UPnP-ZigBee 네트워크 브릿지 시스템

김성중*, 서해문*, 박우출*, 김민호*
*전자부품연구원
e-mail : sungjungk@keti.re.kr

Network Bridge System for Interoperation of ZigBee-UPnP Network

Seong-Joong Kim*, Hae-Moon Park*, Woo-Chool Park*, Min-Ho Kim*
*Korea Electronics Technology Institute

요 약

Universal Plug and Play (UPnP) is one of the most promising home network middleware technologies because of its straightforward implementation and zero configuration characteristics. However, it has a limitation of the operations only IP based network and proceeding in a single IP subnet. In this paper, we proposed network bridge architecture for UPnP network to interoperate heterogeneous network, UPnP and ZigBee. The proposed network bridge architecture is capable of configuration ZigBee device as a virtual UPnP device. This technique is promising for seamless inter-networking with ZigBee and UPnP network. Also, the architecture of the proposed network bridge architecture can be applied for a future smart home.

1. Introduction

Nowadays, the home network has widely spread, but most of it has separated into entertainment, home automation and PC-to-PC connections. Among them, the technology of the home networking can be divided a wired system and a wireless system. The wired system are consists of Phone Line, Power Line, IEEE1394, USB (Universal Serial Bus) and Ethernet, while the wireless system are WLAN, HomeRF, Bluetooth, UWB (Ultra Wide Band), Zigbee and the technology of Hiper-LAN [1].

Home network will progress into the form of the integrated network from various kinds of technologies. These kinds of the future home network will increase the use of simple data networks and the technologies of wireless and will place a great deal of weight on the device of home entertainment which has the forms of HAVi [2]. They will share contents between the broadcasting system and the Internet and will increase interoperability through the device for bridging.

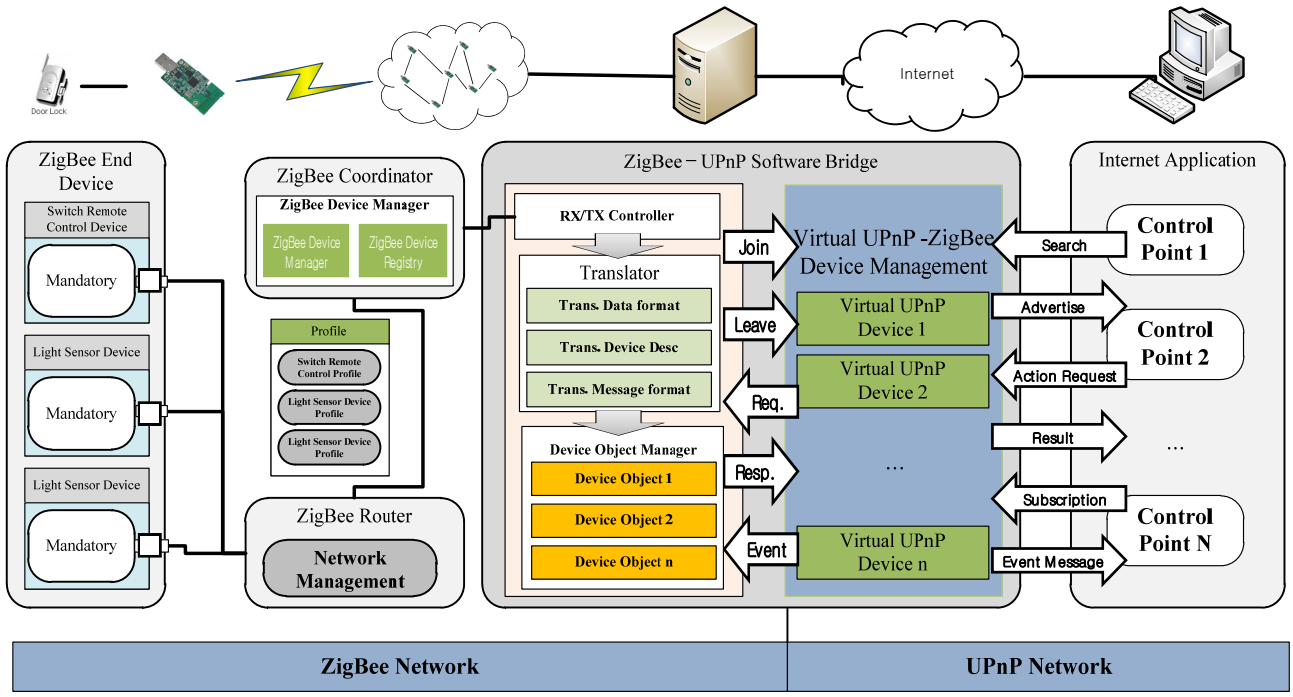
The center of the composition of the most home network is system software, middleware. This middleware play roles in switching into information and streaming multimedia contents and in controlling consumer electronics connected in the network. Especially, we come across with a lot of consumer electronics from many manufacturing companies in home appliances. Therefore, we need to develop home network middleware applications to give the interoperability among various systems.

