Software application for analyzing ZigBee network performance in university courses

Conference Paper · September 2014

DOI: 10.1109/SISY.2014.6923560

D. Dobrilovic
University of Novi Sad

Zeljko Stojanov
University of Novi Sad

Vladimir Brtka
University of Novi Sad

Zlatko Covic

Software Application for Analyzing ZigBee Network Performance in University Courses

Dalibor Dobrilovic*, Zeljko Stojanov*, Vladimir Brtka*, Zlatko Čović**, Nemanja Bilinac*
* University of Novi Sad/Technical Faculty "Mihajlo Pupin", Zrenjanin, Serbia
** Subotica Tech, Subotica, Serbia
ddobrilo@tfzr.rs; zeljko.stojanov@tfzr.rs; chole@vts.su.ac.rs; vbrtka@tfzr.uns.ac.rs; bilinac@tfzr.rs

***Abstract*—Wireless Sensor Networks (WSN) and related technologies have the important impact nowadays. These technologies have variety of applications, and they are key enabling technologies of IoT (Internet of Things). ZigBee standard is the most common WSN technology. Its importance and expansion, as a widely used technology in many different areas, influence the importance of including ZigBee technology in courses at university.**

This paper presents the software application and related hardware platform designed to be used by the students to measure and analyze the performances of ZigBee network in design and pre-implementation process. The presented application can be used during the classes of Communication Systems and Computer Network Courses for ZigBee traffic and packet analyses. In addition, this application can be used for researching in the field of wireless sensor networks, ZigBee and other similar popular technologies such as Bluetooth Low Energy, IEEE 802.15.4, etc. The supporting platform for this application and experiment for its evaluation are also presented in this paper.

Keywords: ZigBee, Wireless Sensor Networks (WSN), Traffic Analyzes Tools, XBee, Arduino

I. INTRODUCTION

Wireless communications are used in many areas today, such as: environmental systems, industrial and medical systems, and vehicular system. The Internet of Things (IoT) and other concepts speed up usage and development of various wireless technologies that exist today such as: ZigBee, Z-Wave, IEEE 802.15.4, WirelessHART, 6loWPAN, Bluetooth, Bluetooth Low Energy, RFID, NFC, etc. ZigBee technology has its applications in all enlisted areas.

Having this in mind, the incorporation of ZigBee and similar technologies in courses at university is reasonable. The ZigBee technology can be studied strictly from the communication technology viewpoint, but it can be also studied as a key enabling technology for Wireless Sensor Networks and IoT (Internet of Things), e.g. in systems for monitoring temperature, light intensity, pollution and other indoor and outdoor environmental conditions.

The application which is designed for analysis of ZigBee [1] traffic in order to be a part of university curricula in topics connected to Wireless Sensor Networks [2] is presented in this paper. This application is presented together with supporting hardware and software platforms used for data acquisition and experimentation. The application can be used for analyzing ZigBee node performance, for ZigBee network deployment planning and testing and for the teaching purposes as well. The aim of the presented application is to combine powerful features of commercial tools for ZigBee protocol analyses such as Ubiqua Protocol Analyzer [3] and Perytons™ Protocol Analyzers [4] with features of free tools such as SmartRF™ Packet Sniffer [5] in the form needed for its successful incorporation in Computer Science and Information Technology curricula for teaching Wireless Sensor Networks. The two mentioned applications (Ubiqua Protocol Analyzer and Perytons™ Protocol Analyzer) are expensive professional tools. Ubiqua Protocol Analyzer is "a tool designed to assist in the various phases of Wireless Sensor Network application development: debugging, testing, and deployment" [3]. The Perytons™ Protocol Analyzer is "an indispensable sniffing and analysis tool for Zigbee, IEEE 802.15.4 and other wireless sensor networks development, integration, installation, monitoring, and troubleshooting" [4]. SmartRF™ Packet Sniffer [5] is described later.

The presented application differs from the professional ones because it is mainly designed to be used in courses at university. Its goal is not to be real time sniffing tool, but the tool that helps students to analyze existing example ZigBee/IEEE 802.15.4 traffic's captures in order to understand ZigBee/IEEE 802.15.4 principles, performance and possible problems in the implementation process. This analysis is based on previously captured traffic with other real time sniffing tools.

