A ZigBee-Based Home Control System Using OSGi Management Platform

Ming-Chiao Chen and Cheng-En Wu

Department of Computer Science and Information Engineering,
National Taitung University, Taiwan, R.O.C.
joechen@nttu.edu.tw, canso.zero@hotmail.com

Abstract

Digital home industry provides a spectrum of services that standardizes digital home appliances and uses network technology allowing integrated control of these devices (thus enhancing the intelligence of home appliances). The application of digital home technology has extended to other domains such as environment, healthcare, security, and other areas. This study proposes replacing conventional wired networks with ZigBee wireless technology and integrating the OSGi (Open Services Gateway initiative) service platform to achieve the following objectives: (1) higher flexibility and mobility compared to wired control systems; (2) lower cost; (3) a mechanism for automatic recovery after node disconnection to adapt to highly mobile nodes; (4) support to web browser-based remote management functions, negating the need to install customer-end programs; and (5) an intelligent home control system with update functions. Through the management interface, users can control intelligent home appliances anywhere and anytime, which enhances convenience, home security, and energy conservation.

Keywords: *OSGi; ZigBee; IEEE 802.15.4; digital home*

1. Introduction

In the information era, the rapid advance of network technology and its ever decreasing cost have made home networks possible and gaining popularity. The prevalence of networks entails many revolutionary changes for business and new opportunities for applications. Among these newly network-enabled conveniences, digital home adds comfort and security through services provided by the interoperability of networkable home appliances, PC and mobiles [2, 17-19]. Currently, the digital home technology is no more limited within a home network but rather, by connecting to the Internet, is expanded to include applications to remote environment monitoring, medical care and security, etc. [14]. From the perspective of digital home, wireless networking techniques is inherent as mobile phones are so common that they are almost considered as standard home devices. Among the current available wireless networking techniques, ZigBee [1, 5, 10, 12, 15] demonstrates outstanding performance in data transmission, cost, standby period, and ease of deployment; furthermore, its coverage is extended from the home level to the farm level and its potential in outdoor remote sensing has been explored. Indeed, the applicable area of ZigBee is expanding.

As mentioned above, by connecting to the Internet, the convenience and effectiveness of a digitalized home can be further improved for users can control and monitor their houses remotely. However, there is no standard protocol for a Zigbee network to join the Internet. A common solution is to use a service platform as a bridge to connect both of them. In this paper, we propose the OSCi (Open Service Gateway Initiative) service platform as the linking bridge [7, 11] for it provides an excellent network management service and its interoperability makes it easy to be integrated into a residual gateway. As Figure 1 shows, users can thus connect to their home appliances or sensors via the Internet.

The rest of the paper describes the combination of the OSCi service platform and a ZigBee based control system for a remotely controllable digital home via the Internet. The paper is structured as follows. In Section 2, a brief introduction to the OSGi service platform and the ZigBee networking technology. While Section 3 outlines the system configuration, Section 4 presents the implementation of the system and is followed by a conclusion.

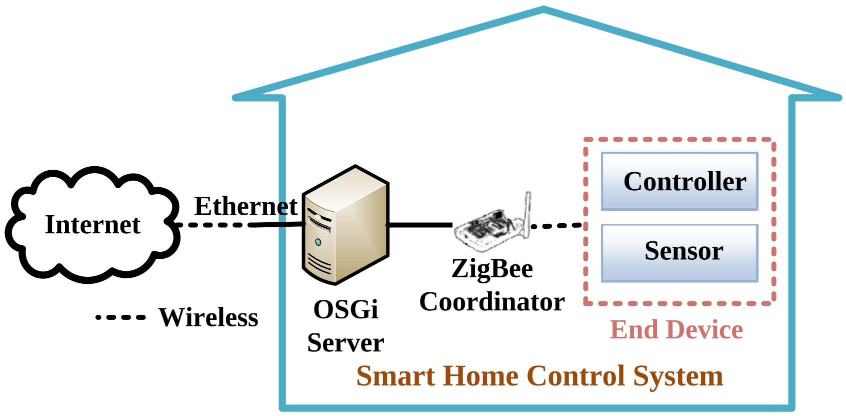

Figure 1. Digital Home Control System Architecture

2. Background

2.1. Development and Advantages of OSGi

OSGi (Open Service Gateway Initiative) was founded in March, 1999. One of the advantages of OSGi is that its development is supported continuously by many international companies such as IBM, Sony, Motorola and Nokia etc. Originally, it aims to provide an integrated information service platform which resides in an OSGi gateway. Its open architecture allows devices or programs which follow its specifications to communicate with each other.

