International Journal of Grid Distribution Computing
Vol. 8, No.6, (2015), pp.49-56
http://dx.doi.org/10.14257/ijgdc.2015.8.6.06

A Delay-aware Application Protocol for Wireless Sensor Networks

Jun-feng Gao, Hong-guo Zhang and Guang-yi Tang

School of Software, Harbin University of Science and Technology, Harbin, China
gaojf@hrbust.edu.cn, zhg07@163.com, tanggy818@126.com

Abstract

Recent advances in micro electro mechanical systems (MEMS), wireless communication, and network technology, wireless sensor networks (WSNs) have become technically and economically feasible. In most case, WSNs are usually applied to many applications that require unattended operation. However unlike traditional networks, WSNs are resource constrained distributed networks. The delay latency, uncertainty and dynamic reliability of the WSNs may strongly impact on the WSNs applications. The information content or meaning of sending and receiving data will happened frequency when exchanging data among these sensor nodes over the WSNs. In addition, the application protocol themselves for data exchange may also impact on the WSNs applications. An inappropriate protocol will result in serious inefficiency so as not to meet application requirement. In order to ensure the reliability, integrity, and the continuity of data operations on the WSNs, this paper presents a delay-aware application protocol for WSNs applications. This protocol is based on ZigBee protocol that is suited for remotely monitor and control because of its low-power. Simulation results show that the protocol is feasible for the wireless sensor network in terms of the reliability, integrity and continuity of data.

Keywords: *application protocol, data exchange, time synchronization, wireless sensor networks, sensor network applications*

1. Introduction

Wireless Sensor Networks are spatially distributed sensor networks, which are distinguished from traditional networks. The rapid deployment, self organization, low-cost and fault tolerance characteristics of WSNs make them a very promising sensing technique for applications. WSNs are generally comprised by a large number of devices, called sensor nodes. These sensor nodes are densely deployed at different locations to sense the physical conditions of the surrounding environment and process data [1]. Sensor nodes are generally comprised of four basic units: power unit, sensing unit, data processing unit and radio unit. Sensor nodes communicate over radio frequency (RF) in short-range and collaborate as a group to provide higher quality resolution for the monitoring area. WSNs were initially designed to facilitate military operations. With the rapid development of wireless technology, integrated circuit (IC), and micro electro mechanical systems (MEMS), WSNs applications have since been extended to environment, healthcare, traffic, and many other commerce and industrial areas.

However, unlike traditional networks, sensor nodes on the WSNs are highly resource-constrained in terms of energy, storage capacity, communication bandwidth and processing capacity [2]. Typically, battery-powered sensor nodes must operate for long periods of time. To minimize energy consumption, most of nodes should be in sleep mode at most of the time. On the other hand, changes in the physical environment where a network is deployed make also nodes experience wide variations in connectivity. Furthermore, sensor nodes have to organize themselves and work all together to implements essential functionality for WSNs applications [3]. Consequently, it is much

ISSN: 2005-4262 IJGDC

harder than communicating individual sensor node. As a result of the wide variety of requirements and characteristics for the WSNs applications, WSNs have attracted many interests in the research community.

Although many applications protocols for WSNs applications are proposed [4, 5], there is not yet a widely accepted application layer protocols for sensor networks. Furthermore unusual application requirements for WSNs make application layer protocol more challenging, due to the delay uncertainty, latency and dynamic reliability of the WSNs. In most setting, data exchange among sensor nodes over application layer protocols of the WSNs requires the information content or meaning of sending and receiving data not to be altered during the transmission. In additions, in distributed environments, the coordination of sensor nodes requires time synchronization that is one of the most important and fundamental issues. As a result, possible problems about data exchange should be addressed by the application layer to provide reliable data transmission for WSNs. Data exchange is mainly related to wireless technology, protocols and topology of WSNs.

The objective of this paper is to ensure reliable data exchange among sensor nodes in the WSNs. In order to guarantee the reliability, integrity and continuity of operations of each data processing, a delay-ware application protocol is proposed for the application layer of WSNs. This protocol is based on the application layer of ZigBee protocol and is specifically designed for monitoring environment in campus.

The remainder of this paper is organized as follows: Section 2 introduces related wireless technology, ZigBee protocol and topology of WSNs applications. Section 3 presents the characterization of the problem considered in this paper. Section 4 describes the main requirements considered for data exchange in the design process. Section 5 presents implementation the specification of the proposed protocol. Section 6 analyzes and discusses results of simulation. Finally, Section 7 concludes the paper and presents the future of our research.