UPnP is a home network middleware which provides us with the service for using the existing IP and HTTP and finding the device on the home network [3]. When you use the UPnP, fixed-lines and wireless based on IP network has to be installed at all times although it can search the device automatically. So, it is difficult to apply the existing home network, because most of the home appliances are based on non-IP network. There is a method to solve this problem using Ethernet or Wi-Fi technology [4]. The method that inserts Ethernet card into the home appliances is too difficult to apply because it is inefficient. And the method that using Wi-Fi technology has characteristics of high power consumption, so it requires high costs for applying to the home network. Zigbee protocol that aims at supports for low-cost, low-data rate and low-power consumption is suitable in this system. But Zigbee protocol based IEEE 802.15.4 has a problem interoperation with UPnP network because it is not IP based network.

To solve this problem, there are two different types of solutions. One way is to insert adaptation layer into Zigbee stack. This way is appropriate to design to translate non-IP based network into IP based network, but it has a problem to complicated design for a protocol. Another way is to design network bridge between Zigbee and UPnP network. Second method is the best way to solve the problem. To implement this method, we designed the technique that Zigbee device recognize as a virtual UPnP device. Through this way, we don't need to change the existing Zigbee protocol. Also, it is possible to recognize zigbee device as a virtual UPnP device easily. In this paper, we propose network bridge system for interoperation with Zigbee and UPnP network.

2. Proposed UPnP-ZigBee Network Bridge System

The proposed network bridge system architecture, which is depicted in Fig. 1, is comprised of Zigbee network part and UPnP network part. In this section, we first explain network bridge design consideration with interoperation of Zigbee and UPnP network. Then, we represent network bridge system operation scheme based virtual UPnP device mapping technique. Lastly, we describe network data flow that is characteristics of seamless.



(Figure. 1) Overall System Architecture

A. Design Consideration

Zigbee is designed for the network with 250 kbps and relatively low computing power. On the other hand, UPnP aims at supports for at least 10 Mbps and ability to processing XML based messages. Because of these gaps, we have to consider to Zigbee for compatibility with UPnP.

Message format– Zigbee message consisted by the 127 bytes, which mainly includes the following sections, MAC header, NWK header and APS header. MAC header is currently being transmitted the message header contains the source address and destination address. NWK header contains the message which is composed the actual source address and final destination address. APS header contains the configuration ID, cluster ID and the current purpose of the message. On the other hands, UPnP message format is based on XML which consist Simple Service Discovery Protocol (SSDP), General Event Notification Architecture (GENA) and Simple Object Access Protocol (SOAP).

Device Description- UPnP and Zigbee differ from one another in application model. Zigbee protocol uses profile based device description which is simply described logical components and their interfaces. And device discovery is performed by using cluster lists. While, UPnP uses descriptions based on XML and discovers using Simple Service Discovery Protocol (SSDP). Hence, it is necessary to translate device descriptions of each standard.

