Aalto University
School of Science
Degree Programme in Computer Science and Engineering

Xiaoyu Min

Experimental study on low power wireless sensor network protocols with native IP connectivity for building automation

Master's Thesis
Espoo, November 12, 2015

Supervisor:	Prof. Heikki Saikkonen, Aalto University
Advisors:	Dr. Zhibo Pang, ABB Corporate Research Dr. Vesa Hirvisalo, Aalto University

Aalto University
School of Science
Degree Programme in Computer Science and Engineering
Master's Programme in ICT Innovation

ABSTRACT OF
MASTER'S THESIS

Author:	Xiaoyu Min		
Title:	Experimental study on low power wireless sensor network protocols with native IP connectivity for building automation		
Date:	November 12, 2015	**Pages:**	vii + 50
Major:	Embedded Systems	**Code:**	T-106
Supervisor:	Prof. Heikki Saikkonen		
Advisors:	Dr. Zhibo Pang, ABB Corporate Research Dr. Vesa Hirvisalo, Aalto University		

The thesis performs an experimental study on the performance of 6LoWPAN network stack running on the latest TI CC2650EM, and evaluates the possibility of implementing the chip in industrial building automation system. The experiment and evaluation is done in three aspects that the industrial mostly concern – stability, latency and reliability, and power consumption.

Although there are already several mature network protocols specially designed for the IoT devices, IPv6 based 6LoWPAN network stack is highly compatible with the TCP/IP based Internet. The self-healing meshing and simple IP-routing mechanism makes 6LoWPAN very attractive to both developers and manufacturers.

Contiki operating system provides a clear layer-separated implementation for the 6LoWPAN stack, and introduces a radio duty cycling protocol for saving power. Therefore, a black-boxed performance test on Contiki OS with 6LoWPAN network stack running on the TI CC2650EM board.

Through the performance tests and comparison between the network protocols for low-power operations, the combination of TI CC2650 platform and Contiki OS with 6LoWPAN network stack is proved to be a promising solution for future building automation systems.

Keywords: Internet of Things, Wireless Sensor Networks, 6LoWPAN, Network Protocols, Contiki OS

Language: English

Acknowledgements

I would like to express my sincere gratitude to my research supervisors Zhibo Pang and Gargi Bag for offering this project opportunity, and for their excellent instruction and guidance. I would like to thank Shaoling Zhu and other colleagues at ABB Automation Solutions Group for all the assistance and help during the thesis period.

I am truly thankful to my supervisor, Professor Heikki Saikkonen and advisor, Doctor Vesa Hirvisalo, for taking the time to read my thesis and giving me a lot of useful advice on writing the thesis. And I would like to thank my parents and my girlfriend for providing support during the thesis period. Thank you!

Västerås, November 12, 2015

Xiaoyu Min

Abbreviations and Acronyms

6LoWPAN	IPv6 over Low Power Wireless Personal Area Networks
BAS	Building Automation Systems
BLE	Bluetooth Low Energy
CCA	Clear Channel Assessment
CDF	Cumulative Distribution Function
CoAP	Constrained Application Protocol
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
EM	Evaluation Module
ETX	Expected Transmission Count
ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IoT	Internet of Things
IPv6	Internet Protocol version 6
MCU	Micro-Controller Unit
MTU	Maximum Transmission Unit
NAT	Network Address Translation
OS	Operating Systems
PDF	Probability Density Function
PDR	Packet delivery Ratio
RDC	Radio Duty Cycling
REST	Representation State Transfer
RPL	IPv6 Routing Protocol for Low-Power and Lossy Networks
RTT	Round-Trip Time
SoC	System-on-Chip
TCP	Transmission Control Protocol
TI	Texas Instruments

UDP	User Datagram Protocol
WSN	Wireless Sensor Networks

Contents

Abbreviations and Acronyms **iv**

1 Introduction **1**
- 1.1 Problem statement 1
- 1.2 Contribution 2
- 1.3 Structure of the Thesis 2

2 Background **4**
- 2.1 Building Automation 4
- 2.2 Network Protocols for WSN 5
  - 2.2.1 Bluetooth Low Energy 5
  - 2.2.2 ZigBee 6
  - 2.2.3 6LoWPAN 8
- 2.3 Embedded Operating Systems 8
  - 2.3.1 TinyOS 9
  - 2.3.2 Contiki OS 10
- 2.4 Chipset for WSN 11
  - 2.4.1 Tmote Sky 12
  - 2.4.2 TI CC2538 12
  - 2.4.3 TI CC2650 13

3 General Test Plan **15**
- 3.1 Stability test 15
- 3.2 Latency and Reliability tests 16
- 3.3 Power Consumption tests 19
  - 3.3.1 Test Group Design 19

4 Experiment Setup **22**
- 4.1 Target Platform Introduction 22
  - 4.1.1 Hardware Platform 22
  - 4.1.2 Network Stack Platform 23

4.2 Stability Test Setup 25
4.3 Latency and Reliability Test Setup 26
4.3.1 Architecture Implementation 26
4.3.2 Test Node Deployment 29
4.4 Power Consumption Test Setup 31

5 Data Analysis 33
5.1 Stability Test Results 33
5.2 Latency and Reliability Test Results 36
5.3 Power Consumption Test Results 41

6 Conclusions 46

Chapter 1

Introduction

The thesis studies the most popular full-stack protocols in wireless sensor networks including Bluetooth low energy, ZigBee and 6LoWPAN (IPv6 over Low-power Wireless Personal Area Networks), and conducts a performance test on the 6LoWPAN protocol stack. Thanks to the rapid growth in the area of Internet of Things (IoT), many communication protocols targeting at low-power wireless sensor networks have been designed and implemented in the industrial world. Of the most popular WSN protocols, the IP-based 6LoWPAN protocol leads the trend of the future WSN protocol design.

The research focuses on the performance of 6LoWPAN protocol stack, which provides native IP support for the wireless sensor networks and proposes an energy-saving solution for the low-power based devices. The 6LoWPAN enables packet forwarding and self-healing mechanism on each node, and thus provides good support for large-scale wireless sensor networks.

1.1 Problem statement

Building automation (BA) is one of the most promising application area of the Internet-of-Things [17]. And wireless building automation system has become the new design paradigm of future building automation systems [15]. The wireless solution improves the user experience and reduces retrofit and maintenance costs. However, it is challenging to implement the wireless solution as the wired systems. Various aspects including message response latency, connection reliability, and power consumption, have to be taken into consideration. Therefore, evaluation for possible wireless solutions is needed before implementing the solution to industrial world.

Among the possible wireless solution, Bluetooth low energy has limited coverage range due to the lack of mesh networking support, and ZigBee is less developer friendly, since modifications on the ZigBee stack needs much effort. The 6LoWPAN absorbs the features of mesh networking support and low-power consumption, and provides IP support on resource and energy constrained devices. The support for Internet Protocol (IP) makes the 6LoWPAN protocol highly compatible with the existing TCP/IP based Ethernet or WiFi networks. With these attracting features, 6LoWPAN has become a candidate for the future building automation implementation.

The thesis tries to perform an experimental research on the 6LoWPAN protocol stack. The test results could be considered as a reference to evaluate the possibility of implementing the 6LoWPAN protocol stack into the industrial building automation design.

1.2 Contribution

The thesis contains constructive work that is needed in order to evaluate the performance of 6LoWPAN protocol stack in the perspective of stability, latency, reliability and power consumption. Firstly, a study on the project background is carried out. Papers related to building automation, WSN protocol stack and embedded operating systems are studied and reviewed. Secondly, a general test plan is carefully designed and some pre-tests are performed, in order to make preparations before the large-scale tests. Thirdly, stability test, latency and reliability test, and power consumption tests are performed one after another, with all the test nodes deployed in the building. Test data are collected and handled by Matlab. Fourthly, the collected data are turned into figures or other forms for analyzing, and the results are compared among each test group. Fifthly, after the test results are confirmed valid, the performance of the 6LoWPAN network stack is evaluated with the industrial standard to confirm the availability of implementing the stack into wireless building automation system design.

1.3 Structure of the Thesis

The thesis contains 7 parts. Chapter 1 covers the general introduction of the whole thesis project. Chapter 2 contains the necessary background within the scope of the thesis. Chapter 3 introduces the general test plan in the perspective of stability, latency and reliability and power consumption. Test group design is also included in this part. Chapter 4 introduces the setup of the experimental environment, and how the tests are performed. Chapter 5 shows the test results and data analysis. Finally, the conclusion of the project and future scope of the wholes thesis work is included in chapter 6.

Chapter 2

Background

This chapter introduces the relevant background information of the whole thesis project. A brief introduction of building automation is included in Section 2.1. Some of the most widely used network protocols for WSN are covered in Section 2.2. Section 2.3 and section 2.4 gives an overview of embedded operating systems and wireless sensor node chipsets.

2.1 Building Automation

Building automation is the centralized control system that is able to automatically adjust the conditions of the buildings according to the response of sensors or pre-set profiles. The Building Management System(BMS) or Building Automation System(BAS) usually includes the control of heating, ventilation, air conditioning(HVAC) and other systems[27][4]. The building automation systems are often designed and built as a distributed system. Compared to a non-controlled building, a well-designed building automation system brings more convinience to the administrators, and at the same time, saves energy consumption, reduces the cost of maintenance for the building.

Building automation system is the basic component of an intelligent building or 'smart building'. When installed in residential rooms or buildings, the concept of BAS and smart buildings could be extended to Home Automation(HA) and 'Smart homes'. However, the communication protocols and properties of the requirements are different between commercial industrial building automation systems and residential home automation systems. The former one relies on robust proven protocols while the latter one requires more purpose-specific protocols. Recent years, IEEE released standards on many physical networks ensuring the basic quality of service(QoS) and failover guarantees. Based on these standards-based foundations, protocol architects could implement different combinations for diverse purposes.

Traditionally, building automation systems are implemented with wired buses sending and receiving control signals or messages. These communications are turning to wireless nowadays, with the development of IoT and wireless sensor networks technology. The wireless devices are easier for deployment and maintenance. Furthermore, wireless communications provides a significant larger coverage than wired buses. Therefore, the building automation systems will be turning to wireless as a trend.

2.2 Network Protocols for WSN

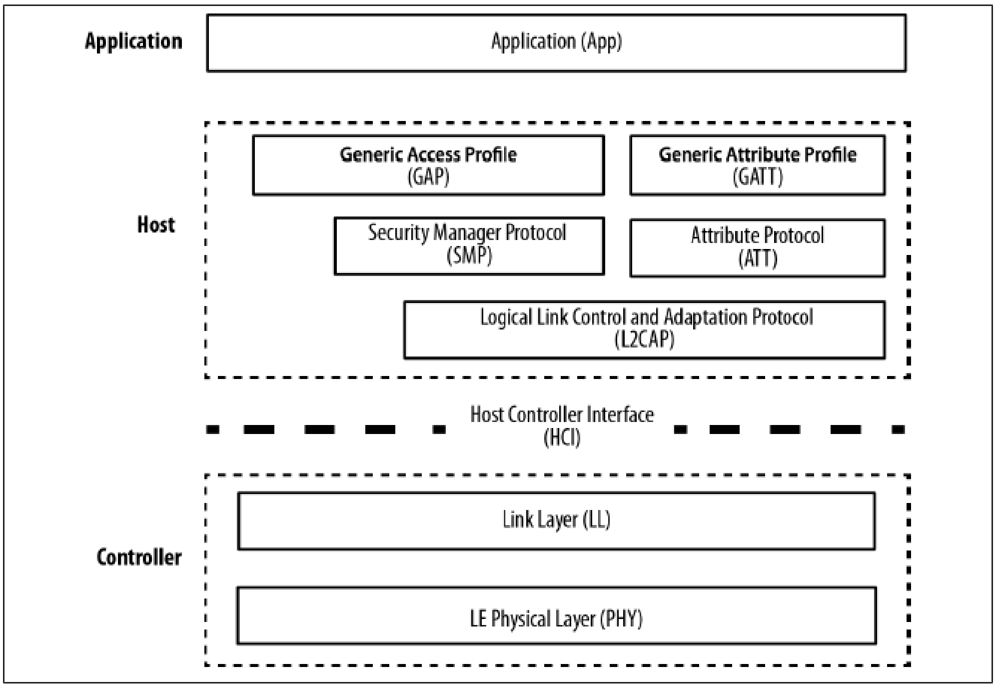
Hardware for wireless sensor networks are designed with scare energy resources. And the WSN are mostly used in low bandwidth and delay tolerant environments. Therefore, the communication protocols between WSN nodes should be carefully designed with specific purposes. Based on the IEEE standard, many different companies released their own WSN protocol stacks. Among these mature full-stack network protocols, Bluetooth Low Energy(BLE), ZigBee, and 6LoWPAN are the most successful ones. They are the most widely used protocol standards in the industrial world.

2.2.1 Bluetooth Low Energy

Bluetooth Low Energy, or Bluetooth Smart is first proposed by Nokia as a light-weight subset of the classic Bluetooth. It has been merged to the main Bluetooth standard from Bluetooth version 4.0. Bluetooth LE is designed as a low-power solution for control and monitoring applications, aiming at novel applications in the health-care, fitness, beacons, security and home entertainment industries [26][9].