Approach like this allows usage of the software on multiple computers in the same time, without having the need for multiple licenses of expensive analyzing software tools and appropriate packet sniffing hardware on those computers. In that way, the total cost of the system for laboratory usage is significantly decreased, allowing flexible, effective, low cost and easy to upgrade platform. This platform, besides the software application, consists of several components which will be presented in the following chapters.

II. THE PLATFORM FOR DATA ACQUISITION

The platform for data acquisition consists of several hardware and software components. It is built around Arduino UNO Rev3 microcontroller unit and XBee communication modules supporting ZigBee technology.

The microcontroller unit has the role of ZigBee remote node. Arduino UNO Rev3 is chosen as a low-cost, reliable and widely spread platform in academic environments for testing, researching and prototyping [6,7]. Its configuration is presented in Table I.

TABLE I.
THE CONFIGURATION OF ARDUINO UNO NODE

No	Component	Description
1	Arduino UNO Rev3	Microcontroller unit
2	XBee Shield	Expansion module
3	XBee ZB module with RP-SMA connector	ZigBee Communication module
4	2.4 GHz Antenna 3 dBi	External Antenna
5	ABS plastic box	Plastic enclosure

The XBee module [8,9] on ZigBee node is configured as a ZigBee router in AT command mode. The receiver station is based on the same XBee ZB module connected to the PC USB port using XBee USB Explorer. Both modules are preconfigured using free X-CTU software available by XBee manufacturer. The second module is configured as a ZigBee coordinator in AT mode. The PAN ID is set to 1001 and communication channel is set to 26 (0x1A).

Arduino is programmed with a free tool available at Arduino site. The main purpose of the Arduino UNO is to simulate wireless node and to generate traffic. The short program is written, and this program sends 2500 packets from the Arduino via XBee router to XBee coordinator connected to PC. Packets are sent each 100ms. The generated traffic is monitored and captured by Texas Instruments SmartRF™ Packet Sniffer 2.17.1. The SmartRF Packet Sniffer is a free software application developed for displaying and storing radio packets captured by a listening RF device connected to the PC via USB. The Packet Sniffer filters and decodes packets and displays them, with options for filtering and storing to a binary file format.

Various RF protocols are supported. These protocols include Bluetooth® Low Energy, ZigBee, IEEE 802.15.4, RF4CE, SimpliciTI™ and generic protocols (raw packet data). Software supports file management (open/close) with captured packets, selection of fields to be displayed and hidden, filtering of packets to be displayed and showing packet details and address book with the list of all known nodes in the network. The additional feature is its possibility to forward captured data to a UDP socket for real time monitoring of packets using custom tools. SmartRF™ Packet Sniffer is free software designed by Texas Instrument which works with variety of capture devices. In this experiment is used TI CC2351 USB Dongle, but the other devices for capturing ZigBee/IEEE 892.15.4 traffic such as CC2530EM+SmartRF05EB and CC2430DB can be used as well. SmartRF™ Packet Sniffer is used for ZigBee Traffic capturing, but it has limited features concerning analyses, overview, making statistics and visualization of data. For this reasons, all captured traffic is saved in SmartRF™ Packet Sniffer custom PSD file format. These PSD files are imported in the developed application in order to be parsed, analyzed and to visualize results.

The PSD file format is described at Texas Instruments Web site and in the SmartRF™ Sniffer accompanying documentation. Generally, it stores packets byte by byte in format corresponding to the ZigBee frame format. Each packet, regardless of its length is saved as a sequence of 151 bytes. If the packet is shorter than 151 byte, the rest of bytes will have value 0. The PSD file format and the ZigBee frame format, captured and saved in the same file, are presented in Fig. 1.

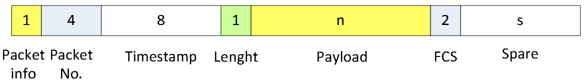

Figure 1. Packet format in PSD file [5]

The first field is the packet info with length of 1 byte. Next 4 byte field holds the packet number. Eight byte field is a timestamp in milliseconds. Packet length information is in the field 1 byte long. The packet payload in the field that depends on the data packet is carrying and FCS (Frame Check Sequence) field is 2 bytes long. The other, not used fields (spare fields) have values 0.