B. Network Bridge Architecture

The system architecture, which is depicted Fig. 1, is comprised of Zigbee network, UPnP network and network bridge for interoperating. The proposed network bridge architecture has four modules; RX/TX Controller, Translator module, Device Object Manager and Virtual UPnP-Zigbee Device Management.

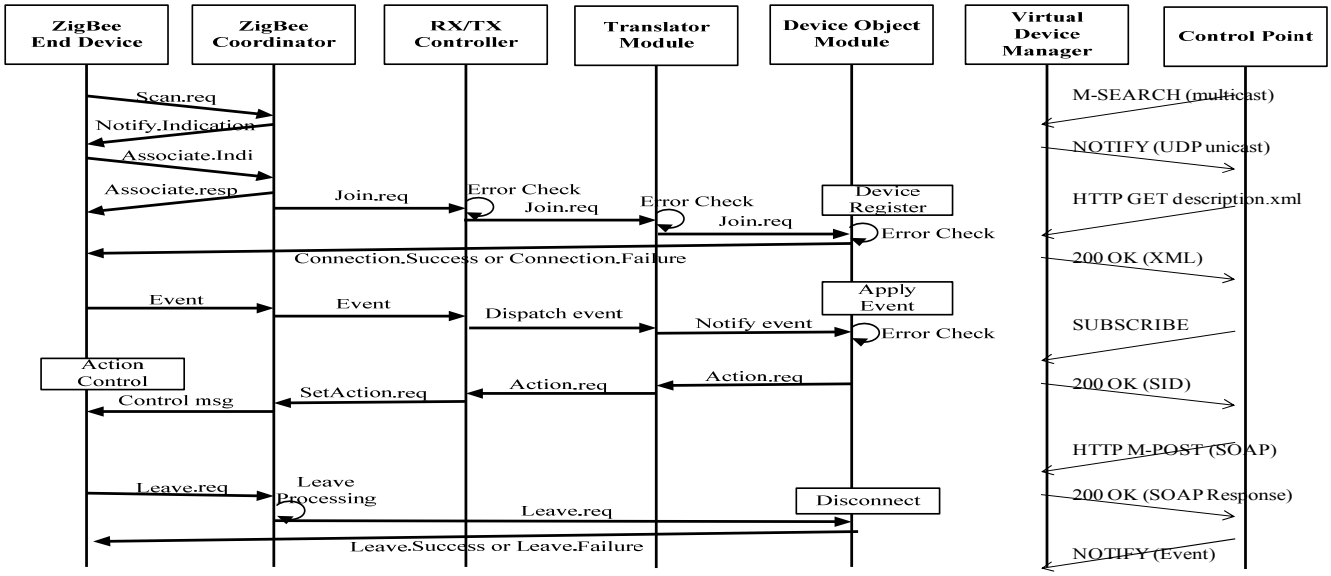
RX/TX Controller– It is mainly responsible for controlling data transmission and receiving the data from Zigbee End Device (ZED) to translator module. The responsibility of it also includes filtering fault messages received from Zigbee devices.

Translator Module– We represent different thing about Zigbee and UPnP's transmission data format and device descriptions, previously described in subsection II.A. Data format in UPnP network is based on Text/XML. On the other hand, Zigbee is composed of binary format. This module plays a role to convert heterogeneous data format. Also, in case of registering device for the first time, this module creates device description and service description files using the information of received from Zigbee End Device (ZED). We create these files in accordance with Zigbee profile. Device and service description fields described in Table 1.

Device Object Module– In this paper, we represent technique that recognizes ZED as a device object form, one to one correspondence. Managing ZED as a form of device object is useful in administrating device: creation, termination, eventing and controlling. If the device is registered for the first time, it creates the object which includes all information about each ZED reference to the files already described service and device information. And if certain device demands for leave this network, this module delete applicable object which includes device information and files.

