INSTITUTO TECNOLÓGICO
Y DE ESTUDIOS SUPERIORES DE OCCIDENTE

Reconocimiento de validez oficial de estudios de nivel superior según acuerdo secretarial 15018, publicado en el Diario Oficial de la Federación el 29 de noviembre de 1976.

Departamento de Electrónica, Sistemas e Informática

DOCTORADO EN CIENCIAS DE LA INGENIERÍA

ANÁLISIS DEL CONSUMO DE POTENCIA DE BLUETOOTH LOW ENERGY EN PRODUCTOS COMERCIALES Y SU IMPLICACIÓN EN APLICACIONES DE INTERNET DE LAS COSAS

Tesis que para obtener el grado de

DOCTOR EN CIENCIAS DE LA INGENIERÍA

presenta: Eduardo García Espinosa

Director de tesis: Dr. Omar Humberto Longoria Gándara

Co-director de tesis: Dr. Arturo Veloz Guerrero

Tlaquepaque, Jalisco. Septiembre de 2019

DOCTOR EN CIENCIAS DE LA INGENIERÍA (2019) ITESO, Tlaquepaque, Jal., México

TÍTULO: **Análisis del consumo de potencia de Bluetooth low energy en productos comerciales y su implicación en aplicaciones de Internet de las cosas**

AUTOR: Eduardo García Espinosa
Ingeniero en Comunicaciones y Electrónica (Universidad de Guadalajara, México)
Maestro en Ciencias de la Ingeniería Eléctrica, opción Telecomunicaciones (CINVESTAV, México)

DIRECTOR DE TESIS: Omar Humberto Longoria Gándara
Departamento de Electrónica, Sistemas e Informática, ITESO
Ingeniero en Electrónica y Comunicaciones (ITESM, Campus Monterrey, México)
Maestro en Ciencias de la Ingeniería Eléctrica, opción Telecomunicaciones (CINVESTAV, México)
Doctor en Ciencias de Ingeniería Eléctrica, opción Telecomunicaciones (CINVESTAV, México)

NÚMERO DE PÁGINAS: xix, 84

ITESO – The Jesuit University of Guadalajara

Department of Electronics, Systems, and Informatics

DOCTORAL PROGRAM IN ENGINEERING SCIENCES

POWER CONSUMPTION ANALYSIS OF BLUETOOTH LOW ENERGY COMMERCIAL PRODUCTS AND THEIR IMPLICATIONS FOR INTERNET OF THINGS APPLICATIONS

Thesis to obtain the degree of

DOCTOR IN ENGINEERING SCIENCES

Presents: Eduardo Garcia-Espinosa

Thesis Director: Dr. Omar Humberto Longoria Gándara

Thesis Co-director: Dr. Arturo Veloz-Guerrero

Tlaquepaque, Jalisco, Mexico

September 2019

TITLE: **Power consumption analysis of Bluetooth low energy commercial products and their implications for Internet of things applications**

AUTHOR: Eduardo Garcia Espinosa
Bachelor’s degree in electronics and communications engineering (University of Guadalajara, Mexico)
Master’s degree in electrical engineering (CINVESTAV, Mexico)

THESIS DIRECTOR: Omar Humberto Longoria Gandara
Department of Electronics, Systems, and Informatics, ITESO
Bachelor’s degree in electronics and communications engineering (ITESM, México)
Master’s degree in electrical engineering (CINVESTAV, Mexico)
Ph.D. degree in electrical engineering (CINVESTAV, Mexico)

NUMBER OF PAGES: xix, 84

This dissertation is dedicated to Hilda, my lovely wife, whose love and support inspired me to pursue and complete this research.

Resumen

Internet de las cosas es un concepto que incluye diferentes tipos de dispositivos tales como: computadoras, tabletas, teléfonos inteligentes y objetos de uso diario. Estos últimos incluyen sensores que comparten información entre ellos y que pueden ser controlados desde Internet. Por lo tanto, un ecosistema de Internet de las cosas usa estos dispositivos inteligentes para conectarse a Internet y así habilitar diferentes tipos de negocios y consumidores. Dado lo anterior, se estima que 50 mil millones de dispositivos estarán interconectados para el año 2020, de tal manera que existen considerables oportunidades de negocio para aquellas empresas que inviertan en el Internet de las cosas. Es importante resaltar que el protocolo de comunicaciones conocido como Bluetooth low energy (BLE por sus siglas en inglés), es ampliamente usado en aplicaciones de Internet de las cosas y está disponible en la mayoría de las plataformas de desarrollo disponibles en el mercado. Así mismo, BLE fue pensado para funcionar con baterías tipo botón al ser un protocolo de muy bajo consumo de potencia. Es por esta razón que es muy importante entender el consumo de potencia de las plataformas de desarrollo para poder implementar, de manera eficiente, aplicaciones de Internet de las cosas. Es por ello que esta investigación doctoral presenta los antecedentes requeridos para entender el protocolo de comunicaciones BLE y sus principales parámetros usados en una conexión entre dispositivos. Adicionalmente se muestran los procedimientos requeridos para calcular y medir el consumo de potencia en diferentes plataformas de desarrollo comerciales y finalmente se presenta una guía de diseño para seleccionar los parámetros de conexión requeridos para maximizar la vida útil de la batería para cualquier aplicación de Internet de las cosas.

Summary

Internet of things (IoT) is a concept that includes different kinds of devices: computers, tablets, smartphones and everyday objects. These devices include embedded sensors that share information between each other and that can be controlled via the Internet. Therefore, an IoT environment will use these smart objects as the main components to enable business and consumers to connect to the Internet and share data. Given these points, the estimation is that 50 billion devices will be connected by 2020, thus posing considerable market opportunities for those companies investing in IoT. It is important to note that the Bluetooth low energy protocol (BLE) is widely used as the wireless communication protocol for IoT applications and is included in many of the hardware platforms available in the market. Additionally, BLE was designed as an ultra-low power consumption protocol in order to allow the devices to work with coin cell batteries. For this reason, it is very important to understand the power consumption of a hardware platform in order to implement an efficient IoT application. This doctoral dissertation presents the required background to understand the protocol and the main connection parameters and presents the setup and procedures to calculate and measure the power consumption of different Bluetooth low energy commercial platforms. In addition, based on the power consumption of the hardware platforms configured as a central or peripheral device, a design guideline is provided to select the required connection parameters of the Bluetooth protocol in order to maximize the battery life of the platform for a given IoT application.

Acknowledgements

The author wishes to express his sincere appreciation to Dr. Omar Longoria Gándara, professor of the Department of Electronics, Systems, and Informatics at ITESO, for his encouragement, expert guidance and keen supervision as doctoral thesis director throughout the course of this work. The author offers his gratitude to Dr. Arturo Veloz Guerrero for his support as doctoral thesis co-director during the development of this work. He also thanks Dr. Jose Luis Pizano Escalante, Dr. Luis Rizo Dominguez, and Dr. Abisaí Ramirez Perez, members of his Ph.D. Thesis Committee, for their interest, assessment, and suggestions.

Special thanks are due to Intel Tecnología de Mexico, for their support provided during the development of this work.

The author gratefully acknowledge the financial assistance through a scholarship granted by the *Consejo Nacional de Ciencia y Tecnología* (CONACYT), Mexican Government, as well as the financial support provided by Intel Tecnología de Mexico.

Finally, special thanks are due to the women in my life: my wife Hilda, my mother Carmelita, and my sister Liz, for their understanding, patience, and continuous loving support.

Contenido

Contents

List of Figures

List of Tables

Introduction

Internet of Things (IoT) has arrived and will soon be a very important business segment for the IT industry due to the number of devices that will be connected. The market trend driving the Internet of Things is that everything needs to be connected, which means that billions of intelligent devices will be sharing data. In order to support this new information era, several architectural challenges and design issues need to be addressed, such as power consumption of sensor networks, channel modeling and MAC protocol design, among others.

The IoT can be defined as "The general idea of things, especially everyday objects that are readable, recognizable, locatable, addressable and controllable via the Internet – whether via RFID, wireless LAN, wide-area network, or other means" [Swan-12] [Liu-12]. This means that the IoT is a broader concept that includes not only traditional devices such as computers, tablets or smartphones, but everyday objects that can gather data that later on can be shared to a user [Yang-11] [Mattern-10]. Nowadays, sensor devices are becoming cheaper and widely available and so more "things" are being connected such as home devices, business and public infrastructure, health care, etc. From a business point of view, IoT represents an enormous opportunity for various types of companies, which are expected to generate thousands of billions in revenue [Mazhelis-13] due to the fact that by 2020, 50 billion of connected devices are expected [Swan-12].

There are several examples for IoT devices and applications, such as monitoring and controlling homes and buildings, automotive and transportation applications, health self-tracking monitoring, among many other examples. Consequently, in order to enable the IoT ecosystem, there exists several enabling communication protocols, which need to be low power due to the fact that many applications are required to operate using coin cell batteries. The main communication protocols for IoT are RFID, ZigBee, WiFi, LoRa, ANT, and Bluetooth low energy (BLE). In general, these communication protocols present different design challenges such as:

a) Frequency selection
b) Channel modeling
c) Antenna design
d) Physical layer protocol design
e) Energy efficient hardware

f) Security and privacy

g) Standardization

Additionally, there are also architectural challenges:

h) Data transmission failure due to: collision, node draining of energy, node busy, or other accidents.

i) The sensors are battery operated and a prolonged network lifetime is preferred. This motivates development of energy aware routing.

j) The sensor mobility generates channel fading during data transmission, which degrades the performance in terms of bit error rate (BER) and frame error rate (FER).

As a result, such design and architecture challenges bring a huge opportunity for different research areas.

During this decade, IoT has been evolving and today we can witness several commercial applications already available in the market as well as different vendors offering development platforms to implement IoT applications. Moreover, from the forecasted 50 billion connected devices, around 30% of them will use BLE [Bluetooth SIG-18]. Additionally, 100% of the smartphones, tablets and laptops shipped in 2018 include BLE [Bluetooth SIG-18]. Therefore, from a business point of view, IoT and BLE represent a huge opportunity for the companies that decide to invest in these technologies. For this reason, the BLE protocol and the analysis of the power consumption of it, becomes the main motivation for this research work. One fundamental aspect while choosing a development platform is to analyze the power consumption of its components and to calculate the battery life of them. These two parameters are very important since the BLE protocol was designed to operate continuously using coin cell batteries from months to years, since depending on the application, the battery replacement will not be an easy task [Garcia-Espinosa-18].

In this doctoral dissertation, different methods to calculate the power consumption of the BLE protocol are analyzed and as a result, design guidelines for the main connection parameters of the protocol are proposed.

The doctoral dissertation is organized as follows:

Chapter 1 provides a review of BLE, focusing on the link layer of the protocol. Additionally, the state of the art regarding the analysis of the power consumption of the protocol is presented. Chapter 2 presents different computational models that allow to calculate the power consumption of the protocol. Chapter 3 presents the setup and methods to perform physical measurements of power consumption from different commercial hardware platforms. Additionally, some experiments and their results are shown. Chapter 4 presents a design guideline to maximize the battery life of a BLE device. Finally, Chapter 5 presents the evolution of the BLE protocol, its challenges and the competitors IoT protocols.

1. Bluetooth Low Energy

BLE is the evolution of the Bluetooth standard. It is an open and license free standard designed for ultra-low power consumption, which is fundamental for sensor devices with very low battery capacity. BLE is presented in 2010 in the 4.0 Bluetooth specification [Bluetooth SIG sd-14] and defines new efficient ways to discover and connect devices. Traditional Bluetooth is connection oriented, meaning that when a device is connected, the link is maintained, even if there is no data to send. BLE on the other hand, will remain in sleep mode 99% of the time [Heydon-12], so the protocol is optimized for the lowest power consumption:

a) Short packets that reduce Tx and Rx peak current

b) Less RF channels to improve discovery and connection time

c) A very simple state machine.

Such characteristics implies some limitations, such as a short range and reduced data throughput (2 Mbps data rate in the latest version of the specification), since BLE will only be able to send small chunks of data, however, this is the main requirement to keep the low energy consumption for IoT applications. In the next section, the importance of the protocol in communications is explained.

1.1. BLE and its importance in IoT communications

Nowadays, IoT applications are exploding and the connectivity among different devices is via wireless, thus new market opportunities are arising. For instance, ABI research[1] states that by 2022 there will be over 930 million combined accessories for personal computers (PCs) and mobiles, including audio, keyboards, mice, and many others. Additionally, there is the need to control accessories wirelessly, posing new market opportunities such as in virtual reality, consumer robotics and smart home, where more than 260 million devices will be shipped. Moreover, wearable devices are set to reach over 330 million annual shipments by 2022, 20 million healthcare devices are also forecasted for 2022. In all these cases, BLE is the primary chosen

[1] ABIresearch Bluetooth opportunities and challenges for the next decade, https://www.abiresearch.com/

solution for connectivity.

Another market opportunity for BLE is in the fourth industrial revolution, also known as Industry 4.0 (I4.0), which is expected to change the future of manufacturing and production processes leading to smart factories and networked industrial environments [Raptis-19]. The first industrial revolution was based on steam, the second in electricity, the third one in computing and the fourth one is based in the connectivity of everything. Additionally, the I4.0 enablers are: artificial intelligence, human-computer interaction, robotics, IoT, big data, and cloud computing among others [Aceto-19]. Thus, BLE plays an important role in the I4.0 by allowing the interface to industrial machinery without the need for complex human interface, such as deployment of sensor nodes in difficult to reach areas that can provide data of industrial equipment: pressure, acceleration, temperature, vibration, etc.

In like manner, 5G, the next generation of mobile-network connectivity, will be a boost for IoT applications due to its offering: increased data volume over previous generations, low latency, higher data transfer rate, 100x more devices capacity and higher availability [Khurpade-18]. Important to realize is that 100% of the smartphones shipped in 2018 have BLE, thus allowing the protocol to be the number one choice for IoT applications. The reason for this is that IoT applications rely on many sensors that are generating data and 5G technology will be able to handle the amount of data generated by these sensors as well as provide them with the required internet access to be controlled and share the data.

Consequently, as new IoT markets are continuously emerging, the BLE protocol has a big opportunity to dominate them, however the protocol needs to be evolving as new communication solutions are also targeting such IoT markets and are directly competing with BLE. In the next section, the BLE architecture is presented.

1.2. BLE architecture

As can be seen in Fig. 1.1 BLE Architecture. Figure taken from [CSR-00]., the BLE architecture is very simple as it is split into three basic parts: controller, host, and applications. The

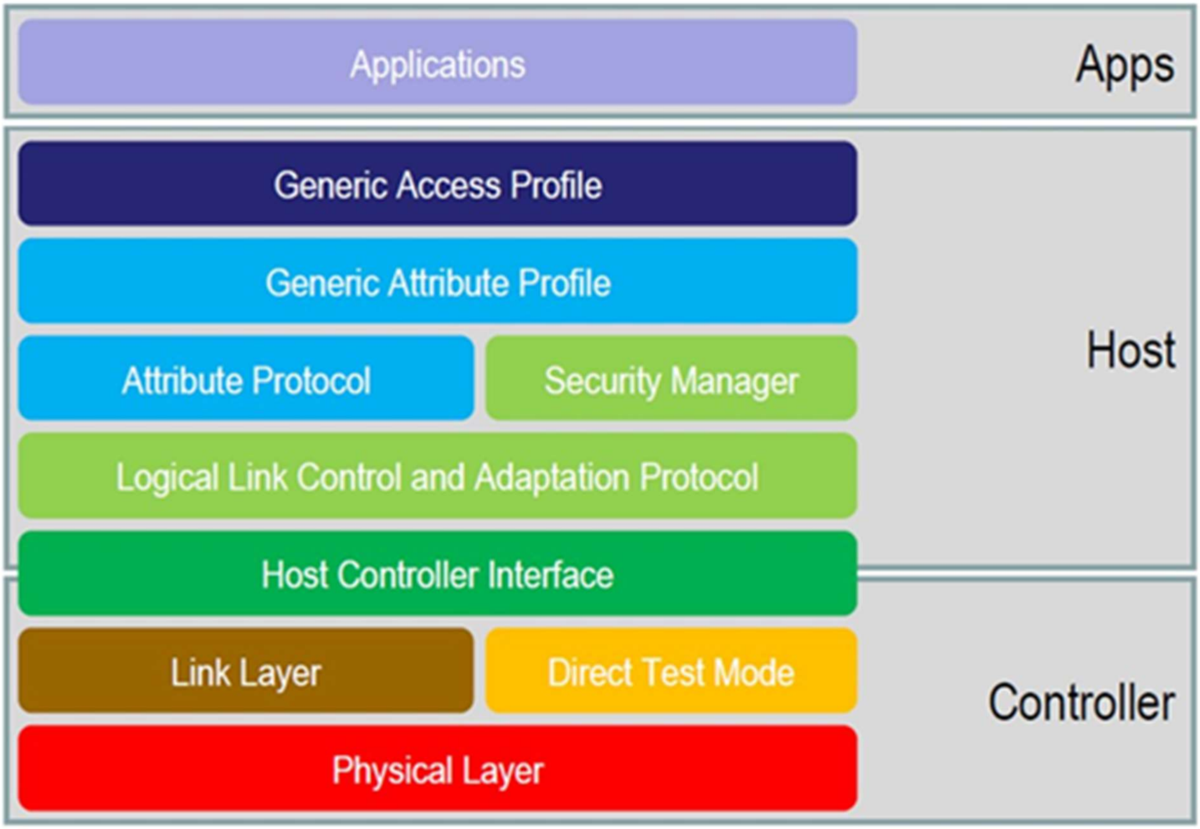

Fig. 1.1 BLE Architecture. Figure taken from [CSR-00].

controller is a physical device that can transmit and receive radio signals, the host is a software stack that manages how two or more devices communicate with each other and the applications use the software stack to enable a use case [Heydon-12].

The controller contains the physical and the link layer which are the interfaces with the outside world. The physical layer uses the ISM band (2.4 GHz) and relies on a Gaussian frequency shift keying (GFSK) modulation to operate, but since the ISM band is used by other several radios, the physical layer needs to use a frequency hopping scheme with 40 RF channels with a 2 MHz separation from each other to avoid, as much as possible, the interference from other devices. Of these 40 channels, 37 are used to transmit data and the remaining 3 are used as Advertising channels as can be seen in Fig. 1.2.

The Link Layer is responsible for advertising, scanning and creating connection between BLE devices. The devices will use the advertising channels to advertise that they are discoverable and connectable, and once they are discovered by a master device, they will establish a connection and will use one of the 37 data channels to send data [Heydon-12].

