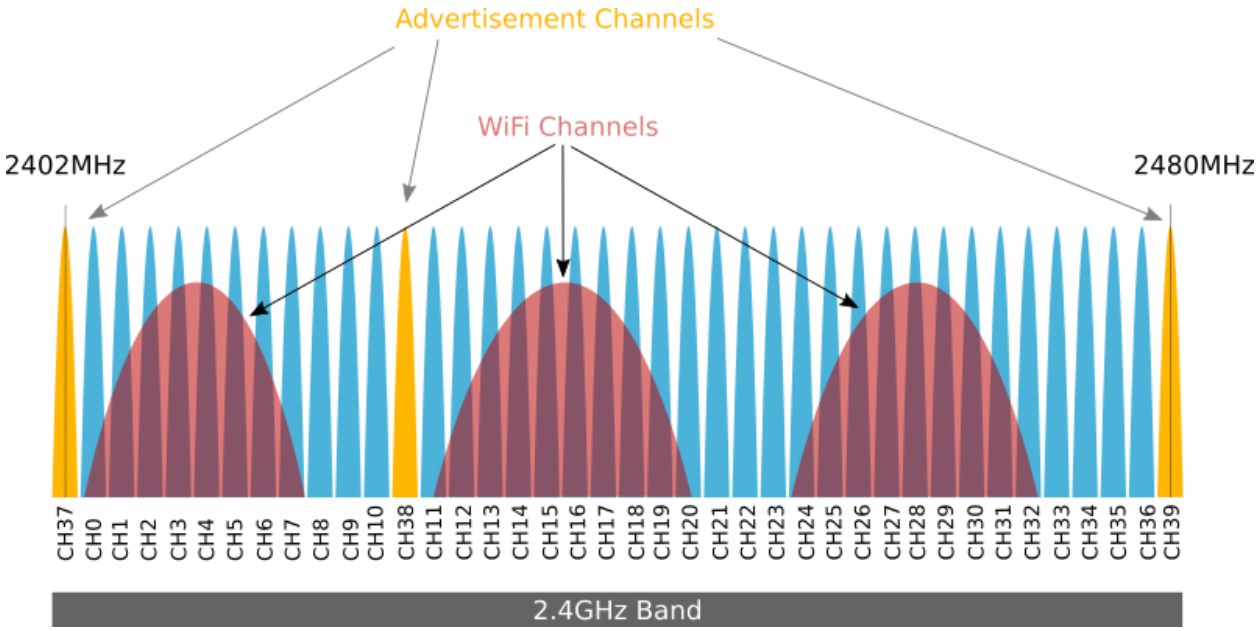

Figure 2-1: The BLE spectrum and its overlap with WiFi common channels. Adapted from [14]

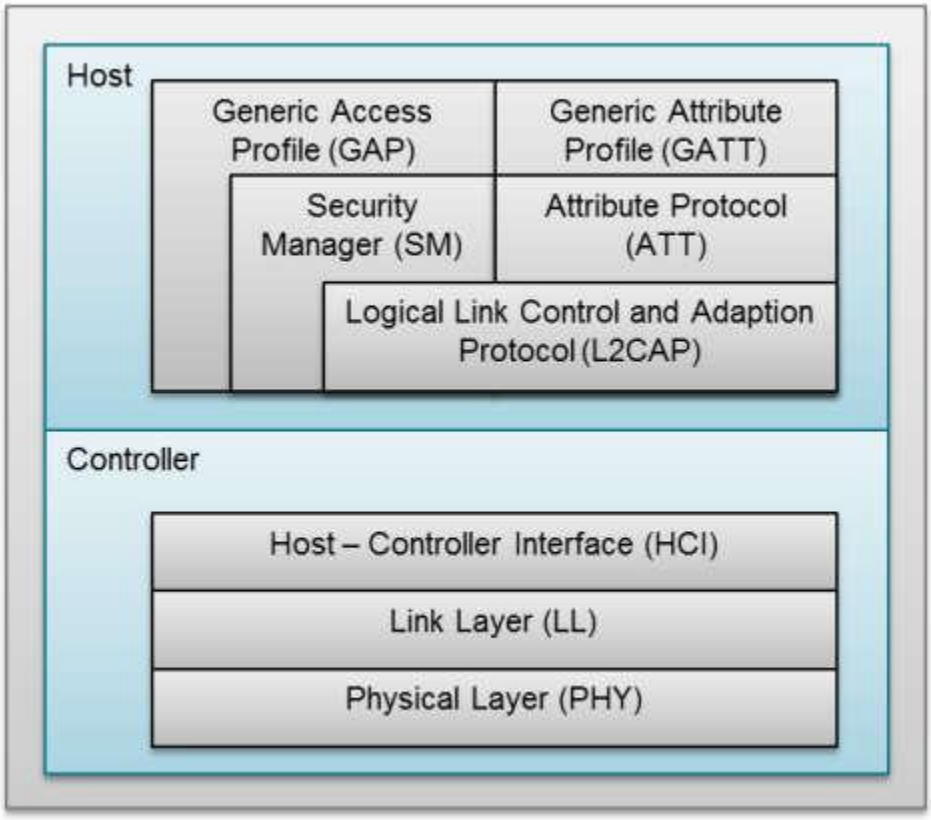

Figure 2-2: Block diagram of the BLE stack. Adapted from [15]

2.3.1 Host subsystem

2.3.1.1 Generic Attribute protocol (GATT)

The GATT layer defines two roles: a server role and a client role. The server maintains a GATT database that is a hierarchy of profiles, services and characteristics, as shown in figure 2.3. A characteristic basically holds some value, for example the heart rate or the sensor battery level. A characteristic may have one or more descriptors that help in interpreting the characteristic's value and specifying how it is delivered to clients. A service includes a set of other services and/or characteristics grouped together. A profile is a collection of services of a certain target application. The database is indexed using 2-byte addresses called handles.

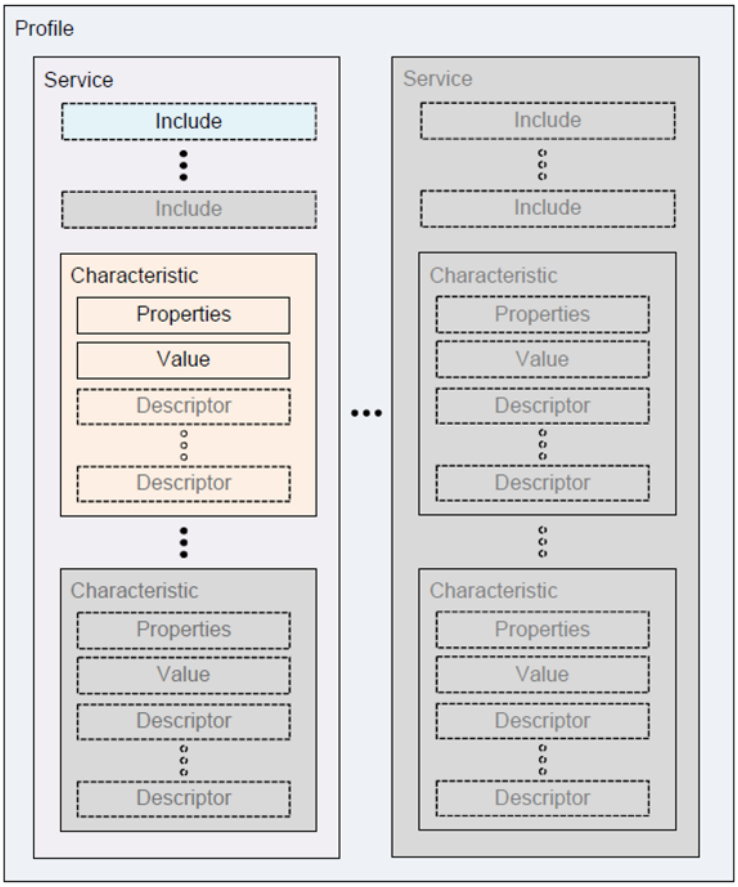

Figure 2-3: An example of a GATT database hierarchy. Adapted from [16]

2.3.1.2 Attribute Protocol (ATT)

The ATT layer handles the communication between a server node and a client node. The ATT layer provides a bearer for the GATT layer, enabling the remote client to discover the server's GATT database and access specific handles by reading or writing them. The client may also ask the server to repetitively initiate a characteristic value delivery to the client. The server may initiate this delivery on a periodic or upon-change bases.

There are two types of server-initiated procedures. The first one is the notification procedure, which does not require a GATT-level acknowledgement (ACK) from the client. The second one is the indication procedure, which requires a GATT-level ACK. The GATT-level ACK is not to be confused with the LL-level ACK which is mandatory for all protocol packets.

The indication procedure is much slower than the notification procedure because the server waits after every indication for the client's GATT-level ACK. To eliminate software-related timing while measuring throughput, notification procedure is typically used.

2.3.1.3 Generic Access Profile (GAP)

The GAP block handles the modes and procedures that are related to the BLE functionality of a BLE device, such as broadcast, observation, discovery, connection and bonding.

2.3.1.4 Security Manager Protocol (SMP)

The SMP block is responsible for peer-to-peer key generation and storage. Section 2.7 discusses the BLE security in more details.

2.3.1.5 Logical Link Control and Adaptation Layer Protocol (L2CAP)

The L2CAP block is responsible for the following:

a. Managing the traffic of different L2CAP channels that correspond to different applications or services, to meet Quality-of-Service (QoS) commitments.
b. Protocol Data Unit (PDU) fragmentation and reassembly.

2.3.2 Controller Subsystem

2.3.2.1 Host Controller Interface (HCI)

The controller subsystem either talks directly to the host subsystem in a full system implementation or through HCI if the controller and host subsystems are implemented separately. Thus HCI is an optional standardized communication layer between both subsystems.

2.3.2.2 Link layer (LL)

The LL defines the connectionless and connection-based states and rules to which the devices shall comply. These states and rules are the heart of the BLE communication protocol and are detailed in section 2.4. The LL also handles packet framing, channel selection, error detection and retransmission.

2.3.2.3 Physical layer (PHY)

The PHY layer sends information over the physical channel. The default PHY is a 1 MHz PHY –also called LE 1M- that uses an uncoded GFSK modulation. Three optional PHYs were introduced in BLE 5: a 1-MHz PHY with coding rates of 1/2 and 1/8 and an uncoded 2-MHz PHY (also called LE 2M).

2.4 Link layer states

The LL state machine is given in figure 2.4 and described in the following subsections.

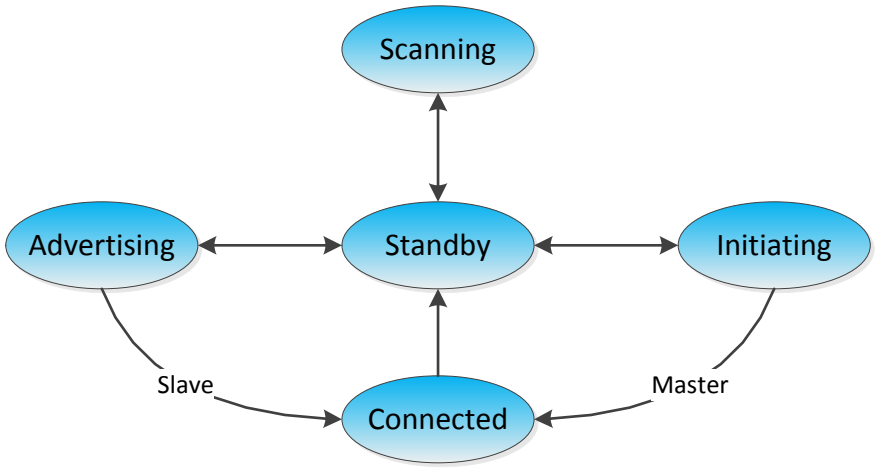

Figure 2-4: LL state machine

2.4.1 Standby state

This is the default state where the device does not do any BLE activity (i.e. it does not transmit nor receive packets).

2.4.2 Advertising state

In this state the device sends advertising packets during advertising events on one or more of the three advertising channels, and may receive scan requests and/or connection requests. Upon receiving a scan request, the device responds with a scan response. Upon receiving a connection request, the device exits the advertising state and enters the connection state in the slave (peripheral) role.

2.4.3 Scanning state

In this state the device listens on one of the advertising channels and scans for advertising packets from other devices, either passively without sending any packets or actively by sending a scan request in response to an advertising packet and waiting for a scan response packet, as shown on figure 2.5.

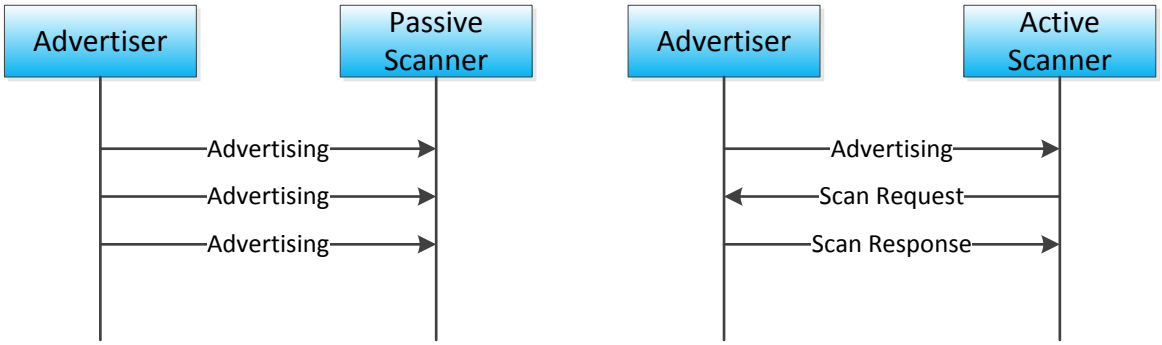

Figure 2-5: Passive scanning (left) versus active scanning (right)

2.4.4 Initiating state

As in the scanning state, the device listens on one of the advertising channels and scans for advertising packets from other devices, but here with the purpose of connecting to one or more target peer devices. When the device receives an advertising packet from the peer device, it responds with a connection request packet, exits the initiating state and enters the connection state in the master (central) role, as shown in figure 2.6.