2. Related Work

2.1. Topologies of ZigBee Network

Unlike other networks, there are only three different types of devices that are used in a ZigBee network, which are Coordinator, Router and End devices. However, only one Coordinator is existed in a ZigBee network, irrespective of the network topology. Coordinators control the whole ZigBee network, which is responsible to the formation and security of networks. The main task of Coordinators is to selects the network parameters used by the network such as radio frequency channel, initiate the network and allows other devices to join the network. In additions, Coordinators are also used as gateway, which connect local area network to the Internet. Coordinators are usually equipped with higher computation and storage capacities as compared to Router and End devices. Routers acts as intermediate nodes, and extend the range of networks. Router can be connected to an already existent network, and also allow other devices to connect it. The main task of Router is to relay messages from one node to another. End devices perform specific sensing or control functions, which are often battery-powered devices. End Devices cannot relay messages to other nodes in the network through them as compared to Coordinator and Router. In most cases, End Devices sleep in order to conserve power when not transmitting or receiving.

In a ZigBee network, each node must have unique identification, which is identified by means of IEEE address and network address. IEEE address is a 64-bit MAC address that is used to identify the device uniquely. Network address is a 16-bit address that is used to identify the node in the ZigBee network. Network address is also called the short address. When any node joins into a network, its network addresses is allocated by its parent node,

such as Router or Coordinator. However, Coordinator always is set to a fixed network address as 0x0000. The nodes on the different networks are able to have the same network address.

A ZigBee network supports three types of topologies: Star, Tree, and Mesh [6], and the corresponding diagrams are shown in Figure 1.

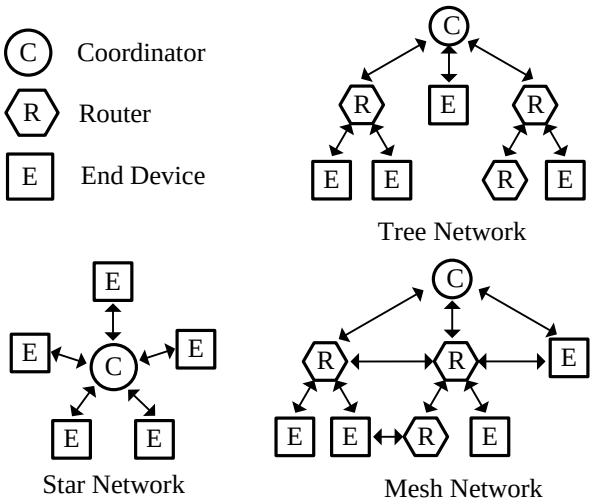

Figure 1. Topology of ZigBee Network

The Star topology is the simplest ZigBee network, which only consists of a Coordinator and a set of End Devices in the network. The Coordinator is a central node that is connected to all End Devices in the network. End Device can communicate only with the Coordinator and communicate with other End Devices indirectly via the Coordinator. Once the RF link fails between the Coordinator and the target End Device, there is no alternative route. In addition, the Coordinator can be a bottleneck that may result in congestion.

The Tree topology may consist of all kinds of devices in the ZigBee network. The structure of a Tree network has a root, branches, and leaves. Coordinator is a root node that is connected to its children, called Routers and End Devices. Routers are branches nodes whose children are End Devices. End Devices are leaves nodes that cannot have children. Only direct communication is valid between child node and its parent node. When communication is from one node to another, the message from the source node is relayed via its parent node. Like the Star topology, there is no alternative route while the RF link fails between the Routers and its parent node.

The Mesh topology has a tree-like structure in the ZigBee network with a root node, called Coordinator. However, the Mesh topology is more flexible in that Router nodes within range of each other can discover and communicate directly without involving parent nodes. Unlike the topology of Star and Tree, the Mesh network can find the best available route for a message when RF link fails. As a result, the issues about selecting an energy efficient route are more complex, which can be addressed by the routing algorithms.

2.2. Basic Architecture of ZigBee Network

The basic architecture of ZigBee protocol stack is summarized in Figure 2, which comprises mainly by four basic layers: physical layer, data link layer, network layer and application layer [7]. The basic layers of the ZigBee protocol stack are described below from bottom to top.