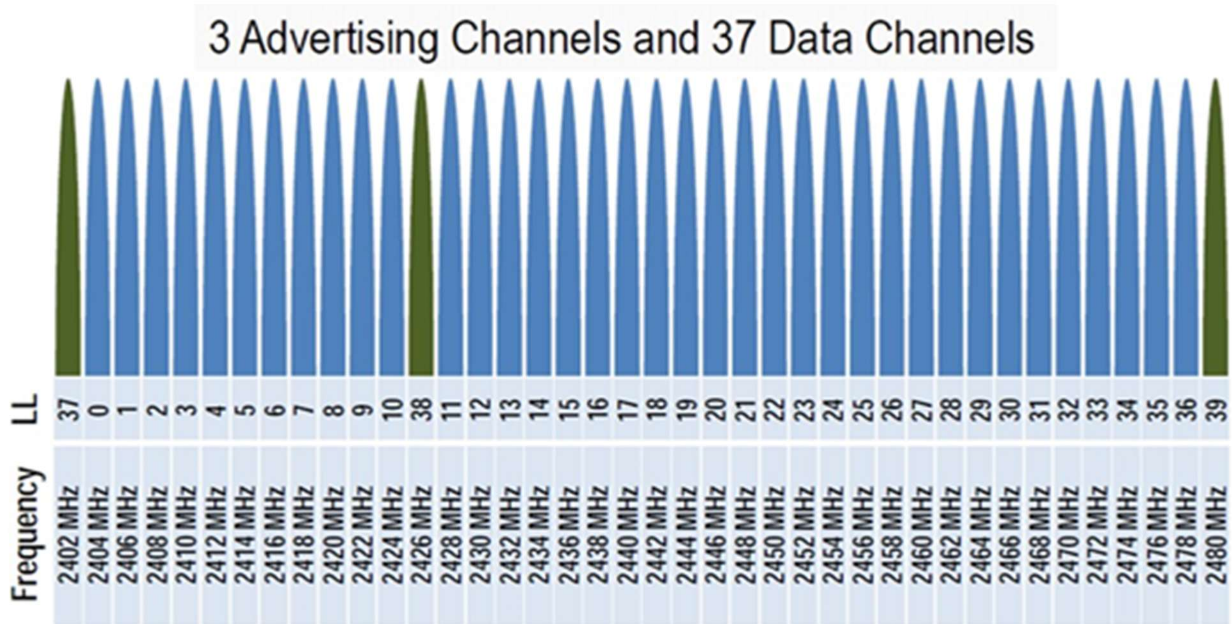

Fig. 1.2 Physical Layer Data and Advertising Channels. 37 data channels and 3 advertising channels. Figure taken from [CSR-00].

For data sending on the 40 channels, either user data or advertising data, the link layer defines packets that include information to identify the intended receiver.

The Host is comprised of different protocols that define profiles and services needed to implement a given application. A BLE device can operate in four different profile roles:

a) Peripheral: An advertiser that is connectable and operates as a slave in a connection. Example: Heart rate sensor
b) Central: Scans for advertisements and initiates connections, thus operates as a master in connections. Example: Smartphone
c) Broadcaster: An advertiser that is non-connectable. Example: Temperature sensor
d) Observer: Scans for advertisements but cannot initiate connections. Example: Temperature display

The Application layer defines the specifications: characteristics, services and profiles. The characteristics is an identifier value of 16 bits known as Universally Unique Identifier (UUID) which can be reused and has no behavior. A service is a human-readable specification which defines the behavior of a characteristic in a server. A profile is a specific use case which consists of a group of services that will satisfy it. In other words, a profile is a specification that ensure the interoperability between vendors for a specific use case. There are several profiles and some examples are:

a) Health care profiles
- BLP (Blood pressure profile)
- GLP (Glucose profile)
- HTS (Health thermometer profile)
- IDP (Insulin delivery profile)

b) Sport and fitness profiles
- HRP (Heart rate profile)
- LNP (Location and navigation profile)
- CPP (Cycling power profile)
- WSP (Weight scale profile)

c) Proximity sensing
- FMP (Find me profile)
- PXP (Proximity profile)

There are three main services that BLE provides to the IoT [Bluetooth-SIG-18]:

a) Data transfer services: used by devices serving applications for sports and fitness, health and wellness, and peripheral and accessories in a point to point topology.

b) Location services: used for navigation and for item and asset tracking in a broadcast topology.

c) Device networks: used to control, monitor, and automate systems using a mesh topology.

For each one of these services, there is a technology backbone supporting them. In the case of the data transfer applications, the enabling technology behind it is the body area network (BAN) and more precisely, a wearable. A wearable can be defined as a device that can sense, collect, and upload physiological data to improve the quality of life [Seneviratne-17]. Such devices come in the form of smart watches, wrist bands, smart clothing, etc.

On the other hand, for location services, the main technology behind it is the BLE beacon. A beacon is a device that is broadcasting a BLE packet with any kind of information, such as ambient data, location data, or orientation data [Lindh-15]. BLE beacons are very easy to implement, thus a beacon's device battery can last for several years. A survey of the state of the art and technical challenges for beacons is presented in [Jeon-18].

Finally, for device networks, the mesh networking is the enabling technology. A mesh network allows each node to communicate with other nodes to share context information, thus increasing the network range and reliability [Benyamina-11]. It is important to realize that the mesh networking in BLE is not dependent on the latest version of the specification (5.0) and its implementation is optional. In other words, the implementation of a mesh network in BLE is backwards compatible and is not mandatory for every BLE device. From the 3 main services provided by BLE, the mesh networking is the one that poses major challenges as there are many different solutions for its implementation. In [Darroudi-17], a survey of the different mesh implementations in BLE is shown.

For this reason, BLE has a big potential to be the main protocol for IoT applications. In the next section, the BLE link layer is explained in detail.

1.3. BLE link layer

The BLE link layer (LL) is responsible for advertising, scanning and creating connections by means of two type of packets: advertising and data packets [Heydon-12]. A device that transmits advertising packets is called an advertiser. If such device requires to broadcast data, it will do it using advertising packets through advertising channels. If the device requires to receive data through advertising packets it will be called a scanner, and if a device requires bi-directional data communication it will advertise its existence to another device which is usually called the initiator. When a bi-directional data connection is needed, the protocol defines two device roles: the master (an initiator) and the slave (an advertiser) [Gomez-12].

A BLE device can operate in four profile roles that are defined by the Generic Access Profile (GAP) as follows:

a) Broadcaster: Advertiser (sends advertising packets) which is not connectable and thus not requiring a receiver, e.g. temperature sensor.

b) Observer: Advertiser (receive advertising packets) which scans for broadcasters to report the information to an application and does not require a transmitter, e.g. temperature display.

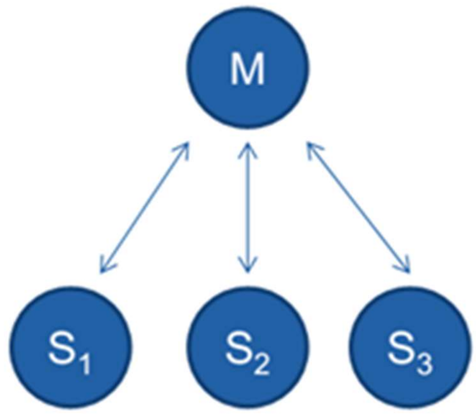

Fig. 1.3 A Piconet showing a Master (M) and Slaves (Sx). Figure taken from [CSR-10].

c) Peripheral: Slave in a connection, requires transmitter and receiver, e.g. heart rate sensor.

d) Central: Master in a connection, requires transmitter and receiver, e.g. smartphone.

A master can have multiple connections at the same time with different slaves, whereas a slave can only be connected to a unique master. Typically, a master can be a smartphone and a slave would be a small device, such as a sensor. This network configuration is called a piconet and it's basically a star topology, as shown in Fig. 1.3. As a naming convention for this document, a master will be referred as central device (CD) and a slave will be referred as a peripheral device (PD).

The link layer has two types of channels (see Fig. 1.2 Physical Layer Data and Advertising Channels. 37 data channels and 3 advertising channels. Figure taken from [CSR-00].):

a) 3 advertising channels used for discoverability/connectability and broadcasting/observing. These channels are indexed as 37, 38, and 39.

b) 37 Data channels, used to send data channel packets and application data which are indexed from 0 to 36.

As mentioned before, the LL defines two type of packets: advertising packets, which are transmitted in the advertising channels and are used to find other devices and data packets, which are transmitted in data channels and are used to send data once a connection between a master and slave has occurred. Fig. 1.4 shows the BLE link layer state machine, which defines five states: standby, advertising, scanning, initiating and connection. In the next sub-sections, each state is explained based on the Bluetooth core specification definition [Bluetooth SIG sd-14].

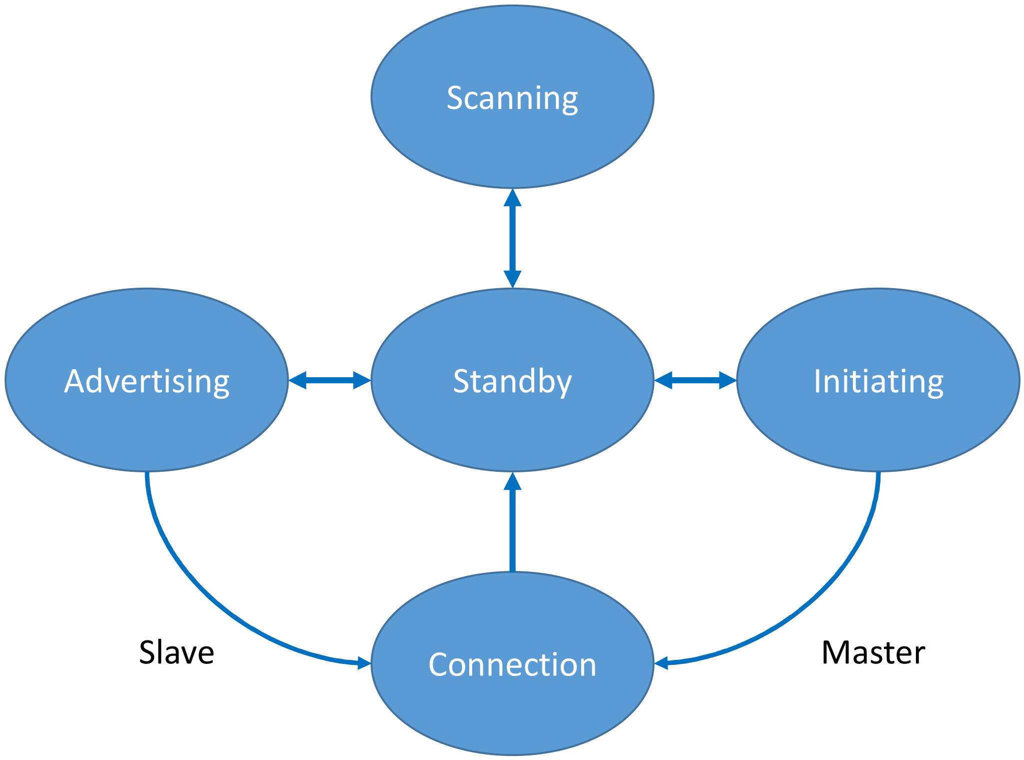

Fig. 1.4 BLE Link Layer state machine.

1.3.1 Standby state

This is the default mode of the LL state machine. A device will remain in standby mode until the host decides if it needs to move to a different state such as advertising, scanning or initiating. No packets are transmitted or received in this state and it can be entered from any other state.

1.3.2 Advertising and scanning states

A device in advertising mode will periodically broadcast advertising information in the advertising channels, where this process is known as advertising event, and will wait for a response from other devices. A device that is in this state is known as advertiser. Consecutive advertising events are separated by advertising intervals. There are different advertising packets, as can be seen in Table 1.1. In the scanning state, a device (known as scanner) will look for advertising information for a time duration known as scan window and will act according to the type of packet received [Bluetooth SIG sd-14].

TABLE 1.1. TYPES OF ADVERTISING PACKETS

Packet name	Description
ADV_IND	Connectable undirected advertising event
ADV_DIRECT_IND	Connectable directed advertising event
ADV_NONCONN_IND	Non-connectable undirected advertising event
SCAN_REQ	Scanner wants information from Advertiser
SCAN_RSP	Advertiser gives more information to Scanner
CONNECT_REQ	Initiator wants to connect to Advertiser

The BLE specification defines several parameters and their ranges to control the timing intervals of the advertising process as follows:

1) The advertising interval (*advInterval*) should be an integer multiple of 625 us in the range of 20 ms to 10 ms.

The time between the start of two consecutive advertising events (*TadvEvent*) is

$$TadvEvent = advInterval + advDelay \tag{1-1}$$

where *advDelay* is a pseudo-random value in a range of 0 to 10 ms.

Scan window (*scanWindow*) should be less than or equal to 10.24 ms. Fig. 1.5 shows the advertising event parameters.

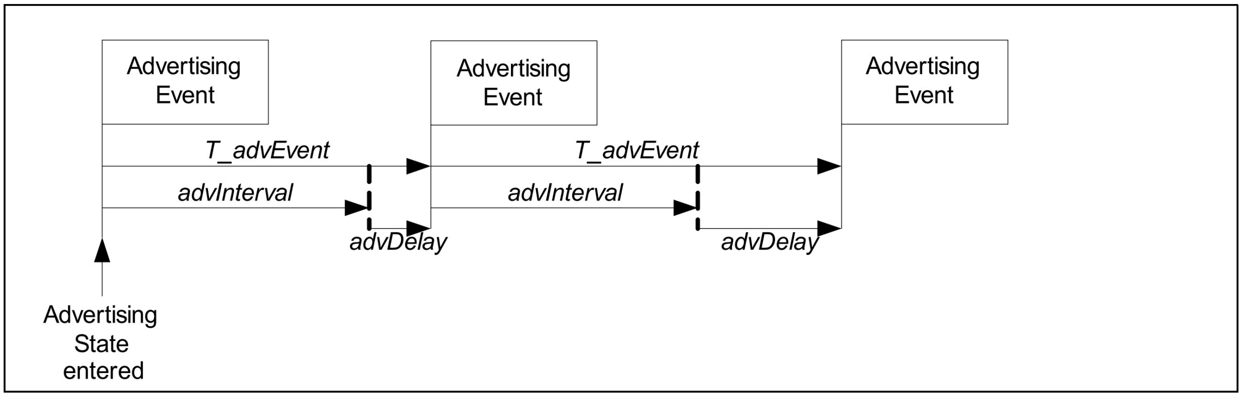

Fig. 1.5 Advertising event. Figure taken from [Bluetooth SIG sd-14].

1.3.3 Initiating state

A device in initiating state will send connection requests as a response of receiving advertising packets from a device that is advertising itself as connectable, meaning that such device will require a bi-directional connection to exchange data.

1.3.4 Connection state

In the connection state, a device can be in one of two roles: master or slave. If it's in a master role, the device will communicate with a slave after receiving an advertising event and the first data packet to be sent will define timing parameters for transmission. Such parameters are sent in a CONNECT_REQ data packet and will contain important parameters such as the transmit window size, the transmit window offset, a connection interval, slave latency, connection timeout, hop sequence and channel map [Bluetooth SIG sd-14]. Once the connection is established between the master and the slave, a point of synchronization is established, and it is known as connection event. During a connection event, the master and the slave will exchange packets. The timing of the connection events is determined by two parameters defined in the BLE specification: Connection event interval (*connInterval*) which shall be a multiple of 1.25ms in the range of 7.5ms to 4s and slave latency (*connSlaveLatency*) which defines the number of consecutive connection events where the slave is not required to listen to the master in order to save power. This parameter is an integer in the range of 0 to 499. If its set to 0, the slave will wake up every connection event, otherwise the slave could ignore a master request for a specific number of consecutive connection events but taking care of not exceed the connection timeout parameter (*connSupervisionTimeout*) which is defined as:

$$connSupervisionTimeout = timeout*10ms \quad (1\text{-}2)$$

and should be less than 6 (*connectionInterval*).

If a device is in a slave role, it will start transmitting after receiving a packet from the master where all the connection parameters are defined, then it can transmit a data packet with information to the master. The slave will be in sleep mode and will wake up periodically to listen for packets from the master, which determines the instants where the slaves are required to wake up and listen by means of a time division multiple access (TDMA) scheme. For the master to manage the scheduling mechanism efficiently, it has the flexibility to setup the connection event at a chosen time by means of parameters defined in the BLE specification:

a) The transmit window starts at *transmitWindowOffset* + 1.25 ms, after the end of the connection request.
b) The *transmitWindowSize* parameter shall define the size of the transmit window and shall be a multiple of 1.25 ms in the range of 1.25 ms to the smaller between 10 ms and (*connInterval* – 1.25 ms).
c) The *connInterval* is used in the calculation of the maximum offset and size of the transmit window.
d) The *transmitWindowOffset* shall be a multiple of 1.25 ms in the range of 0 ms to *connInterval*.

Fig. 1.6 shows the connection setup parameters as stated in the BLE specification [Bluetooth SIG sd-14]. Once the connection is established, devices can send data, using a stop and wait flow control mechanism [Gomez-12], until there is nothing else to transmit, in which case the connection event will be closed until the next event.

1.4. BLE power consumption

Nowadays, IoT applications are exploding and this new and evolving ecosystem requires new electronic devices that need to be powered by batteries, especially coin cell batteries, thus

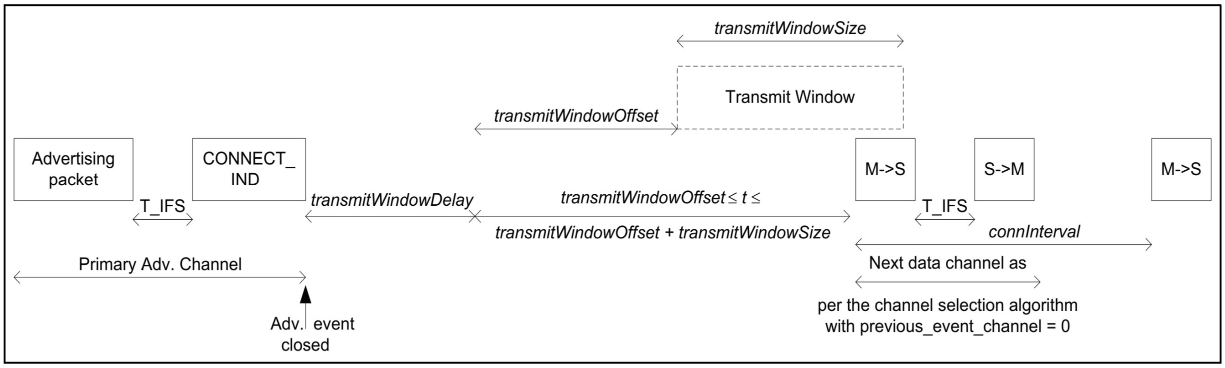

Fig. 1.6 Connection setup parameters sent in a CONNECT_REQ data packet. Figure taken from [Bluetooth SIG sd-14].

posing a challenge to design and fabricate low power devices, as well as to implement power efficient applications. In other words, the power consumption and battery life have become a main concern.

Although the BLE protocol theoretically defines the means to achieve low power consumption, in practice it is more complicated since the hardware and firmware of the development platforms along with different environment variables play a significant role while trying to implement the required low power consumption application. In the next sub-sections, the state of the art regarding BLE power consumption is presented.

1.4.1 BLE low power consumption state of the art

BLE first academic studies were focused on the performance and power consumption. In 2011 [Gomez-11] proposed an analytical model of the maximum BLE throughput is presented and it is shown that the actual throughput is less than the value stated in the BLE specification. [Dementyev-13] presents a comparison between different communication protocols such as ANT, ZigBee and BLE and it is shown that the power consumption of a protocol is determined by different factors and it cannot be predicted by just using the data provided in the datasheets by the vendors. [Liu Chen-12] introduces an analytical model to calculate the average energy consumption during the neighbor discovery procedure and the model is validated by simulation. [Siekkinen-12] provides a comparison between ZigBee and BLE power consumption using commercial devices, and it is concluded that BLE consumes less energy, but the authors documented several limitations from the BLE protocol while performing the experiments.

Additionally, they concluded that in order to reduce the power consumption, the PD should transmit as many as possible data frames within a single connection event. In [Kindt-14], an energy consumption model is presented and validated by means of simulation and physical measurements. [Treurniet-15] analyze the BLE energy consumption under the influence of mutual interference. In this case, an experiment using up to 30 devices was performed in order to understand the performance of the protocol as well as the power consumption under these circumstances. It was concluded that the BLE protocol is very robust and the performance is not heavily degraded in the presence of interference of other devices.

In any case, most of the experiments performed in the literature are using the CC2540 device in a slave configuration. Some other experiments were performed using other vendors, such as the Nordic NRF51822 system on chip (SoC) [Ting-17]. However, the power consumption of a master device it is not documented anywhere in the literature, and all the experimental results are based only on a peripheral device. While this is the most common use case in an IoT application, there might be several other applications where a CD using coin cell batteries is required and therefore there is a need to understand the power consumption in this scenario. Moreover, there is also the need to understand the role of a microcontroller and its firmware (FW) stack in the power consumption of a development platform.