The BLE devices operate in the industrial radio band 2.4GHz, covering 40 2-MHz channels instead of the 79 1-MHz channels for the classic Bluetooth. BLE has a data transmitting rate of 1 Mbit/s, and supports only scattered network topology. There must be only one master device, communicating with multiple slave devices within the communication range. The slave devices are not able to route packets or messages to further devices [3]. Therefore, the BLE network is more likely to be a combination of peer to peer network instead of a meshing network.

Figure 2.1 shows the network stack of BLE. BLE defines its own MAC, network, transport and encryption layer, leaving only the application layer configurable for the developers. Due to the maximum linkable device limitation, the BLE network is not able to hold much devices. However, the light-weighted protocol is the most commonly supported protocol in the mobile world. Almost all the popular mobile platforms including iOS, Android and Windows Phone provides original BLE supports. And many wearable devices choose BLE because of the low-power specification and the popularity in different platforms.



The BLE protocol stack

Figure 2.1: Bluetooth Low Energy protocol stack [24]

2.2.2 ZigBee

ZigBee is a set of specifications based on IEEE 802.15.4 standards for wireless personal area networks[28]. The standard is created to address the need for a cost-effective, standards-based wireless networking solutions that supports low data rates, low power consumption, security and reliability[23]. ZigBee allows the low-power devices to sleep in most of the inactive periods, which makes them very power-efficient. ZigBee is the most popular meshing network standard in the industrial world by far, after more than 10 years of development.

ZigBee devices are defined to work on the industrial radio band 2.4GHz, with data transmitting rates up to 250 kbit/s. The ZigBee network layer provides native supports for star networking topology, tree networking topology and generic mesh networking [2]. The ZigBee wireless sensors network must have at least one coordinate device act as the central node in the network, creating and maintaining the network. Other nodes are allowed to route packets, and thus the network is highly extendable. The flexibility on network topology is also one of the most important feature and advantage for ZigBee against other competitors.

The ZigBee protocol stack is shown on figure 2.2. Both the physical layer (PHY) and part of the MAC layer are compatible with the IEEE 802.15.4 standards. Network layer and application framework are defined by ZigBee itself, forming a ZigBee compliant platform. The upper layer including the application layer could be defined and modified by the users or developers for specific purposes. Early ZigBee does not support IP-based routing, but the ZigBee Alliance has already added some IP specifications in the latest version of ZigBee standards [29].

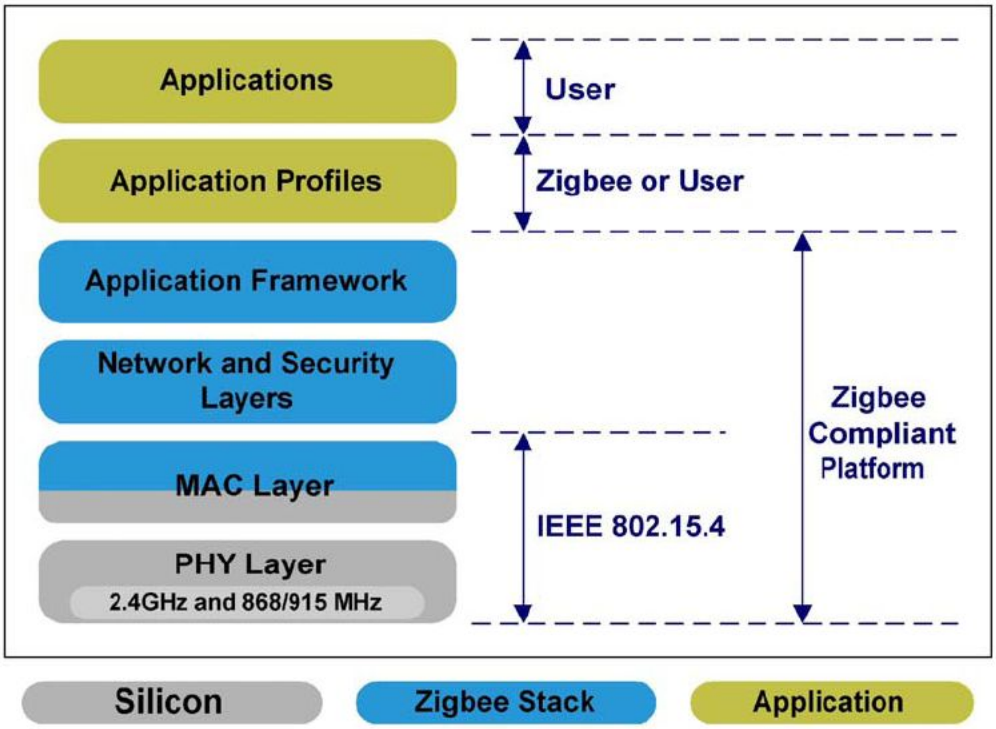


Figure 2.2: ZigBee protocol stack

2.2.3 6LoWPAN

6LoWPAN, with the full name IPv6 over Low-power Wireless Personal Area Networks, is an open standard based on IEEE 802.15.4 [12]. Also one of the working groups in the Internet Engineering Task Force (IETF) is named after 6LoWPAN, working on the standardization and development of 6LoWPAN [25].

The greatest achievement for 6LoWPAN protocol stack is that it enables the possibility to send and receive IP-based packets over low-power IEEE 802.15.4 based networks [14]. Traditionally, the Internet Protocol (IP) stack requires large memory usage and high network bandwidth, which is not applicable with energy constrained devices. Through header compression and encapsulation mechanism for IPv6 addresses, the 6LoWPAN working group managed to adapt the IP-based packets for IEEE 802.15.4 devices [16]. 6LoWPAN tries to compress the original IPv6 packets, using compressed IPv6 Internet Control Message Protocol (ICMP) packet format, and introduces optimized neighbor discovery policies [21].

Figure 2.3 shows the comparison of network stacks between 6LoWPAN and traditional TCP/IP stack. Unlike Bluetooth low energy protocol stack and ZigBee stack, 6LoWPAN defines only an adaption layer between data link layer (MAC layer) and network layer. The developers are able to adapt any other layers for specific requirements. For example, the 6LoWPAN network stack supports multiple light-weighted application protocols including Constrained Application Protocol (CoAP), MQTT and so on. Another big advantage for 6LoWPAN is that the adapted IP-based protocol is highly compatible with the widely used TCP/IP network. The packets sent between 6LoWPAN and TCP/IP networks could be easily transformed by the edge router with very low cost.

2.3 Embedded Operating Systems

With constrained hardware resource and concerns on energy consumption, the operating systems running on the WSN devices shall also be tailored to be light-weight enough for various hardware models. On the other hand, they shall still support as much protocols as possible in order to fulfill different requirements. There are already many mature operating systems designed for WSN devices supporting multiple protocols. For example, TinyOS, ContikiOS, RIOT-OS, openWSN and so on. They support most of the popular IETF protocols such as 6LoWPAN, RPL, CoAP, UDP etc. Of these operating systems for WSN, TinyOS is one of the most mature one. ContikiOS

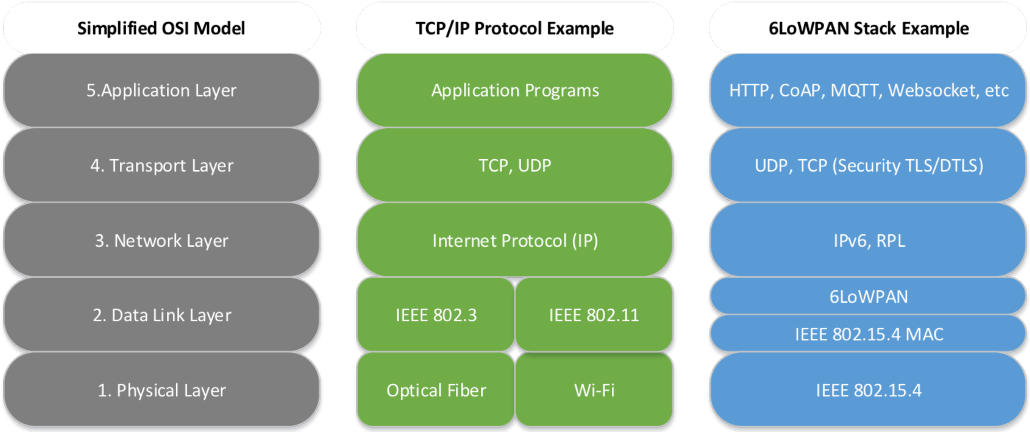


Figure 2.3: Comparison between 6LoWPAN stack and TCP/IP stack

is the representative of the newly designed ones with good cooperation with the industrial world.

2.3.1 TinyOS

TinyOS is an open-sourced, component-based operating system first developed by University of California in Berkeley [10]. The community of TinyOS development has grown to thousands of developers since its first release in 2000. The development and maintenance is now performed by TinyOS Alliance.

TinyOS uses nesC as the official programming language. NesC is a dialect of C programming language, optimized for memory constrained devices [7]. TinyOS programs are built with components. All of the events, tasks, and interfaces are considered as computational abstractions of components. There is a set of basic components defined by TinyOS. These components are connected with each other through interfaces. Tasks are usually posted to the system scheduler for execution without interrupting the normal system work, since TinyOS is a non-blocking operating system.

In terms of WSN network protocol support, TinyOS is also one of the earliest supports of 6LoWPAN. It provides a full stack implementation for 6LoWPAN, as shown in figure 2.4. The BLIP component is the 6LoWPAN implementation in TinyOS, and TinyRPL is the implementation of RPL routing protocol. [11] performed a set of experiments evaluating the RPL routing protocol and 6LoWPAN on TinyOS. The results show that TinyOS provides an efficient routing solution in memory constrained low-power WSN devices.

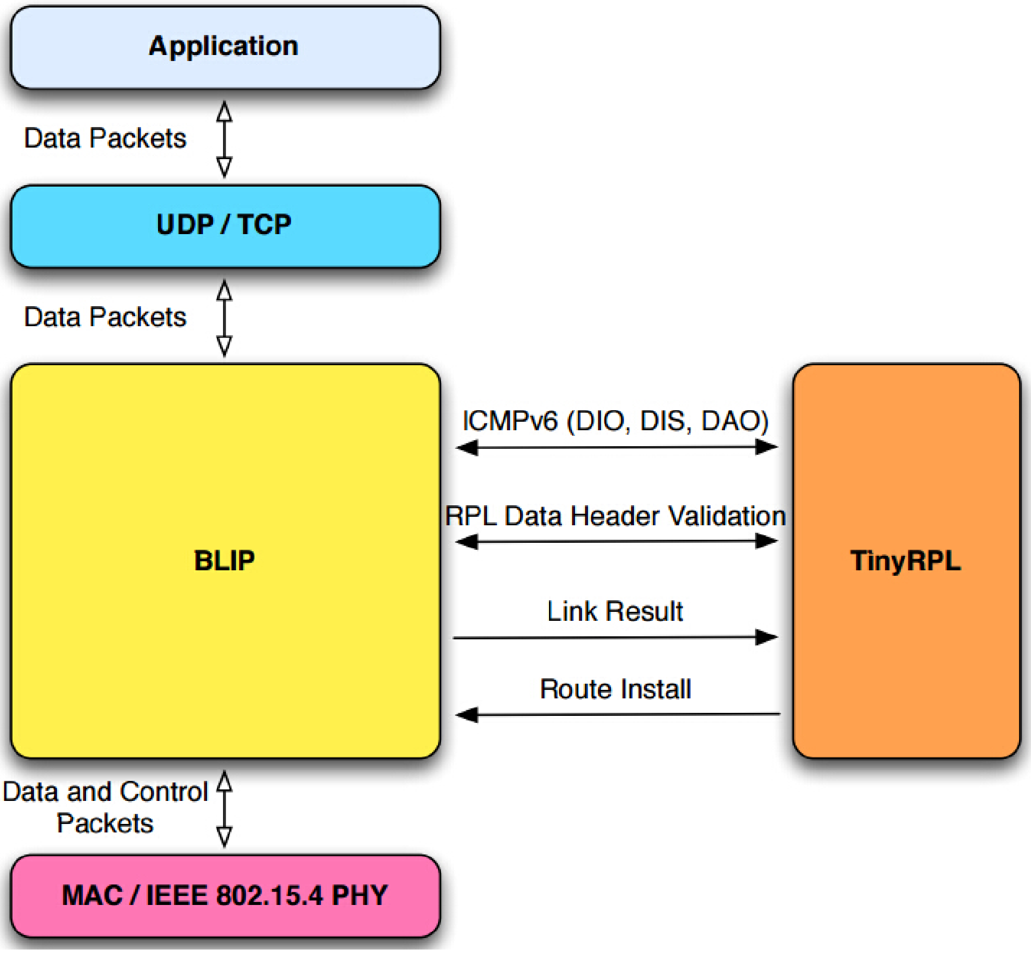


Figure 2.4: TinyOS 6LoWPAN/RPL protocol stack

2.3.2 Contiki OS

Contiki operating system is first created by Adam Dunkels in 2002, and it is now maintained by the Swedish Institute of Computer Science (SICS) in Sweden. The Contiki community is one of the largest and most active IoT communities now. Supported by Texas Instrument (TI), Atmel, Sensinode, Cisco and many other companies and organizations, Contiki has just released the latest stable version Contiki 3.0 on August this year.