2.2. OSGi Architecture and its Goal

The OSGi architecture consists of three components, namely, framework, bundle and service. A framework is layered on Java VM (Java Virtual Machine), bundles are applications composed of jar components which are executed in a framework, and a service is an interface for export or import provided by a bundle. A bundle downloaded from a server can be automatically installed and run in a framework and it will also register services which it requires and/or provides into the OSGi framework. Hence, a software provider can download to the OSGi gateway at the client side their application programs or value-added services for some local appliances according to the need of the user via the internet so that auto installation or dynamic update is possible. The OSGi architecture is shown in Figure 2.

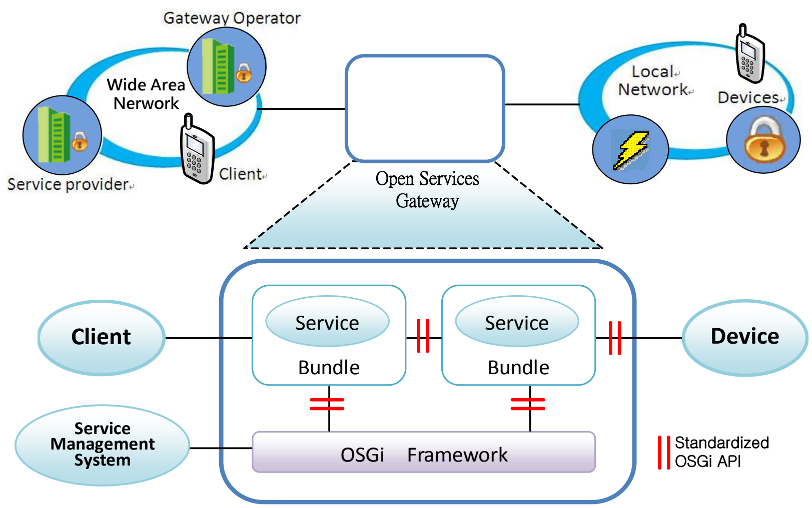

Figure 2. An OSGi Service Platform

2.3. Energy Saving Scheme and Transmission of ZigBee

ZigBee is wireless networking standard which saves the cost of wire lines and the time of wire laying and installation. Also, there is no need for renting a network service from an Internet service provider. As the cost of a ZigBee chip is low, consequently the cost of installation and maintenance of a Zigbee device is also cheap.

ZigBee uses different frequency bands and data transmission rates in different regions; in Europe the allocated frequency band is 868MHz (data transfer rate is 20kbps); in North America and Australia, 915MHz(data transfer rate : 40kbps), The rest regions 2.4GHz(data transfer rate 250kbps). The average working distance is 50m (depending on the environment, the working distance of a ZigBee node may range from 5m to 500m,). ZigBee specifies three types of data transmission： (a) periodic, an example is the pattern of sending the readings of a sensor (b) intermittent, e.g. a message sent by a switch being pushed, and (c) repetitive low latency: every a while, sending a message, e.g. a mouse .

2.4. ZigBee Network Structure

The medium access control layer of ZigBee supports various network topologies [13]. A ZigBee PAN (Personal Area Network) can accept a vast amount of connection requests (up to 65,500 parent nodes and 8.45 × 1018 end devices) so that a flexible network structure can be built. IEEE 802.15.4 [4] defines two types of node: FFD (Full-Function Device) and RFD (Reduced-Function Device). The roles of FFD and RFD in a network are shown in Figure 3. The structure of an RFD node is simpler than that of an FFD node but the cost is cheaper. In general, an RFD node can connect to a network for sending/receiving data but it will not accept connection requests issued from other nodes. Hence, it is an edge/leaf node of the network. As such, it is used as an end device. In comparison with an RDF node, an FFD node can accept connection requests issued from other nodes and it has a larger memory which can store information related to successful connection requests in its routing table; therefore, an FFD node can be used as a coordinator in a PAN. Obviously, applying and deploying RFD nodes adequately can reduce the cost of the network installation.

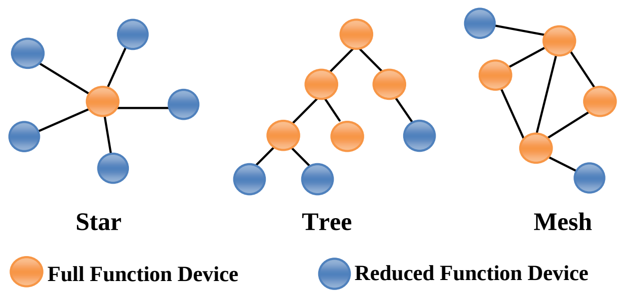

Figure 3. The Roles of FFD and RFD in Various Network Topologies

A coordinator is responsible for constructing and managing the ZigBee network. It acts as a bridge between end devices and the server and routes MAC packages received from end devices to the server. An end device is a leaf node placed at an end of the network, doing certain simple task such as broadcasting the readings of its sensor periodically.