2. BLE power consumption simulation models analysis

The BLE protocol is meant to be used in ultra-low power applications for the IoT. Many of such applications are based on low-power sensors which are required to use a coin cell battery that need to last for a long time –from months to years-, thus the power consumption need to be optimized.

As have been stated in Chapter 1, BLE specifies several connection parameters. Such parameters have an impact in the power consumption of a device. In other words, if such parameters are not chosen properly for a given application, then the power consumption and the battery life will be negatively impacted. For this reason, it is desired to have a simulation model that can handle different values of the BLE connection parameters and provide an estimation of the power consumption. Moreover, having such a model allows to provide a design guideline for different IoT applications by finding the optimal connection parameters, thus increasing the battery life. Table 2.1 shows the main connection parameters of the BLE protocol.

Extensive research has been carried on to model the power consumption of the BLE protocol, either using mathematical models [Gomez-11] as well as using simulation models [Mikhaylov-14] [Kindt-14]. However, authors of the referenced works do not provide a design guideline for a given application but rather explore the energy consumption for a wide range of connection parameter values.

This Chapter evaluates two existing BLE models [Mikhaylov-14] and [Kindt-14], based on the following criteria: easy of setup and model accuracy. Based on these two criteria, a model is chosen to carry on further experiments in order to obtain a design guideline for different IoT applications.

TABLE 2.1. MAIN CONNECTION PARAMETERS OF THE BLE PROTOCOL

Connection parameter	Range
Connection Interval (connInterval)	7.5 ms – 4 s
Connection slave latency (connSlaveLatency)	<500
Connection supervision timeout (connSupervisionTimeout)	100 ms to 32 S
Transmit window size (transmitWindowSize)	1.25 ms to 10 ms
Transmit window offset	0 to *connInterval*

2.1. BLE Omnet++ simulation model

Omnet++ [Virdis-19] is a discrete event simulator (DES) designed to simulate communication networks. Additionally, there are several third-party simulation frameworks that can be integrated into Omnet++, which provide additional libraries to be used in the simulation, such as wireless channels that allow to include more realistic parameters in the simulation. The Omnet++ DES provides the ideal framework to implement the BLE protocol, however, a caveat is the complex setup of the tool and the long learning curve.

A BLE model[2] was implemented in C++ and integrated into Omnet++ by [Mikhaylov-14] and is available under GNU General Public License. Although the model only implements the BLE link layer, it is enough to perform simulations with all the connection parameters defined in the BLE specification, including advertising interval, scan interval, scan window, and connection interval. Additionally, the model can simulate the different operating modes: advertising, scanning, and connected modes. For power consumption, the model uses the data from the datasheet of the widely used BLE transceiver CC2540 from Texas Instrument (TI) [Texas 2540-19]. Additionally, [Mikhaylov-14] performed physical measurements to compare with the simulation results.

In the next sub-sections, the evaluation of this model, based on the criteria specified above,

[2] Bluetooth low energy network simulation tool, Konstantin Mikhaylov, University of Oulu, Finland. 2014. http://cc.oulu.fi/~kmikhayl/BLE.html

is presented.

2.1.1 Easy of setup

The model lacks documentation; therefore, the setup is a difficult process as it requires a depth understanding of the Omnet++[3] framework, the compilation process and the dependencies required for the simulation model to compile. Moreover, the model was implemented in an old version of Omnet++ and Mixim[4], which increases the complexity of the setup if a newer version of the framework is being used. Additionally, there is no information of how to run a simulation, therefore, it takes considerable time to figure out the process to set the parameters and launch the simulation along with the interpretation of the results. As a conclusion, the setup of the model is quite difficult.

2.1.2 Model accuracy

The model can be fed with the specification data from existing BLE transceivers, such as operating voltages, currents, and other parameters of interest. In this case, the model uses the data from the TI CC2540 BLE transceiver [Texas-14]. Also, Mixim provides a battery model that can be used to specify the same parameters of a commercial CR2032 coin cell battery. In general, the model allows a great degree of parameterization which produces results that are close to those obtained by physical measurements.

Simulations were performed using the same setup as in [Mikhaylov-14] to replicate the results, thus validating the simulation setup. In addition, more simulations were performed using different connection parameters. Especially, different values of the connection interval were used, as in [Gomez-12], with the purpose of correlate the results obtained from different authors.

As a conclusion, the accuracy of the simulation results is close to the values provided by TI in the datasheet of the CC2540 [Texas-14].

[3] OMNeT++, ver 5.0, SimulCraft Inc., 2001-2015, https://omnetpp.org/
[4] Mixim, ver 2.3, 2007-2011, http://mixim.sourceforge.net/index.html

2.2. C based simulation model

A C languagebased model[5] of the BLE protocol, called "BLEeMod", is presented by Kindt et al., [Kindt-14] and it was released under the GNU Lesser General Public License. The article explains in depth the protocol's power consumption and the algorithms used in the model to calculate it in every connection mode of the protocol. The authors also used a commercial BLE device based on the CC2540 transceiver from TI [Texas-14] to perform physical measurements in order to compare them with the results obtained from the simulation.

In the next subsections, the evaluation of the model based on the criteria specified at the beginning of this Chapter is described.

2.2.1 Easy of setup

The research study in [Kindt-14] contains valuable information to understand the required parameterization of the values used in the model. Along with that, the model's source contains a detailed documentation of every function of the model and how to use them. Additionally, the model includes an example used to measure the energy of every connection mode. It takes no more than 30 minutes to do the setup of the example, read the documentation, compile it and run the simulation. The only requirement is to understand how to use the gcc compiler in order to get the model working. The example provided is a good starting point and can be easily modified to use custom settings of the connection parameters. As a conclusion, the setup is quite easy.

2.2.2 Model accuracy

As with the Omnet++ model, this one can also be fed with operating data from existing BLE transceivers. In this case, the model also uses the parameters from the TI CC2540 BLE transceiver [Texas-14], which allows to perform a one to one comparison between both models.

Again, simulations were performed using the same parameters as specified by the authors in [Kindt-14] to corroborate the results obtained by them. The next round of simulations performed

[5] BLEeMod, Philipp Kindt, Daniel Yunge,Robert Diemer, Samarjit Chakraborty, 2013, https://www.rcs.ei.tum.de/forschung/wireless-sensor-networks/bleemod/

used the parameters from [Mikhaylov-14] to correlate the results from both models. The results

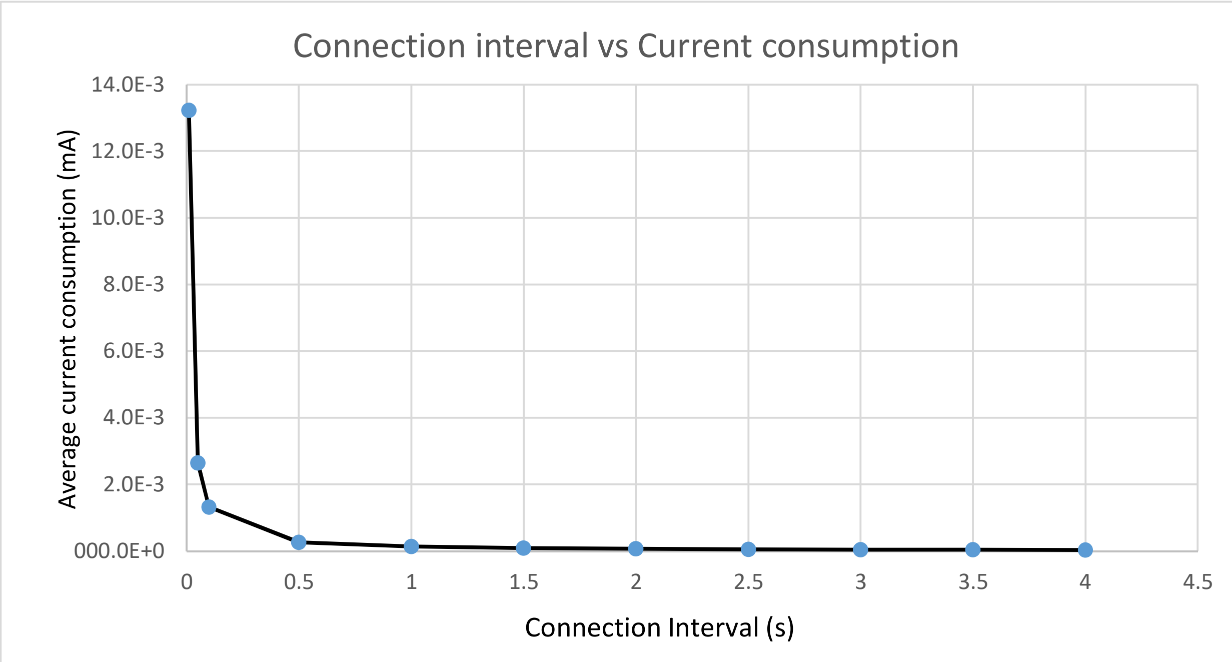

Fig. 2.1 Results obtained in simulation for C based model.

obtained, as shown in Fig. 2.1, are very close to the data reported by TI and both models have a very close correlation in their results.

It is important to realize that, although the CC2540 transceiver was used by both research studies, and that the hardware is cheap and easy to obtain, the software stack from TI requires a commercial compilation tool that is quite expensive and not available for academic licensing. Therefore, for the present work, it is difficult to correlate the physical measurements with both software model results.

2.3. Conclusion

This chapter presented an evaluation of two existing BLE simulation models. The purpose to evaluate them is to choose the simulation model with the best performance, ease of use and accuracy, in order to conduct experiments aimed to provide a design guideline of optimal BLE connection parameters to optimize the power consumption for different IoT applications.

Both models implement the protocol accurately and the results are also consistent with physical measurements performed by both authors. Considering that, the C based model is easier to use and runs faster than the Omnet++ counterpart, therefore the decision is to use the C based model for future experiments.

3. BLE power consumption measurements in commercial platforms

BLE is widely used for IoT applications and it is also included in the majority of the hardware development platforms available in the market. Moreover, BLE is also included in smartphones, tablets and laptops, which allows the IoT applications to use these devices as a gateway for Internet access. Additionally, the coming 5G cellular wireless technology will provide faster networks and the required means to connect billions of smart devices to the internet [Li-18], which will broaden the ecosystem where the IoT applications can be used, thus requiring the hardware to be very flexible. Furthermore, the tendency for the upcoming IoT hardware is that they will be powered by batteries, thus posing a great challenge to implement power efficient applications in order to optimize the battery life of the devices. In the rest of this chapter, the general architecture of a BLE hardware platform and the methodology to measure the power consumption of it is presented.

3.1. Architecture of an IoT platform with BLE

In general, a BLE system on a chip (SoC) block diagram architecture is shown in Fig. 3.1. As can be seen in the figure, the SoC contains a microcontroller unit (MCU) such as an 8051 in the TI CC2540 or an Intel Quark SE in the Intel Curie SE C1000 module. Similarly, products from NXP or Cypress uses an ARM Cortex central processing unit (CPU). Along with the MCU/CPU, the SoC contains analog to digital converters (ADC) and digital to analog converters (DAC) to collect data from sensors; a serial peripheral interface (SPI) to communicate with other peripherals; a universal asynchronous receiver transmitter (UART) to handle asynchronous communication with other serial devices; an inter-integrated circuit (I^2C) interface to communicate with the rest of the integrated peripherals; the input/output subsystem to interact with the outside world; and finally the BLE module.

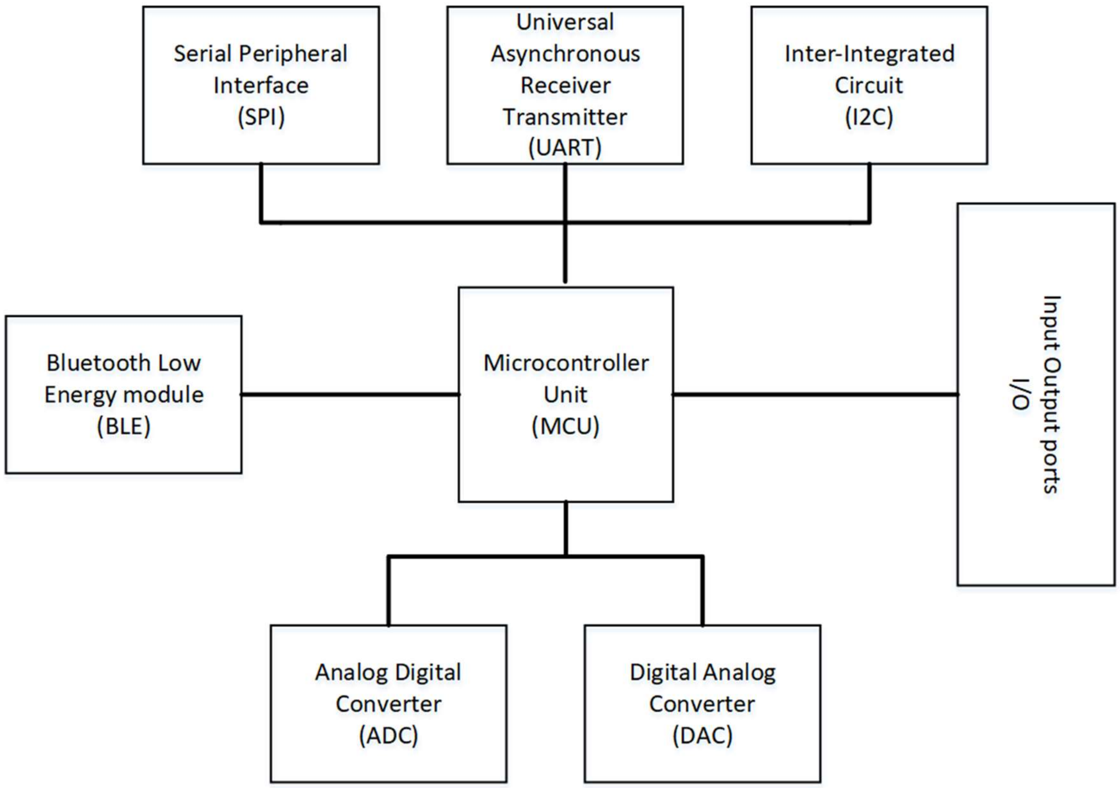

Fig. 3.1 BLE SoC architecture.

While these are the main components of a BLE SoC, there are vendors that also include more functionality in the SoC such as sensing capabilities, accelerometers, gyroscope, etc. to provide a more complete IoT solution. As a result, these devices are often included in development platforms to help in the implementation of prototypes of IoT applications.

Nowadays, there are different vendors that provide BLE SoC solutions along with their development platforms. The main vendors are:

1) TI with the following products in their catalog: CC2540 [Texas 2540-14], CC2541 [Texas 2541-19] and the CC2640 [Texas 2640-19].
2) Nordic Semiconductor with its nRF51822 SoC [Nordic-19].
3) NXP presents the KW31Z [NXP 31Z-19], the KW41Z [NXP 41Z-19], the QN908x [NXP 908x-19] and the QN902x [NXP 902x-19].
4) Cypress with its PSoC BLE [Cypress PSoC-19].

Usually, each vendor has a development kit available for each one of their BLE SoC to help users and IoT companies to implement prototypes before massive production. In the following

Fig. 3.2 Texas Instrument CC2540/41 mini development kit. Picture taken from [Texas-13].

subsections, the commercial platforms used to perform the power consumption measurements are presented.

3.1.1 TI CC2540 mini development kit

The TI CC2540 mini development kit [Texas-13]-is shown in Fig. 3.2. The kit contains a USB dongle and a keyfob. The BLE device used in the keyfob is the CC2540 [Texas 2540-14] which can be configured only as a PD. The USB dongle is configured as CD and is meant to be used under the Windows operating system. The required software to run the examples provided by TI is the BTool and some examples are ready to be flashed in the device. However, if any change is required to the examples to perform some experiments or a custom application is to be developed, the IAR embedded workbench [IAR-19] is required.

3.1.2 Intel Arduino 101

The Intel Arduino 101 (Intel A-101) platform is based on an Intel Curie SoC, which includes an Intel Quark microcontroller and a Nordic nRF51822 BLE module, among other components [Intel-17].

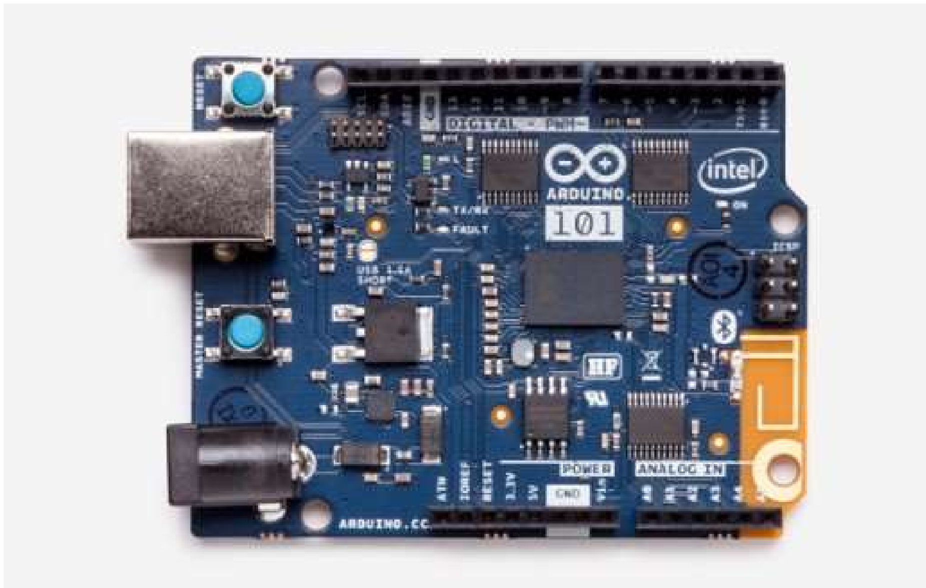

Fig. 3.3 Intel Arduino 101. Picture taken from [Arduino-17].

This platform is fully compatible with the popular Arduino UNO, and since Arduino is an open source platform, the board files and development software are open and free to use. The platform supports the Arduino IDE [Arduino-17] or the Zephyr Project [Zephyr-19], which allows the user to have two options to compile the firmware and compare the performance of the application. The Intel A-101 is shown in Fig. 3.3.

3.1.3 NXP FRDM-KW41Z

The NXP FRDM-KW41Z kit (NXP FRDM) contains two boards with the KW41Z SoC [NXP FRDM-18]. This device can be configured as either PD or CD. Additionally, the kit is compatible with Arduino UNO GPIOs and has an accelerometer and a magnetometer combo sensor. The development kit is shown in Fig. 3.4.

NXP provides all the printed circuit board (PCB) information and collaterals, as well as the development software, named MCUXpresso SDK, which includes the required stack to develop BLE applications. The software has a good number of examples, with its corresponding documentation, that can be easily modified to perform different experiments. All the

Fig. 3.4 NXP FRDM-KW41Z. Picture taken from [NXP FRDM-18].

documentation, software and collaterals can be downloaded from [NXP FRDM-18], previous registration. Additionally, NXP provides an application for mobile devices that contains several examples that allows a smartphone to connect with the development board for testing purposes [NXP IOT-17].

3.1.4 Cypress CY8CKIT-042-BLE-A Pioneer kit

The CY8CKIT-042-BLE-A (CY8CKIT) is shown in Fig. 3.5. [Cypress CY8CKIT-19] and is an Arduino compatible board which uses the PSoC BLE [Cypress PSoC-19] and the CY5677 CySmart BLE 4.2 USB dongle that can be connected to a Windows OS system for debug purposes. Cypress provides the board collaterals as well as the development software tool named PSoC creator [Cypress CY8CKIT-19]. The PSoC creator has several well documented examples that can be flashed in the board for experimental purposes. The device can be configured as CD or as a PD and there are examples for both cases. Additionally, Cypress provides a mobile application named CySmart that can be installed in a smartphone [Cypress CY8CKIT-19].

