Research Article

Experimental Evaluation of ZigBee-Based Wireless Networks in Indoor Environments

Jin-Shyan Lee¹ and Yuan-Ming Wang²

¹ Department of Electrical Engineering, National Taipei University of Technology, Taipei 10608, Taiwan

² Information and Communication Research Labs, Industrial Technology Research Institute, Hsinchu 31040, Taiwan

Correspondence should be addressed to Jin-Shyan Lee; jslee@mail.ntut.edu.tw

Received 13 September 2012; Revised 21 January 2013; Accepted 21 January 2013

Academic Editor: Christos Bouras

Copyright © 2013 J.-S. Lee and Y.-M. Wang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

ZigBee is an emerging standard specifically designed for wireless personal area networks (WPANs) with a focus on enabling the wireless sensor networks (WSNs). It attempts to provide a low-data rate, low-power, and low-cost wireless networking on the device-level communication. In this paper, we have established a realistic indoor environment for the performance evaluation of a 51-node ZigBee wireless network. Several sets of practical experiments have been conducted to study its various features, including the (1) node connectivity, (2) packet loss rate, and (3) transmission throughput. The results show that our developed ZigBee platforms could work well under multihop transmission over an extended period of time.

1. Introduction

For the past few decades, in order to access networks and services without cables, wireless communication is a rapid growing technology to provide the flexibility and mobility [1]. Obviously, reducing the cable restriction is clearly one of the benefits of wireless with respect to cabled devices. Other benefits include the dynamic network formation, easy deployment, and low cost in some cases. In general, the wireless networking has followed a similar trend due to the increasing exchange of data in services such as the Internet, e-mail, and data file transfer. The capabilities needed to deliver such services are characterized by an increasing need for data throughput. However, other applications in fields such as industrial [2], vehicular, and home sensors have more relaxed throughput requirements. Moreover, these applications require lower power consumption and low complexity wireless links for a low cost (relative to the device cost). ZigBee over [3–5] IEEE 802.15.4 [6] is the one that addresses these types of requirements.

Based on the IEEE 802.15.4 standard, ZigBee is a global specification created by a multivendor consortium called the ZigBee Alliance. Whereas 802.15.4 defines the physical (PHY) and medium access control sublayer (MAC) layers of

an application, ZigBee defines the network and application layers, application framework, application profile, and the security mechanism. ZigBee provides users in specific applications with a simple, low-cost global network that supports a large number of nodes with an extremely low power drain on the battery. The ZigBee stack architecture, as shown in Figure 1, is based on the standard open systems interconnection (OSI) seven-layer model but defines only those layers relevant to achieving functionality in the intended market space.

Wireless links under ZigBee can operate in three license free industrial scientific medical (ISM) frequency bands, including 868 MHz in Europe, 915 MHz in the USA and Australia, and 2.4 GHz in most jurisdictions worldwide. Data transmission rates vary from 20 to 250 kbps. A total of 27 channels are allocated in 802.15.4, including 1 channel in the 868 MHz, 10 channels in the 915 MHz, and 16 channels in the 2.4 GHz band. The ZigBee network layer supports star, tree, and mesh topologies. Each network must have one ZigBee coordinator (ZC) to create and control the network. In the tree and mesh topologies, the ZigBee routers (ZRs) are used to extend the communication range at the network level. ZigBee devices are of three types, besides the mentioned ZC and ZR. For sensing applications, the sensors are usually

FIGURE 1: The ZigBee over IEEE 802.15.4 protocol stack.

FIGURE 2: Computing and networking module for ITRI SCAN-ZB32.

programmed as ZigBee end devices (ZED), which contain just enough functionality to communicate with their parent node (either a ZC or ZR) and cannot relay data from other devices.

Up to now, many performance analyses for wireless sensor networks (WSNs) have been proposed [7–10]. Kohvakka et al. [7] studied the performance of ZigBee-based sensor networks based on a cluster-tree topology. The study in [9] also analyzed the reliability of IEEE 802.15.4 cluster trees. For beacon-enabled IEEE 802.15.4 WPAN, Chen et al. [10] evaluated the performance for industrial monitoring applications using OMNeT++ tool. On the other hand, the evaluation of IEEE 802.15.4 for wireless medical applications was performed in [8] via systematic simulations. However, most of the previous work is based on simulation rather than practical experiments. Lee [11] established a five-node sensor

network to experimentally evaluate the IEEE 802.15.4 performance with various features. In [12, 13], the performance of realistic IEEE 802.15.4 was evaluated and compared with existing simulation models in NS-2. They also analyzed the coexistence of IEEE 802.15.4 with IEEE 802.11 and Bluetooth which are operating in the same 2.4 GHz ISM band. Cano-García and Casilari [14] presented an empirical study of the effects of the channel occupation on the consumption of actual ZigBee/IEEE 802.15.4 motes. However, the WSN literature provides few experimental analyses for ZigBee-based sensor networks.