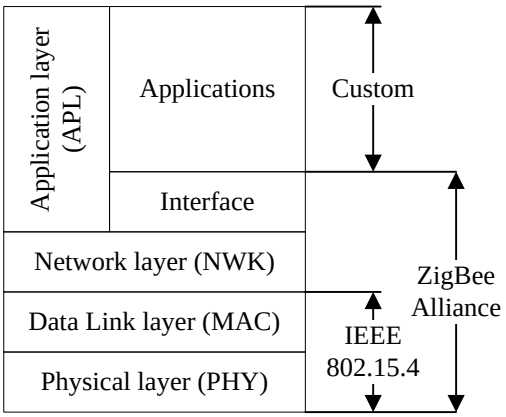

Figure 2. Basic Architecture of the ZigBee Protocol

The physical layer provides an interface to transmit a stream of bits over physical medium, which is responsible for frequency and power selection, modulation, and data encryption. This layer is defined by the IEEE 802.15.4 standard. Cost and power are two main metrics to evaluate the physical layer.

The data link layer provides the multiplexing of data streams, data frame detection and medium access control (MAC), which is responsible to determine who is allowed to access the media at any one time and assembles data packets or frames to be transmitted and disassembling received frames across physical layer. This layer is also defined by IEEE 802.15.4 standard, which can be referred to as IEEE 802.15.4 MAC.

The network layer provides packet forwarding through intermediate routers, which directs the process of selecting paths along which to send data in the network. This layer is defined by ZigBee Alliance that provides the glue between the applications and the IEEE 802.15.4 level.

The application layer provides a general set of application services and interface primitives for transaction management, which works with the lower layers to implement a programmer independently on any platform. Interface is defined by ZigBee Alliance, but application services may be customized by developer, which are established on the interface.

2.3. Time Synchronization

As in all distributed networks, time synchronization [8] is an important component for data fusion and clustering, which provides information about the ordering of events, causal relationships and correlations. Unlike traditional networks such as Internet, the WSNs are resource constrained networks. Therefore, many factors influencing time synchronization in the traditional networks also apply to the WSNs, such as reliability and latency. In addition, the time synchronization techniques have to be energy aware since the nodes on the WSNs are energy-constrained devices with limited processing and transmission power. Various synchronization methods to meet specialized requirement are from simple to complex for WSNs, many of which do not differ much from each other. No matter the method used, the major problem for synchronization protocols is the variance in the send time, access time, propagation time, and the receive time. Reference broadcast synchronization (RBS), timing-sync protocol for sensor networks (TPSN), and flooding time synchronization protocol (FTSP) are the major timing protocols currently in use for wireless networks. But these time synchronization protocols aren’t compatible and they result in a network partition.

3. Data Exchange Challenges and Issues in WSNs

Data exchange is used frequently in many applications that require the information content or meaning assigned to the data is not altered during the transmission process of sending and receiving data. However, data exchange in WSNs is very challenging due to its inherent characteristics that distinguish them from other networks. In the following, we will summarize some challenges of data exchange and design issues that affect data fusion and clustering in WSNs.

Reliability: Remote sensing data of sensor nodes should be reliably delivered to a sink node in time. Similarly, the command and queries from the sink node should also be sent to a target sensor node to assure sensor operation. Given wireless link mobility, congestion and errors, data packet loss would have happened anyway. Hence, some extra essential operations concerning reliable transmission must be applied to maintain the connectivity of nodes in the networks. As a result, to maximize the reliability may result in increasing substantially the energy consumption. The tradeoff between reliability and energy consumption should be considered when designing data exchange protocol for WSNs applications compared to more traditional communication networks [9].

Latency: Inevitably, this is a transmission delay to deliver a data packet from a source node to a sink node [10]. In general, larger latency should affect the efficiency of the WSNs applications more, which are strictly related to the network topology and the nodes density.

Network dynamics: There are two main components in a sensor network. These are the sensor nodes and sink nodes. They are organized themselves to form a dynamic network [11]. For energy considerations, the sensor nodes work in sleep mode. Hence, not all sensor nodes always keep connection with sink nodes on the WSNs. On the other hand, the sensor nodes work in harsh environments. As a result, some of the sensor nodes don’t work well or are damaged. Relaying data from or to these nodes is more challenging task.

4. Methodology of the Delay-Aware Application Protocol

As it well known, all kinds of sensing values arrive at destination nodes in disorder from their source nodes due to the network transmission delay [12, 13, and 14]. The source nodes of these senseing values can be identified by their MAC address because it is unique on the sensor network. Yet it is difficult to distinguish these sensing values from the same sensor or different sensor. Moreover, it is unknown whether these sensing values are the same order as sampling values queue, even though these sensing values comes from the same sensor. The sensing value lacks essential information during network transmission, such as acquisition time and sensor type. Therefore, the integrity and continuity of sensing value can’t be guaranteed.