III. SOFTWARE APPLICATION WSN STAT

The software application called WSN Stat is designed for parsing PSD files. It is designed to be used after data acquisition. This application is a part of an ongoing project. In this stage, application only supports parsing and analyzing of ZigBee traffic files and packets. The next versions will be extended to support other protocols. The priority will be 802.15.4 and Bluetooth Low Energy traffic packets. Its primary application is for post-processing, statistics and visualization of captured ZigBee traffic and for its analyses for the education purposes or for the purpose of designing and planning ZigBee sensors networks.

This application is designed not to be used for real-time traffic monitoring because of the simple reason. If the goal is to use this software application during the Communication System or Computer Network courses for ZigBee protocol and packet analyses it will require presence of a RF capture device on each computer in the classroom in real-time monitoring case. This will increases the cost needed for implementing such approach in the curricula. The better approach is to use the RF capture device once for capturing data traffic in some testing environment and to use WSN Stat application off-line with saved capture files for analyses and experimentation.

The next guideline for application design is the idea to overcome the limits of displaying packet details by SmartRF™ Packet Sniffer and to provide extended statistical data analysis and data visualization. On the other hand, this software tries to cut costs and save funds comparing to buying some professional WSN packet analyzes tools.

The structure of the application is presented in Fig. 2.

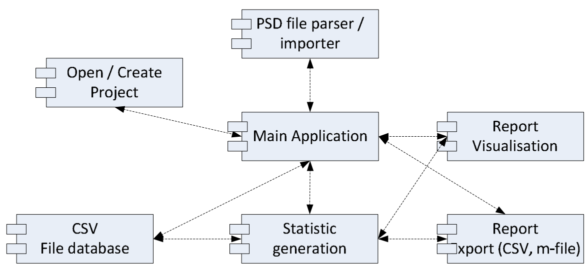

Figure 2. Component diagram of WSN Stat application

The parsed data are stored in CSV files. The first file keeps records of each packet captured and the second file keeps overall statistics of analyzed network traffic. The application's main features are: support for opening of projects, adding PSD files to be parsed in the project, as well as entering the data not present in the PSD files, such as channel used for communication, location marker or remark are also supported. The selection of the criteria for making statistic is key operation. In this case the criterion is position of the ZigBee node. The other criteria might be the communication channel, the device type, the packet length, the security type, data format sent, etc. In this stage of software development only one criterion per one statistic can be chosen.

The important feature of the software tool that helps in analyzing data is the visualization of the results. This application supports drawing of graphs showing statistical information. The additional feature of this application is that result can be exported to Matlab file format, enabling generation of Matlab type of graphs. The tabular data are exported in CSV file format. The application is tested in the experiment presented in the next section.

IV. Experiment

The experiment is performed at the institution's main building at the 1st floor. The layout of the building is given on Fig 3. The deployment of the nodes is presented in Table II. The position of the XBee coordinator, PC and USB CC2351 Dongle are presented as point 0. Other points (from 1 to 7) are the places where a mobile ZigBee node is placed for measurement. The same node is used in sequence for all measurement positions for this experiment.

TABLE II.
THE DISTANCE OF THE NODES IN THE EXPERIMENT

No	Position	Distance (m) approx.
0	Coordinator – Lab 28	-
1	Lab 28	1 m
2	Lab 28	10 m
3	Lab 29	6.5 m
4	Lab 29	18 m
5	Lab 27	12.5 m
6	Lab 24	26.5 m
7	Lab 20	31.5 m

As it was said before, from each position ZigBee node sent 2500 packets in 100 ms interval. The duration of data transmission and the distance of the measurement stations from the coordinator are given in Table II and Table III. The measured results by the position are presented in Table III as well.

The building layout is presented without office furniture. The locations used for this experiment are computer labs, so their furniture consists of computer tables, with computers and monitors and classroom chairs. This type of furniture, may affect the communication because both receivers and transceivers are located in some cases to the 130-140 cm from the floor. Moving objects, e.g. humans, also affect the performance of the network. This is non-controlled environment [10], because movement in the building is allowed during the experiment.