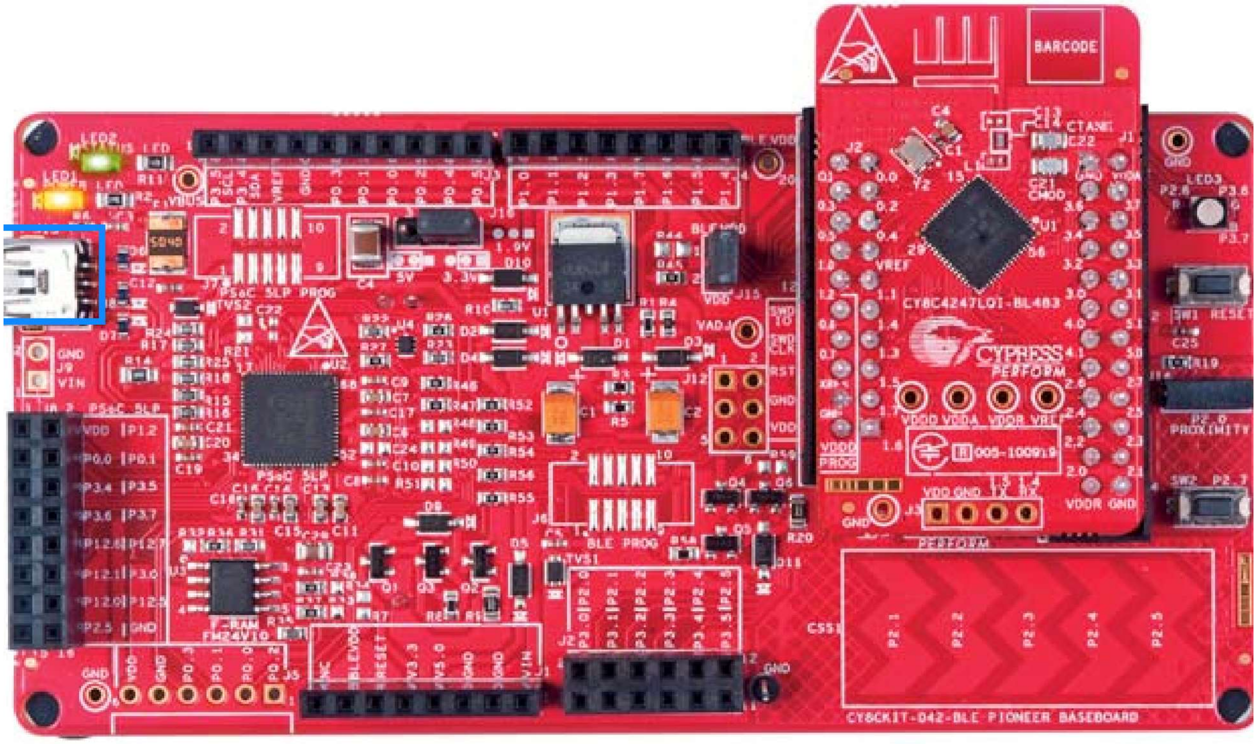

Fig. 3.5 CY8CKIT. Picture taken from [Cypress CY8CKIT-19].

3.2. Power consumption during a BLE communication establishment

In Section 1.2.1 of Chapter 1, it was explained why the power consumption of the BLE protocol is important to understand. However, it is not straightforward to measure it since there are different platform dependent power states, related to the SoC and its hardware that are involved in the communication establishment of a BLE master and slave devices. Moreover, the current consumption will vary in each power state, thus the requirement to measure the average current throughout the duration of the advertising and connection events and not just the peak current.

In what follows, the power states of a BLE device are described [Texas-14] and shown in Fig. 3.6:

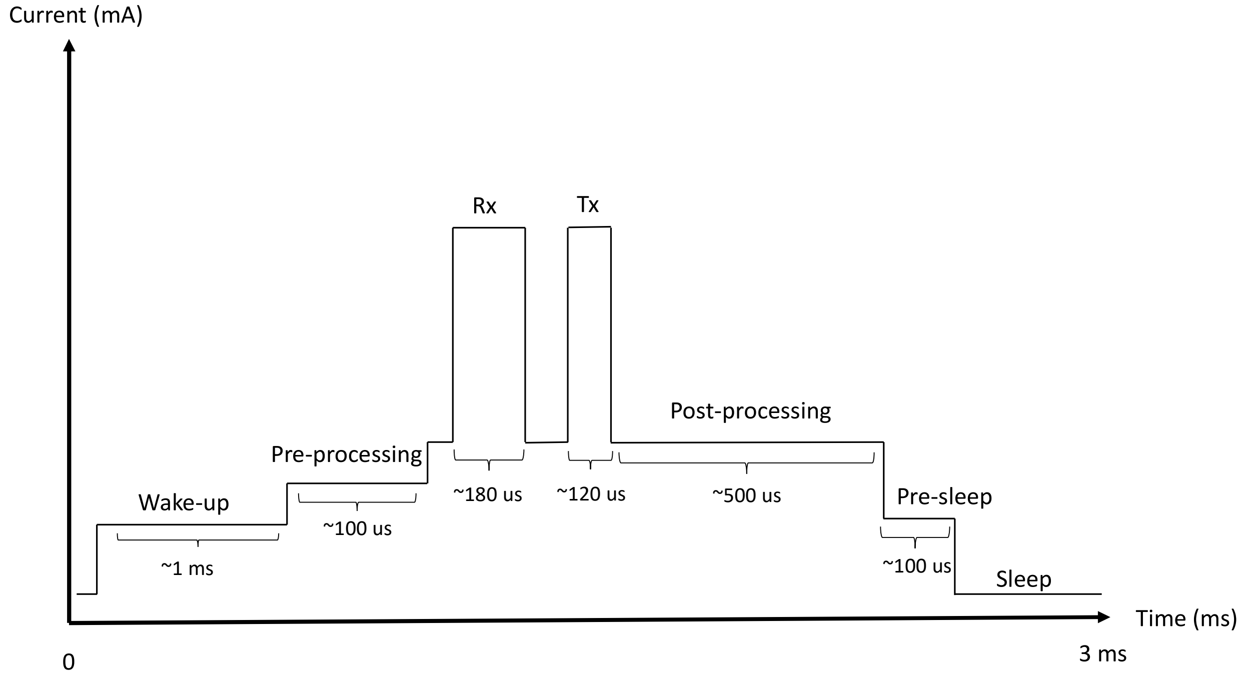

Fig. 3.6 Communication establishment states in a BLE device.

1) **Wake-up**.- When the voltage regulator (VR) is stable, the MCU will wake up consuming current as seen in the figure.
2) **Pre-processing**.- After the MCU is coming out of a low power state, it will perform the required tasks to prepare the radio to transmit (Tx) and receive (Rx) data packets.
3) **Rx**.- In this state, the systems performs a Rx operation.
4) **Tx**.- In this state, the system performs a Tx operation.
5) **Post-processing**.- Here, the MCU processes the received packets and the upper layer of the protocol will perform their required task as specified in [Bluetooth SIG-14].
6) **Pre-sleep**.- The system prepares to transition to low power mode.
7) **Sleep**.- The system is in idle until the next connection event.

These power states are present in every platform and their rate of occurrence will depend on the connection interval that is being set depending on the application. A plot of the power states, measured in a BLE device, is shown in Fig. 3.7 and an example of a connection event of 7.5 ms of duration is shown in Fig. 3.8. Additionally, the time duration of each power state will depend on the platform since the time of each state depends on the MCU and rest of the hardware of the

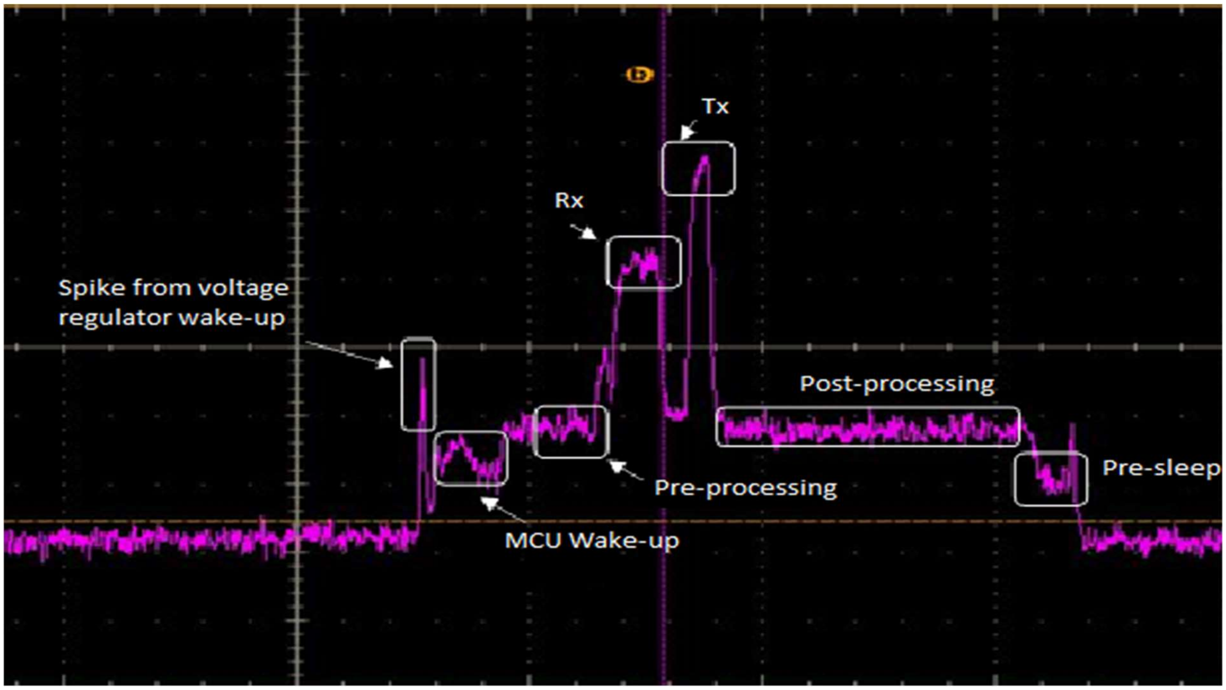

Fig. 3.8 Measurement of communication establishment power states in a BLE device.

platform.

Fig. 3.7 BLE connection event when the connInterval is set to 7.5 ms.

3.3. Power consumption measurement methodology

In order to implement an optimal low power IoT application, a hardware platform needs to be carefully chosen as the MCU, the surrounded hardware and the firmware will play an important role in the current draw of the system. In other words, the power consumption of the platform will depend on how efficient an MCU is in terms of entering/exiting low power modes and how easy is to modify the BLE parameters related to the advertising and connection events. Moreover, each platform might interact with different components and the BLE radio can be included in the SoC as an IP, or it might be in the platform as a separate component, thus impacting the power consumption measurements. Although there exist mathematical and software models (as the ones presented in Chapter 2), along with applications from vendors to predict the power consumption of a device, it is important to understand how to physically measure the power consumption of the platform and get the actual values of the platform under real life conditions. In this case, since there are different connection states and each one of them has different timing, an oscilloscope must be used to measure the average current consumption of the platform. In any case, it is not straightforward to take such measurements, thus a resistor in line with the power supply can be used to obtain the current consumption of the platform [Texas-14]. The value of the resistor should be small to prevent any effect in the platform circuitry and to allow the measurements; for that reason in [Texas-14] a 10 ohm resistor is suggested.

In this study, the four mentioned platforms in Section 3.1 were used to perform several experiments. Additionally, the power consumption measurements were taken using the oscilloscope Tektronix MSO5104.

3.3.1 Power measurement methodology

The first step to measure the power consumption is to perform a simple hardware modification to the boards, as in this case, there was no current probe available. The hardware modification consists to add, or replace in certain cases, a 10 ohm resistor to the voltage input to obtain the current consumption in each connection establishment power states. The reason to

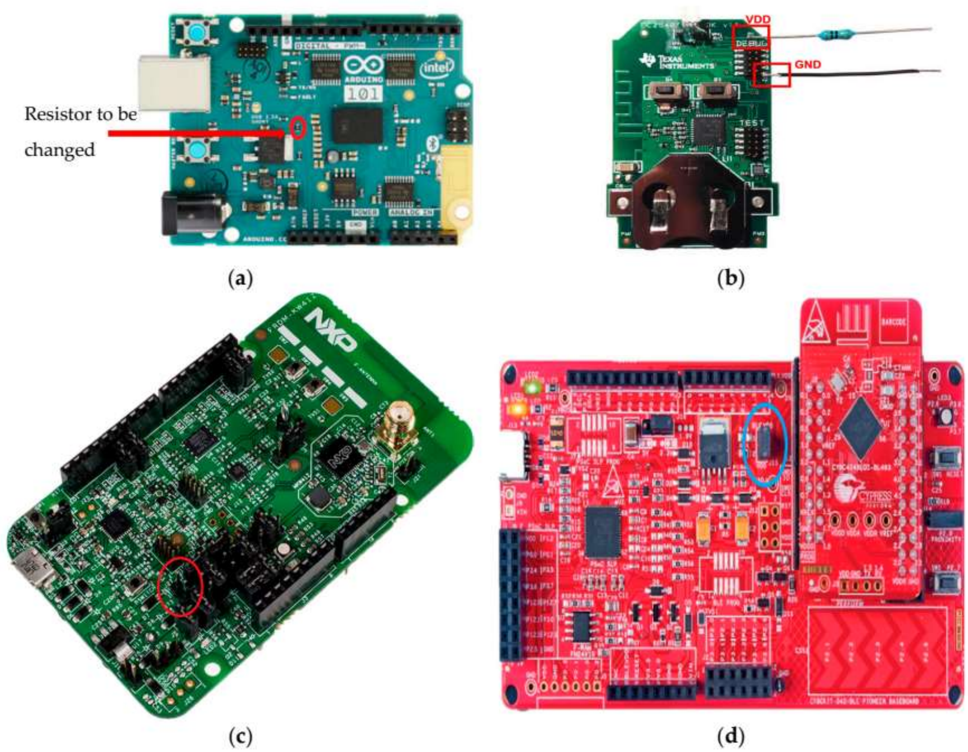

Fig. 3.9 The four platforms used in the experiments and the required hardware modifications: (a) Intel A-101, red circle shows the location of the resistor to be changed; (b) TI CC2540; (c) NXP FRDM, red circle shows the location of the jumper where the resistor can be placed; and (d) Cypress CY8CKIT, where the blue circle shows the location of the jumper where the resistor can be placed. Figure taken from [Garcia-Espinosa-18].

choose the required modifications [Garcia-Espinosa-18] are as follows:

1) Intel A-101 requires replacing a zero-ohm resistor (R58) by a 10 Ω resistor, as shown in Fig. 3.9a.
2) The TI CC2540 requires to add a 10 Ω resistor, as shown in Fig. 3.9b. The figure is taken from Reference [Texas-14].
3) NXP FRDM can be powered by a coin cell, in which case, a 10 Ω resistor can be placed in the jumper J27, as shown in Fig. 3.9c.
4) CY8CKIT requires a 10 Ω resistor in jumper J15 as shown in Fig. 3.9d.

The second step is to load the application firmware in both the CD and PD. In this case, the software configures the devices to perform the advertising event, followed by the connection event.

In order to get familiar with the software stack from each platform, the following setup was used along with the built-in examples provided in the software stack for each platform [Garcia-

Espinosa-18]:

1. A smartphone configured as the CD, with the CySmart application and one Cypress CY8CKIT platform configured as a PD.
2. A smartphone configured as the CD, with the IoT Toolbox application from NXP and one NXP FRDM platform configured as PD.
3. An Intel A-101 configured as the CD and one CC2540 and another Intel A-101 boards were used as PDs in a one-to-one configuration.
4. A Cypress CY8CKIT configured as a CD and the rest of the platforms as PDs.
5. Finally, two NXP FRDM platforms were used, one configured as a CD and the other one as a PD.

The third step is to perform the current measurement of the advertising and connection events. For this step, an oscilloscope trigger is configured to capture the beginning of an advertising event and another trigger is configured to capture the connection event. Once the captures are obtained, the data set is saved for further processing.

The fourth and final step is to calculate the power consumption by means of a Matlab script that processes the saved data set file obtained in step 3 and stores the values in a vector. The script is implemented in the following way:

Let S be a vector, with all the samples taken from the oscilloscope measurements, defined as

$$S = a_0 a_1 \dots a_n \tag{3-1}$$

S is split into sections a_0, a_1, until a_n. Each a_n section has N elements which corresponds to a power state as shown in Fig. 3.10. Additionally, each section has an associated time period T_S. The total time of an event T_E is calculated with

$$T_E = T_{a_0} + T_{a_1} + \dots + T_{a_n} \tag{3-2}$$

where

$$T_{a_n} = N T_S \tag{3-3}$$

The current of each state is calculated with

$$I_{a_n} = \sum_{i=0}^{N-1} \frac{a_n(i)}{r} \tag{3-4}$$

where r is the 10 ohm resistor added in the platforms.

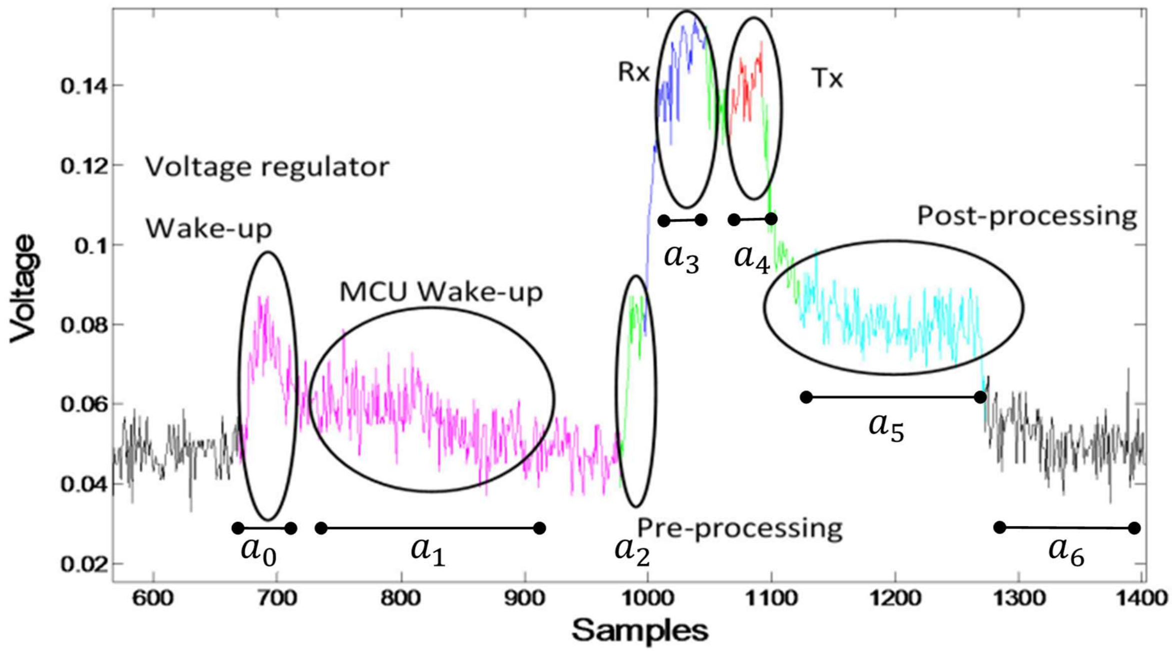

Fig. 3.10 Samples from a connection event for processing in the Matlab script.