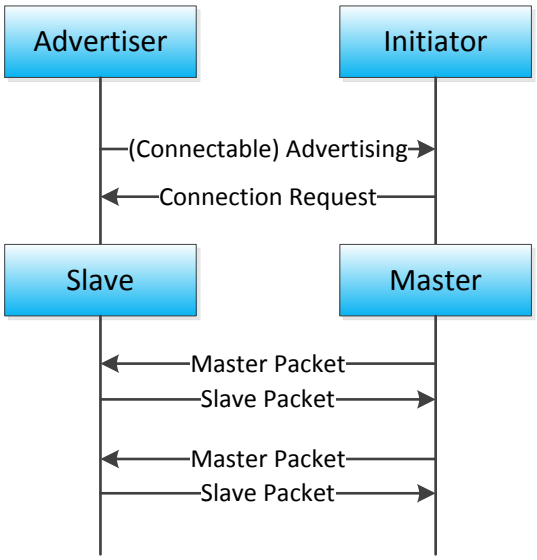

Figure 2-6: Connection Establishment

2.4.5 Connected state

The device in the connected state can be either in the master role or the slave role. Master and slave devices exchange data packets during connection events. A connection event starts by a packet from the master. Whenever the slave receives a packet from the master, the slave shall respond with a packet, thus the data exchange continues in packet pairs. If any or both peers have no data to send, they shall send an empty packet to maintain the connection. Optionally, the master may append any number of additional packet exchanges to the first one within the same connection event, as shown in figure 2.7. Connection events occur periodically every connection interval, whose value is selected by the master between 7.5 ms and 4 seconds, in steps of 1.25 ms. The entire connection event occupies the same frequency channel, and subsequent connection events occupy different frequency channels according to a channel hopping sequence that the master selects and informs the slave about.

While there is no standard limit on the number of packet exchanges per connection event, yet BLE chips -especially the older versions- may set a hard limit due to chip limitations.

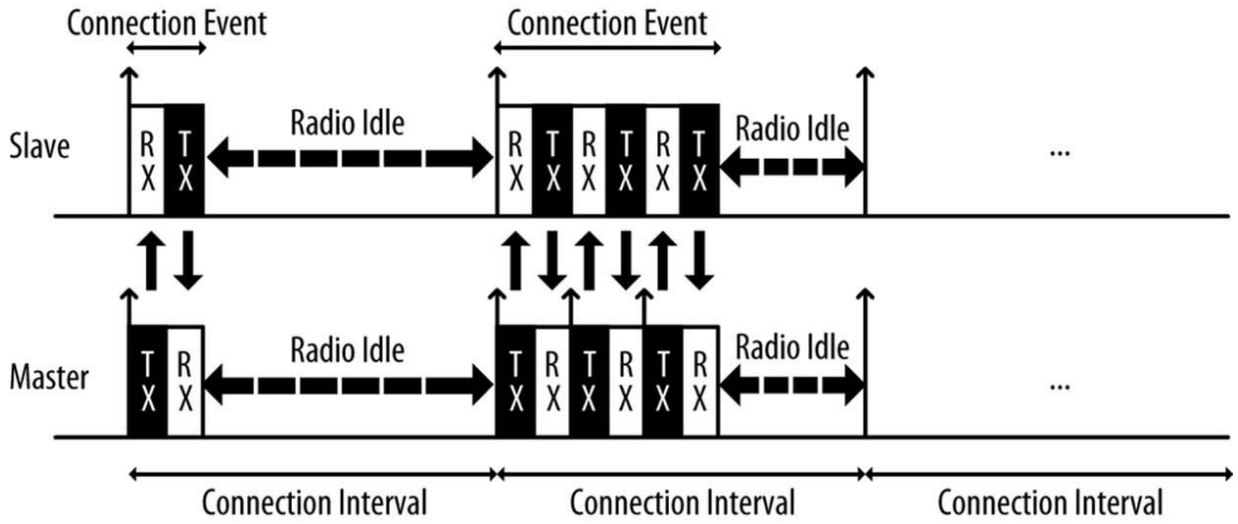

Figure 2-7: Connection Events. Adapted from [17]

2.5 Packet structure

The LL packet structure of uncoded PHY is given in figure 2.8.

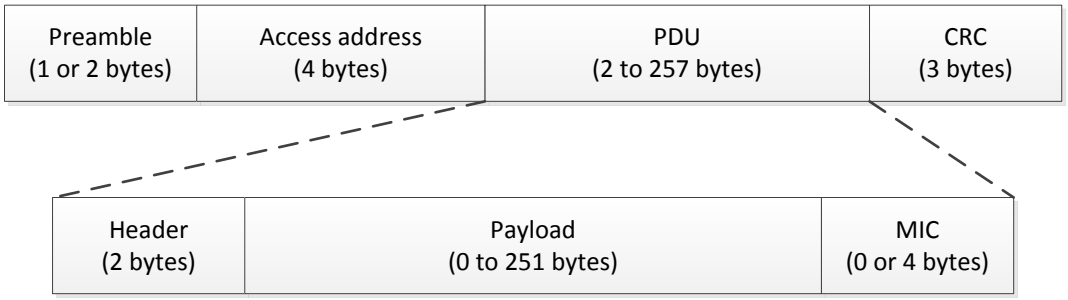

Figure 2-8: LL packet structure

The preamble is a fixed bit pattern whose size is 1 byte for LE 1M and 2 bytes for LE 2M. The access address is a connection identifier value that is assigned by the master. The LL payload is the data-channel protocol data unit (PDU). The PDU starts with a 2-byte header followed by the L2CAP payload. Prior to BLE 4.2, the payload length cannot be more than 27 bytes. Starting from BLE 4.2, the communicating devices can negotiate the maximum length to be more than 27 bytes, up to 251 bytes. The last field is an optional Message Integrity Code (MIC) field.

The L2CAP PDU structure is shown in figure 2.9(a). The L2CAP PDU starts with a 4-byte header followed by an upper-layer payload of maximum length of 65535 bytes. An L2CAP PDU longer than the maximum negotiated LL payload is fragmented at the transmitter side and reassembled at the receiver side.

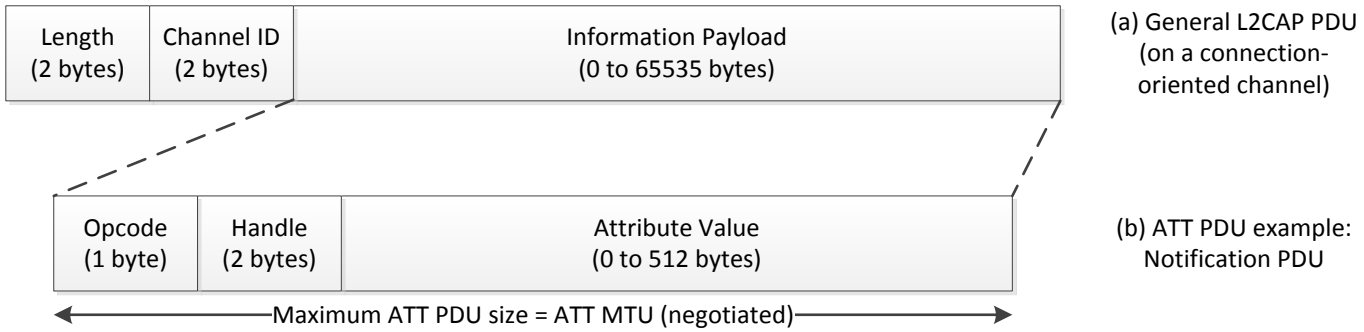

Figure 2-9: L2CAP and ATT packet structure

For GATT-based databases, an L2CAP PDU carries one ATT PDU. The maximum length of an ATT PDU is negotiated between the server and the client and is denoted by the ATT Maximum Transmission Unit (ATT MTU). Although the ATT MTU can be as long as the information payload of an L2CAP packet, yet the common practice is to have it limited by the LL payload size, minus the 4 bytes of L2CAP header, thus avoiding fragmentation at all.

The notification ATT PDU is shown in figure 2.9(b). The maximum length of an attribute value is 512 bytes. If it is required that the notification PDU carries the whole attribute value, then the size of the attribute value should not be greater than (ATT MTU – 3) bytes. Otherwise, only the first (ATT MTU – 3) bytes of the attribute value are sent in the notification PDU and other packet exchanges are needed to read the rest of the attribute value.

In conclusion, if the maximum LL payload length is 251 bytes (for BLE 4.2 and up), then it is very convenient and common to limit ATT MTU to 247 bytes and attribute values (to be notified) to 244 bytes, even if greater values are permitted by the specification.

2.6 BLE versions

Table 2.1 summarizes the BLE standardized versions and the main changes across them.

Table 2-1: BLE Specification Versions

Core Specification version	Publication Date	Main Changes
4.0	Jun 2010	BLE was first adopted.
4.1	Dec 2013	Adding simultaneous multi-role support: a device can simultaneously be advertising, scanning, connected as a slave and/or a master, or any subset of these states.
4.2	Dec 2014	- Adding ECDH security pairing. - Adding length extension feature: maximum length of a LL packet payload on a data channel increased to 251 bytes. - Enhancing privacy: hiding the public BLE address to prevent tracking.
5	Dec 2016	- Adding optional 2 MHz PHY (doubling speed). - Adding optional coded 1 MHz with correction coding rates 1/2 and 1/8, thus quadrupling the communication range. - Adding advertising enhancements and packet length extension.

In the rest of this thesis we mainly focus on the changes that mainly affect throughput and power consumption of BLE devices, namely the data length extension (introduced in version 4.2) and the 2-MHz PHY (introduced in version 5).

2.7 BLE security

Wireless connections are broadcast by nature as any third party can listen to the communication between involved parties. A common example of such third party is a protocol

sniffer. Commercial BLE sniffers are available starting at very low costs and up to thousands of U.S. dollars for highly professional versions.

There are two main classifications of security attacks, namely active attacks and passive attacks. Passive attacks eavesdrop on the wireless channel without intervention with the communicating parties. Active attacks can impersonate one of the communicating parties to the other, or even impersonate the two parties to each other, acting as a Man-In-The-Middle (MITM).

BLE provides different levels of security depending on the capabilities and requirements of both sides entering a connection. The capabilities of a device specify if they have any buttons, number pads or displays. The requirements dictated by a device state for example if a Man-In-The-Middle (MITM) protection is required, if a certain minimum key size is required, or if the keys shall be stored for future reconnections.

The pairing process is the process of exchanging device capabilities, requirements and secrets. At the end of the pairing process, both communicating devices shall have one or more security keys. Prior to BLE 4.2, the pairing process was susceptible to passive attacks because an eavesdropper can easily calculate the shared keys based upon the information that the devices exchange during the pairing process. Starting BLE 4.2, "LE Secure Connections" protocol is introduced, where the key computation involves an Elliptic-Curve-Diffie-Hellman (ECDH) operation which is not vulnerable to passive attacks.

On the other hand, the vulnerability to active attacks depends on the capabilities of the communicating devices. For example, a connection may be required between two embedded devices that do not have any input or output capabilities, thus they cannot check the authenticity of each other, and hence MITM protection cannot be guaranteed. On the contrary, if one of the

devices has a display while the other has a number pad, the former can generate and show a passkey that the user can enter into the latter using its number pad, where the attacker's probability to correctly guess such passkey -which is typically a 6-digit number- is one millionth.

An attacker that does not listen or interfere with the pairing process has a negligible probability to decrypt the messages, once the encrypted session starts.

Different entries in the GATT database can be configured to limit its access to certain security requirements such as authentication, encryption or user authorization. Moreover, security requirements may be configured per access type such as read-only or read-write.

Several BLE security vulnerabilities have been reported, for example the Blueborne attack [18] which is based on bad implementations of the BLE stack, and the pairing vulnerability in [19] where Bluetooth implementations may not sufficiently validate ECDH public keys, allowing a remote attacker to obtain the encryption key. Application-level attacks are also easy to demonstrate, since different smartphone applications can have equal privileges over the phone's BLE module and therefore a malicious application on the phone can impersonate an authentic one, making use of the BLE module as a black box and transmitting and receiving information to the remote device. Such attacks can be life-threatening if used against sensitive applications like BSNs and therefore an additional application-level security layer may be required.

3 Related Work

3.1 BLE technology in literature

While BLE performance is well studied in literature, most of the work is limited to version 4.0. The reason is probably that BLE is a relatively recent technology. After the 4.0 specification was first published in 2010, it took some time until commercial BLE chips became available and stable. One of the widely used early chips supporting BLE 4.0 is CC2540, by Texas Instruments [20], and that's why it is frequently mentioned in the early literature dealing with BLE.

A well-known paper from 2012 [13] provides an overview and evaluation of BLE 4.0 performance in terms of power consumption, throughput, piconet size and latency. The authors of this paper measured the average current consumption for a CC2540 in the slave role, while sending one 20-byte ATT notification per connection event while changing the connection interval, for a transmit power of 0 dBm. The result is shown in figure 3.1. As the connection interval increases, the average current consumption decreases, since the slave remains in sleep mode for a greater fraction of the connection event. In this experiment, the throughput also decreases while increasing the connection interval.

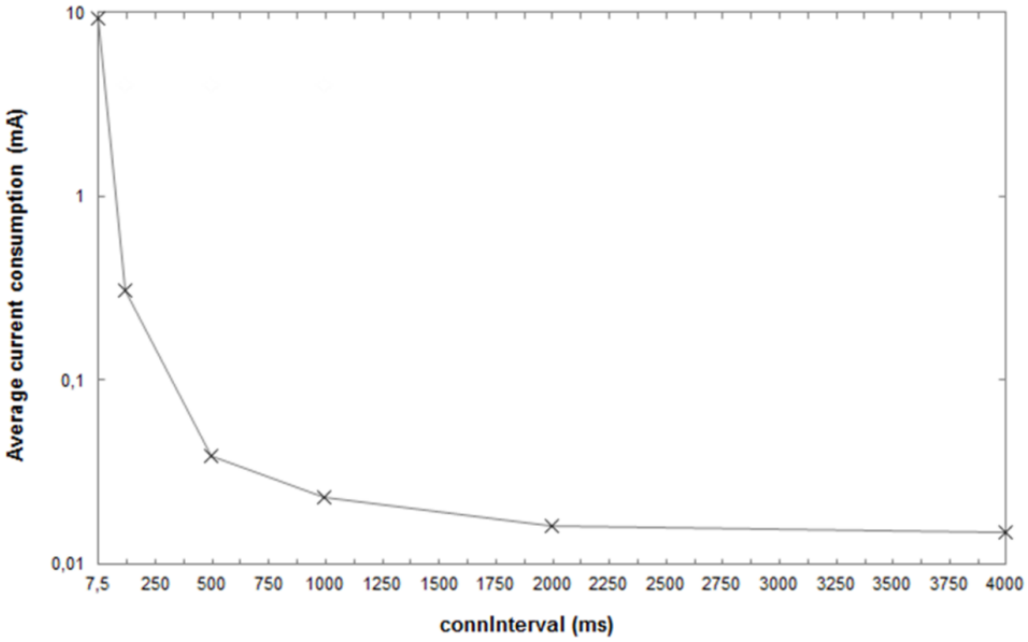

Figure 3-1: Connection interval vs. current consumption when 1 notification is sent per connection event. Retrieved from [13].

To measure throughput, the authors of [13] used two CC2540 devices in the connection state. The CC2540 chip allows 4 notifications per connection interval, and each can carry at most 20 bytes of GATT payload. Theoretically, the maximum transfer rate through notifications is when a connection interval of 7.5 ms is chosen, yielding 20*4 bytes/7.5 ms = 85.33 kbps.

