DEGREE PROJECT IN ELECTRICAL ENGINEERING,
SECOND CYCLE, 30 CREDITS
STOCKHOLM, SWEDEN 2016

Battery Lifetime Modelling and Validation of Wireless Building Automation Devices in Thread

EVA AZOIDOU

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING

Abstract

The need for energy efficiency in wireless communication is prevalent in all areas, but to an even greater extent in low-power and lossy networks that rely on resource-constrained devices. This degree project seeks to address the problem of modelling the battery lifetime of a duty-cycled node, participating in a wireless sensor network that is typically used in smart home and building applications. Modelling in MATLAB combined with experimentation are employed to predict the life expectancy and to validate using a hardware implementation. Various scenarios including sleepy end devices in a wireless sensor network are modelled and validated; these range from variable wake-up frequency and packet payload transmission to increasing network contention with the addition of network load. A comprehensive analysis of the main factors contributing to wasteful energy usage is provided in this thesis project, and it can be concluded that the model can estimate the battery lifetime under different testing scenarios with an error less than 5 %.

Keywords: Internet of Things (IoT), battery lifetime, wireless sensor networks, energy efficiency, IEEE 802.15.4 MAC, power consumption model, Thread.

Sammanfattning

Det finns ett stort behov av energieffektivitet inom trådlös kommunikation, särskilt inom nätverk med bortfall och låg strömförbrukning där resursbegränsade enheter nyttjas. Det här examensarbetet eftersträvar att lösa problemet med att modellera batterilivslängden hos en sensoranordning med en låg driftcykel, som en del av ett trådlöst sensornätverk avsett för att tillämpas i smarta hus och byggnader. Modellering i MATLAB kombinerat med experimentering används för att förutsäga den förväntade livslängden samt för att validera en hårdvaruimplementering. Flertalet scenarier med sovande noder modelleras och valideras, med allt från variabel uppvakningsfrekvens och paketöverföring till ökande resurskonflikter med ytterligare belastning på nätverket. I detta examensprojekt inkluderas en heltäckande analys av huvudorsakerna till energislöseriet hos enheterna och slutsatsen kan dras att modellen kan beräkna batterilivslängden för olika testscenarier med mindre än 5 % fel.

Nyckelord: Sakernas internet, batterilivslängd, trådlösa sensornätverk, energieffektivitet, IEEE 802.15.4 MAC, energiförbrukningsmodell, Thread.

Table of Contents

List of Figures

List of Tables

List of Acronyms and Abbreviations

6LoWPAN IPv6 over Low power Wireless Personal Area Networks

AC/DC alternating current/direct current

ACK acknowledgement

ADC analog-to-digital converter

APP application

BE backoff exponent

BR border router

CCA clear channel assessment

CoAP constrained application protocol

CON confirmable

CSMA-CA carrier sense multiple access with collision avoidance

dBm decibel-milliwatt

DCF distributed coordination function

DVR distance vector routing

ED end device

HBA home and building automation

HTTP hypertext transfer protocol

IoT Internet of Things

IP Internet Protocol

IPv6 Internet Protocol version 6

LLN low-power and lossy network

M2M machine-to-machine

MAC media access control

mAh milliampere hour

MCU microcontroller unit

MLE mesh link establishment

MPDU MAC protocol data unit

MTU maximum transmission unit

NB number of backoffs

NIP native IP

NMAE normalised mean absolute error

NON non-confirmable

OTA over-the-air

PAN personal area network

PCM power consumption model

PD parent device

PHY physical

PSDU PHY service data unit

QoS quality-of-service

RF radio frequency

Rx receive

SED sleepy end device

Tx transmit

UDP user datagram protocol

WPAN wireless personal area network

WSN wireless sensor network

Chapter 1

Introduction

The first chapter briefly introduces the wide realm of the Internet of Things (IoT), with focus on the sensing and wireless communication side of it. The problem addressed by the thesis is defined and the goals are identified, while mentioning the aspects that fall within and outside the scope of this thesis.

1.1 Background

As the number of things being connected to the Internet has already surpassed the world's population, soon things will generate more data than people, thus leading to the migration from the "Internet of people" towards IoT [1]. The term IoT was conceived by Kevin Ashton in 1999, and he described the notion as things being able to *"see, hear and smell the world for themselves"* in all its randomness without the need for human intervention [2].

Intelligent buildings, iHomes or Domotics are just simply different ways of referring to the concept of the smart home and building, which is one application area of the multifaceted IoT. The intended vision with the smart commercial and residential establishments is to create a vast dynamic information space where identities and virtual personalities are assigned to physical objects. These things function in unison while being incorporated into the physical world, rendering people oblivious to their existence. The smart objects connect to the Internet and operate in a manner that gives rise to ambient intelligence. The ultimate intention is to enhance the world by transforming it into a pervasive and immersive 'supernetwork' [3–5].

The required infrastructure to support and enable such a vision starts with the deployment of wireless sensor networks (WSNs), which monitor the environment around them and forward any essential information to base stations or deliver commands to actuators. The continuous monitoring and reporting capabilities of sensors are hindered by several factors, for instance, interference, node failures, delays, and lack of infinite power source. This raises the question of whether and to what extent this kind of limitations affect both the design and performance of WSNs.

1.2 Problem Definition

Arguably, a trade-off needs to be attained among some of the design features. The most critical issue of them all is the energy efficiency of the resource-constrained sensors, which is controlled by their power consumption. Sensors are compact nodes that are equipped with tiny batteries having limited capacity and are required to be operable for long periods of time [6]. Hence, low power consumption has a significant impact on the longevity of sensors, and it leads to the extension of the WSN life expectancy.

The efficiency of a WSN design or lack thereof has a broad effect, which extends from the radio level up to the various high-level applications being developed on top of it. This dependence highlights the urgency to comprehend and analyse the power consumption properties of sensors and examine different factors in order to identify the culprits that cause the increased consumption of power resources [7].

Taking into account that the aspect of energy efficiency constitutes a main concern regarding the WSN design and performance, a methodology is proposed to model the power consumption of duty-cycled and battery-powered devices intended for applications in the field of home and building automation (HBA). In particular, the main objectives of conducting this degree project are described below:

- Investigate the power consumption of a communication protocol, named Thread, with specific hardware devices and software configuration;

- Derive a mathematical model for the power consumption of the sleepy end device (low duty-cycled sensor node) in a Thread network and implement the model in MATLAB;

- Validate the model and evaluate the estimated battery lifetime using an experimental test bed;

- Identify the primary sources of battery depletion for sensor nodes in Thread.

The benefit that stems from the successful implementation of the proposed model is evaluating the effect of the protocol parameters on the battery lifetime of the resource-constrained wireless building automation devices in Thread. Furthermore, different levels of network size and traffic load can be assessed to infer the magnitude of their influence on the longevity of the sensor device.

1.3 Research Methodology

Scientific research deals with the systematic quest for knowledge in all pertinent areas. For the purpose of this project, a quantitative approach has been taken to construct a model that can sufficiently describe the behaviour of the phenomenon and then employ experimentation to validate the efficiency of the model in representing the operations that take place. Specifically, there are two methods that were used in order to reach the final objectives. The first involved creating a theoretical model with the aid of MATLAB that would represent the behaviour of a duty-cycled node in Thread as regards its current consumption, while the second method included the experimental validation of the theoretical model by utilizing various test cases and deploying the scenarios in real networks.

It has been established that the conducted degree project is of quantitative type. Hence, the research method, strategy, data collection and analysis techniques are to be germane to the quantitative type according to [8]. In particular, for the mathematical model it is necessary to identify all the steps of operation of a battery-powered sensor device. The aspects that remain the same, such as the duration and current consumption of certain processes, led to their values being acquired using precise multiple measurements and then substituting the mean value in the model. However, there exist specific relationships among several variables that are related to the theory of wireless networks. In order for these variabilities to be taken into consideration by the model, a statistical model accounting for the real-world processes is constructed.

As regards the validation using the experimental test bed, a WSN is deployed and various network scenarios are tested. The average duration and current consumption are obtained using precise measuring instruments under controlled conditions to the extent possible. After all the data for the implementation of the mathematical model, and the experimental validation is collected, the data analysis stage follows, which involves statistics and computational mathematics. Further detailed information is provided in Chapters 3, 4, and 5.

1.4 Contributions

In this report, a thorough power consumption model (PCM) is proposed that is validated using experimental means. The contribution of this degree project is threefold:

- The model can predict with a small forecasting error the battery lifetime of a sleepy end device that performs a certain number of operations per hour;
- Several parameters of the model can be modified and adjusted appropriately to observe their effect on battery lifetime. These parameters can be, for

instance, packet size and polling interval;

- Channel access failures and collisions due to increased network size and higher contention levels are represented in the model and their influence on battery lifetime is substantiated.

1.5 Delimitations

It should be noted that the main objectives of this project is the battery lifetime modelling and its experimental validation. The proposal of novel adaptive algorithms to optimise the network operation with respect to power consumption as well as the implementation of the proposed improvements are not considered as part of this project. There are some recommendations about the optimisation of certain features, such as, sleeping polling intervals, transmission of application layer acknowledgements, time-out counter value for retransmissions, but they remain plain suggestions and no verification is provided as to their efficiency.

1.6 Thesis Structure

The remainder of this report is structured as follows: Chapter 2 is an introduction to the Thread protocol, providing some pertinent terminology and few details about the layers of which Thread is composed, with focus on the mechanisms of the MAC sublayer. Furthermore, the chapter presents the related work within the topic of battery lifetime modelling. Chapter 3 delves into the low-power aspect of Thread, concentrating on, and modelling the most common operations that draw current. In Chapter 4, the experimental setup, and the validation environment employed for the purpose of this degree project are described. A comprehensive evaluation of the results, and their significance within the context of energy efficiency in HBA systems is provided in Chapter 5. The final chapter concludes the report, states the delimitations of this degree project and provides alternative ways with which the battery lifetime could have been investigated as future directions for research.

Chapter 2

Background

This section provides a general introduction to the primary terms and notions that lay the foundations for the understanding of the subsequent chapters.

2.1 Thread - a new IoT technology

A plethora of communication technologies have been developed that facilitate the deployment of IoT. Few examples of such technologies are Zigbee, Bluetooth Low Energy, and Z-Wave. Thread is a relatively new wireless networking protocol with founder members stemming from big companies such as Nest Labs of Google, ARM, NXP, and Samsung, to name a few. With the specific application area of HBA in mind, Thread is considered to be the best-suited technology thanks to a mixture of features that provide a great deal of benefits. To begin with, Thread supports interoperability in the application layer by enabling multiple 3$^{\text{rd}}$ party options to be employed on top of it. Furthermore, low power operation is provided using duty-cycled nodes based on the power efficient IEEE 802.15.4 standard, and the support of mesh topologies allows self-healing in the network with no single point of failure. By bringing the flexible and pervasive Internet Protocol (IP) into the field of constrained embedded devices, Thread enables device-to-device, device-to-cloud, and device-to-application or backend service communication.

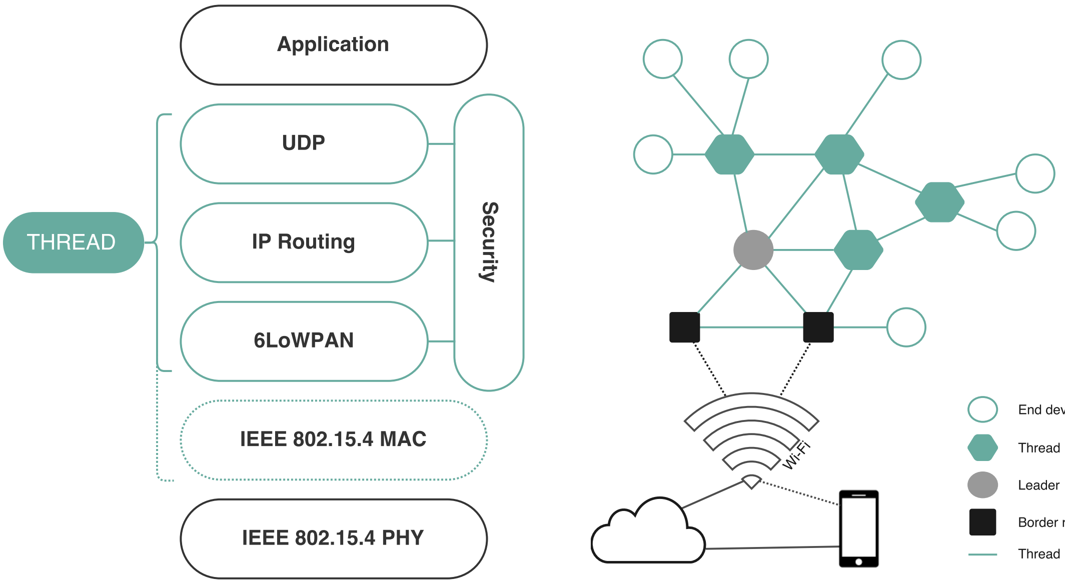
Table 2.1 provides a comparison of various networking protocols with respect to important features necessary for the new phase of IoT [9] [10]. It is apparent that Thread meets the requirements of a low power, IP-based, mesh supporting, interoperable, and open standard that is highly sought after within IoT.

In Fig. 2.1a, the stack that constitutes the Thread communication protocol is illustrated. Thread employs the IEEE 802.15.4 standard for the physical (PHY) layer and the media access control (MAC) sublayer, while providing Internet Protocol version 6 (IPv6) connectivity with the aid of IPv6 over Low power Wireless

Table 2.1: Comparison of Thread with existing protocols

Feature	WiFi	Zigbee Pro	Bluetooth LE 4.x	Z-Wave Plus	Thread
Low power	✘	✔	✔	✔	✔
Mesh network	✘	✔	✘	✔	✔
IPv6 support	✘	✘	✔	✘	✔
Interoperability	✔	✘	✔	✘	✔
Open standard	✔	✔	✔	✘	✔

Personal Area Networks (6LoWPAN) [11]. For the routing in the network layer, Thread uses distance vector routing (DVR), and in the transport layer, the stack supports the user datagram protocol (UDP) in order to enable device messaging. The application (APP) layer is open for developers to make use of as they deem fit, with Thread being able to support several low bandwidth APP layer protocols that run over IPv6. An example of a wireless mesh network with IP connectivity in Thread is shown in Fig. 2.1b.



(a) Thread communication stack

(b) Thread wireless mesh network

Figure 2.1: Thread - a new wireless networking protocol, adapted from [11].

There can be four types of devices in a Thread network: border routers (BRs), routers, router-eligible end devices and sleepy end devices (SEDs). Out of the four, it is only SEDs that are battery-operated while the rest are mains-powered. As a result, the focus of this thesis is on the operations performed by a SED that have impact on its battery lifetime. In the presence of more than one router, the Thread stack supports mesh topologies. The mesh is reduced to a star topology when there is only one router in the network, which provides connection to the end devices.

In the studied implementation of Thread [12], which is used in the devices throughout the degree project, the constrained application protocol (CoAP) is employed in the APP layer. CoAP is a compact web transfer protocol, similar in nature to the hypertext transfer protocol (HTTP), intended for resource-constrained devices participating in low-power and lossy networks (LLNs) [13]. Additional information about CoAP is provided in Section 2.2; it describes the types of messages that are usually exchanged among nodes participating in a Thread network. Moreover, the mesh link establishment (MLE) protocol is used in Thread [12], which is responsible for the establishment and dynamic configuration of secure radio links in IEEE 802.15.4 networks with mesh topologies [14].

A duty-cycled sensor node, which in the context of Thread is a SED, is a host device that can communicate with the rest of the mesh network solely via its parent (active router), which is responsible for holding packets destined for it and for forwarding messages to/from the mesh network [15]. Such a device is in a deep sleep mode for the majority of its operational cycle. Otherwise, an active mode is resumed, which can occur at regular intervals of different duration. During a polling event that is repeated once every χ milliseconds (ms), battery-operated devices wake and check their parent's status about incoming packets. Depending on whether there are any further transactions that need to take place or not, the duration of this polling event is extended or shortened.

In the subsequent sections (2.2 - 2.5), a short overview of the protocols and standards that are employed in each layer of the Thread stack is provided. By using a top-to-bottom approach, CoAP is introduced in the beginning to start with the application layer, and the parts of the IEEE 802.15.4 standard (MAC and PHY layer) pertinent to Thread are described at the end.