The Contiki OS is designed particularly for low-power wireless IoT devices with constrained memory and resources. The minimum memory required for a complete IP-supported Contiki OS could be less than 10 kilobytes, with less than 30 kilobytes' ROM required [5]. Contiki provides a light-weight programming model based on protothreads, achieving low memory overhead of each process. Protothreads absorbs the features of both multi-threading and event-driven programming [5]. Contiki manages a real-time clock and an event clock. System level operation and low layer of network operation is scheduled and triggered by the real-time clock. Event clock, on the other hand, serves the upper layer processes and application defined processes that do not require high accuracy.

Besides multi-tasking, Contiki provides full stack support for different networking mechanisms, including uIP-based TCP/IP stack, Rime stack, and the uIPv6 stack. The Contiki uIPv6 stack was the smallest IPv6 stack to receive the IPv6 Ready certification when it was first released in 2008. Figure 2.5 presents the implementation of uIPv6 stack in Contiki OS. The code structure in Contiki OS is correspondent to the protocol stack layers. Contiki inserted an additional radio duty cycling (RDC) layer between radio layer and MAC layer. With RDC layer enabled, the device is able to shutdown the radio chip in idle time and wake it up when active, which reduces power consumption and extends battery life [6].

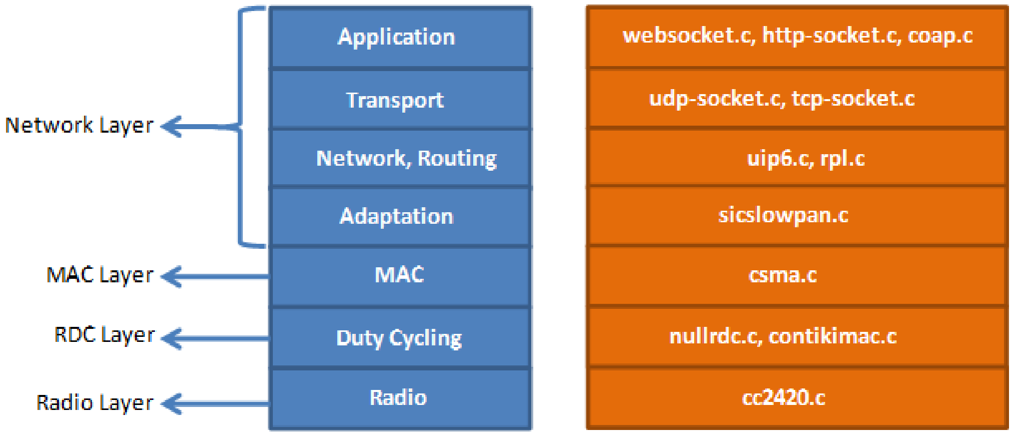


Figure 2.5: The Contiki uIPv6 protocol stack

2.4 Chipset for WSN

Currently, the trend towards IoT in both building automation and wearable devices is inspiring the development of different chipsets from different vendors. The chipsets are able to achieve ultra-low power consumption thanks to the development of both low-power wireless sensors and power-saving protocols. Of the various chipsets, Tmote Sky has been the sample hardware for TinyOS, while TI CC-series WSN chips have very good support for the ZigBee stack and 6LoWPAN stack.

2.4.1 Tmote Sky

Tmote Sky, also named Telos B, is an ultra low-power wireless sensor module used in wireless sensor networks. The device integrates a TI MSP430 MCU and TI CC2420 radio chip. Tmote Sky adopts many industrial standards such as IEEE 802.15.4 and USB for communication with other devices wirelessly or through serial line. Furthermore, Tmote Sky provides developing support for thousand of mesh networking applications by integrating a series of sensors and peripherals such as temperature and humidity sensors. Tmote Sky has passed rigorous tests and is supported officially by TinyOS, Contiki OS, and many other open-source IoT embedded OS. It is a smart module with the features of robustness and lightweight. Figure 2.6 shows the model of Tmote Sky.

Figure 2.6: Tmote Sky Module

2.4.2 TI CC2538

TI CC2538 is a power wireless MCU System-On-Chip (SoC) for high performance IoT applications. Figure 2.7 shows the small CC2538 evaluation module. The chip combine an ARM Cortex-M3 based MCU, providing up to 32KB on-chip RAM, and up to 512KB on-chip flash together with an IEEE

802.15.4 radio. The tiny shaped chip is able to run the most up-to-date network stacks with high-level security and robustness applications. The 32 GPIO ports and serial peripherals enables the connection between the chip and TI evaluation board. There is also a micro-USB port on the board, which could be connected to external power source. The SoC allows efficient authentication and encryption process, while minimizing the workload for the MCU. Furthermore, three sets of low-power modes with retention enables the quick sleep and recharge for periodic tasks, leveraging the performance and power consumption. TI has provided a comprehensive driver library and a series of debugging tools, which guarantees the smooth development of CC2538. The chip is also equipped with state of the art IoT technologies and solutions such as ZigBee and 6LoWPAN.

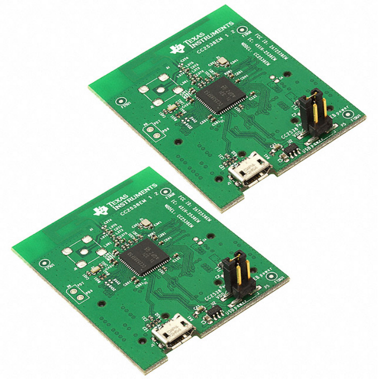


Figure 2.7: Texas Instrument CC2538 Evaluation Module

2.4.3 TI CC2650

TI CC2650, as shown in figure 2.8, is the latest released WSN chip in TI CC-series low-power chips. It is designed for the next-generation IoT solutions. The chip supports not only ZigBee and 6LoWPAN network stack, but also the BLE protocol stack. TI CC2650 belongs to the CC26xx chip family, targeting at cost-effective, ultra-low power and 2.4 GHz devices. Compared to CC2538, CC2650 provides a better low-power management and longer battery lifetime, minimizing the current consumption of RF and MCU. Even coin cell batteries can support the device running energy-harvesting applications for up to years of lifetime. Similar to CC2538, CC2650 also contains a 32-bit ARM Cortex-M3 based MCU and other ideal peripherals. The SoC integrate an ultra-low power sensor controller for data collection even when the MCU is in sleep mode. The CC2650 targets at application domains within industrial, consumer electronics, medical and many other areas. The BLE controller and the IEEE 802.15.4 MAC are embedded into the board and are partly running on a separate ARM Cortex-M0 processor. This architecture improves the overall system performance, decreases the power consumption, and frees up flash memory for the application.

Figure 2.8: Texas Instrument CC2650 Evaluation Module

Chapter 3

General Test Plan

This chapter concentrates on the general test plan for the evaluation of 6LoWPAN network protocol. As mentioned in Section 2.1, reliability of the networks, latency of the message communication, and power consumption are the most intuitive properties that the system architecture and end users concern. Thus, the thesis tried to design a black-boxed test plan, evaluating the performance in these three aspects.

3.1 Stability test

To evaluate the possibility of introducing a working protocol into industrial world, the stability is the priority concern. There are many factors that may affect the stability performance of the protocol. For example, the power of the RF chip, the interference from other electronic devices, and even the structure of the building may influence the performance. Therefore, the stability test plan shall be carefully designed in order to take as much considerations as possible.

First of all, the standard of stability could be simplified to the maximum transmit range of two devices, or single-hop coverage. The single-hop coverage test result could be of great importance in the evaluation process. It will expose the limitations of the simplest wireless sensor networks established by the minimum of two nodes. And further test node deployment plan could be designed based on the test results, which could make the test plan more rigorous and scientific.

Secondly, different test groups shall be designed. There are three major factors that may affect the single-hop coverage – the hardware itself, the software network stack running, and the building structure. Therefore, the test group and comparison groups division could be based on these factors.

For the hardware, they could have different RF power settings, and the built-in or external antenna based on the hardware design. The laboratory has only two hardware models that is able to perform the test, and both of them are equipped with built-in antenna. Thus the test group shall be directly based on the target chip TI CC2650. According to the official document from TI, the RF output power on the chip CC2650 is adjustable, ranging from -3dbm to +5dbm. Thus, the comparison groups could be the same chip with different RF power settings.

Section 2.2 introduces the most popular WSN network protocols. Fortunately, TI CC2650 supports multi-network stack including ZigBee, Bluetooth LE, and 6LoWPAN. Since our target protocol is 6LoWPAN, our test group will definitely running 6LoWPAN, and the comparison groups will be running ZigBee and Bluetooth Low Energy stack. The single-hop performance may also vary between these protocols.

The structure of the building is somewhat subjective since there may be a lot of different in-door designs. However, the test may concern only the simplest models that may affect the wireless communications. One of the major advantage of connecting nodes wirelessly is that they are able to connect with each other through the air, regardless of the walls and floors between them. Thus, the test considers only the effects that the walls on the same floor or different floors brings to the communications.

	HW(Power Conf.)	SW(Contiki)	ENV(floors)
HW(Power Conf.)		x	x
SW(BLE/ZigBee)	x		x
ENV(Walls)	x	x	

Table 3.1: Stability Test Groups Design

3.2 Latency and Reliability tests

While stability test aims at confirming the basic functionality and availability of the combination of hardware (CC2650) and software (6LoWPAN), the evaluation needs to perform a further performance test on the situation where there are many nodes deployed in the building. Latency and reliability are the most important quality standards that are used to judge whether the protocol is applicable in industrial environment.

Latency and reliability are also the most straightforward indicators that may affect the performance of the product, in terms of the end user experience. If the product running the 6LoWPAN protocol act with high latency, user may feel a significant delay, which will bring bad user experience. Similarly, low reliability may increase the cost of node deployment, and may not meet the need of some special circumstances that require high reliability. However, it maybe hard to obtain both promising latency and reliability and the cost of constructing the WSN at the same time. With certain hardware and software combination, the test intends to evaluate the latency and reliability of the whole wireless sensor network, and compare the result with some industrial standards. The protocol is applicable in the industrial environment if the results meet the need, otherwise, some improvements need to be taken before it is introduced to the product production departments.

There are several papers and studies which performed some QoS-based tests on 6LoWPAN and RPL networks in various occasions. [13] proposed a 6LoWPAN performance test via Contiki Cooja simulator. [1] studied the memory usage, network quality and power consumption of 6LoWPAN network based on the hardware TI CC2530EM. [19] focused on the network quality research in TinyOS running Constrained Application Protocol(CoAP). However, most of the evaluation are performed in simulators. Rarely did the researchers test the performance in real experimental environments. Therefore, the thesis tried to release an experimental test to determine whether CC2650EM running 6LoWPAN network stack fulfills the requirement of low power wireless sensor networks in the industry field.

The latency defined in the WSN could be described as the response time needed from the request is sent to the target node, to the target node finally makes the response. More exactly, the latency could be measured by the round-trip time of the requesting packet. The round-trip time includes the time taken from the source node to the destination, time for the target node handling the request, and the time taken from the target node to the source node (acknowledgement packet).

Reliability of the whole wireless sensor network system is more or less related with the latency of the network. Different scenarios require different acceptance deadlines for the requests. For example, rule-of-thumb of soft real-time for non-critical control applications suggests a maximum RTT of 150ms, while more critical cases, such as control loops in future sustainable buildings, require bounded latency and guaranteed availability to manage the power generation, storage and consumption [18]. To determine the reliability of the WSN system, several soft real-time deadlines could be set. The WSN system could be recognized as reliable, if the request could be handled within the bounded deadline.

Figure 3.1 shows the abstract network topology of the 6LoWPAN wireless sensor network system. There will be traffic between border-router and other nodes. And the nodes will be able to communicate with each other at the same time. There may be network congestion in the wireless network, and therefore brings longer average response time and lower reliability.

Based on the network traffic environment, the latency and reliability test could be done in the following test cases:

- There are only slight traffic between RPL border-router and target node.
- There are only slight traffic while extra traffic exists between test nodes
- There are heavy traffic between RPL border-router and the target node, no extra traffic
- There are heavy traffic between border-router and the test node, while interference traffic exists between the nodes.