Though conditions for maximum throughput were applied in the previous experiment, yet the maximum GATT throughput that the authors have practically measured is 58.48 kbps. This is because less than four notifications are actually transmitted in most connection events during the experiment. The authors mentioned that the same phenomenon occurs less frequently for connection intervals greater than 7.5 ms, which means that it is probably a chip limitation.

In [21] a more detailed analysis of the maximum number of notifications that the CC2540 can send within a connection event is provided, where the authors concluded that this value depends on the size of the notification payload, being three notifications when the payload length

is 17-20 bytes, four notifications for payload length within 13-16 bytes and five notifications for 10-byte-long payload. Because the number of packet exchanges within a connection interval is limited, the authors concluded that the maximum throughput decreases with increasing the connection interval. This is no longer valid with modern chips because bigger connection intervals can currently handle more and more packet exchanges per connection event.

The variation of power consumption versus the packet length is analyzed in [22] for two commonly-used BLE 4.0 chips, CC2541 by Texas Instruments and nRF51822 by Nordic Semiconductors. The authors conclude that data should be packaged in larger packets in order to reduce the energy consumption of the wireless sensor. This is logical because the payload to protocol overhead ratio increases as the payload length increases while the protocol overhead remains constant.

The authors of [23] discuss the use of BLE in Opportunistic Sensor Data Collection (OSDC), where a sensor node is not part of an infrastructure network, but can rather buffer data and send them intermittently when the sensor is within communication range of a data collector. This requires the sensor to be in the advertising state most of the time to be found by the collector. When the sensor and collector become within communication range, the sensor delivers its data either over advertisement packets or by establishing a connection. OSDC is especially useful for applications that involve logging rather than real-time monitoring of sensor information. The authors divide the advertising and connection intervals into smaller stages and estimate the average current consumption of each small in each stage to develop a current consumption model. This is a common technique that is used throughout the literature and will be also used within this thesis.

The connection-based data collection in [23] is based on transmitting multiple packet exchanges of BLE 4.0 within the same connection event, with each exchange carrying up to 20 bytes of attribute notification. The measured throughput -using Bluegiga's BLE121LR module- is much less than the calculated maximum throughput because of implementation limitations. Nevertheless, their results show that a BLE sensor node transferring around 10 Mbit/day can achieve a lifetime beyond one year on a 230-mAh coin cell battery, based only on wireless activity.

In [10], an interesting comparison of the performance of three BLE chips that were available in the market in 2015 is given. In the experiment, an Android phone is used to connect to each of these chips. Figure 3.2 shows the notification throughput versus power consumption of each of the three chips, using either one packet exchange per connection event (PPCE) or the maximum number of PPCEs permitted by the chip. The connection interval is selected as the minimum value that achieves a given throughput. The cases where multiple PPCEs are transmitted give less power consumption for the same throughput. A throughput up to 64 kbps can be achieved.

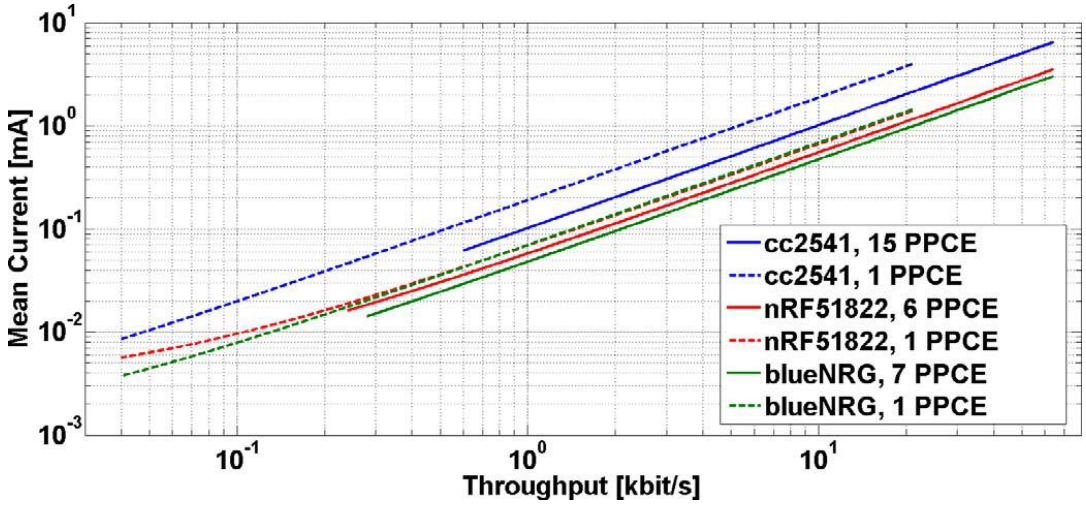

Figure 3-2: Throughput vs mean current for three chips, for minimum and maximum number of PPCE per chip. Retrieved from [10].

A recent paper that describes the BLE protocol stack in details and presents a systematic review of the literature on BLE performance evaluation is [24], published in 2017. However, most of the reviewed literature is limited to BLE 4.0, and the practical throughput that is found in that review is limited to ~100 kbps. Comparing this value to the ~1.3 Mbps that the currently available BLE chips can reach, we can imagine the revolution that took place in BLE chips in about a couple of years.

3.2 Recent literature on Bluetooth 5

Having been released in December 2016, Bluetooth 5 is still somewhat new in the literature. In [25] the new technical features that are included in Bluetooth 5.0 are presented, and their advantages and drawbacks are described. In particular, the new data rates introduced by Bluetooth 5 have been discussed.

In [26], the maximum combined throughput is measured using nRF52840 from Nordic Semiconductors, when both the central and peripheral exchange data packets at almost equal rates. This maximum combined throughput is found to be 1553 kbps, or 1549 kbps on average after considering losses, compared to an average combined throughput of 770 kbps when BLE 4.2 is used (i.e. 2M versus 1M PHYs). In contrast to [26], we focus in this thesis on the throughput in one direction rather than combined throughput in both directions, as BSNs usually consist of sensors sending data to a collector.

The connectionless states of Bluetooth 5 have been studied as well. The BLE discovery process has been assessed in [27] based on new features of Bluetooth 5. Furthermore, the performance of the advertising extension feature of Bluetooth 5 is evaluated in [28]. While acceptable performance is shown, high losses compared to connection state and longer delays

compared to legacy advertising are encountered. The connectionless states are useful for broadcast purposes, however in the rest of this thesis we focus on applications requiring peer-to-peer connection to guarantee packet delivery.

3.3 Use of BLE in medical applications

Integrating BLE in medical applications encountered a slow start. For example, when the authors of [2] conducted a survey on wireless body-area networks for eHealthcare systems and examined 35 research articles on WBSNs between 2010 and 2015, none of the included studies used BLE.

However, there were early attempts to integrate BLE in medical applications. The suitability of BLE as a wireless layer for EKG systems has been studied in [29], which illustrated a system where EKG values are sent over BLE from Bluegiga's DKBLE112 module -powered by CC2540 chip- to a BLE USB dongle that is connected to a PC. The raw payload was 200 bytes per heartbeat, and since this throughput was too high for the DKBLE112 module at this early stage of BLE development, the authors also proposed a compression technique for EKG.

A similar system is proposed by [30], which is again based on the BLE112 module. The system reads EKG and transmits it via BLE to a smartphone. The EKG sensor measures the bio-signal at a rate of 200 times/second, samples the EKG signal using 14-bit A/D conversion and sends 10 data points per packet to further reduce the power consumption. However, the authors did not provide any measurements for the system's power consumption.

An ultra-low power wireless health monitoring system capable of measuring a subject's EKG, respiration, and body temperature is proposed in [31], using BLE112 module. The system outperforms an old system that uses classic Bluetooth 2.1 provided by the same authors, saving

about 75% of power consumption. The battery life estimation is extended to 107 hours, compared to 26 hours of the former design, both using a 3.7 V lithium polymer battery with the capacity of 1100 mAh. The average current consumption of the new system during BLE connection is about 10.25 mA. Four 24-bit ADC channels are used: one for EKG, one for body temperature and two for respiration. The sampling rate is 250 samples per second.

The power consumption breakdown (for a 15-ms connection interval) is shown in table (3.1).

Table 3-1: Power consumption breakdown of the experiment in [31]

Working Mode					Current Consumption (in average)
Standby	Advertising	Connected	Sampling	Transmitting	
√					0.62 mA
√	√				1.1 mA
			√		8.76 mA
	√		√		9.25 mA
		√			9.05 mA
		√	√		9.4 mA
		√	√	√	10.25 mA

The transmit current in [31] at 0 dBm is 27 mA, and the sleep mode consumes only 0.4 uA. It has to be noted that recent BLE chips like Nordic's nRF52840 can transmit at about 6 mA at 0 dBm, which saves up to 78% of transmit power consumption. However, this study is useful to compare BLE and classic Bluetooth in terms of power consumption and also understand the performance and limitations of earlier BLE chips.

In [10], a throughput of 64 kbps is required for 12-lead EKG or 8-channel Electromyography (EMG). This throughput can be obtained using different configurations of connection interval values and the corresponding number of packets per connection event. The power consumption of the different configurations for three different chips is compared in figure

3.3. The graph lines cease when the connection interval cannot be used to achieve the 64-kbps throughput. It is also noticed that the minimum mean current to achieve such throughput is about 2.5 mA using blueNRG module from ST Microelectronics.

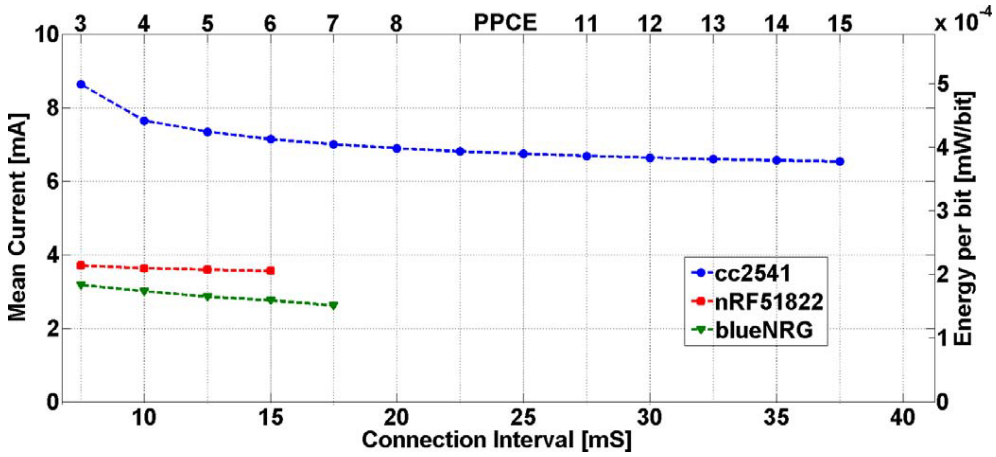

Figure 3-3: Mean current for different BLE configurations of the experiment in [6]

A recent system prototype for temperature and heart rate sensing that claims to use BLE 5 is implemented in [32], where a CC2640R2F module by Texas Instruments is used to send the readings to an Android phone and then to the cloud. However, the authors did not provide any information about throughput or power consumption of their system.

3.4 Comments and conclusion

A summary of the related work is presented, but most of the work is limited to specification version 4.0, where the practical throughput can reach tens of kbps and the average current consumption is in the range of milli-Amperes. With the increased throughput and power saving options of newer BLE versions, together with the production of more enhanced chip implementations, there is still big need to understand and characterize the performance of newer versions of BLE and how they fit in BSN applications.

Finally, one major missing feature in most of the reviewed literature is the possibility of having one attribute packet fragmented over more than one LL PDU. For example, the maximum achievable application throughput using BLE 4.0 is reported as 236.7 kbps [34]. However, this value represents only the case when the application packet is not fragmented. Larger application packets can be fragmented over more than one link layer packet and can achieve throughputs higher than 236.7 kbps. It is usually assumed that the ATT payload occupies only up to 20 bytes out of a 37-byte LL PDU, where the L2CAP and ATT headers take seven bytes and the LL overhead is 10 bytes. In such case the protocol overhead is very large. However, if a long L2CAP PDU is fragmented, only the first fragment will contain the L2CAP and ATT headers, while the rest of fragments will carry up to 27 bytes of ATT payload. If an L2CAP packet is long enough, the average ATT payload per LL PDU can get closer to 27 bytes instead of only 20 bytes. This case is shown in figure 3.4. In this thesis we discuss this case along with different throughput cases and the factors affecting them.

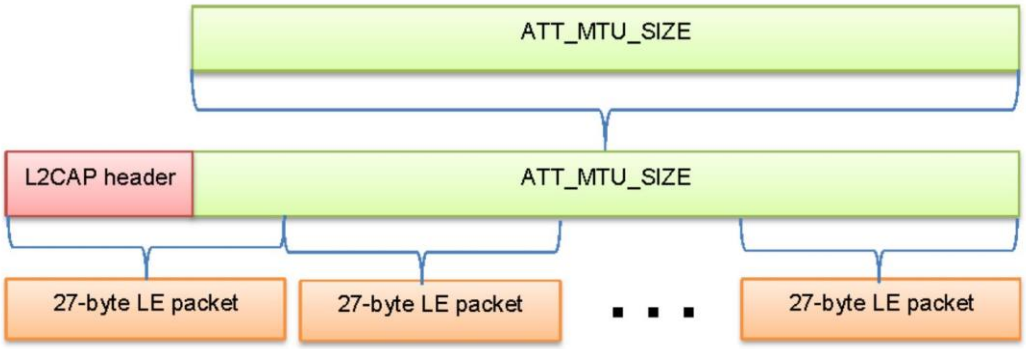

Figure 3-4: L2CAP PDU fragmentation over multiple LL Low-Energy (LE) PDUs. Default LL data length is used (27 bytes). From [33].

4 System Model and Setup

4.1 System components

4.1.1 nRF52840 System-on-Chip (SoC)

The nRF52840 SoC [35] by Nordic Semiconductor is one of the most widely used Bluetooth Low Energy recent solutions that support BLE 5. It includes an ARM Cortex M4 32-bit processor that runs on an on-chip 64 MHz oscillator, a 32-kHz Low-Frequency RC (LFRC) oscillator, a 1-MB flash and a 256-kB RAM. The supply voltage range is 1.7 V-5.5 V, with a typical case of 3V.

The chip also features a low-dropout (LDO) regulator and an optional DC/DC converter with automated low current modes. Since the required voltage level is 1.7 V for the analog parts and 1.2 V for the digital parts, either of the DC/DC converter or the LDO regulator will lower the input supply voltage (which is typically 3 V) to the desired levels. However, the DC/DC converter will use the energy obtained from lowering the voltage in increasing the current and thus it is more power-efficient, except that the DC/DC converter consumes some current by itself, so it is not efficient when the current is already low. To get the best power consumption whenever the DC/DC converter is enabled, the chip automatically uses the DC/DC converter when high current is needed and switches to the LDO regulator otherwise [36].