2.2 CoAP

The resource-restricted nature of low-power and lossy networks is in need of a protocol that can provide generic web services suited for the demands of devices participating in machine-to-machine (M2M) applications in the field of HBA as well as other related fields [13]. CoAP is a compact protocol that is characterised by an interaction process between entities resembling the one that is encountered in HTTP, namely the client/server model. The communicating parties in a CoAP implementation interact by assuming the role of either a client or a server, and in various occasions these roles are interchanged.

There are certain type of messages that are commonly exchanged in a *request/response* manner between the client-node and the server-node. The *request* messages or methods as defined by CoAP are:

- CoAP GET: with the GET method the client acquires a representation for the information that it is interested in. For instance, it might wish to retrieve the temperature or humidity data from the node that is currently monitoring the corresponding phenomena.
- CoAP POST: the POST method entails that the information included in the request be processed by the server's side. For example, the client requests the recipient to turn their LED switch on/off, and in order for that to happen the message needs to be processed.
- CoAP PUT: the PUT method involves the update or creation of a resource based on the enclosed information in the message. The resource can be an old address in need of modification or a new multicast group address.
- CoAP DELETE: the DELETE method requests the removal of the resource identified by the request.

The *request* messages can be either acknowledged or left unacknowledged; these messages are called confirmable (CON), and non-confirmable (NON) respectively. In the confirmable case, the response of the recipient is usually piggybacked in the acknowledgement.

2.3 IPv6

IoT promotes a vision of information ubiquity that is achieved by the collaboration of many heterogeneous elements and their respective networks. The size of IoT is escalating at a high rate, thus necessitating a sufficiently large addressing space that can sustain the need for unique identification and addressing of a plethora of smart things. To address the vast amount of things that will populate the IoT market, IPv6 is an indispensable and integral part.

It is common knowledge that the IPv4 address space has already been depleted; it has been exhausted since February 2001 [16]. With IPv4 no longer a viable option, the addressing constraints have been alleviated with the introduction of IPv6, which can provide up to 2^{128} addresses. IPv6 includes a subspace of addresses that should be sufficiently large to uniquely identify all the emerging devices of IoT for many years to come. For that reason, IPv6 is considered to be a pivotal enabler of IoT, thus rendering it imperative for IoT devices to have IPv6 functionalities that facilitate smooth and transparent Internet communication. Thus far, incorporating IP connectivity capabilities in WSN devices has been a difficult task for a handful of reasons. However, a general direction towards establishing native support for IP in the field of WSNs has been recently identified as a prosperous step for the future of communication technologies with respect to home and building automation systems [17]. As a result of the transition from non-IP, a new term arises - native IP (NIP) - that entails a novel form of connectivity for the resource-constrained devices with multifold benefits. These benefits range from interoperability facilitation up to expedited access to the assets that the fast-paced evolution of the Internet domain involves.

2.4 6LoWPAN

Sensor devices are proliferating in various IoT implementation scenarios and a great deal of these devices are equipped with IEEE 802.15.4 radios. The communication among them is facilitated due to the use of the common radio frequency (RF) connectivity protocol. However, the growing demands of the HBA field necessitate a universal form of connectivity that can be achieved with the adoption of IPv6, which can provide not only device communication but also cloud access services, and user interaction through mobile applications. This need is covered by the 6LoWPAN layer, which is considered an adaptation layer that lies between the IPv6 network layer and the IEEE 802.15.4 link layer [18, 19].

The transmission of IPv6 packets over IEEE 802.15.4 links, if not properly handled by 6LoWPAN, faces certain challenges. IPv6 requires that a maximum transmission unit (MTU) of 1280 bytes is satisfied as regards packet sizes. On the other hand, IEEE 802.15.4 has set the value of 127 bytes as the maximum size for the PHY service data unit (PSDU), which is the PHY payload carrying the MAC frame to the PHY layer. The MAC payload is further reduced in available size depending on the communication overhead.

The 6LoWPAN adaptation layer implements some mechanisms in order to resolve the issues involved with the interoperability between IPv6 and IEEE 802.15.4. These mechanisms are:

- IPv6 packet fragmentation and reassembly: the large IPv6 packets are fragmented at the transmitter's side and reassembled at the receiver's side;
- IPv6 header compression: compression of IPv6 and UDP headers are also handled by 6LoWPAN to reduce the size of frames being transmitted over-the-air (OTA), which also assists in energy conservation;
- Link layer packet forwarding: for the purpose of forwarding packets in a multihop mesh network, Thread employs the link layer forwarding capability of 6LoWPAN by employing IP layer routing without sending the packets up to the network layer.

2.5 IEEE 802.15.4

As already mentioned, the IEEE 802.15.4 specification is used for the two underlying layers of the Thread stack. Specifically, the 2006 version [20] of it is employed for its channel access and congestion control mechanisms as well as for the fundamental command and data frame management. There are mainly three packet formats that are exchanged for the communication among the children (end devices) and the parent (border or active router): a MAC command frame for polling the parent, a data frame for transmitting messages along with acknowledgement (ACK) frames for confirmation of successful receptions.

2.5.1 Carrier sense multiple access with collision avoidance

In Thread, all nodes employ the unslotted, nonbeacon-enabled and contention-based medium access method of carrier sense multiple access with collision avoidance (CSMA-CA). The CSMA-CA algorithm is used prior to the transmission of MAC command or data frames with the exception of ACK frames.

Fig. 2.2 is a flowchart representing the procedure for channel access and frame (re)transmission that is encountered in CSMA-CA. In preparation for a frame transmission, the source node is required to follow the CSMA-CA mechanism. The process involves initializing certain parameters: the number of backoffs (NB) in CSMA-CA, the value of the backoff exponent (BE), which is set equal to *macMinBE*, and the number of retransmissions (R). The algorithm proceeds by selecting a random number of unit backoff periods in the range (0, 2^{BE}-1] to delay the channel sensing using clear channel assessment (CCA). If the channel is sensed idle, it continues with the data transmission. Otherwise, NB increments and the range of random backoff period increases as well because the new BE value is determined by *min(BE+1, macMaxBE)*. The channel access attempts are repeated every time the channel is sensed busy until NB exceeds the maximum number of CSMA backoffs, which ultimately leads to channel access failure.

The secondary part of the flowchart involves the data transmission stage. After transmitting a frame, the sender waits up to a maximum (*macAckWaitDuration*) amount of time (the attribute is defined by IEEE 802.15.4) in order to receive the link layer ACK. The lack of ACK arrival within the specified time signifies the occurrence of frame collision, which triggers link layer retransmissions. With every retransmission, NB and BE are initialized and follow the CSMA-CA algorithm until successful packet transmission. If the same frame is retransmitted for more than *macMaxFrameRetries*, a data transfer failure is declared and the frame is discarded.

The most pertinent MAC attributes for the understanding of the CSMA-CA algorithm are listed below and they have the following default values, which are

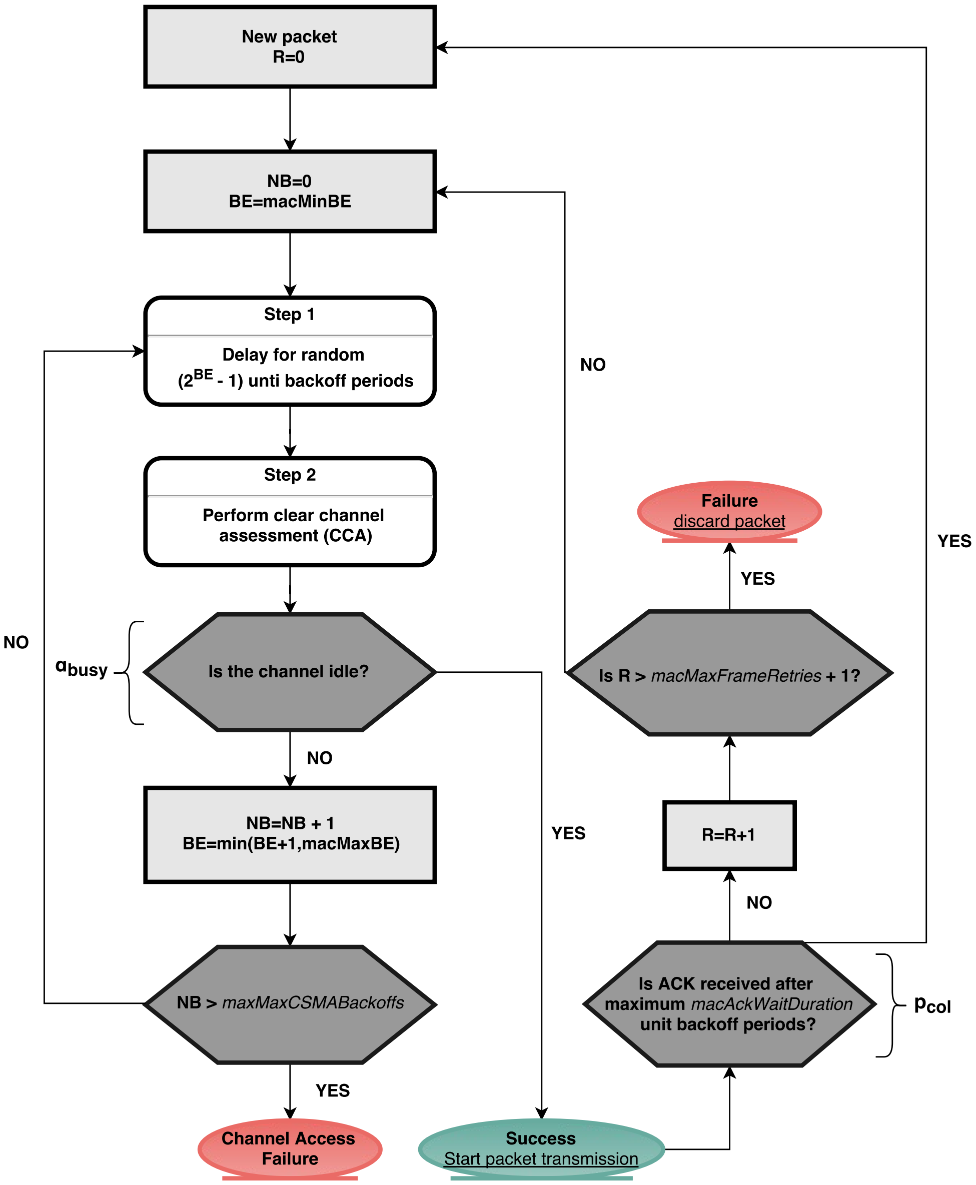


Figure 2.2: Flowchart for the channel access in the unslotted case of CSMA-CA considering packet transmissions and retransmissions, adapted from [20].

obtained from the 2006 version of [20]:

- minimum backoff exponent: $macMinBE = 3$
- maximum backoff exponent: $macMaxBE = 5$
- maximum CSMA backoffs: $macMaxCSMABackoffs = 4$
- maximum MAC retransmissions: $macMaxFrameRetries = 3$

Let β be the number of backoff periods that a node has to wait before performing a CCA. The value of β is equal to $2^{BE}-1$, where BE ranges from one to $macMaxBE$.

A unit backoff period duration in symbols is 20 and the following equation converts the backoff delay from symbols to milliseconds (ms).

$$t_{\text{backoff}} = \frac{20 \cdot \beta}{62.5 \cdot 10^3} = \frac{20 \cdot \beta}{62.5} \text{ ms} \tag{2.1}$$

where 62.5 is the symbol rate measured in ksymbol/s associated with the 2450 MHz PHY in IEEE 802.15.4 and β is the amount of 20-symbol unit backoff periods. For instance, BE values equal to 1 and 5 result in t_{backoff} equal to 0.32 ms and 9.92 ms respectively.

It has been studied in [21] that the selected MAC attributes can have a big impact on energy efficiency, latency, and reliability. According to the paper, a larger minimum backoff exponent (default macMinBE = 3) leads to fewer collisions in the initial backoff stages as well as it reduces the per-packet energy consumption, while an increased amount of backoffs improves general reliability.

2.5.2 Busy channel and collision probability

It can be seen in Fig. 2.2 that there are two probabilities involved in the channel access and data transmission process; these are the busy channel and collision probability that are denoted by α_{busy} and p_{col} respectively. Each time a network node wishes to access the channel, there is an inherent uncertainty regarding its success. This is expressed by the busy channel probability, which essentially controls the incidence rate of CSMA backoffs. As regards retransmissions, they are caused either because the transmitted frame collided and it never arrived at the destination, or the ACK frame was not received by the source. The related probability is p_{col}, which regulates the occurrence of collisions and the need for retransmissions.

2.5.3 Duty cycling mechanism

Periodic wake-ups have been used in various wireless protocols to achieve energy efficiency because it has been observed that idle listening to incoming packets in a wireless medium leads to energy waste [22]. In order to alleviate this issue, numerous duty cycling MAC protocols have been proposed, which can be separated into a number of categories and sub-categories. There exist *synchronous* and *asynchronous* duty cycling mechanisms depending on the need for synchronization among the neighbouring sleepy devices or lack thereof. Furthermore, these techniques can be divided into *contention-based* or *contention-free MAC*, which is determined by the method for accessing and utilizing the medium resources. The last classification is according to the side that instigates the communication, thus leading to transmitter-initiated or receiver-initiated MAC protocols. A detailed discussion of these categories is presented in [22].

The sleepy end devices in Thread employ an *asynchronous*, *contention-based* duty cycling mechanism that is *receiver-initiated.* The SEDs in a Thread network wake up periodically using a fixed wake-up period and they are not required to synchronize their schedules with one another to avoid collisions. Instead, in order for them to access the medium and transmit the required MAC command frame to the parent, they contend with their neighbouring devices using the CSMA-CA mechanism. Furthermore, the communication is always commenced by the side of the duty-cycled node. In any subsequent data transaction with the parent device (PD), the SED acts as a receiver awaiting for data to be transmitted by the parent. More details about these transactions between SED - PD are provided in the following chapter.

2.6 Related Work

In this section, previous work in generic and specific battery lifetime models is reviewed, and literature regarding the modelling of CSMA-CA using Markov chains is briefly discussed.

2.6.1 Battery lifetime models

As the fundamental topic of this thesis is to perform a battery lifetime analysis on a sensor node by identifying and modelling the operations that take place during a communication session, and to quantify their impact on energy consumption, there are several research papers that tackled with similar issues before this current attempt. A few generic energy models are proposed in [23] and [24] for diverse cyber-physical systems. Wang et al. [23] identify the wireless communication module as the primary energy squanderer and therefore employ an alternative sampling technique, called compressed sensing, which disregards the notable Nyquist sampling theorem, to provide a considerably lower power consumption. Their novel proposal is intended for applications within wireless body sensor networks and the energy model is the result of the multiplication of three parameters: the sampling rate, bit resolution, and energy per bit of the communicated data - hence it mainly evaluates the units of a sensor node concerning the aspects of data acquisition and communication. For the purpose of a smart parking application, the work in [24] presents an energy consumption model in which the estimated lifetime derived from the device is intended to be predetermined and fixed. Ultimately, the goal is not to maximize the longevity of the battery-powered sensors but to prioritize the sensing accuracy. The basic energy expenditures that are considered stem from the collection of the samples and the single-hop communication. The proposed energy dimensioning in neither paper is adopted to a specific wireless technology, which leads to lack of detailed communication parameter estimations.

In [25], energy profiling schemes introduced by others are assessed and their effectiveness put to the test by deploying off-the-shelf sensors, and by acquiring both short and long-term measurements. A case study involving a low-data rate WSN application is described, in which the long-term energy monitoring resulted in identifying the severe impact of sink node failures on the lifetime of battery-powered nodes that led to extreme battery depletion in cases of sink node outages.

In [26], Ren et al. put forward an energy consumption and total network lifetime model in which they consider a multi-hop WSN environment where sensors that rely on batteries for their operation, collect measurements and use data periods to transmit the information to a sink employing a collision-free MAC protocol. The total network lifetime is estimated with a small error rate, and they have additionally considered the emergence and effect of energy holes in WSNs. The efficiency of the model is verified using simulations in OMNET++, but no experimental validation using real hardware and real-life network conditions is provided.