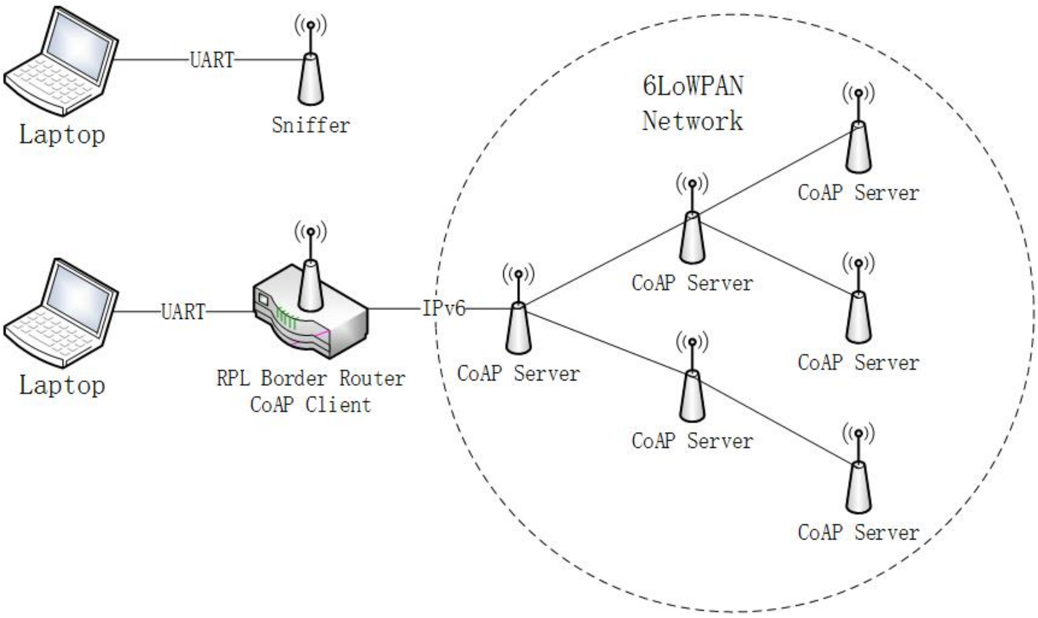


Figure 3.1: The abstract topology of the whole working system

3.3 Power Consumption tests

Power consumption is another property that researchers and manufacturers are very concerned about. With very limited resource and computing power on the MCU, most of the wireless sensor node devices are designed to achieve a long network lifetime. A combination of low power hardware and power-saving software algorithms is needed.

Texas Instrument released its latest CC2650 multi-standard wireless MCU early February this year. One of the major features of the chip is the ultra-low power hardware design. According to the technical document, the active current could be as low as 9.1mA when the chip is transmitting in the maximum power (+5dBm). That is a significant improvement compared to its predecessor's 20mA on CC2538. Since the radio transmitter is the major power consumer on the wireless sensor nodes, the decrease in power consumption for RF-chip could have great impact on the battery life of the whole device.

The sensor node shall keep the radio transmitter off as much as possible, and drop to some low power mode to save the battery life. However, the node may miss some packets if some nodes are sending messages to it when it is still sleeping in low power mode. The node is not able to send or receive any packets when it is turned off.

If the test nodes could be time-synchronized, they could schedule the wake-up and sleep period together, so that there will not be any message missed due to the inconsistency. However, the low-power wireless sensor network is a loosely-established network. The network topology may change quite frequently, since it employs a self-healing meshing mechanism. The cost of synchronization between different WSN nodes is too high for the low-power oriented devices to afford.

Contiki operating system introduces a power-saving duty cycling protocol on the MAC layer and moves it to a new layer above the MAC layer, called the RDC layer. If a device is running ContikiMAC over normal 802.15.4 MAC layer, it will periodically activate the RF radio and check if the channel listening is busy. If there are messages in the channel, the radio will be kept on until it receives the message and quickly turn to sleep again. On the other hand, the sending node will also re-transmit the message to send several times before it receives the response from the target.

3.3.1 Test Group Design

The major objective of the power consumption test is to estimate the battery life of the embedded devices and try to make some optimizations to the network protocols. Additionally, a trade-off could be found between the power consumption and the performance of the devices if power consumed by the devices could be precisely measured.

Since most of the smart devices working in the building automation or smart homes are scenario-based, the power consumption test should also be carried out according to their working environment.

The node devices in the IEEE 802.15.4 6LoWPAN wireless sensor networks are connected with each other automatically, establishing a multi-hop network. Therefore, the nodes in the network could be divided into three groups. Firstly, the node could be running as a border router in the network, and there are only one border router act as root router in the network topology. The border router should always have stable power supply, thus the power consumption test does not need to take much attention on the border router. For other node devices in the network, they could act either as an end node receiving request from other nodes or border router and transmitting response to the sender, or as a forwarding node forwarding the packets to their destination.

[6] states that the ContikiMAC RDC mechanism could reduce the power consumption of the device for up to 80%. The sender-initiated asynchronous mechanism keeps the radio off for roughly 99% according to the network congestion. Therefore, the test also would like to confirm whether the power saving mechanism work as described in the paper, and how the performance would be influenced when using ContikiMAC RDC.

		Border Router	**Forward Node**	**End Node**
Radio Off (Idle)				x
Device configured with **NullRDC**	Idle Listening			x
	Transmitting		x	x
	Receiving		x	x
Device configured with **ContikiMAC**	Idle Listening			x
	Transmitting		x	x
	Receiving		x	x

Table 3.2: Power Consumption Test Groups

Based on the RDC configuration and roles a node plays in the WSN network, the test group could be divided as the table 3.2. Firstly, the test would like to measure the energy cost when the radio module is off, or the node is in idle mode. There are two major test sample groups when the RF module is active, with different RDC configurations. The first group is the default NullRDC sample group, where the nodes would keep the RF chip always on, waiting for incoming messages. The second group is configured with the power-saving ContikiMAC RDC, where the RF chip will be waked up periodically according to the occupancy of the wireless channel.

For each test groups in the table, the test shall try to estimate the energy consumption of each state, and modularize the whole process. Partial network optimizations could have been performed with an estimation of energy cost. Additionally, the modularization will be of grate benefits for evaluation of battery life and the performance of both hardware device and software efficiency.

Chapter 4

Experiment Setup

The experimental setup according to the test plan in the previous chapter is demonstrated in this chapter. Detailed parameters of the target hardware and network stack are listed in Section 4.1. Section 4.3 and Section 4.4 describes the setup of the test environments for latency, reliability and power consumption.

4.1 Target Platform Introduction

4.1.1 Hardware Platform

The major test hardware chosen for the test is TI CC2650EM. As introduced in Section 2.4, the newly released board is equipped with ARM Cortex M3 SoC, supporting multiple network stacks based on 802.15.4 low-power radio module. The board is officially supported by both Texas Instrument and Contiki open-source community for the ContikiOS-based 6LoWPAN network stack. Most of the performance tests are carried out on the CC2650 board.

TI CC2538EM acts as the comparison group in the test. It is the previous generation of the TI CC-series chip-set, supporting both ZigBee stack and 6LoWPAN stack. It does not support Bluetooth LE, and the power consumption is higher than the later one, according to the user manuals. Since CC2538EM is a more mature chip, the test tried to establish a complete test bench on the old chip first, then ported the code to the new chip.

Both of the CC2650EM and CC2538EM are compatible with the TI SmartRF Evaluation Board. The evaluation board extends various I/O ports on the development board, including UART, I2C, GPIO ports etc. And it includes an independent debugging circuit and a current measuring circuit, which provides a convenient way for performance evaluation. There

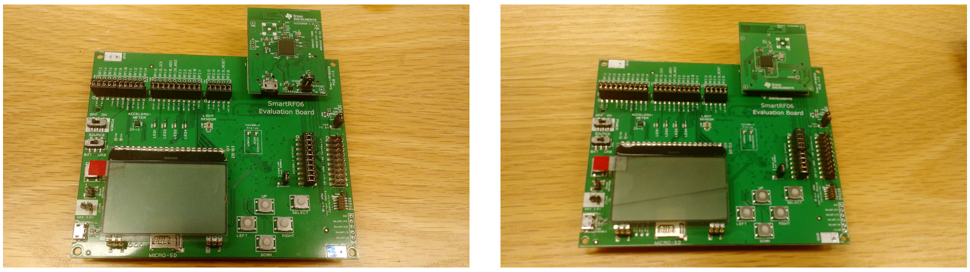

(a) TI CC2538 with Evaluation Board (b) TI CC2650 with Evaluation Board

Figure 4.1: Test hardware platform

is a micro-USB power supply port on the CC2538EM board, therefore, the board could operate independently, without being connected to the evaluation board. However, CC2650EM removed that port with updated hardware design. As a result, the CC2650EM have to work with connection to the SmartRF evaluation board. Both power supply and firmware update are done through the evaluation board. All the I/O and power supply ports are directly extended on the evaluation boards when attached to the larger mother board.

4.1.2 Network Stack Platform

Table 4.1 shows the Contiki network stack within the scope of the thesis. It is layer-based and well-structured thanks to the highly modularized Contiki operating system. All of the 6LoWPAN network stack implementation in the test are built on the Contiki OS, which is highly portable to other hardware models, and easy to maintain.

Layer	Protocol
Application	Erbium-CoAP
Transport	UDP
Network	IPv6/ContikiRPL
Adaptation	6LoWPAN
MAC	CSMA/CA
RDC	NullMAC/ContikiMAC
Physical	IEEE 802.15.4 PHY

Table 4.1: Network stack in the test

The application protocol chosen for the test is Erbium-CoAP. Erbium (Er) is a low-power REST Engine designed for Contiki. It is the official CoAP implementation engine on Contiki OS. Erbium CoAP follows the guidelines of RFC 7252 with features of blockwise transfers and observing[20]. CoAP adopts many features and patterns from HTTP, such as URIs and resource abstractions. These HTTP-like features makes it highly compatible with the most widely used world wide web.

UDP transport protocol is preferred in resource constrained network environment, since energy consumption weighs more than packet losses or message delay in most cases. TCP transport protocol consumes more energy, though higher reliability is achieved. The reliability of the communication is enhanced by both the CoAP implementation in the application layer, and MAC layer. Both of the MAC layer and application layer introduces the re-transmission mechanism. Once a packet is sent from source node, it will re-transmit the packet certain times before receiving a kind of acknowledgment message(ACK) from the target node. The packet will be considered as lost packet if no ACK is received after several times of re-transmission.

For the network layer, uIP-based IPv6 implementation with Contiki RPL protocol is the default configuration in the Contiki 6LoWPAN network stack. The Contiki RPL provides a low-cost but efficient routing strategy based on ranks, and it has been accepted by the IPSO Alliance as a standard implementation. The core part of the Contiki RPL are the two objective functions, which plays an important role when establishing the wireless sensor networks. One of the objective function is hop-based OF0 [22]. The other is MRHOF, which is based on Expected Transmission Count(ETX) [8].

The RDC layer is more kind of a power saving mechanism implemented between physical layer and MAC layer. There are multiple RDC drivers supported by Contiki OS, including X-MAC, CX-MAC, LPP NullMAC and ContikiMAC. ContikiMAC is the most power efficient RDC solutions among these implementations, while NullMAC brings the highest reliability but highest energy consumption at the same time [6]. Figure 4.2 shows the basic transmit principle of ContikiMAC. The ContikiMAC driver will try to keep the radio off in the transmit intervals, and will keep transmitting data packets in each duty cycle until ACK message is received. Both the MCU and RF chip are kept sleep most of the time, since the radio cycle period is large. And the power consumption could drop to a significant low level, compared to the always-on mechanism in NullMAC driver. Both NullMAC and ContikiMAC driver are tested in the power consumption test, in order to confirm the significant power save when using ContikiMAC.

Carrier sense multiple access with collision avoidance (CSMA/CA) protocol is the default MAC option in Contiki OS, in order to avoid packet collision.

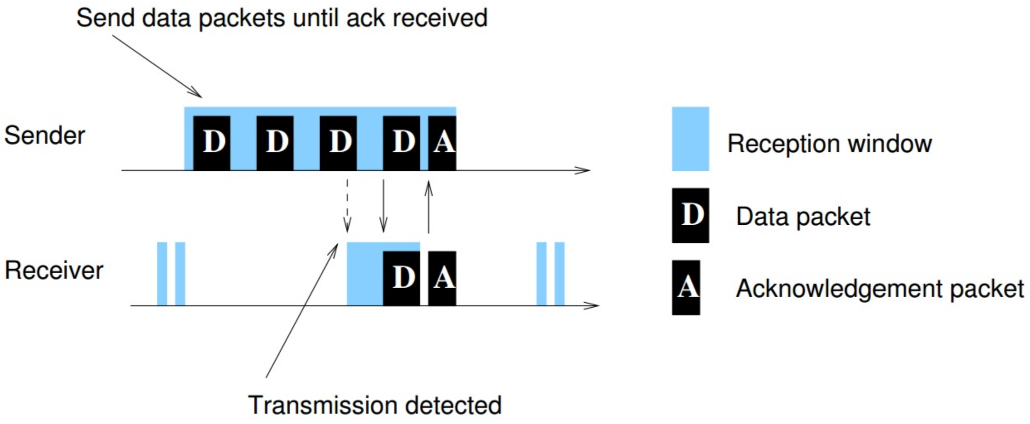


Figure 4.2: A ContikiMAC unicast transmission [6]

On the physical layer, IEEE standard 802.15.4 PHY specially designed for low-power devices is implemented and supported by both the hardware and software. The 2.4 GHz frequency band is accepted worldwide as a standard.

4.2 Stability Test Setup

According to Section 3.1, the test shall perform a single-hop coverage test in test groups with different power configuration, software configuration and indoor environments. Based on the hardware and software availability, the test is performed with the following 4 test cases:

- CC2650 BLE module is used to send bacon and Google Nexus 5 (with standard BLE module) is used to receive bacon
- CC2650 BLE module is used both as transmitter and receiver
- CC2650 802.15.4 module is used both as transmitter and receiver (ZigBee)
- CC2650 as both transmitter and receiver, one of them is border-router and the other is test node (6LoWPAN)

These test cases are evaluated under different transmission power settings of 3dBm, -3dBm and -9dBm. The sender is located at the same place, while the receiver is kept moving in different floors until the round-trip time (RTT) goes to an abnormal value or packets are not able to reach the destination.

