

---

# 홈 네트워크에서의 맥내외 기기들의 동적 바인딩 시스템 설계 및 구현

## Design and Implementation of Dynamic Binding System among the Devices Located in Inside and Outside of Home Networks

---

이재현, 권경희  
단국대학교 전자계산학과

Jae-Hyun Lee(wogusking@dankook.ac.kr), Kyung-Hee Kwon(khwon@dankook.ac.kr)

---

### 요약

홈 네트워크는 주택건축 시점에 건축물의 내장시스템을 중심으로 구축되기 때문에 건축 후 새롭게 구입한 기기와의 네트워크 호환성을 보장하지 못한다. 본 논문에서는 맥내외 기기를 통제하기 위해 맥내·외 기기와 효과적으로 연결할 수 있는 동적인 바인딩 시스템(Dynamic Binding System:DBS)을 제안한다. DBS는 WPAN(Wireless Personal Area Network)에서 동작하는 ZigBee 바인딩 기술을 기반으로 맥외 모바일 기기와 맥내 기기와의 연결을 시도하였다. 본 연구를 위한 시스템 구성으로는, 홈 서버에 프록시 서버와 location 서버가 위치하고 맥내의 기기들은 tinyOS가 탑재된 telos보드로 구성하였으며, 모바일 기기는 에뮬레이터로 구현하였다. 맥내의 통신은 ZigBee에서 사용하는 IEEE 802.15.4 (2.4GHz)프로토콜을 이용하고, 그 외는 TCP/IP를 이용했다. DBS는 맥내 기기들이 주소형식에 상관없이 맥내·외 기기와 연결되는 것을 용이하게 하였다. 또한 홈 네트워크로 맥내 기기가 새로이 진입될 경우, 기기간 메시지교환을 통해 자동으로 인식될 수 있도록 하였다.

■ 중심어 : 홈 네트워크 | 지그비 | 바인딩

### Abstract

Since home network system is usually set up based on the built-in system inside home at the time of construction, it has a difficulty in being compatible with new devices after construction. In this paper, we propose DBS(Dynamic Binding System) that can make links effectively among the devices located in inside and outside of home networks. Implementation of the DBS is based on the ZigBee's binding in WPAN(Wireless Personal Area Network). For this research, we constructed the system consisting of home server that contains proxy server and location server, devices inside home in which tinyOS is embedded, and mobile device that is an emulator. The communication inside home is done by IEEE 802.15.4 protocol and the communication between servers and mobile devices outside home is done by TCP/IP protocol. Regardless of device's address format, the DBS facilitates a link among the devices located inside and outside of home networks. And, introducing a new device into home network can be recognized easily by exchanging messages between devices.

■ keyword : Home Networks | ZigBee | Binding

---

\* 본 연구는 2009년도 단국대학교 대학 연구비의 지원으로 연구되었습니다.

접수번호 : #081020-003  
접수일자 : 2008년 10월 20일

심사완료일 : 2008년 12월 08일  
교신저자 : 이재현, e-mail : wogusking@dankook.ac.kr

## 1. 서론

홈 네트워크에서의 네트워킹 방식은 응용 분야의 특성에 따라 서로 다른 방식이 도입될 것으로 기대된다 [1]. 그 예로 Echelon사의 LonWork 솔루션은 전력선을 기반으로 하는 통신방식으로 building automation에서부터 home automation으로 그 적용분야를 확장하고 있다[2]. WLAN은 이미 그 성능이 검증되어 산업계에 널리 보급되어 있으며, 홈 네트워크에서도 주요 네트워킹 방식으로 자리를 차지할 것으로 많은 전문가들은 전망하고 있다. ZigBee Alliance에서 기술표준화를 추진하는 ZigBee는 IEEE 802.15.4 네트워크를 이용하여 통신하며, ZigBee Protocol Stack을 정의함으로써 home automation 분야에 적용 가능한 기술로써 인정받고 있다[3].