Martinez et al. in [27] and [28] view the issue of energy dimensioning for IoT devices from a system-wide standpoint, covering all aspects from data acquisition, and processing to data communication, and system management. Their model covers point-to-point communications for sensor nodes with a fixed or variable data update periods. It is based on a simplified version of MAC (ideal channel conditions, no interference, no congestion) with the addition of opportunistic retransmissions. The model estimation of various factors affecting energy efficiency is obtained through simulations and validation is provided using real hardware.

2.6.2 Markov chain modelling of CSMA-CA

There is a great deal of papers with topics related to analytical models of the CSMA-CA scheme as a channel access mechanism for nodes in wireless networks. On account of that both IEEE 802.11 and IEEE 802.15.4 employ a similar MAC technique for carrier sensing and frame transmission, the modelling of the network traffic in either case can be applied to the other by using few adjustments. In particular, [29] presents an analytical model of the distributed coordination function (DCF) scheme in IEEE 802.11 under the assumption of saturated traffic and ideal channel conditions. Bianchi is the original contributor of the Markov chain model for the CSMA-CA scheme in IEEE 802.11, which led to numerous extensions of this work with respect to various quality-of-service (QoS) metrics for networks that employ either the IEEE 802.11 or the 802.15.4 CSMA-CA algorithm.

As energy efficiency is of the essence in this degree project, the following work studied the CSMA-CA considering also the power consumption of 802.15.4 nodes in the network. The authors in [30] provide a three-dimensional Markov model of the slotted CSMA-CA reviewing only saturated traffic conditions. They propose modifications in order to achieve optimal design. For instance, it is argued that the double carrier sensing is redundant because a single CCA can have the same effect on collision avoidance while the energy consumption is decreased.

Sahoo and Sheu in [31] study the same topic as [30] but for unsaturated traffic conditions, as is the norm for the low data rate wireless personal area networks (WPANs) that operate customarily under light or medium network load conditions. However, the analytical results acquired from the three-dimensional discrete time Markov chain model and the ones obtained from simulation exhibit a poor match quality.

An aperiodic traffic pattern in a single-hop star network is studied in [32] from a power consumption perspective. The study is based on the IEEE 802.15.4 MAC and it examines the effect of the MAC parameters, such as CSMA backoffs, and retransmissions, on energy consumption. The adaptive parameter setting of IEEE 802.15.4 is implemented and validated on a hardware platform. The authors claim that the power consumption of a sensor is not substantially influenced by the MAC retransmissions and the amount of backoffs needed to access the channel.

In this work, the processes that occur in the IEEE 802.15.4 MAC are examined with respect to the operation of a duty-cycled end device in Thread, which employs child polling and exchanges request/response messages with its parent device using CoAP in the APP layer. The ultimate objective is to provide a viable energy dimensioning of battery-operated devices in Thread.

Chapter 3

Battery Lifetime Model

In this chapter, an analytical model for the battery lifetime of a duty-cycled node in Thread is presented, followed by the introduction of the implementation process in MATLAB and the experimental validation.

3.1 CSMA-CA in IEEE 802.15.4

If the small network consisting of the active router - being the parent - and all the connected end devices - acting as children - is isolated from the rest of the Thread mesh network, it is evident that the created sub-network forms a star topology. In such a topology, all nodes have a 1-hop distance from the PD that assumes the role of a central point providing the sole means for connectivity to the rest of the mesh network. This observation proves helpful to model the busy channel and collision instances that might transpire in a star network.

It can be argued that the mesh network topology of Thread can be approximated with a star topology under the assumption that the end devices are detached from the mesh network, if no link to it is provided by the parent device. For that reason, the work introduced in [32] and the derived equations prove helpful in supplying the groundwork for the battery lifetime modelling as regards the mechanism the nodes employ to access the channel and contend with one another. In [32], a generalized Markov chain is proposed that serves as a model for the MAC traffic in a single-hop IEEE 802.15.4 network consisting of N devices that communicate with a personal area network (PAN) coordinator. The N devices contend to transmit data to the PAN coordinator that serves as a data sink in a star network and it is also assumed that no hidden terminals exist.

The nodes generate packets in an asynchronous manner with a probability 1-ρ after a successful packet transmission, or data transfer failure due to packet collision, or channel access failure. If there is no packet generation, the nodes stay in idle state for a duration of L_0S_b seconds with probability ρ, where S_b is the unit backoff period and it is equal to 20 symbols while L_0 is an integer multiple of the unit backoff period. The frame transmission is considered to be completed if a MAC ACK is received, otherwise it is assumed that a collision has occurred. The duration of the successful frame transmission and the one of the collided frame have the following forms respectively:

$$L_s = L + t_{\text{ack}} + L_{\text{ack}} + \text{LIFS} \tag{3.1}$$

$$L_c = L + t_{\text{macAckWaitDuration}} \tag{3.2}$$

where L is the length of the data packet, t_{ack} is the acknowledgement waiting time between *aTurnaroundTime* $\leq t_{\text{ack}} \leq$ *aTurnaroundTime+aUnitBackoffPeriod* [20], L_{ack} is the length of the ACK frame, LIFS is the long interframe spacing (minimum value of 40 symbols), and $t_{\text{macAckWaitDuration}}$ is the ACK timeout. L_s and L_c are defined as multiples of S_b. The long interframe spacing is used instead of the short one because the frames being exchanged have a length above 18 bytes, which is the limit of SIFS according to the standard in [20]. The notations L_0, L_s, and L_c are of importance for the understanding of the equations that are presented subsequently.

The slotted version of CSMA-CA with retransmissions and ACKs is considered in [32], resulting in a three-dimensional Markov chain. A modified and simplified version of the chain is presented in Fig. 3.1 for the unslotted case of CSMA-CA with single carrier sensing instead of double CCAs. The Markov chain is characterized by three stochastic processes s(t), c(t) and r(t), which represent the CSMA backoff stage (i), the random backoff period selection (k), and the retransmission counter (j). In Fig. 3.1, the triad (i,k,j) describes the state of a node when attempting channel access and frame transmission. The following notations are used for the most pertinent MAC attributes, which are also mentioned in subsection 2.5.1:

- minimum backoff exponent: m_0 = *macMinBE*
- maximum backoff exponent: m_b = *macMaxBE*
- maximum CSMA backoffs: m = *macMaxCSMABackoffs*
- maximum MAC retransmissions: n = *macMaxFrameRetries*

The states (i,W_0-1,j) and (i,W_m-1,j) correspond to the maximum backoff period per backoff stage i, where W_0 and W_m are equal to 2^{m_0} and $2^{\min(m_0+m,m_b)}$. For instance, if the state (i,W_0-1,j) is selected by the contending node, its backoff counter is going to decrement until it reaches the state (i,0,j), at which point a CCA is performed to sense the channel idle/busy. The same procedure is followed by the nodes that sense the channel busy and have to backoff again; the value of the backoff stage i is the one that increments until it reaches m. As regards retransmissions, which are triggered when the channel is sensed idle ($1-a_{\text{busy}}$) but a collision occurs (p_{col}), the CSMA-CA algorithm is repeated for each retransmission until the $\text{n}^{\text{th}} + 1$ time.

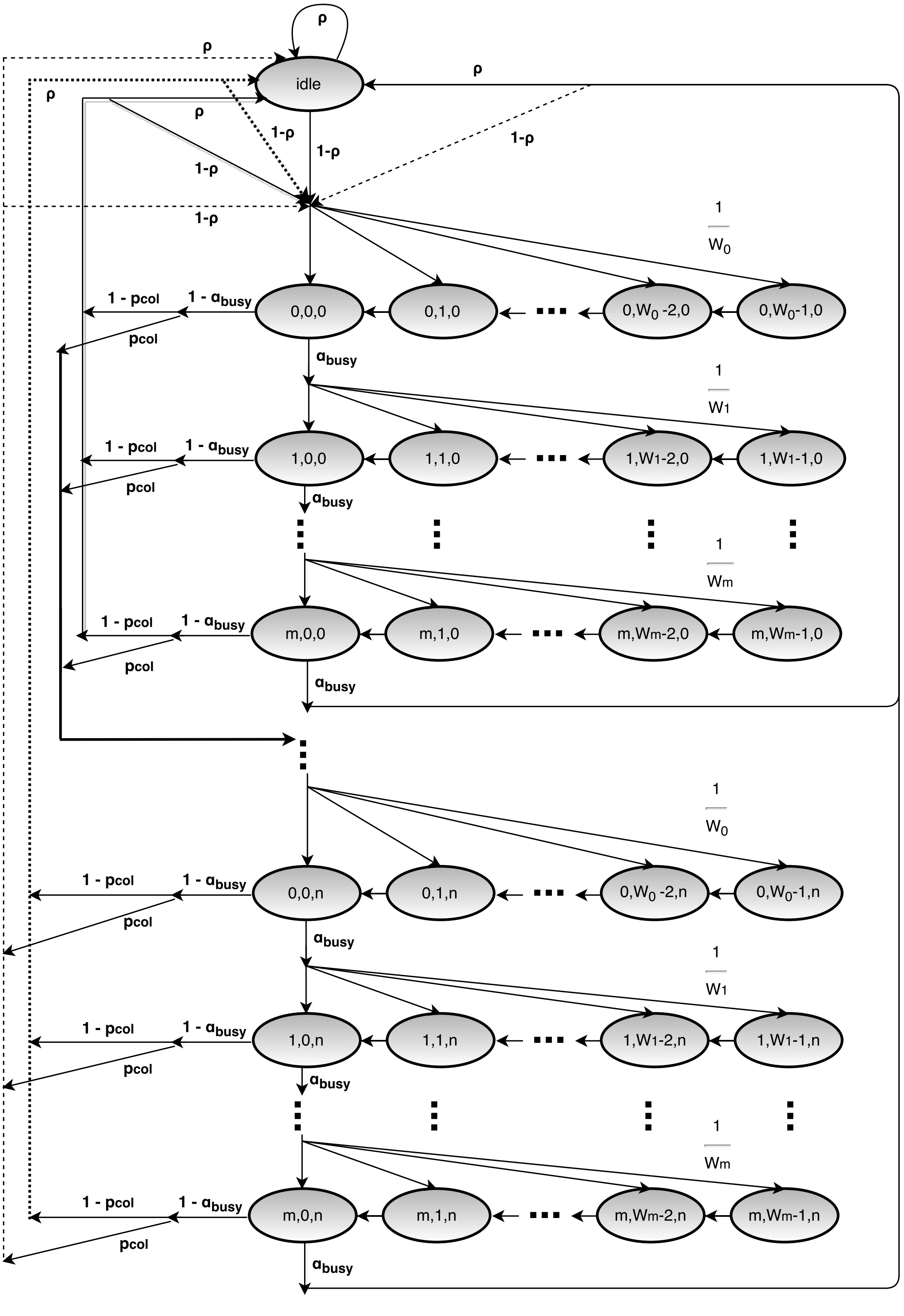


Figure 3.1: Three-dimensional Markov chain model of the CSMA-CA algorithm in IEEE 802.15.4, adapted from [32].

For $0 \leq i \leq m$, $0 \leq k \leq W_i - 1$, and $0 \leq n \leq n$, the stationary probability with the expression $b_{i,k,j} = \lim_{t\to\infty} P(s(t) = i, c(t) = k, r(t) = j)$ is reduced to Eq. 3.3 by taking into consideration the state transition probabilities that are associated with the Markov chain. A comprehensive list of the equations linked to the state transition probabilities is given in [33] for the slotted CSMA-CA. By replacing β, which is the second carrier sensing, with zero and adjusting accordingly all necessary equations, it is possible to acquire the busy channel and collision probabilities for the unslotted CSMA-CA. Detailed proofs of the equations that follow are provided in [33].

$$b_{i,k,j} = \frac{W_i - k}{W_i} b_{i,0,j} \tag{3.3}$$

$$W_i = \begin{cases} 2^{m_0+i}, & 0 \leq i \leq m_b - m_0 \\ 2^{m_b}, & \text{otherwise} \end{cases}$$

$$b_{i,0,j} = \left[(1 - \alpha_{\text{busy}}) \cdot p_{\text{col}} \sum_{i=0}^{m} \alpha_{\text{busy}}^{i}\right]^{j} \alpha_{\text{busy}}^{i} \cdot b_{0,0,0}$$

With the aid of the normalization condition $\sum_i \sum_k \sum_j b_{i,k,j} = 1$ and the derivation of the constituent terms in the expression, $b_{0,0,0}$ is obtained from Eq. 3.4.

$$b_{0,0,0} = (y_1 + y_2)^{-1} \tag{3.4}$$

$$y_1 = \left[\frac{1 - (2\alpha_{\text{busy}})^{m+1}}{2(1 - 2\alpha_{\text{busy}})} \cdot W_0 + \frac{1 - \alpha_{\text{busy}}^{m+1}}{2(1 - \alpha_{\text{busy}})} + (1 - \alpha_{\text{busy}}^{m+1})(1 + L_s(1 - p_{\text{col}}) + L_c \cdot p_{\text{col}})\right] \cdot y_0$$

$$y_2 = \frac{L_0 \cdot \rho}{1 - \rho} \cdot \left[(1 - y) \cdot y_0 + y^{n+1}\right]$$

$$y_0 = \frac{1 - p_{\text{col}}^{n+1}(1 - \alpha_{\text{busy}}^{m+1})^{n+1}}{1 - p_{\text{col}}(1 - \alpha_{\text{busy}}^{m+1})}$$

$$y = p_{\text{col}}(1 - \alpha_{\text{busy}}^{m+1})$$

The probability that a node attempts CCA at a random time unit is given by Eq. 3.5, and it is assumed that the sensing probability is independent of other nodes.

$$\tau = \sum_{i=0}^{m} \sum_{j=0}^{n} b_{i,0,j} = \left(\frac{1 - \alpha_{\text{busy}}^{m+1}}{1 - \alpha_{\text{busy}}}\right)\left(\frac{1 - y^{n+1}}{1 - y}\right) b_{0,0,0} \tag{3.5}$$

It can be observed that τ depends on the collision probability p_{col} and busy channel probability α_{busy}. These are essential aspects for the battery lifetime model because they determine the likelihood of retransmissions and CSMA backoffs in a similar network setup such as the one modelled by the Markov chain in Fig. 3.1. The collision probability can take two forms depending on whether packets are lost solely because of collisions with other transmitted packets (see Eq. 3.6) or bad channel conditions can also contribute to the need for retransmissions (see Eq. 3.7). The bad channel probability p is added as a new term to the collision probability plus the probability of a transmitted packet to collide in the good channel (*1-p*) with a packet transmitted by the N-1 remaining devices in the network. Eq. 3.7 defines the alternative version for the collision probability that is denoted by the subscript *alt*. In spite of introducing the alternative version, it is not used in the battery lifetime model that is proposed in this degree project.

$$p_{\text{col}} = 1 - (1 - \tau)^{N-1} \tag{3.6}$$

$$p_{\text{alt}} = (1 - (1 - \tau(1 - p))^{N-1})(1 - p) + p \tag{3.7}$$

The term α_{busy} is composed of the probabilities for busy channel during CCA due to data transmission (α_1), and due to ACK transmission (α_2). On account of two versions of the collision probability, there are two different forms for the busy channel probability as well.

$$\alpha_{\text{busy}} = \alpha_1 + \alpha_2 \quad \text{or} \quad \alpha_{\text{busy}} = \alpha_{1\text{alt}} + \alpha_{2\text{alt}} \tag{3.8}$$

$$\alpha_1 = L(1 - (1 - \tau)^{N-1})(1 - \alpha_{\text{busy}}) \tag{3.9}$$

$$\alpha_2 = L_{ack} \cdot \frac{N \cdot \tau(1 - \tau)^{N-1}}{1 - (1 - \tau)^N} \cdot \frac{\alpha_1}{L} \tag{3.10}$$

$$\alpha_{1\text{alt}} = L(1 - (1 - \tau(1 - p))^{N-1})(1 - \alpha_{\text{busy}}) \tag{3.11}$$

$$\alpha_{2\text{alt}} = L_{ack} \cdot \frac{N \cdot \tau(1 - p)(1 - \tau(1 - p))^{N-1}}{1 - (1 - \tau)^N} \cdot \frac{\alpha_1}{L} \tag{3.12}$$

This part brings to completion the description of the necessary parameters to account for the considerable variability in time that stems from the random backoff delay, and the occurrence rate of CSMA backoffs as well as retransmissions in an IEEE 802.15.4 network. For further details about the terms and the derivation of the equations, please refer to [32–34].