Figure 4.3 shows how the stability test is performed. The border-router or the sender node is placed in the red icon inside the office. It will keep sending bacon or packets continuously to the target node. One of the tester will hold a working receiver walk along the corridor on different floors. The other one will be monitoring the bacon or packet sending results at the sender side. The RTT result will be printed on either the sender side or the receiver side. The test will terminate once the RTT reaches a significant large value or even when the connection is lost. Then the place where the connection is hold at last will be considered as the maximum transmit range with reliable transmissions between the sender and the receiver.

Figure 4.3: Reliability test setup

4.3 Latency and Reliability Test Setup

This is the most important part in the test. A complete wireless sensor network needs to be established, covering the whole building. The experimental test aims at simulating the real working environment, and evaluate the whole system performance in terms of the latency and reliability. It is a combination of qualitative and quantitative test. Round-trip time (RTT) and packet delivery rate (PDR) of different number of hops and different traffic loads are recorded and analyzed.

4.3.1 Architecture Implementation

The abstract topology of the latency and reliability test is shown in Figure 4.4. One of the TI CC2538 board will be connected with a laptop and act as a sniffer. The sniffer will be placed near the border-router, sniffing the packets sent to or received by the border-router. Since all the 6LoWPAN network will have to go through the border-router before they reach the outer network, the sniffer could have been a good debugging tool. One of the TI CC2650 boards will act as an RPL border-router, and a CoAP Client as well. All the test packets will be initiated by the border-router and then sent to the target node. The border-router is connected to a laptop through UART-USB port, and print the RTT time and other node information on the laptop's terminal.

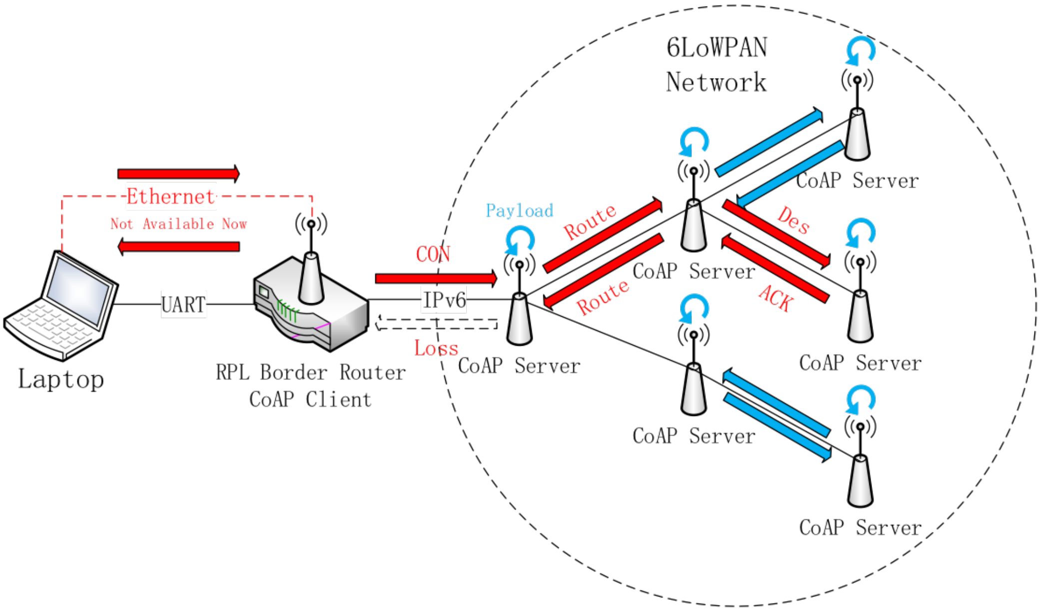


Figure 4.4: Abstract topology of the RTT/PDR test

After several pre-experimental tests, detailed experimental parameters are set according to the pre-test results. These parameters are shown in table 4.2. The Contiki version is kept the latest beta version with version number 3.x. The stable version Contiki 2.7 does not support the CC2650 and lack many new functionality. Hardware platform is introduced in section 4.1, with TI CC2650 and SmartRF06 Evaluation Board as target platform, and TI CC2538 as border-router. The RDC layer is switched off as NullRDC at the first stage of evaluation. There are a total number of 20 CC2650 boards, one of them act as a border-router, the rest are test nodes. Transmit power is set to the maximum available value +5dBm, in order to have the maximum coverage range. The communication channel chosen for the test is channel 26, because severe interference are observed when set to lower channels. These might be related to WiFi signals in the air. And that is why Bluetooh communications are set to channel 25 or channel 26 as well.

Parameter	Value
Contiki Version	3.x
Hardware Platform	TI CC2650 with SmartRF06EB
Radio Duty Cycling	NullRDC
Communication Channel	26
Transmit Power	+5 dBm
Total number of Nodes	1 BR + 19 Nodes
BR CoAP GET packet size	53 Bytes
Node CoAP GET packet size	70 Bytes
Node CoAP ACK packet size	63 Bytes

Table 4.2: Experiment parameters

When the test starts, the RPL border-router will try to communicate to the neighboring nodes first, and quickly establish the 6LoWPAN network according to the RPL information from the nodes. The other nodes will then automatically join the network, since they have the same PANID, communication channel, and the same 6LoWPAN network configurations. When all the nodes are in the same 6LoWPAN wireless network, the network establishment stage is complete. The border-router could then send test messages and collect the RTT and PDR data.

The border router node sends only CoAP GET messages to other test nodes, and measure the RTT and PDR information according to the time taken before ACK message is received. At the same time, the test nodes will randomly send CoAP GET to a random node, in order to simulate the random network traffic in the wireless sensor networks. If the target node received a CoAP GET message, it will respond a CoAP ACK message to the sender with some answering information in the payload.

Random traffic generation could be turned on or off independently, therefore, the interference traffic load in the network is also controllable according to the needs. With the parameters set, the test groups listed in section 3.1 could be detailed to the following groups:

- **2 transmissions per second and load in the network is zero:** The border-router acts as a CoAP client and the other nodes in the network act as CoAP servers. Every 0.5s the router sends a CoAP GET message to a selected node and waits for ACK message. The total transaction for a node is 300. There is no extra traffic in the network.

- **2 transmissions per second and load in the network is 0.2 transmissions per second per node:** Every 0.5s the router sends CoAP GET message to a selected node and waits for ACK message. The total transaction for a node is 300. Meanwhile every node in the network sends CoAP GET message to a random node and waits for ACK message with time interval of 5s.
- **Burst traffic and load in the network is zero:** The router continuously sends CoAP GET message to a selected node and waits for ACK message. The total transaction for a node is 300. There is no extra traffic in the network.
- **Brust traffic and the load in the network is 0.2 transmissions per second per node:** The router continuously sends CoAP GET message to a selected node and waits for ACK message. The total transaction for a node is 300. Meanwhile, every node in the network sends CoAP GET message to a random node and waits for ACK message with time interval of 5 seconds.

There are 19 target test nodes in the network and 4 test groups. And each test group has 300 valid RTT data. Therefore, a total number of 22800 valid data shall be acquired.

4.3.2 Test Node Deployment

There a a total number of 20 nodes in the test. One of the nodes acts as the border router and it is connected to a laptop in a fixed place. The other 19 nodes are deployed in the whole office building on different floors, in order to reach the maximum coverage inside the buildings. Node deployment are implemented as figure 4.5 shows. The nodes are spread into the two office buildings. There is a bridge connection between building 206 and building 194 on floor C, which is within the effective communication range of nodes.

The red circle on floor C or the third floor denotes the place where the border-router is installed. Other nodes are placed elsewhere in the building, either in different rooms or at different places along the corridor on different floors. Figure 4.6 provides a vertical view of the node placement in the buildings. The estimated network topology should be distance-based as what the figure shows. However, the real network topology is slightly different from that, since there are interference at different parts of the building, and the structure of the indoor environment might also affect the communication quality. The RPL routing protocol will try to optimize the routing if the network quality is not good enough.

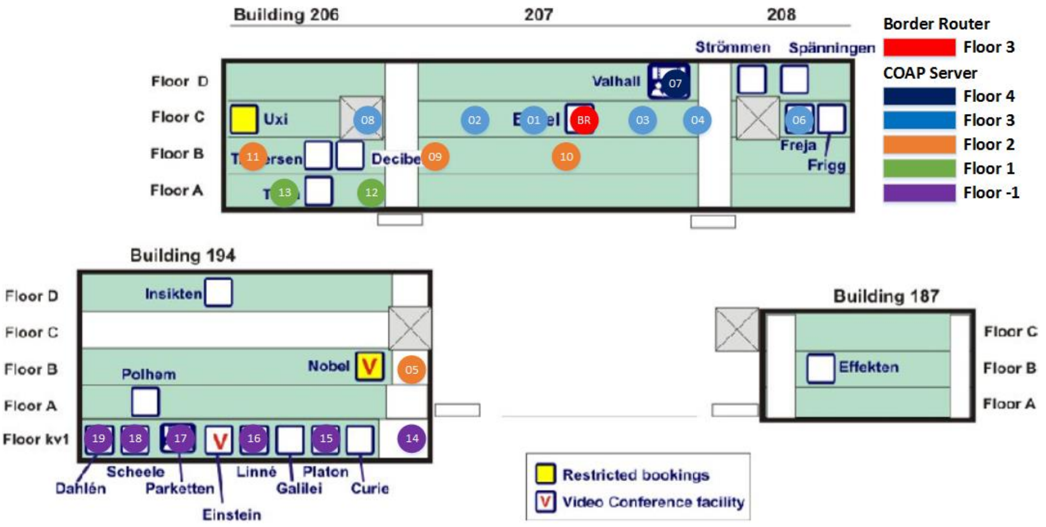


Figure 4.5: Node deployment in the building

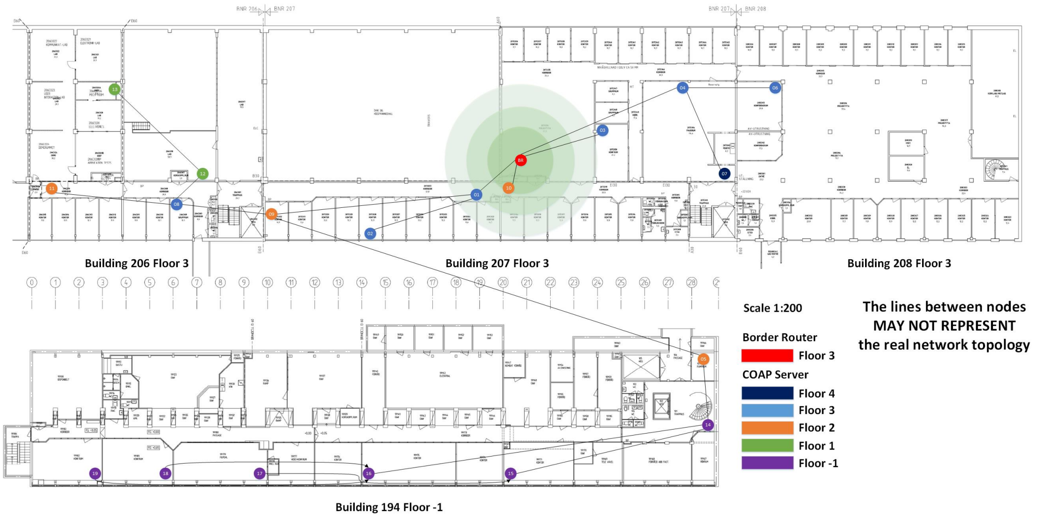


Figure 4.6: Estimated Network Topology

4.4 Power Consumption Test Setup

Power consumption test are mainly performed using the oscilloscope and the current probe, since the current running on the low-power boards are rather low. Figure 4.7 and figure 4.8 shows the target node installation and the equipment prepared for the power consumption test. All the unnecessary jumpers on the SmartRF06 Evaluation Board are removed, in order to eliminate the current through peripheral. And all of the unnecessary modules in the test code are removed, so that the test could focus on the normal sending and receiving tasks. The current measured is amplified by 30 times on the jumper to raise the accuracy, because of the rather low current running on the chip.

