BRNO UNIVERSITY OF TECHNOLOGY
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF INFORMATION TECHNOLOGY
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

DEPARTMENT OF COMPUTER SYSTEMS
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

EXTENSION OF WIRELESS SENSOR PROTOCOL

ROZŠÍŘENÍ PROTOKOLU BEZDRÁTOVÉ SENZOROVÉ SÍTĚ

BACHELOR'S THESIS
BAKALÁŘSKÁ PRÁCE

AUTHOR **KLÁRA NEČASOVÁ**
AUTOR PRÁCE

SUPERVISOR **Ing. JAN KOŘENEK, Ph.D.**
VEDOUCÍ PRÁCE

BRNO 2017

Abstract

The aim of this thesis was to extend the wireless communication FIT protocol including mechanisms for adding new devices to the network, packet routing within the network and network reinitialization after device movement. The theoretical part of the work focuses on smart homes protocol Z-Wave, ZigBee and Thread. In the practical part, new protocol features have been designed and implemented. Moreover, various types of communication between devices also extend the FIT protocol. All new functionalities were carefully tested. Moreover, the tests focused on the influence of a distance and a device PCB antenna position on the success rate of unreliable and reliable communication. The results showed that the success rate of data transfer depends only on a distance between devices and a selected type of communication. The influence of obstacle made of a different material on received signal strength during data receiving was confirmed. The work also discusses dependency of energy consumption on transmitted data length and used communication protocol.

Abstrakt

Cílem práce bylo rozšířit bezdrátový komunikační FIT protokol o mechanismy zajišťující přidávání zařízení do sítě, směrování paketů v síti a obnovení sítě po přesunu zařízení. Teoretická část práce se zabývá komunikačními protokoly Z-Wave, ZigBee a Thread využívanými v inteligentních domácnostech. V rámci praktické části byly zmíněné mechanismy navrženy a implementovány. Dále byl protokol rozšířen o různé způsoby komunikace mezi zařízeními. Všechny nové funkcionality byly důkladně otestovány. Testy se mimo jiné zaměřovaly na vliv vzdálenosti a natočení antén zařízení na úspěšnost komunikace, která probíhala spolehlivým i nespolehlivým způsobem. Z výsledků vyplynulo, že úspěšnost komunikace závisí pouze na vzdálenosti mezi zařízeními a na zvoleném způsobu komunikace. Dále byl potvrzen vliv materiálu překážky umístěné mezi zařízeními na sílu signálu při příjmu dat. Práce se také zabývá spotřebou energie koncového zařízení v závislosti na délce vysílaných dat a použitém komunikačním protokolu.

Keywords

Wireless Sensor Network (WSN), FIT Protocol, Internet of Things (IoT), Smart Home.

Klíčová slova

Bezdrátová senzorová síť (WSN), FIT protokol, internet věcí (IoT), inteligentní domácnost.

Reference

NEČASOVÁ, Klára. *Extension of Wireless Sensor Protocol.* Brno, 2017. Bachelor's thesis. Brno University of Technology, Faculty of Information Technology. Supervisor Kořenek Jan.

Extension of Wireless Sensor Protocol

Declaration

Hereby I declare that this bachelor's thesis was prepared as an original author's work under the supervision of Mr. Ing. Jan Kořenek, Ph.D. All the relevant information sources, which were used during preparation of this thesis, are properly cited and included in the list of references.

.....
Klára Nečasová
May 18, 2017

Acknowledgements

I would like to thank my supervisor Ing. Jan Kořenek, Ph.D. for his patient guidance and valuable advice. I would also like to thank my advisor Ing. Josef Hájek, Ph.D. for all his support.

Contents

1 Introduction **5**

2 Smart Home on the Application Level **7**
2.1 Energy Management . . . 8
2.2 Renewable Energy Management Driven Smart Home . . . 8
2.3 Health Care Systems . . . 9
2.4 Advanced Multimedia Services . . . 9

3 Overview of Selected Protocols **10**
3.1 Z-Wave . . . 10
3.1.1 Joining Process . . . 11
3.1.2 Routing Process . . . 12
3.1.3 Network Reinitialization . . . 13
3.2 ZigBee . . . 14
3.2.1 Joining Process . . . 14
3.2.2 Routing Process . . . 16
3.2.3 Network Reinitialization . . . 17
3.3 Thread . . . 17
3.3.1 Joining Process . . . 18
3.3.2 Routing Process . . . 19
3.3.3 Network Reinitialization . . . 20

4 Protocol Design **21**
4.1 Protocol Requirements . . . 21
4.2 FIT Protocol . . . 22
4.3 Types of Communication . . . 23
4.3.1 Reliable Data Delivery . . . 23
4.3.2 Unreliable Data Delivery . . . 24
4.3.3 Control Packets . . . 25
4.4 Joining Process . . . 25
4.5 Routing Process . . . 27
4.6 Network Reinitialization . . . 32

5 Hardware Platform **33**

6 Testing **35**
6.1 Test Case 1: Success Rate of Joining Process in Various Network Topologies 35
6.2 Test Case 2: Parent Choice of Joining Device . . . 36

6.3 Test Case 3: Success Rate of Joining Process in Various Distances 36
6.4 Test Case 4: Success Rate of Routing Process in Various Network Topologies 37
6.4.1 Sleepy Packets 38
6.5 Test Case 5: Success Rate of Network Reinitialization in Various Network Topologies 39
6.6 Test Case 6: Dependency of Data Transfer Success Rate on Distance Between Devices 40
6.7 Test Case 7: Dependency of Received Signal Strength on Obstacles Between Devices 42
6.8 Energy Consumption 43
6.9 Testing in the Simulator 45

7 Conclusion **46**

Bibliography **48**

Appendices **54**

A Results of Tests **55**

B Contents of the Attached CD **61**

List of Figures

3.1 Z-Wave: Joining process (inclusion) . . . 12
3.2 ZigBee: Joining process (commissioning) – forming a network . . . 15
3.3 ZigBee: Joining process (commissioning) – joining a device . . . 16

4.1 Data transaction from end device to PAN coordinator . . . 22
4.2 Four-way handshake communication . . . 24
4.3 Communication without waiting for ACK . . . 25
4.4 General packet sequence in joining process . . . 26
4.5 Joining process in tree topology . . . 27
4.6 FIT protocol structure . . . 28
4.7 Sample tree topology . . . 29

5.1 The Olimex A10-OLinuXino-LIME with the BeeeOn PAN coordinator module 33
5.2 The Microchip MiWi Demo Kit-868 MHz MRF89XA – front and rear view . 34
5.3 The BeeeOn sensor v1.2 – front and rear view . . . 34

6.1 Influence of distance on success rate of joining process . . . 37
6.2 Device movement and change of routing tables . . . 39
6.3 Positions of PCB antennas . . . 40
6.4 Success rate of reliable (blue columns) and unreliable (orange columns) data transfer . . . 41
6.5 Dependency of RSSI on distance and PCB antenna position (parallel – blue columns, orthogonal – orange columns) . . . 42
6.6 Device and obstacle placement . . . 42
6.7 Energy consumption of selected protocols . . . 45

List of Tables

4.1 Part of packet sent during distribution of routing tree 28
4.2 PAN RT – step 1 29
4.3 $COORD_2$ RT – step 1 29
4.4 PAN RT – step 2 30
4.5 $COORD_2$ RT – step 2 30
4.6 PAN RT – step 3 30
4.7 $COORD_2$ RT – step 3 30
4.8 PAN RT – step 4 30
4.9 $COORD_2$ RT – step 4 30
4.10 PAN RT – step 5 30
4.11 $COORD_2$ RT – step 5 30
4.12 PAN RT – step 6 31
4.13 $COORD_2$ RT – step 6 31
4.14 PAN RT – step 7 31
4.15 $COORD_2$ RT – step 7 31

6.1 Success rate of joining process 36
6.2 Success rate of joining process in dependency on distance 37
6.3 Success rate of routing process 38
6.4 Success rate of sleepy packet sending 38
6.5 Success rate of network reinitialization 40
6.6 Success rate of unreliable and reliable data delivery with respect to distance and PCB antenna position 41
6.7 Dependency of RSSI on obstacle 43
6.8 Energy consumption of selected protocols 44

A.1 Average of RSSI in various topologies 55
A.2 Variance of RSSI in various topologies 55
A.3 Standard deviation of RSSI in various topologies 56
A.4 Parent choice of a joining device – topology 1 56
A.5 Parent choice of a joining device – topology 2 57
A.6 Parent choice of a joining device – topology 3 58
A.7 Parent choice of a joining device – topology 4 59
A.8 Parent choice of a joining device – topology 5 60

Chapter 1

Introduction

The significant change in the world was caused by the introduction of computers and information technology. Suddenly information can be shared quickly and easily from any part of the world. Thanks to the opportunity of sharing ideas and information with each other, linguistic and geographic boundaries become much smaller. Two most important strategic issues for a success of every enterprise are especially information and communication. Nearly every organisation uses computers and communication tools today. Nevertheless, they are often still isolated, people do not communicate with others, and necessary information cannot be immediately accessed. These obstacles can be overcome through the effective usage of information technology, more specifically, computer networks. These networks are a new kind of computer systems produced by a need to merge computers and communications. Only with the help of computer networks can a borderless communication and information environment be built [6].

With a need of continuous managing and controlling of processes, sensors become essential. In the broadest definition, a sensor is an object whose purpose is to detect events or changes in its environment and then provide a corresponding output. It can be necessary to process gathered data and measure information on variables of interest, in a systematic fashion that enables one to answer research questions, test hypotheses and evaluate outcomes [59].

Industries around the world intend to use wireless sensor networks (WSNs) which, in contrast to their wired counterparts, are readily deployed with minimal effort while providing clear advantages in cost, size, power, mobility and flexibility [71]. First designed for military industry, they became very popular, and now wireless sensor networks are widely used in industrial, residential and wildlife environments, animal tracking, structure health monitoring, healthcare applications and home automation [35].

The Internet of Things (IoT) was originally defined for a machine to machine communication. Then it became to merge physical and virtual worlds. It results in smart environments where a large number of devices is connected to the Internet. Their task is to communicate and to share data within a wireless sensor network. the IoT applications help to make the world smarter in many fields. For instance, sensors can be helpful in home automation. They are suitable for monitoring of energy and water supply consumption. As a result, advice how to save costs and resources is obtained. the next application is associated with health service. Elderly or disabled people can live independently thanks to sensors which, for example, can detect a fall of a person. Sensors can also be appreciated in the environment field for monitoring of combustion gases and pre-emptive fire conditions to define alert zones. Monitoring of tap water quality in cities or detection of

leakages and wastes in rivers caused by factories are examples of a sensor usage in water management. the IoT application can be helpful as well in cities for monitoring of available parking spaces or monitoring of vibrations and material conditions in buildings.

One research area of the IoT is home automation, where home's electrical devices are connected to a central system with automatic control of devices based on user input [24]. It improves convenience, comfort, energy efficiency and security. Connected devices are smart electrical devices, connected to the Internet and sensors. the smart home is a combination of home automation, connected devices, and the IoT. Modern smart home can be easily controlled through a smartphone, a tablet or a computer [70].

It is quite difficult to find a suitable smart home protocol because of special requirements. the protocol has to support a large number of devices and offer the best possible device interoperability (ability for devices to talk each other). Other factors that have to be considered are power consumption, frequency band, range, data rate and, of course, cost [5]. None of existing protocols fulfils all these demands. ZigBee protocol is not very suitable for home automation because it is too complicated [42] and any implementation consumes a lot of power. Moreover, its signals are not directly compatible with any known computing device, like a smartphone, a tablet or a laptop and it is limited by the lack of interoperability among ZigBee devices [66]. Z-Wave protocol is primarily developed for home control, but it is not an open protocol [30]. Next drawback is a dependency on the only one manufacturer. Z-Wave also requires a gateway, which is a single point of failure. Furthermore, the protocol assumes that all devices are static [22]. Thread protocol has a close connection to Google which can be a drawback. Next disadvantage is radio communications in the 2.4 GHz band which might interfere with Wi-Fi signals [57]. Moreover, Thread network becomes saturated when it reaches approximately 200 nodes, but in the future, a typical family house could accommodate more than 500 smart devices [62]. More information about these protocols can be found in chapter 3.

Therefore new open source FIT protocol has been designed and implemented with the goal to meet all defined requirements for a smart home protocol. the FIT protocol is low-cost, low-power, with the range from 10 to 100 meters and with a low-frequency band. the protocol has been deployed in more than 10 households and used several months.

The work is divided into six chapters. the second chapter describes smart home functioning on the application level. Selected smart home protocols, their advantages and disadvantages and technical parameters are discussed in the third chapter. the design of the FIT protocol is the main topic in the fourth chapter. the main requirements of the protocol, types of communication and fundamental functions are also discussed in this chapter. the fifth chapter describes technical parameters of devices used for testing purposes. the sixth chapter includes several test cases that prove the correctness of implementation, the energy consumption analysis is also discussed. the seventh chapter concludes the work and proposes further improvements.

Chapter 2

Smart Home on the Application Level

A great extent of innovation and an expansion of new technologies allow humans to achieve a high level of comfort and wellbeing. However, an exploitation of our resources is not sustainable. If a consumption of energy remains at the same level, it can cause the end or at least a pause in a prosperity cycle. Taking into account given scenario, states will not be able to afford such expenses. For example, it can cause that the health care system will become expensive and pressure and demands on the health care providers will increase. Considering that the society is gradually getting older and the birth rate decreases, the medical care and services for the elderly has been a major health and ethical issue.

One of the solutions how to partly solve the problem is to build smart homes which offer accessibility technologies for elderly or disabled people. An example can be assistive domotics that is a form of home automation, and it focuses on making it possible for older adults and people with disabilities to remain at home. Moreover, smart homes allow people to manage home energy resources and improve their behaviour so as to reduce energy consumption. This appreciate all age groups of inhabitants. Smart home has recently become family-friendly. It's easy to operate, affordable, convenient, energy-efficient and professionally installed and supported. More information about smart home technologies can be found in [41] and [33].

Following four factors make this concept important:

- a fast progress and a miniaturisation observed in semiconductor technology resulting in a proliferation of computing and electronic devices in our everyday lives;
- an exponential growth of microcontrollers unit (MCU's) processing power;
- an integration of advanced signal conditioning in tiny sensor nodes that can measure and store data using complex processing techniques; and
- rapid development and progress of wireless technologies, essentially short range and low power applications.

A smart home or building is a home or a building that is equipped with specially structured wiring to enable occupants to remotely control or program various automated home electronic devices by entering a single command.

The smart home definition has not been established yet. L.C.D. Silva et al. [27] describe a smart home as a "home-like environment that possesses ambient intelligence and automatic control" capable of reaction to the behaviour of residents and to offer various accommodations. D. Zhang et al. [76], and M.A.A. Pedrasa [58] define four types of smart homes: healthcare based, multimedia and entertainment based, security based and energy efficiency based smart homes. The critical aspect is to possess a cheap, reliable and easy designed structure of communication.

The smart home will enable the management and control of different areas of a residence. Four distinctive general functional areas of service can be classified, which are: Energy Efficiency and Management, Health Care, Entertainment, and Security.

2.1 Energy Management

More than half of energy consumption in homes comes from electricity. The main task of energy management is to reduce costs for a provision of energy in households and residential building facilities without affecting the user's wellbeing. Functions of the home energy management are: controlling activation/deactivation of home appliances, collecting real-time energy consumption from smart meter and power consumption data from various household appliances, generating and monitoring a dashboard to provide feedback about power usage, providing control menus for controlling appliances and providing a universal link to the broadband Internet.

A few key features that apply to various energy efficiency driven Smart Homes are:

- An available node energy, which is frequently limited, i.e., battery supplied nodes, which work with limited amounts of energy.
- Smart devices and equipment, which can offer an opportunity to monitor and to remotely control key features within homes.
- Decision-support tools designed to assist users in making smarter decisions and based on getting the most out of benefits gained by end users when they use energy saving services. It becomes then necessary that at the same time with an energy management challenge, a proper communication protocol between smart devices would regularly improve the system performance.