3.2 SED operations in Thread

In this section, the application of CSMA-CA in Thread is presented by describing all the aspects of SED operations in which the algorithm appears. Furthermore, by observing the current consumption waveforms, it is possible to identify the way the CSMA-CA scheme affects the duration and current consumption of these operations.

Let us recap the fundamental functionality of the sleepy device in Thread. Due to the fact that SEDs run on batteries, they remain in dormant state for the majority of their deployment cycle. In a periodic manner, the SEDs wake up from their dormant state and perform certain operations; such an incident is called a polling event. This event is repeated once every T_{poll} milliseconds and its duration varies determined by the need for further transactions between the communicating parties, namely the SED and PD, or lack thereof.

Every time a sleepy end device wakes up, it transmits a Data Request MAC command frame to the parent awaiting a link layer acknowledgement. If the frame pending bit in the ACK is set to true, then there is data to be followed. Otherwise, this indicates the lack of upcoming frames. With the distinction of the frame pending bit, the main SED events that take place are categorised into data polling and no-data polling events respectively.

Table 3.1 illustrates the packets that are exchanged between child and parent device with detailed information about the type of packets and their sizes along with source and destination addresses; this information is acquired with the aid of a packet sniffer and its captured packets. The short MAC address of the PD is 0x0000 and the address of the SED is 0x0001. The previously mentioned Data Request MAC command frame and its corresponding ACK (ID: Aa) have the fixed lengths of 22 and 5 bytes respectively. These exchanged frames appear in every wake-up cycle no matter the type of the polling event. The frame sequence with the ID *Aa* can be followed by a confirmable or non-confirmable CoAP command (see Section 2.2). For the sake of demonstrating a typical case, Table 3.1 outlines a complete transaction between SED and PD - the so-called data polling event. The CoAP GET CON and the CoAP ACK packets are always followed by link layer acknowledgements (IDs Bb and Dd), and they can have variable sizes. The application layer packet lengths used in this example, that is ID B with 73 bytes and ID D with 68 bytes, correspond to a payload of approximately 10 bytes.

Table 3.1: Exchanged packets between child and parent device

ID	STACK	LAYER	LENGTH	PACKET	SRC	DST
A	THREAD	MAC	22 bytes	DATA REQUEST	0x0001	0x0000
a	THREAD	MAC	5 bytes	LINK LAYER ACK		
B	THREAD	CoAP	73 bytes	GET CON	0x0000	0x0001
b	THREAD	MAC	5 bytes	LINK LAYER ACK		
C	THREAD	MAC	22 bytes	DATA REQUEST	0x0001	0x0000
c	THREAD	MAC	5 bytes	LINK LAYER ACK		
D	THREAD	CoAP	68 bytes	APP LAYER ACK	0x0001	0x0000
d	THREAD	MAC	5 bytes	LINK LAYER ACK		

In Fig. 3.2, the Markov chain represents the governing (sequence of) events that transpire in a Thread network with respect to usual SED operations. To be more specific, the chain is regarded as a semi-Markov process because the inter-arrival and service times do not follow an exponential distribution. The packet IDs and the packet exchange order, which are shown in Table 3.1, are also indicated in the generalized Markov chain to facilitate the understanding of the content of each state.

The no-data polling event, which constitutes the vertical part of the chain with ID *Aa* and arrival rate /service time λ_0/μ_0, is straightforward and the same pattern is repeated with respect to length and sequence of MAC frames, if we disregard the randomness of CSMA-CA. The data polling events, however, can take many forms. For instance, if the CoAP packet that is received in *Bb* is not confirmable and there is no need for a response, then the event will stop at *Cc*. Another example is the case when fragmentation is needed either for the reception or transmission of a packet. In such a case, *Bb* with different packet lengths and *Cc* are repeated until the whole message from the parent device is received, or *Dd* with different packet lengths is repeated until the full response of the SED is transmitted. A representative case of a data polling event is depicted in the horizontal part of the Markov chain with IDs *Aa* - *Dd*, and it takes into consideration all essential features for the derivation of the battery lifetime model.

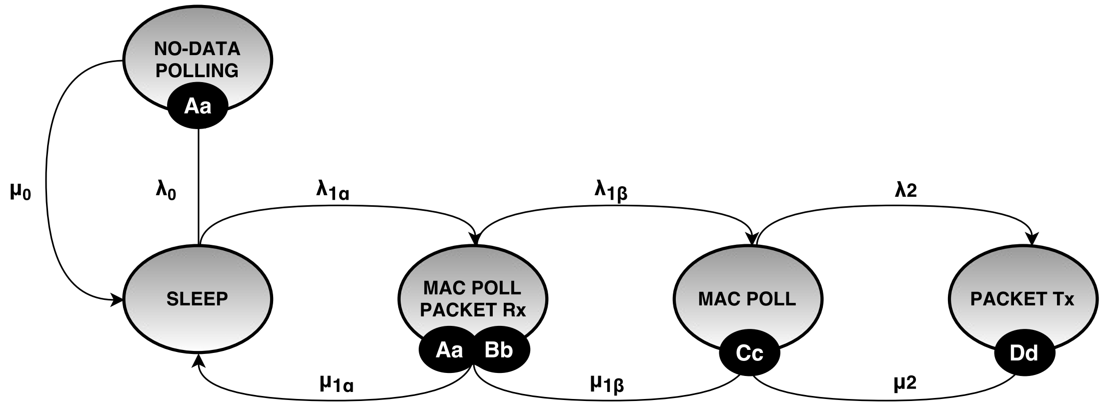


Figure 3.2: Markov chain for the main events taking place in a Thread network as regards SED operations.

Regarding the arrival rates and service times, it is evident that the value of λ_i at each stage differs and μ_i is also variable depending on the size of the packets being transmitted and received. For reasons of associating each λ and μ to the respective state, different notations are used. However, certain arrival rates in states remain the same. For example, λ_0 and $\lambda_{1\alpha}$ are the same because no matter what event is about to follow, when the SED wakes-up, it always transmits the Data Request awaiting for a link layer ACK. The service times, on the other hand differ, due to larger packet lengths in the data polling case. The arrival rate of the second MAC Poll (ID Cc) has a higher value than $\lambda_{1\alpha}$. The reason for this is because once the SED has received the CoAP packet, on all occasions, it proceeds by double-checking whether there is more data to receive from the PD. The service times μ_0 and $\mu_{1\beta}$ are alike due to identical frame lengths. Last but not least, the arrival rate λ_2 and service time μ_2 can vary depending on the need for application layer acknowledgement or not, and the packet size of the SED response. The following text sheds more light on the transitions from the sleeping state into the no-data or data polling states, and what they entail.

Next, the attention is turned to the aspect of the current consumption during the wake-sleep cycle of a SED. The stages through which the SED goes, starting from the instance it wakes up from deep sleep until it returns to the same state, after having completed some sort of operation, are depicted in Fig. 3.4. In order to be able to draw a comparison between data and no-data polling events, the current waveform in Fig. 3.3 refers to the latter case. The stages of the data polling event are listed and briefly explained in the following text.

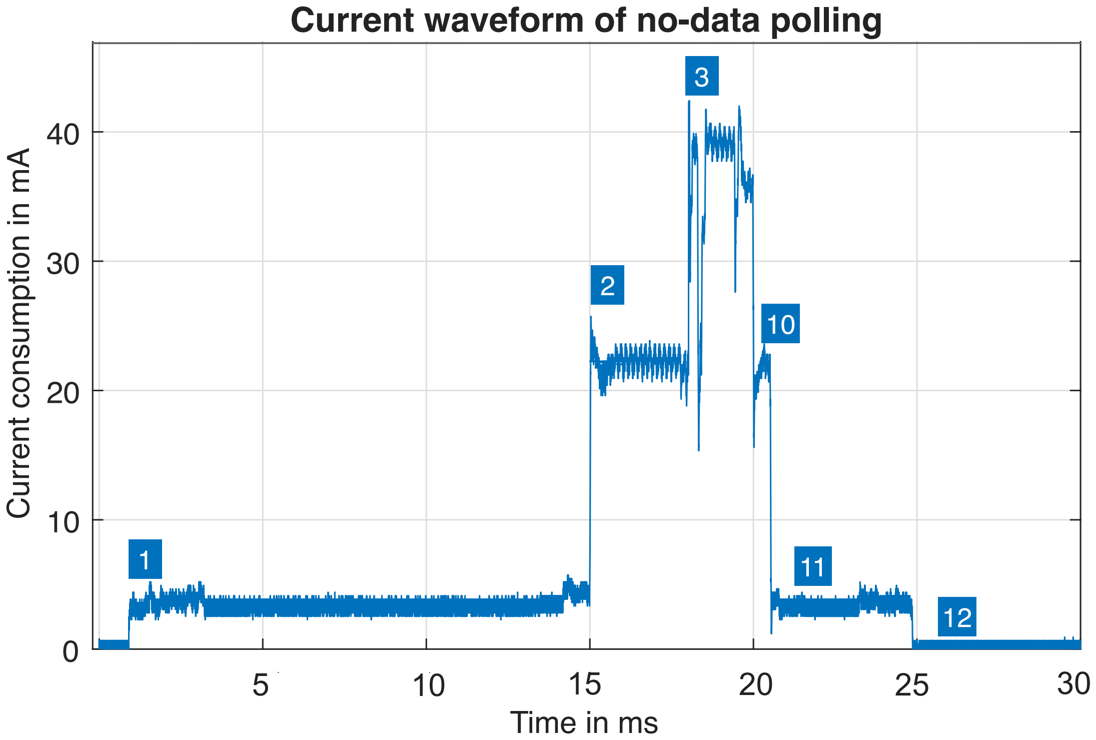


Figure 3.3: Stages of current consumption during a no-data polling event of a sleepy end device.

STAGE 1: In the first stage, a start-up sequence is activated that stops the low-power timers, which control the sleeping and waking process of the SED, and wakes up the microcontroller unit (MCU) to start the analog-to-digital converter (ADC) and acquire some measurements.

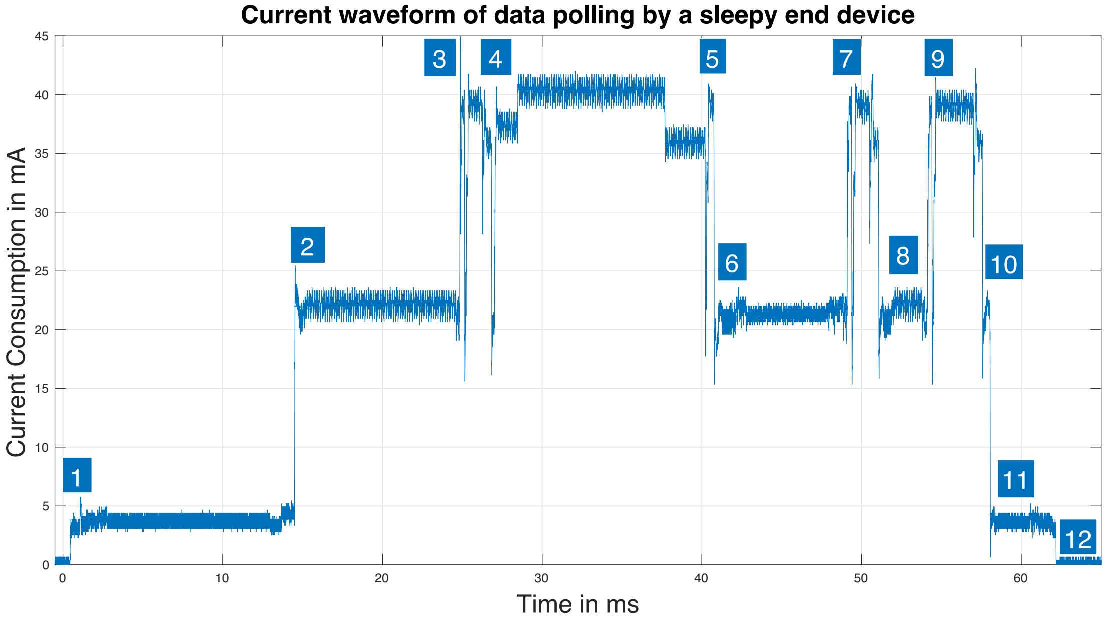


Figure 3.4: Stages of current consumption during a data polling event of a sleepy end device.

<u>STAGE 2:</u> Stage 2 is initiated with the inrush current, which is followed by the sleepy end device deferring for a random period of time when it performs step 1 in the CSMA-CA flowchart (see Fig. 2.2).

<u>STAGE 3:</u> Stage 3 contains the radio initialization, which is followed by the device going into receive (Rx) mode and performing a CCA (step 2 in the CSMA-CA scheme). Then, the sensor device switches to transmit (Tx) mode to transmit a DATA REQUEST IEEE 802.15.4 MAC command frame to the PD and goes back to Rx mode to wait and receive the link layer acknowledgement. The reception of the MAC ACK (link layer ACK) indicates whether there is data being held in the buffer of the parent to be retrieved by the SED or not. If PD replies with ACK frame pending set to false, no further data follows.

<u>STAGE 4:</u> If the frame pending bit in the corresponding field of the ACK frame is set to true (1), the PD has data to transmit, which leads to the CoAP packet reception that is shown in stage 4.

<u>STAGE 5:</u> In 5, the child sends a link layer ACK to the parent to acknowledge the 1-hop successful reception of the foregoing packet.

<u>STAGE 6:</u> The stages that follow include a series of actions that have as purpose the transmission of the SED's response, which is piggybacked in a CoAP ACK. Before sending the reply, the sleepy node performs another session of MAC polling, which starts with the random backoff delay and is followed by a CCA, as shown in stage 6.

STAGE 7: A DATA REQUEST frame and a MAC ACK are exchanged to ensure no further data is to be received from the parent device.

STAGE 8: The node performs CSMA-CA one last time and delays for random unit backoff periods before sensing the channel.

STAGE 9: After the channel is sensed idle, the CoAP ACK packet transmission commences, which is concluded with a 1-hop link layer acknowledgement.

STAGE 10: The application layer ACK is processed and the buffer is released. Meanwhile the transceiver changes from active state back to doze.

STAGE 11: While awaiting all tasks to be in idle state, the SED goes gradually to deep sleep mode.

STAGE 12: This stage refers to the deep sleep state, during which the MCU is in deep sleep mode and the transceiver is in a so-called hibernation mode.

It can be noted that the current levels and waveform vary based on the task that is being handled during each time instance. Furthermore, it is apparent that the wireless transceiver operation has the highest current consumption.

3.3 Basic model

In this section, the intricacies of the SED battery lifetime model in the studied implementation of Thread are discussed, and a set of equations is provided to describe the primary SED tasks that take place during a full operational cycle.

During a SED operational cycle, there exist currents (I_i) associated to the most prevalent activities being executed by the battery-powered sensor devices. These are the following:

- I_{SLEEP}: the sleeping current refers to the current consumption of a duty-cycled node during its deep sleep mode when only the low-power timers for wake-up scheduling are running while the rest of the circuitry is off.
- I_{NHK}: the network housekeeping current is associated with the current consumption of no-data polling events, meaning that the node wakes up and sends a DATA REQUEST MAC command but no further data follows. These periodical frames serve as keep-alive messages by the child device to PD, which are necessary for parent-child connection maintenance.
- I_{OPT}: the operational current corresponds to the data polling events during which application layer packets of variable size along with their acknowledgements are exchanged.

For the purpose of modelling the battery lifetime, it is essential to express the current consumption and time duration of all the implicated tasks relevant to the acquisition, communication, and processing aspects of the SED wake-sleep cycle. The following equations provide a condensed version of the model with respect to the current consumption expressions encountered in no-data polling and data polling events according to the Markov chain presented in Fig. 3.2. The notations that are used and a short description is provided in Table 3.2.

I_{MCP} refers to the current consumption of stages 3 and 7, during which a Data Request frame is transmitted by the SED and a MAC ACK is received to indicate a successful frame transmission. The current consumption of the stages 1 ($I_{1\mathrm{a}}$), 10 ($I_{1\mathrm{b}}$), and 11 ($I_{1\mathrm{c}}$) are fixed and repeated every time at the beginning and end of the SED wake-up cycle. I_{NHK} in Eq. 3.14 refers to the current consumption of the whole waveform depicted in Fig. 3.3. Finally, I_{OPT} in Eq. 3.17 is the final expression of the current consumption of all stages portrayed in Fig. 3.4, which are described in detail in Section 3.2.