4.1.2 nRF52840 Preview Development Kit (PDK)

For development purposes, Nordic Semiconductor also provides a development kit, the nRF52840 PDK [37], which is shown in figure 4.1. The nRF52840 DK includes the nRF52840 SoC along with extra components, among which are the following:

1) Interface microcontroller: it is used for loading the firmware to the nRF52840 flash and for debugging using Segger's J-link interface.
2) External High-Frequency Crystal Oscillator (HFXO): this is a 32-MHz crystal oscillator that is crucial for the correct functionality of the BLE radio. Because of its high power consumption, the oscillator is only enabled during BLE events and disabled otherwise.
3) External Low-Frequency (32-kHz) Crystal Oscillator: this is a 32-kHz crystal oscillator that is optionally used as a sleep timer clock instead of the nRF52840 LFRC oscillator. The internal LFRC oscillator frequency is affected by temperature variations so it needs periodic calibration using the HFXO, which increases the power consumption. Therefore, when sleep timers are required it is more efficient to use the LFXO instead of the internal LFRC oscillator.
4) Buttons and LEDs: these are programmable, to provide user interaction.

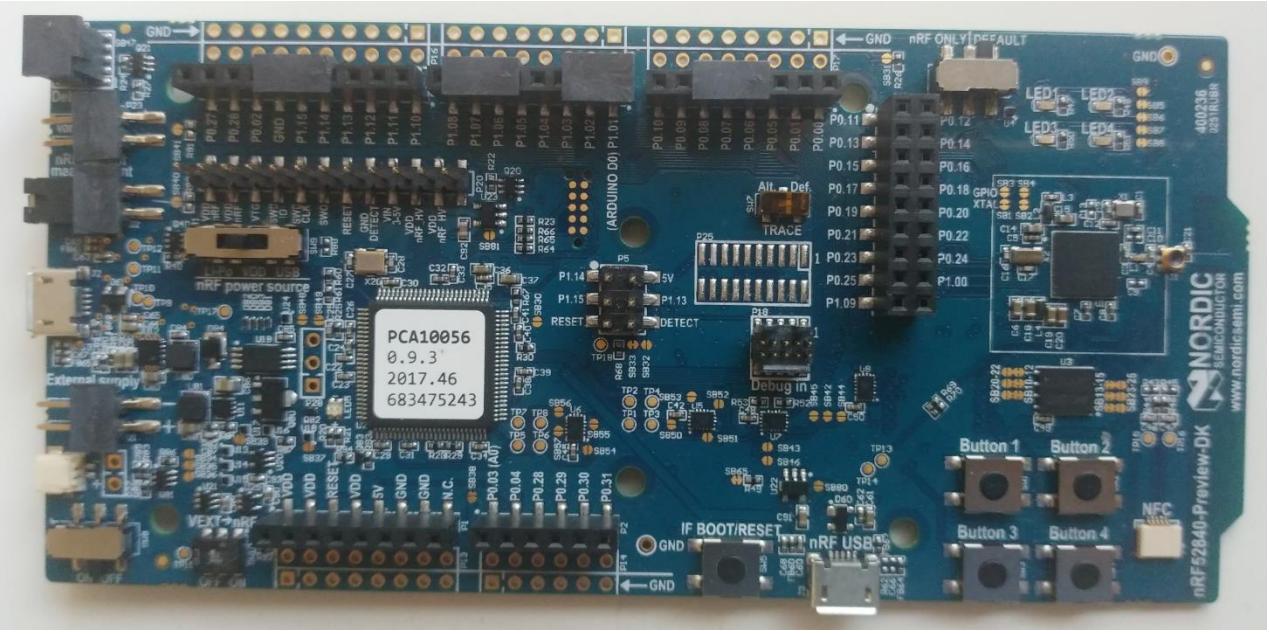

Figure 4-1: nRF52840 PDK

In order to understand the effect of the optional use of the DC/DC converter and LFXO on the current consumption, we ran a simple advertising case (non-connectable, transmit power = 0 dBm, advertising interval = 100 ms, LE 1M, zero payload) and monitored the current consumption as shown in the figures below. In figure 4.2, the DC/DC converter and LFXO were both disabled. In figure 4.3, only the DC/DC converter is enabled, where the average current decreased from about 128 µA to 90 µA. In figure 4.4, the DC/DC converter and LFXO were both enabled, and the average current further decreased to about 78 µA.

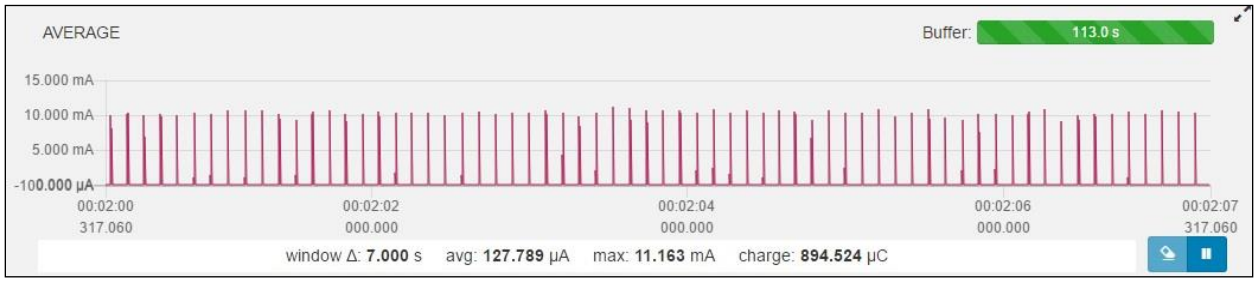

Figure 4-2: Current consumption when DC/DC converter and LFXO are disabled

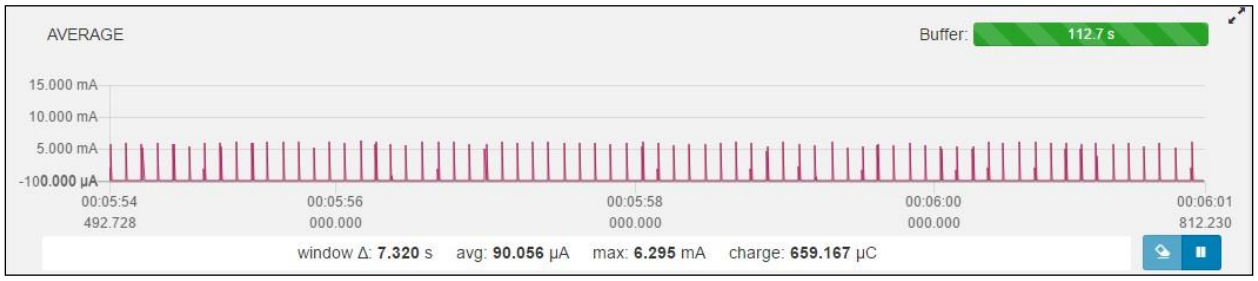

Figure 4-3: Current consumption when DC/DC converter is enabled and LFXO is disabled

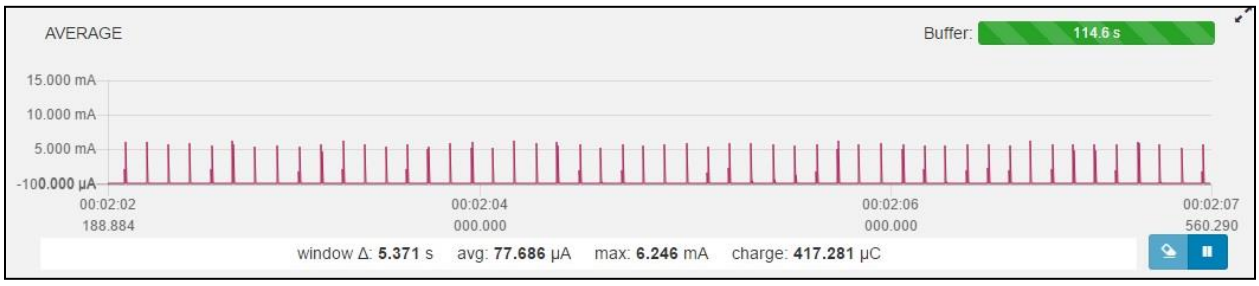

Figure 4-4: Current consumption when DC/DC converter and LFXO are enabled

4.1.3 SoftDevice

The SoftDevice is a BLE stack that is compliant with Nordic's hardware. We use S140 SoftDevice that supports BLE 5 [38].

4.1.4 Power Profiler Kit (PPK)

Another product of Nordic Semiconductor, the Power Profiler Kit (PPK), is a useful tool to measure and log current consumption with a resolution down to 0.2 µA [39], which is a suitable resolution for our target applications. The PPK is shown in figure 4.5.

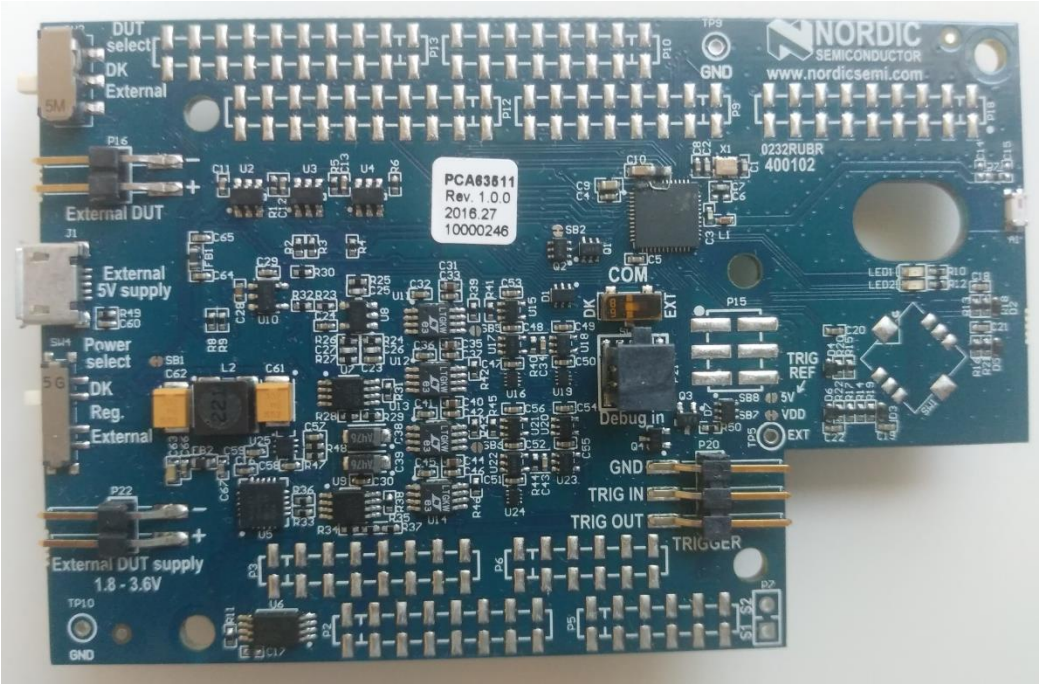

Figure 4-5: Power Profiler Kit

4.1.5 BLE sniffer

We use a BLE USB dongle as a sniffer tool to record the BLE packets on air. However, commercial USB dongles are limited to BLE 4.0 so they are only useful to monitor BLE 4.0 traffic.

4.2 System setup

We used the nRF52840 PDK for both central and peripheral nodes. We implemented a proprietary characteristic with configurable length. This characteristic is notified periodically with configurable time interval between successive notifications. Notifications are queued till the next connection event, when they have the opportunity to be sent. Therefore the choice of the connection interval controls the delay introduced by the wireless protocol. The connection establishment and the start of the notifications procedure are shown in figure 4.6.

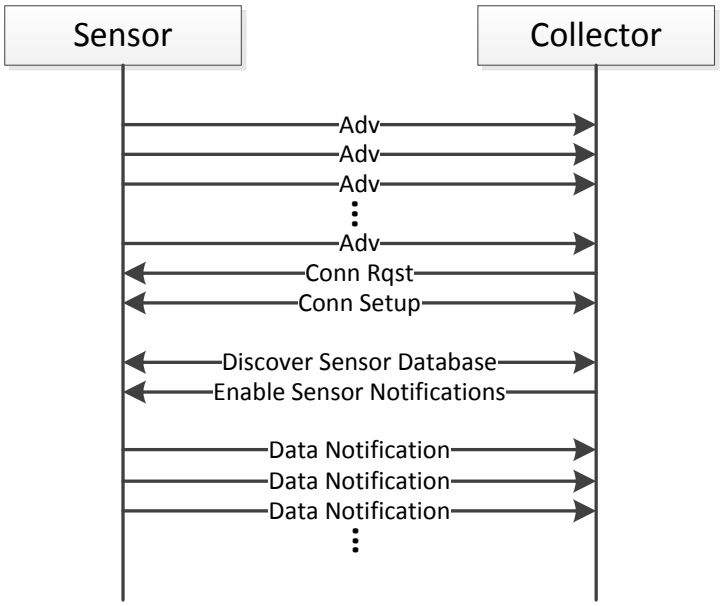

Figure 4-6: Connection establishment and enabling notifications

To measure maximum throughput for different scenarios, the MTU size is set to 247 bytes and the characteristic's maximum length is 244 bytes. The reason of choosing these values is that they correspond to the highest value of MTU size that can be transported over one LL PDU, if the LL data length is set to its maximum value of 251 bytes. Otherwise, the ATT PDU is fragmented to fit the maximum LL data length. The experiments are done in excellent channel conditions i.e. the nodes are very close to each other and the packet error rate is negligible. Using

the PPK, the average current consumption over one minute of measurement is recorded for every case.

To demonstrate a sensor's activity, we let the processor wake up at the desired rate and log data of the specified size each time. Readings are aggregated in notification PDUs and sent in the next available connection event. Sensor current is not considered in the power consumption. To get an estimate of the overall current consumption of the system, the sensor current -as given by the sensor's specification- shall be accounted.

4.3 Throughput model

The application (GATT) throughput of a BLE link depends on the scenario of data exchange. One common scenario of throughput calculation is when the server transmits notification packets while the client transmits LL packets with zero payload length. Consider a GATT notification packet that carries L_{char} bytes of a characteristic value. Considering the notification ATT PDU and L2CAP PDU overheads as shown in figure 2.9, the L2CAP PDU length of such notification packet is $L_{L2CAP} = (L_{char} + 7)$ bytes.

Consider the simple case when the L2CAP PDU fits in one LL PDU i.e. L_{L2CAP} is less than or equal to maximum LL data length. Let L_S and L_C be the total LL length in bytes of the server's packet and client's packet respectively, then

$$L_S = L_{L2CAP} + \text{LL overhead} \tag{4.1a}$$

$$L_C = \text{LL overhead} \tag{4.1b}$$

where LL overhead is 10 bytes for LE 1M and 11 bytes for LE 2M (as in figure 2.8) and the client's packet has zero-length LL payload. The time interval $T_{exchange}$ that is required for one packet exchange is given by

$$T_{exchange} = \frac{(L_S + L_C) * 8}{R} + (2 * IFS) \tag{4.2}$$

where *R* is the PHY rate and *IFS* is the inter-frame spacing between two successive LL packets and is equal to 150 µs.