Proposed energy efficiency driven smart home systems by the literature are based on task assignment, integration of various physical sensing information and control of different devices.

2.2 Renewable Energy Management Driven Smart Home

This concept of an energy management could include using of solar, wind and/or other renewable energy sources with an intelligent power consumption mechanism for electric appliances placed inside a house and a collaborative smart grid to ensure interconnections between them. A user or a system itself is capable of lowering energy consumption or postponing energy demanding operations concerning the present electricity price by managing household electrical features and under the condition of ensuring a positive comfort level.

2.3 Health Care Systems

Monitoring of a person's cognitive and physical health and an area of critical need is eldercare. Suggestive kinds of smart healthcare technologies contain simple devices (blood glucometers, oximeters, blood pressure monitors, etc.) which deliver standardised outputs for specific physiological conditions, smart applications or software able to analyse and process body signals, sensor integrated smart devices (gaming devices, smartphones, and pads), wearable sensors (e.g., wrist straps, T-Shirts) and additional devices entirely manufactured for the purpose of body signal monitoring/processing (e.g., mainframe computers, tablets). Healthcare IT infrastructures transfer sensitive patient health information and, as such, this issue faces several constraints and information security threats. Security safeguards and controls, data quality, and integrity are classified as the top priority, mostly because they arise by different fields of information security, but protecting a patient's location and purpose specification, remain the least addressed requirements.

2.4 Advanced Multimedia Services

Media consumption within a home has been growing over the years, and new forms of domestic entertainment are very popular. The main promoter for an evolution of future Home Area Media Networks (HAMNs) is an emergence of beyond High Definition (HD) media formats. These formats oblige far greater demands on networks for low latency, high-capacity and rigorous Quality-of-Service (QoS) in comparison to other existing formats. Furthermore, their data-intensiveness will require real-time interconnection of multiple, probably distributed, high-performance media processing and storage resources. Surveillance and Security System requires a robust configuration to collect meaningful, reliable, and accurate data. As the result, they do have a need for adequate support for the QoS required by the delay-sensitive and bandwidth-intensive multimedia data that they currently do not display. These restrictions can have important consequences for real-time surveillance or monitoring applications as they often lead to insufficient or improper measurements and erroneous event detection.

Chapter 3

Overview of Selected Protocols

There are a wide variety of technology platforms, or protocols, on which a smart home can be built. Choosing a proper smart home protocol is not easy. A communication protocol should support a large number of devices and offer the best possible device interoperability (an ability for devices to talk to each other). But there are also other factors to consider, such as power consumption [5]. An ideal smart home gadget would use a wireless transmitter and receiver that require very little power, so that devices could go for months, or even years, without needing a new battery. Their signals would pass through walls and floors inside and outside a home, yet without interfering with other wireless networks. Signals can be encrypted for security reasons, and a user would be able to add devices to the network easily [57]. Other important parameters are for example low bandwidth, wide area coverage and, of course, cost [5].

The overview of some of the most popular smart home protocols is introduced in the following sections.

3.1 Z-Wave

Z-Wave protocol is primarily developed for home control and monitoring, but it can cover quite large areas. It is mainly used for door or window sensors, thermostats and other home automation devices that are accessible through high-level applications or over the Web.

The protocol supports full mesh networks without the need for a coordinator node, and it is very scalable, enabling control of up to 232 devices [52]. The range is from 40 to 200 meters [12], and the data rate is 9.6, 40 or 100 kbps. It depends on regional frequency allocations and bandwidth requirements [56]. Two basic types of devices are supported: controllers and slaves [55]. It is based on ITU-T G.9959 standard which describes MAC and physical layer [52]. Z-Wave runs on 868.42 MHz (Europe) and 908.42 MHz (United States) frequency band [30].

The frequency band is much lower than the one used by most household wireless products (2.4 GHz) therefore it is not affected by their interference and "traffic jams". Moreover, the lower a communication frequency is, the better signal transmitted through walls and other obstacles is. It means that lower frequencies are more suitable for buildings [52]. Z-Wave also provides a simple system that customers can install themselves [25]. Another advantage of Z-Wave is its interoperability. Resulting Z-Wave products introduced today will work with Z-Wave products from a decade ago and with products in the future [52].

On the other hand, Z-Wave is not an open protocol. Following drawback is a dependency on only one manufacturer (Sigma Designs[1]). The next disadvantage is that Z-Wave uses different sub-GHz frequency bands in various parts of the world. Therefore it is required so that a customer develops and certifies different product versions for various regions in the world [18]. The high price of Z-Wave products is also considered as a drawback [57]. Moreover, the protocol assumes that devices are static. Therefore movable devices are not allowed to join a network [9].

Three types of frames are defined on MAC layer. A unicast frame is sent to only one destination device [28]. If the frame is delivered, a recipient responds with an acknowledgement frame which confirms delivery of the frame. However, it does not guarantee that the packet was delivered correctly. If a transmitter does not receive the acknowledgement frame, it will try to resend the frame up to three times. If none of these attempts is successful, it will report a failure message to the user [7]. The structure of unicast and acknowledgement frame is similar, but acknowledgement frame does not contain data [11]. The next type of frame is a multicast frame that is sent to multiple destination devices without acknowledgement. A broadcast frame is received by all devices in a network and is not confirmed by an acknowledgement packet [28].

3.1.1 Joining Process

A fully equipped home always manages from 50 to 80 devices and all of them should be included in one single wireless network. The process of joining a new device is called **inclusion**. Z-Wave network is built by a controller. Only one primary controller can be in a network and is responsible for it. If an IP gateway is available, then it takes over primary controller's duties. In case that an IP gateway is not available, any other controller can behave as the primary controller. When it is required, a functionality of the primary controller is given to a different controller. It is also possible that all inclusions are realised by a remote control as the primary controller and then the IP gateway manages all further operations instead of the primary controller. If a node is not within a direct range of the primary controller, routing can be used [54].

The inclusion starts when a controller is turned into an inclusion mode. Then a device sends its Node Information Frame (NIF) to confirm the inclusion. The NIF is a special frame which contents description of the network and application capabilities of the device. Different devices can have various capabilities. As the result of the inclusion, NodeID and HomeID are assigned to the device. The whole process is shown in Figure 3.1. NodeID identifies a device, and all nodes in a network have the same HomeID [74]. Controllers have HomeID preprogrammed but slaves' HomeID is initially set to zero, and they need to have HomeID assigned by a controller to communicate with a network [13]. Devices with different HomeID's cannot communicate with each other and within one network it is not allowed to have two devices with identical NodeID [73].

[1] http://www.sigmadesigns.com/

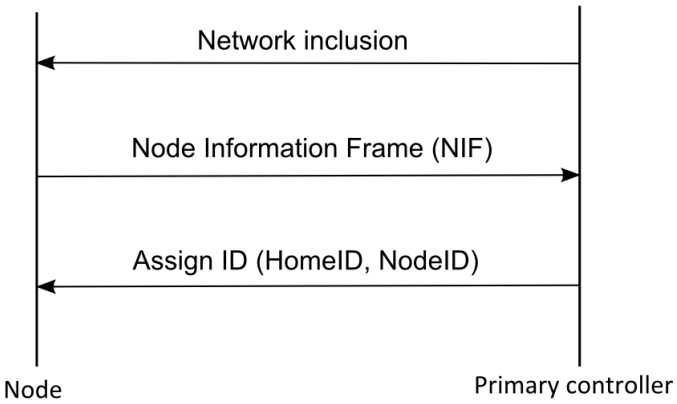


Figure 3.1: Z-Wave: Joining process (inclusion)

3.1.2 Routing Process

It is typical for a wireless network, that central controller has a direct wireless connection to other devices. A radio link has to be direct to enable it. However, if some problem occurs, for example, a device is out of a controller's range, the connection between devices is interrupted, or an obstacle appears, the controller has not any other possible route to communicate with the target device and communication will break. Z-Wave solves this problem by a **routing** mechanism. Devices can send on and repeat packets for nodes which are not in a controller's direct range. Thanks to this mechanism a very flexible and large network can be built. It is appropriate to make a compromise between network size and stability and maximum time for packet travelling within a network. Therefore packets can be routed via up to four repeating nodes.

Z-Wave uses a source routing mechanism. It means that a controller device sending a packet generates a complete route to the target destination through several devices. The route is placed in the frame, and every device that receives the frame with the route information resends it according to the frame content. The source routing mechanism allows implementation of a quite simple protocol with network information which is not distributed. The disadvantage of this mechanism is increasing the frame length since the route has to be included inside of a payload [53].

The routing can automatically adapt to any changes in a network. If a device joins the network, a controller requires actualized list of neighbouring devices from the newly added device so that it can update its routing table. In case that a new controller is added to the network, the primary controller sends it a snapshot of its routing table. If more nodes are added later, it is possible that a secondary controller does not have updated routing table while the routing table of the primary controller is always updated. In this case, it is necessary to update secondary controller's routing table manually. If devices are excluded from the network, the corresponding entries in routing table are deleted. In case that the secondary controller is excluded, it will delete its old HomeID and also its old routing table [10].

All nodes can recognise which nodes are their neighbours. A neighbour is a device in direct wireless range of a node. In some cases, the node can send a list of its neighbours to a controller that can use it to create a routing table. All information about possible communication routes within a network is stored in this table. At first, the controller always tries to transmit a packet directly to the target device. If this is not possible, it

will utilise its routing table to determine the next best way to the target. Up to three alternative routes can be chosen and used for sending the packet. If none of the transfers via three routes is successful, the controller will signalise a failure by an error message. Delivery of the packet is successful if a sending device receives an acknowledgement.

A device type called a routing slave is associated with routing process. It provides advanced routing capabilities [7].

3.1.3 Network Reinitialization

Z-Wave network is able to detect device movements and update routing table automatically in some cases. **Auto-healing** of a network is possible if following requirements are fulfilled. A controller with additional functions called a Static Update Controller (SUC) must be present in the network [73]. It receives updated routing table from the primary controller and provides it to all other controllers in the network. The next requirement puts emphasis on a device type that was moved. This device must be a routing slave. At least one routing slave must be in the range of the device which was moved to a new position. The moved node must recognise it was moved. It means that it must be able to send out an unsolicited packet.

The SUC can determine a new position of a slave and also update routing table accordingly. The process is called Get Lost Algorithm in Z-Wave terms. It is possible only for routing slaves because they can send unsolicited packets. If an unsolicited packet sent from a routing slave fails, this routing slave will come to a conclusion that its routing table is not longer valid. Therefore it sends out a broadcast packet called cry for help that is able to auto-heal a network. A device that received this unsolicited packet knows that a sender can not communicate successfully with another node in the network. If this device is also a routing slave and has information how to route the packet to reach the SUC, it will send on the cry for help message to the SUC. The SUC can actualize its routing table and assign new routes to the lost device by realising the same steps as during the inclusion [4].

If a slave is moved and communication with it is not possible, it will be marked as a failed node by a controller. This node will be also added into a so-called failed node list which contains nodes with failed communication. These nodes can be removed from the network on user request. Network rebuilding places demands on the number of packets, and this is a reason why this process is not done automatically as soon as the failed node is detected. The controller scans the whole network so that it finds the moved node. It asks every known node to actualize its list of neighbours. If the moved node is in the range of at least one node, it will be detected. In this case, the controller will update its routing table, and the removed node will be moved back to the routing table. In case that a device is battery powered, it is mostly in an energy savings mode and is woken up from time to time. Therefore during a network scan, maximum timeout is set to wait for any life signal from the battery powered device.

Controllers can always find a proper route to the destination device because they know the whole network topology. Distinguishing two types of controllers is possible. A static controller should be placed in a fixed position in a network and should not be moved. It is mains powered and can route packets. If it is moved, a network reorganisation or a network scan is demanded. On the contrary, a portable controller is supposed to change its position and therefore it is typically battery powered. It does not route packets because it is mostly in a sleep mode. It always tries to communicate with nodes in wireless range.

If a problem appears, it will try to generate a temporary routing table to find a possible route to the target device [10].

3.2 ZigBee

The ZigBee protocol is designed to provide high data throughput in applications where the duty cycle is low (device is not frequently operating). It is often used in industrial automation and physical plant operation. It is also associated with machine-to-machine (M2M) communication and the Internet of Things (IoT) [14].

The protocol offers three types of topologies. Star topology which is the simplest and the most limited, tree topology and mesh topology that is one of the most flexible [48]. A network is capable of accepting about 65 535 nodes thanks to short addresses [72]. The range is from 70 to 400 meters [16], it depends on power output and environment characteristics. The data rate is 20, 40 or 250 kbps. The maximum data rate is available at 2.4 GHz frequency band [26]. According to a device's role, ZigBee defines three types of devices: coordinators, routers and end devices. ZigBee technology builds on IEEE standard 802.15.4 which defines physical and MAC layer [60]. It can operate in 868 MHz (Europe), 915 MHz (United States) or 2.4 GHz (global) frequency band [26].

ZigBee network can contain a lot of nodes; therefore, large area coverage is supported. The price of the ZigBee devices is lower than Z-Wave devices [25]. This protocol offers higher maximum data rate than Z-Wave protocol [48].

The main disadvantage is interference caused by using the same frequency band of most household wireless products at the same time. The ZigBee's range is lower than Z-Wave's [30].

Four MAC frame structures are defined by the IEEE 802.15.4. A coordinator uses beacon frame to transmit beacons that are used to keep nodes synchronised within a network. The beacon frame also wakes up devices that check whether some packet is addressed to them. If they are not addressed, they go to sleep. The next frame type is MAC command frame used for remote control and configuration of devices. A successful delivery of a frame is confirmed by an acknowledgement frame. All data transfers are realised using a data frame. The unicast, multicast and broadcast are also supported [65].

3.2.1 Joining Process

Before any ZigBee node may communicate on a network, it must create a new network or join an existing network. New devices are added to a network by the process called **commissioning** [31]. A coordinator is responsible for choosing a channel, PAN ID, security policy and stack profile for a network. A network can be created only by the coordinator; therefore each network must have one coordinator. Routers and end devices may join a network [17].

Every node has unique 64-bit IEEE address (MAC address) assigned by the OEM (Original Equipment Manufacturer) during manufacturing. During the commissioning, each node gains 16-bit short address (NwkAddr) that is unique within that network and is used for nearly all communications. It leads to a reduction of over-the-air protocol overhead and it offers more space for application payload [31].

At first, the coordinator must choose an appropriate channel for the network to operate on it. This is the reason why it executes an energy scan on channels. Detecting energy level on each channel is necessary. If a channel with excessive energy level is detected, it is

deleted from coordinator's possible channels to start on. When the energy scan is finished, the coordinator surveys its list of possible channels. It is typically called as an active scan or a PAN scan. It means that the coordinator sends a beacon request transmission on each possible channel. All near coordinators or routers that have already joined the network will send the beacon back to the coordinator as it is shown in Figure 3.2. This beacon contains quite a bit of information about the ZigBee network, including PAN ID, extended PAN ID, join enable and whether a node has a capacity for router or end device which wants to join [17]. After completion the PAN scan, the joining device chooses a random channel and 16-bit PAN ID which is not used to start on [17].