위와 같이 홈 네트워크 시스템을 구축하는 기술은 다양하게 존재하지만, 각 기술간 호환성에 대한 문제는 홈 네트워크를 발전시키기 위해 반드시 해결해야 하는 과제이다. 이런 호환성 문제로 인해 홈 네트워크 시공업체들은 건축되는 시점에 선정된 기술방식만으로 주택에 내장되는 기기들이 홈 네트워크를 구성하게 된다. 따라서 주택의 완공 후에 추가로 구입되는 기기들은 내장된 기기와 동일한 홈 네트워크로 구성되기가 쉽지 않다. 본 논문에서는 상이한 네트워크 통신방식에 대한 해결방안 및 맥내·외 망을 연결하는데 현실적인 문제점을 확인하고 그 해결책을 제시하고자 한다. 첫째, 맥내의 기기들에 대한 주소 방식이 상이할 수 있다. 생산업체마다 자신들의 통신방식에 맞는 주소체계를 선택하여 기기에 사용함으로써 다른 프로토콜을 사용하는 업체의 기기와 호환될 수 없다. 이를 해결하기 위해 각 주소체계마다 자신들의 주소 변환기술들이 등장하게 되지만 본 논문에서 제안하는 방식은 모든 변환기술을 동적바인딩 테이블을 통하여 해결할 수 있다. 둘째, 건축 당시 내장된 기기와의 호환성을 맞추어 구입한 기기들이 향후 이사를 갈 때 호환성을 보장 받을 수 없다. 이와 같은 문제를 해결하기 위해 DBS(Dynamic Binding System)은 맥내의 기기들과 홈 서버 사이의 정해진 바인딩 절차를 통하여 각 기기의 주소체계와 상

관없이 홈 서버를 통한 연결을 설정할 수 있도록 하였고, 맥외 기기를 통해 맥내 기기를 통제할 수 있도록 하였다. 또한 이 방식은 다양한 통신방식을 동시에 지원할 수 있는 홈 서버에게는 더욱 효율적이다. 본 시스템에서 제안하는 바인딩은 기기들 간에 통제를 보다 원활하게 하기 위한 수단으로 ZigBee Alliance가 WPAN(Wireless Personal Area Network) 환경에서 제안한 바인딩방식을 기본으로 하며 그 방식과 형식을 수정 확장하였다[4]. ZigBee의 바인딩 기술 자체를 연구하는 논문은 있으나, 이를 변형하여 홈 네트워크의 주소체계에 유연성과 확장성을 부여하는 기술은 본 논문이 처음으로 제안하는 것이다.

본 시스템은 다음과 같은 구성으로 구현하였다. TinyOS를 탑재한 가정 내의 기기는 802.15.4 (2.4GHz) 네트워크상에서 근거리 무선통신을 하고, 홈 서버는 맥외 망의 통제 기기와 맥내 망의 기기를 연결해주는 proxy 서버와, 바인딩정보를 유지하기 위한 location 서버로 설계 및 구현 하였다. 맥외 망의 모바일 기기는 emulator로 구현하여 바인딩 및 간단한 on/off 명령을 수행할 수 있도록 했다.

본 시스템을 설계하는데 기본이 되는 ZigBee의 바인딩기술을 이해하여 본 시스템에 대한 이해를 돕고자 한다. ZigBee의 바인딩기술은 ZigBee Spec.에서 light profile을 통하여 설명하고 있다. 본 논문에서 언급하는 profile은 향후 light profile과 같이 ZigBee Alliance에서 제공될 것이고 또한 범위 역시 다양하게 마련될 것으로 전망한다[3]. 또한 무선 네트워크 routing에 대한 관련 연구를 살펴보고 홈 네트워크에 효과적인 방식을 선택하여 구현하고자 한다.

### 1. ZigBee 바인딩

바인딩은 두 개의 장치 사이에 존재하는 end-point들 간의 연결을 table을 이용하여 논리적으로 연결해 주는 개념이다. 각각의 end-point의 연결이 1:n 혹은 1:m일 수 있으므로 이를 해결하기 위해 클러스터 id를 사용한다[4].