An upper bound for the GATT throughput can be easily derived, assuming continuous transmission of packet exchanges, as follows:

$$\text{Throughput upper bound} = \frac{L_{char} * 8}{T_{exchange}} \tag{4.3}$$

To achieve this upper bound, the used connection interval value *CI* must be an integer multiple of $T_{exchange}$, and the chip must support sending up to (CI / $T_{exchange}$) packet exchanges per connection event. In the absence of packet errors, the actual throughput can be calculated as

$$\text{Throughput} = \frac{L_{char} * 8 * PPCE}{CI} \tag{4.4}$$

where *PPCE* is the number of packet exchanges per connection event that the chip can transmit when a connection interval of *CI* is used and each packet exchange takes $T_{exchange}$ seconds. *PPCE* is upper-bounded by the value (CI / $T_{exchange}$) that denotes the continuous transmission case.

Similar analysis can be done for the case when the L2CAP PDU is fragmented across more than one LL PDU. However, these LL PDUs may not be equal in length and therefore the amount of transmitted bytes may differ from one connection event to another. The average throughput can still be calculated using a modified version of the previous equations as follows. If the L2CAP PDU is fragmented over *N* LL packets, such that

$$N = \left\lceil \frac{L_{L2CAP}}{\text{max LL data length}} \right\rceil \tag{4.5}$$

where ⌈ ⌉ denotes the ceiling function. Each LL packet carries on average a portion of length $L_{L2CAP,avg}$ of the L2CAP PDU and a portion of length $L_{char,avg}$ of the characteristic value, such that

$$L_{L2CAP,avg} = \frac{L_{L2CAP}}{N} \tag{4.6a}$$

$$L_{char,avg} = \frac{L_{char}}{N} \tag{4.6b}$$

then L_{L2CAP} and L_{char} are substituted with $L_{L2CAP,avg}$ and $L_{char,avg}$ respectively in (4.1)-(4.4). A local maximum throughput value occurs whenever L_{L2CAP} is a multiple of the max LL data length, because this ensures the usage of maximal-length LL PDUs and thus minimizes the ratio of the LL overhead to the payload.

4.4 Current consumption model

The power consumption of nRF52840 in the connection state can be profiled according to the following model. If there is only one packet exchange within a connection event, then the connection interval duration can be divided into nine stages [38], as shown in figure 4.7, namely: (a) pre-processing (b) standby and crystal oscillator ramp (c) standby (d) radio startup (e) radio reception (for peripheral) or transmission (for central) (f) radio switch (g) radio transmission (for peripheral) or reception (for central) (h) post-processing (i) idle time. If multiple packet exchanges are included in the connection event, then the sequence of stages (f), (e), (f) and (g) are repeated for the number of additional packet exchanges, before the final stages (h) and (i).

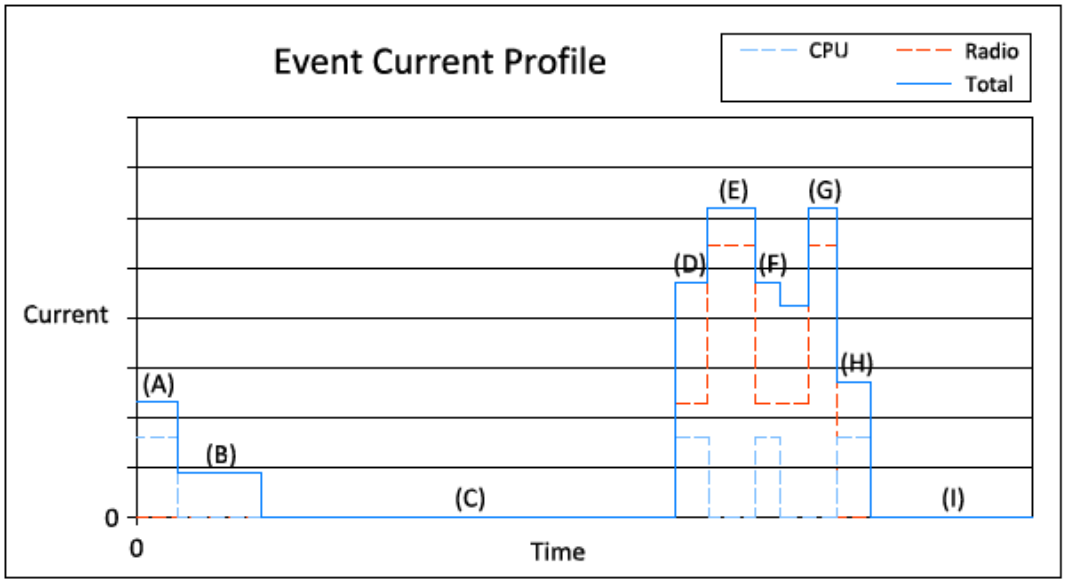

Figure 4-7: Current Profile Model for Peripheral Connection. Retrieved from [38]

Figure 4.8 shows the actual power profile captured by the power profiler kit in the peripheral connection state, for a single packet exchange per connection event. The stages above can be spotted.

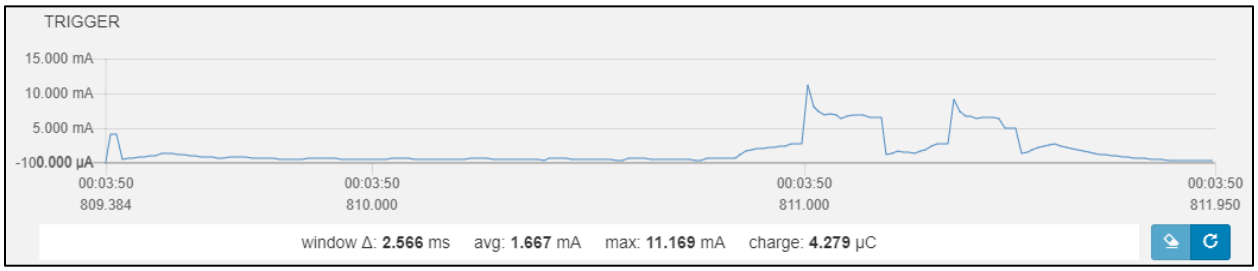

Figure 4-8: Actual Current Profile Measurement for Peripheral Connection

The estimated power consumption can be calculated as:

$$\text{Average Current} = \frac{\sum\left(T_{stage} * I_{stage}\right)}{CI} \tag{4.7}$$

where T_{stage} is the duration of a stage, I_{stage} is the average current consumption during that stage, CI is the connection interval value and the summation is carried over all the stages of the

connection interval duration. Approximate values for the duration and average current consumptions of each stage are given by the manufacturer for an older chip version [40], for a peripheral that transmits only one LL packet per connection event, with up to 27 bytes of LL payload. These values can be extrapolated to match the case where the connection event includes more than one packet exchange and for LL data length up to 251 bytes. We also updated average current consumption values for transmission and reception cases in the model to match the documentation of the latest version of the chip [35]. This gives a good approximate model for the current consumption of nRF52840. We tested this model against the measurements recorded by our setup and the error is within 5% of the measured values. Table 4.1 shows an example for the power profile of different stages of a peripheral connection transmitting at 0 dB, where the transmission and reception durations depend on the packets length.

Table 4-1: Average current consumption and duration of event stages

Stage	Duration (µs)	Current (mA)
pre-processing	61	3.5
ramp	440	1.5
standby	1004	0.4
start	133	2.8
reception	Variable	5.8
switch	102	3.8
transmission	Variable	5.6
post-processing	205	3.2
idle	Variable	0.002

5 Results and Discussion

5.1 Introduction

In this chapter we present our results. To simplify the notations throughout this chapter, we use BLE 4.0 to denote the case of LE 1M with 27-byte LL data length, and we use BLE 4.2 and BLE 5 to denote the cases of 251-byte LL data length for LE 1M and LE 2M respectively. Excellent channel conditions with negligible packet error rate are always assumed.

5.2 BLE throughput

5.1.1 Maximum application throughput

We start with investigating the maximum application throughput that can be achieved using server's notifications. First, we calculate the throughput upper-bound using the model of section 4.3, for notification length = 244 bytes (corresponding to MTU = 247 and LL PDU length = 251 bytes). For better illustration, the analysis and time required to transmit such notification is shown in figure 5.1. Using BLE 5 the 244 bytes of application data are sent in 1392 µs, while BLE 4.2 takes 2468 µs and BLE 4.0 takes 6608 µs to transmit the same size of application data. This gives a throughput upper bound of 1402.3 kbps, 790.9 kbps and 295.4 kbps for BLE 5, BLE 4.2 and BLE 4.0 respectively. This shows that the upper bound has increased by about 168% by increasing the LL data length from 27 bytes to 251 bytes, and by about 77% by doubling the PHY rate.

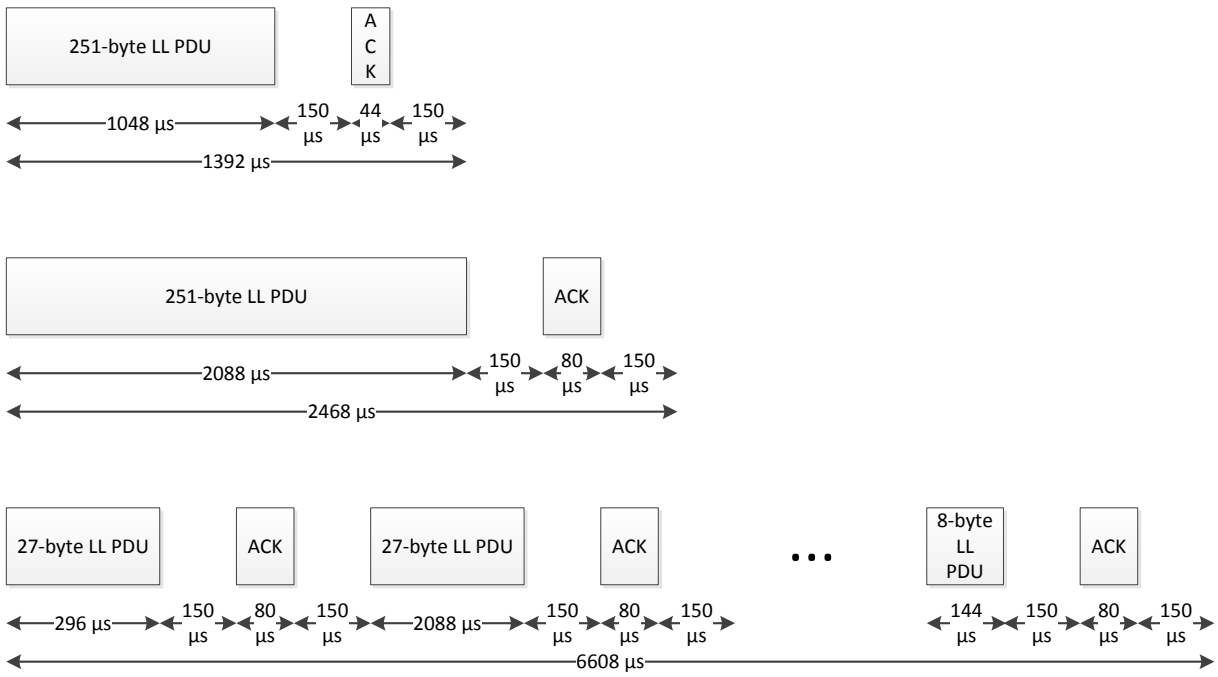

Figure 5-1: LL packet exchanges required to transmit a 244-byte notification for BLE 5 (top), BLE 4.2 (middle) and BLE 4.0 (bottom)

The throughput upper bound may not be reached, because only integer number of packets exchanges shall be transmitted in every connection event. The maximum theoretical throughput across different connection interval (CI) values and BLE version features is shown in figure 5.2, again for 244-byte notification payload (MTU = 247 bytes). The throughput varies with the choice of the CI value depending on how many packet exchanges can fit per event. The figure also shows the upper bounds previously calculated as dashed lines. The throughput approaches the upper bounds whenever the connection event fits a number of packet exchanges such that the unused time at the end of a connection event is minimal.

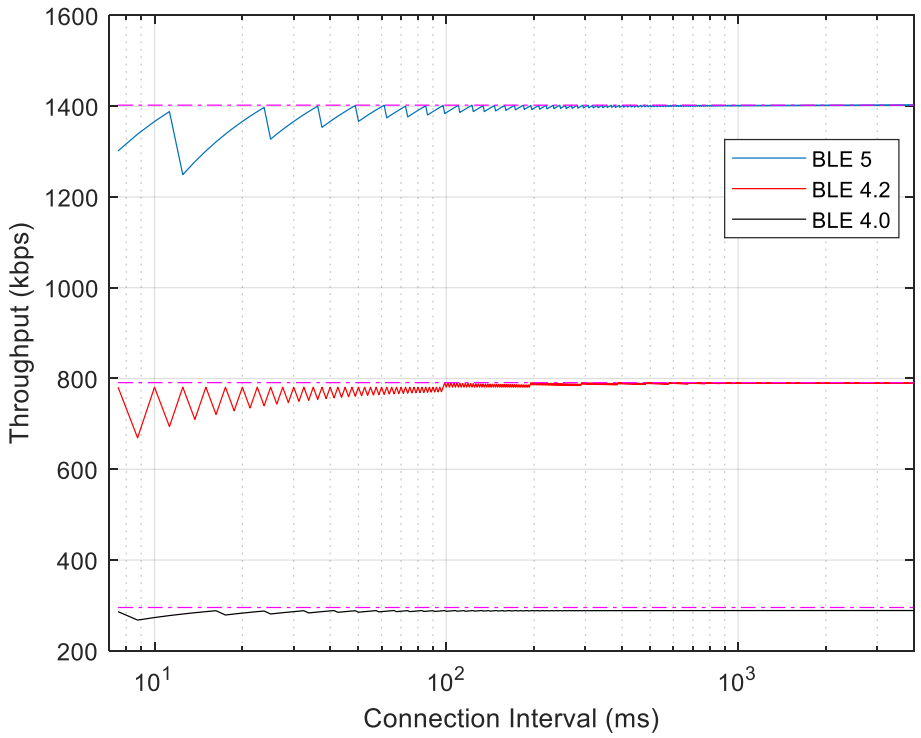

Figure 5-2: Maximum theoretical throughput across different connection interval (CI) values and BLE versions for 244-byte notification payload (MTU = 247 bytes)

It has to be mentioned that the application throughput and the previously calculated upper bound values depend on the relation between the MTU size and maximum LL data length as well as how the L2CAP packet is fragmented. The MTU size we selected is suboptimal for BLE 4.0 when the maximum LL data length is 27 bytes, because the last LL packet of every fragmented L2CAP packet will not be of maximal LL length. In figure 5.1, the last LL packet in BLE 4.0 case carries only 8 bytes of payload. We show in figure 5.3 the maximum theoretical throughput across different CI and MTU values for BLE 4.0. The throughput in the first two cases is greater than the maximum achievable throughput of 236.7 kbps reported in [34] because the MTU size is limited to 23 bytes in [34]. While we unify the MTU size as 247 bytes in our experiments, we

shall remind the reader that the MTU size is not limited by the BLE specification and can be increased even for BLE 4.0 implementations.