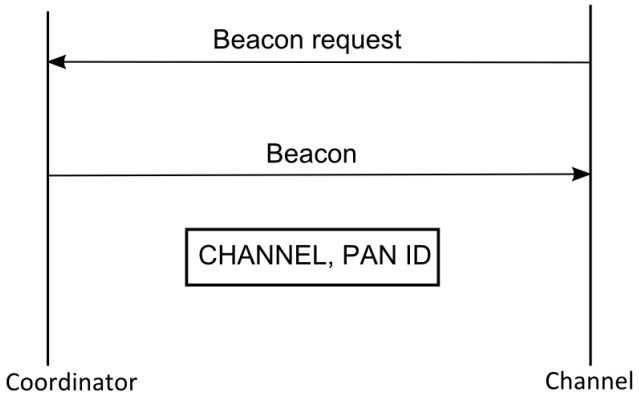


Figure 3.2: ZigBee: Joining process (commissioning) – forming a network

Devices which have not been added to the network yet capture by accident the beacon or try to find out a network by broadcasting a beacon request on available channels. This process is not realised when radio channel has been preconfigured or determined in the application profile. In case that the network was created on one of the channels by the coordinator, it reacts to the beacon request by broadcasting a beacon which contents 16-bit PAN ID of the network, the coordinator's address in short 16-bit format or extended 64-bit format, optionally extended PAN ID (EPID) and the ZigBee stack profile supported by the network. Moreover, the beacon includes a flag indicating whether the responding node has capacity for routers or end devices joining as new children and also a device depth of the sending device – it means its level in the tree rooted at the coordinator. Any routers that have already joined the network will also react with the beacon if they detect the beacon request.

After that, the new node sends an association request command to a particular parent node's address to which it wants to join. For example, the node must join a node whose device depth is the smallest. Extended 64-bit address of the joining node is included in the association request as a source address. The parent node confirms the received association request by sending an acknowledgement. If it accepts the association request, it sends an association response command to the extended address of the node. The 16-bit short address that the joining device should use in the future is defined in this association response. The joining device acknowledges receiving of the association response. In case that the joining device is in a sleep mode, the association response is stored and sent when the sleeping node asks about it. After confirmation of association, a data request command is usually sent by the device which is addressed using assigned short 16-bit address. Its parent sends pending configuration data as a reply. Battery-powered end devices go to

sleep after receiving the packet. They will be woken up according to their next scheduled wake-up time or interrupt. After successful join, the device sends a device announcement message that informs other nodes in the network about its presence in the network, 16-bit short address and extended 64-bit address. The whole process of joining a device is presented in Figure 3.3 [34].

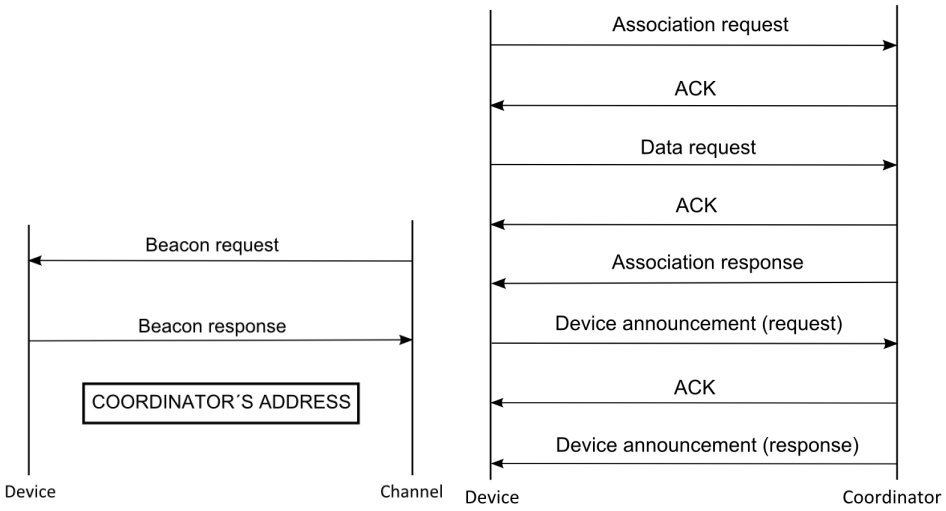


Figure 3.3: ZigBee: Joining process (commissioning) – joining a device

A network is defined with a unique PAN ID. All devices in the network have the same PAN ID. Devices are preconfigured with PAN ID to which they can join, or they can select the PAN ID during searching for near networks. Routers and end devices that join a specific node, not a network in general, they use 64-bit MAC addresses for source and destination addresses of a MAC association request.

After an active scan is finished and a suitable parent is selected, an authentication procedure starts in secured networks. A network security key for the network must also be selected by the coordinator and can be set by a special write-only command. Two possibilities are presented, a random network key will be selected or the network security key will use a value specified by a network key. Routers and end devices that have enabled security obtain the network key when they join the network. Transmission of this key is encrypted. If the authentication is not successful, the device that wants to join the network will not be added to the network. The authentication gives the network a chance to reject the node [17], [31].

3.2.2 Routing Process

ZigBee uses mainly two types of **routing** algorithms, a hierarchical tree routing algorithm and also an Ad Hoc On-Demand Distance Vector (AODV) algorithm [37]. Tree routing is a clever and simple way how to route packets. It is not too complex as the AODV and it does not require routing tables. This type of routing is very suitable for networks where devices, for example, can allocate only limited space in memory. On the other hand, it is not reliable because of static addresses. If network structure is changed, devices can rejoin the network, but this process can be very demanding. Another disadvantage is that coordinator is a single point of failure for this routing method because it is impossible to recover the network if coordinator fails. The AODV routing algorithm creates routes

between nodes only when source nodes request them. Thanks to it the network is flexible and allows nodes to enter or leave the network.

Tree routing uses addresses of routers in the network to route packets. It is necessary to calculate the size of address space so that it is clear which addresses can be assigned to devices. Routers' addresses must be allocated in a special way. It means that they are fixed for position in a tree where the device is placed. Addresses move upwards from top to bottom and from left to right in the tree. During routing process, packets can be routed only in two possible directions, up to the top of the tree or down the tree. It depends on the position of the target device. If the destination address is in device's address space, it should be routed down. Otherwise, it will be routed up [19], [61].

The AODV routing algorithm uses routing tables that store destination address, next hop to reach the destination node, destination sequence number and lifetime. Sequence number helps to avoid forwarding the same packet more than once. Lifetime is updated every time the route is used if it is not used within the lifetime, the route expires. Each node in the network contains this table. If a next hop is not found, route discovery must be performed to find another route. When a source node must find out a path to the destination node, it sends a broadcast route request command that contains source and destination network address and field with a path cost. This path cost is a metric for determining the quality of a route. Each node receiving the route request command broadcasts the packet again. It also updates the path cost field and makes a provisional entry in its route discovery table. If the packet is delivered to the target device, the field with a path cost is compared with previously delivered path cost field on the target node. If the path cost of a newly received packet is better than previous one, the target node will respond by sending a route reply packet to a node that sent the route request first. Delivery of the packet is ensured by intermediate nodes that receive and send on the route reply packet towards the source node. After that, the source node can send a packet to the destination node [2], [15].

3.2.3 Network Reinitialization

The network must also solve **device movement**. When a node that routes a packet to the target node is unable to forward the packet to the destination device, it sends a route error message. If a source node receives a route error message, it initiates a new route discovery to reach the destination node. The link failure can also be detected using hello messages which are periodically exchanged between neighbouring nodes. If a hello message is not delivered, it means that the link failed. Unsuccessful delivery of several acknowledgements on MAC layer may also be used to detect the link failure [2].

3.3 Thread

Thread is a new protocol, first announced in July 2014. Its aim is to provide reliable, low-power, secure and scalable networking solution for connecting of hundreds of products around home to the IoT.

Thread protocol supports mesh and star topology. It depends on the number of routers in a network. If there is only one router or border router, then star topology with a single router is created. A network can handle 250–300 devices [57], [20]. The range is sufficient to cover typical home, and the data rate is 250 kbps. It specifies four types of devices: border routers, routers, router-eligible end devices and sleepy end devices. The protocol runs

on IEEE standard 802.15.4 that defines physical and MAC layer. Thread uses the 6LoWPAN protocol. It means that the network is IP-based and devices can connect to each other and also directly to the Internet. The protocol operates in 2.4 GHz frequency band [57], [68].

IP addresses are assigned to all devices on a network. The Thread protocol provides device-to-device application without the need for an application gateway; this leads to the elimination of a single point of failure [48]. Device authorisation is required before joining a network. All communications are secure and encrypted. Another advantage is, that application layer is not defined, devices are able to talk to multiple applications [21].

A frequency band is the same as the one used by most household wireless products; therefore, interference can occur [48]. Moreover, the protocol aims at home automation with many devices which also causes its complexity. It is a new protocol that needs to be self-established [3].

The User Datagram Protocol (UDP) is used for packet transfer. It is not too complicated as the Transport Control Protocol (TCP) that supports, for example, error checking and retransmissions. This approach makes transmissions quicker and more efficient. Therefore the UDP is suitable for battery-powered and resource-limited devices. The Constrained Application Protocol (CoAP) is used to limit some restrictions of the UDP. Thanks to these two lightweight protocols it is possible to query IoT devices directly from a browser. Unicast, multicast and broadcast can also be used for data transfer [8].

3.3.1 Joining Process

A **commissioning process** consists of petitioning and joining. Only one authorised commissioner that authenticates following devices must be present in a network, therefore petitioning takes place before any device can join. Two types of commissioner exist. An external commissioner is a device that uses a WLAN (Wireless Local Area Network) interface for commissioning purposes. If it wants to become a sole authorised commissioner, it must ask the Thread network. A representative (a border router) will be used for communication with a leader. First of all, the commissioner must prove that it is eligible to become the sole authorised commissioner and establish a secure commissioning session. For this purpose, it must use an authentication handshake with the border router. Then it petitions the leader through the border router because only one authorised commissioner can exist. If petitioning is successful, it becomes the sole authorised external commissioner. The secure commissioning session is not cancelled, the border router will be made known throughout the network, and next communication with other devices will be realised using the border router. A periodic keep-alive message is sent on the secure commissioning session to assure it is still open.

The other type of commissioner is a native commissioner that uses a Thread network interface for commissioning purposes. The petitioning is similar to the presented one, but some differences appear. The native commissioner communicates with the network through representative called a commissioner router. If petitioning is successful, the commissioner joins the network and becomes an active device. All communication with other devices is realised directly with it.

When an authorised commissioner is associated with the Thread network, it is possible to join other eligible devices. Before they can actively communicate within the network, they are called as joiners. One of the various scenarios must be realised to join a device. They depend on a system topology. The DTLS (Data Transport Layer Security) handshake

between joiner and commissioner must occur in each of them, but they differ in the number of devices which are used for realization of the joining process and also in the level of authentication. If traffic is authenticated, the established commissioning session is used [67].

3.3.2 Routing Process

Thread network can usually contain up to 32 active routers. Routers use next-hop **routing** for packets that is based on a device routing table. All routers have connectivity and current paths for other routers in the network because the device routing table is maintained by the stack. The Routing Information Protocol (RIP) algorithm is used for routing packets.

The Message Link Establishment (MLE) messages are used for creating and configuring secure radio links, discovering neighbouring devices and maintaining costs of routing between devices in a network. All routers periodically exchange single-hop MLE advertisements messages that contain information about link cost to all neighbouring routers and also path costs to all other routers within the network. If a router receives the MLE message, it stores next hop information which is not sent in the advertisement. On-demand route discovery is not used because all routers have current path cost information to any other router. It is an advantage as route discovery requests flood the network and require costly network overhead. The quality of a link in each direction is built on the link cost of incoming packets from the neighbouring device. The incoming link cost is mapped to the link quality. The link cost is determined using results from measurement of RSSI (Received Signal Strength Indicator). The path cost to any other node in the network can be counted as minimum sum of link costs to reach a particular node. Routers observe costs, and if a change of radio link quality or network topology occurs, they send out new costs within the network using periodic MLE advertisement messages. Bi-directional link quality between two devices determines routing cost.

Firstly a router's table contains only costs to single-hop neighbours. When routers start noticing advertisements from neighbours that contain costs to other routers, they also store multi-hop path costs. Then they spread them to ensure that all routers have information of connectivity. If a received MLE advertisement message has already been in a neighbour table, the incoming cost from the neighbour can cause entry update. Otherwise, a new entry is created. Updated routing information for other routers is also included in the MLE advertisement, and it is used for updating the device routing table.

If a packet is intended for a child device, routing is realised by looking at higher bits of its address to determine an address of the parent router. In case that the device knows its parent router, it has path cost information and next-hop routing information for that device. IP routing is used for sending on of packets by devices. The device routing table contains compressed form of the mesh local ULA (Unique Local Address) address for routers and the proper next hop. Distance vector routing ensures that routes to router addresses will be found. If the routing is performed, router address of the destination router is defined by the upper 6 bits of a 16-bit address. The lower bits of a 16-bit destination address serve to reach the destination node. In case that this part of the address is filled with zeros, the final destination was reached. A leader assigns and manages router addresses. Thanks to the fact that the leader and other routers have the same information, a situation when the leader becomes unreachable can be solved. In this case, another router is autonomously elected and becomes the leader without user intervention.

3.3.3 Network Reinitialization

If the destination becomes unreachable due to unavailable route, for instance, a device was dropped off a network; routers can quickly calculate the best path to maintaining connectivity to all other devices in the network. It is called as **self-healing** routing mechanism [68].

Chapter 4

Protocol Design

Typical smart home protocols, specifically Z-Wave, ZigBee and Thread protocol, were carefully studied and several drawbacks were found in each of these protocols. Disadvantages of Z-Wave are the high price of products and the fact that the protocol is not open source. The main drawback of ZigBee and Thread protocol is a possible interference caused by using the same frequency band of most household wireless products at the same time. Moreover, they are too complicated. These disadvantages imply that none of the existing wireless communication protocols is entirely suitable for the smart home. Therefore we have designed new open source FIT protocol that eliminates all presented problems. Moreover, the new protocol should use low data transfers in order to achieve low-energy consumption.

4.1 Protocol Requirements

Based on the analysis of other smart home protocols, following requirements were defined.

1. The protocol has to be supported on various types of end devices: sensors that gather information about their environment and actuators interacting with them. Both usually have memory limited processors. Generally, end devices use processors with very limited hardware resources. Therefore, the protocol design has to take this limitation into account.

2. In a household, sensors and actuators are supposed to be connected through one coordinator node (PAN coordinator) to the Internet. PAN coordinator has sufficient hardware equipment to collect and store all information about the network; therefore it can provide information to other devices (such as mobile phones, tablets, etc.). The range of wireless network can be gradually increased via special elements placed within a network.

3. Another requirement of the newly designed protocol is to ensure low energy consumption. As the result, a small amount of data has to be transferred not very frequently.

4. The protocol has to enable easy connecting or disconnecting of nodes to provide easy network scalability up to hundreds of nodes.

5. All source codes have to be open source to make modifications and compilation on any platform possible. It implies that the source code should be multi-platform.

4.2 FIT Protocol

The FIT protocol is the new wireless communication protocol which is designed specifically for IoT. It fulfils all smart home requirements: low-cost, low-power consumption, open source and low-frequency band. Benefits of an open standard are application independence, platform independence, long-term access and architectural integrity. The other characteristics are reliability, security and easy portability [48].

The protocol uses tree topology which can contain up to 64 coordinators (including PAN coordinator) and up to 2^{32} end devices. The node range is from 10 to 100 meters, and the data rate is 200 kbps. Three types of devices can be present in a network: end device, coordinator, and PAN coordinator. It is not based on any standard, and it runs on 860–870 MHz, 902–928 MHz and 950–960 MHz frequency band [48].

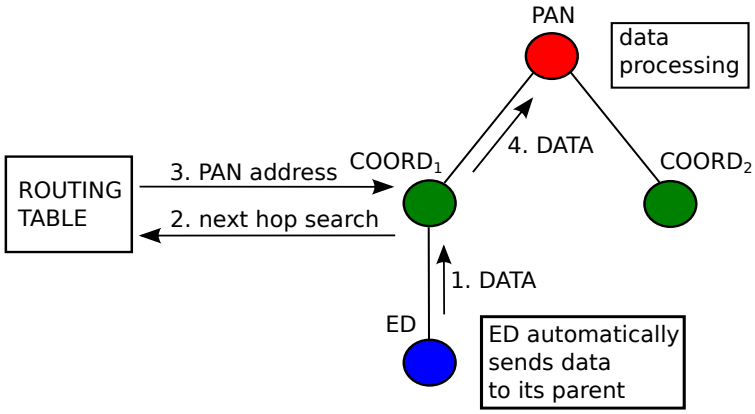


Figure 4.1: Data transaction from end device to PAN coordinator

Figure 4.1 shows simple data transaction from end device to PAN coordinator within tree topology.