This paper evaluates the performance of a ZigBee-based wireless network under multihop transmission over an extended period of time. The ITRI ZBnode [15, 16], developed by the Industrial Technology Research Institute (ITRI), is applied and deployed in an indoor environment. Totally, 51

FIGURE 3: Sensing module for ITRI SCAN-ZB32. A prototype area is provided for other extended sensing functions.

FIGURE 4: Power management mechanism for ITRI SCAN-ZB32.

sensor nodes are deployed in a hallway and a room. One of the nodes is a coordinator, and all the others are sensors with the routing capability. Each sensor regularly transmits packets to the coordinator during a long-term operation time. The practical performance of this wireless sensor network is evaluated in terms of (1) node connectivity, (2) packet loss rate, and (3) transmission throughput. This paper is an extension of our previous work [17] with newly added experiments on the node connectivity.

The rest of the paper is organized as follows: Section 2 introduces the developed ZigBee devices, and then experimental configurations are described in Section 3. The experimental results are shown in Section 4, and finally, Section 5 gives the conclusions.

2. Proposed Platform: ITRI ZBnode

Based on an IEEE 802.15.4 radio module and an ARM processor, the developed ZBnode [15, 16] is an autonomous, light weight wireless communication and computing platform. As ITRI performed a project for developing a small device with sensing, computing, and networking (SCAN) capabilities, the first version of ZBnode was designed and implemented. Since the first version used a powerful 32-bit processor, it was named SCAN-ZB32. Since individual sensor configurations are required depending on applications, the ZB32 device has

no integrated sensors for general-purpose applications. In the development stage, ZB32 can be used with various serial devices through predetermined sockets. The serial devices include sensors, actuators, power chargers, RFID readers, and even user interface components.

2.1. ZigBee Hardware Design. The ZBnode hardware adopts a 32-bit RISC processor which features an ARM 720T CPU core running up to 70 MHz, as shown in Figure 2. Several integrated peripherals including timers, counters, 10-bit AD converter, USB, UARTs, LCD controllers, infrared communications, controller area network (CAN) interfaces, pulse-width modulation (PWM), and JTAG for debugging are built around the processor. The external memory comprises an in-system programmable Flash ROM (16 MB) and an SDRAM (16 MB). To implement a visual user application interface, ZB32 provides four buttons and five LEDs. An IEEE 802.15.4 compliant RF transceiver, Chipcon CC2420, is connected to an on-board 2.4 GHz chip antenna and to one of the serial peripheral interfaces of the processor. Moreover, the RF module can be switched to an external antenna via a standard SMA (subminiature type A) connector for a better performance. The sensing module, where several sensors are supplied for detecting environment conditions (such as temperature, humidity, and light), is shown in Figure 3. For general-purpose development, a prototype area is provided

FIGURE 5: Experimental network structure.

FIGURE 6: The deployment layout for sensor nodes.

for other extended sensing functions. In addition, a sounder which could be used as an alarm notification is also available. About the energy source, the ZBnode can be powered with an AC adapter or an Li-ion battery with a DC input range from 3.5 to 5.5 V.

2.2. Power Management Mechanism. In in-door applications to environmental monitoring, for the most time only the sensing module is active while the computing and communication modules are power down. As shown in Figure 4, ZB32 uses a separate power-gating circuit to perform the power management. The power streams of the computing module (including ARM processor, ROM, and SDAM), communication module (IEEE 802.15.4 RF), and sensing module (temperature, humidity, and light sensors) when not in use can be completely disconnected from the energy source. A complex programmable logic device (CPLD) is adopted to control power stream via three power switches and also to handle power management requests in the developed ZBnode platform. The power management requests could be issued

from either the computing, communication, or the sensing module. For example, as soon as the sensing module detects an abnormal situation, it would signal the CPLD to wake up the computing module to process such events.