[그림 1]은 바인딩 테이블의 예를 보여주며, 그림에서 바인딩 테이블은 ZigBee의 네트워크 구성 중

coordinator에 위치한다. 바인딩 테이블은 또한 indirect addressing 방식을 가능하게 하는데 indirect addressing 방식이란 송신 기기가 source의 주소를 coordinator에게 전송하면 coordinator가 바인딩 테이블을 검색하여 해당 destination으로 해당 event를 전달하는 방식이다. 이 방식을 통하여 전송데이터 중 도착지 주소로 발생하는 오버헤드(10 octets)를 줄일 수 있다.

바인딩 테이블이 [그림 1]와 같이 존재한다면 Device1의 EP1는 Device2의 EP1, EP2를 통제할 수 있는 것이다.

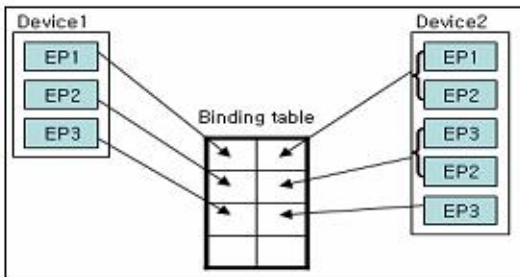


그림 1. End-point를 연결하는 Binding table

Device1와 Device2가 1:1이 아닌 end point라는 서브 주소와 바인딩 테이블을 두어 통제 방식을 그룹화 할 수 있다.

ZigBee 내부에서의 바인딩 과정을 살펴보면, 바인딩 요청은 Bind\_req 클라이언트 서비스를 이용하여 수행되고, Unbind\_req는 바인딩 테이블에서 바인딩 정보를 삭제하는 작업을 수행한다. ZigBee coordinator 내부적으로 바인딩 테이블 정보생성은 APS(Application Support Sublayer)가 APSME-BIND.request를 요청 받으면 APSME (APS Management Entry)가 바인딩 정보 생성작업을 수행한다. 삭제요청은 APSME-UNBIND.request에 의해 수행된다[1][4].

## 2. Ad-hoc Routing

무선 네트워크에서의 많이 알려진 ad-hoc routing 방식은 다음과 같다.

DSR(Dynamic Source Routing): 경로발견을 위하여 송신노드는 route discovery 프로세스를 수행한다. 중

간노드들은 자신의 캐쉬를 확인하여 송신노드가 원하는 목적지노드를 확인하여 응답할지 수신한 패킷을 broadcasting할지 결정하며, 이러한 절차를 반복하여 도착지와의 경로가 설정된다[5].

AODV(Ad-hoc On-demand Distance Vector Routing): 메시지를 broadcasting하면서 역 경로(reverse path)를 생성한다. 목적지에 도착하면 역 경로를 따라 송신자까지 경로가 설정된다. Sequence number를 이용하여 최신의 정보만을 유지한다[6].

CBRP(Cluster Based Routing Protocol): 클러스터를 구성하여 모든 전송 패킷은 클러스터 헤더에게 전송하고, 각 헤더는 DSR을 이용하여 패킷을 전송하는 방식으로 전체적인 트래픽을 줄였다[7].

DSDV(Destination Sequenced Distance Vector): Table driven 방식으로 각 호스트들은 routing table을 유지하면서 패킷을 전달한다. Sequence number를 이용하여 최신의 정보를 유지한다[8].

본 시스템에서 사용하는 네트워크는 다음의 기능을 가지고 있어야 한다. 홈 서버를 중심(Root)로 하는 트리 모양의 토폴로지, ad hoc 네트워크 그리고 멀티홉 라우팅의 구현이다. 이들을 만족시키기 위해 모든 노드는 상호간 Beacon 메시지 전송을 통해 자신의 부모노드를 항상 최신으로 유지하고 있어야 한다. 이때 최상위 루트는 홈 서버가 될 것이고 부모의 선정은 MAC(CC2420)에서 전달하는 Link Quality Indication 값으로 계산한다.