1. The end device automatically sends data to its parent that has to ensure data delivery to the destination device.
2. The parent searches its routing table to determine the next coordinator address, i.e. the next hop address.
3. It obtains the next hop address (PAN address in this case).
4. Finally, the PAN coordinator receives and processes received data.

The FIT protocol defines various types of communication. Reliable delivery (using four-way handshake) provides reliable data transfer including data retransmission in case of communication failure. Unreliable delivery (using packet without acknowledgement) ensures a bandwidth rather than reliable delivery. As a consequence, a large amount of data can be transferred.

Advantages of the FIT protocol are simple implementation and minimal requirements to hardware resources. The FIT protocol is highly power efficient because devices mostly operate in sleep mode and are not often active. As the result, batteries do not have to be changed for a long time. The FIT protocol is open source. Radio modules can be changed (both frequency and power), and it is possible to reach higher data rate and longer range if

necessary. One of the following advantages is the usage of lower bands. As a consequence, the signal does not interfere with Wi-Fi signals, and it is better transmitted through walls and other obstacles. Moreover, the FIT protocol is not influenced by any company and its acceptance will not face any problem [48].

The main disadvantage is that the protocol is not designed for general purpose. For example, security and encryption are highly important in healthcare due to private data transfer [23]. Big data transfer is required in fire security systems to analyse and detect fire thread [38]. Long distance wireless sensor networks are desirable to cover a large geographic area. For instance, farming or water quality monitoring requires sensor monitoring over long distances [75]. The presented use cases are not aimed in the FIT protocol, but thanks to open source codes, it is possible to implement any required features. The FIT protocol is very specialised for the solution of smart home communication. Therefore its attention can be focused on all needs of a target user group.

4.3 Types of Communication

Communication is realised among all types of devices, namely PAN coordinator, coordinator and end device. Only end devices cannot exchange packets between themselves. The typical role of PAN coordinator is to control all devices within a network, including packet processing and routing. The network can include only one PAN coordinator. Coordinator extends the range of the network and it serves mainly as a router that sends a packet to the destination. It allows more devices to join the network. The last type of device is end device. Two types of end devices can be distinguished: sensors and actuators. Unlike sensors that can measure values of quantities, actuators help to perform certain actions thanks to the opportunity to change its value. An example of the actuator is a light bulb where luminosity can be set. Another actuator is a socket which can be switched on or off.

Link layer ensures data transfer between two neighbouring devices. Change of source and destination address on link layer depends on packet route. On the contrary, network layer is responsible for packet routing and source and destination address does not differ.

Possibilities of addressing depend on device type. End device can be standalone, or it can be connected to coordinator or PAN coordinator. It can be addressed only by its end device identifier (EDID). In this case, a packet has to be routed through PAN coordinator that knows all information about network topology (including parents of all devices), in contrast to coordinators. A coordinator can be addressed using the coordinator identifier (CID). A packet does not have to be routed through PAN coordinator. Packet route depends on the position of the destination device in the network. If source and destination device are placed in different subtrees of PAN coordinator, the packet has to be routed through it. PAN coordinator can also be addressed using CID.

4.3.1 Reliable Data Delivery

If an actuator is set, the corresponding action should be performed immediately. Therefore reliable packet transfer using four-way handshake is necessary. When a packet cannot be delivered, upper layers are informed about it, and network reinitialization begins.

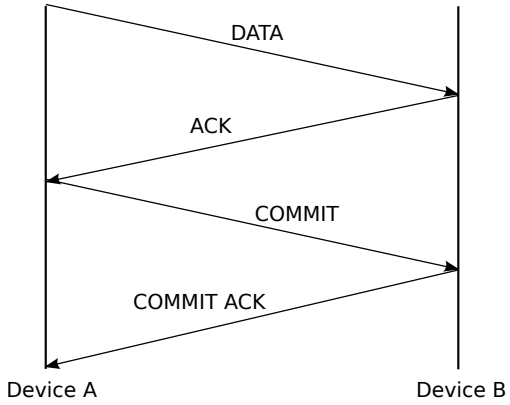


Figure 4.2: Four-way handshake communication

Figure 4.2 shows communication using the four-way handshake. Every packet has a specific function. `DATA` packet is sent by the Device A to initiate communication. This type of packet contains data appointed to the specific device (Device B). Timeout[1] for `ACK` packet delivery is set on the sender (Device A), then it waits for `ACK` packet. If given time is exceeded, the packet is considered to be lost, and the sender (Device A) tries to resend the packet four times maximally. If the receiver (Device B) receives `DATA` packet correctly, it sends `ACK` packet back to the sender (Device A). `ACK` packet acknowledges receipt of data. The Device A then sends `COMMIT` packet to the Device B. It confirms successful completion of data transfer. The sender (Device A) is waiting for `COMMIT ACK` packet unless timeout is expired. If it occurs, the Device A tries to resend the packet four times maximally. The Device B sends `COMMIT ACK` packet to the Device A to acknowledge that transfer is successfully finished. `ACK`, `COMMIT` and `COMMIT ACK` packets do not contain any information apart from a header, so that packet size is the smallest possible to achieve minimal power consumption.

In some cases, a device may be unable to receive a packet, e.g., because of full storage. It can be caused by a long-lasting packet operation in higher layers of the protocol. Therefore a relevant packet is sent back as if the request was successfully processed, however, a header is set to signify busyness. In this case, the timer is set on the sender (Device A) to greater value because packet processing can take a longer time in comparison with non-full storage. A repeated timer setting signifies that it is not a fault in the network but only busyness of a specific device.

4.3.2 Unreliable Data Delivery

With low data rates and high noise levels, it is not suitable to wait for an acknowledgement for every transferred packet. Much better solution is transferring data as a bulk without waiting for `ACK` packet. It is desirable to ensure better performance regarding higher throughput so that communication was more effective and more power efficient. On the other hand, it is not guaranteed that packets were correctly delivered. For example, sensors sending temperature values periodically, use this type of communication, because in this case, data loss is not critical.

[1] Timeout for `ACK` packet is predefined, it is not included in the packet.

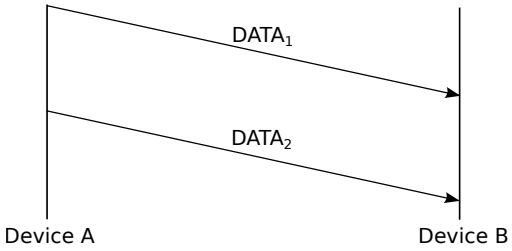


Figure 4.3: Communication without waiting for ACK

Figure 4.3 represents sending of two `DATA` packets from the Device A to the Device B using this transfer type.

4.3.3 Control Packets

If communication among devices is not possible, then it is necessary to reinitialize the network. **Broadcast packets** are used for fast network rebuilding. In this case, it is essential to use a special destination address in the packet header.

One of the methods how to prolong battery lifetime is to use **sleepy packets**. If end device is in sleep mode and some packet is addressed to it, the packet is stored only on PAN coordinator. As soon as end device asks for it, PAN coordinator sends the packet to end device.

4.4 Joining Process

The aim of joining process is adding new devices to a network. A device that tries to join a network finds all available networks within its range. Receipt of acknowledgement signifies availability of a network.

After correct exchange of the join packets, the device is added to the network and PAN coordinator updates information in the table of devices. The newly added device obtains an identifier of network, parent and coordinator. Selection of parent identifier is based on the best-received signal strength. PAN coordinator can also refuse the device that intends to join the network. In case of an unsuccessful attempt to join the network, the device searches for other available networks.

Joining process can be realised between end device and PAN coordinator or coordinator and PAN coordinator. The most important packets in joining process are `JOIN REQUEST`, `ACK JOIN REQUEST` and `JOIN RESPONSE`. Moreover, two additional packets `JOIN REQUEST ROUTE` and `JOIN RESPONSE ROUTE` are used in the case that joining process is realised indirectly through several coordinators.

The following steps describe actions that have to be performed during joining process.

1. PAN coordinator and coordinators have to be in pair mode, otherwise, `JOIN REQUEST` packet is ignored.

2. End device provides a periodic channel scanning by sending `JOIN REQUEST` packet. The periodic channel scanning is an essential procedure to discover available access point in the vicinity quickly.

3. If some coordinator or PAN coordinator is listening on a channel, it answers to end device with `ACK JOIN REQUEST` packet.

4. Each coordinator which is situated on the way to PAN coordinator resends `JOIN REQUEST` packet signed as `JOIN REQUEST ROUTE` packet to the next coordinator until this packet arrives at PAN coordinator.

5. If end device receives `ACK JOIN REQUEST` packet, it stops channel scanning and waits on the channel for `JOIN RESPONSE` packet.

6. PAN coordinator selects a parent of end device according to the best-received signal strength. It also stores information about end device to table of devices (it is necessary to verify if this end device is not in the table), then PAN coordinator answers with:

   (a) `JOIN RESPONSE` packet if it is selected as a parent of end device (PAN coordinator – end device),

   (b) `JOIN RESPONSE ROUTE` packet otherwise (PAN coordinator – coordinators – end device), this packet is routed back to the parent of end device which changes type of `JOIN RESPONSE ROUTE` to `JOIN RESPONSE` packet and sends it to end device.

7. End device gains its network identifier, parent identifier and coordinator identifier that is set to zero in case of end device.

Joining process successfully ends on condition that this process is completed before given timeout.

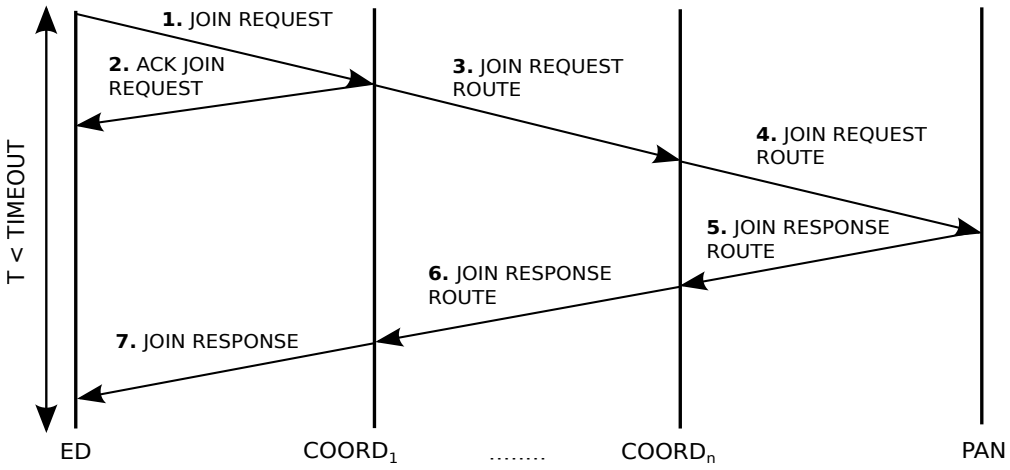


Figure 4.4: General packet sequence in joining process

Figure 4.4 shows the packet sequence that has to be sent during joining process before timeout is expired.

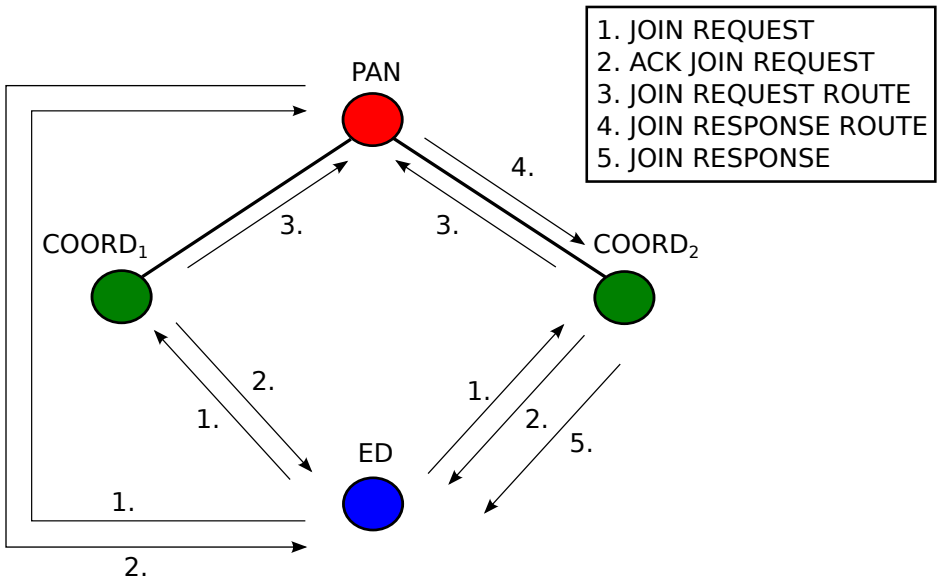


Figure 4.5: Joining process in tree topology

The particular example representing joining process within tree topology is shown in Figure 4.5. The numbers represent order of packets.

Device type is evaluated by PAN coordinator according to one byte in `JOIN REQUEST` packet. Three device types can be recognised: ready end device, sleepy end device and coordinator. However, joining process for separate end device is the same as joining process for end device connected to coordinator.

To sum up, when a new device attempts to join a network, the network identifier is unknown. In case that the new device is accepted to the network, it obtains `JOIN RESPONSE` packet with the network identifier. Then the device can start to communicate, and the network identifier is always checked during packet processing.

The design of joining process reflects a need for maximum efficiency. Moreover, it enables to add new devices which are not in direct range of PAN coordinator.

4.5 Routing Process

The basic task of the routing process is to enable packet transmission between PAN coordinator and end device. The main task is to determine next hop for any outgoing packet. Decision about packet route is made on each particular node that lies on a route between source and destination device.

The FIT protocol is inspired by the ISO/OSI (International Organization for Standardization/Open Systems Interconnection) reference model. Its layers are (ascending order): hardware, physical, link, network, and application layer (see Figure 4.6). Unlike link layer supporting communication between neighbouring devices, network layer realises packet routing. It implies that routing process is performed on network layer.

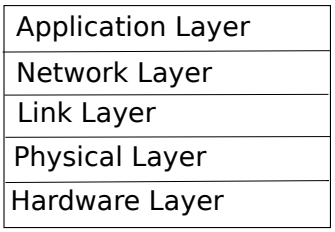


Figure 4.6: FIT protocol structure

A device that controls the whole network has to keep information about devices within the network so that packets could be routed. If some device is added, removed or moved, corresponding information is updated. The controller also distributes relevant information to its neighbours that spread chosen information further to the network.

Routing algorithm uses routing tree which contains information about network topology of all devices and is stored on PAN coordinator. Each coordinator stores information about network topology of its descendants (routing subtree). The distribution of the routing subtree is initiated by PAN coordinator which performs broadcast of this subtree to neighbouring coordinators. These coordinators broadcast corresponding routing subtree again to child coordinators until all coordinators have correct information about network topology.

Routing tree can be represented by a table that contains two columns: coordinator identifier and its parent identifier. The table is stored in an array row by row. The length of coordinator identifier is 6 bits. Therefore it is possible to address up to $2^6 = 64$ coordinators (including PAN coordinator). Each coordinator has to know its parent identifier. It implies that maximum size of the array is 128 bytes. All items in this array are initialized to the value 255.

During routing table distribution, the routing table is segmented into several packets of length 40 bytes. The number of packets depends on a size of the routing table. Important part of each packet sent in the course of distribution of routing table is described in Table 4.1. The first byte of the packet contains a total number of packets and order of an actual packet. The routing table is sent in remaining bytes of the packet.

Packet segmentation is required because it is not suitable to send too long packets in a network. If an error occurs, a lot of data could be lost. Moreover, the probability of jammed packet by external influences increases with packet length.