2.3. ZigBee Protocol and Software Stack. In our design, the SCAN-ZB32 platform uses an ARM Linux kernel 2.4. In addition, the system software is a Linux-based framework written in ANSI C. For general-purpose development, an open ARM-Linux compiler suite is applied. Within this framework, applications are typically partitioned into (1) a ZigBee protocol layer for communicating with the IEEE 802.15.4 front end through commands/events and keeping track of point-to-point connections with individual state machines, (2) a command line terminal for debugging and control, and (3) a user-defined application object. The whole application is defined at compile time and then programmed into the flash memory using an Xmodem terminal via the RS-232 port if the Linux kernel is loaded in advance. In addition, an in-system programmer through the JTAG interface is

FIGURE 7: Sensor nodes attached to the ceiling of the hallway and room.

FIGURE 8: Node status and network topology monitored via internet.

provided to download the application program. Furthermore, the developed protocol stack is embedded into the driver with convenient APIs available for application developers.

3. Experimental Configurations

3.1. Performance Metrics. In this paper, the node connectivity, packet loss rate, and transmission throughput are used as the performance metrics.

- (i) *Node Connectivity.* In our work, if a node can send out packets to the coordinator within a specified period of time, the node is defined as “connected.” Otherwise, the node is “disconnected.” Obviously, if more nodes in a sensor network are connected, the network is more stable.

- (ii) *Packet Loss Rate.* The packet loss rate of a node is defined as the number of packets lost by the coordinator divided by the number of packets transmitted by the node. The less a packet loss rate is, the better a network performance is. Moreover, the ZigBee standard provides an optional acknowledged service in application support sublayer (APS) for reliable transmission.

- (iii) *Transmission Throughput.* For simplicity, the transmission throughput is measured under a two-hop communication (from a sensor to the coordinator) in our experiments. In a specific period, the more packets received by the coordinator results in a higher transmission throughput.

3.2. Node Deployment. The experimental network structure is presented in Figure 5. The coordinator stores data received

FIGURE 9: Sniffed packet: the packet size used for performance analysis is 91 bytes.

FIGURE 10: Average number of disconnections with varied time intervals.

from each node in an MySQL database of a gateway server via RS-232 port. Meanwhile, there is a sniffer near the coordinator (within one hop) to show and record the over-the-air data.

Figure 6 shows the deployment layout for sensor nodes. There are totally 51 sensor nodes in the network, of which one is the coordinator (node number C) and the others are sensors with the routing capability (node number 1–50). Nodes 3–29 are attached to the ceiling of the hallway. The coordinator and nodes 1, 2, and 30–50 are in the same room due to the security policy. In our experiments, each node is equipped with an AC power source for long-term operation time. Figure 7 shows several sensor nodes attached to the ceiling of the hallway and rooms. Figure 8 draws the node status and network topology monitored via internet using a web browser. Each circle respects a sensor, and the coordinator is the root node. Obviously, a cluster tree network with multihop transmission can be successfully formed.

3.3. Transmitted Packets. Figure 9 shows the content of sniffed data packets during transmission. The APS overhead in packets used for analyzing network performance is 12 bytes (8 bytes for destination address and 4 bytes for serial number), and the APS payload in a packet is 64 bytes. Moreover, with the overhead of other protocol layers (network, MAC, and physical layers), the total data size of a packet is 91 bytes.

4. Experimental Results

4.1. Node Connectivity. In this experiment, the transmission rate is 1 packet per 10 seconds for each node, and the operation time is seven days. As defined before, if a node can send out packets to the coordinator within a specified period of time, the node is defined as connected. Figure 10 shows the average number (50 nodes) of disconnections for varied specified period of time (20 seconds to 1 minute). Obviously, as the specified period of time is tight as 20 seconds, the number of

FIGURE 11: Number of disconnections for each node with varied time interval.

FIGURE 12: Average packet loss rate with varied operation time.

disconnections is the highest. Also, the fact that the average disconnection of 30 seconds is significantly less than that of 20 seconds shows that sensors with the routing functions are able to retransmit packets via APS ACK mechanism.

Figure 11 shows the number of disconnections for each node with varied time interval. According to the empirical results, when the time interval changes from 20 seconds to 30 seconds, the number of disconnections between the coordinator and each sensor node significantly decreases.