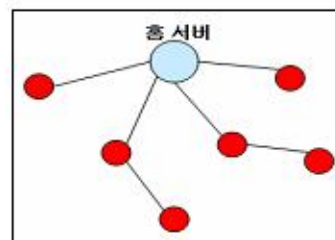


그림 2. 맥내 망 토폴로지

반대로 맥외 망의 기기로부터 전달되는 메시지가 맥내의 기기로 전송될 때는 broadcast 방식으로 하향 전송 하되 각 수신 노드는 이웃노드로부터 전송되는 패킷에

대해 sequence number를 통하여 중복되는 패킷을 소멸시킨다. [그림 2]는 본 논문에서 사용한 맥내의 기기들과 홈 서버의 토폴로지를 나타낸다.

### 3. nesC

NesC는 TinyOS의 실행모형을 구조화시킨 언어이다. 구조적 특징으로 프로그램이 컴포넌트로 구성되어 있으며 각 컴포넌트는 implementation (module, configuration) 과 interface (provides interface, uses interface) 부분으로 크게 나눌 수 있다.

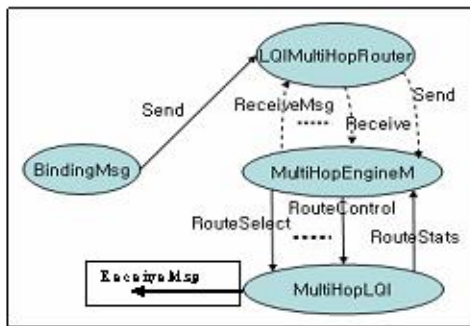


그림 3. 주요 컴포넌트 wiring

또한 application의 구조를 파악하기 용이하게 각 컴포넌트 간의 관계를 wiring을 이용하여 나타낼 수 있다. 따라서 구현 및 분석 시 컴포넌트 사용방식(uses / provides)를 쉽게 이해할 수 있다[9]. NesC를 이용한 구현부분에서 UART 통신, Radio 통신, Timer 및 LED 관련 컴포넌트는 TinyOS에서 지원하는 컴포넌트를 이용했으며, 본 시스템에 필요한 라우팅 및 응용 계층은 3개의 주요 컴포넌트로 다음과 같이 구현하였다. MultiHopLQI module은 이웃노드와 연결유지를 위해 주기적인 간격으로 메시지를 전송한다. BindingMsg module은 홈 서버로 바인딩 메시지를 전송한다. 그리고 MultiHopEngineM module은 서버로부터 명령을 처리한다. [그림 3]은 위의 3개의 module에 대한 wiring을 보이고 있다. BindingMsg에서 주기적인 메시지 전송을 하기 위해 Timer를 가동하고 그 때 호출되는 Send interface는 MultiHopEngineM module에서 제공하게 된다. MultiHopLQI의 ReceiveMsg interface는 beacon

메시지를 수신 하여 자신의 이웃 노드들과 연결을 유지하고 갱신하는 역할을 수행한다. 명령 메시지는 MultiHopEngineM의 ReceiveMsg interface을 통해서 ReceiveMsg.receive 함수로 수신된다. 수신 메시지를 분석 후 상위(홈 서버)로 전송할 것인지 하위(target 노드)로 전달할 것인지 판단한다.

## III. 시스템 설계

본 장에서는 전체적인 시스템을 설계하기 위해 각 서버들의 특징과 기능 그리고 서버 및 기기들 간의 프로토콜을 분석 한다. 또한 proxy 서버와 기기, proxy 서버와 location 서버 그리고 proxy와 mobile 기기사이에서 사용되는 패킷을 설계한다.