3. System Architecture

As Figure 4 shows, the proposed digital home control system consists of a ZigBee network and an OSGi based control server. They are described in the following sections.

3.1. The ZigBee Network Configuration

The ZigBee network consists of three kinds of nodes, namely, a main coordinator, coordinators and end devices. The main coordinator is connected to the server via an RS-232 interface while the rest nodes communicate with each other wirelessly. The end device connects with a home appliance or sensor. It relays commands from the server to the appliance or reports to the server the measurement values of the related sensor. Fig 5 shows the flow charts of the connection initialization procedures of these three kinds of nodes. More precisely, Fig 5-A, 5-B and 5-C are the initialization flow charts for the main coordinator, the coordinator and the end device, respectively. Each node will search for the coordinator of the PAN of the node; if it exists, the node will try to connect to the coordinator by sending an Associate request. When a node of the ZigBee network is disconnected, the node will try to reconnect to the network automatically.

When a node connects to a coordinator, the node becomes a child node of the coordinator and the coordinator is the parent node of the child node. A coordinator can connect to another coordinator, while it can also accept connection requests from other nodes. A node is identified by a unique 64 bit MAC address (aka extended address) in the process of connection or data transmission. In the proposed ZigBee network, the server is the root node whose MAC address is set to 0x0.

Figure 6 shows the data reception processing flow of the three types of node, which includes some mechanisms of redundant data package avoidance and linking state maintenance. From the charts, it is obvious that a coordinator is responsible for relaying packages. Like a router, each coordinator stores a routing table in which every successful parent-child node association is recorded. A coordinator will route a MAC package to its neighbor node according to its routing table. An end device mainly relays command packages from the server to its associated appliance or reports to the server the data reading packages from its related sensor. Through setting the destination MAC address of a package and packed in the MAC package routed via coordinators, the server can issue commands to a certain end device while an end device can send to the server the reading of a sensor. Figure 7 shows the format of the data MAC package designed by the authors. The first byte used to prevent receiving a redundant MAC package is followed by the source and the destination MAC addresses, both 16 bytes in length; then the data bytes follows.

In the memory of each coordinator stored is a routing table which records every parent-child association between the nodes of the subtree started from the coordinator node, in terms of MAC address. Whenever a successful node connection occurs, the node will issue a NodeAssociate MAC package to the server. Likewise, when a connection is broken, a NodeDisassociate MAC package is issued to the server. A coordinator which relays a NodeAssociate or Node Disassociate MAC package will extract the MAC addresses of the parent-child pair contained in the relayed MAC package and update its routing table accordingly. Figure 8 summarizes the process of updating the routing table of a coordinator. When a coordinator encounters with a NodeAssociateMAC package, it will insert a record of the MAC addresses of the parent-child nodes in its routing table. On the contrary, it will delete the corresponding record of the MAC address pair in the routing table that is also carried in the NodeDisassociate MAC package relayed. Moreover, the coordinator will remove recursively the subtree of the disassociated node as shown in Figure 9.