...	**INFO BYTE**		**DATA**		**DATA**	
...	4 b	4 b	8 b	8 b	8 b	8 b
...	COUNT	ORDER	CID	PARENT CID	CID	PARENT CID

Table 4.1: Part of packet sent during distribution of routing tree

Meaning of fields in Table 4.1 is:

- COUNT [4 b] – number of packets,
- ORDER [4 b] – order of an actual packet,
- CID [8 b] – identifier of coordinator,
- PARENT CID [8 b] – identifier of coordinator parent.

The following steps describe routing subtree counting algorithm. Let us denote the coordinator for which the routing table is created as $COORD_2$ and its parent as $COORD_1$. The routing table of the coordinator $COORD_2$ is denoted as $COORD_2$ RT (Coordinator Routing Table). Routing subtree counting algorithm is performed by its parent $COORD_1$.

1. It is necessary to loop through the whole routing table stored on the $COORD_1$ to fill the coordinator routing table on the $COORD_2$.
2. If the coordinator is joined to the PAN coordinator, no record is stored into $COORD_2$ RT on the position corresponding to the coordinator.
3. If the coordinator is a direct descendant of the $COORD_2$, the identifier of the direct descendant and the $COORD_2$ are stored into the coordinator routing table.
4. Otherwise, the reverse loop starts to find if the current coordinator identifier is located in the subtree of the $COORD_2$.

The particular example of routing subtree counting algorithm is described and shown in Tables 4.2 – 4.15. The PAN coordinator and five coordinators form the tree topology that is presented in Figure 4.7.

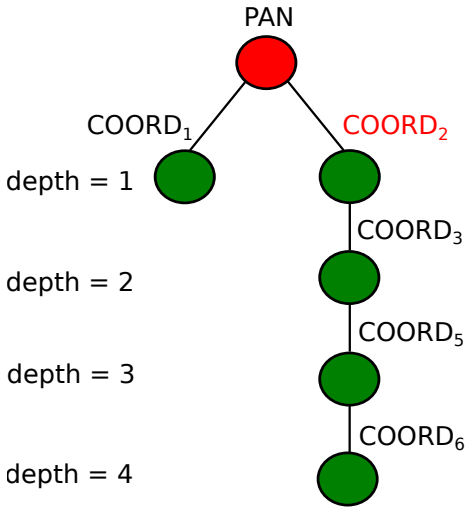


Figure 4.7: Sample tree topology

CID	PARENT CID
0	0
1	0
2	0
3	2
4	255
5	3
6	5

Table 4.2: PAN RT – step 1

CID	PARENT CID
0	255
1	255
2	255
3	255
4	255
5	255
6	255

Table 4.3: $COORD_2$ RT – step 1

CID	PARENT CID
0	0
1	0
2	0
3	2
4	255
5	3
6	5

Table 4.4: PAN RT – step 2

CID	PARENT CID
0	255
1	255
2	255
3	255
4	255
5	255
6	255

Table 4.5: $COORD_2$ RT – step 2

CID	PARENT CID
0	0
1	0
2	0
3	2
4	255
5	3
6	5

Table 4.6: PAN RT – step 3

CID	PARENT CID
0	255
1	255
2	255
3	255
4	255
5	255
6	255

Table 4.7: $COORD_2$ RT – step 3

CID	PARENT CID
0	0
1	0
2	0
3	2
4	255
5	3
6	5

Table 4.8: PAN RT – step 4

CID	PARENT CID
0	255
1	255
2	255
3	2
4	255
5	255
6	255

Table 4.9: $COORD_2$ RT – step 4

CID	PARENT CID
0	0
1	0
2	0
3	2
4	255
5	3
6	5

Table 4.10: PAN RT – step 5

CID	PARENT CID
0	255
1	255
2	255
3	2
4	255
5	255
6	255

Table 4.11: $COORD_2$ RT – step 5

CID	PARENT CID
0	0
1	0
2	0
3	2
4	255
5	3
6	5

Table 4.12: PAN RT – step 6

CID	PARENT CID
0	255
1	255
2	255
3	2
4	255
5	3
6	255

Table 4.13: $COORD_2$ RT – step 6

CID	PARENT CID
0	0
1	0
2	0
3	2
4	255
5	3
6	5

Table 4.14: PAN RT – step 7

CID	PARENT CID
0	255
1	255
2	255
3	2
4	255
5	3
6	5

Table 4.15: $COORD_2$ RT – step 7

The routing table is created for the coordinator $COORD_2$ (marked in red colour, see Figure 4.7). The algorithm is performed by the PAN coordinator (the parent of the $COORD_2$). Let us denote the PAN routing table as PAN RT and the coordinator routing table as $COORD_2$ RT. The routing tables in the example are shortened for demonstrative purposes. Each record of the routing table contains coordinator identifier (denoted as CID) and its parent identifier (denoted as PARENT CID).

1. It is necessary to loop through PAN RT and to check PARENT CID.
2. If current PARENT CID equals zero, it means that depth of the current CID is the same or lower than depth of the $COORD_2$. In this case, no record is stored into the $COORD_2$ RT (see Table 4.5). The same situation holds for the next two records (see Table 4.7).
3. If PARENT CID equals the $COORD_2$, the current row from PAN RT (see Table 4.8) is inserted into the COORD RT (CID=3, PARENT CID=2) (see Table 4.9).
4. If current PARENT CID equals initial value (255), it implies, that the current CID is not included in the network and no record is stored into the COORD RT (see Table 4.11).
5. If none of the previous conditions in 2, 3 and 4 is met, the current CID is stored in a temporary variable. Then reverse loop starts with the aim to find if current PARENT CID is in the $COORD_2$ subtree.
   (a) If current PARENT CID is found, the current row from PAN RT is stored into the COORD RT (CID=5, PARENT CID=3) (see Table 4.13).
   (b) Otherwise, no record is stored into the COORD RT.

6. In the next step, the situation is the same as in 5 (see Table 4.15).

Apart from routing with decision process on each node located on the route between a source and a destination node, the other possible solutions exist. Regarding the FIT protocol, routing is performed on each network node. In this case, the packet has not to include a list of nodes through which it should be routed to the destination node. As the result, the packet can contain only address of the destination node. In comparison to the other protocols (e.g. Z-Wave), a considerably smaller amount of data is transferred and both, influence of interference and energy consumption of packet transfer, are decreased.

4.6 Network Reinitialization

If end device or coordinator sends data towards PAN coordinator using four-way handshake, then movement out of its parent range can be detected. In this case, the four-way handshake fails, and network reinitialization is initiated. As the result, end device or coordinator is able to connect to another coordinator node.

1. Movement of an end device is detected when communication towards PAN coordinator using four-way handshake fails (in case of an unsuccessful retransmission of `DATA` or `COMMIT` packet).
2. The moved end device broadcasts `MOVE REQUEST` packet to the network.
3. All coordinators which receive this packet resend it as `MOVE REQUEST ROUTE` packet toward PAN coordinator.
4. PAN coordinator chooses parent of the end device according to the best-received signal strength. Then PAN coordinator answers with:
   (a) `MOVE RESPONSE` packet if it is selected as a parent of the end device (PAN coordinator – end device),
   (b) `MOVE RESPONSE ROUTE` packet otherwise (PAN coordinator – coordinators – end device), this packet is routed back to a new parent of the moved device which changes the type of `MOVE RESPONSE ROUTE` to `MOVE RESPONSE` packet and sends it to the end device.
5. The end device obtains its new parent identifier.

Thanks to the designed process of network reinitialization, fast reaction to changes (e.g. device movement or error occurrence) in a network is possible. If communication failure occurs among nodes in the Z-Wave network, for example, some node is not able to communicate with its parent node, two possible scenarios can occur. If a static controller is moved, network reorganisation or network scan is demanded. On the contrary, a portable controller is supposed to change its position. In case of the FIT protocol, complete network reinitialization is always performed after an error is detected. This process can be considered as an advantage in case that more places in the network show communication problems because the network can be initiated faster. On the other hand, a possibility of joining to another available network is not supported because channel scanning is not performed. This opportunity can be useful if traffic in a network is highly influenced by interference.

Chapter 5

Hardware Platform

Communication among all types of devices (end devices, coordinators, and PAN coordinator) is realised using the radio module MRF89XA in version MRF89XAM8A operating in the frequency range 863–870 MHz [46]. During testing all devices operated in the 863 MHz frequency band. The radio module supports data rates up to 200 kbps, transmit output powers up to 13 dBm [45] and maximum range about 100 meters [48]. SPI (Serial Peripheral Interface) is used as communication interface between microcontroller and radio module. It also serves for data transfer among devices [43].

PAN coordinator is based on the open source hardware board A10-OLinuXino-LIME by Olimex with the A10 1GHz Cortex-A8 ARMv7. This board is extended by the BeeeOn PAN coordinator module with RF (Radio Frequency) capability for wireless communication that includes LAN adaptor, RTC (Real-time Clock) and a pressure sensor (see Figure 5.1). PAN receives data from end devices (sensors and actuators) and sends them to a server for further analysis. It is also possible to send various commands from the server to end devices through PAN coordinator. PAN coordinator has to be connected to the Internet via Ethernet cable so that communication with the server was possible. PAN coordinator has to stay awake continuously to communicate with end devices which wake up from sleep mode. Therefore a power adaptor is used for constant power supply [49], [50], [39], [1].

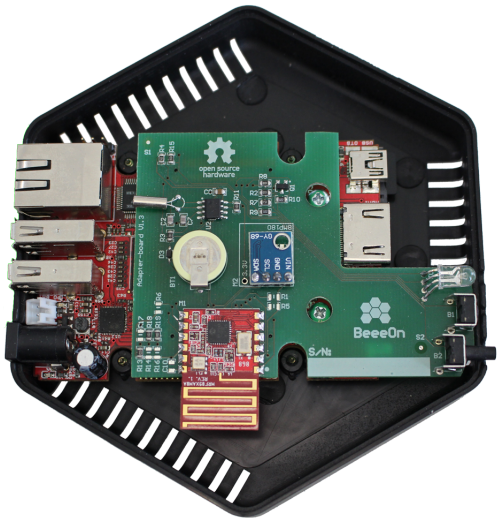


Figure 5.1: The Olimex A10-OLinuXino-LIME with the BeeeOn PAN coordinator module

The Microchip MiWi Demo Kit-868 MHz MRF89XA serves as a coordinator (see Figure 5.2). It uses two AAA batteries for power supply. Two push-buttons SW1 and SW2 are integrated on the board and are assigned to individual interrupt lines of a microcontroller. By pressing button SW1, an interruption is caused and joining process is initiated. The board also includes three status indicator LED diodes and LCD character display. The next component is PIC18F46J50 microcontroller. It is an 8-bit XLP 44-pin microcontroller with 64 KB program memory and 3776 B SRAM. Current consumption for the entire board can be measured at JP1 and JP2 without disturbing it [47], [44].

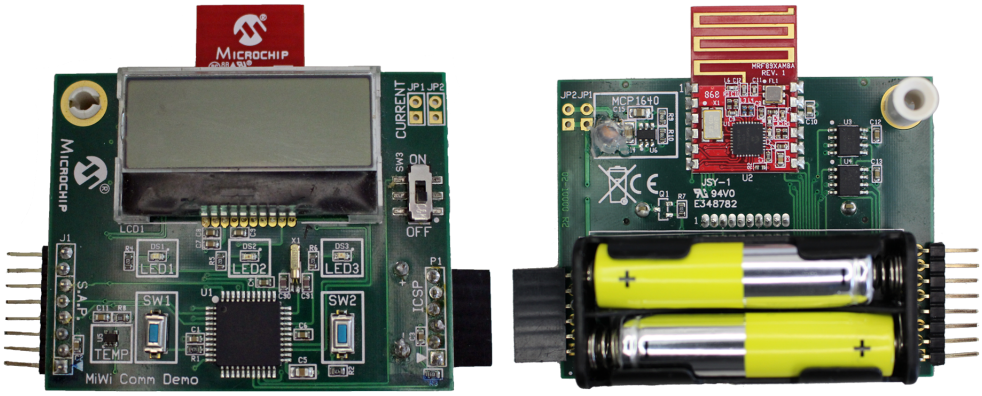


Figure 5.2: The Microchip MiWi Demo Kit-868 MHz MRF89XA – front and rear view

End device is represented by the BeeeOn sensor v1.2 that uses two AAA batteries (see Figure 5.3). The BeeeOn sensor can measure temperature and humidity (using humidity and temperature sensor HTU21/SHT21). It is also possible to connect external temperature sensor [51], [39].

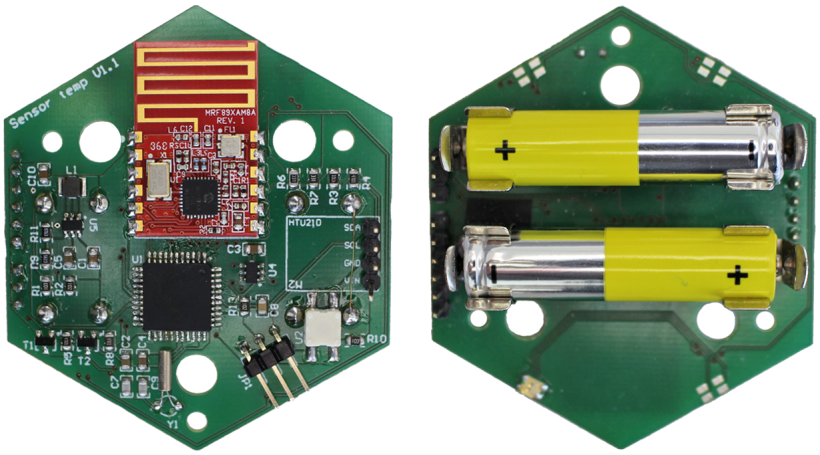


Figure 5.3: The BeeeOn sensor v1.2 – front and rear view

Chapter 6

Testing

The aim of the FIT protocol testing was to prove the correctness of protocol implementation. Firstly, it was necessary to verify the functionality of basic mechanisms (joining process, routing process and network reinitialization). In the case of joining process, correct parent choice of a joining device and dependency of success rate on a distance between devices were tested. The success rate of reliable and unreliable data transfer was compared concerning a distance between devices and position of the PCB antenna integrated into devices. Then the influence of obstacles on received signal strength was explored. Finally, energy consumption of FIT, Z-Wave, ZigBee and Thread protocol was calculated. It was supposed that the BeeeOn sensor v1.2 is able to communicate using all these protocols. Tests were carried out in a laboratory, in corridor and in flat. In the tables, end device is marked as E, coordinator x as C_x and PAN coordinator as P.

In the test case 1, 2, 4 and 5, certain steps had to be carried out before the beginning of testing. It was necessary to program the devices and to place them in the particular positions. The distance between devices was one meter, and devices were placed one meter above the ground.

6.1 Test Case 1: Success Rate of Joining Process in Various Network Topologies

The purpose of the first test case was verification of joining process functionality. The success rate of the process depends on the number of successful attempts to add a device to a network. The expected number of unsuccessful attempts was low because of short distance among devices. Tests were performed in five topologies (see Table 6.1) that cover direct joining (topology 1, 2) and indirect joining (topology 3, 4, 5).

A joining device broadcasted 1000 `JOIN REQUEST` packets[1] one by one. Each packet was received by all devices in a network which routed the packet towards PAN coordinator. As the result, PAN coordinator chose a parent of the joining device from all devices in the network. Source codes were slightly modified for testing purposes with the aim to make testing more efficient. Each of the sent packets had a particular length. Specifically, lengths of `JOIN REQUEST`, `ACK JOIN REQUEST` and `JOIN RESPONSE` packets were 20 bytes, 10 bytes and 14 bytes, respectively.

[1] The amount of packets was chosen on basis of the document [63].