But the application layer of ZigBee protocol stack lacks flexibility for different applications of the WSNs, although it provides a general set of services using an interface between the network layer (NWK) and the application layer (APL). As previously described, an application layer of ZigBee protocol can be customized partly. It is feasible to design a novel data frame that provides an application service. Depending on the applications, different extended information should be considered for sensor networks. To address above issue descript, a delay-aware application protocol was designed by us. This application protocol is based on the application layer of ZigBee protocol. Since the performance of the application protocol is closely related to the communication model and data exchange format, in this section we strive to descript the implementation methodology of the delay-aware application protocol in details.

4.1. Communication Mechanism

On the ZigBee network, there are three main devices. They are Coordinator, Router and End devices. Coordinator and Router are also called the sink nodes, and End devices belong to the sensor nodes. The sensor nodes are responsible for sampling the sensor signals, and the sink nodes maintain overall knowledge of the ZigBee network. In other words, the sink nodes are master device, and the sensor nodes are the slaver device. The master device is able to send command packets to the slaver device and receive data from the slaver device [15]. The slaver device is responded to the commands of the sink nodes, completes some operations, and sends back responder packets to the master devices. The communication mechanism between the sink nodes and the sensor nodes is shown as Figure 3.

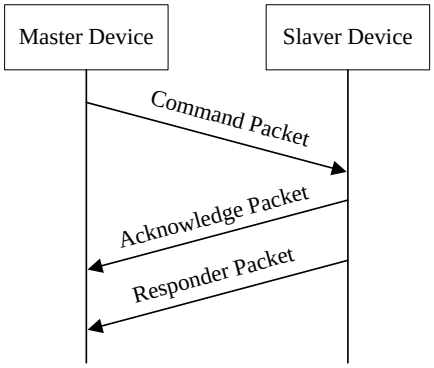

Figure 3. Communication Mechanism

Typically, the slaver device wakes from sleep mode to normal mode, and join into WSNs. Then the slaver device polls the master device at period. Once the data received from master device is the PREAMBLE, the slaver device begins to receive command packets. Otherwise, the slaver device polls the master device again. After receiving command packets successfully, the slaver device sends acknowledge packets (ACK) back to inform the master devices whether the command frames has been successfully received. Meanwhile, the slaver device begins to work as the command of the master device. Once finishing operation, the slaver device transmits responder packets to the master device. It is obvious that the sink nodes sequentially access the sensor nodes so as to keep the integrity and continuity of sensing value.

In the application protocol, command packets and responder packets uses data frame format, and acknowledge packets employ acknowledge frame format. These frames structure are described in details as the follow.

4.2. Data Frame Format

Data frame format are separately used to send command or receive data. They are designed to the same frame structure, and the structure specification of data frame is shown as Table 1. The number of the data frame can be exchanged, which is limited to 250 bytes.

Table 1. Data Frame Structure

Field	Length	Description
Preamble	2 bytes	The start flag of data frame, fixed as 0x00 and 0xFF.
Length	1 byte	Payload length, its value is equal to the number sum of all data fields bytes, which includes FLAG, TIME, TYPE and VALUE.
Flag	1 byte	State flag of the packets, its value is true when the command is valid.

Field	Length	Description
Timestamp	2 bytes	Timestamp of packets, recording the sampling time of data frame. The first byte indicates date, and includes year, month and day. The second byte indicates time, and includes hours, minutes and seconds. All data are encoded in BCD code.
Type_i	1 byte	Sensor type of the number i.
Value_i	2bytes	Sensor value of the number i.
Checksum	1 byte	Data payload checksum, its value satisfies the relation: [Length + Flag + Timestamp + Type_i + Value_i + Checksum] MODE 0xFF = 0x00.
Postamble	1 byte	The end flag of frame, fixed as 0x00.

4.3. Acknowledge Frame Format

The acknowledge frame is normally used for the synchronization of data frame, its structure specifications is listed in Table 2.

Table 2. Acknowledge Frame Structure

Field	Length	Description
Preamble	2 bytes	Start flag of frame, fixed as 0x00 and 0xFF.
ACK	2 bytes	Its value is 0xAA and 0x55 when frame has been successfully received; otherwise its value is 0xFF and 0xFF.
Postamble	1 byte	The end flag of frame, fixed as 0x00.

5. Analysis and Simulation

Table 3. Simulation Data of Sensor

order	1	2	3	4	5
Time	02:00:03	02:00:00	08:00:31	11:00:51	12:00:51
Temperature	10	9	12	20	22
Humidity	89	92	72	43	35

For additional timestamp, even though the network delay or data loss occurs on some sensor node, the queue of collected-data is not altered. On another hand, one sensor node can not impact on the other sensor nodes. As some of sensor node can not affect the overall performance of the WSNs. Consequently, we only consider the normal case in simulation that sensing value comes from the same sensor node. Table 3 lists the collected-data according to the receiving time sequence, which is from temperature sensor and humidity a sensor.