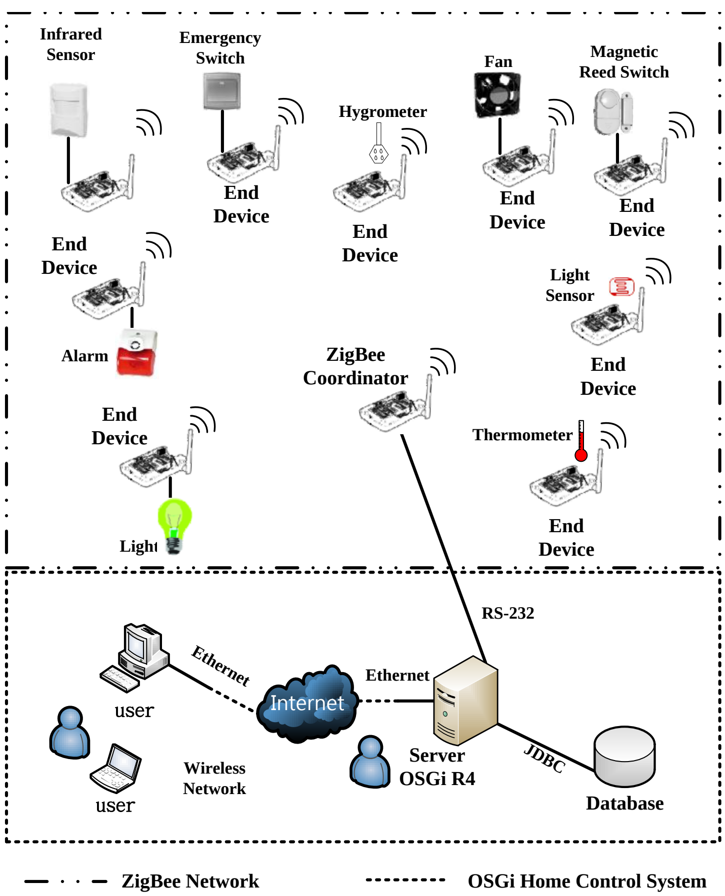

Figure 4. System Architecture

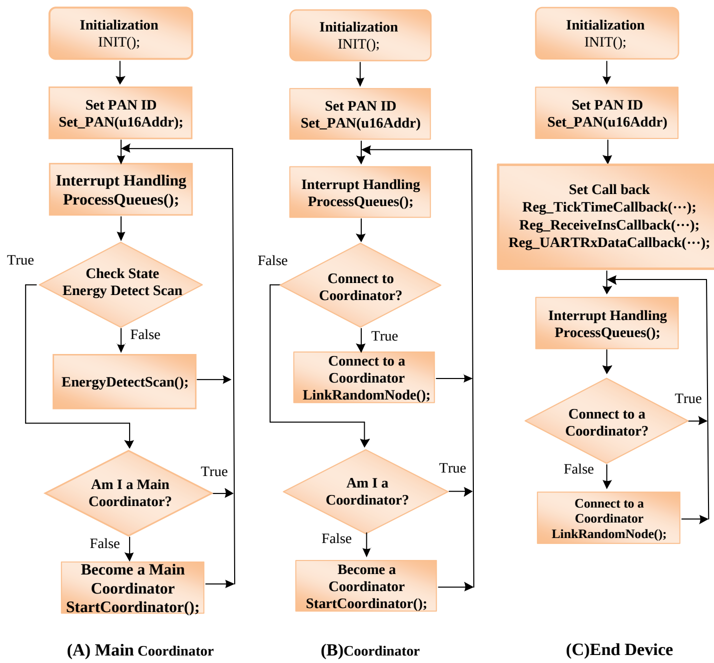

Figure 5. The Flow Charts for Initializing a Node Connection

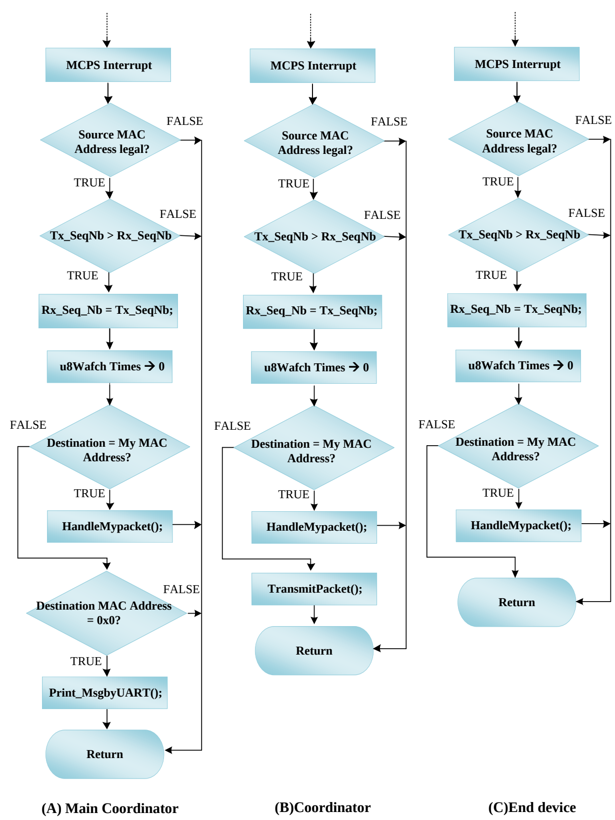

Figure 6. The Process Flow Chart of the Reception of a Package

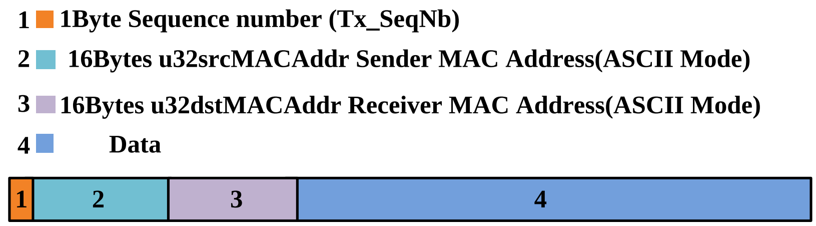

Figure 7. The Format of a Data Frame

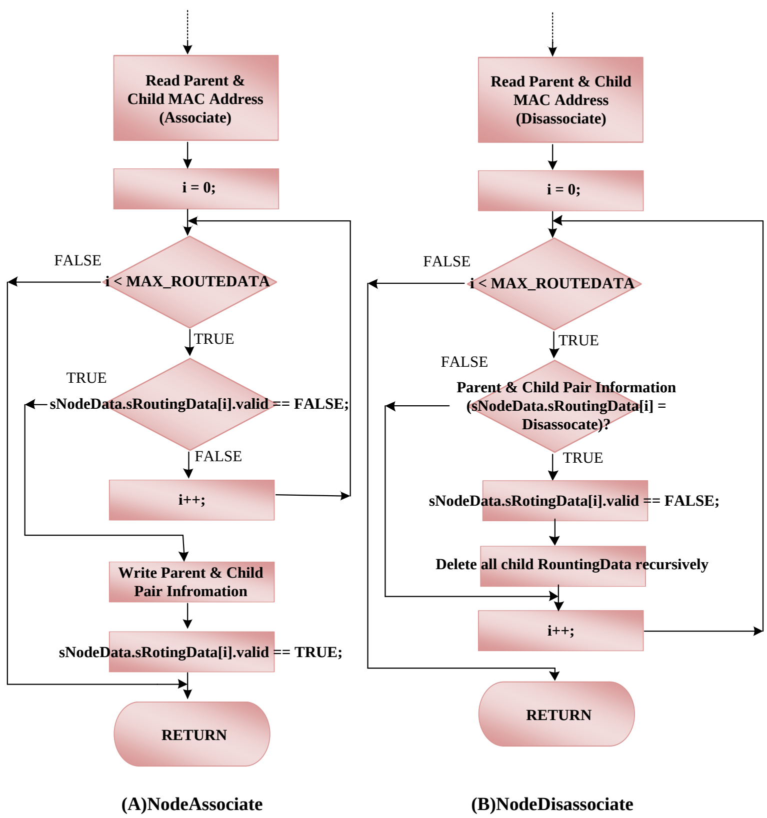

Figure 8. The Process Flow Chart of the Reception of an Associate/DisAssociate MAC Package

3.2. Configuration of An OSGi Home Control System

The functions of bundles are listed in Table 1. The following gives a more detailed account on these bundle functions.

3.2.1. ZKernel Bundle: ZKernel bundle is responsible for the two tasks: (1) monitoring and regulating home appliances. It will connect to the DBMS and record the setting values and appliance events into a database. It also controls the operation modes of home appliances according to the conditions stored in the database. For example：a thermo Sensor will send its readings via the ZigBee network periodically or intermittently (for an emergency switch) to the ZKernel bundle; the ZKernel bundle will react immediately according to the readings and their predefined actions. (2) recording the status of each coordinator or end device node into the database. Hence, a user can monitor the appliances via the ZPanel bundle.