The experiment is performed in the late afternoon, so the number of people in the building is minimal, and their influence on the results of the experiment is also minimal. The walls are approx. 35 cm tick and built of bricks. The outer wall of the building is made of concrete, while windows make larger percentage of the outer walls.

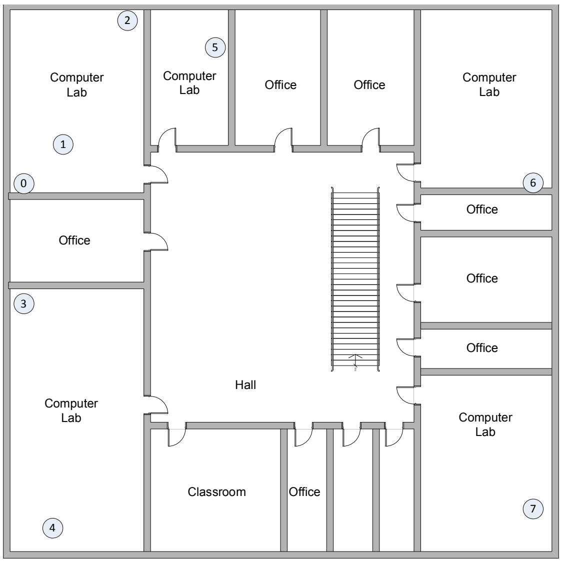

Figure 3. The building 1st floor layout and ZigBee station deployement

The Fig. 3 presents the space between position 0 and other positions, e.g. ZigBee node placed at the position 4 communicates with the coordinator at the position 0 through two walls. The communication between the coordinator (position 0) and ZigBee node at position 7 goes through 4 walls and so on.

V. Results

Numerical results presented in Table III show several parameters. The first presented parameter is the number of bytes sent during the transmission. Although it is said that experiment involves sending of 2,500 packets with length of 127 Bytes from ZigBee router (node) to Coordinator, the number of approximately 10,000 packets per test are due to numerous ACK and CMD frames sent during the test. The small difference of packet numbers per test come from the different time of the experiment duration, so a certain number of packets is sent before or after generated transmission.

The next presented parameter is PER (Packet Error Rate) in percent. The average, min and max RSSI (Received Signal Strength Indication) are given for each place measured in dBm. All numerical results are given in Table III. The throughput in Kbps and the time of transmission are presented in the last two columns. All presented values are important for analyses to determine the quality of data transmission from the given place.

TABLE III.
THE RESULTS OF ZIGBEE NODES MEASUREMENTS FOR THE DIFFERENT POSITIONS

Position	Packets Sent (No.)	PER (%)	RSSI Avg. (dBm)	RSSI Max. (dBm)	RSSI Min. (dBm)	Throughput (Kbps)	Test Time (min)
1	10087	0.73	-45.98	-40	-55	7.2	8.02
2	10865	1.51	-83.95	-66	-97	6.64	9.33
3	10091	0.66	-66.01	-64	-70	7.2	8.1
4	10153	1.82	-89.05	-82	-97	7.36	7.86
5	10126	0.41	-70.09	-66	-78	7.28	8.09
6	10231	1.29	-77.27	-71	-88	7.36	8.03
7	10171	1.7	-92.54	-91	-96	7.36	7.81

All presented results are similar, meaning that all potential positions for placement of ZigBee nodes are acceptable. It should be taken into consideration that position 2, although located in physically the same room as Coordinator station, has the problem with the large number of obstacles between them. Computer tables, computers, monitors and classroom chairs are placed between them. The transmitter and the receiver are both located lower than others, on 130-140 cm from the floor and the furniture and the equipment obstructed the communication.

The most important parameters for the link quality determination are extracted and presented in figures separately. Fig. 4 presents RSSI average value by measurement position. It shows that the stronger signal is in position 1 on the distance of 1 m from the coordinator.

The weakest signal are at position 7 and position 4, placed at the furthest positions from the coordinator and at the positions placed behind two or more walls. The position 2 also has low RSSI value, and the reason is that it is placed in the same room as the coordinator at 10m distance, but between two nodes are placed several computer tables with computers and monitors and classroom chairs and they might cause low RSSI average values.