Number	Joining device	Topology	Success rate [%]
1	E	P	99,9
2	C_1	P	99,9
3	E	$C_1 \rightarrow$ P	99,8
4	E	$C_1 \rightarrow$ P $C_2 \nearrow$	99,8
5	E	$C_2 \rightarrow C_1 \rightarrow$ P	99,8

Table 6.1: Success rate of joining process

The results are presented in Table 6.1. The joining device is in the second column, and the network topology is shown in the third column. The last column shows the success rate of the joining process. The success rate of joining process was not less than 99.8 %. Unsuccessful attempts to join the network can be caused by environmental factors such as temperature, humidity or presence of people [40], [29]. Success rate could by increased using acknowledgement of each transmitted packet.

6.2 Test Case 2: Parent Choice of Joining Device

Tests focused on the correct choice of a parent which was assigned to a joining device. The parent was chosen on the basis of received signal strength (RSSI). It was expected that device with the highest RSSI becomes the parent of a joining device. The network topologies were the same as in the test case 1 (see Table 6.1).

Joining process was manually initiated 20 times by shaking with an end device or pressing the SW1 button on a coordinator board. A joining device broadcasted `JOIN REQUEST` packet. Coordinators sent on this packet with RSSI towards PAN coordinator which determined a parent of the joining device according to the highest RSSI. The test setup was the same as in the test case 1 (see section 6.1).

Tables A.4–A.8 in Appendix A show RSSI of devices that received `JOIN REQUEST` packet. If an RSSI value was not delivered on PAN coordinator, the unknown RSSI value is denoted by a dash (-). The last column contains selected parent. All attempts to join the network were successful.

6.3 Test Case 3: Success Rate of Joining Process in Various Distances

The third test case examined dependency of success rate on a distance between devices. It was expected that success rate of joining process would decrease with longer distance between devices. On the contrary to the first and the second test cases, only direct joining process between PAN coordinator and end device was considered.

Firstly, it was necessary to determine the maximum distance between devices in which joining process is possible. Then the distance was gradually decreased by 5 meters. End device broadcasted 1000 `JOIN REQUEST` packets one by one. The packet was received by PAN coordinator. As the result, end device was added to a network. Moreover, joining process was three times manually initiated in each distance to calculate average RSSI of PAN coordinator. A length of sent packets and slight modifications in source codes were the same as in the case of the test case 1 (see section 6.1).

Distance [m]	RSSI of PAN [dB]	Success rate [%]
34	31.33	81.60
29	30.33	91.20
24	32.67	96.70
19	33.67	99.80
14	42.00	99.60
9	45.33	100.00
4	58.00	99.80

Table 6.2: Success rate of joining process in dependency on distance

The results are in Table 6.2. The first column presents distance between devices, RSSI of PAN coordinator is showed in the second column and success rate is presented in the last column. Tests showed that success rate of joining process more or less increased with the shorter distance between devices but it did not hold in all cases (see Figure 6.1). The reason can be that `JOIN REQUEST` packet was not retransmitted if the acknowledgement was not received. Packet loss can be caused by environmental factors as mentioned in section 6.1.

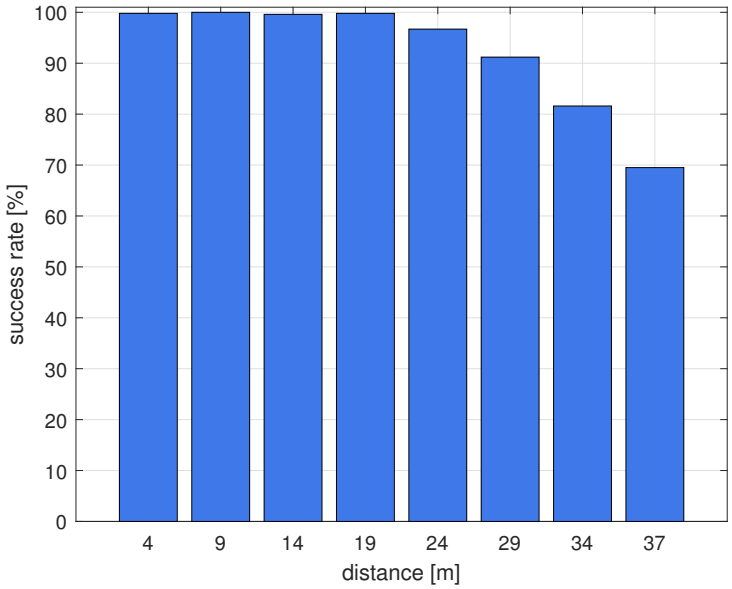


Figure 6.1: Influence of distance on success rate of joining process

6.4 Test Case 4: Success Rate of Routing Process in Various Network Topologies

The fourth test case was focused on verification of correct packet delivery to the destination device. It was supposed that success rate would reach 100 % due to reliable data transfers. Testing took five various network topologies (see Table 6.3) into account. Arrow orientation expresses the direction of data transfers. PAN coordinator and coordinators on the packet delivery path are in bold. Packet routing depended on the position of source and destination

device in the topology and also on the way of addressing (see the last paragraph of section 4.3).

The first topology (topology 1) shows data sending from the end device to the PAN coordinator. The second topology (topology 2) is the same as the first one, but data are sent from the PAN coordinator to the end device. The following topology (topology 3) shows data sending from the coordinator C_1 to the coordinator C_2. Data sending from the coordinator C_3 to the end device connected to the coordinator C_1 is shown in the next topology (topology 4). Even though sender and receiver are in the same branch of the topology, a packet is firstly routed to the PAN coordinator that knows the parent of the end device. Then the PAN coordinator sends the packet back to the end device connected to the coordinator C_1. The last topology (topology 5) is similar to the previous one. Data are also sent from the coordinator C_3 but the destination device is the PAN coordinator.

We can see that 1000 packets were successfully sent using the four-way handshake. The packets were sent consecutively. It means that next packet was sent only if the previous packet reached the destination device. The total length of each packet was 34 bytes (link header length was 10 bytes, network header length was also 10 bytes, and data length was 14 bytes).

Number	**Topology**	**Success rate [%]**
1	E $\rightarrow$ **C_1** $\rightarrow$ P	100
2	P $\rightarrow$ **C_1** $\rightarrow$ E	100
3	C_1 $\rightarrow$ **P** $\rightarrow$ C_2	100
4	C_3 $\rightarrow$ **C_2** $\rightarrow$ **C_1**(E) $\leftrightarrow$ **P**	100
5	C_3 $\rightarrow$ **C_2** $\rightarrow$ **C_1** $\rightarrow$ P	100

Table 6.3: Success rate of routing process

The results are presented in Table 6.3. Tests showed that the routing algorithm works correctly and data sending using the four-way handshake is a reliable way of transfer. Success rate reached 100 % in all tested topologies.

6.4.1 Sleepy Packets

It was needed to check if PAN coordinator is able to react to data request sent by end device and provide requested data to end device. It was expected that success rate should be 100 %. The first topology (topology 1) shows direct data request. Indirect request for data is presented in the second topology (topology 2) where the coordinator C_1 routes the request to the PAN coordinator, and then it relays the data back to the end device.

The end device successively sent 1000 requests for data towards the PAN coordinator. For testing purposes, it was ensured that requested data was available on the PAN coordinator. The packet length was the same as in case of data sending using the four-way handshake.

Number	**Topology**	**Success rate [%]**
1	E $\rightarrow$ P	100
2	E $\rightarrow$ **C_1** $\rightarrow$ P	100

Table 6.4: Success rate of sleepy packet sending

Table 6.4 shows the success rate of sleepy packet sending. All direct and indirect requests for data were successful, and end device always received the requested data.

6.5 Test Case 5: Success Rate of Network Reinitialization in Various Network Topologies

The test case 5 was focused on network reinitialization after device movement. It was performed in four different topologies (see Table 6.5). Data was sent in a direction marked by the arrow. Device movement was realised by switching off device parent which is marked in red colour. For example, the coordinator C_1 in the first topology is switched off, it means that the end device lost its parent and network reinitialization starts. Tests considered various types of topologies (linear and tree topology) and also a different number of moved devices (single or multiple movement). The topology 1 and the topology 3 show single device movement in linear and tree topology respectively. Multiple device movement is performed in the topologies 2 and 4. The success rate of 100 % was requested.

Sender sent 1000 `MOVE REQUEST` packets one by one. Each packet was received by all devices in the network (except the moved device) which routed the packet towards PAN coordinator. As the result, PAN coordinator chose a new parent of a device that lost its parent from all devices in the network.Source codes were slightly modified for testing purposes to make testing faster. All packets had the same length of 20 bytes.

The particular example of device movement is shown in Figure 6.2. The network topology is the same as in section 4.5. It is supposed that the coordinator C_6 sends packets to the PAN coordinator using the four-way handshake. The coordinator C_5 (marked in red colour and crossed out) is moved out of the network. It implies that the coordinator C_6 lost its parent and network reinitialization starts. As the result, only the last row of the PAN, C_2, C_3 and C_6 routing table is modified. These devices are inside the blue dashed ellipse. The parent of the coordinator C_6 is changed from the value 5 (C_5) to the value 3 (C_3).

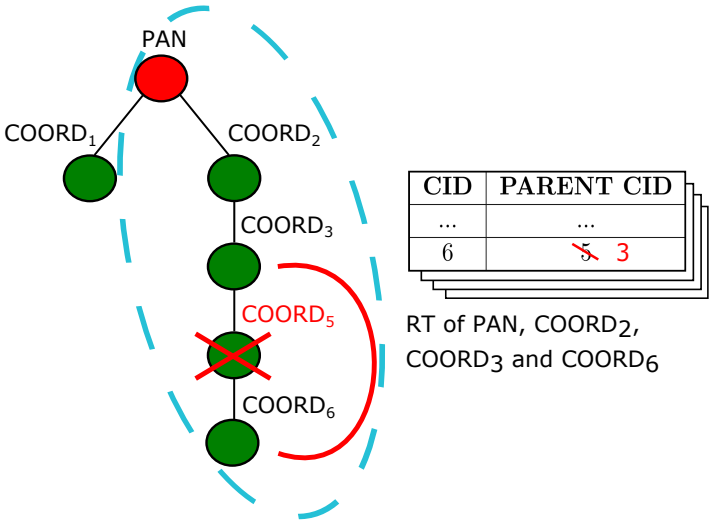


Figure 6.2: Device movement and change of routing tables

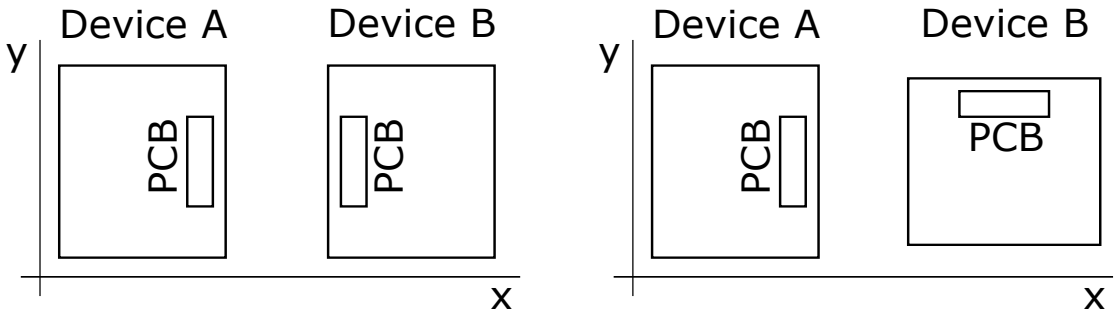
Number	Topology	Success rate [%]
1	E → C_1 → P	100
2	E → C_3 → C_2 → C_1 → P	100
3	C_3 → C_1 → P C_4 → C_2 ↗	100
4	E → C_3 → C_1 → P C_4 → C_2 ↗	100

Table 6.5: Success rate of network reinitialization

Table 6.5 shows that success rate of network reinitialization reached 100 % in all topologies. It implies that network reinitialization was always correctly performed. All devices that lost its parent were able to communicate again.

6.6 Test Case 6: Dependency of Data Transfer Success Rate on Distance Between Devices

The aim of the sixth test case was to determine how a position of the PCB antenna and distance between devices influence success rate of reliable and unreliable data transfers. Reliable transfers should be more successful than unreliable. In case of the PCB antenna position, higher success rate of data delivery was expected in the parallel antenna position. The distance between devices was from 5 to 25 meters with 5-meter increment. Two PCB antenna positions were considered in each distance (see Figure 6.3a and 6.3b). For both positions, the PCB antenna was vertically oriented with respect to the ground (Z plane). The parallel orientation of antennas meant, that front sides of devices were faced to each other. The orthogonal position was similar to the previous one, but one device was rotated by the angle of 90 degrees in XY plane.



(a) Parallel position

(b) Orthogonal position

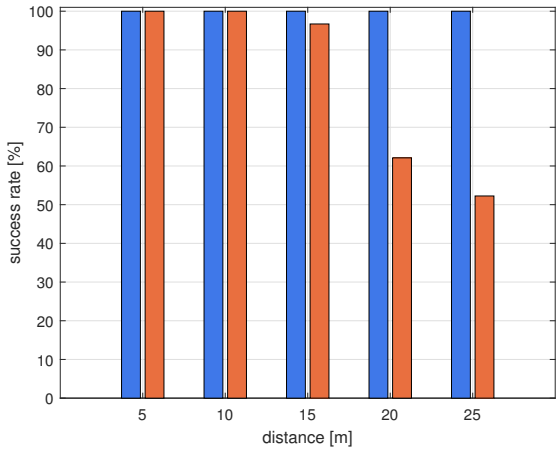
Figure 6.3: Positions of PCB antennas

For each distance and antenna position, source device successively sent 1000 data packets using reliable (four-way handshake) and unreliable (without confirmation) transfers. The length of the packet was 34 bytes.

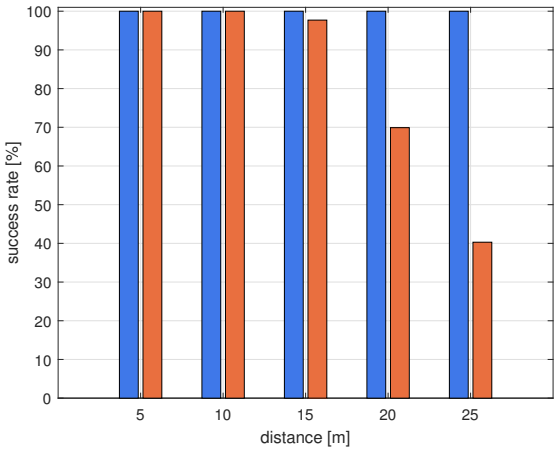
Position of the antenna	**Distance [m]**	**Success rate [%]**	
		Unreliable Transfer	Reliable Transfer
parallel	5	100.00	100.00
orthogonal		100.00	100.00
parallel	10	100.00	100.00
orthogonal		100.00	100.00
parallel	15	96.70	100.00
orthogonal		97.70	100.00
parallel	20	62.10	100.00
orthogonal		69.90	100.00
parallel	25	52.25	100.00
orthogonal		40.30	100.00

Table 6.6: Success rate of unreliable and reliable data delivery with respect to distance and PCB antenna position

Table 6.6 shows influence of distance and PCB antenna position on unreliable and reliable data transfer. The results confirmed that longer distance influences more unreliable than reliable data transfers (see Figure 6.4a and 6.4b). The position of PCB antenna did not influence either unreliable or reliable data delivery success rate.



(a) Parallel PCB antenna position



(b) Orthogonal PCB antenna position

Figure 6.4: Success rate of reliable (blue columns) and unreliable (orange columns) data transfer

Dependency of RSSI on a distance between devices and PCB antenna position is shown in Figure 6.5. RSSI decreases with distance between devices as was expected. On the other hand, RSSI is not influenced by PCB antenna position.