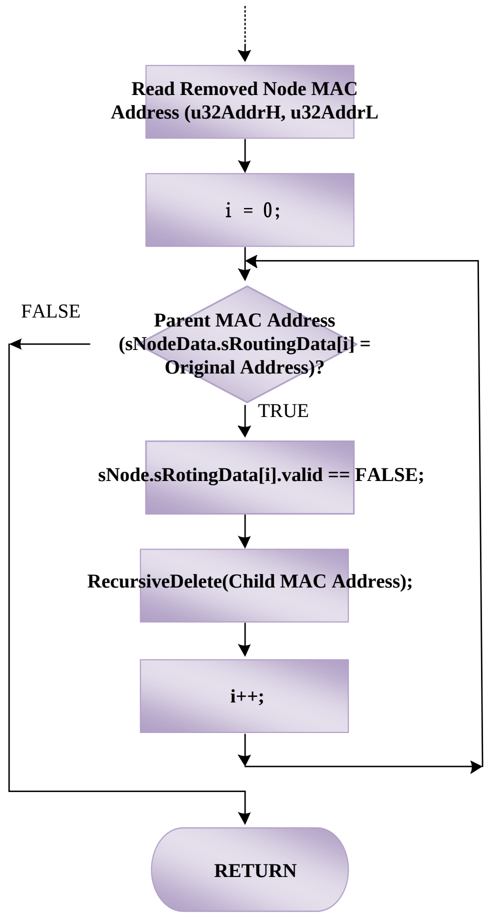

Figure 9. Recursive Removal of an Node and the Subtree of the Node

Table 1. Services of Bundles

Bundle	service
ZKernel	➢ manage/control home appliances ➢ event warning ➢ log file
ZPanel	➢ managerial panel exportation
Updater	➢ bundle update

3.2.2. ZPanel Bundle: ZPanel bundle provides a graphical user interface for the system management and control. The GUI is shown on a web browser in the animated form provided by Flash. The ZPanel bundle links to the database on the server and it will send a Flash animation file to the client browser that connects to the server. Then, the Flash animation program will request the status of nodes, sensor readings or appliances and refresh the web page accordingly. Hence, a user can monitor and control their(his/her) home network and appliances by examining the log file, sensor readings or modifying the control conditions for an appliance. The modification of the user will be written to the database and used by the ZPanel bundle for the control of the targeted appliance. When the user request the log file, the ZPanel bundle will download and show the records in the log file on the screen.