4.2. Packet Loss Rate. The average packet loss rate (50 nodes) with varied operation time has been recorded as shown in Figure 12. In this experiment, the transmission rate is 1 packet per 10 minutes for each node with APS ACK. When the operation duration is 7 days, the packet loss rate has the highest value. Overall, the shorter an operation time is, the lower packet a loss rate is. However, the reason why the packet loss rate of 1-hour operation time is higher than that of 12-hour one is the fact that only a small number of packets are

transmitted of 1-hour operation time so that losing one packet increases the packet loss rate a lot.

Figure 13 shows the average packet loss rate with and without APS ACK. The operation time for this experiment is 1 hour. The maximum loss rate is 0.147% and 3.86% with and without APS ACK, respectively. The result shows that packet loss rate could be significantly decreased to almost zero by using the APS ACK mechanism.

Figure 14 shows the packet loss rate of each node with APS ACK. The operation duration for this experiment is 7 days. The experimental results show that the more hops to the coordinator, the higher packet loss rate will be.

4.3. Transmission Throughput. In this experiment, the transmitted packets are with and without APS ACK. The empirical results are shown in Table 1. If the APS ACK is applied, the coordinator totally received 1300 packets in 27 seconds, indicating the transmission rate is 24.65 Kbits/sec. As the transmission is without APS ACK, the result shows the

FIGURE 13: The effect of APS ACK to packet loss rates.

FIGURE 14: Packet loss rate of each node for seven-day transmission.

TABLE 1: Empirical results of transmission throughput.

	With APS ACK	Without APS ACK
Number of packets received by the Coordinator	1300	1097
Duration (second)	27	23
Transmission rate (Kbps)	24.65	24.42

coordinator receives fewer packets at the similar transmission rate.

5. Conclusion

This paper presents the practical performance of a ZigBee wireless network with multihop transmission in an indoor environment during a long-term operation time. Totally, 51 sensor nodes are deployed in a hallway and a room. Several sets of practical experiments are conducted to study its various features, including the (1) node connectivity, (2) packet loss rate, and (3) transmission throughput. The results show that our developed ZigBee platforms could work well

under multi-hop transmission over an extended period of time.

The overall goal of this paper is to contribute and help through realistic measurements towards the dimensioning of the sensor networks for future applications using ZigBee/IEEE 802.15.4 technology. The developed ITRI ZB-nodes are employed for realistic experiments. During our experiments, we found that the achieved transmission rate is around 25 kbps. Note that this is substantially below the nominal value of 250 kbps, and reasons may be due to the transmission overhead (such as frame headers), the CSMA-CA random backoffs, the presence of interframe spacing, and the concurrent transmission of multiple nodes. Note that the CSMA-CA mechanism in 802.15.4 automatically backs off initially when a transmission is imminent; that is, each data and command frame transfer will at least have one backoff. This would also reduce the performance.

Future work includes the measurements on different deployments to evaluate the impact of the deployment on the ZB32 platform's performance, including the evaluation of network parameters under a mesh (data flow) topology [18] and the power consumption under a node mobility condition

[19]; Moreover, a kind of performance comparison between the presented ITRI ZBnodes and other platforms (or results from network simulators or mathematic models) should be further investigated.

Acknowledgments

This research was supported by the Ministry of Economic Affairs, Taiwan, under Grant 100-EC-17-A-02-01-0617, and in part by the National Science Council, Taiwan, under Grant NSC-100-2221-E-027-010. The authors declare that there is no conflict of interests.