<Table 1> DATA AND SERVICE DESCRIPTION

Device description file	Service description file	
Device type	Action	Name
Friendly Name		Related State Variable
Manufacturer		Direction
Manufacturer URL	Service State Table (Boolean type)	Name
Model Description		Type
Model Name		Default value
Model Number	Service State Table (value type)	Name
Model URL		Type
Serial Number		Allowed value Range
-		Step



(Figure. 2) Network daate flow

Virtual UPnP-ZigBee Device Management - It creates a virtual UPnP device using each device object, one to one correspondence using libupnp software library, provided by Intel [6]. Virtual UPnP device has a ZED's device and service description and acts as if real ZED. UPnP control point control the virtual UPnP devices, because of it recognizes virtual UPnP devices as real device.

C. Network Data Flow

Network flow is classified into four kinds; join, leave, event and control, which is depicted in Fig. 2. We don't need to consider that ZED connects into the process of Zigbee network, because this connection process is the same as the conventional Zigbee network protocol. Also, we have no regard for the connection process of UPnP network, because we can use the existed UPnP protocol which performs high quality. We just focus on the network bridge through the way for interoperability between Zigbee and UPnP network.

Join is the process that ZED tries to connect to network bridge. At first, a ZED wants to request a coordinator to connect in this network, then it sends JOIN REQ message to the Zigbee coordinator which manage the Zigbee network. And then the Zigbee coordinator creates ZED table that has each ZED's real-time information in object form and transmits the message of CONNECTION SUCESS to corresponding with ZED when the ZED tables are available. If it is not, the coordinator sends the message of CONNECTION FAILURE. Then ZED should proceed with this process, until connection success. The coordinator which has the ZEDs table sends the message of device information to the network bridge at regular intervals. Using this message, proposed network bridge manages registered ZED. If it didn't receive this message for a while, it disconnects corresponding ZED in this network. With the updated information, UPnP Control Point will discover devices automatically.

When the already registered ZED in the proposed network bridge wants go out in this network, leave process will occur. Zigbee sends LEAVE REQ message to the coordinator that has managed the Zigbee network. The coordinator request the network bridge to remove ZED, in object form, from the ZED table and it sends LEAVE SUCCESS message to the corresponding ZED. Using the information of device table which is given from the coordinator at regular intervals, inside of the network bridge lets the registered device in device object manager (DOM) remove from the coordinator. If the device in DOM is canceled, it is also removed automatically in UPnP Control Point by an UPnP protocol.

Event is the process when extraordinary circumstances change for example, emergency, too high temperature and unlocked valve in ZED. If there are unusual events in ZED, it sends this event to a coordinator. And it convey this event data to network bridge. The network bridge which is received this message from the ZED informs users of the relevant information as the form of warning.

Control is the process to conduct the order from UPnP Control Point. For example, when there is a ZED in an air conditioner, we can control the temperature and turn on or turn off the power. This kind of information transmits users from control point to network bridge. The network bridge which is given the message creates the form of binary data which can be sent to ZED and sends the data to Zigbee coordinator. After the coordinator transmits the data to the applicable devices, these devices performs the order and sends the confirmation message to the coordinator. When this message transmits it to network bridge, the data in device object manager is updated, and then we can check the updated information in an UPnP Control Point automatically.

3. Implementation

This section describes implementation network bridge system and experiment of the performance of the network bridge. In order to demonstrate that the proposed network bridge works successfully, we have built up a test bed which

consists of a Zigbee network, the network bridge system and control point. The Zigbee network is composed of end devices, routers and a coordinator which is provided by Ember that integrates an ARM Cortex-M3 processor, IEEE 802.15.4 RF transceiver, 192 KB Flash and 12 KB RAM and EmberZNet PRO network protocol stack supporting the ZigBee PRO feature set in Fig. 3(a). The proposed network bridge is implemented through a libupnp library supplied by Intel, and uses an embedded Linux system that has an s3c6410 core in Fig. 3. (b) [7]. And we show how our network bridge operates Zigbee devices and is notified from UPnP control point.