3.2.3. Updater bundle: Updater bundle is in charge of system maintenance and update. When it connects to the service provider, it can download the newest ZKernel and Zpanel bundle, even the newest Updater bundle. After the download, the old version bundle will be stopped and unistalled; then, the new version will be installed and started.

4. System implementation

In the implementation of the proposed system, a network node is realized by a FT-6200 development board [16] as shown in Figure 10. The onboard wireless microcontroller chip is JN5121, a product of Jennic Limited JN5121 [6]. The development board consists of a 32-bit RISC microprocessor, an IEEE802.15.4 compliant wireless transceiver [4], a 64Kb ROM, a 96KB RA, 4 input analog/digital converters, 2 output digital/analog converters, and 4 digital I/O ports. It conforms to AES [3].

Figure 10. FT-6200 Development Board for ZigBee

First, we connect to each FT-6200 development board via an RS-232 interface and burn the control codes onto the board. After that, the board is tested and an example snapshot from the super terminal is shown in Figure 11, in which a NodeAssociate package sent by a node requesting for a connection is observed. Each node will report periodically its status (connection signal quality, the numbers of MAC package sent/received, the number of line interruption, etc.) The status MAC package contains a ASCII column symbol "：" filed, a 16 byte source MAC address field, a 16 byte destination MAC address filed, a data field followed by a 2 byte LRC error-detecting code field, and ends with "\r\n" symbols. The data field includes a header delimited by "< >" so that the system can identify the category and purpose of the data.

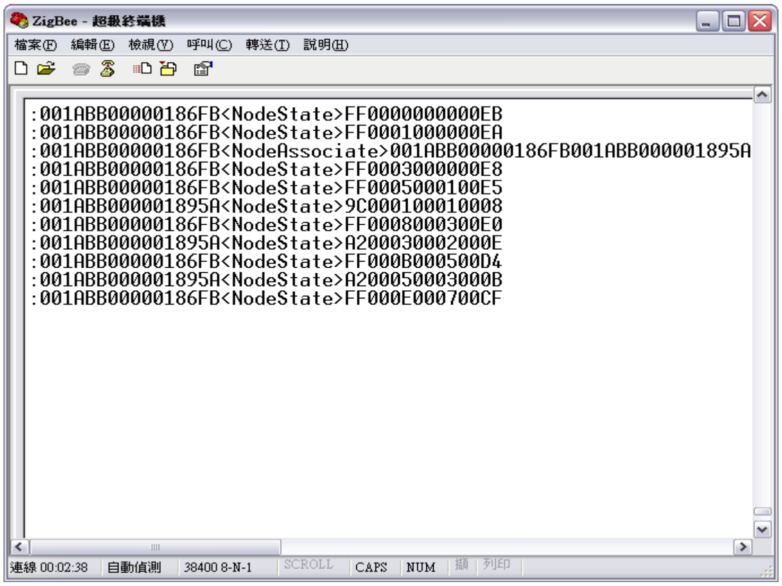

Figure 11. Massage Content Reported to the Server by a Coordinator

Figure 12 shows our implementation of an end device node which connects to a fan control circuit via its DIO port. The end device can operate the appliance (in this case an fan) by sending a signal to the control circuit.

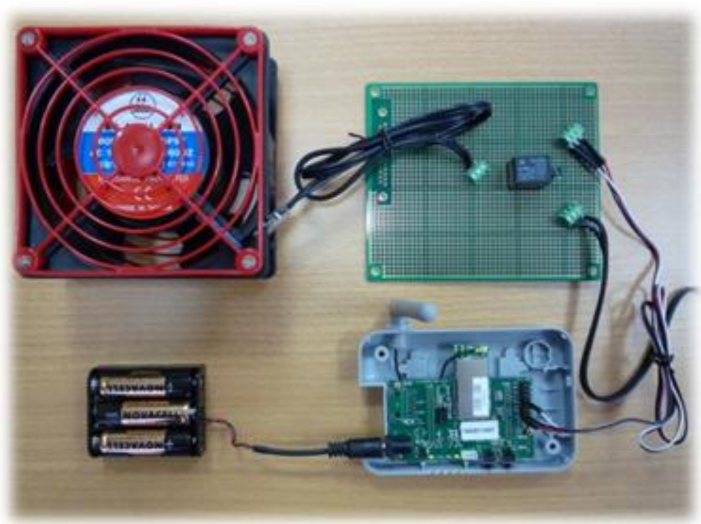

Figure 12. An End Device Node which Connects to an Appliance

The server can connect to the Internet and the ZigBee home network. An OSGi platform is also installed in the server so that a user can connect to the server. The OSGi platform adopted is the Knopflerfish implementation on which we install our home developed bundles (indicated by a broken line rectangle) as shown in Figure 13. The database needed by the ZKernel bundle and the ZPanel bundle is created and managed by MySQL DBMS [9]. The two bundles use JDBC (Java Database Connectivity) to link the MySQL database. Thus, the