That problem will be solved in the case of the real ZigBee network, because in that case the ZigBee nodes will be mounted on the walls avoiding the furniture and equipment as obstacles. The problem with the human body interference will be solved by mounting the nodes on the wall as well. It is important to note that these measurements are made for the demonstration of software application and not for analyses of ZigBee network performance, so the measurement results will not be discussed in more details.

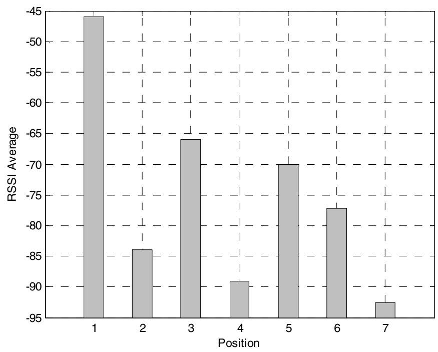

Figure 4. The RSSI [dBm] for 7 different positions of ZigBee station

The next parameter is Packet Error Rate (PER) presented as the percentage of the number of packets with errors comparing to the total number of packets sent. The largest PER are again at positions 7, 4 and 2. The position 6 also has higher PER value. In all cases, PER values are in the acceptable limits.

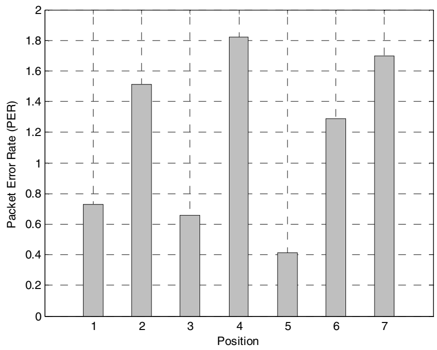

Figure 5. The PER [%] for 7 different positions of ZigBee station

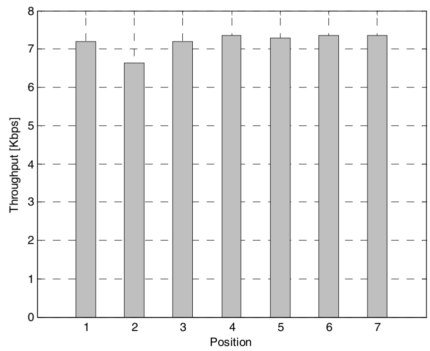

Figure 6. The Throughput [Kbps] for 7 different positions of ZigBee station

The figure (Fig. 6.) presents the throughput. This time, all positions, have similar values. Considering the high rate of packet transmission it seems that obstacles and node distance did not affect the link quality for all positions. So, the sensor data transmission will be possible from all tested positions with the same high quality.

Finally, the statistical results concerning RSSI are visualized in function of time. The RSSI average values for each second of transmission are presented graphically to analyze the changes in RSSI and to determine the stability of communication. Both graphs show that there are no significant differences between signal strength during the test.

It should be also taken into consideration that the test duration was not too long (from 7 to 10 minutes) and that better estimation of link quality can be given in the longer period of time.

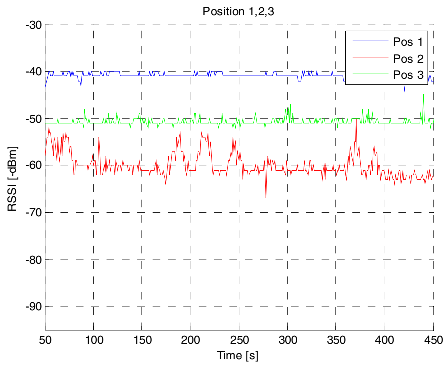

Figure 7. Avg RSSI [-dBm] by seconds for positions 1, 2 and 3 of ZigBee station

Because of better view to the RSSI and the difference for the various positions, the presentation of RSSI in function of time is presented with two separate graphs (Fig. 7 and Fig. 8) although by default, application gives this statistics in one graph for all positions, and additional graphs for each position separately.