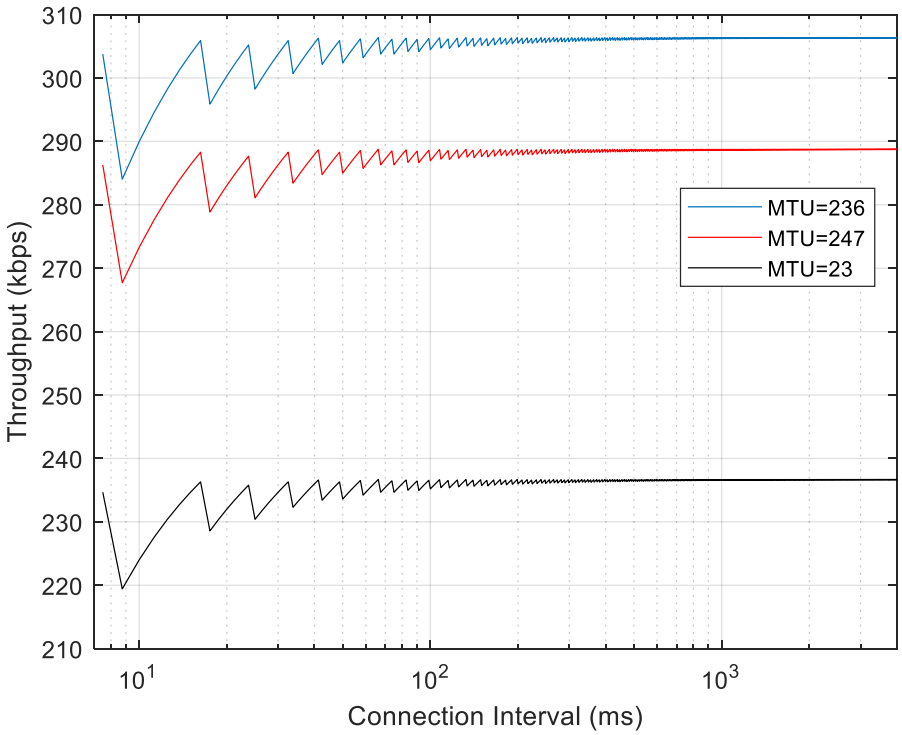

Figure 5-3: Maximum theoretical throughput across different CI and MTU values for BLE 4.0

For illustration purposes, the LL throughput, rather than the application throughput, may offer a more accurate comparison of the throughput of different PHY rates and LL data lengths, regardless of the MTU size and maximum LL data length. Moreover, an upper bound of LL throughput that is independent of the choice of CI can be calculated by using the maximum LL data length instead of L_{char} in equation (4.3). This gives the results in Table 5.1.

Table 5-1: LL throughput upper bound for different BLE versions

BLE version	PHY	Max LL data length (bytes)	LL throughput upper-bound (kbps)
4.0	**1M**	**27**	**319.5**

BLE version	PHY	Max LL data length (bytes)	LL throughput upper-bound (kbps)
4.2	**1M**	**251**	**813.6**
5	**2M**	**251**	**1442.5**

This shows that the LL throughput has increased by about 155% by increasing the LL data length from 27 bytes to 251 bytes, and by about 77% by doubling the PHY rate. The improvement in the application throughput we examined earlier is also close to these percentages.

5.1.2 Measured throughput

The measured throughput across different CI values and BLE version features is shown in figure 5.4, again for 244-byte notification payload (MTU = 247 bytes). We are not interested in CI values above 1 s because these impose high latency that is not suitable for medical streaming applications.

It has been discovered that commercial BLE chips cannot typically use the whole duration of the connection event in exchanging packets but require some idle duration between connection events. Such duration may be used by the chip to prepare for the next connection event. This idle duration is significant at low CI values, affecting the throughput of such values at the left side of figure 5.4. The useful and idle durations of a connection event for low CI values are shown in figure 5.5, for BLE 5.

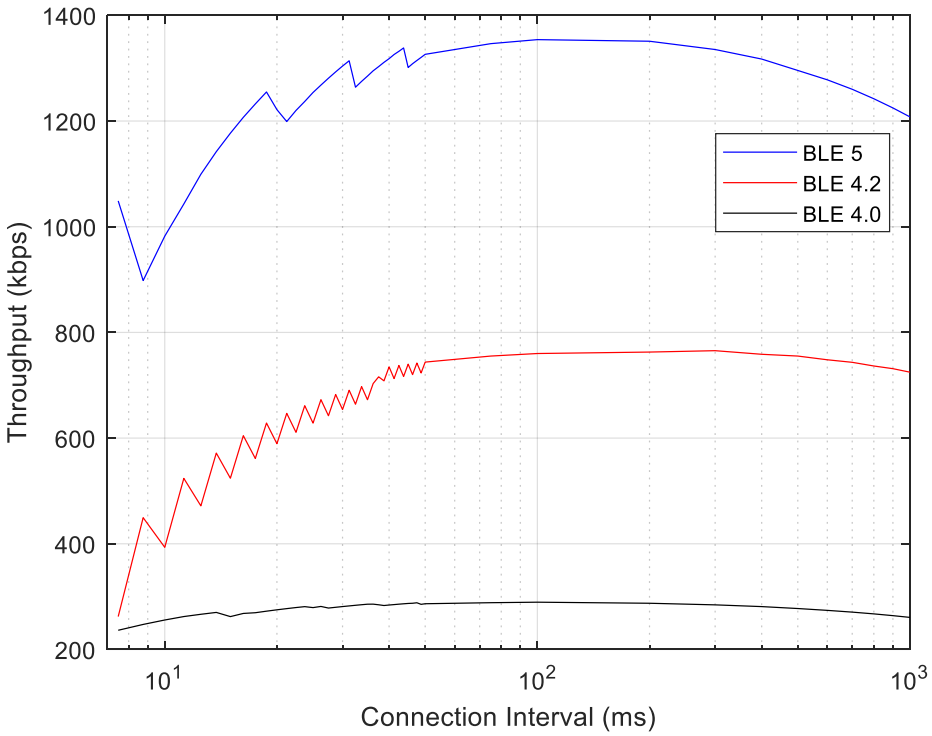

Figure 5-4: Measured throughput across different CI values and BLE versions for 244-byte notification payload (MTU = 247 bytes)

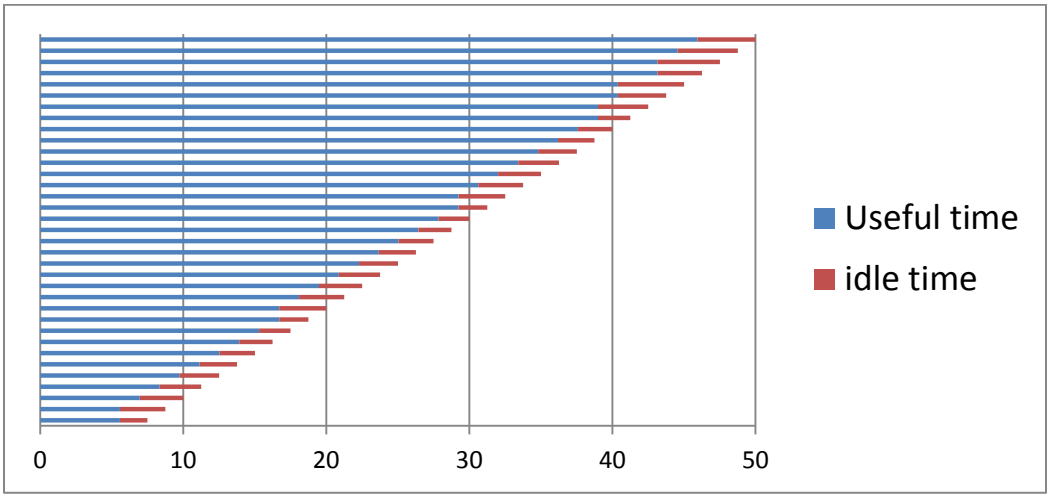

Figure 5-5: Useful and idle durations of a connection event for low CI values for BLE 5

The idle duration also depends on the maximum LL data packet length, because the chip may be uncertain whether the packet exchange will fit in the remaining duration of the event or not. For this reason, the idle durations in the BLE 4.0 case are smaller than other cases and the throughput variation at low CI values for the BLE 4.0 case is not huge. The useful and idle durations of a connection event for BLE 4.0 are shown in figure 5.6.

Additionally, we notice that the throughput decreases at large CI values, which could be because of chip speed limitations. But anyway there is no gain of working at high throughput at a large CI value, because this case will be highly limited by retransmissions if the channel conditions worsen. Thus, there is a CI range in between the effects of low CI and high CI values where the throughput is good. Empirically, this range is between 50 ms and 400 ms.

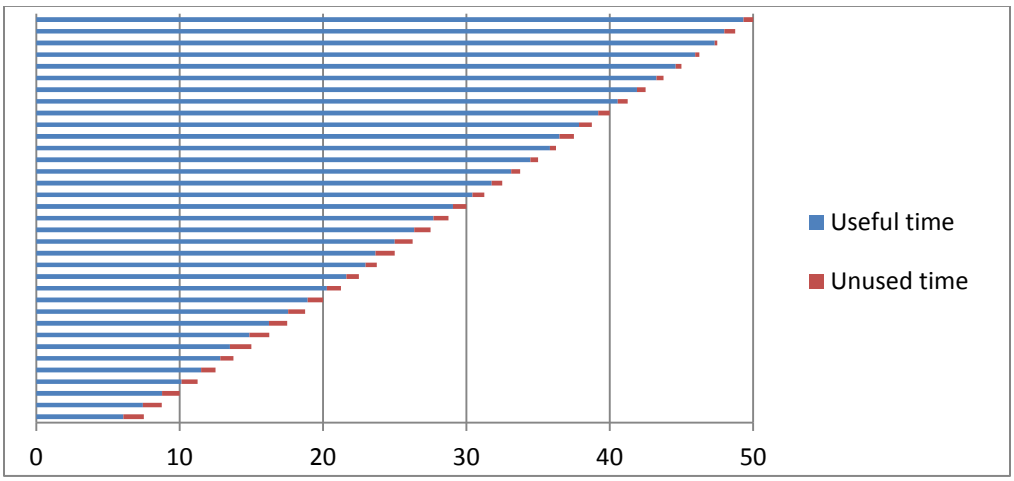

Figure 5-6: Useful and idle durations of a connection event for BLE 4.0

We end this section by comparing the measured throughput numbers using our setup to the values provided by the manufacturer, if available [38], and the upper bound values that are calculated in the previous section. Table 5.2 shows the comparison. The difference between the measured throughput results and the values reported by the manufacturer is less than 6%. Additionally, our measurements achieve 94%-97% of the throughput upper bound.

Table 5-2: Application throughput for different BLE parameters

PHY	CI (ms)	Max LL data length (bytes)	Our measured throughput (kbps)	Throughput reported by manufacturer (kbps)	Throughput upper-bound (kbps)
1M	**50**	**27**	**286.6**	**N/A**	**295.4**
1M	**400**	**27**	**281.2**	**N/A**	
1M	**50**	**251**	**743.9**	**702.8**	**790.9**
1M	**400**	**251**	**758.7**	**771.1**	
2M	**50**	**251**	**1326.2**	**1327.5**	**1402.3**
2M	**400**	**251**	**1317.3**	**1376.2**	

5.3 BLE current consumption

5.3.1 Measured current consumption of a sensor node

In this section, we investigate the current consumption of a BLE sensor node that is working in the peripheral state and acting in the server's role, sending notifications to the client to achieve the required throughput. We start by fixing the CI at 1 s and the transmit power at 0 dBm and determining the least current consumption that achieves every throughput value. Figure 5.7 shows this case for different BLE versions. To highlight different regions of the relation we also plot the same measurements on log-log scale in figure 5.8.

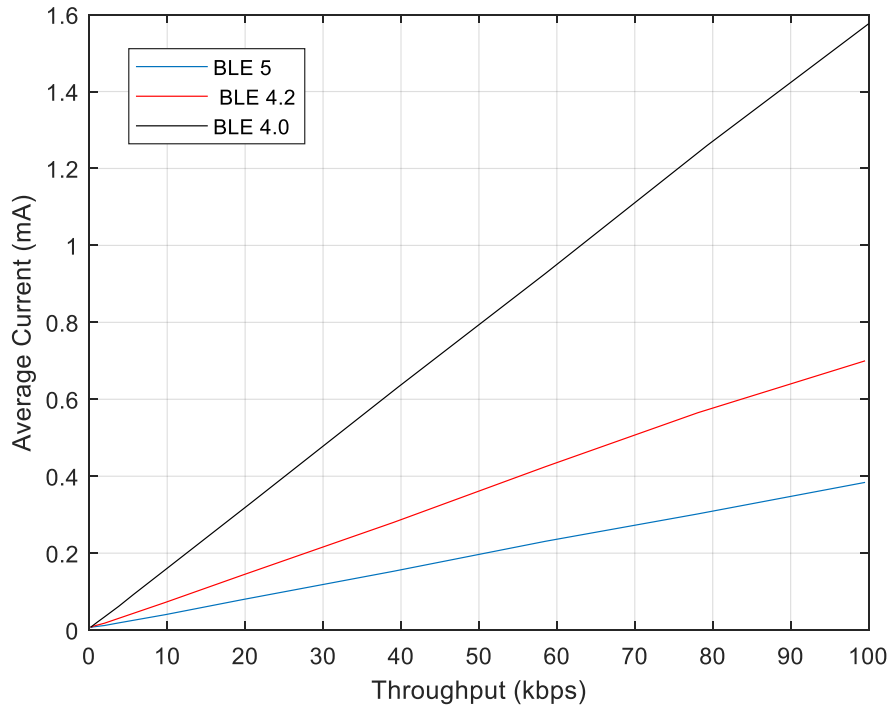

Figure 5-7: Throughput vs. current consumption for different BLE versions, for CI = 1 s