References

- [1] J. S. Lee, Y. W. Su, and C. C. Shen, "A comparative study of wireless protocols: bluetooth, UWB, ZigBee, and Wi-Fi," in *Proceedings of the 33rd Annual Conference of the IEEE Industrial Electronics Society (IECON'07)*, pp. 46–51, Taipei, Taiwan, November 2007.
- [2] J. S. Lee, C. C. Chuang, and C. C. Shen, "Applications of short-range wireless technologies to industrial automation: a ZigBee approach," in *Proceedings of the 5th Advanced International Conference on Telecommunications (AICT'09)*, pp. 15–20, Venice, Italy, May 2009.
- [3] C. H. Wu, H. S. Liu, Y. F. Lee, M. S. Wei, Y. J. Li, and J. S. Lee, "A gateway-based inter-PAN binding mechanism for ZigBee sensor networks," in *Proceedings of the 37th Annual Conference on IEEE Industrial Electronics Society (IECON'11)*, pp. 3808–3813, Melbourne, Australia, November 2011.
- [4] ZigBee Alliance, *ZigBee Document 053520r25: Home Automation Profile Specification*, ZigBee Alliance, San Ramon, Calif, USA, 2007.
- [5] ZigBee Alliance, *ZigBee Document 053474r17: ZigBee Specification*, ZigBee Alliance, San Ramon, Calif, USA, 2007.
- [6] IEEE 802. 15. 4, Wireless Medium Access Control (MAC), and Physical Layer (PHY), *Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)*, IEEE, New York, NY, USA, 2006.
- [7] M. Kohvakka, M. Kuorilehto, M. Hännikäinen, and T. D. Hämmäläinen, "Performance analysis of IEEE 802. 15. 4 and ZigBee for large-scale wireless sensor network applications," in *Proceedings of the 3rd ACM International Workshop on Performance Evaluation of Wireless ad hoc, Sensor and Ubiquitous Networks (PE-WASUN'06)*, pp. 48–57, Torremolinos, Spain, October 2006.
- [8] H. López-Fernández, P. Macedo, J. A. Afonso, J. H. Correia, and R. Simões, "Performance evaluation of a ZigBee-based medical sensor network," in *Proceedings of the 3rd International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth'09)*, pp. 1–4, London, UK, March 2009.
- [9] G. Anastasi, M. Conti, M. Di Francesco, and V. Neri, "Reliability and energy efficiency in multi-hop IEEE 802.15.4/ZigBee wireless sensor networks," in *Proceedings of the 15th IEEE Symposium on Computers and Communications (ISCC'10)*, pp. 336–341, Riccione, Italy, June 2010.
- [10] F. Chen, N. Wang, R. German, and F. Dressler, "Simulation study of IEEE 802.15.4 LR-WPAN for industrial applications," *Wireless Communications and Mobile Computing*, vol. 10, no. 5, pp. 609–621, 2010.
- [11] J. S. Lee, "Performance evaluation of IEEE 802.15.4 for low-rate wireless personal area networks," *IEEE Transactions on Consumer Electronics*, vol. 52, no. 3, pp. 742–749, 2006.
- [12] M. Petrova, J. Riihijärvi, P. Mähönen, and S. Labella, "Performance study of IEEE 802.15.4 using measurements and simulations," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC'06)*, pp. 487–492, Las Vegas, Nev, USA, April 2006.
- [13] K. Shuaib, M. Alnuaimi, M. Boulmalf, I. Jawhar, F. Sallabi, and A. Lakas, "Performance evaluation of IEEE 802. 15. 4: experimental and simulation results," *Journal of Communications*, vol. 2, no. 4, pp. 29–37, 2007.
- [14] J. M. Cano-Garcia and E. Casilari, "An empirical evaluation of the consumption of 802.15.4/ZigBee sensor motes in noisy environments," in *Proceedings of the International Conference on Networking, Sensing and Control (ICNSC'11)*, pp. 439–444, Delft, The Netherlands, April 2011.
- [15] J. S. Lee and Y. C. Huang, "Design and implementation of ZigBee/IEEE 802.15.4 nodes for wireless sensor networks," *Measurement and Control*, vol. 39, no. 7, pp. 204–208, 2006.
- [16] J. S. Lee and Y. C. Huang, "ITRI ZBnode: a ZigBee/IEEE 802.15.4 platform for wireless sensor networks," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, pp. 1462–1467, Taipei, Taiwan, October 2006.
- [17] J. S. Lee, Y. M. Wang, and C. C. Shen, "Performance evaluation of ZigBee-based sensor networks using empirical measurements," in *Proceedings of the IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER'12)*, pp. 58–63, Bangkok, Thailand, May 2012.
- [18] C. Antonopoulos and S. Koubias, "Congestion control framework for Ad-hoc wireless networks," *Wireless Personal Communications*, vol. 52, no. 4, pp. 753–775, 2010.
- [19] T. Sun, N. C. Liang, L. J. Chen, P. C. Chen, and M. Gerla, "Evaluating mobility support in ZigBee networks embedded and ubiquitous computing," in *Proceedings of the International Conference on Embedded and Ubiquitous Computing (EUC'07)*, vol. 4808 of *Lecture Notes in Computer Science*, pp. 87–100, Springer, 2007.