The average current I_{avg} is the sum of the current draw of each state considering also the sleep current during an advertising or a connection and is obtained with

$$I_{avg} = I_{a_1} + I_{a_2} + \ldots + I_{a_n} \tag{3-5}$$

The power consumption of each state is obtained by

$$P_{a_n} = \sum_{i=0}^{N-1} \frac{a_n(i)^2}{r} \tag{3-6}$$

The total power consumption P_E of an event, either advertising or connection, is computed by

$$P_E = P_{a_1} + P_{a_2} + \ldots P_{a_n} \tag{3-7}$$

Finally, the battery lifetime in hours is given by

$$T_{bat} = \frac{C_{bat}}{I_{avg}} \tag{3-8}$$

where C_{bat} is the battery capacity in milli-amperes per hour (mAh).

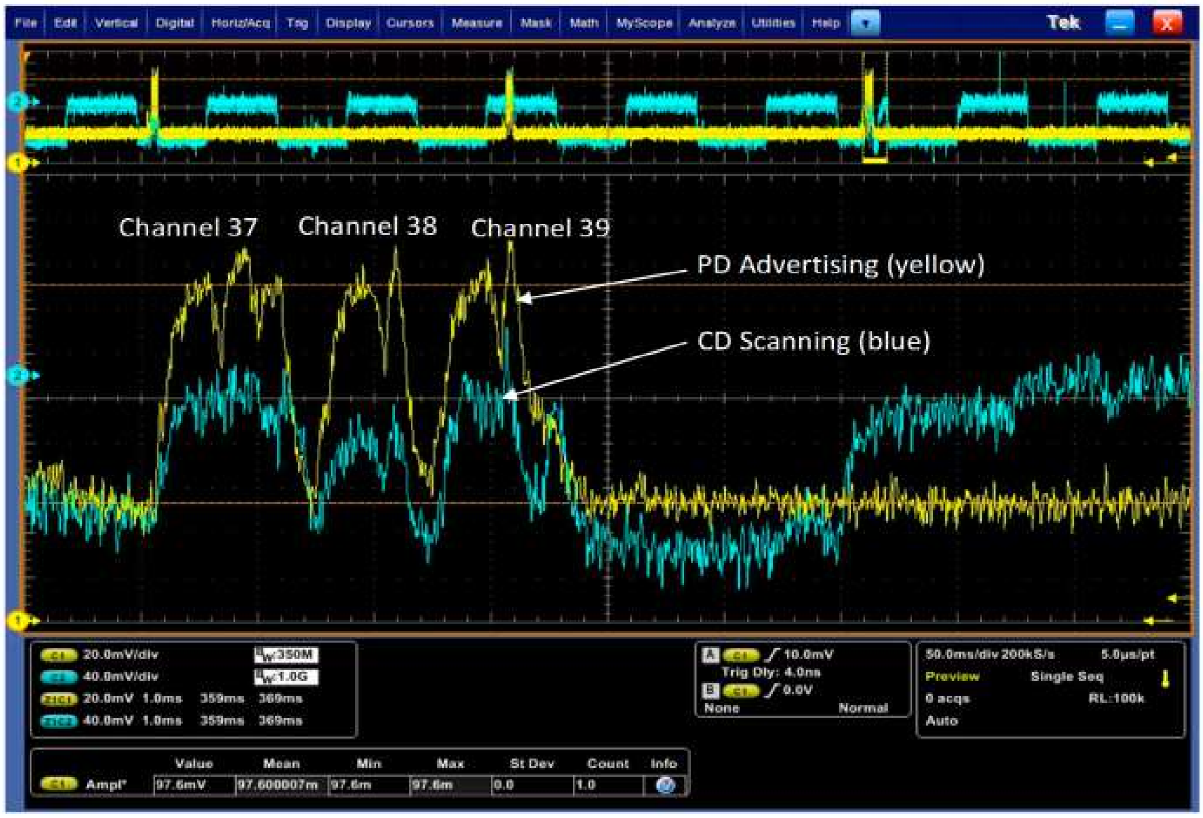

Fig. 3.11 Advertising event waveform for Intel A-101. Figure taken from [Garcia-Espinosa-18].

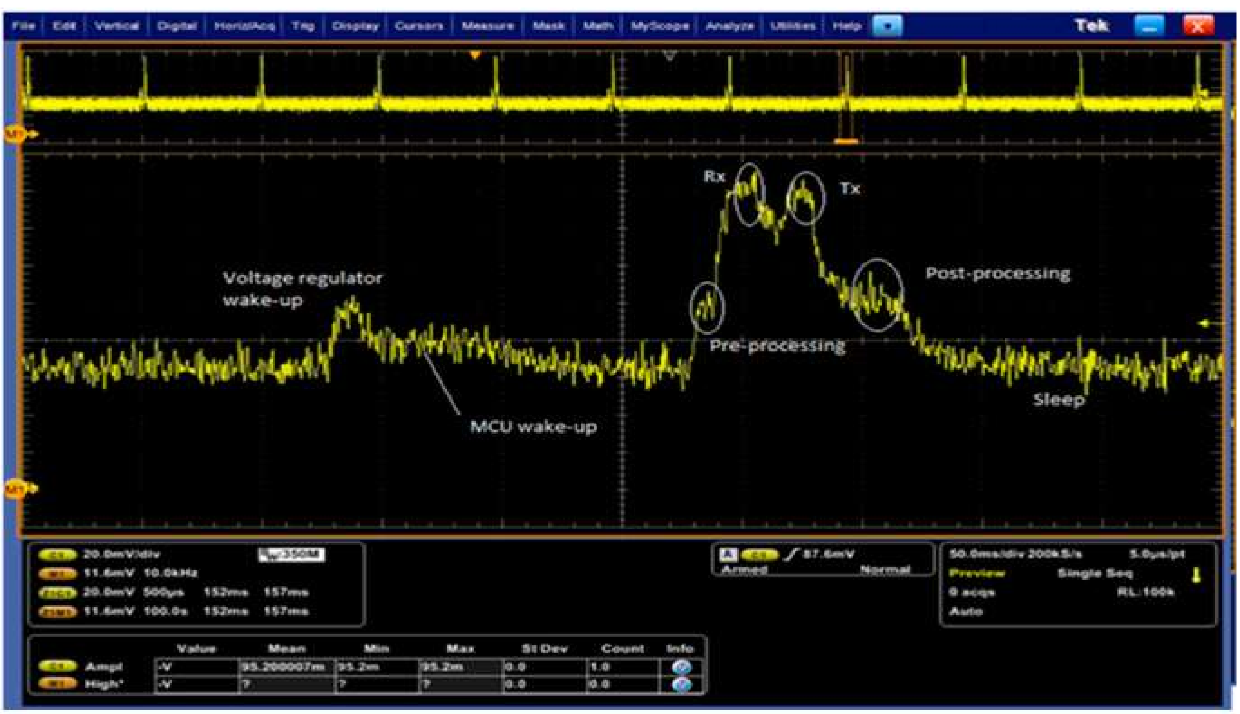

Fig. 3.12 Connection event waveform for Intel A-101. Figure taken from [Garcia-Espinosa-18].

3.4. Results

In Fig. 3.11, the obtained waveform from an advertising event is shown. In this case, the PD is advertising its presence while a CD is scanning the advertising packets in the same channels. Once the CD finds a PD, it will send a connection request and if the PD acknowledges such a request, a connection is established and periodic connection events, set by a connection interval parameter, will be set to transmit data [Garcia-Espinosa-18].

In Fig. 3.12, the obtained waveform from a connection event in an Intel A-101 is shown. The shape of the waveform is the same in the rest of the platforms, except of course, for the duration and the voltage amplitude of the states. In this case, the figure shows the waveform after the CD has established a connection with a PD which will be repeated depending on the selected connection interval of the application.

The steps to measure the power consumption presented in the previous section were performed to obtain the power consumption of the four development boards. The current consumption of the platforms configured as PD using an interval connection of 100 ms is presented in Table 3.1. Likewise, the current consumption of a connection event for a platform configured as CD is shown in Table 3.2. Important to note is that the CC2540 does not support a CD configuration, therefore the comparison is performed with the other three platforms. Additionally, the power consumption of a device configured as CD is not presented in the datasheets nor in any application note from any of the vendors. The average power consumption of the platforms configured as PD and CD in a connection event is presented in Table 3.3.

Extrapolating the values for different connection intervals, an approximation of the battery life associated to a connection interval for each platform configured as a PD is shown in Table 3.4. The calculation of the battery life is based on a CR-2032 battery, which has a typical capacity of 235 mAh [Garcia-Espinosa-18]. Table 3.5 shows the approximation of the battery life associated to a connection interval for each platform configured as CD.

TABLE 3.1. CURRENT CONSUMPTION COMPARISON BETWEEN THE PLATFORMS CONFIGURED AS PD USING A CONNECTION INTERVAL OF 100MS

State	I_{avg} (mA) *CC2540*	I_{avg} (mA) Intel A-101	I_{avg} (mA) CY8CKIT	I_{avg} (mA) NXP FRDM
Wake-up	5.69	0.77	2.11	3.73
Pre-processing	8.80	2.56	2.28	2.34
Rx	19.37	8.99	13.83	5.63
Tx	23.81	8.93	15.97	8.54
Rx-Tx	10.79	8.68	15.81	7.55
Post-processing	7.89	2.25	5.71	5.02
Total average current	0.279	0.067	0.141	0.118
Expected battery life (days)	35	145	69	82

TABLE 3.2. CURRENT CONSUMPTION COMPARISON BETWEEN THE PLATFORMS CONFIGURED AS CD USING A CONNECTION INTERVAL OF 100MS

State	I_{avg} (mA) Intel A-101	I_{avg} (mA) CY8CKIT	I_{avg} (mA) NXP FRDM
Wake-up	0.517	0.194	0.304
Pre-processing	3.95	0.573	0.704
Rx	9.02	1.48	1.97
Tx	7.97	1.8	1.48
Rx-Tx	7.37	1.39	1.77
Post-processing	2.14	0.63	1.03
Total average current	0.089	0.018	0.036
Expected battery life (days)	109	514	267

TABLE 3.2. POWER CONSUMPTION OF THE PLATFORMS CONFIGURED AS PD AND CD USING A CONNECTION INTERVAL OF 100MS AND A 3.3V POWER SUPPLY

Configuration	CC2540 (mW)	Intel A-101 (mW)	CY8CKIT (mW)	NXP FRDM (mW)
PD	0.837	0.201	0.423	0.354
CD	N/A	0.267	0.054	0.108

TABLE 3.3. CURRENT CONSUMPTION AND EXPECTED BATTERY LIFE FOR EACH PLATFORM CONFIGURED AS PD FOR DIFFERENT CONNECTION INTERVALS

CONNECTION INTERVAL (ms)	I_{avg} (mA)				Battery Life (in Days)			
	CC2540	Intel A-101	CY8C KIT	NXP FRDM	CC2540	Intel A-101	CY8C KIT	NXP FRDM
10	2.78	0.66	1.404	1.176	3.5	14	7	8
50	0.558	0.130	0.281	0.236	17	73	35	41
100	0.279	0.067	0.141	0.118	35	145	69	82
300	0.093	0.023	0.047	0.04	104	423	205	243
500	0.056	0.014	0.029	0.024	172	686	337	399
800	0.035	0.009	0.018	0.015	273	1054	528	624
1000	0.028	0.007	0.015	0.012	339	1283	651	767
2000	0.014	0.004	0.008	0.006	655	2269	1221	1423
3000	0.01	0.003	0.005	0.004	951	3050	1725	1990
4000	0.007	0.002	0.004	0.003	1229	3684	2171	2485

TABLE 3.4. CURRENT CONSUMPTION AND EXPECTED BATTERY LIFE FOR EACH PLATFORM CONFIGURED AS CD FOR DIFFERENT CONNECTION INTERVALS

Connection Interval (ms)	I_{avg} (mA)			Battery Life (in Days)		
	Intel A-101	CY8CKIT	NXP FRDM	Intel A-101	CY8CKIT	NXP FRDM
10	0.892	0.177	0.357	11	55	27
50	0.179	0.036	0.072	54	270	135
100	0.089	0.018	0.036	109	525	267
300	0.029	0.006	0.012	326	1424	760
500	0.019	0.004	0.008	544	2163	1204
800	0.011	0.003	0.005	871	3056	1793
1000	0.008	0.002	0.004	1089	3543	2144
2000	0.004	0.001	0.0027	2178	5204	3518
3000	0.002	0.0015	0.0021	3268	6167	4473
4000	0.001	0.0014	0.0018	4357	6796	5176

3.5. Conclusion

In this chapter, an architecture overview of a BLE SoC and examples of commercial development platforms were presented. Additionally, a methodology to perform physical power consumption measurements using four different commercial platforms was shown. The results presented in section 3.4 are useful to predict the battery life of an IoT application based on the connection interval parameter of the BLE protocol. Another key point is that the power consumption measurements for a commercial device in a CD configuration was presented. For the best knowledge of the author, these contributions and data are not published in open literature nor in the application notes from the vendors [NXP—17]. Therefore, these data are useful to achieve a dependable design when the CD needs to be powered by batteries that might not be easily replaced.

In the next chapter, the implications of using short connection intervals in the battery life are presented. Additionally, a design guideline and examples of IoT applications based on the BLE protocol are presented.

4. A design guideline for maximizing the battery life of a BLE device

In the previous chapter, the power consumption of four commercial platforms was presented, and from the results, the Intel A-101 outperforms the rest of the platforms with the lowest power consumption when configured as a PD. On the other hand, when the platforms are configured as a CD, the Cypress CY8CKIT has a better performance. In any case, there are different factors that might affect the power consumption of a device. As an example, the MCU/CPU plays a significant role, since depending on the application, it might be performing different tasks such as controlling other peripherals. Moreover, the software and firmware stacks will also play a significant role, since they need to provide the means to implement the required low-power modes under different scenarios [Garcia-Espinosa-18].

As mentioned in Chapter 1, the BLE specification defines the means for a device to transfer data through the Generic Attribute Profile (GATT) [Bluetooth-18]. As an example, a profile describes the use case and the roles that a CD and PD will have. In general, a PD will be connected only to one CD at a time meaning that once a CD discovers a PD, the device will stop advertising and will only be available to transfer data to one master device in every connection interval. Thus, it is relevant to understand the power consumption and the expected battery life for different connection interval values, which are required to be defined depending on the data transfer needed for a given application, as specified in a profile [Garcia-Espinosa-18]. A list of defined BLE profiles can be consulted in [Bluetooth-18].

The power consumption of a BLE CD is often minimized due to the assumption that a smartphone or a tablet, which are typically used as CDs, can be charged more easily than a PD. While this can be true for different applications, there might be other use cases, such as in the smart farming [SysAgria-19], where this assumption is incorrect. As an example, sensors can be placed across a crop field to sense different variables such as humidity, lightning, temperature among others, to detect a crop disease. Thus, the sensors (PDs) as well as the CD need to be powered by batteries that might not be easily replaced. Therefore, in order to provide a general design guideline for an optimal implementation of a BLE based application, the power consumption and expected battery life metrics for both the CD and PD, becomes relevant [Garcia-

Espinosa-18].

In what follows, the different areas that can be optimized to achieve low power consumption while optimizing the battery life is presented.

4.1. The battery capacity role in a BLE device

Since the BLE protocol was designed to remain in a sleep state for the majority of the time and only to wake up to transmit data or to maintain a connection [Gomez-12], it is important to understand the implications of using a coin cell battery when designing an IoT application. During the sleep state, the devices will typically consume few uA since the MCU and peripherals, including the BLE radio, are turned off. However, when a device wakes up, the current peak drawn by the BLE radio waking up and performing Rx and Tx operations, will typically consume above 10 mA by a few micro-seconds as shown in Table 4.1.

TABLE 4.1 CURRENT CONSUMPTION DURING A CURRENT PEAK DRAWN

AVERAGE TIME DURATION (us)	I_{avg} (mA)			
	CC2540	Intel A-101	CY8CKIT	NXP FRDM
80	18	15	23	17

This current peak is known as high current pulse (HCP) [Furset-11]. In this case, the typical continuous current drain specified in a coin cell battery (~190 uA in a CR2032) [Energizer-18], will be surpassed. In other words, the battery capacity will start degrading when the load current increases and the battery voltage will start dropping, as shown in Fig. 4.1 (taken from [Furset-11]). In such results, the battery capacity follows the typical value specified in [Energizer-18] when the current is 500 uA, but when the current draw is 3 mA, the battery capacity decreases to ~150 mAh as shown in Fig. 4.2.

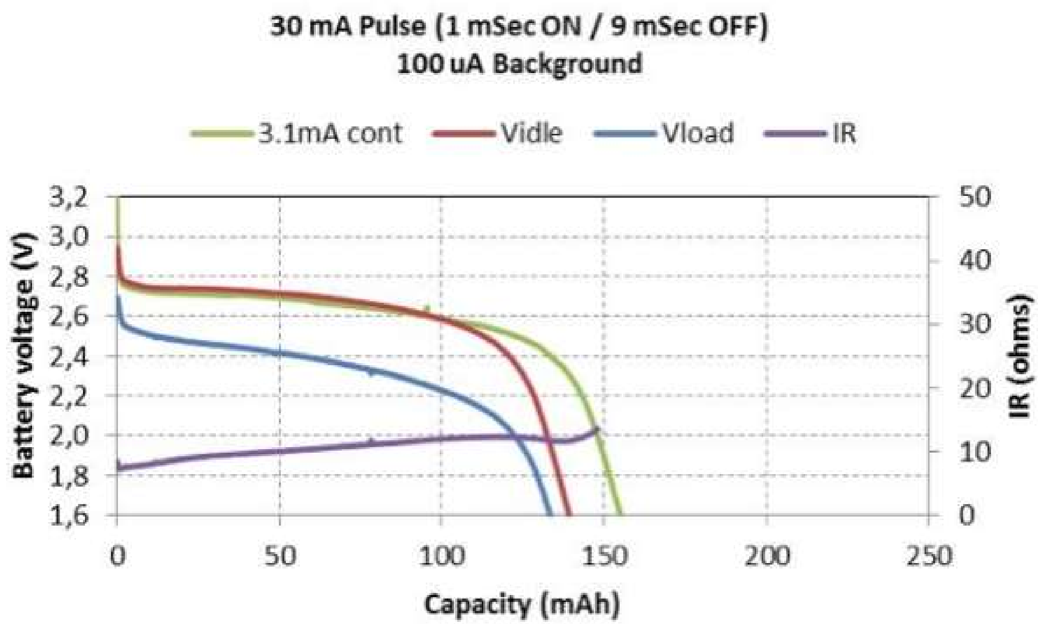

Fig. 4.1 Battery drain vs pulse duty cycle for a CR2032 coin cell battery. Figure taken from Fig. 5a in [Furset-11].