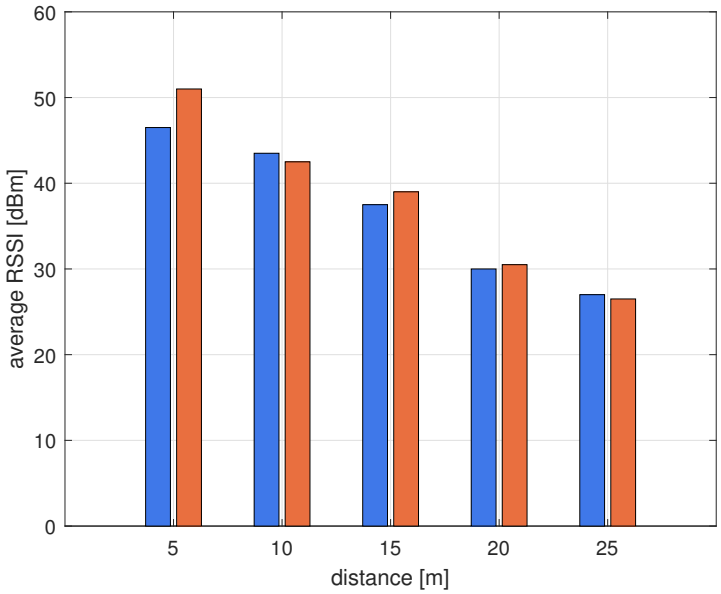


Figure 6.5: Dependency of RSSI on distance and PCB antenna position (parallel – blue columns, orthogonal – orange columns)

6.7 Test Case 7: Dependency of Received Signal Strength on Obstacles Between Devices

The seventh test case confirmed the influence of obstacles on received signal strength. It was supposed that the least permeable material of obstacle would influence received signal strength in the largest extent. Device placement and position of the obstacle are shown in Figure 6.6. The distance between device and obstacle was 20 cm, the total distance between devices was 40 cm.

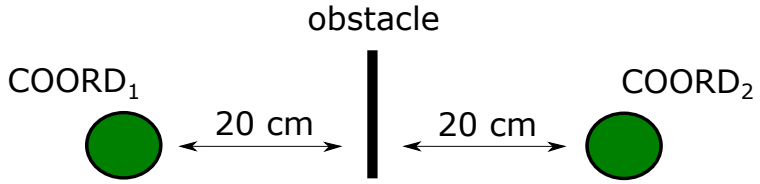


Figure 6.6: Device and obstacle placement

The sender successively sent 1000 packets using the four-way handshake. The length of each packet was 34 bytes (see section 6.4).

Obstacle	Thickness [mm]	Size [cm]	RSSI [dB]	Success rate [%]
None	-	-	69.00	100.00
Glass	20	-	58.56	100.00
Wood plank	62	50 × 50	60.16	100.00
Wood door	40	-	62.15	100.00
Aluminium	1.5	50 × 50	39.80	100.00
Stainless steel	1	50 × 50	41.90	100.00
Brick thick wall	300	-	60.00	100.00
Brick thin wall	97	-	58.66	100.00

Table 6.7: Dependency of RSSI on obstacle

The results in Table 6.7 show signal attenuation through obstacle for various materials. All sent packets were successfully delivered to the destination device. It was found out that permeability of aluminium and stainless steel is the lowest. Other materials influenced the signal only moderately.

6.8 Energy Consumption

The aim was to calculate energy consumption of the BeeeOn sensor v1.2 (see chapter 5) with respect to various transmitted data lengths in the FIT protocol. Then it was supposed that the BeeeOn sensor v1.2 communicates using Z-Wave, ZigBee and Thread protocol. Energy consumption of the FIT and above-mentioned protocols was compared.

Firstly, it was necessary to measure a voltage of batteries that supplied the BeeeOn sensor v1.2. Then a current was measured during data transfer from the sensor to PAN coordinator. Current-to-voltage converter had to be used because the used oscilloscope was only able to measure voltage.

Energy consumption during the 1 bit transmission E [J] was calculated using equation (6.1) where U [V] is voltage, I [A] is current and t [s] is 1 bit transmission time.

$$E_{bit} = U \cdot I \cdot t \tag{6.1}$$

The 1 bit transmission time was calculated by equation (6.2) where $bitrate$ [bps] is bitrate of data transfer.

$$t = \frac{1}{bitrate} \tag{6.2}$$

The bitrate was $66\,kbps = 66000\,bps$, therefore $t = \frac{1}{66000} = 1.52 \cdot 10^{-5}\,s$. Energy consumption during the 1 bit transmission (equation (6.1)) was $E_{bit} = 3.1{\cdot}0.025{\cdot}1.52{\cdot}10^{-5} = 1.178 \cdot 10^{-6}\,J$.

The overall energy consumption E_{total} with respect to various packet lengths (denoted as n) was calculated using equation (6.3).

$$E_{total} = n \cdot E_{bit} \tag{6.3}$$

Energy consumption depends on packet length. Each packet consists of headers and data, the total packet length $n = header\,length + data\,length$. The overall energy consumption of each protocol was calculated using the same data length settings. Therefore energy consumption was only influenced by header lengths of the particular protocol. The header

lengths of FIT, Z-Wave, ZigBee and Thread protocol are 20 bytes, 24 bytes [36], 31 bytes [64] and 25 bytes [69], [32] respectively.

The overall energy consumption of each protocol can be calculated using equation (6.3). For example, the E_{total} of the FIT protocol was $E_{total} = 8{\cdot}(20{+}5){\cdot}1.178{\cdot}10^{-6} = 2.356{\cdot}10^{-4}\,J$ in case of 5 bytes data length.

Table 6.8 shows energy consumption of the chosen protocols. The data lengths were 5, 10, 15 and 20 bytes. The results confirmed that the FIT protocol is the most energy efficient of all compared protocols. On the other hand, the greatest amount of energy consumes the ZigBee protocol.

Protocol	**Lengths [B]**			**Energy Consumption [J]**
	Data	**Header**	**Total**	
FIT	5	20	25	$2.35600 \cdot 10^{-4}$
Z-Wave		24	29	$2.73296 \cdot 10^{-4}$
ZigBee		31	36	$3.39264 \cdot 10^{-4}$
Thread		25	30	$2.82720 \cdot 10^{-4}$
FIT	10	20	30	$2.82720 \cdot 10^{-4}$
Z-Wave		24	34	$3.20416 \cdot 10^{-4}$
ZigBee		31	41	$3.86384 \cdot 10^{-4}$
Thread		25	35	$3.29840 \cdot 10^{-4}$
FIT	15	20	35	$3.29840 \cdot 10^{-4}$
Z-Wave		24	39	$3.67536 \cdot 10^{-4}$
ZigBee		31	46	$4.33504 \cdot 10^{-4}$
Thread		25	40	$3.76960 \cdot 10^{-4}$
FIT	20	20	40	$3.76960 \cdot 10^{-4}$
Z-Wave		24	44	$4.14656 \cdot 10^{-4}$
ZigBee		31	51	$4.80624 \cdot 10^{-4}$
Thread		25	45	$4.24080 \cdot 10^{-4}$

Table 6.8: Energy consumption of selected protocols

Figure 6.7 shows graphical representation of Table 6.8.

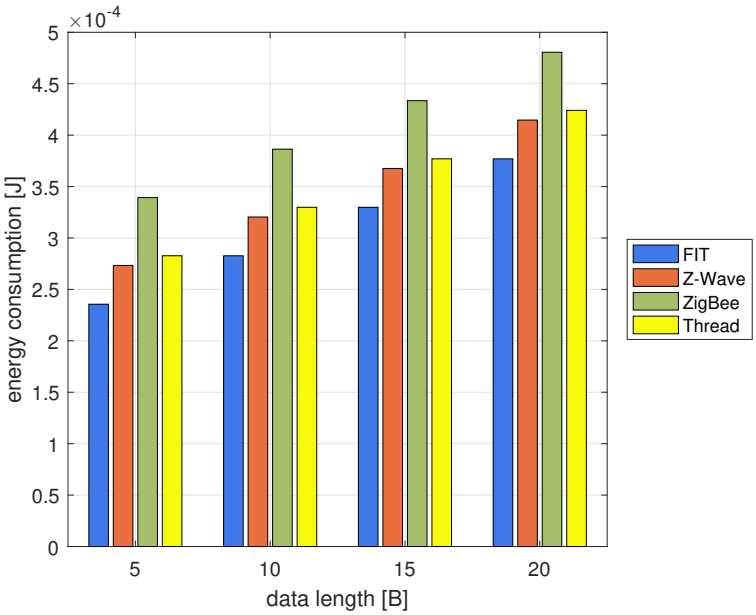


Figure 6.7: Energy consumption of selected protocols

6.9 Testing in the Simulator

The simulator was used for verification of the most basic functionalities provided by the FIT protocol: direct joining process and direct data sending between end device and PAN coordinator. The simulator has limited function support which implies that more difficult topologies and functionalities could not be verified. Therefore careful tests were mainly performed in a real environment.

Tests were focused on correct exchange of the join packets and successful delivery of network and parent identifier to a joining device. Due to the restricted simulator functionality, channel scanning and parent choice based on RSSI were ignored.

During testing of direct communication between devices, data was sent using the four-way handshake. Tests were concentrated on correct packet exchange in the course of the four-way handshake.

Chapter 7

Conclusion

The main aim of the Bachelor's thesis was to extend the FIT protocol that was developed at the Faculty of Information Technology (FIT) within the BeeeOn project [49]. All the goals of the work were completely fulfilled. At first, it was necessary to find out what the expression "smart home" means and what benefits it could provide people in the future. After that, the most popular smart home protocols (Z-Wave, ZigBee and Thread) were carefully analysed. Their characteristics, advantages, disadvantages, supported ways of communication and basic mechanisms (joining process, routing process and network reinitialization) were pointed out. Then the source code of the FIT protocol was analysed to understand its behaviour. Basic mechanisms and the other ways of communication were subsequently designed, implemented and tested.

Periodic channel scanning has been used in joining process for fast detection of available channels. Parent choice is based on received signal strength indicator (RSSI). Both, data sending from end device towards PAN coordinator or from PAN coordinator towards end device are supported. Apart from reliable data transfers with the four-way handshake, it is possible to send data without confirmation of correctness by `ACK` packet on link layer, to use broadcast packet or sleepy packet. A special algorithm for routing subtree counting ensures that device knows network topology of its descendants. Routing subtree is distributed only to device descendants; therefore network is not flooded as it is in case of broadcasting. If device indicates parent loss, network reinitialization ensures a new parent choice according to RSSI. As a result, device is able to communicate again.

The real hardware developed by Microchip, Olimex and BeeeOn was used for testing purposes. Seven test cases were performed to verify functionality of all implemented mechanisms. Each test case included several network topologies to cover different situations within the mechanism. The simulator was used to test basic test cases. The tests focused on the influence of the PCB antenna position and distance between devices on the success rate of data transmission. The dependency of received signal strength on obstacles between communicating devices was explored.

All measured results proved the correctness of implementation. In contrary to unreliable data transfer, reliable data transfer is not influenced by longer distance between devices. It was also found out that success rate of unreliable or reliable data transfer does not depend on the PCB antenna position. Moreover, influence of data length on energy consumption of the BeeeOn sensor v1.2 was analysed. Energy consumption was calculated for FIT, Z-Wave, ZigBee and Thread protocols.

The FIT protocol could be further improved by next functional properties. Communication could be encrypted to improve the security of data transmission. If a network

is reinitialized, it would be useful to support rejoining to another available network. For example, the current channel, highly influenced by interference could be changed to a less jammed channel.

Bibliography

[1] A10-OLinuXino-LIME. Online. Accessed on 2017-04-18.
Retrieved from: https://www.olimex.com/wiki/A10-OLinuXino-LIME

[2] Ad hoc On Demand Distance Vector (AODV) Routing Protocol. Online. Accessed on 2017-01-15.
Retrieved from: http://www.cs.jhu.edu/~cs647/aodv.pdf

[3] Connectivity of the Internet of Things. Online. Accessed on 2016-09-15.
Retrieved from: https://learn.sparkfun.com/tutorials/connectivity-of-the-internet-of-things#Thread

[4] The Guide to a User Friendly & Stable Z-Wave Home Automation Network. online.
Retrieved from: http://www.vesternet.com/resources/technology-indepth/user-friendly-z-wave-network

[5] Home Automation Protocols: A Round-Up. Online. Accessed on 2016-11-29.
Retrieved from: https://www.electronichouse.com/smart-home/home-automation-protocols-what-technology-is-right-for-you/

[6] Importance of Networking. Online. Accessed on 2016-01-21.
Retrieved from: http://techno-services.net/

[7] Understanding Z-Wave Networks, Nodes & Devices. Online. Accessed on 2016-09-15.
Retrieved from: http://www.vesternet.com/resources/technology-indepth/understanding-z-wave-networks

[8] Ushering in a New Era of Internet Connectivity with Thread Networking Protocol. Online. Accessed on 2016-09-15.
Retrieved from: https://www.silabs.com/products/wireless/Pages/thread-ushering-in-new-era-of-internet-connectivity.aspx

[9] What is Z-Wave? Online. Accessed on 2017-01-10.
Retrieved from: http://www.smarthome.com/sc-what-is-zwave-home-automation

[10] Z-Wave Network Layer. Online. Accessed on 2017-01-15.
Retrieved from:
http://wiki.zwaveeurope.com/index.php?title=Z-Wave_Network_Layer

[11] z-wave protocol stack. Online. Accessed on 2016-09-15.
Retrieved from:
http://www.rfwireless-world.com/Tutorials/z-wave-protocol-stack.html

[12] Z-Wave: The Basics. Online. Accessed on 2016-09-15.
Retrieved from: http://www.z-wave.com/faq

[13] Z-Wave Tutorial. Online. Accessed on 2017-01-15.
Retrieved from: https://iotpoint.wordpress.com/z-wave-tutorial/

[14] ZigBee. Online. Accessed on 2017-01-15.
Retrieved from:
http://internetofthingsagenda.techtarget.com/definition/ZigBee

[15] Zigbee AODV protocol basics. Online. Accessed on 2016-09-15.
Retrieved from:
http://www.rfwireless-world.com/Tutorials/Zigbee-AODV-protocol.html

[16] ZigBee Home Automation. Online. Accessed on 2016-09-15.
Retrieved from: http://www.zigbee.org/zigbee-for-developers/applicationstandards/zigbeehomeautomation/

[17] ZigBee tutorial (v12). Online. Accessed on 2017-01-15.
Retrieved from: http://www.libelium.com/development/waspmote/documentation/zigbee-tutorial-v12/

[18] Zigbee vs. Z-Wave. Online. Accessed on 2016-09-15.
Retrieved from:
http://support.greenologic.co.uk/SharedFiles/Download.aspx?pageid=23&mid=30&fileid=114

[19] Zigbee Tree Routing - How It Works and Why It Sucks. Online. 2009. Accessed on 2017-01-15.
Retrieved from: http://www.freaklabs.org/index.php/blog/zigbee/zigbee-tree-routing-how-it-works-and-why-it-sucks.html

[20] Thread Group Broadens Focus to Encompass the Places Where People Live and Work with Expansion Into Commercial Building Space. Online. November 2016. Accessed on 2016-09-15.
Retrieved from: https://threadgroup.org/news-events/press-releases/ID/124/Thread-Group-Broadens-Focus-to-Encompass-the-Places-Where-People-Live-and-Work-with-Expansion-Into-Commercial-Building-Space

[21] Thread Vs. ZigBee (For IoT Engineers). online. March 2016.
Retrieved from:
https://www.link-labs.com/blog/thread-vs-zigbee-for-iot-engineers

[22] Tying the IoT Together: Wireless Protocols. Online. 2017. Accessed on 2017-01-15.
Retrieved from: http://community.silabs.com/t5/Official-Blog-of-Silicon-Labs/Tying-the-IoT-Together-Wireless-Protocols/ba-p/186767

[23] Al Ameen, M.; Liu, J.; Kwak, K.: Security and privacy issues in wireless sensor networks for healthcare applications. *Journal of medical systems*. vol. 36, no. 1. 2012: pp. 93–101.