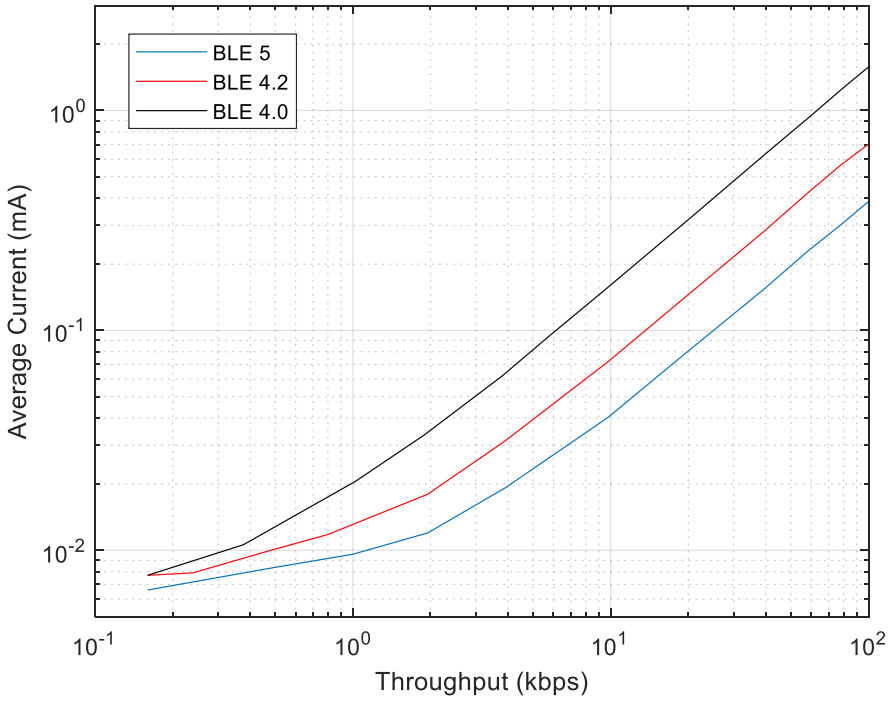

Figure 5-8: Throughput vs. current consumption for different BLE versions, for CI = 1 s (log-log scale)

The measurements show that when the throughput is sufficiently high (i.e. higher than approximately 2 kbps), we find an almost linear relationship between the throughput and average current, because the total power consumption is dominated by the radio transmission power which is proportional to the throughput. There is an average reduction of 55% of power consumption when the LL data length is extended from 27 bytes to 251 bytes, for the same throughput. Another 40% of power reduction comes when LE 2M is used, compared to LE 1M. This can be roughly deduced from figure 5.1 regarding the connection event time savings that are encountered when the same amount of data is transmitted using different BLE parameters. The tradeoff of the increased packet length is the increase in the probability of errors and collisions. Since we are using uncoded PHYs, a packet that encounters an error has to be retransmitted. The extra cost of LE 2M is the increased signal bandwidth.

5.3.2 Battery lifetime expectancy

Given the average current for different throughput values, we can estimate the battery lifetime for a coin cell battery. The battery lifetime estimate (in hours) can be calculated as the quotient of the division of the total capacity of the battery (in mAh) by the average current consumption (in mA). Figure 5.9 shows the battery lifetime estimate for the case of section 5.3.1 on a 100-mAh battery.

For a constant throughput of 10 kbps -which is considered a moderate throughput value for medical streaming applications- a BLE 5 node can last for about 15 weeks on a 100-mAh battery, compared to about 8 weeks on BLE 4.2 and about 4 weeks on BLE 4.0. These numbers consider only the BLE activity, away from the sensing and data processing current which could be estimated using the sensor's datasheet. These results can also be scaled to fit a required

lifetime duration or a certain battery capacity. For example, a 200-mAh coin cell battery is expected to live for double the lifetime durations of figure 5.9.

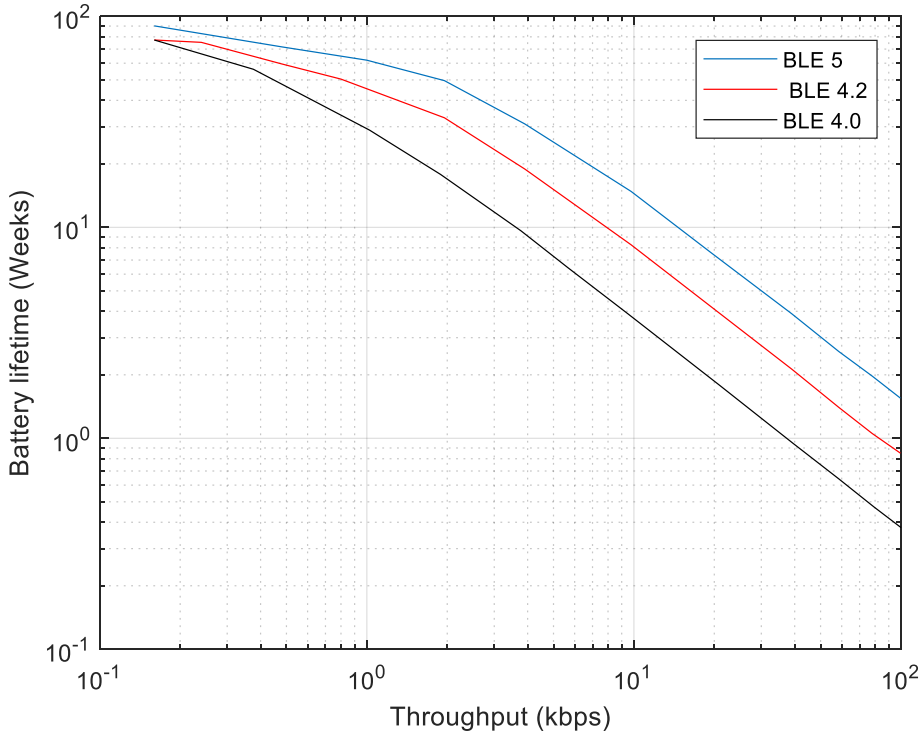

Figure 5-9: Battery lifetime estimate for a 100-mAh battery

5.3.3 Effect of transmit power

BLE permits a wide range of transmit power values. Version 5 of the core specification allows a maximum transmit power up to +20 dBm, while earlier versions allow up to +10 dBm. nRF52840 SoC allows a set of transmit power values between +8 dBm and -40 dBm, with 0 dBm as the default. For many BLE applications, values above 0 dBm are typically considered high-power, values around -8 dBm are considered moderate-power and values below -15 dBm are considered low-power. If the BLE nodes are close enough to each other (e.g. several medical patches within the same body area talking to each other), transmit power values well below 0

dBm may be used without affecting the performance. Obviously, decreasing the transmit power lowers the power consumption, at the expense of a reduced communication range.

Figure 5.10 shows the power consumption versus throughput for the previous BLE 5 case of section 5.3.1, for transmit power values of 0 dBm, -8 dBm and -16 dBm. Power savings going from 0 dBm to –8 dBm are about 15-20%, and less saving percentages are achieved going from -8 dBm to -16 dBm because as the transmit power faints, other power consumption factors dominate. More power savings are achieved at high throughputs, when the transmit durations increase.

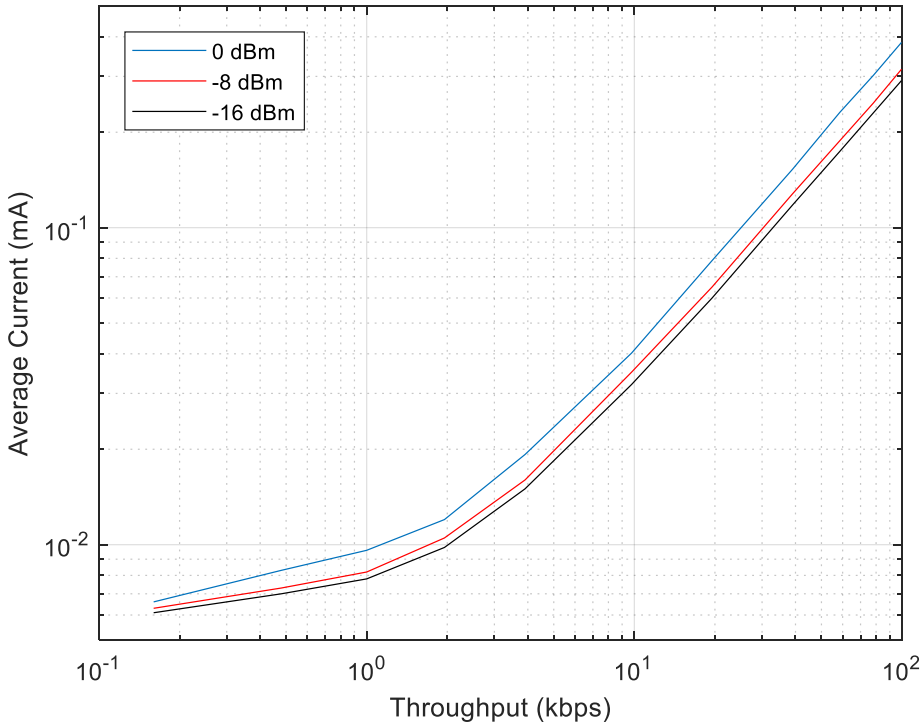

Figure 5-10: Power consumption versus throughput for BLE 5 for different transmit power values

5.3.4 Client's power consumption

So far only the server's power consumption is considered. While the client device is usually a high-power device such as a smartphone, there are use cases that require two or more low-power devices to communicate directly with each other, forming some sort of a mesh network. This raises the importance of measuring the current consumption on the client's side, which is usually the central (though this is not mandatory). Figure 5.11 shows the central (client) and peripheral (server) current consumption versus the required throughput for BLE 5 case with 0 dBm transmit power. Current consumption of both nodes is comparable in the connection state because the transmit power at 0 dBm and the receive power are close to each other. At low throughputs, the peripheral's power consumption is higher because the peripheral has to wake up for longer time than the central to account for the uncertainty of the central's packet timing, while as the throughput increases the radio time becomes dominant.

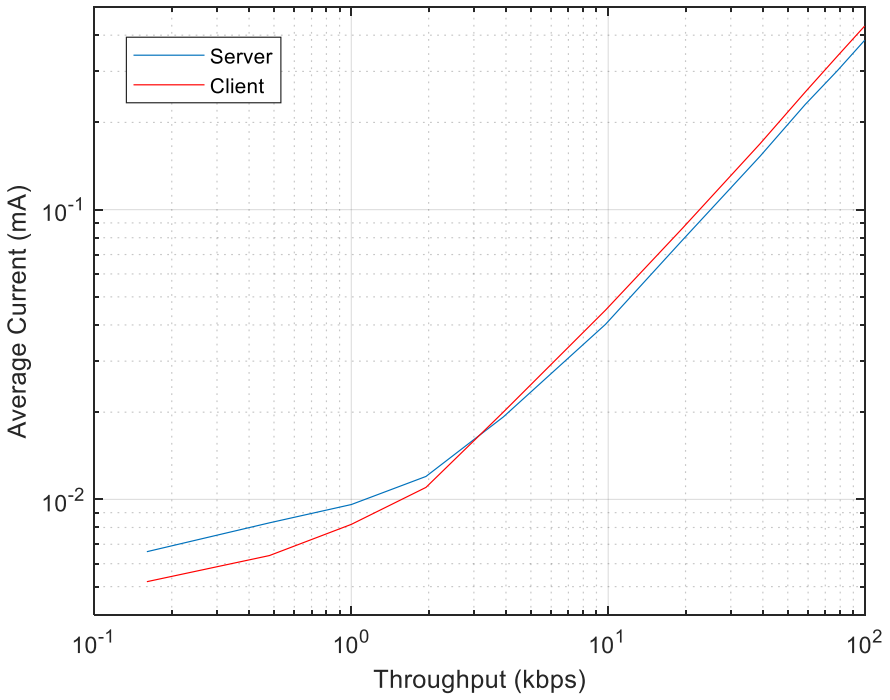

Figure 5-11: Server vs client current consumption

The problem with the central's role is not the connection state but rather the scanning and initiating phases prior to establishing a connection, when the device has to listen for long times waiting for advertising packets. An application-level algorithm may try to synchronize the advertising and scanning parameters of the system to optimize the connectionless power consumption.

5.4 Application performance

5.4.1 EKG gateway

As discussed in chapter 1, the EKG nodes may communicate to each other through wires, BLE, IBC or any other low-power technology. However, a gateway is required for such network data. BLE is a very good candidate to transmit such data to a smartphone, tablet or another personal device. In [10], BLE is used to transmit a 64-kbps load, which is claimed to be sufficient for 12-lead EKG, to a smartphone. In this section we simulate the same case and compare our results with those in [10]. Figure 5.12 shows the average current consumption for different CI values and BLE versions, at a constant throughput of 64 kbps, while the results of [10] are shown in figure 3.3.

Even for BLE 4.0 and the same CI values, the current nRF52840 SoC outperforms the three chips compared in [10], which is mainly because of the radio power reduction and the effect of the DC/DC converter. The least current that could be achieved in [10] is about 2.5 mA, while nRF52840 can achieve less than 1 mA for BLE 4.0. Additionally, the required throughput could not be achieved in [10] with CI values higher than 37.5 ms, while nRF52840 can achieve such throughput basically at any CI value. This is because earlier chips are limited in terms of the number of packets that can be exchanged during a connection event. Meanwhile, nRF52840 buffers new packets for transmission at almost the same speed of transmitting earlier packets, is

thus able to queue a “virtually infinite” number of packets and does not limit the number of packet exchanges per connection event.

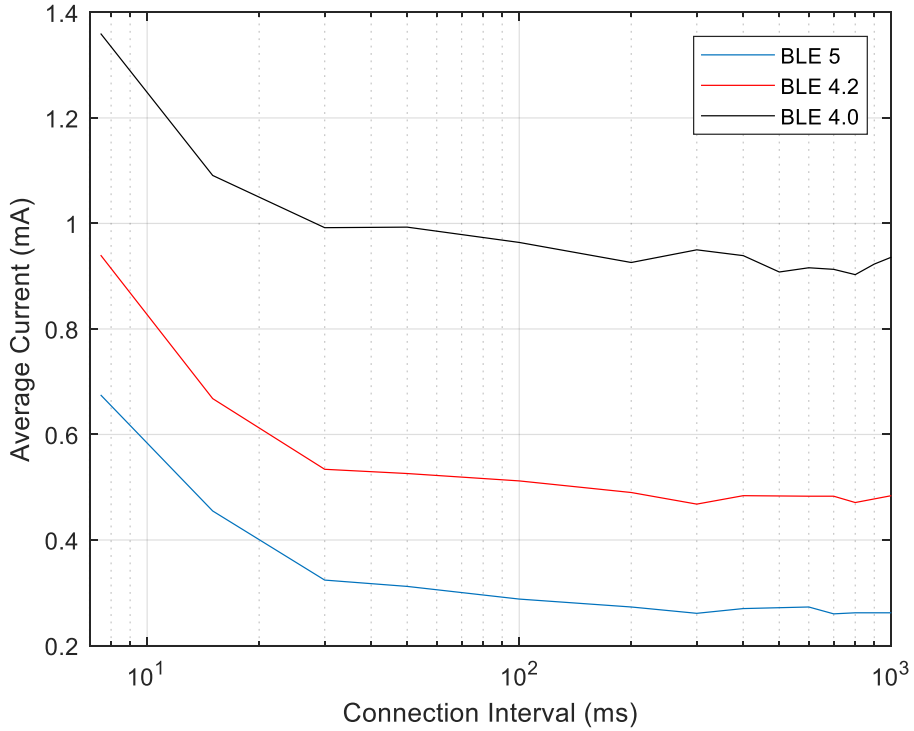

Figure 5-12: Average current consumption for different CI values and BLE versions at a constant throughput of 64 kbps

The current decreases when larger CI values are chosen, because readings are aggregated into larger and fewer packets and the device wakes up less frequently to send them. When CI is sufficiently large (i.e. above ~300 ms), there is no much gain for furtherly increasing CI. This is because for a sufficiently large CI and throughput, approximately the same average number of BLE packets per second is required to achieve such throughput. The power consumption is dominated by the radio transmission power rather than the consumption of other stages of the connection interval duration.