### 1. 시스템 개요

Proxy 서버와 location 서버는 분산 환경에 적합하도록 TCP/IP 통신으로 연결한다. Mobile 기기는 emulator를 만들어 홈 서버 접속, 바인딩, 명령어 전송의 순으로 각 단계를 테스트 할 수 있도록 메뉴를 구성하였다. 맥내 망 기기로 구성되는 노드는 2.4GHz Radio 송수신기(Chipcon CC2420)를 가지고 있으며 TI사의 MSP430 프로세서가 탑재되어 있다. 노드 중에 proxy 서버에 연결되는 노드는 Base Station역할을 하여 Proxy 서버의 UART와 연결되어 있다. 따라서 다른 노드가 base station에게 메시지를 전송하면base station 노드는 그 메시지를 UART를 통해서 proxy 서버로 전송하게 된다. 맥외의 Mobile기기의 명령어를 담고 있는 메시지는 proxy 서버로 전송되고, proxy 서버는 base station 노드를 통하여 맥내의 각 기기들을 통제하게 된다.

### 2. 메시지 흐름

Proxy 서버는 target 노드(맥내기기)로부터 바인딩 요청을 받은 후 location 서버에게 target 노드의 정보를 저장한다. 이 후에 mobile 기기는 현재 바인딩된 target 노드 리스트로부터 원하는 target 노드를 선택하여 그

노드와의 바인딩 요청을 proxy 서버에게 전송한다. Proxy 서버는 이미 바인딩된 target 노드의 정보에 mobile기기의 바인딩 정보를 추가한다. [그림 4]에서 Target 노드 list요청과 BIND 요청이 이 단계에 수행되는 부분이다.

바인딩이 이루어진 후에는 맥 외의 mobile 기기로부터 target 노드에게 command를 전송할 수 있게 되며, mobile기기의 명령이 정확한 target 노드로 전송되기 위해서는 Location 서버로부터 mobile 기기와 바인딩된 target 노드의 주소정보를 얻어 와야 한다.

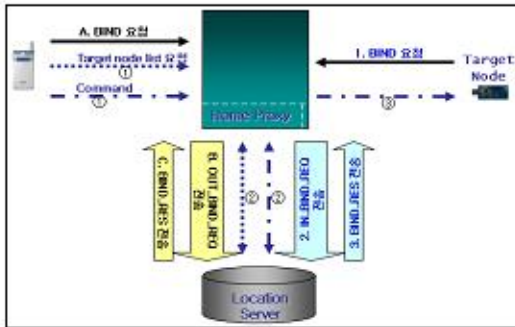


그림 4. DBS의 메시지 흐름

위와 같은 절차를 통해서 proxy 서버는 맥외의 mobile 기기와 맥내 가전기기의 주소형식을 고려하지 않고 상호간의 통신이 원활하게 이루어지도록 할 수 있다.

### 3. 패킷 설계

#### 3.1 Target 노드의 패킷구조

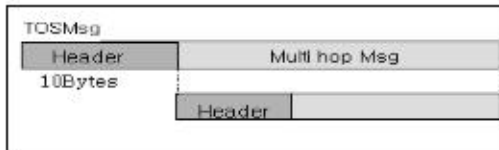


그림 5. 802.15.4 네트워크에서의 패킷 구조

#### TOMsg Header

Len	Addr	Type	Group	Data	.....
-----	------	------	-------	------	-------

TinyOS패킷으로 가장 바깥쪽 헤더이고, length는 data의 길이, addr는 패킷의 도착주소, type과 group은 서비스에 따라 지정할 수 있다[10][11].

#### Multihop Msg Header

Srcaddr 12bit	Destaddr 12bit	Seqno 12bit	Opseqno 12bit	Hopc 12bit	Destaddr 12bit
------------------	-------------------	----------------	------------------	---------------	-------------------

Multihop Msg는 도착지의 주소(Destaddr)를 통하여 해당 노드에게 전송되며 노드들의 중복 수신을 방지하기 위해 Seqno를 사용한다.

#### Binding Msg

Type(1bit)	Rsv(11)	Pid(2)	Cid(2)	EP(2)	Desc(2)
------------	---------	--------	--------	-------	---------

Binding Msg는 바인딩 요청/응답 그리고 통제 요청 시 명령 패킷으로 사용된다.

#### Beacon Msg

Parent(12bit)	Cost(2)	Hop(2)	TimeSlot(4)
---------------	---------	--------	-------------

Beacon Msg는 이웃 node들과의 연결을 생성/유지하기 위해 주기적으로 전송된다.