"JDBC driver for MySQL" bundle is the first bundle in the broken rectangle. Apart from that, the ZKernel bundle has to connect to a coordinator via an RS232 interface; therefore the second bundle needed is Java Communications API. The rest three bundles are introduced in section 3.2; they are ZKernel, ZPanel and Updater bundle.

When a user intends to supervise the system, (he/she) they can connect to the server via a browser. The ZPanel bundle on the server is continuously listening to port 80 and whenever a client request arrives at port 80, it responds to the request with a Flash based animated home page which includes an action script code. The execution of the action script code will ask the ZPanel bundle to show the status of nodes. Thus, the ZPanel bundle will query the MySQL database and return the query result to the Flash animated home page for display. The graphic interface is illustrated in Figure 14. Through the managerial interface, we can monitor the status of every node and every appliance and we can also insert or modify conditions for the operation of appliances. When a user change a condition setting, the ZPanel bundle will update the new setting to the MySQL database and the ZKernel bundle will control the related appliance accordingly.

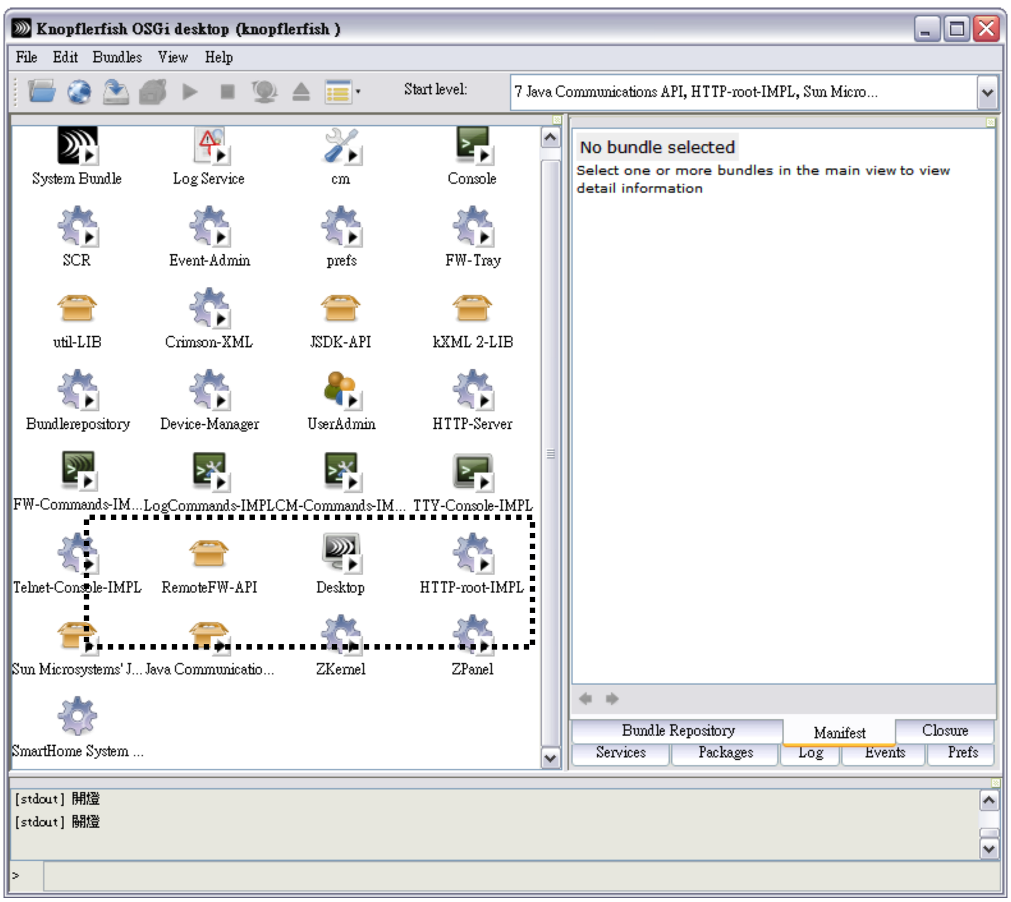

Figure 13. Bundles Installed on Knopflerfish

Figure 14. A Graphic Managerial Interface

5. Conclusion

In this paper, a smart home control system which consists of an OSGi (Open Service Gateway Initiative) service platform with ZigBee wireless networking replacing the traditional networking is demonstrated. The main contributions of the paper are as follows (1) the use of ZigBee wireless networking allows more flexible placement of a sensor or an appliance as their deployment is no more limited by the configuration of the traditional wires. (2) By adopting the ZigBee network technology, the cost of networking and deployment of the system is lowered and hence appealing. (3) Combined with a control interface which permits remote monitoring through the Internet, users can inspect the status of their home appliances via the interface. A registered security service provider can also connect to the OSGi server and monitor certain event warnings. When they occur, the company can send their security personnel to handle the problem. (4) As the OSGi supports remote bundle installation, the service provider can maintain the bundles on the platform remotely so that the system is updated conveniently and system stability many be better achieved. Thus, by the system, a digital home is, indeed, not only smart but also friendly to the earth.

Acknowledgements

This work is partially supported by National Science Council, Taiwan, Republic of China, under grant NSC 98-2221-E-143-005 and 99-2815-C-143 -010-E.

References

[1] A. Sikora, “ZigBee Competitive Technology Analysis Section Summary”, www.zigbee.org, **(2006)** November.