When Zigbee devices were connected to the network bridge system through the Zigbee coordinator, these devices were represented as virtual Zigbee devices. At that time, the proposed network bridge creates virtual Zigbee devices service and device description files. Then, UPnP control point requests UPnP descriptions to the virtual Zigbee devices in order to know their detailed information, which is shown in Fig. 4. The UPnP control point then could control the Zigbee devices and got the status feedback as normal UPnP devices.

4. Conclusion

The UPnP which is characteristics of straightforward implementation and zero configurations is useful in home network. But, the UPnP is difficult to apply consumer electronics because it operates only IP based network. So, we designed network bridge for interoperation between Zigbee and UPnP home network middleware. In this paper, we describe Zigbee and UPnP network bridge for overcome ing the UPnP's considerably restricted drawback. Using Zigbee protocol is suitable method for bridging UPnP because it is wide spread technique for the characteristics of low cost and low power in wireless sensor network. And we implemented seamless network bridge system for interoperation with UPnP-Zigbee network. This architecture will be useful in the wireless sensor network.

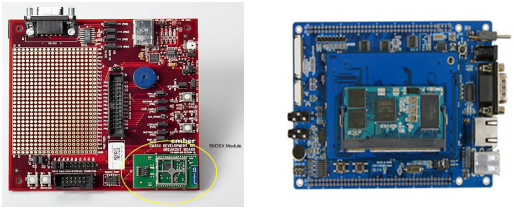


Fig. 3. (a) Zigbee Experiment board (b) network bridge board

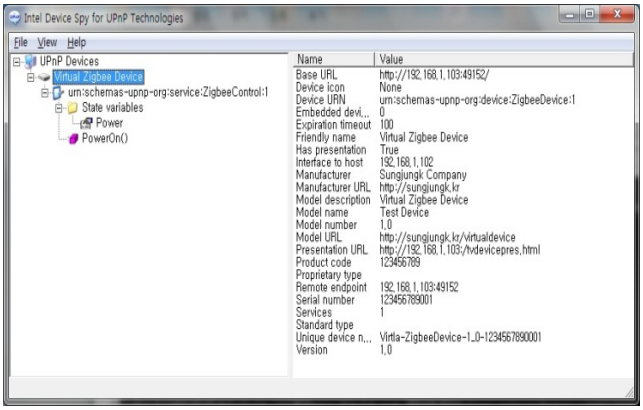


Fig. 4. Virtual device list

Reference

[1] Yamauchi, N.; Ohta, T.; Takami, K., "Home Appliance Virtualization Software Configuration for Home Network Services", Software Engineering Advances (ICSEA), 2010 Fifth International Conference on, pp. 26-31

[2] Kuo-Wei Hsu; Chuen-Liang Chen; Ting-Ying Yu; Wu-Cheng Li, "Design and implementation of a HAVi stack with Linux and Java", Signal Processing and Information Technology, 2003. ISSPIT 2003. Proceedings of the 3rd IEEE International Symposium on, pp. 66-69

[3] Miller, B.A.; Nixon, T.; Tai, C.; Wood, M.D.," Home networking with Universal Plug and Play", Communications Magazine, IEEE, 2001, vol. 39, pp. 104-109

[4] Afanasyev, M.; Tsuwei Chen; Voelker, G.M.; Snoeren, A.C.,"Usage Patterns in an Urban WiFi Network" Networking, IEEE/ACM Transactions on, vol. 18, pp. 1359 – 1372

[5] Tal Zhang, Madhani.S, Gurung. P, Van den Berg. E, "Reducing Energy Consumption on Mobile Devices with WiFi interfaces," Proceeding Globecom 05. IEEE, 28 Nov.-2 Dec. 2005

[6] http://upnp.sourceforge.net

[7] http://www.samsung.com/us/business/semiconductor/news/downloads/s3c6410Datasheet_0408