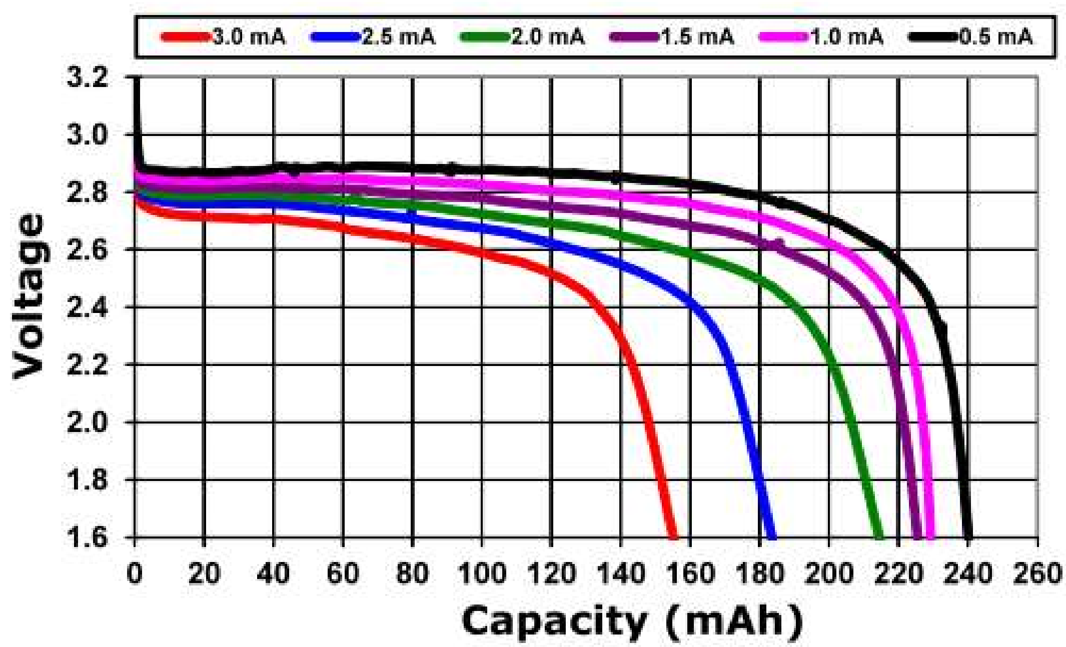

Fig. 4.2 CR2032 coin cell battery continuous discharge patterns. Figure taken from Fig. 3 in [Furset-11]

Another key point to consider is the functional end point (FEP), defined in [Furset-11], which is the minimum voltage level required by the platform circuitry to operate. Thus, an IoT application will need to take into account these factors to either select the appropriate battery or to forecast the duration of it for an application. In other words, the effective battery capacity will need to be calculated based on the requirements of the application in terms of the FEP and the duty cycle of the HCP. The HCP duty cycle becomes relevant when a short advertising or connection interval is set. For example, consider a 10 ms connection interval and a HCP of 30 mA with a duration of 1 ms. In this scenario, the device will be in sleep mode for 9 ms and will wake up at the 10th ms with a current consumption peak of 30 mA. When this load occurs every 10 ms, the battery capacity is heavily impacted, as can be seen in Fig. 4.1 where the battery capacity drops to 150 mAh at 2.0 V of FEP.

To illustrate this effect in a BLE connection, Fig. 4.3 shows the load current and the battery output voltage for a connection event of 50 ms, where the HCP can be observed as well as the impact that it has in the battery output voltage. In this example, the battery output voltage recovers after 3 ms. In other words, the output voltage is fully recovered when the next connection event happens. However, if the connection interval is 7.5 ms (the minimum value allowed by BLE specification), it becomes clear that the battery capacity will be heavily impacted [Garcia-Espinosa-18].

According to [Furset-11], the battery life estimation can be calculated by modifying Equation (3-8) by

$$T_{bat} = \frac{C_{corrected}}{I_{avg}} \tag{4-1}$$

where $C_{corrected}$ is the corrected battery capacity estimated by the test results performed in [Furset-11]. Following such test results and their recommendation, the $C_{corrected}$ parameter should be used when lower values of connection intervals are required, mainly from 7.5 to 25 ms since for longer time intervals, the battery will have enough time to recover. Thus, Table 4.2 shows the updated values for the platforms configured as a PD and with a corrected battery capacity of 190 mAh, which was obtained as the mean value from the test results and curves obtained in [Furset-11].

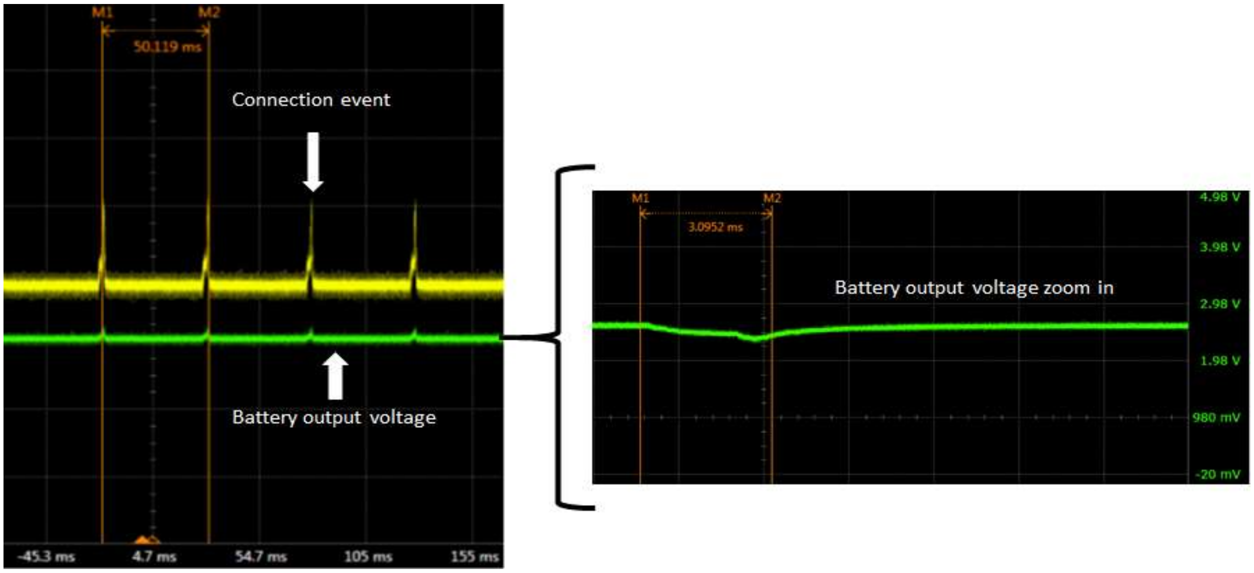

Fig. 4.3 Battery output voltage drop during a BLE connection event.

TABLE 4.2 CURRENT CONSUMPTION AND EXPECTED BATTERY LIFE FOR EACH PLATFORM CONFIGURED AS PD FOR DIFFERENT CONNECTION INTERVALS AND A CORRECTED BATTERY CAPACITY OF 190 MAH

CONNECTION INTERVAL (ms)	I_{avg} (mA)				Battery Life (in Days)			
	CC2540	Intel A-101	CY8C KIT	NXP FRDM	CC2540	Intel A-101	CY8C KIT	NXP FRDM
7.5	3.43	1.12	1.87	1.56	2.3	6	4.2	5
10	2.57	0.66	1.40	1.17	3	8.4	5.6	6.7
15	1.72	0.51	0.93	0.78	4.6	12	8.5	10
20	1.29	0.35	0.70	0.58	6	16.7	11.2	13.4
25	1.06	0.23	0.56	0.47	7.6	21	14	16.8

Fig. 4.4 shows a flow diagram of the methodology to calculate the corrected battery capacity.

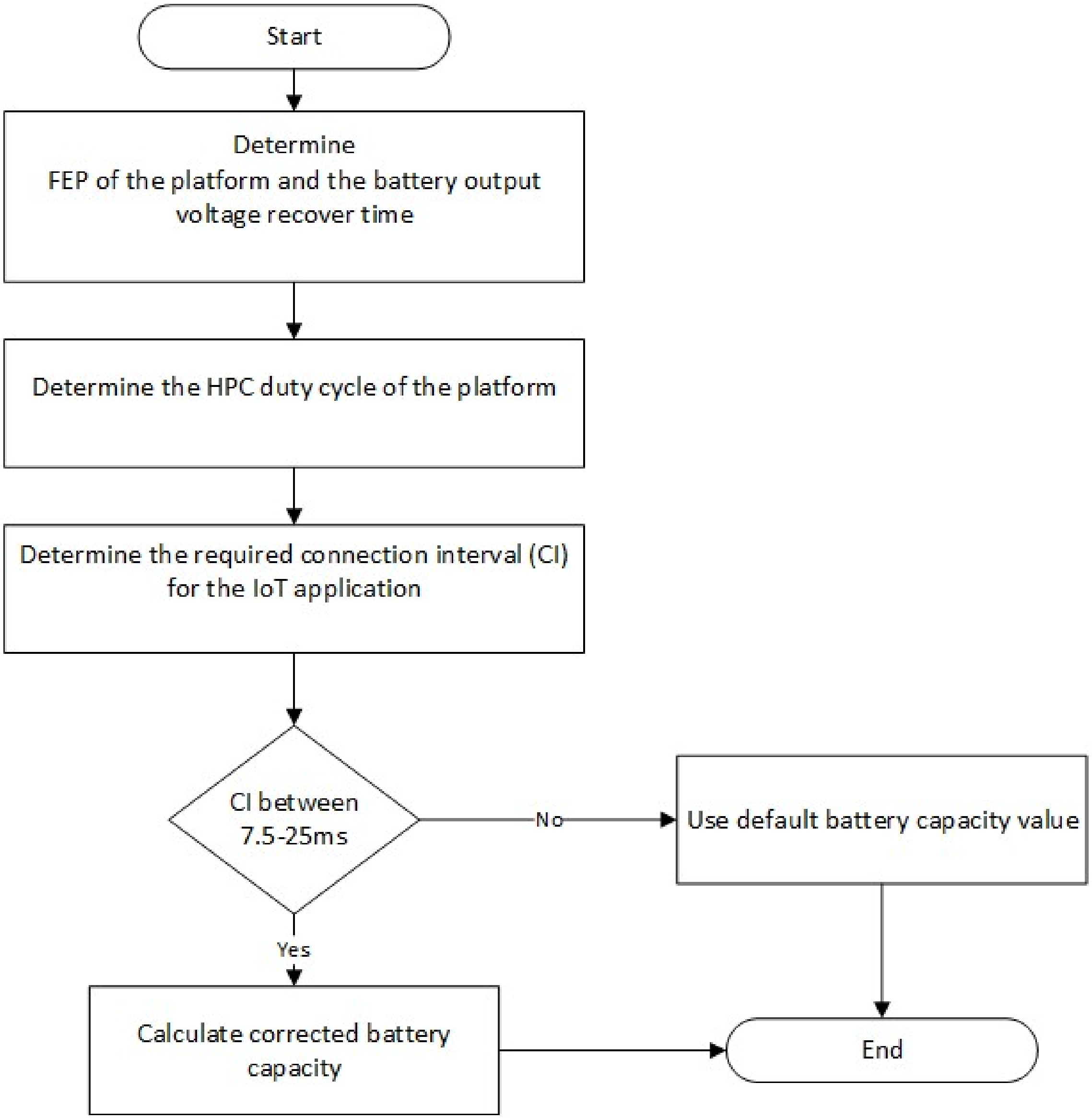

Fig. 4.4 Flow diagram of the methodology to calculate the corrected battery capacity. -

4.2. BLE connection parameters adjust

Estimating the battery life for IoT applications depends on several other parameters apart from the peak current caused by the wake up and Rx and Tx operations, such as data packet size, MCU operations and peripherals required for a given application. In this case, the connection parameters of the BLE protocol, such as advertising and connection intervals plus slave latency, need to be fine-tuned to achieve the required performance. Additionally, when designing a low power application, the processing of information and the use of peripherals should be performed in a way that the current peak does not increase while performing the Rx and Tx operation [Garcia-Espinosa-18]. The BLE specification defines several parameters that can be modified, however the main parameters that can be easily modified are:

1) Advertising interval. It is the time between advertising events. The allowed values in the BLE specification are from 20 ms to 10.24 seconds.
2) Connection interval. Determines how frequently the CD will communicate with the PD. The allowed values ranges from 7.5 ms to 4 seconds.
3) Connection supervision timeout. This is the maximum time allowed between two received data packets before a connection is considered lost by the CD. It ranges from 100 ms to 32 seconds.
4) Slave latency. It specifies the number of connection events that the PD can ignore the CD without risking the connection, thus maintaining the low power consumption. There is no limit in the values except that it cannot be longer than the connection supervision timeout. Usually, the default value in the devices is 0, which means that the PD is not allowed to skip a connection event.

As explained in Chapter 1, there are three services that BLE provides: data transfer, location and device networks. Additionally, the GAP specifies four roles: broadcaster, observer, central, peripheral. For each one of these services, there are different BLE profiles that can be used and thus, different connection schemes requiring certain parameter values as is shown in the next sub-sections.

4.2.1 Data transfer applications

Nowadays, wearables have become part of the daily life. They are used to measure physiological variables meant to track data to improve health or athletic form and they are now using artificial intelligence to make a better use of the collected data. Some examples are heart rate monitors and fitness trackers [Polar-18], biometric clothes with embedded sensors [Hexoskin-18], smart shoes [Sensoria-18], etc. These wearable products are collecting data from a wide range of sensors such as accelerometers, gyroscopes, heart rate monitors, among others, to track fitness, training progress, and injury risks. They also allow to determine the walk or run cadence, the foot landing technique, impact forces, the watts delivered, the sweat ph level and the foot ground contact time, along with heart rate monitoring. Thus, the collected data is used for customized training plans and to reduce the risk of injuries.

For certain of these applications, such as temperature sensors or heart rate monitors, the device can be configured as a broadcaster, which is a GAP role optimized for transmit-only applications. In this case, the device will send advertising packets with data, therefore the only parameters to be considered are the advertising interval and data length and no receiver operations will be required to be implemented in the application. On the other hand, when an application requires to send large amount of data, the GAP role would be central and peripheral since it will be required to establish a connection and transfer data periodically. In this case, the battery life estimation should follow the provided guidelines provided in the previous section if low values of connection interval are required. Additionally, if the application is required to send long burst of data, the connection interval needs to be adjusted to avoid the battery stress and additional tasks might be required, such as compressing the data before transmitting, reducing the usage of other peripherals in the platform while transmitting, and clearly define how often the data needs to be transferred in order to adjust the connection supervision timeout and slave latency accordingly.

4.2.2 Location service applications

A beacon is a device that transmits a signal that allows another device to determine its proximity with the broadcaster, thus enabling the technology for location service applications. ABI Research[6] forecasts that BLE beacon shipments will increase from 4 million in 2015 to 367 million by 2022, being this BLE service the one with highest growth rate. Examples of location service are:

1) Asset tags used to track the location and usage of diverse assets, such as inventory tracking.
2) People tags used to track the movement and location of employees or visitors in a given location, e.g., to locate a doctor in a hospital or a patient.

[6]ABIresearch for visionaries, https://www.abiresearch.com/

3) Condition tags to monitor environment variables, such as temperature, humidity, etc. in order to take actions based on the obtained data.
4) Proximity tags for museums or retails stores to provide content information for visitors.

In order to implement these applications, the devices requires to be configured as a broadcaster and observer, where the broadcaster will send advertising packets with information related to its identity (type of beacon or company identification), values to distinguish multiple beacons within a facility (such as a hospital or a factory) and a way to determine the distance between the beacon and the object. Since the device is configured as a broadcaster, the receiver functions are disabled as no connection to a CD is required. Thus, the advertising interval is the main parameter to be considered. In general, vendors use values ranging from 100 ms to 1 second. However, in [Kontakt-17] it is determined that the optimal value for the advertising interval is 350 ms.

4.2.3 Device network applications

Wireless mesh networks (WMN) have been around for some years and other communication protocols, such as ZigBee or Z-wave, have been used for IoT applications, thus representing a strong competition for BLE, whose mesh capabilities are relatively new in the market. The main usage of WMN are in smart cities, smart home, farming, and the industrial IoT. Additionally, BLE defines models, where a model is a software module that define the behavior of a BLE mesh device. There are four models: lightning, sensors, timing and scenes, and generics [Bluetooth SIG sd-17]. Some examples of applications are:

1) Lightning - smart lightning, which are lighting bulbs that can be controlled through a smartphone application (iOS or Android). The control consists in changing the light color and intensity, preset the timing to turn on and off the light, and presence detection to power on a light [Ilumi-18]. The bulbs are connected in a mesh network.
2) Generics – goTenna is an example of generics model. This company offers a device that creates a low-frequency radio network, with ranges up to four miles, which can be paired with a smartphone through BLE and allows users to send messages and GPS location to other users. Using the capabilities of a mesh network, a group of friends can pair their devices with their smartphones and be able to be in touch even if there is no

cellular network available (if hiking in the wilderness) or voice/data plan (if travelling to another country) [goTenna-18].

3) Sensors – Lumenradio is an example of these models. This company provides the hardware and operating system required to implement mesh networks for multiple business segments, such as Industry 4.0, smart buildings, and smart lighting. The sensors can detect and report events such as occupancy of a production line, occupancy of a meeting room, etc.
4) Timing and scenes – A WMN can memorize particular scenes. This is, a meeting room can be heat up and adjust the lightning at a certain time of a day just before a staff meeting is about to start. The memorized scenes can be executed according to a tme schedule.

BLE uses the flooding mesh technique, which consists of multiple relay nodes that can forward packets from each node to every other node, meaning that no routing tables are required. However, one of the main issues with this approach is the power consumption of the nodes in the network, since the devices need to be in receive operation mode constantly, therefore consuming more power that reduce the battery life. BLE is addressing this problem using the "Managed Flooding Mesh Network" which has features to reduce the power consumption and increase the reliability [Woolley-17]. In view of the recent introduction of mesh networking BLE specification, several research opportunities exist in this area to propose different techniques to achieve low power consumption.

4.3. Conclusion

This chapter presented examples of BLE applications and a power consumption design guideline to implement them. The main considerations are how to estimate the battery life under low values of connection intervals and a table was provided with different connection interval values and a corrected battery capacity. Additionally, four BLE connection parameters were identified as the ones that can be optimized to achieve a low power application.

4. A DESIGN GUIDELINE FOR MAXIMIZING THE BATTERY LIFE OF A BLE DEVICE

5. BLE evolution and market challenges

BLE was first introduced in 2010, in the Bluetooth core specification 4.0. Since then, BLE has evolved and, as of 2019, four revisions of the spec have been released. In the next section, the evolution of the protocol is explained.

5.1. BLE specification evolution

The first low energy version of the protocol was introduced in July 2010, in the 4.0 version, bringing a major change from the classic Bluetooth: a new modulation scheme and a new link layer optimized for low power, thus making this low energy version incompatible with classic Bluetooth devices. This is, a BLE device cannot be paired with a traditional Bluetooth device. For this reason, CD devices, such as a PC or smartphone, have support for both classic Bluetooth and BLE devices.

The next release of the BLE specification was in December of 2013, the 4.1 version, which has no big architectural or hardware changes but improvements to the protocol stack such as:

1) Improvement in the coexistence with 4G: there was a signal interference in version 4.0 and the long term evolution (LTE) network which caused a degradation in the performance.
2) Introduction of Privacy 1.1 feature, where the private addresses are scrambled to avoid eavesdrop, thus increasing the data security.
3) Dual mode topology, which allows a device to act as a hub or an endpoint. This feature allows the devices to exchange data with each other in a better way. This is, a device can be a central and peripheral device at the same time. A CD can be connected to multiple PD and a PD can be connected to multiple CD. As an example, if a user is using a smartphone, a smartwatch and a pedometer, the smartwatch and the pedometer can exchange data directly without the need to communicate with the smartphone.
4) Smart connectivity, which allows the manufacturers to manage their power consumption based on a defined power plan, thus allowing the devices to remain connected for longer periods of time without impacting the battery life.