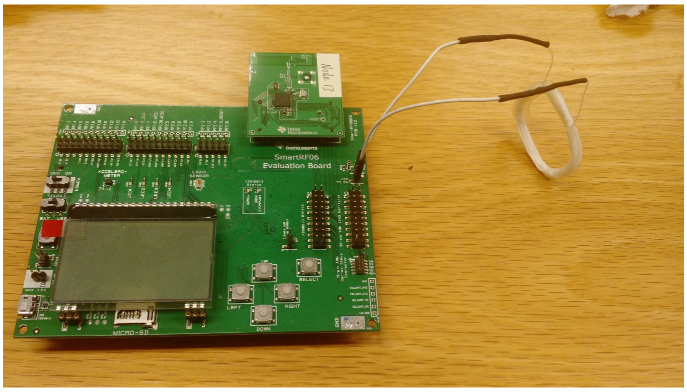
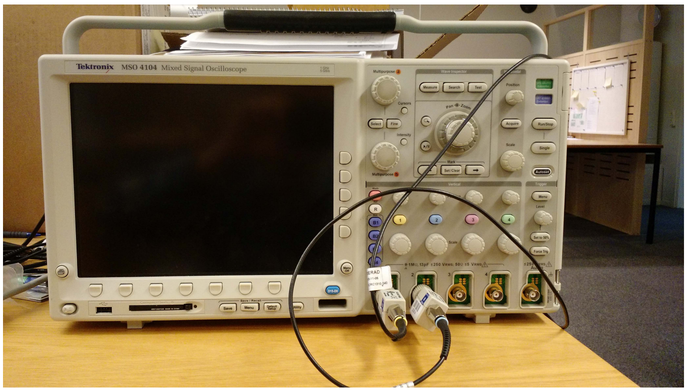
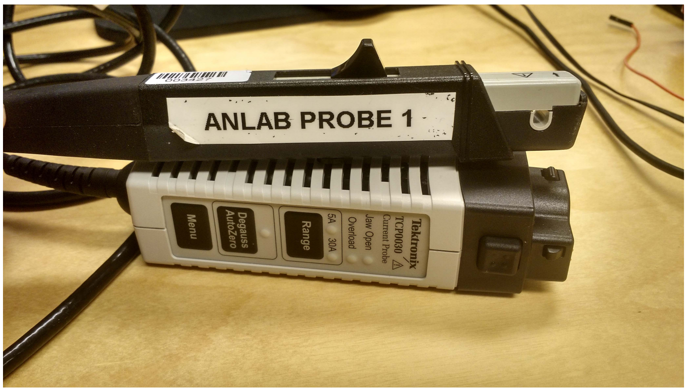


Figure 4.7: Target power node for testing

The target device is configured as a CoAP client, running a simple application unicasting a CoAP GET message to a specific CoAP server. Both Null-RDC and ContikiMAC RDC drivers are tested, in order to confirm whether ContikiMAC could save as much energy as the developer proposed in the paper. The real-time current measurement could be shown on the oscilloscope. Andhe data collected could then be stored to an USB device, and handled in the Matlab. The three stages listed in table 3.2 including idle listening, transmitting and receiving could be recognized directly on the current graph.



(a) Tekronix MSO 4104 Oscilloscope



(b) Tekronix TCP0030 Current Probe

Figure 4.8: Oscilloscope and current probe

Chapter 5

Data Analysis

This section presents the data collected and the results from the performance test. Stability test is more qualitative test, trying to find the maximum single-hop coverage or the outbound place where the connection is still stable. The results are analyzed in section 5.1. Latency and reliability test and power consumption test are more quantitative tests. In section5.2 RTT and PDR data are collected and analyzed, evaluating the performance of the whole 6LoWPAN network. Power consumption data and results are shown in section 5.3.

5.1 Stability Test Results

The test results for the stability test is very interesting. The coverage range results could be divided into three groups, according to the distance away from the border-router or 6LoWPAN gateway at the center. Figure 5.1 is the visual map of the communication map on floor C, where the border-router lies on. Figure 5.2 shows the coverage range on floor B and floor D, which is result on the upper floor and lower floor. Figure 5.3 are the coverage ranges for the floors that is furthers away from the central border-router. The red signal on each figure stands for the relative place of the central border-router on each floor. Different colors stands for different radio power settings, while the shapes stands for different test groups.

First of all, it could be easily found on each figure, that almost all the result points are distributed symmetrically around the central border-router, on both vertical side and horizontal side. And that is one of the reasons why the test results could be divided into the three groups. Note that the point distribution on floor K is quite different from floor A. That is because floor K is the underground floor, and there are more metal materials between floor

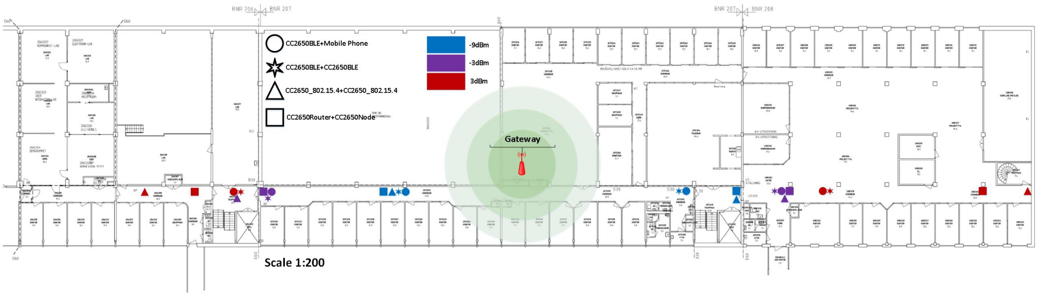
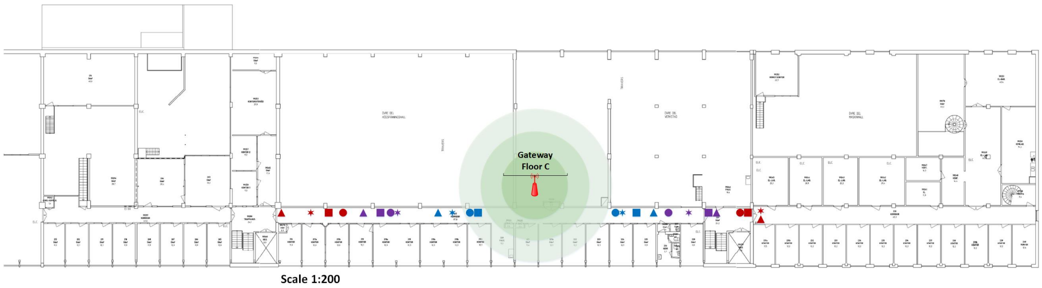


Figure 5.1: Coverage range on border-router floor

K and floor A due to the labs located on floor K. Therefore, hardly could the wireless signals go through 3 floors reaching the underground floor directly.

(a) Floor D

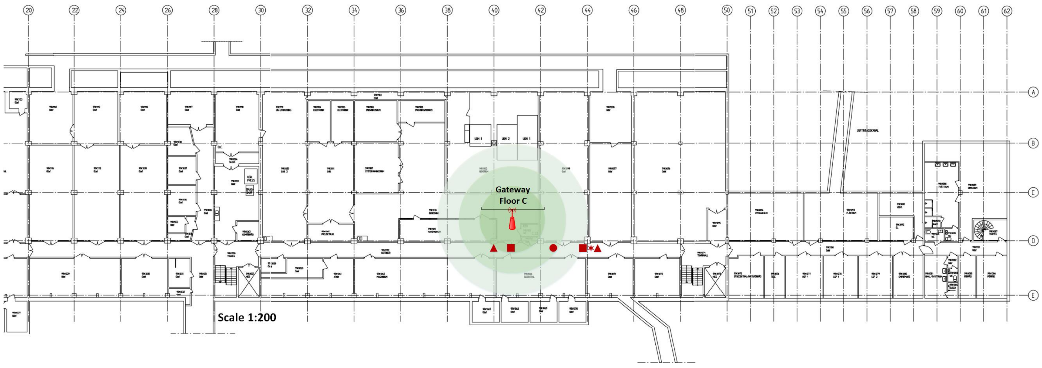


(b) Floor B

Figure 5.2: Coverage range on floor D and floor B

In terms of different protocols or different test groups, the single-hop coverage range seems to be more or less the same when the radio power is the same. Different shapes with the same color are almost distributed on the same place. Especially for the first 2 test groups, the BLE communication range is less dependent on the type of receiver. When considering the transmit range on the same floor, pure IEEE 802.15.4 packets and CoAP packets are able to reach slightly longer distance than BLE packets. The transmit power seems to be the major factor that affects the coverage range, which is within our expectation. Larger radio transmit power brings larger coverage range. -9dBm power setting sample dots lies closest to the border-router, while +3dBm power setting sample dots reaches the furthest place. On floor K, only samples with +3dBm are able to receive the packets from the border-router, but also only with a short distance from the place right under the border-router. On the other hand, the device consumes more power at the same time if the radio power is high. There is a trade-off between the coverage range and the performance.

(a) Floor A



(b) Floor K

Figure 5.3: Coverage range on floor A and floor K

Table 5.1 presented the exact single-hop coverage test results. These numbers are measured in meters and are calculated from the maps. Obviously test group 3 with pure IEEE 802.15.4 packets have the best signal coverage, closely followed by the 6LoWPAN packets. The two groups sending Bluetooth LE bacon have almost the same coverage, which is mentioned before. But, they have significant disadvantage on single-hop coverage or stability in terms of coverage. The results for the 6LoWPAN test group are also regarded as an important reference for the node deployment in the latency and reliability test.

Transmit Power (dBm)	Floor	BLE→ Phone	BLE→ BLE	802.15.4→ 802.15.4	Border Router→ 6LoWPAN node
+3	C	39.5	40.7	66.3	60.5
	B	27.6	30.5	34.0	28.7
	K	6.8	11.5	12.3	10.3
-3	C	34.2	33.5	37.3	35.2
	B	19.5	21.0	24.5	23.5
	K	Null	Null	Null	Null
-9	C	21.8	21.1	28.3	28.3
	B	11.4	12.4	16.3	14.1
	K	Null	Null	Null	Null

Table 5.1: Signal coverage result displayed in meters

5.2 Latency and Reliability Test Results

As mentioned in section 3.2 and section 4.2, the latency and reliability of the 6LoWPAN network are measured with the round-trip time (RTT) and packet delivery rate (PDR). This section provides the evaluation results of the 6LoWPAN network in terms of RTT and PDR.

The latency and reliability test is carried out inside two buildings with complex indoor environment. The real network topology may vary due to the self-healing characteristic provided by RPL routing protocol. The nodes tend to find a better route if the network quality of the original one is not good enough or there are interference on the old route. When processing the data, it can be easily observed that the RTT data is accurately related to the hop count from the sender (border-router). The round trip time taken for the successfully sent requests are approximately proportional to the hops taken from the border-router. Therefore, the data collected are divided into different groups according to the exact hops from the root or the border-router. Cumulative distribution function (CDF) curves are drawn to visualize the data.

Figure 5.4-5.9 are the CDF curves generated by Matlab, based on the data collected in the latency and reliability test. There are three vertical lines on each of the graph, indicating the possible deadline requirements for some specific industrial scenarios. The value of these deadlines for RTT are 150ms, 250ms and 1000ms from left to right.

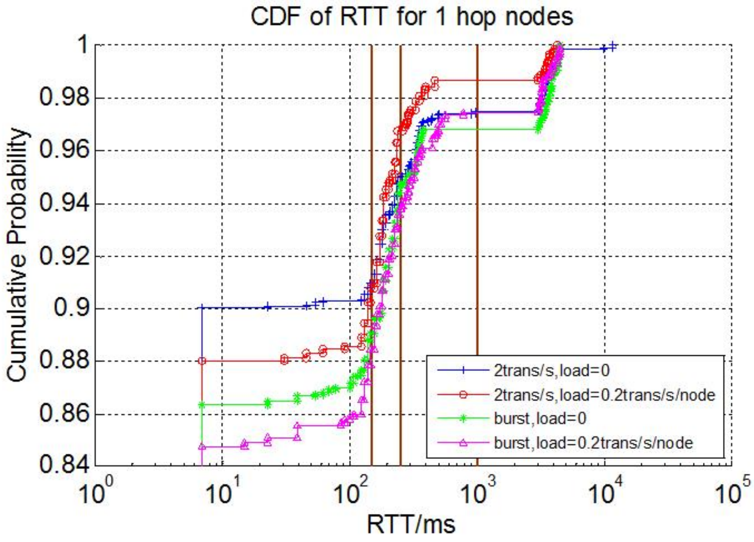


Figure 5.4: Latency result for 1-hop nodes

From the single-hop curve in figure 5.4, the probability of RTT equal to 10ms or less varies from 0.9 to 0.85 depending on the scenario discussed in section 4.2. Most of the responses are received within 10ms, which shows that the network quality is very good. The probability of achieving the first deadline 150ms, varies from 0.88 to 0.91 depending on the scenario, in which the group with 2 transactions per second and no extra traffic is the highest and the group with burst request packets and extra load perform the worst. Also for one-hop nodes, a packet has a probability of at least 0.97 to respond with 1000ms, and the number could reach as high as 0.99 when there are no extra traffic load in the network.

For other multi-hop nodes, it can be seen that the probability of achieving the deadlines of each scenario suffers from a slight decrease. For example, when the number of hops is 3, the probability of achieving a RTT of 250ms varies from 0.85 to 0.93, depending on the scenario, whereas it changes to 0.5 to 0.92. It can also be observed that the probability that RTT has a value of less than 250ms is always around 0.9 for all hops, when there are no extra traffic loads in the network. When extra traffic exists, the performance of the network is downgraded.