When BLE 4.2 and BLE 5 are considered, we can achieve an average current of 0.5 mA and 0.26 mA, respectively, saving up to 90% of power consumption with respect to the least achieved current in [10] for the continuous transmission of 64 kbps. For example, at CI = 300 ms, the current consumption is 950 µA, 468 µA and 261 µA on BLE 4.0, BLE 4.2 and BLE 5 respectively, which translates to a battery lifetime estimate of about 4.5, 9 and 16 days respectively on a 100-mAh small coin cell battery.

5.4.2 Single EKG node

In the final experiment, we show the current consumption of the processor and the BLE radio events for a single EKG node, where the processor continuously logs a 16-bit data field demonstrating an EKG reading, at a rate of 300 samples per second, thus the required throughput is 4.8 kbps. The readings are aggregated into one characteristic value. When the characteristic value reaches its maximal length or when the next connection event is due, a notification holding this characteristic value is queued for transmission, and the new readings are once again aggregated into the characteristic value. The current consumption for different CI values is shown in figure 5.13, where changing CI affects the protocol's latency.

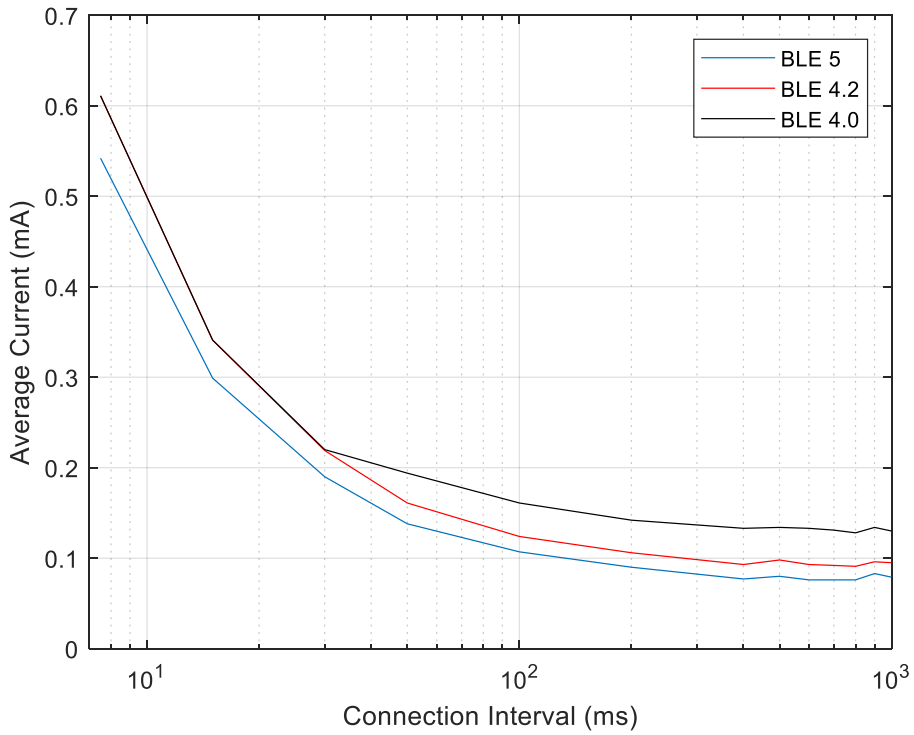

Figure 5-13: Current consumption for different CI values for a single EKG node

We found that the 300-Hz logging operation alone takes an average current of about 57 µA, while the BLE current depends on the choice of CI. A delay of 400 ms is generally considered acceptable for EKG, however a lower CI can be chosen if needed. At CI = 400 ms, the total current consumption is 133 µA, 93 µA and 77 µA on BLE 4.0, BLE 4.2 and BLE 5 respectively, which translates to a battery lifetime of 31, 45 and 54 days respectively on a 100-mAh small coin cell battery.

As the logging alone consumes a significant portion of power, a separate low power microprocessor running on a lower clock can be used for this purpose, leaving only BLE activity for the M4 ARM cortex.

6 Conclusion

This thesis presented a theoretical model and experimental setup for measuring the throughput and current consumption for continuous data monitoring and compared the performance for different BLE versions and parameters. The requirements for EKG monitoring were considered and they translate to an adequate coin cell battery lifetime and demonstrate the low power feature of BLE.

We conclude that the low power consumption of BLE is the fruit of the combination of the BLE specification capabilities, the hardware capabilities of a BLE chip and the good understanding of the use case that results in the choice of appropriate BLE modes and parameters. Upper-layer algorithms contribute to lowering the BLE power consumption.

Our hope is that this work contributes to an advanced understanding of the BLE performance and tradeoffs and its application in the field of wireless BSNs.

Bibliography

[1] Ayatollahitafti, V., Ngadi, M. A., Sharif, J. B., & Abdullahi, M. (2016). An Efficient Next Hop Selection Algorithm for Multi-Hop Body Area Networks. *Plos One, 11*(1). doi:10.1371/journal.pone.0146464

[2] Ghamari, M., Janko, B., Sherratt, R., Harwin, W., Piechockic, R., & Soltanpur, C. (2016). A Survey on Wireless Body Area Networks for eHealthcare Systems in Residential Environments. *Sensors,16*(6), 831. doi:10.3390/s16060831

[3] Radio Versions Bluetooth Technology (2018). Bluetooth Basic Rate/Enhanced Data Rate (BR/EDR). Retrieved from: https://www.bluetooth.com/bluetooth-technology/radio-versions

[4] Dementyev, A., Hodges, S., Taylor, S., & Smith, J. (2013). Power consumption analysis of Bluetooth Low Energy, ZigBee and ANT sensor nodes in a cyclic sleep scenario. *2013 IEEE International Wireless Symposium (IWS)*. doi:10.1109/ieee-iws.2013.6616827

[5] Rezaei, F., Hempel, M., & Sharif, H. (2015). A Survey of Recent Trends in Wireless Communication Standards, Routing Protocols, and Energy Harvesting Techniques in E-Health Applications. Wearable Technologies, 1479-1502. doi:10.4018/978-1-5225-5484-4.ch068

[6] Gertsch, M. (2009). *The ECG Manual: An Evidence Based Approach*. London: Springer-Verlag.

[7] Yahia, H. (2016). *Performance Analysis of Real-Time Wireless Body Sensor Networks using 802.15.4 and ZigBee Standards under Maximum Payload Conditions* (Unpublished MSc. Thesis), School of Science and Engineering. University of Kurdestan Hewler, Erbil, Kurdistan.

[8] Censi, F. Calcagnini G., Corazza I., Mattei E., Triventi M., Bartolini P., Boriani G., On the resolution of ECG acquisition systems for the reliable analysis of the P-wave, *Physiol. Meas.*, *33*(2), N11-7. doi: 10.1088/0967-3334/33/2/N11

[9] iRhythm (2018). Zio XT. Retrieved from: https://www.irhythmtech.com/products-services/zio-xt

[10] Giovanelli, D., Milosevic, B., & Farella, E. (2015). Bluetooth Low Energy for data streaming: Application-level analysis and recommendation. *2015 6th International Workshop on Advances in Sensors and Interfaces (IWASI)*. doi:10.1109/iwasi.2015.7184945

[11] Chevrollier, N. & Golmie, N. (June 2005) On the Use of Wireless Technologies in Healthcare Environments, *Proc. ASWN '05*, Paris, France.

[12] Bluetooth SIG (2016), *Bluetooth Core Specification*, version 5.0.

[13] Gomez, C., Oller, J., & Paradells, J. (2012). Overview and Evaluation of Bluetooth Low Energy: An Emerging Low-Power Wireless Technology. *Sensors*, *12*(9), 11734-11753. doi:10.3390/s120911734

[14] Argenox Technologies LLC (2017). *A BLE Advertising Primer*. Retrieved from http://www.argenox.com/a-ble-advertising-primer/

[15] Texas Instruments (2016). *BLE5-Stack User's Guide*. Retrieved from: http://dev.ti.com/tirex/content/simplelink_cc2640r2_sdk_1_35_00_33/docs/ble5stack/ble_user_guide/html/ble-stack/overview.html

[16] Bluetooth SIG Inc. Official Website (2018). *GATT Specifications*. Retrieved from: https://www.bluetooth.com/specifications/gatt/generic-attributes-overview

[17] *Getting Started with Bluetooth Low Energy* (2014). Retrieved from: http://apprize.info/hardware/bluetooth/3.html

[18] Seri, B., Vishnepolsky, G. (2017). BlueBorne: The dangers of Bluetooth implementations: Unveiling zero day vulnerabilities and security flaws in modern Bluetooth stacks. *Armis*. Retrieved from: https://www.armis.com/blueborne/

[19] Carnegie Mellon University in conjunction with CERT (2018). *Vulnerability Note VU#304725*. Retrieved from: https://www.kb.cert.org/vuls/id/304725.

[20] Texas Instruments (2013). *2.4-GHz Bluetooth® low energy System-on-Chip*. Retrieved from http://www.ti.com/lit/ds/symlink/cc2540.pdf

[21] Afonso, J.A., Maio, A.J.F. & Simoes R. (2016). Performance Evaluation of Bluetooth Low Energy for High Data Rate Body Area Networks, *Wireless Pers Commun*, *90(*1), 121–141. https://doi.org/10.1007/s11277-016-3335-4

[22] Feng, Z., Mo, L., & Li, M. (2015). Analysis of low energy consumption wireless sensor with BLE. *IEEE Sensors*, 2015. doi:10.1109/icsens.2015.7370563

[23] Aguilar, S., Vidal, R., & Gomez, C. (2017). Opportunistic Sensor Data Collection with Bluetooth Low Energy. *Sensors, 17*(12), 159. doi:10.3390/s17010159

[24] Tosi, J., Taffoni, F., Santacatterina, M., Sannino, R., & Formica, D. (2017). Performance Evaluation of Bluetooth Low Energy: A Systematic Review. *Sensors, 17*(12), 2898. doi:10.3390/s17122898

[25] Collotta, M., Pau, G., Talty, T., & Tonguz, O. K. (2018). Bluetooth 5: A Concrete Step Forward toward the IoT. *IEEE Communications Magazine, 56*(7), 125-131. doi:10.1109/mcom.2018.1700053

[26] Yaakop, M. B., Malik, I. A., Suboh, Z. B., Ramli, A. F., & Abu, M. A. (2017). Bluetooth 5.0 throughput comparison for internet of thing usability a survey. *2017 International Conference on Engineering Technology and Technopreneurship (ICE2T).* doi:10.1109/ice2t.2017.8215995

[27] Hernández-Solana, Á, Perez-Diaz-De-Cerio, D., Valdovinos, A., & Valenzuela, J. L. (2017). Proposal and Evaluation of BLE Discovery Process Based on New Features of Bluetooth 5.0. *Sensors ,17*(9), 1988. doi:10.3390/s17091988

[28] Marco, P. D., Skillermark, P., Larmo, A., Arvidson, P., & Chirikov, R. (2017). Performance Evaluation of the Data Transfer Modes in Bluetooth 5. *IEEE Communications Standards Magazine, 1*(2), 92-97. doi:10.1109/mcomstd.2017.1700030

[29] Jara, A. J., Fernndez, D., Lopez, P., Zamora, M. A., Ubeda, B., & Skarmeta, A. G. (2012). Evaluation of Bluetooth Low Energy Capabilities for Continuous Data Transmission from a Wearable Electrocardiogram. *2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computin*g. doi:10.1109/imis.2012.201

[30] Strey, H., Richman, P., Rozensky, R., Smith, S., & Endee, L. (2013). Bluetooth low energy technologies for applications in health care: Proximity and physiological signals monitors. *2013 10th International Conference and Expo on Emerging Technologies for a Smarter World (CEWIT).* doi:10.1109/cewit.2013.6851347

[31] Zhou, B., Chen, X., Hu, X., Ren, R., Tan, X., Fang, Z., & Xia, S. (2013). A Bluetooth low energy approach for monitoring electrocardiography and respiration. *2013 IEEE 15th International Conference on E-Health Networking, Applications and Services* (Healthcom 2013). doi:10.1109/healthcom.2013.6720653

[32] Sayeed, T. M., Rayhan, M. T., & Chowdhury, S. (2018). Bluetooth Low Energy (BLE) based portable medical sensor kit platform with cloud connectivity. *2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2).* doi:10.1109/ic4me2.2018.8465645

[33] Texas Instruments (2016). *Logical Link Control and Adaptation Layer Protocol (L2CAP).* Retrieved from http://dev.ti.com/tirex/content/simplelink_cc2640r2_sdk_1_35_00_33/docs/ble5stack/ble_user_guide/html/ble-stack/l2cap.html

[34] Gomez, C., Demirkol, I., & Paradells, J. (2011). Modeling the Maximum Throughput of Bluetooth Low Energy in an Error-Prone Link. *IEEE Communications Letters, 15*(11), 1187-1189. doi:10.1109/lcomm.2011.092011.111314

[35] Nordic Semiconductor (2018). *nRF52840 Product Specification v1.0*. Retrieved from http://infocenter.nordicsemi.com/pdf/nRF52840_PS_v1.0.pdf

[36] Nordic Devzone Developer Support (2014), *nRF51822 power consumption when BLE advertising/connected*. Retrieved from https://devzone.nordicsemi.com/f/nordic-q-a/4211/nrf51822-power-consumption-when-ble-advertising-connected#post-id-18589.

[37] Nordic Semiconductor (2018). *nRF52840 Preview Development Kit v0.11.x User Guide v1.2*. Retrieved from http://infocenter.nordicsemi.com/pdf/nRF52840_PDK_User_Guide_v1.2.pdf

[38] Nordic Semiconductor (2018). *S140 SoftDevice Specification v1.1*. Retrieved from http://infocenter.nordicsemi.com/pdf/S140_SDS_v1.1.pdf

[39] Nordic Semiconductor (2018). *Power Profiler Kit Product Brief Version 2.0*. Retrieved from http://infocenter.nordicsemi.com/pdf/Power_Profiler_Kit_PB_v2.0.pdf

[40] Nordic Semiconductor (2016). *Online Power Profiler*. Retrieved from https://devzone.nordicsemi.com/power/