5. BLE EVOLUTION AND MARKET CHALLENGES

The next version, 4.2, was released in December 2014 and brings enhancements in security and a 2.5x increase in the data throughput. The list of changes in this version are:

1) Data packet length extension. The previous version of the specification defined the maximum packet size as 27 octets, leading to a maximum throughput of 300 Kbps. The 4.2 version increased the packet data unit (PDU) from 27 to 251 octets, being this the amount of data that can be transmitted during the connection events and increasing the throughput to a theoretical maximum of 750 Kbps.
2) Privacy 1.2. Due to several vulnerabilities found in the protocol, the private addresses are now resolved and generated by the controller without involving the host. To achieve this, two features were added in the link layer: LL Privacy and Extended scanner filter policies.
3) Secure connections. In previous versions, the secure simple pairing model was implemented. In the 4.2 version, the pairing of the devices uses the algorithms approved by the National institute of standards and technology (NIST). Thus, the connection signature resolving key (CSRK) is used for authentication of unencrypted data and identify resolving key (IRK) for device identity. These keys are exchanged between the CD and the PD.
4) Low power IP (IPv6/6LoWPAN). By using the low power wireless personal area network, the BLE devices are able to directly access the Internet rather than have to be connected to a smartphone or other device with an IP address.

The 5.0 version was released in December of 2016 and has some major improvements:

1) 2 Mbps Physical (PHY) layer. The maximum throughput is now 1.4 Mbps, however the 5.0 devices are backwards compatible with previous versions, meaning that these devices should also implement the 1 Mbps PHY layer.
2) Long range. The goal for this new feature is to extend the range by decreasing the data rate. This is achieved by adding forward error correction (FEC) capabilities to detect and fix errors in the transmission. By adding the FEC, the data rate is reduced to 125 Kbps or 500 Kbps depending on the coding scheme but allows to increase the range by 4x as there is a higher signal to noise ratio (SNR) tolerance. However, the tradeoff is the increased average current consumption.
3) Advertising extensions. Beacons have become one of the main use cases for BLE. Thus,

the 5.0 version provides an enhancement to the advertising process. In previous versions, there are three dedicated channels for advertising packets and the 37 remaining channels dedicated for data transmission. Additionally, the advertiser cannot send packets at intervals of less than 20 ms and the advertising packet is limited to 31 bytes. The advertising extension allows the devices to send advertising packets in the data channels with a minimum interval of 7.5 ms and as long as 255 octets. This change is key for beacons as now they will be able to broadcast more information since 31 bytes were not enough.

Finally, the latest version, 5.1, was released in December of 2018. This version includes minor changes from the previous one and mostly dedicated to the advertising capabilities:

4) Angle of arrival (AoA) and angle of departure (AoD). This feature allows higher location accuracy. This means that a BLE 5.1 device can figure out the location of an asset down to centimeters. This is achieved by means of an array of antennas that one of the devices must have. The data received from the antennas allows to identify the direction of the BLE signal and thus determine the exact location. This feature is useful for indoor navigation and identify asset location (such as a worker in a factory).
5) GATT caching. A device performs a service discovery function when a connection is established in order to discover what services the master supports. By performing a cache functionality, the time to connect is reduced and thus less power is consumed.
6) Advertising channel index. This feature allows a device to randomly select an advertising channel. This allows to reduce the probability of interference between devices. This is useful when several BLE devices are advertising, such as in a beacon usage case.

5.2. BLE competition analysis - Other IoT protocols

The IoT market is evolving at a very fast pace, thus fostering the evolution of existing communication protocols or the creation of new ones. In any case, this evolution is triggered by the different challenges that arise with the creation of a new IoT use case. As an example, one of the main issues for the IoT communication protocols is the required low power consumption, thus several initiatives exist to address this problem and compete with BLE. In the next sub-sections, the competing BLE protocols are presented.

5.2.1.1 WiFi

Over the years, WiFi has evolved and the Internet traffic has exploded, meaning that the IPv4 addresses have been exhausted and therefore the adoption of IPv6 is now mandatory to keep up with the current demand. Nowadays, the devices accessing the Internet are not just computers but also smartphones, tablets and IoT smart devices such as wearables, autonomous vehicles and many other use cases that will arise in the next decade. For this reason, the WiFi technology requires enhancements to address the evolution for interconnection, such as reducing the power consumption while keeping the range and throughput. As an example, these are the trends for WiFi:

- Low power wake-up radio (LP-WUR), which is meant to reduce the power consumption by powering off the receiver circuitry when idle and just wakening up when there is a transmission in place [Piyare-17].
- WiFI HaLow, which is a new standard (IEEE 802.11ah), yet to be released, that will allow wireless sensor network (WSN) applications to connect to WiFi with low power consumption and long range transmission (up to 1 Km) [Adame-14]. It is designed to use the sub one GHz license free industrial, scientific and medical (ISM) radio spectrum which will allow IoT applications to interface with other WiFi existing installations and applications [IEEE 802.11ah-sd-16].
- 802.11az – Next generation positioning (NGP), allows to identify the positioning of a WiFi device in a range of 0.5 meters or less. This results in new applications for asset location, public building navigation and others. The estimation is that the approval for this standard will be in the first half of 2021.

Important to realize is that these enhancements in one of the most used communications protocols, will pose a serious competition to BLE across different use cases.

5.2.1.2 Low power wide area networks (LPWAN)

This technology complements the cellular network for IoT applications by achieving long range of operation and battery life of years [Raza-17]. LPWAN is designed to allow the wireless communication of devices in the range of kilometers with a low throughput up to 200 Kbps while maintaining a low power consumption. LPWAN can make use of either licensed or unlicensed frequencies, meaning that the technology can include proprietary or open standard options and their range of applications includes smart cities, I4.0, smart grid, smart lighting, etc. SigFox and LoRaWAN are some examples of this technology.

In this case, LPWAN is more suited for outdoor IoT applications which require to transmit data through long distances and with a very low data throughput very few times a day. In other words, there is a different market opportunity for LPWAN compared with BLE, however there is a competitive space between these technologies that needs to be considered.

5.2.1.3 802.15.4 based protocols: ZigBee and Thread

The IEEE 802.15.4 standard [IEEE 802.15.4-15] is a communication protocol standard that enables the wireless personal area network (WPAN) and supports the ISM spectrum and the sub one GHz band. ZigBee is based on this protocol and was released in 2003 by the ZigBee Alliance.

Similarly to BLE, ZigBee is a low power protocol with low data transfer rates and intended for short transmission ranges. It also provides with a long battery life and provides a very low-cost infrastructure. Additionally, ZigBee allows different network topologies such as star, peer to peer/mesh and cluster tree, thus providing support for a high number of nodes. The 3.0 version is under development and is addressing interoperability issues that exists in previous versions of the protocol.

The Thread protocol is an IPv6 low power mesh networking protocol aimed for device-to-device and device-to-cloud communications [Thread-19]. It supports IPv6 addresses and is optimized for low power consumption with a data throughput of 250 Kbps. The mesh network is connected to a router that uploads the data to the cloud. Additionally, it also provides WiFi connectivity to a smartphone or tablet.

The main applications are related to the smart home, such as smart lighting, sensors, smoke detectors, doorbells, cleaning robots, body sensors, door locks, etc. Its low cost and wide availability from different vendors are a serious competition to BLE.

5.2.1.4 Power consumption of competing protocols

Low power design, along with self-powered devices, is a key differentiator in the industry as it helps to reduce maintenance costs and is essential in the deployment of a new technology. Nowadays, new markets are emerging on regular basis and all of them are requiring low power consumption in order to prevail. As explained in the previous section, different communication protocols are competing to be the IoT protocol of choice, however, it is difficult to predict which one will scale and which one will not. For instance, BLE, ZigBee and Thread have already significant market share as opposed to WiFi HaLow, which is a technology still under development. Moreover, a power consumption comparison between BLE, ZigBee and WiFi makes no sense since WiFi consumes significant more power since the use mode is different. In any case, there exist multiple opportunities depending on these emerging markets. Although different communication protocols are intended to be used for IoT, not all of them will compete directly among each other's but are complementary between them, such as BLE and 5G. For this reason, Table 5-1 shows the power consumption of ZigBee and ANT to provide a reference point of personal area network (PAN) protocols. As can be seen on the table, BLE has the lowest power consumption [Dementyev-13].

TABLE 5.1 CURRENT CONSUMPTION OF BLE, ZIGBEE AND THREAD USING A SLEEP INTERVAL OF 5 AND 10 S

SLEEP INTERVAL (S)	I_{avg} (mA)		
	BLE	ZigBee	ANT
5	0.2	0.45	0.43
10	0.05	0.21	0.25

5.3. Conclusion

BLE is addressing the new market challenges with the evolution of the protocol up to the 5.1 version. This is, BLE has included the mesh networking capabilities, the IPv6 support and has increased its data throughput and the range to compete with the rest of the IoT protocols. Moreover, the great advantage for BLE, is the wide deployment of the protocol in smartphones and the availability of different SoC from different vendors. However, there is the need to create new use cases that can interoperate with other protocols to create better IoT applications. Additionally, several research opportunities exist for BLE, and for IoT protocols in general, such as solve the security challenges, create energy harvesting systems to provide power autonomy for sensors and use machine learning techniques to make sense of the huge amount of data that the IoT applications can collect.

General Conclusions

IoT has evolved significantly in the past years and new business models and applications have hit the market and are generating significant revenue. To this end, several communication protocols are still trying to dominate the market. The IoT devices are not only smartphones, tablets, or computers, but any other device that can be connected and controlled through the internet to exchange data. In other words, objects can be connected, monitored, and controlled to enable smart environments, e.g., smart homes and smart cities, and also to enable devices to be able to transfer data for health, wellness, and fitness applications, among others. In order to achieve this, there are several enabling communication protocols for the IoT, which need to be low power due to the fact that many applications are required to operate using coin cell batteries. Moreover, of the predicted 50 billion connected devices by 2020, around 30% of them are forecasted to use BLE since 100% of the smartphones, tablets and laptops shipped in 2018 include BLE. Therefore, from a business point of view, IoT and BLE represent a huge opportunity for the companies that decide to invest in these technologies.

In Chapter 1, an overview of the BLE protocol was presented. Additionally, the state of the art regarding the study of the power consumption of the protocol was presented. It was noted that in the literature, the research and experiments were always focused in the peripheral device and there was no power consumption data available for a device configured as central device.

Chapter 2 provided an overview and evaluation of different BLE power consumption simulation models. The evaluation of the models was intended to choose the one with the best performance and accuracy to conduct experiments and compare them with physical measurements in order to provide a design guideline for the implementation of BLE applications.

In Chapter 3, an overview of the architecture of a BLE system on chip was presented. Additionally, four commercial BLE development platforms were introduced and a power consumption measurement methodology was shown. The measurements included the power consumption of a BLE central device, data that is not reported elsewhere in the literature nor in the datasheets from the main BLE vendors. Moreover, a comparison of the power consumption of the commercial platforms is presented along with an expected battery life for different connection intervals. This data is key to achieve a dependable design when a central device needs to be powered by batteries that might not be easily replaced.

Chapter 4 presents a low power design guideline for BLE applications. It was shown, in the first place, the implications of a high current pulse in the battery capacity and how that will affect the battery life under short connection intervals. Additionally, a corrected battery capacity, taking into account the high current pulse is presented and a table with expected battery life using short connection intervals is presented. Secondly, the main connection parameters that can be modified to maintain the low power are presented. Important to note is that the BLE specification defines several parameters for the advertising and connection events, making difficult to understand which of them are really relevant for the power consumption of a device.

Chapter 5 presents the evolution and changes of the BLE protocol as well as an overview of other IoT communication protocols that compete with BLE.

It is difficult to keep pace with new IoT business models and address all the technical challenges. For example, smart homes, smart cities, healthcare applications, and Industrial IoT are exploding and more use cases are work in progress or already hitting the market. The BLE opportunities and challenges are:

Machine learning - New techniques are required to process the collected data from the sensors, such as machine learning algorithms, to make sense of the data and provide innovative solutions to specific markets. Several products are using machine learning to differentiate from other competitors and they provide data management to offer personal training, injuries prevention, and comprehensive medical data analysis to prevent diseases. The machine learning algorithms are usually implemented in a cloud server but, if the complexity and compute resources are enough, they might also be implemented in the sensor platform. However, the trade-off of this solution would be the power consumption and thus the requirement of a longer duration of the platform battery. In any case, this area poses significant research opportunities to enable a better ecosystem for IoT BLE applications.

Power consumption and energy harvesting - Power consumption is a big challenge for IoT applications. A critical differentiator from different IoT enabling protocols would be the reduction of power consumption, especially in outdoor applications, such as the smart farming and smart cities, where replacing the batteries of sensors is not an easy task. Thus, equally important are the energy harvesting techniques that could be applied to the BLE devices. Some studies suggest the use of human body motion to harvest energy and other studies propose hybrid methods for energy harvesting. With this in mind, this area requires further research to reduce the power consumption of the devices and discover new methods to provide energy to them.

Mesh networking - As explained in Chapter 5, mesh support by BLE will help the protocol to compete with other solutions (ZigBee, Thread and WiFi). However, one of the main problems of BLE mesh network solution is the power consumption of the nodes in the network, since the devices need to be in receive operation mode constantly, therefore consuming more power that reduce the battery life. In view of the recent introduction of mesh networking BLE specification, several research opportunities exist in this area to propose different techniques.

As a final summary, this doctoral dissertation presents power consumption measurements of different commercial platforms, including the central device configuration, and a design guideline for BLE based IoT applications.

Conclusiones Generales

IoT ha evolucionado significativamente en los últimos años y nuevos modelos de negocios y aplicaciones han llegado al mercado a generar importantes ingresos. Debido a esto, varios protocolos de comunicación todavía están intentando dominar el mercado. Los dispositivos IoT no son solo teléfonos inteligentes, tabletas o computadoras, sino cualquier otro dispositivo que se pueda conectar y controlar a través de Internet para intercambiar datos. En otras palabras, los objetos se pueden conectar, monitorear y controlar para habilitar entornos inteligentes, por ejemplo, hogares y ciudades inteligentes, y también para permitir que los dispositivos puedan transferir datos para aplicaciones de salud, bienestar y estado físico, entre otros. Para lograr esto, hay varios protocolos de comunicación que habilitan IoT, los cuales deben ser de baja potencia debido al hecho de que las aplicaciones y dispositivos requieren operar con baterías de tipo moneda. Además, de los 50 mil millones de dispositivos conectados pronosticados para 2020, se prevé que alrededor del 30% de ellos usen BLE, ya que el 100% de los teléfonos inteligentes, tabletas y computadoras portátiles embarcados en 2018 incluyen BLE. Por lo tanto, desde un punto de vista comercial, IoT y BLE representan una gran oportunidad para las empresas que deciden invertir en estas tecnologías.

En el Capítulo 1, se presentó una descripción general del protocolo BLE. Además, se presentó el estado del arte con respecto al estudio del consumo de energía del protocolo. Se observó que en la literatura, la investigación y los experimentos realizados, siempre se centraron en el dispositivo configurado en modo esclavo y no existían datos disponibles en la literatura del consumo de energía para un dispositivo configurado como dispositivo maestro.

El Capítulo 2 proporcionó una descripción general y la evaluación de diferentes modelos de simulación de consumo de energía para BLE. La evaluación de los modelos tiene la intención de elegir el que tuviera el mejor rendimiento y precisión para realizar experimentos y compararlos con mediciones físicas para proporcionar una guía de diseño para la implementación de aplicaciones BLE.

En el Capítulo 3, se presentó una descripción general de la arquitectura de un sistema BLE. Así mismo, se presentaron cuatro plataformas comerciales de desarrollo de BLE y se mostró una metodología de medición del consumo de energía. Las mediciones incluyeron el consumo de energía de un dispositivo central BLE, datos que no se muestran en la literatura ni en las hojas de datos de los principales proveedores de BLE. Además, se presenta una comparación del consumo de energía de las plataformas comerciales junto con la vida útil esperada de la batería para diferentes intervalos de conexión. Estos datos son clave para lograr un diseño confiable cuando un dispositivo central necesita ser alimentado por baterías que podrían no ser fácilmente reemplazadas.

El Capítulo 4 presenta una guía de diseño para aplicaciones de baja potencia para BLE. Se mostró, en primer lugar, las implicaciones qué un pulso de alta corriente tiene en la capacidad de almacenamiento de la batería y cómo eso afectará la vida útil de la misma cuando se configuran intervalos de conexión cortos para cierta aplicación. Además, se presenta una capacidad de almacenamiento corregida para la batería, teniendo en cuenta el mencionado pulso de alta corriente y se presenta una tabla con la vida útil esperada de la batería utilizando intervalos de conexión cortos. En segundo lugar, se presentan los principales parámetros de conexión que se pueden modificar para mantener la baja potencia del diseño. Es importante tener en cuenta que la especificación de BLE define varios parámetros para la comunicación, lo que dificulta la comprensión de cuáles son realmente relevantes para el consumo de energía de un dispositivo.

El Capítulo 5 presenta la evolución y los cambios del protocolo BLE, así como una descripción general de otros protocolos de comunicación IoT que compiten con BLE.

Es difícil seguir el ritmo de aparición de los nuevos modelos de negocio de IoT y abordar eficientemente todos los desafíos técnicos que surgen día a día. Por ejemplo, las aplicaciones para las casas y ciudades inteligentes, aplicaciones médicas y el IoT industrial están apareciendo de manera exponencial y hay más casos de uso que ya están llegando al mercado. Por lo tanto, las oportunidades y desafíos de BLE son:

Machine learning: se requieren nuevas técnicas para procesar los datos recopilados de los sensores, como los algoritmos de aprendizaje automático, para procesar los datos obtenidos y proporcionar soluciones innovadoras a mercados específicos. Varios productos están utilizando el aprendizaje automático para diferenciarse de otros competidores y proporcionar una gestión de datos para ofrecer prevención de lesiones y un análisis integral de datos médicos para prevenir enfermedades. Los algoritmos de aprendizaje automático generalmente se implementan en un servidor en la nube pero, si la complejidad y los recursos informáticos son suficientes, también pueden implementarse en la plataforma que realiza el proceso de adquisición de datos. Sin embargo, el consumo de energía requerido para llevar a cabo esta tarea es considerablemente mayor y, por lo tanto, se requiere de una mayor duración de la batería de la plataforma. De cualquier manera, esta área de investigación presenta importantes oportunidades para habilitar un mejor ecosistema para aplicaciones IoT basadas en BLE.

Consumo de energía y recolección de energía: el consumo de energía es un gran desafío para las aplicaciones de IoT. Un diferenciador crítico de los diferentes protocolos de habilitación de IoT sería la reducción del consumo de energía, especialmente en aplicaciones al aire libre, como la agricultura y las ciudades inteligentes, donde reemplazar las baterías de los sensores no es una tarea fácil. Por lo tanto, las técnicas de recolección de energía (energy harvesting) que podrían aplicarse a los dispositivos BLE son de gran interés. Algunos estudios sugieren el uso del movimiento del cuerpo humano para recolectar energía y otros estudios proponen métodos híbridos para la recolección de energía. Tomando esto en cuenta, esta área requiere investigación para reducir el consumo de energía de los dispositivos y descubrir nuevos métodos para proporcionarles energía.

Redes de malla: como se explicó en el Capítulo 5, el soporte para redes de malla de BLE ayudará al protocolo a competir con otras soluciones existentes en el mercado (ZigBee, Thread y WiFi). Sin embargo, uno de los principales problemas de la solución de red de malla de BLE, es el consumo de energía de los nodos en la red, ya que los dispositivos deben estar en modo de operación de recepción constantemente, por lo que consumen más energía que reduce la vida útil de la batería. En vista de la reciente introducción de las redes de malla en la especificación de BLE, existen varias oportunidades de investigación en esta área para proponer diferentes técnicas.

Como resumen final, esta tesis doctoral presenta mediciones de consumo de energía de diferentes plataformas comerciales, incluida la configuración del dispositivo como maestro, y una guía de diseño de bajo consumo de potencia para aplicaciones de IoT basadas en BLE.