Figure 5.10-5.12 shows the probability density function (PDF) curves

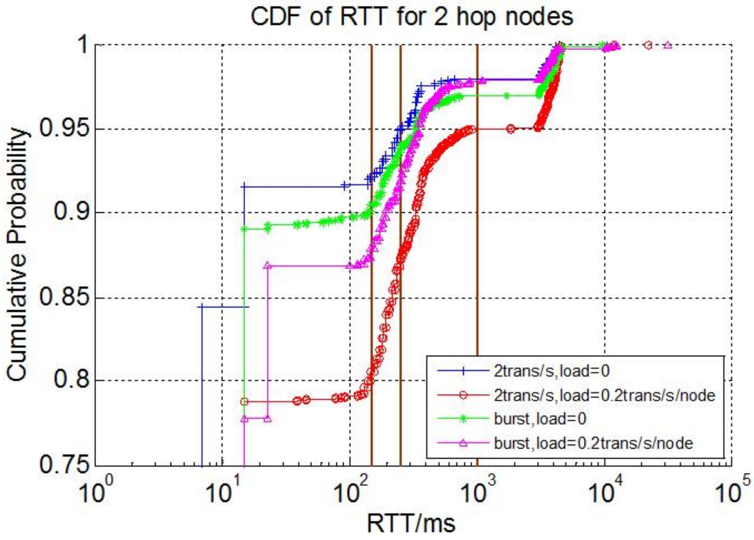


Figure 5.5: Latency result for 2-hop nodes

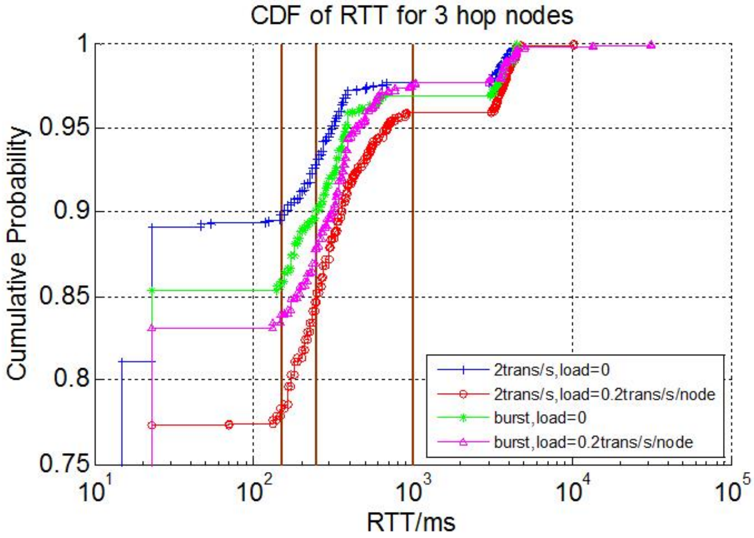


Figure 5.6: Latency result for 3-hop nodes

with respect to number of hops for each test group. Since re-transmission mechanism is introduced in CoAP, almost all the packets are successfully sent to the target node. The only difference is that the re-transmission takes significant longer time if the re-transmission is triggered multiple times.

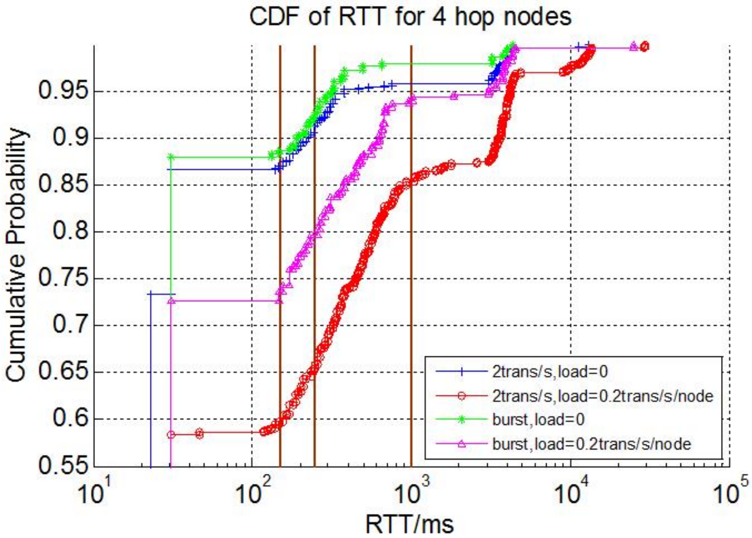


Figure 5.7: Latency result for 4-hop nodes

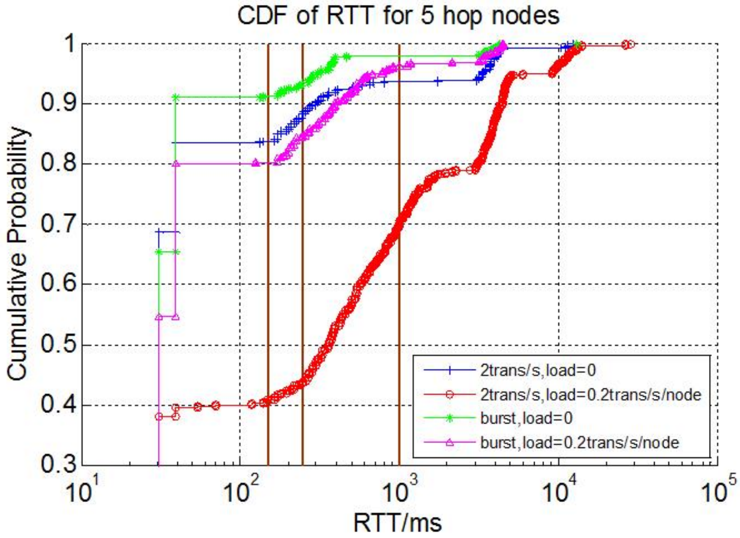


Figure 5.8: Latency result for 5-hop nodes

Therefore, the deadlines are defined to filter these cases. If the RTT of a packet is longer than the deadline, it could be considered 'lost', because it does not meet the deadline required. Thus, the PDR is calculated in terms of deadlines.

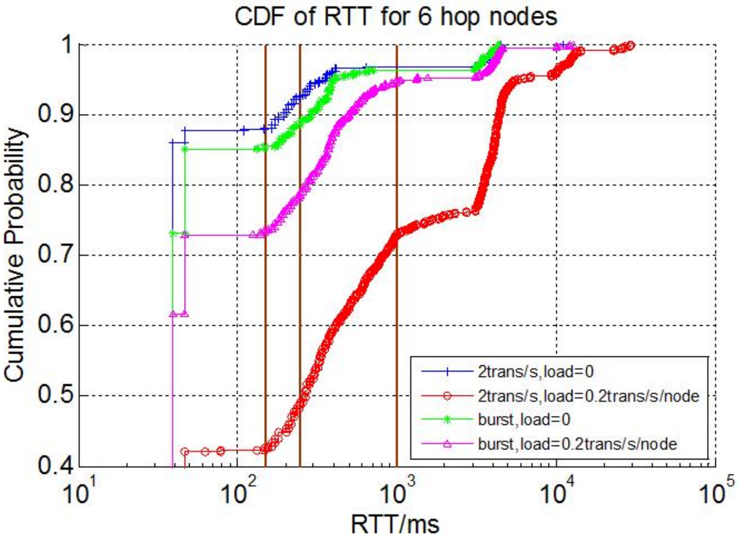


Figure 5.9: Latency result for 6-hop nodes

When the deadline is set to 150ms, it can be clearly seen from figure 5.10 that one-hop nodes have the highest PDR. Note also, that test cases with no extra traffic in the network all have a better performance than those with extra traffic. The PDR decreases as the hop increases. It is also within our expectations, since more hops increases the possibility of packet loss, and may trigger more re-transmissions, leading to longer transmission time. An exception appears that the PDR of 5-hop nodes is larger than 6-hop nodes when extra traffic load exists in the network. The reason for the exception might lie in the network topology or the node deployment. The 5-hop node is the only routing point between two sub-nets. Therefore, there may be much network congestion if the nodes in both sub-nets try to communicate with nodes in the other sub-net, which causes the strange decrease on 5-hop nodes.

On figure 5.11 and figure 5.12, the deadline is extended to 250ms and 1000ms, which brings higher PDR than the first deadline. Compared to the PDR at 150ms, the PDR at the latter two graphs reaches as high as 0.95 for 1-hop nodes. Even for the longest route, 95 per cent of the test packets are successfully sent to 6-hop nodes.

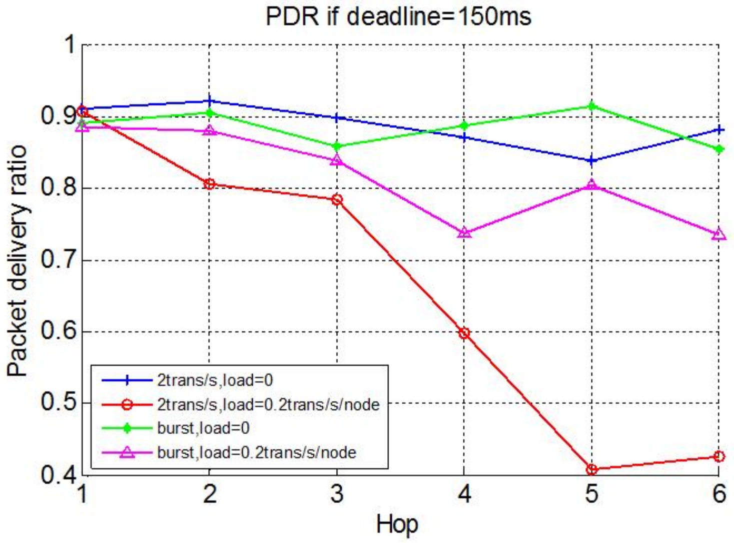


Figure 5.10: Packet delivery ratio with 150ms

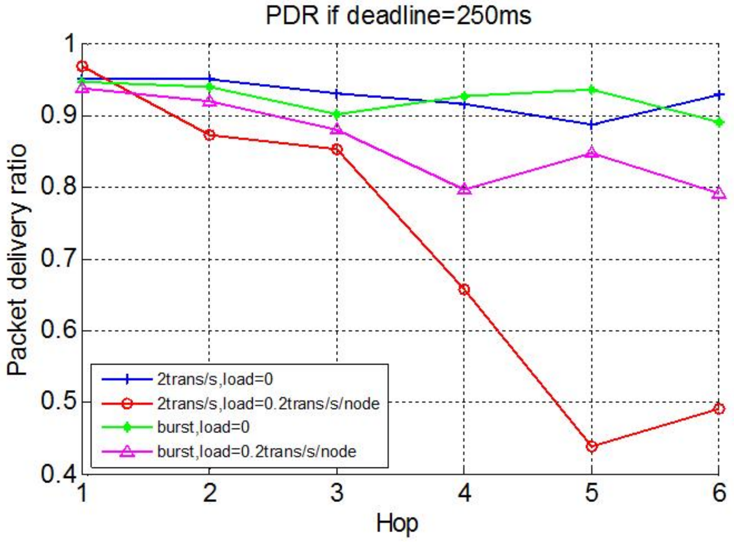


Figure 5.11: Packet delivery ratio with 250ms

5.3 Power Consumption Test Results

The power consumption test data and analysis are presented in this section. To obtain more accurate data, all the power data are captured by the oscil-

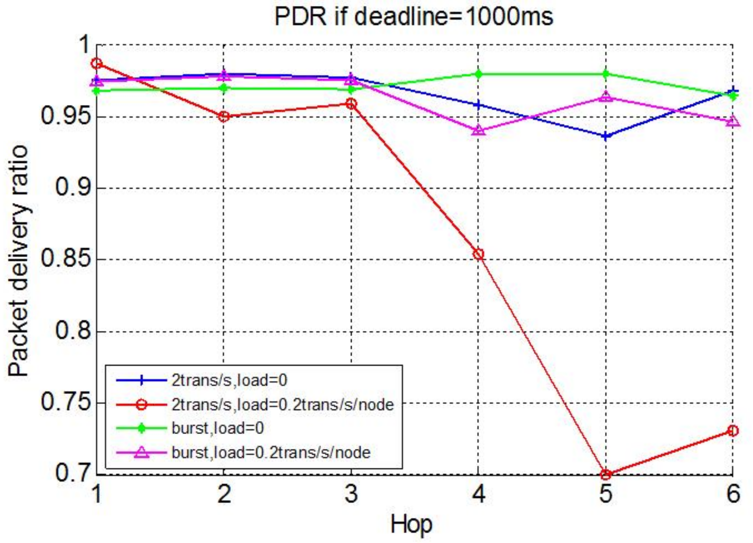


Figure 5.12: Packet delivery ratio with 1000ms

loscope. Current curves are generated with these data as input, and those valid groups of data are filtered manually. The analysis are based on those filtered data.

The power consumption test mainly focuses on the power consumed on each transaction, and try to analyze the packet delivery process based on the current graph observed. There are many RDC drivers supported by Contiki, and the test is done for the NullRDC and the power-efficient ContikiMAC. Several groups of tests are performed in order to reduce the standard errors. Figure 5.13 is the current curve for a single transaction with NullRDC, while figure 5.14 is the current curve for ContikiMAC nodes.