From Table 3, we found at first that the first data and the second data don't arrive as sampling sequence. Furthermore, some data of the sensor lose at some time for unknown factors. The most notable is delay-ware application protocol that was designed by us to provide redundant information. The correct order can be restored from timestamp.

6. Conclusion

This paper presents a novel delay-ware application protocol so as to address these problems on the integrity and continuity of collected-data for WSNs. To overcome the effect of network transmission delay, some essential redundant information was added into data frame structure of this protocol. These extra identifier fields mainly include flag, timestamp, and sensor type and sensor value. The collected-data can restored from these

fields, even if receiving data in disorder. In the future, we will enrich and improve this protocol and make it meet versatility to a certain extent.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E Cayirci, "A survey on sensor networks", IEEE Communications Magazine, vol. 40, no. 8, **(2002)**, pp. 102-114.
[2] W. Li, M. Bandai and T. Watanabe, "Tradeoffs among Delay, Energy and Accuracy of Partial Data Aggregation in Wireless Sensor Networks", The 24th IEEE International Conference on Advanced Information Networking and Applications (AINA), Perth, WA, **(2010)** April, pp. 20-23.
[3] S. Li-xin and D. He, "Research on WSN Routing Protocol based on Ant Colony Algorithm", Journal of Harbin University of Science and Technology, vol. 19, no. 6, **(2014)**, pp. 88-92.
[4] W. B. Heinzelman, A. P. Chandrakasan and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks", IEEE Transactions on Wireless Communications, vol. 1, no. 4, **(2002)**, pp. 660-670.
[5] Z. Pei, Z. Deng, B. Yang and X. Cheng, "Application-oriented wireless sensor network communication protocols and hardware platforms: A survey", IEEE International Conference on Industrial Technology (ICIT), Chengdu, China, **(2008)** April 21-24.
[6] A. Wheeler, "Commercial Applications of Wireless Sensor Networks Using ZigBee", IEEE Communications Magazine, vol. 45, no. 4, **(2007)**, pp.70-77.
[7] W. Wang, G. He and J. Wan, "Research on Zigbee wireless communication technology", 2011 International Conference on Electrical and Control Engineering (ICECE), Yichang, China, **(2011)** September 16-18.
[8] F. Sivrikaya and B. Yener, "Time synchronization in sensor networks: a survey", IEEE Network, vol. 18, no. 4, **(2004)**, pp. 45-50.
[9] K. Islam, W. Shen and X. Wang, "Wireless Sensor Network Reliability and Security in Factory Automation: A Survey Systems", IEEE Transactions on Man, and Cybernetics, Part C: Applications and Reviews, vol. 42, no. 6, **(2012)**, pp. 1243-1256.
[10] M. Drieberg, V. S. Asirvadam and F-Chun Zheng, "Accurate Delay Analysis in Prioritised Wireless Sensor Networks for Generalized Packet Arrival", IEEE Wireless Communications Letters, vol. 3, no. 2, **(2014)**, pp. 205-208.
[11] M. Asim, M. Yu, H. Mokhtar and M. Merabti, "A Self-Configurable Architecture for Wireless Sensor Networks", Developments in E-systems Engineering (DESE), London, UK, **(2010)** September 6-8.
[12] C.-Tsun Cheng, C. K. Tse and F. C. M. Lau, "A Delay-Aware Data Collection Network Structure for Wireless Sensor Networks", IEEE Sensors Journal, vol. 11, no. 3, **(2011)**, pp. 699-710.
[13] G. B. Eslamlu, M. Sabaei and M. Fereydooni, "A new delay constraint topology control algorithm in WSN", 2012 International Conference on Innovations in Information Technology (IIT), Abu Dhabi, **(2012)** March 18-20.
[14] C T Cheng, H Leung and P. Maupin, "A Delay-Aware Network Structure for Wireless Sensor Networks With In-Network Data Fusion", IEEE Sensors Journal, vol. 13, no. 5, **(2013)**, pp. 1622-1631.
[15] J. Gao, H. Huang and F. Wang, "A Lightweight Data Exchange Protocol for Campus Environmental Monitoring", 2014 International Conference on Frontiers of Internet of Things (FIT), Hsinchu, Taiwan, **(2014)** December 4-6.