Table 3.2: Main symbols used in the battery lifetime model

Symbol	Description
NB	number of CSMA backoffs
R	number of MAC retransmissions
$\mathrm{I}_{\mathrm{MCP}}$	MAC polling current
$\mathrm{I}_{\mathrm{DRQS}}$	Data Request transmission current
$\mathrm{I}_{\mathrm{MACK}}$	MAC ACK reception current
$\mathrm{I}_{\mathrm{RCH}}$	random channel access current
$\mathrm{I}_{\mathrm{CCA}}$	clear channel assessment current
$\mathrm{I}_{1\mathrm{a}}$	current consumption of stage 1 in Fig. 3.4
$\mathrm{I}_{1\mathrm{b}}$	current consumption of stage 10 in Fig. 3.4
$\mathrm{I}_{1\mathrm{c}}$	current consumption of stage 11 in Fig. 3.4
$\mathrm{I}_{\mathrm{pktRx}}$	current consumption of the CoAP packet reception process
I_{Rx}	representation of CoAP packet Rx current and MAC ACK
$\mathrm{I}_{\mathrm{pktTx}}$	current consumption of the CoAP ACK transmission process
I_{Tx}	representation of CoAP packet Tx current and MAC ACK
$\mathrm{T}_{\mathrm{cycle}}$	full operational cycle = 1h
$\mathrm{N}_{\mathrm{NHK}}$	amount of network housekeeping events per full cycle
$\mathrm{N}_{\mathrm{OPT}}$	amount of useful operations per full cycle

$$I_{\mathrm{MCP}} = \frac{I_{\mathrm{DRQS}} \cdot t_{\mathrm{DRQS}} + I_{\mathrm{MACK}} \cdot t_{\mathrm{MACK}}}{t_{\mathrm{MCP}}} \tag{3.13}$$

$$I_{\mathrm{NHK}_0} = \frac{I_{1\mathrm{a}} \cdot t_{1\mathrm{a}} + I_{1\mathrm{b}} \cdot t_{1\mathrm{b}} + I_{1\mathrm{c}} \cdot t_{1\mathrm{c}}}{t_{\mathrm{NHK}_0}}$$

$$I_{\mathrm{NHK}_1} = \frac{\left[(\mathrm{NB}+1) \cdot (I_{\mathrm{CCA}} \cdot t_{\mathrm{CCA}}) + I_{\mathrm{RCH}} \cdot t_{\mathrm{RCH}} + I_{\mathrm{MCP}} \cdot t_{\mathrm{MCP}}\right] \cdot (\mathrm{R}+1)}{t_{\mathrm{NHK}_1}}$$

$$I_{\mathrm{NHK}} = \frac{I_{\mathrm{NHK}_0} \cdot t_{\mathrm{NHK}_0} + I_{\mathrm{NHK}_1} \cdot t_{\mathrm{NHK}_1}}{t_{\mathrm{NHK}}} \tag{3.14}$$

$$I_{\mathrm{pktRx}_0} = \frac{(\mathrm{NB}+1) \cdot (I_{\mathrm{CCA}} \cdot t_{\mathrm{CCA}}) + I_{\mathrm{RCH}} \cdot t_{\mathrm{RCH}} + I_{\mathrm{MCP}} \cdot t_{\mathrm{MCP}}}{t_{\mathrm{pktRx}_0}}$$

$$I_{\mathrm{pktRx}} = \frac{I_{\mathrm{pktRx}_0} \cdot t_{\mathrm{pktRx}_0} + I_{\mathrm{Tx}} \cdot t_{\mathrm{Tx}}}{t_{\mathrm{pktRx}}} \tag{3.15}$$

$$I_{\mathrm{pktTx}_0} = \frac{2(\mathrm{NB}+1) \cdot (I_{\mathrm{CCA}} \cdot t_{\mathrm{CCA}}) + 2(I_{\mathrm{RCH}} \cdot t_{\mathrm{RCH}}) + I_{\mathrm{MCP}} \cdot t_{\mathrm{MCP}}}{t_{\mathrm{pktTx}_0}}$$

$$I_{\mathrm{pktTx}} = \frac{I_{\mathrm{pktTx}_0} \cdot t_{\mathrm{pktTx}_0} + I_{\mathrm{Tx}} \cdot t_{\mathrm{Tx}}}{t_{\mathrm{pktTx}}} \tag{3.16}$$

$$I_{\mathrm{OPT}_0} = \frac{I_{1\mathrm{a}} \cdot t_{1\mathrm{a}} + I_{1\mathrm{b}} \cdot t_{1\mathrm{b}} + I_{1\mathrm{c}} \cdot t_{1\mathrm{c}}}{t_{\mathrm{OPT}_0}}$$

$$I_{\mathrm{OPT}} = \frac{(\mathrm{R}+1)(I_{\mathrm{pktTx}} \cdot t_{\mathrm{pktTx}} + I_{\mathrm{pktRx}} \cdot t_{\mathrm{pktRx}}) + I_{\mathrm{OPT}_0} \cdot t_{\mathrm{OPT}_0}}{t_{\mathrm{OPT}}} \tag{3.17}$$

$$I_{\mathrm{SLEEP}} = 1\ \mu A, \text{ value obtained from chipset datasheet [35]}$$
$$t_{\mathrm{SLEEP}} = T_{\mathrm{cycle}} - (t_{\mathrm{OPT}} \cdot N_{\mathrm{OPT}} + t_{\mathrm{NHK}} \cdot N_{\mathrm{NHK}}) \tag{3.18}$$

The current level and state duration acquired from Eqs. 3.14, 3.17 in combination with the value of $\mathrm{I_{SLEEP}}$ are substituted in Eq. 3.19 for the sake of obtaining the overall average current consumption. The average current consumption is calculated by multiplying the current and time with the amount of instances that each type of event has occurred. For instance, with an operational cycle of 3 600 000 ms and a polling interval of 3000 ms, the amount of wake-ups is calculated using

$W = \frac{T_{\text{cycle}}}{T_{\text{poll}}}$, which is equal to 1200. Out of the 1200 SED wake-ups, a portion of them includes network housekeeping events (N_{NHK}) and the remaining part involves operational events (N_{OPT}). If the node is not awake, then it remains in deep sleep mode, hence (t_{SLEEP}) is determined using Eq. 3.18. The battery lifetime in days with respect to useful operations per hour (q) is derived by applying Eq. 3.20 for every useful operation x from 0 to q.

$$I_{\text{avg}} = \sum_{x=0}^{q}\left(\frac{t_x}{T_{\text{cycle}}} \times I_x\right) + \left(1 - \sum_{x=0}^{q}\left(\frac{t_x}{T_{\text{cycle}}}\right)\right) \times I_{\text{sleep}} \tag{3.19}$$

$$I_{\text{avg}} = \frac{I_{\text{NHK}} \cdot t_{\text{NHK}} \cdot N_{\text{NHK}} + I_{\text{OPT}} \cdot t_{\text{OPT}} \cdot N_{\text{OPT}} + I_{\text{SLEEP}} \cdot t_{\text{SLEEP}}}{T_{\text{cycle}}} \tag{3.20}$$

$$\text{Battery Lifetime [days]} = \frac{\frac{\text{Battery Capacity [mAh]}}{I_{\text{avg}} \text{ [mA]}}}{24} \tag{3.21}$$

3.4 Data collection methods for the model parameters

In the last section of this chapter, the different techniques, which are employed in order to acquire the necessary parameter values for the construction of the mathematical model representing the battery lifetime of a sleepy end device in Thread, are described.

Theoretical method

As regards the modelling in MATLAB, there are a number of aspects that need to be dealt with. Starting with the CSMA-CA mechanism, the random nature of it is modelled by using the command *datasample*, which is an efficient method to sample data from a defined data set with or without replacement and it provides optional weighted sampling. For instance, in order to acquire the appearance frequency of each CSMA backoff stage and assuming that *macMaxCSMABackoffs* = 2, the following syntax is employed:

$$\text{NB} = \text{datasample } (\text{N}_{\text{BKF}}, \kappa, \text{`Weights'}, [(1\text{-}\alpha_{\text{busy}})\ \alpha_{\text{busy}}\ \alpha_{\text{busy}}])$$

which samples κ times out of $\text{N}_{\text{BKF}} = [0, 1, 2]$ using replacement and with probability proportional to the elements of *Weights*. It can be noted that the sample weights are acquired from the busy channel probability because it is the factor that dictates the channel access retries. In a similar manner, the occurrence rate of retransmissions is modeled as follows:

$$\mathrm{R} = \text{datasample}\ (\mathrm{N}_{\mathrm{MAC\_reTx}},\ \kappa,\ \text{`Weights'},\ [(1\text{-}\mathrm{p}_{\mathrm{col}})\ \mathrm{p}_{\mathrm{col}}\ \mathrm{p}_{\mathrm{col}}])$$

where $\mathrm{N}_{\mathrm{MAC\_reTx}}$ = [0,..., *macMaxFrameRetries*] and the weights are proportional to the collision probability due to collisions being the trigger of retransmissions. For reasons of simplicity in illustration, *macMaxFrameRetries* is also set to 2. Lastly, the duration of the random backoff delay is adjusted to the backoff stage and is initialized to the minimum range with every retransmission.

According to the model, the current consumption in one operation per hour and every additional operation is the average of κ code runs. The term κ refers to the amount of samples randomly acquired for the channel access, backoff stage and retransmissions; it lies in the range [100 000, 200 000] in order to get consistent results.

It is worth mentioning that because of the variable arrival rates and service times of the Markov chain in Fig. 3.2, which represents the primary events that occur in a Thread network related to SED operations, the NB and R values are different for each transition with different λ_i and μ_i. This observation entails that the network housekeeping events and the operational events have different NB and R values, and the same applies to the different stages during an operational event that require the performance of CSMA.

Measurement-based method

A specific chipset, which is set up as a SED node in Thread, is utilized as a target platform. Accurate and repetitive measurements of a single task, such as the one identified in stage 1 of Fig. 3.4, are performed to establish the time and the current use. The current drawn from the specific process is measured with the aid of a precise current probe connected to an oscilloscope. After a set of measurements is obtained, the average values for time and current consumption are calculated and inserted in the model.

Fixed-value method

The third method includes the fixed values that are obtained from datasheets and specifications. For instance, the value for the sleeping current is directly taken from the NXP MKW2xDxxx datasheet [35]. The $\mathrm{I}_{\mathrm{SLEEP}}$ value is substantially smaller compared to the majority of the currents involved; thus, a precise measurement of it is hindered.

Other values

The operational cycle T_{cycle} in the model is 1 hour and the polling period $\mathrm{T}_{\mathrm{poll}}$ can be adjusted to different values ranging from 125 ms to 3000 ms. The amount of

SED wake-ups is determined by $\frac{T_{\text{cycle}}}{T_{\text{poll}}}$, and the maximum number of useful operations per hour is fixed at 1000 in order to be able to compare the different test cases. Furthermore, in the application layer, CoAP packets of variable payload length can be exchanged up to the value representing the boundary of fragmentation in IEEE 802.15.4 networks. A common battery capacity for WSN applications is in the range of couple thousands milliampere hour (mAh). In this model, the selected capacity is 2200 mAh.

After all the values for the average current consumption and duration of each phase in no-data polling or data polling scenarios are collected, the equations describing the processes that take place in the MAC sublayer and APP layer are derived and inserted in the model. The average aggregate cost in time and current is obtained for all executed tasks, and the average battery lifetime with respect to useful operations per hour is attained with the aid of Eqs. 3.19 and 3.21.

Chapter 4

Experimental Setup

The purpose of this chapter is to present the experimental design employed in order to collect all data necessary for the implementation and validation of the PCM. In the pages that follow, a thorough description of the experimental setup including both hardware and software is provided, along with details about the data collection techniques.

4.1 Experimental design

Fig. 4.1 illustrates the experimental setup that is employed to perform the tests. The border router is positioned on a desk and a mix of SEDs and end devices (EDs) (exact number is dependent on the test case) are spread around it in an open-plan space in the ABB Corporate Research offices, 1-hop away from the BR. With a Tx power of 0 decibel-milliwatt (dBm), the maximum 1-hop distance for nodes placed on the same floor is approximately 35 meters in Thread. In the deployed network, the 1-hop ranges from a distance of 2 meters up to 15 meters. The end devices are not equidistant from the border router, but they are randomly placed around the office where power sockets are present and within the maximum 1-hop distance. The nodes communicate using channel 11 of the 2450 MHz ISM band.

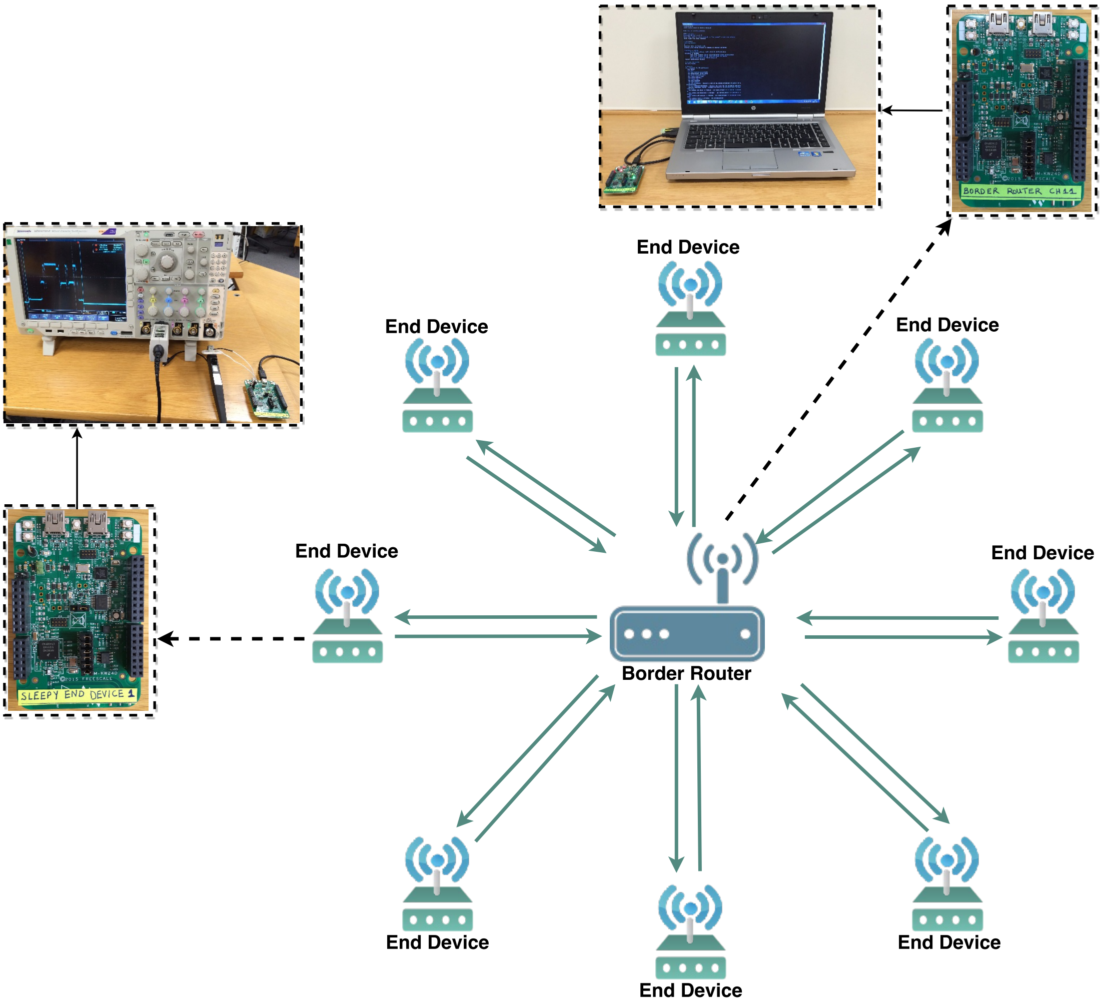
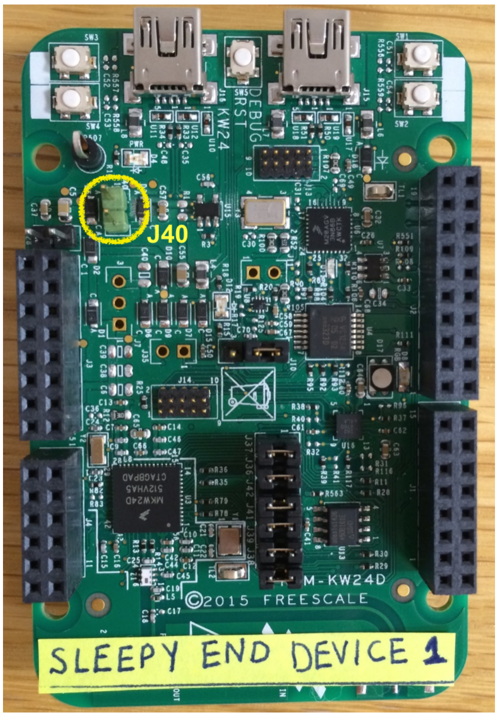


Figure 4.1: Illustration of the experimental setup.