### 3.2 Proxy와 location 서버사이의 패킷 종류

표 1. 패킷 구조와 각 패킷의 설명

Byte Index	0	1~4	5~Len-2	Len-1
Field	Packet Spec	Len	Data	Sum
Packet Spec	설 명			
OUT_BIND_REQ	맥외기기로부터 바인딩 요청.			
IN_BIND_REQ	맥내기기로부터 바인딩 요청.			
OUT_UNBIND_REQ	맥외기기로부터 바인딩 해제 요청.			
IN_UNBIND_REQ	맥내기기로부터 바인딩 해제 요청.			
PROFILE_LIST_REQ	요청 프로파일에 속하는 모든 바인딩 클러스터 아이디들을 요청.			
BIND_INFO_REQ	해당 클러스터 아이디에 대한 바인딩 정보 요청.			
BIND_RES	바인딩 요청 및 해제에 대한 결과를 반환.			
PROFILE_LIST_RES	프로파일에 속하는 바인딩 클러스터 아이디의 리스트를 반환.			
BIND_INFO_RES	바인딩 클러스터에 해당하는 바인딩 정보를 반환.			

위의 [표 1]은 proxy 서버와 location 서버 사이에 교환하는 패킷의 형식과 종류이다. 이 패킷은 mobile 기기와 proxy 서버와의 통신에서도 동일하게 사용된다.

### 3.3 바인딩 테이블

다음의 [표 2]는 맥내 기기와 맥외 기기간의 바인딩 정보를 저장하는 테이블 정보를 나타낸다. 테이블내의 정보를 검색할 때는 Profile 과 ClusterID를 사용해야 한다. 또한 바인딩되는 순서에 있어서 맥내기기 바인딩이 먼저 이루어지며, 그 다음으로 맥외 기기의 바인딩이 이루어지게 된다.

표 2. location 서버의 바인딩 테이블

Field	설명	영역
ProfileID	프로파일 (현재는 light profile 기준임.)	
ClusterID	Src와 dest 생이 갖는 유일한 아이디.	
DescName	현재 binding한 기기의 이름.	
SrcAddr	맥내 기기의 주소.	
SrcEip	맥내 기기내의 end point 주소.	
DestAddr	맥외 기기의 주소.	
DestEip	맥외 기기의 end point 주소. (공백가능)	
.....	..... (확장)	

Location 서버는 unbinding에 대한 정책을 맥내·외 기기에 대해 다음과 같이 가지고 있다. 첫 번째 정책은, 맥내 기기의 경우 맥내 기기가 proxy서버에게 응답하지 않을 경우에 proxy서버로부터 unbinding 요청을 수신한다. 두 번째 정책은, 맥외 기기의 경우 proxy 서버는 맥외 기기로부터 도착한 마지막 제어메시지의 도착 시간을 기억한 후, 그 시간에서 3분이 초과되면 unbinding을 요청한다.

## IV. 구현

### 1. 구현

본 논문의 구성은 홈 서버(Proxy Server)로 리눅스 머신을 사용하며, 맥내의 홈 기기에 tinyOS를 내장하였

고, 그리고 맥외의 기기는 가상의 모바일기기로 에뮬레이터를 구현하였다. 홈 서버와 Location 서버 그리고 데이터베이스는 분산 환경으로 구현하여 홈 네트워크 시스템의 유연성을 부과하였다. Base station은 홈 서버와 RS232C 통신으로 연결하여 맥내외간의 컨트롤 메시지를 중계한다. 또한 맥내의 기기들의 바인딩 및 해제 메시지를 서버에게 전달한다. Location 서버는 [표 1]에 정해진 메시지를 통하여 홈 서버와 데이터베이스와의 상태를 동기화시킨다.