[2] C. X. Chang, "A Study of the Competitive Environment of Taiwan's Digital Home Industry", Master Thesis, Executive Master of Business Administration, National Chiao Tung University, Taiwan, **(2006)**.
[3] H. Li, Z. Jia and X. Xue, "Application and Analysis of ZigBee Security Services Specification", Proceedings of Second International Conference on Networks Security Wireless Communications and Trusted Computing (NSWCTC), vol. 2, **(2010)** April, pp. 494-497.

[4] IEEE Std 802.15.4-2006, "Part 15.4: Wireless Medium Access Control (MAC) and PhysicalLayer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)", **(2006)** September.
[5] I. A. Zualkernan, A. R. Al-Ali, M. A. Jabbar, I. Zabalawi and A. Wasfy, "InfoPods: Zigbee-Based Remote Information Monitoring Devices for Smart-Homes", IEEE Transactions on Consumer Electronics, vol. 55, no. 3, **(2009)** August, pp. 1221-1226.
[6] JN5121 Wireless Microcontroller, http://www.jennic.com/products/wireless_microcontrollers/jn5121.
[7] J. E. Lim, O. H. Choi and D. K. Baik, "An evaluation method for dynamic combination among OSGi bundles based on service gateway capability", IEEE Transactions on Consumer Electronics, vol. 54, no. 4, **(2008)** November, pp. 1698-1704.
[8] Knopflerfish OSGi - open source OSGi service platform, http://www.knopflerfish.org/.
[9] MySQL, http://dev.mysql.com/.
[10] N. Watthanawisuth, A. Tuantranont and T. Kerdcharoen, "Microclimate real-time monitoring based on ZigBee sensor network", IEEE Sensors, **(2009)** October, pp. 1814-1818.
[11] OSGi Alliance, http://www.osgi.org/.
[12] P. Kinney, "ZigBee Technology: Wireless Control that Simply Works", www.zigbee.org.
[13] S. Jung, A. Chang and M. Gerla, "Comparisons of ZigBee Personal Area Network (PAN) Interconnection Methods", Proceedings of 4th International Symposium on Wireless Communication Systems, **(2007)** October, pp. 337-341.
[14] ZigBee Alliance, "ZigBee Wireless SensorApplications for Health, Wellness and Fitness", www.zigbee.org, **(2009)** March.
[15] ZigBee Alliance, http://www.zigbee.org/.
[16] FT-6200 Zigbee Development Kit, http://www.sruewin.com.tw/.
[17] L. Oborkhale and A. Salatian, "Challenges of the Digital Home in a Developing Economy", International Journal of Smart Home, vol. 5, no. 1, **(2011)** January, pp. 31-36.
[18] D. Bregman, "Smart Home Intelligence – The eHome that Learns", International Journal of Smart Home, vol. 4, no. 4, **(2010)** October, pp. 35-46.
[19] D. Kubat, M. Drahansky and J. Konecny, "Wireless Control of Intelligent Houses", International Journal of Control and Automation, vol. 4, no. 3, **(2011)** September, pp. 123-126.

Author

Ming-Chiao Chen received the Ph.D. degree in Computer Science and Information Engineering from National Dong Hwa University, Hualien, Taiwan, in 2008. He joins the department of Computer Science and Information Engineering, National Taitung University, Taitung, Taiwan, since 2008 and is now an assistant professor. His current research interests are in the area of home networks, wireless networks, and sensor networks.