[24] Black, W.: Home automation. Online. Accessed on 2016-10-04.
Retrieved from: http://www.billyblackelectrical.co.uk/home-automation

[25] Brown, S.: ZigBee vs. Z-Wave Review: What's the Best Option for You? Online. Accessed on 2017-01-10.
Retrieved from: http://www.safewise.com/blog/zigbee-vs-zwave-review/

[26] Chaki, N.; Chaki, R.: *Intrusion Detection in Wireless Ad-hoc Networks*. CRC Press. first edition. 2014. ISBN 9781466515659.

[27] De Silva, L. C.; Morikawa, C.; Petra, I. M.: State of the art of smart homes. *Engineering Applications of Artificial Intelligence*. vol. 25, no. 7. 2012: pp. 1313–1321.

[28] Devito, M.: A security assessment of Z-Wave devices and replay attack vulnerability. *SANS Institute Magazine*. 2016.

[29] Ferrari, G.; Medagliani, P.; Martalo, M.: Performance analysis of Zigbee wireless sensor networks with relaying. *Grid Enabled Remote Instrumentation*. 2009: pp. 55–79.

[30] Frenzel, L.: What's The Difference Between ZigBee And Z-Wave? Online. 2012. Accessed on 2016-09-15.
Retrieved from: http://electronicdesign.com/communications/what-s-difference-between-zigbee-and-z-wave

[31] Gislason, D.: *Zigbee Wireless Networking*. Newnes. first edition. 2008. ISBN 9780750685979.

[32] González González, H. J.: *Study of the protocol for home automation thread*. B.S. thesis. Universitat Politècnica de Catalunya. 2017.

[33] Harper, R.: *Inside the smart home*. Springer Science & Business Media. softcover reprint of the original 1st ed. 2003 edition. 10 2013. ISBN 9781852336882.

[34] Hersent, O.; Boswarthick, D.; Elloumi, O.: *The internet of things: Key applications and protocols*. John Wiley & Sons. second edition. 2012. ISBN 9781119994350.

[35] Hu, C.: Research on Security Mechanisms for Wireless Sensor Network. *International Journal of Future Generation Communication and Networking*. vol. 9, no. 7. 2016: pp. 173–184.

[36] Huttenlocher, E.: Cyber-Warfare and Cyber-Terrorism: Step to Learning to Knowing the Difference. In *11th International Conference on Cyber Warfare and Security*. 2016. page 391.

[37] Kasraoui, M.; Cabani, A.; Mouzna, J.: Zbr-M: A New Zigbee Routing Protocol. *IJCSA*. vol. 10, no. 2. 2013: pp. 15–32.

[38] Khaday, B.; Matson, E. T.; Springer, J.; et al.: Wireless sensor network and big data in cooperative fire security system using harms. In *Automation, Robotics and Applications (ICARA), 2015 6th International Conference on*. IEEE. 2015. pp. 405–410.

[39] Korček, P.: BeeeOn - otevřený systém pro IoT. Online. Accessed on 2017-03-03.
Retrieved from: https://www.nic.cz/public_media/IT16/prezentace/Korcek.pdf

[40] Marfievici, R.; Murphy, A. L.; Picco, G. P.; et al.: How environmental factors impact outdoor wireless sensor networks: a case study. In *Mobile Ad-Hoc and Sensor Systems (MASS), 2013 IEEE 10th International Conference on*. IEEE. 2013. pp. 565–573.

[41] Mendes, T. D.; Godina, R.; Rodrigues, E. M.; et al.: Smart home communication technologies and applications: Wireless protocol assessment for home area network resources. *Energies*. vol. 8, no. 7. 2015: pp. 7279–7311.

[42] Merritt, R.: IoT in Protocol War, Says Startup. Online. 2014. Accessed on 2016-09-15.
Retrieved from: http://www.eetimes.com/document.asp?doc_id=1325114

[43] Microchip: *SPI: Overview and Use of the PICmicro Serial Peripheral Interface*.
Retrieved from: http://ww1.microchip.com/downloads/en/devicedoc/spi.pdf

[44] Microchip: *PIC18F46J50 Family Data Sheet: 28/44-Pin, Low-Power, High-Performance USB Microcontrollers with nanoWatt XLP™Technology*. 2009.
Retrieved from: http://ww1.microchip.com/downloads/en/DeviceDoc/39931b.pdf

[45] Microchip: *MRF89XA Data Sheet: Ultra Low-Power, Integrated ISM Band Sub-GHz Transceiver*. 2010.
Retrieved from: http://ww1.microchip.com/downloads/en/DeviceDoc/70622C.pdf

[46] Microchip: *MRF89XAM8A Data Sheet: 868 MHz Ultra-Low Power, Sub-GHz Transceiver Module*. 2010.
Retrieved from: http://ww1.microchip.com/downloads/en/DeviceDoc/70651A.pdf

[47] Microchip: *MiWi™Demo Kit User's Guide*. 2012.
Retrieved from: http://ww1.microchip.com/downloads/en/DeviceDoc/70687A.pdf

[48] Nečasová, K.: FIT protocol. Online. 2016. Accessed on 2016-09-15.
Retrieved from: https://beeeon.org/wiki/FIT_protocol

[49] Nečasová, K.: FIT protocol. Online. 2016. Accessed on 2017-03-15.
Retrieved from: https://beeeon.org/wiki/Main_Page

[50] Nečasová, K.: FIT protocol. Online. 2016. Accessed on 2017-02-20.
Retrieved from: https://beeeon.org/wiki/Gateway_-_standalone

[51] Nečasová, K.: FIT protocol. Online. 2016. Accessed on 2017-03-10.
Retrieved from: https://beeeon.org/wiki/Sensors

[52] Nečasová, K.: Optimisation of Wireless Communication for IoT. Technical report. Brno University of Technology. 2016. projektová praxe 1 (IP1).

[53] Nečasová, K.: Optimisation of Wireless Communication for IoT. Technical report. Brno University of Technology. 2016. projektová praxe 2 (IP2).

[54] Paetz, C.: *Z-wave Basics: Remote Control in Smart Homes*. Create Space Independent Publishing Platform. 2013. ISBN 9781783017317.

[55] Parrish, K.: z-wave Tutorial-frequency, frame, protocol, PHY, MAC, z-wave security basic tutorial. Online. Accessed on 2016-09-15.
Retrieved from:
http://www.rfwireless-world.com/Tutorials/z-wave-tutorial.html

[56] Parrish, K.: *Z-Wave Transceivers - Specification of Spectrum Related Components*. Z-Wave Alliance. 2014.
Retrieved from:
http://z-wavealliance.org/wp-content/uploads/2015/02/ZAD12837-1.pdf

[57] Parrish, K.: ZigBee, Z-Wave, Thread and WeMo: What's the Difference? Online. July 2015. Accessed on 2016-09-15.
Retrieved from: http://www.tomsguide.com/us/smart-home-wireless-network-primer,news-21085.html

[58] Pedrasa, M. A. A.; Spooner, T. D.; MacGill, I. F.: Coordinated scheduling of residential distributed energy resources to optimize smart home energy services. *IEEE Transactions on Smart Grid*. vol. 1, no. 2. 2010: pp. 134–143.

[59] Prabha, R.; Kabadi, M. G.: Overview of Data Collection Methods for Intelligent Transportation Systems. *The International Journal Of Engineering And Science*. vol. 5. 2016: pp. 16–20.

[60] Ramya, C. M.; Shanmugaraj, M.; Prabakaran, R.: Study on ZigBee technology. In *Electronics Computer Technology (ICECT), 2011 3rd International Conference on*, vol. 6. IEEE. 2011. pp. 297–301.

[61] Rasheed, A.; Mohammad, K.: Exploration and Comparison of Several AODV Implementations: A Survey. *Communications of the ACS*. vol. 2. 2009.

[62] Rzadkosz, S.: A tale of five protocols: The ultimate guide to the IoT wireless communication landscape. Online. December 2015. Accessed on 2016-09-15.
Retrieved from: https://www.silvair.com/assets/contents/media/A_Tale_of_Five_Protocols.pdf

[63] Shamanna, P.: *Simple Link Budget Estimation and Performance Measurements of Microchip Sub-GHz Radio Modules*. Microchip. 2013.
Retrieved from:
http://ww1.microchip.com/downloads/cn/AppNotes/cn567167.pdf

[64] Shi, G.; Li, K.: Signal Interference in WiFi and ZigBee Networks. *Wireless Networks*. 2016.

[65] Sohraby, K.; Minoli, D.; Znati, T.: *Wireless Sensor Networks: Technology, Protocols, and Applications*. John Wiley & Sons. 2007. ISBN 9780471743002.

[66] Staff, B.: ZIGBEE: Competitive Features. online. November 2016.
Retrieved from: http://bostoncommons.net/zigbee-competitive-features/

[67] Thread Group: *Thread Commissioning*. July 2015.
Retrieved from:
https://www.silabs.com/SiteDocs/white-papers/Thread-Commissioning.pdf

[68] Thread Group: *Thread Stack Fundamentals*. July 2015.
Retrieved from: https://www.silabs.com/SiteDocs/white-papers/Thread-Stack-Fundamentals.pdf

[69] Thread Group: *Thread Usage of 6LoWPAN*. July 2015.
Retrieved from: https://threadgroup.org/Portals/0/documents/whitepapers/Thread%20Usage%20of%206LoWPAN%20white%20paper_v2_public.pdf

[70] Tuohy, J.: What is home automation and how do I get started? Online. Accessed on 2016-01-30.
Retrieved from: http://www.networkworld.com/article/2874914/internet-of-things/what-is-home-automation-and-how-do-i-get-started.html

[71] Walters, B.: Designing low cost Wireless sensor networks for real-life applications. Online. August 2012. Accessed on 2016-11-09.
Retrieved from: http://www.ecnmag.com/article/2012/08/designing-low-cost-wireless-sensor-networks-real-world-applications

[72] Wotton, P.: Providing reliable sensing and control using ZigBee wireless networks. *RF DESIGN*. vol. 29, no. 7. 2006: page 18.

[73] Z-Wave Alliance: *Z-Wave Technical Basics*. 2011.
Retrieved from: https://www.domotiga.nl/attachments/download/1075/Z-Wave%20Technical%20Basics-small.pdf

[74] Z-Wave Alliance: *Z-Wave Networking Basics*. 2016.
Retrieved from: http://z-wave.sigmadesigns.com/wp-content/uploads/2016/08/APL13031-2-Z-Wave-Networking-Basics.pdf

[75] Zennaro, M.; Bagula, A.; Gascon, D.; et al.: Planning and deploying long distance wireless sensor networks: The integration of simulation and experimentation. In *International Conference on Ad-Hoc Networks and Wireless*. Springer. 2010. pp. 191–204.

[76] Zhang, D.; Shah, N.; Papageorgiou, L. G.: Efficient energy consumption and operation management in a smart building with microgrid. *Energy Conversion and Management*. vol. 74. 2013: pp. 209–222.

Appendices

Appendix A

Results of Tests

The network topologies 1–5 were specified in section 6.1. Arithmetic mean was calculated using equation (A.1), variance using equation (A.2) and standard deviation using equation (A.3).

$$\overline{x} = \frac{\sum_{i=1}^{n} x_i}{n} \tag{A.1}$$

$$Variance = \frac{\sum(x - \overline{x})}{(n-1)} \tag{A.2}$$

$$Stdev = \sqrt{\frac{\sum(x - \overline{x})^2}{(n-1)}} \tag{A.3}$$

Number	**Average of RSSI [dB]**		
	P	**C_1**	**C_2**
1	49.55	-	-
2	61.15	-	-
3	58.05	60.20	-
4	60.45	51.70	44.46
5	48.15	45.20	44.50

Table A.1: Average of RSSI in various topologies

Number	**Variance of RSSI [dB]**		
	P	**C_1**	**C_2**
1	13.3132	-	-
2	2.1342	-	-
3	2.0500	1.4316	-
4	0.9974	7.5895	1.2692
5	24.7658	7.4316	1.2105

Table A.2: Variance of RSSI in various topologies

Number	Standard deviation of RSSI [dB]		
	P	**C_1**	**C_2**
1	3.6487	-	-
2	1.4609	-	-
3	1.4318	1.1965	-
4	0.9987	2.7549	1.1266
5	4.9765	2.7261	1.1002

Table A.3: Standard deviation of RSSI in various topologies

Number	RSSI of P [dB]	Parent	Success
1	45	0	Y
2	45	0	Y
3	51	0	Y
4	48	0	Y
5	55	0	Y
6	48	0	Y
7	51	0	Y
8	52	0	Y
9	48	0	Y
10	54	0	Y
11	57	0	Y
12	54	0	Y
13	51	0	Y
14	48	0	Y
15	49	0	Y
16	48	0	Y
17	42	0	Y
18	48	0	Y
19	49	0	Y
20	48	0	Y

Table A.4: Parent choice of a joining device – topology 1

Number	RSSI of P [dB]	Parent	Success
1	61	0	Y
2	60	0	Y
3	61	0	Y
4	60	0	Y
5	63	0	Y
6	60	0	Y
7	60	0	Y
8	62	0	Y
9	63	0	Y
10	60	0	Y
11	63	0	Y
12	63	0	Y
13	64	0	Y
14	60	0	Y
15	60	0	Y
16	60	0	Y
17	60	0	Y
18	60	0	Y
19	63	0	Y
20	60	0	Y

Table A.5: Parent choice of a joining device – topology 2

Number	**RSSI of P [dB]**	**RSSI of C_1 [dB]**	**Parent**	**Success**
1	60	60	0	Y
2	58	60	1	Y
3	60	60	0	Y
4	58	60	1	Y
5	60	63	1	Y
6	57	60	1	Y
7	57	60	1	Y
8	57	60	1	Y
9	57	60	1	Y
10	57	60	1	Y
11	60	57	1	Y
12	60	63	1	Y
13	57	60	1	Y
14	57	60	1	Y
15	57	60	1	Y
16	56	60	1	Y
17	59	61	1	Y
18	60	60	0	Y
19	57	60	1	Y
20	57	60	1	Y

Table A.6: Parent choice of a joining device – topology 3

Number	**RSSI of P [dB]**	**RSSI of C_1 [dB]**	**RSSI of C_2 [dB]**	**Parent**	**Success**
1	60	57	45	0	Y
2	63	54	-	0	Y
3	63	54	-	0	Y
4	61	54	45	0	Y
5	60	51	45	0	Y
6	60	51	45	0	Y
7	60	54	45	0	Y
8	60	50	42	0	Y
9	60	51	45	0	Y
10	62	54	-	0	Y
11	60	54	45	0	Y
12	60	54	45	0	Y
13	60	54	42	0	Y
14	60	51	-	0	Y
15	60	48	44	0	Y
16	60	48	-	0	Y
17	60	48	45	0	Y
18	60	51	-	0	Y
19	60	48	-	0	Y
20	60	48	45	0	Y

Table A.7: Parent choice of a joining device – topology 4

Number	**RSSI of P [dB]**	**RSSI of C_1 [dB]**	**RSSI of C_2 [dB]**	**Parent**	**Success**
1	54	48	42	0	Y
2	49	48	42	0	Y
3	42	48	45	1	Y
4	42	48	45	1	Y
5	39	48	45	1	Y
6	38	48	42	1	Y
7	51	48	45	0	Y
8	41	45	45	1	Y
9	52	45	45	0	Y
10	48	42	45	0	Y
11	51	45	45	0	Y
12	53	45	45	0	Y
13	51	45	44	0	Y
14	49	42	45	0	Y
15	48	45	45	0	Y
16	51	42	45	0	Y
17	48	40	45	0	Y
18	54	48	45	0	Y
19	51	42	45	0	Y
20	51	42	45	0	Y

Table A.8: Parent choice of a joining device – topology 5

Appendix B

Contents of the Attached CD

Following directories and files can be found on the attached CD:

- file `xnecas24.pdf` – an electronic version of this thesis in PDF format,
- directory `text` – a directory containing the LaTeXsource files of this thesis,
- file `README` – installation instructions,
- directory `src` – a directory containing all the source code files,
  - directory `original` – original version of all source code files,
  - directory `extended` – extended version of original source code files,
- directory `doc` – a source code documentation.