### 2. 실행 화면

맥내 망의 각 노드들의 주소는 2bytes 주소체계를 사용하였다. 각각의 host 주소를 0x0001, 0x0002로 그리고 Node1의 기기코드는 "거실 등"으로 Node2의 기기 코드는 "거실 에어컨"으로 설정한 후 시스템을 다음과 같이 수행한다.

- 1) Location 서버 가동.
- 2) Proxy 서버 가동.
- 3) Node1(0x0001) 가동, Node2(0x0002) 가동.

각 노드는 가동과 동시에 약 2초간의 라우팅 설정 시간을 가지며, 그 시간이 경과하면 주기적인 바인딩 요청 메시지를 전송한다. 바인딩에 성공하여 결과메시지(BIND\_OK)를 수신 받으면 더 이상의 바인딩요청을 하지 않는다.

- 4) Proxy 서버의 화면에 해당 가전제품의 정보가 출력되고, 바인딩 상태가 ▲로 표시된다.
- 5) Mobile 기기를 구동한다. 해당 프로파일을 요청한 후 리스트가 나오면 원하는 기기와의 바인딩을 수행한다.
- 6) 바인딩이 끝나면 이제부터 해당 기기에 대해 On/Off 명령을 실행할 수 있다.
- 7) On명령을 전송하면 해당 기기에 Blue light에 불이 켜지며 proxy 서버의 STATUS가 ●으로 표시된다.
- 8) Off명령을 전송하면 해당 기기의 Blue light가 꺼지며 proxy 서버도 갱신한다. (Blue light: 노드의 3개의 LED 중 가장 우측.)



그림 6. DBS 실행화면

## V. 결론

홈 서버로 확장된 바인딩 기법을 통하여 맥외 망의 모바일 기기를 통하여 무선 ad hoc 네트워크로 연결된 맥내 망의 기기를 홈 서버를 통하여 통제할 수 있음을 확인하였다. 이러한 시스템을 통하여 맥내 망의 시스템과 맥외 망의 시스템 그리고 홈 서버 자체의 연구 개발 분야에 있어서 상호간의 발전된 호환성을 제공할 수 있을 것으로 기대한다. 그리고 홈 네트워크에 IPv6 및 RFID(active 혹은 passive 방식)가 활용되었을 때에도 시스템의 구조적 변화 없이 받아들일 수 있을 것이다. 또한 다양한 주소체계에 대해서 동적 바인딩 시스템은 기존의 주소변환 기술을 적용하지 않고 적용할 수 있을 것을 위의 구현을 통하여 확인할 수 있었다.

향후 과제로 ad hoc 네트워크에서 문제가 되고 있는 아웃 노드의 사멸을 인식하는 것과 동일한 문제인 unbinding에 대한 사멸과 방식에 대한 연구가 추가되어야 할 것이다. 마지막으로 맥내외의 기기에 대한 주소문제는 현재의 시스템으로 해결가능하나 향후 맥외 망의 기기들이 많아지고 그에 따른 주소문제가 대두될 것이다. 이에 대한 연구로 본 연구에 IPv6와 SIP를 향후 연구과제로 진행하고자 한다.

## 참고 문헌

- [1] 이재현, 권경희, "Home Network에서 ZigBee Binding을 이용한 맥내외 망의 연계방법에 대한 연구," 한국정보처리학회 춘계 학술발표대회 논문집, 제13권, 제1호, 2006(5).
- [2] <http://www.echelon.com>
- [3] <http://www.zigbee.org>
- [4] ZigBee Alliance, *ZigBee Specification v1.0*, December 14th, 2004.
- [5] D. B. Johnson, D. A. Maltz, and Y. C. Hu, *The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)*, IETF Draft, July 2004.
- [6] C. E. Perkins, E. M. Royer, and S. Das, *Ad hoc on-demand distance vector (AODV) Routing*, RFC 3561, Category: Experimental, July 2003.
- [7] M. Jiang, J. Li, and Y. C. Tay, *Cluster based routing protocol(CBRP)*, IETF MANET Working Group Internet-Draft, July 1999
- [8] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance Vector (DSDV) for Mobile Computers," Proc. Of the SIGCOMM 1994 Conference on