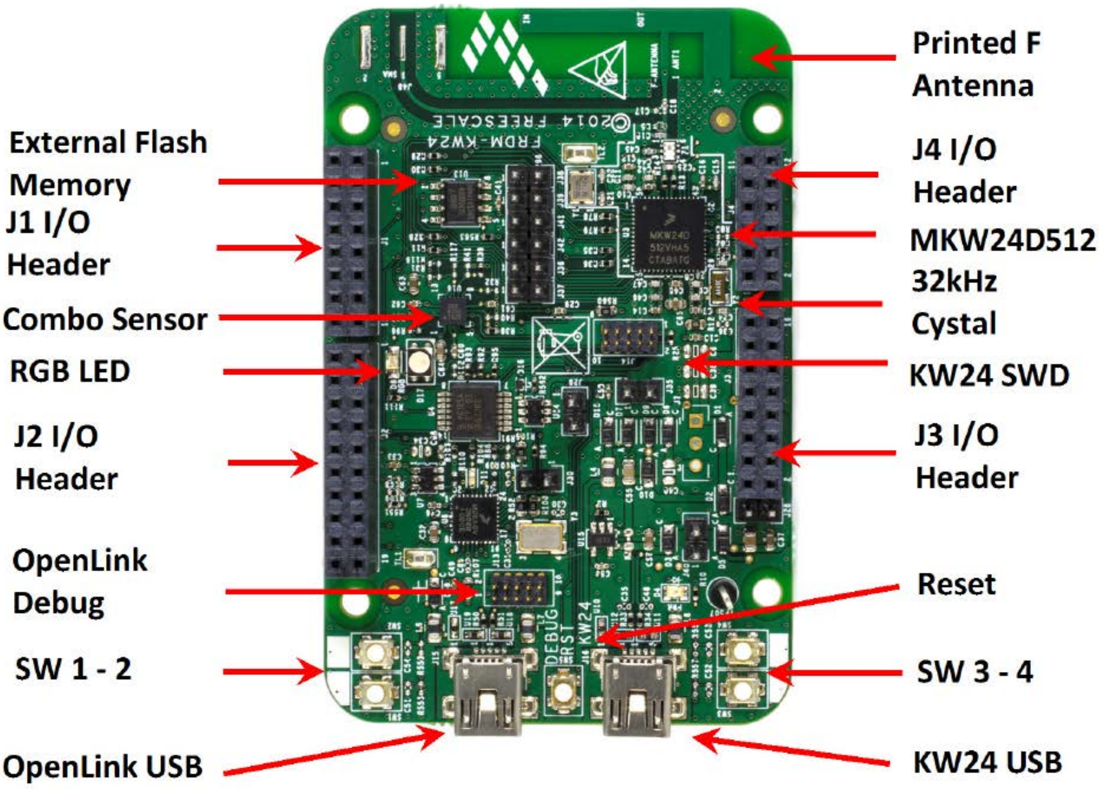
4.2 Hardware/Software

The hardware platform for the sensor nodes is realized with FRDM-KW24D512 evaluation boards from NXP. The outline of the board is portrayed in Fig. 4.2 with information about the components and the Input/Output (I/O) headers. The FRDM-KW24D512 boards belong to the KW2xD family and their MCUs are based on the ARM Cortex-M4 CPU core. They have reliable, secure, low-power and high-performance 2.4GHz IEEE 802.15.4 compliant radio transceivers with support for various 802.15.4 protocol stacks, such as Thread, ZigBee, WirelessHART, and ISA 100.11a [35].

As regards the current consumption measurements, the required equipment includes a MDO4104-6 mixed domain oscilloscope and a TCP0030A alternating current/direct current (AC/DC) current probe from Tektronix. The current from the board is measured using the probe that is attached to the oscilloscope; the configuration is depicted in Fig. 4.3a.

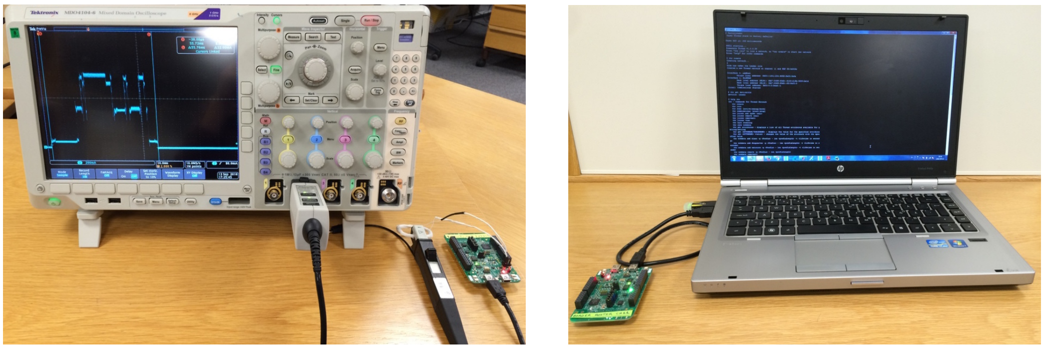


(a)



(b) [35]

Figure 4.2: FRDM-KW24D512 boards



(a)

(b)

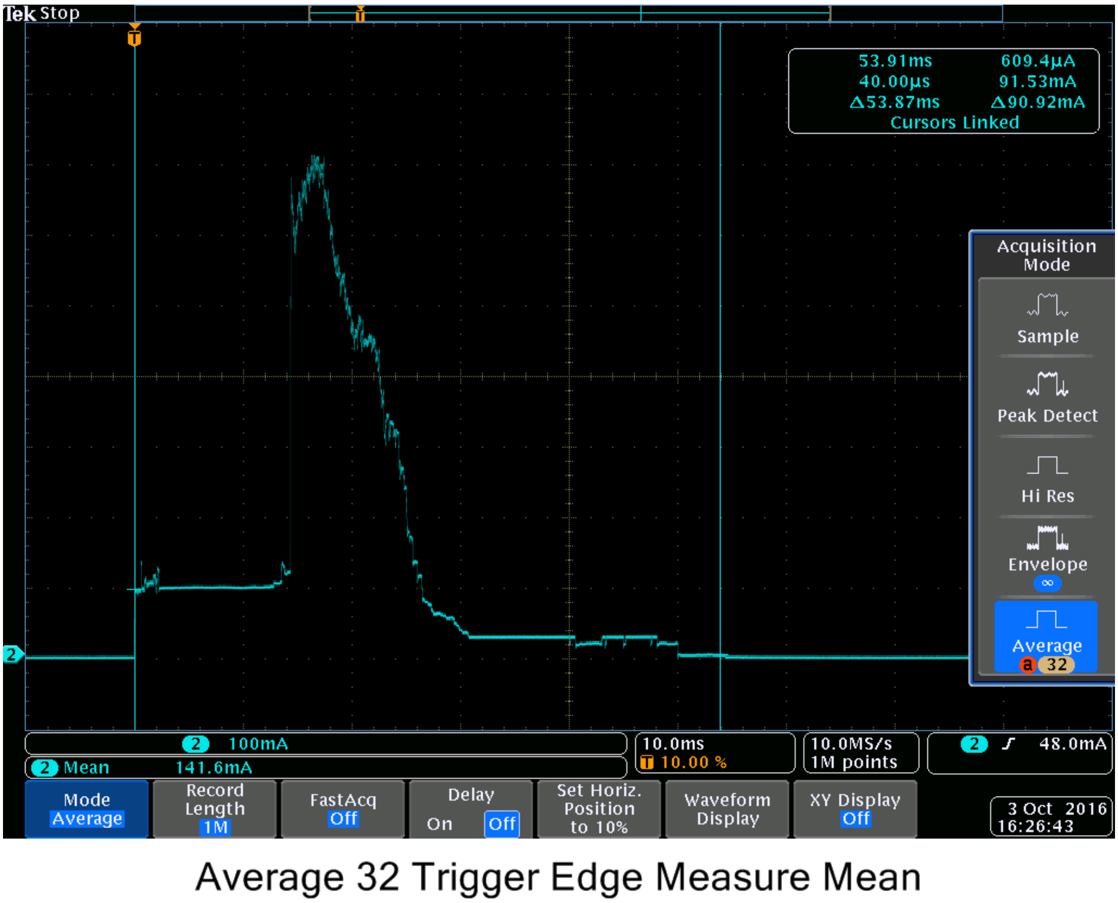
Figure 4.3: Measurement and testing equipment (a) oscilloscope and current probe attached to the SED (b) the router, namely the parent device, being controlled from the shell.

The source code for each device role in Thread is implemented in IAR Embedded Workbench for ARM (version 7.60.2) and once compiled, it can be downloaded to the NXP boards using a USB port. All Thread devices, apart from SEDs, can be connected via one of their USB ports to a PC using a shell command line interface (see Fig. 4.3b). The shell interface is used to send different commands, such as create or join a network, send CoAP commands, and configure simple parameters.

4.3 Validation environment

The validation set is created by means of oscilloscope measurements, which has an acquisition mode that performs waveform averaging over multiple sequences of the same executed operation. With a stable oscilloscope trigger, the duration and the mean current consumption are measured from the average waveform capture. As can be seen in Fig. 4.4 that provides a demonstration of the way average measurements are obtained, the mean duration of the no-data polling event is 53.87 ms and the average current consumption is the mean measurement between the linked cursors (141.6 mA) divided by 30.

The current probe is attached to a copper wire that is connected to the J40 cut-trace header of the FRDM-KW24D512 evaluation board (see Fig. 4.2a). The wire has 30 windings in order to amplify the involved currents by the same amount, with the ultimate objective of acquiring more precise measurements with respect to very low current levels (below 1 mA). For that reason, the current levels in the oscilloscope screen are in the range of hundreds mA.



```
53.91ms      609.4µA
40.00µs      91.53mA
Δ53.87ms     Δ90.92mA
Cursors Linked
```

```
2 100mA            10.0ms       10.0MS/s     2 ƒ 48.0mA
2 Mean 141.6mA     T 10.00 %    1M points
```

Figure 4.4: Oscilloscope capture of the averaging operation over 32 waveforms for the case of no-data polling event in Thread.

Chapter 5

Performance Evaluation

The test cases for the evaluation of the model are described in this chapter, followed by the verification of the model. The efficiency of the proposed model in predicting the battery lifetime of a SED in Thread is validated using the validation environment and a specific metric to acquire the error rates. Finally, a concise summary of the acquired results is provided.

5.1 Test Cases

In order to evaluate the robustness of the model under various network conditions, different test cases are employed, which are listed below:

- Network scenario 1 (NS1) with N nodes *(small network)*: *low traffic* with burst mode on during which packets are transmitted at every wake-up instance for SED_0 and the remaining N-1 end devices perform no-data polling asynchronously compared to the schedule of SED_0.
- Network scenario 2 (NS2) with $X > N$ nodes to realize a *large network: low traffic* with burst mode on and a setup similar to NS1.
- Network scenario 3 (NS3) with N nodes *(small network)*: *high traffic* with burst mode on during which packets are transmitted at every wake-up instance for SED_0, and half of the remaining N-1 end devices perform no-data polling while the other half receive and acknowledge CoAP packets every 100 ms.
- Network scenario 4 (NS4) with $X > N$ nodes to realize a *large network: high traffic* with burst mode on and a setup similar to NS2. The same amount of end devices as NS3 receive and acknowledge CoAP packets every 100 ms.

The selected values for N, and X are five, and ten respectively. The amount of end devices added to create additional load in test cases NS3 and NS4 are two.

Furthermore, different polling intervals are tested and validated to identify the effect the wake-up frequency has on the average current consumption and hence battery lifetime. Apart from variable $\mathrm{T}_{\mathrm{poll}}$, scenarios with increasing CoAP payload size are studied as well.

5.2 Validation metric

For the purpose of computing the prediction errors, and validating the efficiency of the model with respect to battery lifetime estimation, the evaluation metric normalized mean absolute error (NMAE) is employed, which is equal to:

$$\mathrm{NMAE} = \frac{1}{\bar{y}_{\mathrm{osci}}}(\frac{1}{q}\sum_{i=1}^{q}|y^{(i)}_{\mathrm{osci}} - y^{(i)}_{\mathrm{model}}|) = \frac{1}{\bar{y}_{\mathrm{osci}}}(\frac{1}{1000}\sum_{i=1}^{1000}|y^{(i)}_{\mathrm{osci}} - y^{(i)}_{\mathrm{model}}|) \tag{5.1}$$

where y_{model} contains the model-predicted values, whereas y_{osci} and $\bar{y}_{\mathrm{osci}}$ contain the values from the validation set. The absolute difference of the model-predicted, and the oscilloscope-measured value is calculated for increasing amount of useful operations/hour i until the maximum value q=1000. Then, the mean value of the absolute difference is obtained and normalized with respect to the mean oscilloscope-measured battery lifetime.

The MATLAB code for the calculation of NMAE is run 250 times. Afterwards, the mean (μ) and standard deviation (σ) values of NMAE are acquired in order to present the validation results in percentage form.

5.3 Applicability of the model

The model has been applied using various scenarios, which include variable wake-up frequency, packet length and increasing network size and traffic load with the implementation of the four test cases.

Unless the polling interval is variable, the graphs have been plotted using as polling period the value $\mathrm{T}_{\mathrm{poll}}$=3000 ms, which is considered the default wake-up frequency of SEDs in the studied implementation of Thread. In addition, the useful operations per hour have the maximum value of 1000. If the battery lifetime is not a function of the operations/hour, then the graphs portray the obtained battery lifetime after 1000 operations/hour have been performed by the SED. As regards the length of the packets being transmitted and received, the following packet payloads are assumed for Tx and Rx in cases of non-variable packet sizes:

- Rx payload length = 10 bytes and 20 bytes
- Tx payload length = 10 bytes

5.3.1 Wake-up frequency

Fig. 5.1 portrays a three-dimensional bar plot, which indicates the proportion of the operational, housekeeping, and sleeping current as a function of polling interval (1000-3000 ms) and useful operations/hour (0-1000). The depicted results in the plot are acquired from the model. The figure demonstrates that for smaller polling intervals, the network housekeeping current dominates the current consumption and the proportion of the operational current has no substantial effect on battery lifetime.

The battery lifetime versus the useful operations/hour for six different polling intervals is shown in Fig. 5.2. It can be noted that the plotted data include both the model predictions, which are represented by solid lines, and the experimentally acquired battery lifetime, which is denoted by dashed lines. As can be seen in the figure, there is a decline in sensor longevity as the polling interval decreases because the number of wake-ups (W) grows drastically. For instance, with T_{poll}=3000 ms and T_{poll}=125 ms, W is equal to 1200 and 28800 wake-up instances respectively, which shows the intensity in the difference of the wake-up frequency. For polling intervals below 1000 ms, the data points seem to follow a straight line, which is not the case. This is simply further proof that the operational current, which varies with the amount of operations per hour, ceases to have as strong influence on battery lifetime as the network housekeeping current.