Appendix

A. LIST OF INTERNAL RESEARCH REPORTS

1) E. Garcia-Espinosa, O. H. Longoria-Gandara, and A. Veloz-Guerrero, “Bluetooth low energy as internet of things enabling technology” Internal Report *PhDEngScITESO-14-14-R*, ITESO, Tlaquepaque, Mexico, Dec. 2014.
2) E. Garcia-Espinosa, O. H. Longoria-Gandara, and A. Veloz-Guerrero, “Bluetooth low energy link layer overview” Internal Report *PhDEngScITESO-14-15-R*, ITESO, Tlaquepaque, Mexico, Dec. 2014.
3) E. Garcia-Espinosa, O. H. Longoria-Gandara, and A. Veloz-Guerrero, “Discrete event simulation overview” Internal Report *PhDEngScITESO-14-17-R*, ITESO, Tlaquepaque, Mexico, Dec. 2014.
4) E. Garcia-Espinosa, O. H. Longoria-Gandara, and A. Veloz-Guerrero, “Hearing aid devices: a survey” Internal Report *PhDEngScITESO-15-13-R*, ITESO, Tlaquepaque, Mexico, Dec. 2015.
5) E. Garcia-Espinosa, O. H. Longoria-Gandara, and A. Veloz-Guerrero, “Application cases for Bluetooth low energy hearing aid devices” Internal Report *PhDEngScITESO-15-16-R*, ITESO, Tlaquepaque, Mexico, Dec. 2015.
6) E. Garcia-Espinosa, O. H. Longoria-Gandara, and A. Veloz-Guerrero, “Bluetooth low energy simulation models evaluation” Internal Report *PhDEngScITESO-16-34-R*, ITESO, Tlaquepaque, Mexico, Dec. 2016.
7) E. Garcia-Espinosa, O. H. Longoria-Gandara, and A. Veloz-Guerrero, “Post-silicon validation and timing anomalies,” Internal Report *PhDEngScITESO-18-08-R*, ITESO, Tlaquepaque, Mexico, Apr. 2018.
8) E. Garcia-Espinosa, O. H. Longoria-Gandara, and A. Veloz-Guerrero, “Post-silicon validation based on synthetic test patterns for early detection of timing anomalies,” Internal Report *PhDEngScITESO-18-16-R*, ITESO, Tlaquepaque, Mexico, May 2018.
9) E. Garcia-Espinosa, O. H. Longoria-Gandara, and A. Veloz-Guerrero, “A survey of bluetooth low energy internet of things applications” Internal Report *PhDEngScITESO-19-02-R*, ITESO, Tlaquepaque, Mexico, Apr. 2019.
10) E. Garcia-Espinosa, O. H. Longoria-Gandara, and A. Veloz-Guerrero, “Bluetooth low energy power consumption evaluation for commercial devices” Internal Report

PhDEngScITESO-19-04-R, ITESO, Tlaquepaque, Mexico, May. 2019.

11) E. Garcia-Espinosa, O. H. Longoria-Gandara, and A. Veloz-Guerrero, “A survey of bluetooth low energy internet of things applications” Internal *Report PhDEngScITESO-19-05-R*, ITESO, Tlaquepaque, Mexico, Jun. 2019.

B. LIST OF PUBLICATIONS

1) E. Garcia-Espinosa, O. Longoria-Gandara, A. Veloz-Guerrero, and G. G. Riva, "Hearing aid devices for smart cities: a survey," in *2015 Fisrt IEEE International Smart Cities Conference (ISC2)*, Guadalajara, Mexico, Oct. 2015, pp. 1-5. (ISBN: 978-1-4673-6551-2; e-ISBN: 978-1-4673-6552-9; INSPEC: 15671680; DOI: 10.1109/ISC2.2015.7366198)
2) E. Garcia-Espinosa, E. Gonzalez-Garcia, O. Longoria-Gandara, and A. Veloz-Guerrero, "Post-Silicon validation based on synthetic test patterns for early detection of timing anomalies," in *IEEE Latin American Test Symp. (LATS 2018)*, Sao Paulo, Brazil, Mar. 2018, pp. 1-5. (ISSN: 2373-0862; ISBN: 978-1-5386-1473-0; e-ISBN: 978-1-5386-1472-3; INSPEC: 17733132; DOI: 10.1109/LATW.2018.8347237)
3) E. Garcia-Espinosa, O. Longoria-Gandara, E. Gonzalez-Garcia, A. Veloz-Guerrero, "A survey of Bluetooth low energy IoT applications for investing in attractive opportunities," in *IEEE Latin-American Conf. on Communications (LATINCOM-2018)*, Guadalajara, Mexico, Nov. 2018, pp. 1-5. (p-ISBN: 978-1-5386-6755-2; e-ISBN: 978-1-5386-6754-5)
4) E. Garcia-Espinosa, O. Longoria-Gandara, I. Pegueros-Lepe, and A. Veloz-Guerrero, "Power consumption analysis of Bluetooth low energy commercial products and their implications for IoT applications," *Electronics*, vol. 7, no. 12, p. 386, pp. 1-12, Dec. 2018. (ISSN: 2079-9292; CODEN: ELECGJ; https://doi.org/10.3390/electronics7120386)

Bibliography

[Aceto-19] G. Aceto, V. Persico, and A. Pescape, "A survey on Information and Communication Technologies for Industry 4.0: state of the art, taxonomies, perspectives, and challenges," IEEE Communications Surveys Tutorials (Early access), pp. 1–1, Aug. 2019.

[Adame-14] T. Adame, A. Bel, B. Bellalta, J. Barcelo, and M. Oliver, "IEEE 802.11AH: the WiFi approach for M2M communications," *IEEE Wireless Communications*, vol. 21, no. 6, pp.144-152, Dec. 2014.

[Arduino-17] Arduino (2017). *Getting started with Arduino/Genuino 101*. [Online]. Available: https://www.arduino.cc/en/Guide/Arduino101

[Benyamina-11] D. Benyamina, A. Hafid, and M. Gendreau, "Wireless mesh networks design - a survey," IEEE Communications Surveys & Tutorials, vol. 14, no. 2, pp. 299-310, May 2011

[Bluetooth-18] Bluetooth technology website. (2018). *GATT specifications*. [Online]. Available: https://www.bluetooth.com/specifications/gatt

[Bluetooth SIG-18] Bluetooth SIG. (2018). *Bluetooth Market update*. [Online]. Available https://www.bluetooth.com/markets/market-report.

[Bluetooth SIG sd-14] Bluetooth SIG. (2014). *Specification of the Bluetooth System: Covered Core Package version: 4.1*, [Online]. Available: https://www.bluetooth.org/Technical/Specifications.

[Bluetooth SIG sd -17 Bluetooth SIG. "Bluetooth mesh networking: paving the way for smart lighting," 2017.

[Burrascano-99] P. Burrascano and M. Mongiardo, "A review of artificial neural networks applications in microwave CAD," *Int. J. RF and Microwave Computer-Aided Eng.*, vol. 9, no. 3, pp. 158-174, May 1999.

[CSR-00] CSR. (2010). *Bluetooth 4.0: Low Energy* [Online]. Available: http://chapters.comsoc.org/vancouver/BTLER3.pdf

[Cypress CY8CKIT-19] Cypress (2019). *CY8CKIT-042-BLE-A Bluetooth Low Energy 4.2 Compliant Pioneer Kit*. [Online]. Available: https://www.cypress.com/documentation/development-kitsboards/cy8ckit-042-ble-bluetooth-low-energy-42-compliant-pioneer-kit

[Cypress PSoC-19] Cypress (2019). *PSoC BLE (Bluetooth Smart)*. [Online]. Available: https://www.cypress.com/products/psoc-4-ble-bluetooth-smart

[Darroudi-17] S.M. Darroudi and C. Gomez, "Bluetooth low energy mesh networks: a survey," Sensors, vol. 17, no. 7, pp. 1-19, Jun. 2017.

[Dementyev-13] A. Dementyev, S. Hodges, S. Taylor, and J.Smith., "Power consumption analysis of Bluetooth Low Energy, ZigBee and ANT sensor nodes in a cyclic sleep scenario," *Proc. IEEE Int. Wireless Symp.*, Beijing, China, 2013, pp. 1-4.

[Energizer-18] Energizer (2018). *CR2032 Product Datasheet*. [Online]. Available: http://data.energizer.com/pdfs/cr2032.pdf

BIBLIOGRAPHY

[Furset-11] K. Furset, Nordic Semiconductor (2011). *High pulse drain impact on CR2032 coin cell battery capacity*. [Online]. Available: https://m.eet.com/media/1121454/c0924post.pdf

[Garcia-Espinosa-18] E. Garcia-Espinosa, O. Longoria-Gandara, I. Pegueros-Lepe, and A. Veloz-Guerrero, "Power Consumption Analysis of Bluetooth Low Energy Commercial Products and Their Implications for IoT Applications," Electronics, vol. 7, no. 12, pp 386-398, Dec. 2018

[goTenna-18] goTenna (2018). *Go Anywhere, Connect Everywhere*. [Online]. Available https://gotenna.com/

[Gomez-11] C. Gomez, I. Demirkol, and J. Paradells, "Modeling the maximum throughput of Bluetooth low energy in an error-prone link," IEEE Communications Letters, vol. no. 11, pp. 1187-1189, Nov. 2011

[Gomez-12] C. Gomez, J. Oller, and J. Paradells, "Overview and evaluation of Bluetooth low energy: An emerging low-power wireless technology," Sensors, vol. 12, no. 9, pp. 11734–11753, Aug. 2012

[Hexoskin-18] Hexoskin (2018). *Wearable Body Metrics*. [Online]. Available https://www.hexoskin.com/

[Heydon-12] Robert Heydon, Bluetooth Low Energy: The Developer's Handbook, Crawfordsville, IN: Prentice Hall, Oct. 2012.

[IAR-19] IAR embedded workbench (2019). [Online]. Available: https://www.iar.com/iar-embedded-workbench/

[IEEE 802.11ah-sd-16] IEEE 802.11ah-2016 – *IEEE Standard for Information technology--Telecommunications and information exchange between systems - Local and metropolitan area networks--Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 2: Sub 1 GHz License Exempt Operation*. [Online]. Available: https://standards.ieee.org/standard/802_11ah-2016.html

[IEEE 802.15.4-15] IEEE 802.15.4-2015 - *IEEE Standard for Low-Rate Wireless Networks*, [Online]. Available: https://standards.ieee.org/standard/802_15_4-2015.html

Ilumi-18] Ilumi (2018). *The World's Smartest Light Bulb*. [Online]. Available https://ilumi.co/

[Intel-17] Intel (2017). *Intel Curie module* [Online]. Available: https://www.intel.com/content/dam/support/us/en/documents/boardsandkits/curie/intel-_curie-module-datasheet.pdf

[Jeon-18] K.E. Jeon, J. She, P. Soonsawad, and P. Chet Ng, "BLE beacons for internet of things applications: survey, challenges, and opportunities," IEEE Internet of Things Journal, vol. 5, no. 2, pp. 811-828, Jan. 2018.

[Kindt-14] P. Kindt, D. Yunge, R. Diemer, and S. Chakraborty, "Precise energy modeling for the Bluetooth low energy protocol," *arXiv:1403.2919 [cs.NI]*, Mar. 2014.

[Kontakt-17] Kontakt.io (2017). *Advertising interval – best practice*. [Online]. Available: https://support.kontakt.io/hc/en-gb/articles/201567802-Advertising-Interval-best-practise

[Khurpade-18] J. M. Khurpade, D. Rao, and P. D. Sanghavi, "A Survey on IOT and 5G Network," *in 2018 International Conference on Smart City and Emerging Technology (ICSCET)*, Jan. 2018, pp. 1–3.

[Li-18] S. Li, L. Da Xu, and S. Zhao, "5G Internet of Things: A survey," Journal of Industrial Information Integration, vol. 10, pp. 1-9, Feb. 2018.

[Lindh-15] J. Lindh, "Bluetooth low energy beacons", Texas Instruments, Jan. 2015, p. 2.

[Liu Chen-12] J. Liu, C. Chen, and Y. Ma., "Modeling Neighbor Discovery in Bluetooth Low Energy Networks," IEEE Communication Letters, vol. 16, no. 9, pp. 1439-1441, Sept 2012.

[Liu-12] T. Liu and D. Lu, "The application and development of IOT," in 2012 International Symposium on Information Technology in Medicine and Education (ITME), Aug. 2012, vol. 2, pp. 991–994.

[Mattern-10] F. Mattern and C. Floerkemeier, "From the internet of computers to the internet of things," From Active Data Management to Event-Based Systems and More, vol. 6462, K. Sachs, I. Petrov, and P. Guerrero, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 242–259.

[Mazhelis-13] O. Mazhelis, H. Warma, S. Leminen, P. Ahokangas, P. Pussinen, M. Rajahonka, R. Siuruainen, H. Okkonen, A. Shveykovskiy and J. Myllykoski, "Internet-of-Things market, value networks, and business models: state of the art report," Computer Science and Information Systems Reports TR-39, University of Jyvaskyla, Finland, 2013.

[Mikhaylov-14] K. Mikhaylov, "Simulation of network-level performance for Bluetooth low energy," *in 2014 IEEE 25th Annual International Symposium on Personal, Indoor, and Mobile Radio Communication (PIMRC)*, Washington, DC, Sep. 2014, pp. 1259-1263.

[Nordic-19] Nordic (2019). *nRF51822 System on Chip*. [Online]. Available: https://www.nordicsemi.com/Products/Low-power-short-range-wireless/nRF51822.

[NXP-17] NXP (2019). *AN12059. FRDM-KW41Z RF System evaluation report for BLE applications.* [Online]. Available: https://www.nxp.com/docs/en/application-note/AN12059.pdf

[NXP 31Z-19] NXP (2019). *KW31Z: Kinetis KW31Z-2.4 GHz Bluetooth Low Energy Wireless Radio Microcontroller (MCU) based on Arm Cortex-M0+ Core*. [Online]. Available: https://www.nxp.com/products/wireless/bluetooth-low-energy/kinetis-kw31z-2.4-ghz-bluetooth-low-energy-wireless-radio-microcontroller-mcu-based-on-arm-cortex-m0-plus-core:KW31Z.

[NXP 41Z-19] NXP (2019). *KW41Z: Kinetis KW41Z-2.4 GHz Dual Mode: Bluetooth Low Energy and 802.15.4 Wireless Radio Microcontroller (MCU) based on Arm Cortex-M0+ Core.* [Online]. Available: https://www.nxp.com/products/wireless/bluetooth-low-energy/kinetis-kw41z-2.4-ghz-dual-mode-bluetooth-low-energy-and-802.15.4-wireless-radio-microcontroller-mcu-based-on-arm-cortex-m0-plus-core:KW41Z.

[NXP 902x-19] NXP (2019). *QN902X: Ultra low power Bluetooth LE system-on-chip solution*. [Online]. Available: https://www.nxp.com/products/wireless/bluetooth-low-energy/ultra-low-power-bluetooth-le-system-on-chip-solution:QN902X.

[NXP 908x-19] NXP (2019). *QN908x: Ultra-Low-Power Bluetooth Low Energy System on Chip Solution*. [Online]. Available: https://www.nxp.com/products/wireless/bluetooth-low-energy/qn908x-ultra-low-power-bluetooth-low-energy-system-on-chip-solution:QN9080.

[NXP FRDM-18] NXP (2018). *FRDM-KW41Z: Freedom development kit for Kinetis KW41Z/31Z/21Z MCUs.* [Online]. Available: https://www.nxp.com/design/development-boards/freedom-development-boards/wireless-connectivy/freedom-development-kit-for-kinetis-kw41z-31z-21z-mcus:FRDM-KW41Z

[NXP FRDM-17]. NXP (2017). *UM11087 IoT toolbox mobile application user manual.* [Online]. Available: https://www.nxp.com/docs/en/user-guide/KBLETMAUG.pdf?fromsite=ja

[Piyare-17] R. Piyare, A.L. Murphy, C. Kiraly, P. Tosato, and D. Brunelli, "Ultra low power wake-up radios: a hardware and networking survey," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2117-2157, Jul. 2017.

[Polar-18] Polar (2018). *Smart Coaching*. [Online]. Available https://www.polar.com/us-en/smart_coaching

[Raptis-19] T. P. Raptis, A. Passarella, and M. Conti, "Data Management in Industry 4.0: State of the Art and Open Challenges," IEEE Access, vol. 7, pp. 97052–97093, Jul. 2019.

[Raza-17] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low power wide area networks: an overview," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 855-873, Jan. 2017.

[Siekkinen-12] M. Siekkinen, M. Hiienkari, J. Nurminen, and J. Nieminen., "How low energy is Bluetooth low energy? Comparative measurements with ZigBee/802.15.4", *in IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, Paris, France, April 2012, pp. 232-237.

[Seneviratne-17] S. Seneviratne, Y. Hu, T. Nguyen, G. Lan, S. Khalifa, K. Thilakarathna, M. Hassan, and A. Seneviratne, "A survey of wearable devices and challenges," IEEE Communications Surveys & Tutorials, vol. 19, no. 4, pp. 2573-2620, Jul. 2017

[Sensoria-18] Sensoria (2018). *Sensoria Artificial Intelligence Sportswear*. [Online]. Available http://www.sensoriafitness.com/

[Swan-12] M. Swan, "Sensor Mania! The internet of things, wearable computing, objective metrics, and the quantified self 2.0," Journal of Sensor and Actuator Networks, vol. 1, no. 3, pp. 217–253, Nov. 2012.

[SysAgria-19] SysWin Solutions (2019). *SysAgria*. [Online]. Available: https://www.syswinsolutions.com/sysagria/

[Ting-17] Y. Ting and F. J. Lin, "A comparison and evaluation of different BLE connection methods for wearable devices," in 2017 IEEE Conference on Standards for Communications and Networking (CSCN), Helsinki, Finland, pp. 240–245, Oct. 2017.

[Texas-13] Texas Instruments (2013). *SWRU270C Bluetooth low energy CC2540/41 mini development kit user´s guide.* [Online]. Available: http://www.ti.com/lit/ug/swru270c/swru270c.pdf

[Texas-14] Kamath and J. Lindh, Texas Instruments (2014). *Measuring Bluetooth low energy consumption*, [Online]. Available: http://www.ti.com/lit/an/swra347a/swra347a.pdf

[Texas 2540-19] Texas Instruments (2019). *CC2540* [Online]. Available: http://www.ti.com/product/CC2540

[Texas 2541-19] Texas Instruments (2019). *CC2541* [Online]. Available: http://www.ti.com/product/CC2541

[Texas 2640-19] Texas Instruments (2019). *CC2640* [Online]. Available: http://www.ti.com/product/CC2640

[Thread-19] Thread (2019). *What is Thread*? [Online]. Available: https://www.threadgroup.org/what-Is-thread

[Treurniet-15] J. Treurniet, C. Sarkar, R.V. Prasad, and W. Boer., "Energy Consumption and Latency in BLE Devices under Mutual Interference: An Experimental Study", *in Proc. IEEE Future Internet of Things and Cloud (FiCloud)*, Rome, Italy, pp. 333-340, Aug. 2015.

[Virdis-19] A. Virdis and M. Kirsche, Eds., Recent Advances in Network Simulation: The OMNeT++ Environment and its Ecosystem. Cham: Springer International Publishing, 2019.

[Woolley-17] M. Woolley and S. Schmidt, "Bluetooth mesh networking: an introduction for developers," in Bluetooth SIG, 2017.

[Yang-11] Z. Yang, Y. Peng, Y. Yue, X. Wang, Y. Yang, and W. Liu, "Study and application on the architecture and key technologies for IOT," in 2011 International Conference on Multimedia Technology (ICMT), Hangzhou, China, Jul. 2011, pp. 747–75.

[Zephyr-19] Zephyr Project (2019). *Arduino/Genuino 101*. [Online]. Available: https://docs.zephyrproject.org/latest/boards/x86/arduino_101/doc/index.html

Author Index

Subject Index