The whole transaction shown on figure 5.13 could be roughly divided into three stages, the transmitting stage (TX), idle stage (Idle), and the receiving stage (RX). The first peak of the TX stage indicates the wake up of MCU, and the chip starts to do some pre-processing work, including message packaging and some hardware initiation. The RF chip will then enter working mode, performing a clear channel check (CCA) to confirm the communication channel is clear for use. The CCA is required by the CSMA/CA mechanism at MAC layer in order to avoid the collision with other traffic on the channel. The chip starts to transmit CoAP GET message after CCA check is done. The transmit time is determined by the packet length and radio frequency. The current level reaches about 10mA in the transmit stage, and drops to about 8mA in a post-processing stage, waiting for a link

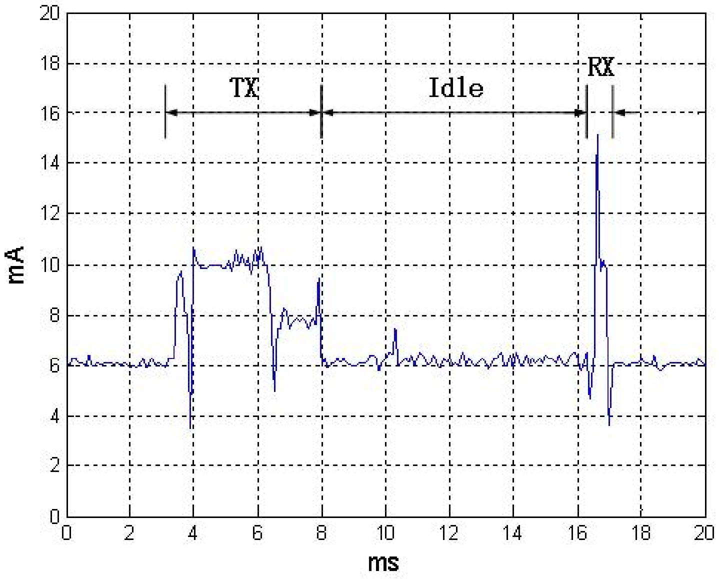


Figure 5.13: Power consumption without RDC (NullRDC)

layer acknowledgement. Then the MCU drops to low-power mode again, while the RF chip stays awake, waiting for the response message. Therefore, the power consumed in the idle stage is mainly from the RF chip. When the CoAP response ACK packet arrives, the chip enters receiving stage (RX), reaching another peak on the current curve. After receiving the packet, the chip returns a link layer ACK message, and unpacks the received packet at the same time. Therefore, the current level climbs to the highest in the whole process and drops quickly afterwards. The MCU turns back to low-power mode again, waiting for next wake-up on other tasks.

The current level for ContikiMAC configuration seems to be more complicated than the NullRDC one. As shown on figure 5.14, the whole transaction also includes three stages similar to the NullRDC one. However, the curve for the ContikiMAC is starting from the place near 0mA, while the NullRDC one is from 6mA. This is because the ContikiMAC shuts down the RF chip in idle stage, and tries to keep both MCU and RF chip in low-power mode for most of the time. The RF chip in NullRDC configuration device is kept always on, listening on the channel for incoming packets. For ContikiMAC devices, the RF chip will be turned on periodically, and the MCU will be wake up once link layer ACK is received in the previous period.

In the transmitting stage, the MCU wakes up to do some pre-processing

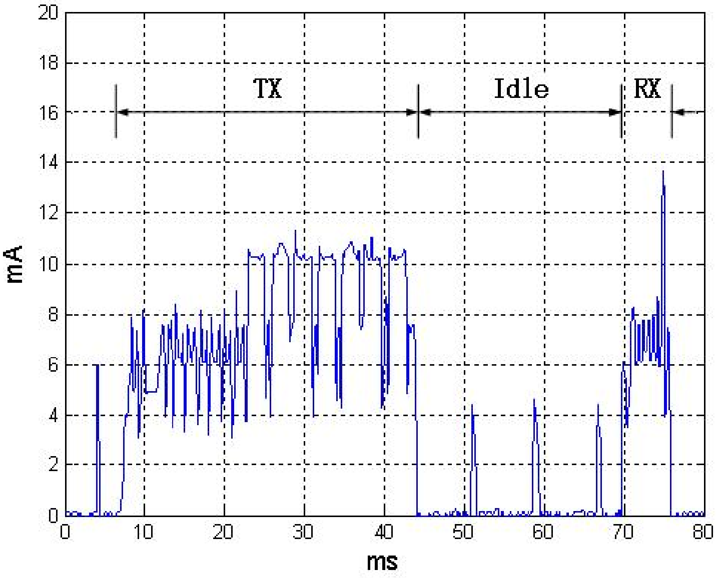


Figure 5.14: Power consumption with RDC (ContikiMAC)

work before transmission. After that, the RF chip performs 6 groups of CCA checks, 2 times for each group by default. Therefore, there are 12 small peaks in the TX stage. If the channel is clear after multiple CCA checks, the chip starts to send the packet continuously, in order to confirm that the packet could reach the target node when it is in active period. The re-transmission will stop once a link layer ACK is returned from the target node, or when the re-transmission exceeds the maximum times configured. If the does not receive the link layer ACK message, the continuous transmission will be performed in the next active period. Both MCU and RF chip will be dropped to low-power mode in the idle stage. The small pulses in idles stage stands for the wake-up period for the RF chip. If no ACK message is received, the chip will then quickly turn back to sleep mode again, keeping the current at low level most of the time. The default channel check rate is 8Hz, or the period is 125ms, which is exactly the same as expected on the graph. When traffic in the channel is detected in an active period, the RF chip will be kept on waiting for the next packet sent to the device. The high peak in the RX stage is similar with the one in the NullRDC graph, indicating the reception of a full packet. A link layer ACK message is returned to the target node after a successful reception, and the whole chip drops to low-power mode again.

Table 5.2 shows the exact average current at each stage in a single transaction. The total amount of power consumed is not calculated, because the total transmitting time of the ContikiMAC device depends largely on the network environment and the RDC of the receiver. However, the current level statistics still clearly shows that the average current is lower in all stages for the ContikiMAC devices, especially for the idle stage. If the device is in idle stage for most of the time, the ContikiMAC configuration will be having a grate advantage compared to the NullRDC configuration.

RDC type	State	Average Current (mA)
NullRDC	TX	8.765
	Idle	6.198
	RX	8.514
ContikiMAC	TX	7.754
	Idle	0.311
	RX	6.463

Table 5.2: Average current level in each stage

Chapter 6

Conclusions

The thesis performed a thorough evaluation of the Contiki-based 6LoWPAN network stack running on the newly marketed TI CC2650 platform, since the combination of CC2650 and 6LoWPAN protocol stack are the most promising choice that fulfills the requirements of native IP support for building automation. A black-boxed test plan is designed in chapter 3, aiming at testing the stability, communication latency and reliability, and power consumption of a specific combination of WSN hardware and software. Chapter **??** states the experimental setup of the performance test, with detailed introduction on the target hardware and selected network stack. Test code are written according to the test plan, implementing the 6LoWPAN network protocol stack on the CC2650 board. Finally, test data are collected to evaluate the performance scientifically.

From the stability test, it could be concluded that the communication range is slightly affected by the protocols running. But the affect is not obvious, especially when the radio chip is configured to transmit in lower power. Still, the most important factors that affect the single-hop coverage range or the stability of transmission is the radio transmit power. The coverage gets significant increase as the transmit power increases. However, the power consumption shall increase at the same time theoretically. There should be a trade-off between increasing the transmit power and reducing the total power consumption. Unfortunately, there are not enough time and equipment to confirm the assumption. On the other hand, the structure seem to have little effect on the transmission range.

The latency and reliability test results shows that the 6LoWPAN network has a low RTT and high PDR when the traffic load is low and the route length is short. When the route length increases, the RTT increases as more hops leads to longer response time. The reliability of the network drops due to the accumulating re-transmission at each hop on the route. When the traffic load in the network increases, the reliability of the network also decreases, because of the CSMA/CA collision avoidance mechanism. According to the PDR test result, the PDR increases for a higher deadline and decreases with number of hops for all the scenarios. However, it remains around 0.9 for all hops. When there is no traffic in the network, the probability that RTT has a value of 250ms is always around 0.9 for all hops, which meets the expectation for building automation.

The power consumption test compares two different RDC mechanisms. The results show that ContikiMAC is very power-efficient and suitable for low-power wireless sensor networks. Also, the TI CC2650 platform with Contiki OS running 6LoWPAN shows competitive results for the ideal industrial wireless sensor network solutions for IP-based building automation applications.

In conclusion, the TI CC2650 platform running 6LoWPAN network stack has great advantage in terms of power consumption, multi-standard radio and support for Contiki and 6LoWPAN. The stability test with multiple radios shows that the platform is stable. The CDF for a 250ms RTT is 0.9 for all hops when there are no extra traffic in the network, which serves the timing requirements required by user experience in building automation systems. Also a relatively high value of PDR of around 0.9 for all deadlines and hops is established when there is no traffic in the network. However, further improvement in terms of RTT and PDR is required when there is additional traffic load in the network. And the precision of the power consumption tests needs to be improved, since the current probe used in the test is on the scale of ampere rather than the required scale of microampere. The researchers might be able to get more accurate test results with advanced equipm.

Bibliography

[1] Assarsson, J., and Karlsson, L. Performance of constrained wireless devices in the internet of things. *University Essay from Lunds Universitet/Datavetenskap* (2013), 1650–2884.

[2] Baker, N. Zigbee and bluetooth: Strengths and weaknesses for industrial applications. *Computing and Control Engineering 16*, 2 (2005), 20–25.

[3] Bluetooth, S. Specification of the bluetooth system: Covered core package version: 4.0, 2010.

[4] Controls, K. Understanding building automation and control systems, 2013. [Online; accessed 27-July-2015].

[5] Dunkels, A. Contiki: Bringing ip to sensor networks. *ERCIM News*, 76 (2009).

[6] Dunkels, A. The ContikiMAC radio duty cycling protocol. Technical report, SICS, Dec 2011.

[7] Gay, D., Levis, P., Von Behren, R., Welsh, M., Brewer, E., and Culler, D. The nesc language: A holistic approach to networked embedded systems. In *Acm Sigplan Notices* (2003), vol. 38, ACM, pp. 1–11.

[8] Gnawali, O. The minimum rank with hysteresis objective function.

[9] Gomez, C., Oller, J., and Paradells, J. Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology. *Sensors 12*, 9 (2012), 11734–11753.

[10] Hill, J., Szewczyk, R., Woo, A., Hollar, S., Culler, D., and Pister, K. System architecture directions for networked sensors. In *ACM SIGOPS operating systems review* (2000), vol. 34, ACM, pp. 93–104.

[11] Ko, J., Dawson-Haggerty, S., Gnawali, O., Culler, D., and Terzis, A. Evaluating the performance of rpl and 6lowpan in tinyos. In *Workshop on Extending the Internet to Low Power and Lossy Networks (IP+ SN)* (2011).

[12] Ko, J., Terzis, A., Dawson-Haggerty, S., Culler, D. E., Hui, J. W., and Levis, P. Connecting low-power and lossy networks to the internet. *Communications Magazine, IEEE 49*, 4 (2011), 96–101.

[13] Kovatsch, M., Duquennoy, S., and Dunkels, A. A low-power coap for contiki. In *Mobile Adhoc and Sensor systems (MASS), 2011 IEEE 8th International Conference on* (2011), IEEE, pp. 855–860.

[14] Kushalnagar, N., Montenegro, G., and Schumacher, C. RFC 4919: IPv6 datagrams over low-power wireless personal area networks (6LoWPANs) overview, assumptions, problem statement, and goals. Informational, IETF, Aug 2007.

[15] Langhammer, N., and Kays, R. Performance evaluation of wireless home automation networks in indoor scenarios. *Smart Grid, IEEE Transactions on 3*, 4 (2012), 2252–2261.

[16] Montenegro, G., Kushalnagar, N., Hui, J., and Culler, D. RFC 4944: Transmission of IPv6 packets. Proposed standard, IETF, Sept 2007. DOI 10.17487/RFC4944.

[17] Pang, Z. *Technologies and Architectures of the Internet-of-Things (IoT) for Health and Well-being.* PhD thesis, KTH Royal Institute of Technology, 2013.

[18] Pang, Z., Cheng, Y., Johansson, M. E., and Bag, G. 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* (2015), IEEE, pp. 163–167.

[19] Potsch, T., Kuladinithi, K., Becker, M., Trenkamp, P., and Goerg, C. Performance evaluation of coap using rpl and lpl in tinyos. In *New Technologies, Mobility and Security (NTMS), 2012 5th International Conference on* (2012), IEEE, pp. 1–5.

[20] Shelby, Z. Constrained restful environments (core) link format.

[21] Shelby, Z., Chakrabarti, S., Nordmark, E., and Bormann, C. Neighbor discovery optimization for ipv6 over low-power wireless personal area networks (6lowpans). Tech. rep., 2012.

[22] Thubert, P. Objective function zero for the routing protocol for low-power and lossy networks (rpl).

[23] TI. Zigbee wireless networking overview. Tech. rep., Texas Instruments, 2013.

[24] Townsend, K., Cufí, C., Davidson, R., et al. *Getting started with Bluetooth low energy: Tools and techniques for low-power networking.* " O'Reilly Media, Inc.", 2014.

[25] Wikipedia. 6lowpan — wikipedia, the free encyclopedia, 2015. [Online; accessed 11-July-2015].

[26] Wikipedia. Bluetooth low energy — wikipedia, the free encyclopedia, 2015. [Online; accessed 11-July-2015].

[27] Wikipedia. Building automation — wikipedia, the free encyclopedia, 2015. [Online; accessed 3-July-2015].

[28] Wikipedia. Zigbee — wikipedia, the free encyclopedia, 2015. [Online; accessed 12-July-2015].

[29] ZigBee, A. Zigbee network specifications, 2015. [Online; accessed 12-July-2015].