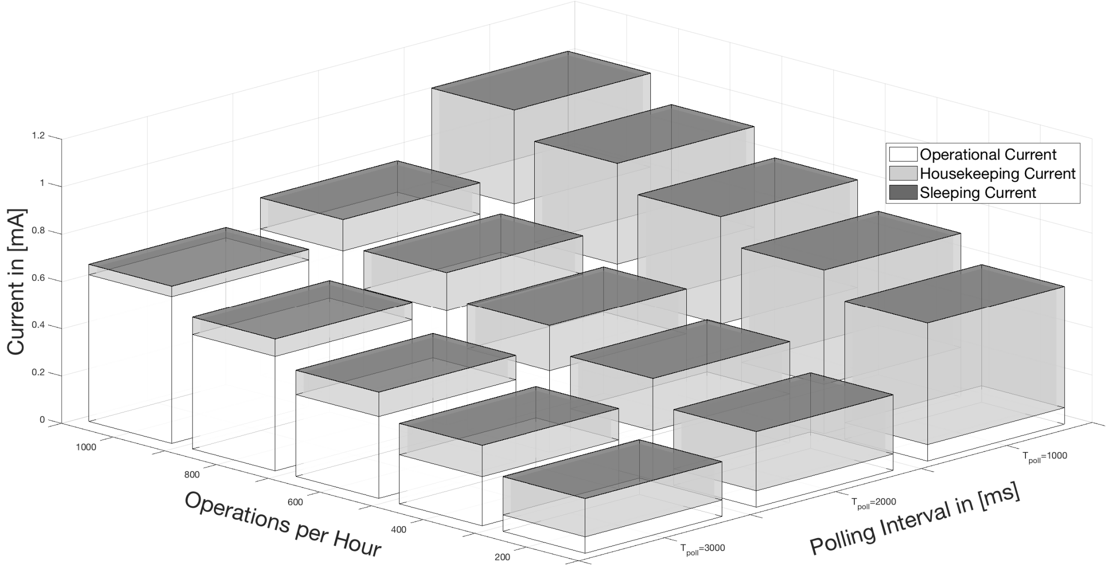


Figure 5.1: Proportionate current consumption of the sensor node with respect to operations per hour and polling intervals.

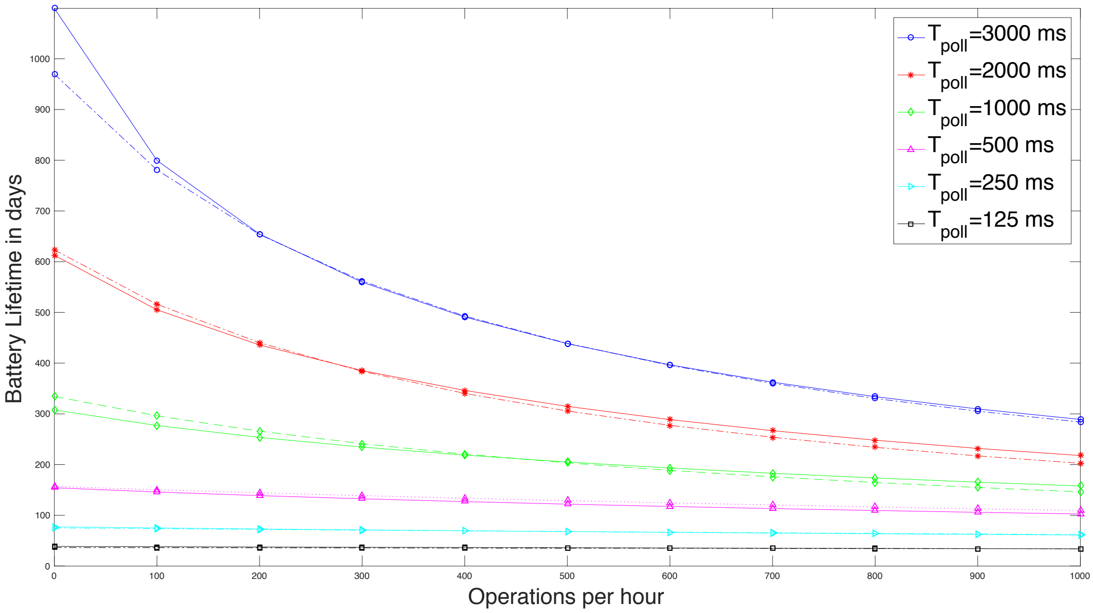


Figure 5.2: Predicted battery lifetime from the model and its proximity to the experimentally acquired average battery lifetime.

5.3.2 Packet Length

Figs. 5.3 and 5.4 depict the effect of increased packet payload transmission on the battery lifetime of a SED in Thread. The estimated battery lifetime with respect to useful operations per hour for variable CoAP payloads can be seen in Fig. 5.3. Due to the need for a greater amount of time required to handle the packet transmission and processing in all related layers, it is expected that the increased length of the packet would have the opposite effect on lifetime, thus leading to less battery running time because of prolonged current consumption.

Furthermore, the effect of fragmentation can be observed in the transmission of a packet having a payload of L=80 bytes (red solid line plot with asterisk markers in Fig. 5.3), which exhibits the drastic decrease in battery lifetime in comparison with the other packet payloads. The bar plot in Fig. 5.4 illustrates the battery lifetime decrease in days compared to the case of the 10-byte payload transmission for four different payloads: L=30, 50, 70, and 80 bytes. The 10-byte payload transmission (blue dash-dot line with circle markers in Fig. 5.3) is used as a reference battery lifetime for the plotting of the additional decrease in lifetime as a result of the transmission of larger packets. A MAC frame in an IEEE 802.15.4 network (such as the one realised by Thread) consists of the payload, which is prefixed with a variable-sized header and appended with a variable-sized footer. In Thread, a packet with a payload of 80 bytes exceeds the maximum size (127 bytes) of a MAC protocol data unit (MPDU), thus necessitating fragmentation. If compared to the case with the 70-byte payload transmission, it can be noted that with the addition of just 10 bytes in the packet payload, the SED drains the battery faster, which causes the battery to last for a shorter amount of time (approximately 80 days less).

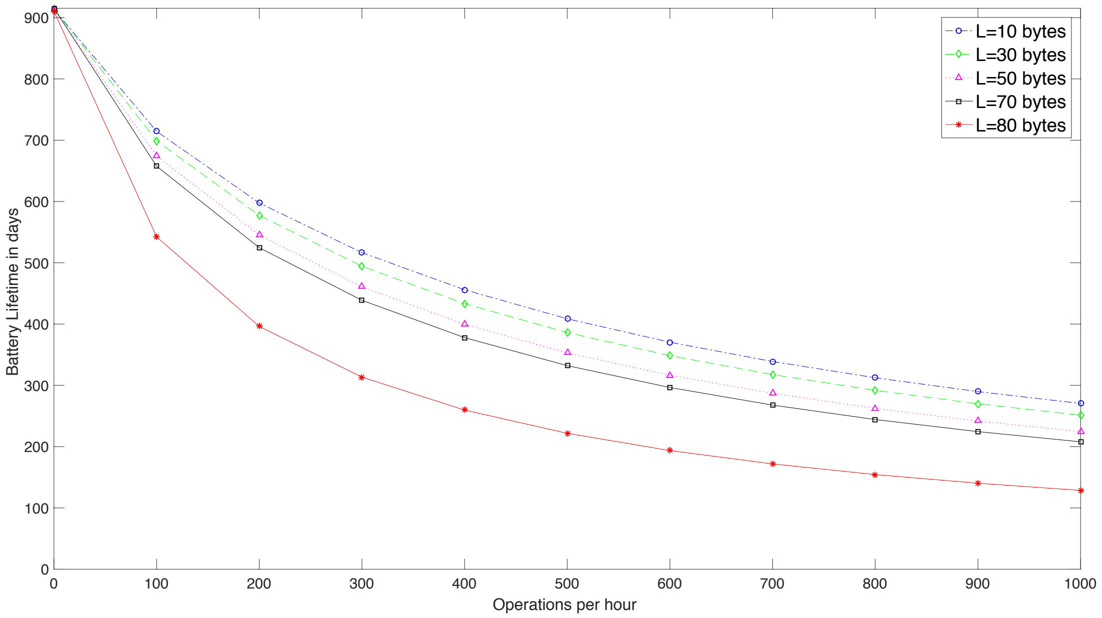


Figure 5.3: Battery lifetime for transmission of different packet payload lengths.

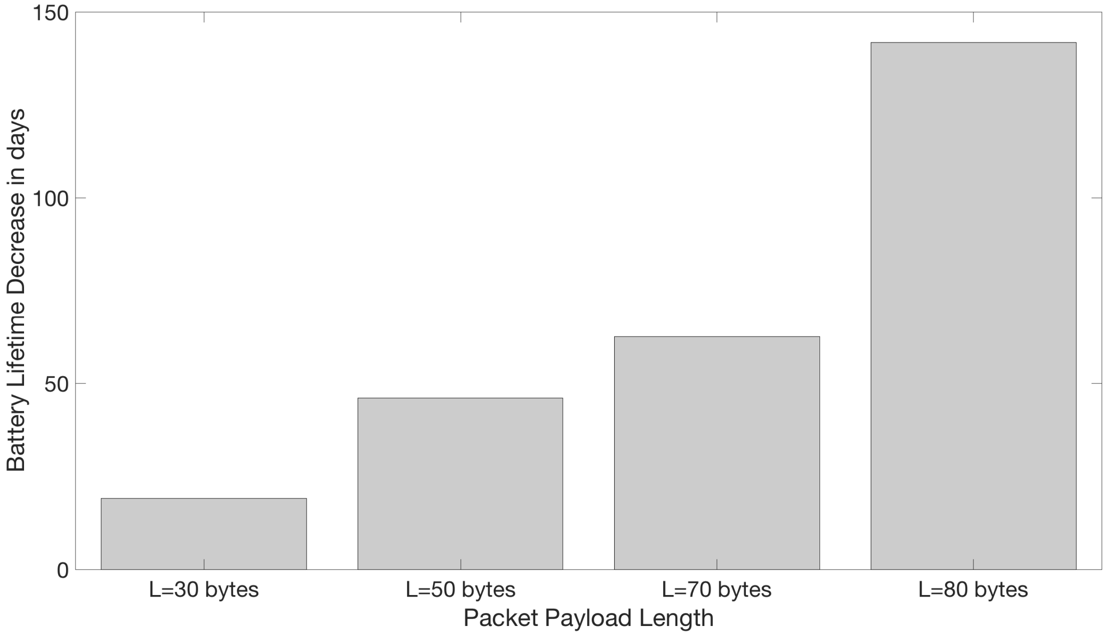


Figure 5.4: Decrease in battery lifetime compared to a 10-byte payload transmission.

5.3.3 Network size and traffic

After the implementation of the four test cases using the model, the obtained results are illustrated in Fig. 5.5. A benefit that stems from the model is substantiating the effect that the number of sleepy devices attached to the same parent has on the battery lifetime. As portrayed in Fig. 5.5, the increase in the amount of children indicates a decrease in sensor life expectancy, which is an anticipated behaviour. As the network contention rises with the addition of end devices and the introduction of network load, more frequent CSMA backoffs and retransmissions are encountered as a result.

The unexpected behaviour, however, is the intensity in lifetime decrease as the nodes increase in amount from 5 to 10 in both cases of *load off* and *load on*. The added network load in network scenarios three and four had a substantial effect on battery lifetime, if NS1 and NS2 are used as reference points. On the other hand, the network handled the insertion of additional 5 SEDs without the occurrence of highly frequent CSMA backoffs and retransmissions. This difference can be observed in the predicted battery lifetime of the pairs NS1-NS2 and NS3-NS4. A possible explanation for the observed behaviour can be the selected IEEE 802.15.4 MAC attributes used in this implementation, which due to confidentiality reasons cannot be disclosed. The specific values of these attributes can influence energy efficiency, latency, and reliability in a different manner. In this case, the value of one attribute or a combination of them leads to fewer collisions in the initial backoff stages as well as it reduces the per-operation energy consumption.

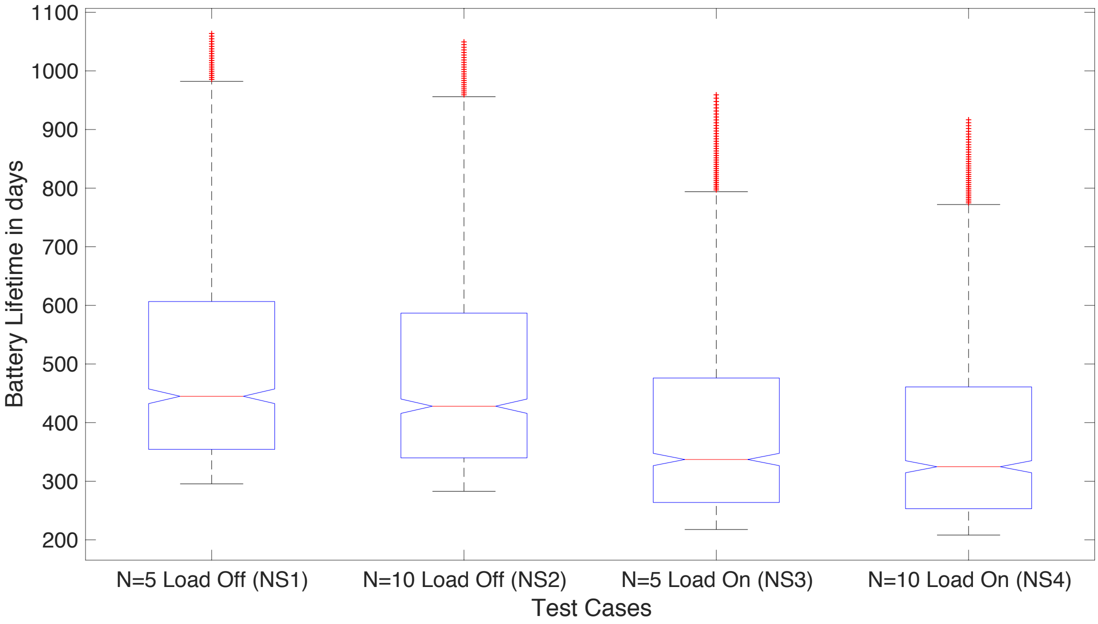


Figure 5.5: Battery lifetime of a sleepy end device under the influence of variable network contention.

5.3.4 Accuracy of the model

The numerical validation of the model is presented in Table 5.1 and 5.2, where the NMAE is employed to prove the validity of the model, and to further support the graphical results presented in Fig. 5.2. Polling intervals from 3000 to 125 ms are validated and the NMAE lies in all cases below 5 %. The wake-up frequency of SEDs is not meant to be high in Thread, so generally T_{poll} of 1000 and above are used in current implementations of Thread. The model exhibits an adequate capability to predict the battery lifetime in occasions of variable polling periods, as can be observed in Table 5.1. Furthermore, the implementation of the four test cases, which involved a *small* and *large* network set-up with *load off* and *load on*, using experimental means indicates a validation error rate less than 4 %. The validation results shown in Table 5.2 prove that the model can effectively represent network contention.

Table 5.1: Validation with NMAE for different polling periods

Polling Period [ms]	NMAE $[\mu \pm \sigma]$
$T_{poll} = 3000$	2.83% ± 0.54%
$T_{poll} = 2000$	2.78% ± 0.05%
$T_{poll} = 1000$	3.95% ± 0.02%
$T_{poll} = 500$	4.33% ± 0.09%
$T_{poll} = 250$	1.42% ± 0.03%
$T_{poll} = 125$	3.79% ± 0.12%

Table 5.2: Validation with NMAE for different test cases

Test Case	NMAE $[\mu \pm \sigma]$
NS1	1.56% ± 0.31%
NS2	2.82% ± 0.60%
NS3	2.65% ± 0.53%
NS4	3.81% ± 0.81%

5.4 Battery Lifetime Improvement

In this section, a few instructive observations regarding the optimisation of the SED current consumption are presented.

Starting with the topic of polling periods and their influence on the radio duty cycle and ultimately the battery lifetime, it is clear that since the radio activity duration is up to a certain extent fixed with every type of operation, the duty cycle within the operational cycle is going to be high with small polling intervals (in the range of hundreds ms), which is an undesirable feature for SEDs. Furthermore, the number of devices and the amount of children are generally large because the redundancy is added to deal with the nature of low-power and lossy networks, such as the networks in Thread. For that reason, there is no critical need for the sleepy devices to wake-up frequently to observe a phenomenon because there can be multiple nodes monitoring the same phenomenon, so it is important to decide on the balance between battery consumption and message latency related to the larger polling periods.