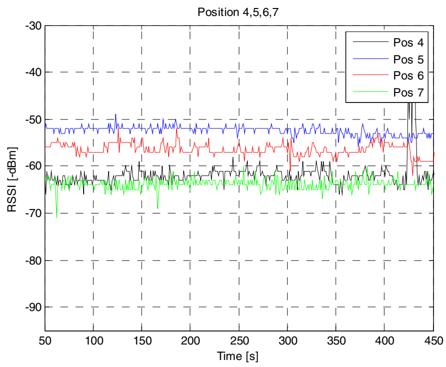

Figure 8. Avg RSSI [dBm] by seconds for positions 4, 5, 6 and 7 of ZigBee station

VI. Conclusion and Further Work

The approach in developing and using application for ZigBee protocol analyzes is presented in this paper. This application is designed not to replace the existing commercial wireless protocol analyzers, but to make easy-to-use, low-cost application suitable for integration in Computer Networks and Communication Systems course.

The presented application WSN Stat is designed to be used for parsing previously captured ZigBee traffic. Application is designed for parsing SmartRF™ Packet Sniffer PSD files. For demonstration and validation of WSN Stat application, the supporting platform is built upon microcontroller unit Arduino UNO Rev3 and XBee wireless modules.

The experiment, where the presented platform is used to simulate transmission with ZigBee protocol, is also described in this paper. Finally, the statistical results of measured traffic are presented. Presented results are used to evaluate performance of ZigBee network in the same way as it can be used during classes teaching students to analyze main issues concerning ZigBee standard.

Also, presented results can be used in pre-implementation and testing period of ZigBee based Wireless Sensor Networks. Experimental results showed that planned locations for ZigBee nodes allow quality communication so the network can be deployed as it is planned.

This application is a part of an ongoing project and still has space for improvement. One of its major weaknesses is its heavy dependence of Texas Instruments PSD capture files and USB listening RF device (CC2351 or similar). Nevertheless, the support for variety of other RF sniffer devices will not be priority in the further application development.

The priority will be development of module for deep insight of ZigBee packets, similar like in Wireshark (free and open-source packet analyzer). The next priority will be expansion of this tool for analyzing other protocol packets such as IEEE 802.15.4 and Bluetooth Low Energy, which is currently in testing phase.

Acknowledgment

This research is supported by Ministry of Education and Science of the Republic of Serbia under the project number TR32044 “The development of software tools for business process analysis and improvement”, 2011-2014.

References

[1] Robert Faludi, Building Wireless Sensor Networks, O’Reilly Media, Inc., USA, 2011.

[2] Edgar H. Callaway, Wireless Sensor Networks: Architectures and Protocols, CRC Press, 2004.

[3] Ubilogix International, Inc., Official Web Site, http://www.ubilogix.com/products/ubiqua, retrieved May 2014.

[4] Perytons Ltd., Official Web Site, http://www.perytons.com, retrieved May 2014.

[5] SmartRF™ Packet Sniffer User Manual, Texas Instruments Incorporated, 2010.

[6] T. AL-Kadi, Z. AL-Tuwaijri, A. AL-Omran, “Arduino Wi-Fi Network Analyzer”, Procedia Computer Science, Vol. 21, 2013, pp 522-529, http://dx.doi.org/10.1016/ j.procs.2013.09.073.

[7] Q. Abu Al-Haija, H. Al-Qadeeb, A. Al-Lwaimi, “Case Study: Monitoring of AIR Quality in King Faisal University Using a Microcontroller and WSN”, *Procedia Computer Science*, Vol. 21, 2013, pp 517-521, http://dx.doi.org/10.1016/ j.procs.2013.09.072.

[8] R. Hussin, R.C. Ismail, E. Murrali, A. Kamarudin, “Wireless Traffic Light Controller for Emergency Vehicle through XBee and Basic Stamp Microcontroller”, *Procedia Engineering*, Vol. 41, 2012, pp 636-642, http://dx.doi.org/10.1016/j.proeng.2012.07.223.

[9] Kuang-Yow Lian, Sung-Jung Hsiao, Wen-Tsai Sung, Intelligent multi-sensor control system based on innovative technology integration via ZigBee and Wi-Fi networks, Journal of Network and Computer Applications, Volume 36, Issue 2, March 2013, pp 756-767, http://dx.doi.org/10.1016/ j.jnca.2012.12.012.

[10] William R. Shadish, Thomas D. Cook, Donald T. Campbell, Experimental and Quasi-Experimental Designs for Generalized Causal Inference, Houghton Mifflin, 2002.