From the previous sections, it is evident that different types of current (I_{NHK}, I_{OPT}, I_{SLEEP}) influence the overall battery lifetime depending on the polling period. Determined by the specific measure of T_{poll}, it is advisable to further reduce the current and/or duration for the transmission of the MAC Data Request for small polling periods in network housekeeping events. Furthermore, it is advantageous to decrease the current consumption and/or duration of the packet transmission and reception in data polling events.

The MAC sublayer of IEEE 802.15.4 implements its own retry mechanism by retransmitting a frame, if the 1-hop link layer acknowledgement has not arrived. The retransmission is repeated a specific amount of times. Due to the star network topology of the implementation studied in this project that contains nodes positioned 1-hop away from the parent device, it is apparent that the MAC retransmissions are adequate to ensure reliability within the network. However, if there are strict requirements as regards packet delivery success, one can employ retransmission mechanisms in the application layer. With this in mind, the CoAP protocol implements a retransmission mechanism suited for constrained networks and triggers the retransmission of a packet 2 seconds after the time indicated in the packet departure timestamp contained in the source node. However, it is detrimental to the battery lifetime of a SED to stay awake for 2 seconds in an idle state, awaiting the successful reception of the CoAP ACK by the parent device.

Finally, the need for application layer acknowledgements is superfluous in some cases during which the sleepy device is not required to reply to the message by piggybacking its reply in the CoAP ACK. The link layer acknowledgements are sufficient to ensure successful packet transactions in the majority of cases. Considering the process of transmitting a CoAP ACK, which entails a second MAC polling requiring the CSMA-CA algorithm to be performed twice before finally transmitting the ACK packet, it is prudent to bypass this step when there are no strict performance requirements in place.

5.5 Summary

The proposed model in this thesis report has addressed a number of points; the key ones can be summarized as follows:

- The model can predict with a small forecasting error (below 5%) the battery lifetime of a SED and various parameters can be altered and tuned accordingly to infer the magnitude of the effect on network lifetime. Few of these parameters are: the polling interval of SED, the total number of useful operations being performed per hour, and the MAC attributes related to the CSMA-CA algorithm in the IEEE 802.15.4 standard.
- The model takes into consideration different packet sizes that are being transmitted and received up to the fragmentation threshold in IEEE 802.15.4 networks. However, the model can easily be adjusted to account for fragmentation in cases the application is in need of exchanging larger packet sizes. Such an example is provided in the model applicability section.
- The attachment of additional children nodes to the same parent as well as the increased network size and traffic are represented in the model, which evaluates the new network conditions and adjusts the probability of channel access failures and collisions that consequentially affect the battery lifetime.
- The proposed model along with the obtained results can prove to be informative as well as beneficial not only to researchers and device developers but also to installers and system integrators.
- If the results acquired from the experimental validation are considered, the model together with the methods that are employed are accurate judging from a perspective of applicability. That is to say, the PCM can serve as a tool to facilitate installers and provide them with instructions for taking appropriate measures in order to extend the battery lifetime of sensor devices.

Chapter 6

Concluding Remarks

6.1 Conclusions

In this degree project, several features of a battery-powered sensor node are considered in order to implement a model that can profile the current consumption of a WSN node with a prediction error within a satisfactory range (less than 5 %). The performance of the model is assessed with the aid of MATLAB and validated by employing a hardware test bed as means to reach the thesis objectives.

As the summary in the preceding section makes clear, the proposed model provides a sufficient estimation of the expected battery lifetime under numerous conditions. Nevertheless, one can argue that since the analysis is performed on certain hardware devices, this approach generates non-universal results as it is hardware-specific and the reproducibility of the experimental environment poses also further difficulties. A single model, however, cannot cover the needs of the vast heterogeneity that exists within IoT as regards battery-operated nodes. In spite of that, there is one ubiquitous communication standard for low-power and lossy networks, namely IEEE 802.15.4, that is the common denominator of plenty communication protocols. This suggests that while the results cannot be applied to all sensor devices running on battery, they are still applicable to a number of cases in the field of WSNs.

6.2 Future Work

Concerning future work, alternative perspectives from which the battery lifetime could have been studied involve the topic of fragmentation in WSNs that employ IEEE 802.15.4 for the underlying layers and provide IP connectivity. Up to this point, there was no identified need to send larger MAC frames than the maximum size of 127 bytes. The reason is that, in the majority of cases, the sensor nodes employ the *sense-and-send* mechanism instead of collecting samples, storing them and then transmitting them to a gateway, which is the so-called *batch-and-send* method. However, as more advanced applications are being developed, larger packet sizes are required and along with the batch-and-send mechanism and firmware upgrades, fragmentation is going to take a prevalent position in sensor node operations. Packet fragmentation can take a toll on battery lifetime, so it is essential that its effects are studied sufficiently.

Due to memory limitations in the parent device, the network size was restricted to 10 children devices. Another possible topic for future work can be a large-scale deployment up to 50 children devices to monitor the way CSMA-CA can deal with this amount of network contention, and to discover the impact on the battery lifetime of individual sensor devices. Furthermore, a long-term validation test, ranging from several hours to days, can prove more reliable in establishing the efficiency of the model. Due to the longer duration, any hidden sources of energy waste can manifest themselves and be identified.

6.3 Reflections

IoT advocates a form of connectivity that takes place *anywhere*, *any time* and spans *anything* instead of just anyone. This vision of information ubiquity that is being promoted creates an omnipresent intelligence that is indistinguishable to the human eye. A direct benefit that stems from such a new reality is that it revolutionises in a good way for the most part the way people live, work, communicate and commute; IoT has become an indispensable part of our existence, whether people realise it or not. Specifically, smart homes and buildings have been a feature of science fiction writing for many years, but only after the beginning of 20$^{\text{th}}$ century, they started being implemented. Currently, this field has countless applications, which create intelligent environments able to “think” for themselves and co-operate with other parts to perform various functions.

IoT can be considered as an immense blessing, however, there are numerous issues that can arise, if the evolving landscape introduced by IoT is not handled with caution. For instance, violation of privacy, and security matters related to the vast amount of information being “seamlessly” circulated pose serious ethical issues. The transparency of IoT blurs the lines between the physical and virtual world, excluding the human element from the interaction of the physical objects, as if this interaction occurs in a parallel universe. Questions, such as, who are the owners of the information that is propagated and how much control can the owner have over the dissemination of the IoT-collected data, need to be contemplated.

As regards the specific topic of this degree project, which is modelling the battery lifetime of a sensor device that utilizes a new communication technology, namely Thread, it is essential to test the technology before implementing it in commercial products to ensure that it is indeed capable of delivering the promised benefits that the proprietors claim. One dimension of the evaluation is the study of its energy efficiency. Furthermore, after the suitability for commercial products has been verified, the model presented in this thesis and the acquired results can serve as a tool for device manufacturers and system integrators to select the appropriate hardware, battery capacity and optimize the communication protocol parameters in order to achieve the desired device battery lifetime for the intended application.

With respect to the disposal methods of the IoT devices, which constitute the last stage of the life-cycle analysis of a product/technology, focus needs to be put on the end-of-life practices for such devices. Irresponsible recycling due to e-waste can lead to dramatic impacts on the health of humans and on the environment. Electronic waste can give rise to major pollutants that are caused by the emissions from the decomposition or combustion of circuit boards and wiring, among other things.

The sensor devices, the required batteries, and the need for battery replacement or the option of energy harvesting in the power management unit of the device are all important decisions that can be made in advance in order to avoid unnecessary energy waste and additional burden on the environment and its natural resources. Being able to take proactive decisions and enforcing some of the application-specific and energy-efficient improvements can boost the economy and guard Earth's sustainability. Considering that the IoT envisions billions of connected devices, improving the energy efficiency of the ones that are battery-operated and extending their life expectancy is of utmost importance and can have a meaningful impact on Earth's natural resources that are diminishing day by day.

Bibliography

[1] N. Khalil, M. Abid, D. Benhaddou, and M. Gerndt, “Wireless sensors networks for Internet of Things,” in *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2014 IEEE Ninth International Conference on*, April 2014, pp. 1–6.

[2] K. Ashton, “That ‘Internet of Things’ Thing,” *RFID Journal*, June 2009.

[3] R. Van Kranenburg, *The Internet of Things: A Critique of Ambient Technology and the All-Seeing Network of RFID.* Insitute of Network Cultures, Amsterdam, September 2008.

[4] L. D. Xu, W. He, and S. Li, “Internet of Things in Industries: A Survey,” *Industrial Informatics, IEEE Transactions on*, vol. 10, no. 4, pp. 2233–2243, Nov 2014.

[5] “Internet of Things in 2020: A Roadmap for the Future,” INFSO D.4 Networked Enterprise & RFID INFSO G.2 Micro & Nanosystems in cooperation with RFID Working Group of the European Technology Platform on Smart Systems Integration (EPOSS), Sep. 2008.

[6] G. Y. Li, Z. Xu, C. Xiong, C. Yang, S. Zhang, Y. Chen, and S. Xu, “Energy-Efficient Wireless Communications:,” *IEEE Wireless Communications*, vol. 18, no. 6, pp. 28–35, December 2011.

[7] Q. Wang, M. Hempstead, and W. Yang, “A Realistic Power Consumption Model for Wireless Sensor Network Devices,” in *Sensor and Ad Hoc Communications and Networks, 2006. SECON ’06. 2006 3rd Annual IEEE Communications Society on*, vol. 1, September 2006, pp. 286–295.

[8] A. Håkansson, “Portal of research methods and methodologies for research projects and degree projects,” in *Proceedings of the International Conference on Frontiers in Education : Computer Science and Computer Engineering FECS’13.* CSREA Press U.S.A, 2013, pp. 67–73, qC 20131210.

[9] “Getting started with Thread Webinar,” Available online: https://threadgroup.org/portals/0/documents/resources/Webinar_NavigatingTheEcosystem.pdf, [Date accessed: 19th-October-2016].

[10] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," *Communications Surveys Tutorials, IEEE*, vol. 17, no. 4, pp. 2347–2376, 2015.

[11] "Thread Overview - White Papers," Available online: https://www.threadgroup.org/ourresources, [Date accessed: 12th-August-2016].

[12] "OpenThread: open-source implementation of the Thread networking protocol," Available online: https://github.com/openthread/openthread/wiki, [Date accessed: 9th-October-2016].

[13] Z. Shelby, K. Hartke, and C. Bormann, "The constrained application protocol (coap)," Internet Requests for Comments, RFC Editor, RFC 7252, June 2014, http://www.rfc-editor.org/rfc/rfc7252.txt. [Online]. Available: http://www.rfc-editor.org/rfc/rfc7252.txt

[14] R. Kelsey, "Mesh Link Establishment," Internet Engineering Task Force, Internet-Draft draft-kelsey-6lo-mesh-link-establishment-00, Jul. 2015, work in Progress. [Online]. Available: https://tools.ietf.org/html/draft-kelsey-6lo-mesh-link-establishment-00

[15] "Battery-Operated Devices - White Papers," Available online: https://www.threadgroup.org/Portals/0/documents/whitepapers/Thread%20Battery-Operated%20Devices%20white%20paper_v1_public.pdf, [Date accessed: 9th-October-2016].

[16] S. Ziegler, P. Kirstein, L. Ladid, A. Skarmeta, and A. Jara, "The Case for IPv6 as an Enabler of the Internet of Things," IEEE Internet of Things, 2016. [Online]. Available: http://iot.ieee.org/newsletter/july-2015/the-case-for-ipv6-as-an-enabler-of-the-internet-of-things.html

[17] Z. Pang, Y. Cheng, M. E. Johansson, and G. Bag, "Work-in-progress: Industry-friendly and native-IP wireless communications for building automation," in *Industrial Networks and Intelligent Systems (INISCom), 2015 1st International Conference on*, March 2015, pp. 163–167.

[18] G. Montenegro, C. Schumacher, and N. Kushalnagar, "IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals," IETF RFC 4919, Feb. 2016. [Online]. Available: https://rfc-editor.org/rfc/rfc4919.txt

[19] G. Montenegro, J. Hui, D. Culler, and N. Kushalnagar, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks," IETF RFC 4944, Oct. 2015. [Online]. Available: https://rfc-editor.org/rfc/rfc4944.txt

[20] "IEEE Standard for Information technology– Local and metropolitan area networks– Specific requirements– Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless

Personal Area Networks (WPANs)," *IEEE Std 802.15.4-2006 (Revision of IEEE Std 802.15.4-2003)*, pp. 1–320, Sept 2006.

[21] G. Anastasi, M. Conti, and M. D. Francesco, "A Comprehensive Analysis of the MAC Unreliability Problem in IEEE 802.15.4 Wireless Sensor Networks," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 1, pp. 52–65, Feb 2011.

[22] T. Kim, I. H. Kim, Y. Sun, and Z. Jin, "Physical Layer and Medium Access Control Design in Energy Efficient Sensor Networks: An Overview," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 1, pp. 2–15, Feb 2015.

[23] A. Wang, F. Lin, Z. Jin, and W. Xu, "A Configurable Energy-Efficient Compressed Sensing Architecture With Its Application on Body Sensor Networks," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 1, pp. 15–27, Feb 2016.

[24] B. Martinez, X. Vilajosana, I. Vilajosana, and M. Dohler, "Lean Sensing: Exploiting Contextual Information for Most Energy-Efficient Sensing," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 5, pp. 1156–1165, Oct 2015.

[25] J. Brusey, J. Kemp, E. Gaura, R. Wilkins, and M. Allen, "Energy profiling in practical sensor networks: Identifying hidden consumers," *IEEE Sensors Journal*, vol. PP, no. 99, pp. 1–1, 2016.

[26] J. Ren, Y. Zhang, K. Zhang, A. Liu, J. Chen, and X. S. Shen, "Lifetime and Energy Hole Evolution Analysis in Data-Gathering Wireless Sensor Networks," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 2, pp. 788–800, April 2016.

[27] B. Martinez, M. Montn, I. Vilajosana, and X. Vilajosana, "Early Scavenger Dimensioning in Wireless Industrial Monitoring Applications," *IEEE Internet of Things Journal*, vol. 3, no. 2, pp. 170–178, April 2016.

[28] B. Martinez, M. Montn, I. Vilajosana, and J. D. Prades, "The power of models: Modeling power consumption for iot devices," *IEEE Sensors Journal*, vol. 15, no. 10, pp. 5777–5789, Oct 2015.

[29] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, March 2000.

[30] Z. Tao, S. Panwar, D. Gu, and J. Zhang, "Performance Analysis and a Proposed Improvement for the IEEE 802.15.4 Contention Access Period," in *IEEE Wireless Communications and Networking Conference, 2006. WCNC 2006.*, vol. 4, April 2006, pp. 1811–1818.

[31] P. K. Sahoo and J.-P. Sheu, "Modeling IEEE 802.15.4 Based Wireless Sensor Network with Packet Retry Limits," in *Proceedings of the 5th ACM*

Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks, ser. PE-WASUN '08. New York, NY, USA: ACM, 2008, pp. 63–70. [Online]. Available: http://doi.acm.org/10.1145/1454609.1454624

[32] P. Park, P. Di Marco, C. Fischione, and K. H. Johansson, "Modeling and optimization of the IEEE 802.15.4 protocol for reliable and timely communications," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 3, pp. 550–564, 2013.

[33] P. Park, P. Di Marco, P. Soldati, C. Fischione, and K. H. Johansson, "A generalized markov chain model for effective analysis of slotted ieee 802.15. 4," in *2009 IEEE 6th International Conference on Mobile Adhoc and Sensor Systems*. IEEE, 2009, pp. 130–139.

[34] P. Park, C. Fischione, and K. H. Johansson, "Adaptive IEEE 802.15. 4 Medium Access Control Protocol for Control and Monitoring Applications," in *Wireless Networking Based Control*. Springer New York, 2011, pp. 271–300.

[35] "FRDM-KW24D512 Freedom Development Platform Users Guide," Available online: http://www.nxp.com/files/32bit/doc/user_guide/FRDMKW24DUG.pdf?fasp=1&WT_TYPE=Users%20Guides&WT_VENDOR=FREESCALE&WT_FILE_FORMAT=pdf&WT_ASSET=Documentation&fileExt=.pdf, [Date accessed: 3rd-November-2016].

TRITA EE 2016:187