

## 편리한 사물인터넷 서비스를 위한 IoT 관리 허브 연결 방법

김상현<sup>1</sup> · 김영돈<sup>1</sup> · 이창석<sup>2</sup> · 이동호<sup>2</sup> · 박현주<sup>2\*</sup>

### The Way of IoT Management Hub Connection for Convenient IoT Service

Sang-hyun Kim<sup>1</sup> · Young-don Kim<sup>1</sup> · Chang-seok Lee<sup>2</sup> · Dong-ho Lee<sup>2</sup> · Hyun-ju Park<sup>2\*</sup>

<sup>1</sup>Radiowave Engineering, Hanbat National University, Daejeon 305-719, Korea

<sup>2</sup>Information and Communication Engineering, Hanbat National University, Daejeon 305-719, Korea

#### 요 약

IoT 단말을 관리하는 IoT 관리 허브가 수집한 센싱데이터를 원격의 서버로 저장하기 위해서 인터넷 네트워킹이 가능해야 한다. 네트워크에 접속하기 위한 방법은 유선연결 방법과 WLAN AP에 접속하는 방법이 있는데, WLAN AP에 접속하기 위해서는 접속 가능한 영역의 AP 목록을 조회하고 특정 AP를 선택해야 한다. AP에 보안키가 설정되어 있는 경우에는 해당 보안키를 입력 할 수 있는 방법 또한 필요하다. 기존 방식에서는 IoT 관리 허브가 AP의 목록조회를 위한 디스플레이 장치가 필요했고, 목록을 선택하고 보안키를 입력하기 위한 입력 장치가 필요했다. 이러한 방식은 IoT 관리 허브가 디스플레이 장치와 입력 장치를 포함한 형태로 구성되기 때문에 단말의 소형화와 저가격화를 방해하게 된다. 또한 IoT 서비스가 대중화 되었을 때 사용자들에게 불편한 서비스로 인식될 우려가 있다. 본 논문에서는 스마트폰의 Wifi, 디스플레이 장치, 입력 장치를 이용하여 사용자에게 친숙한 방법으로 IoT 관리 허브를 WLAN 네트워크에 연결하는 방법을 제안한다.

#### ABSTRACT

IoT management hub has to request the WLAN AP list of the accessible areas in a place that is installed, or choose a specific WLAN AP you want to connect in order to be connected to the Internet. the specific method also is required to input the password when the security key was set. As that way, IoT management hub needs both the display device and the input device to see the list of WLAN AP and to input the security key. If the IoT management hub is consist of them, It is difficult to achieve the objective of the miniaturization and cost reduction. In this paper, we propose a method to connect to a WLAN AP network using a smart-phone without the display device and the input device.

**키워드** : 사물인터넷, 무선랜 네트워크, 허브 자동연결, 스마트폰 사용, IoT 서비스

**Key word** : IoT, WLAN Network, Hub Connection, Using Smart-phone, IoT Service

Received 16 July 2015, Revised 17 August 2015, Accepted 02 September 2015

\* Corresponding Author Hyun-Ju Park(E-mail:phj@hanbat.ac.kr, Tel:+82-42-821-1220)

Department of Information Communication Engineering, Hanbat National University, Daejeon 302-719, Korea

Open Access <http://dx.doi.org/10.6109/jkiice.2015.19.11.2656>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.  
Copyright © The Korea Institute of Information and Communication Engineering.

## I. 서 론

스마트폰의 보급으로 인하여 다양한 이종 스마트 기기들은 기술의 발전과 더불어 인터넷을 통해 연결되면서 실생활과 더욱 밀접해지고 있다. 본 논문에서 다루는 사물인터넷(Internet of Thing:IoT)기술이란 사물지능통신(Machine to Machine:M2M)의 확장으로서 단말 노드들을 인터넷에 연결시켜 발생하는 정보들을 이용하여 실체화하고 사용자들의 접근을 용이하게 만들어 다양한 서비스가 가능하도록 구현한 기술이다. 사물인터넷 기술은 활용범위가 매우 넓다. 시장조사기관 가트너에 따르면 2015년 ‘인터넷 연결 기기(connected things)’의 대수가 올해 보다 30% 증가한 49억 대, 2020년에는 250억 대의 장치들이 무선 인터넷을 기반으로 연결될 것이라고 예상하였다[1,2]. 따라서, 가까운 미래에는 IoT 기술뿐만 아니라, 편리한 IoT 서비스 또한 중요해질 것이라고 예상된다.

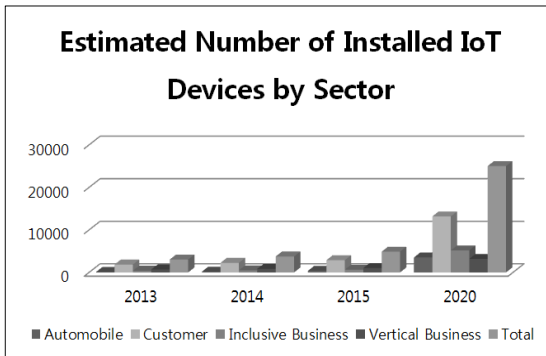


Fig. 1 Number of use, IoT Devices in the Field

IoT 기기들이 늘어나면서 Home Automation의 일환으로 Home Network에 IoT장비를 붙여 거주지의 자동화를 도모하는 Home IoT Network System 들이 점차 늘어나고 있다[3]. 본 논문에서 활용한 Home IoT Network System 에서는 IEEE 802.15.4의 Zigbee 프로토콜을 사용하여 메쉬업 서비스를 제공한다. 단말을 관리하는 IoT 관리 단말이 Coordinator 역할을 하고 센싱 디바이스나 액추에이터와 연결된 단말들이 End Device의 역할을 한다. [그림 2]의 Coordinator인 IoT 관리 허브는 IoT 서비스를 위하여 인터넷 연결이 필요하다. 본 시스템에서는 Access Point(AP)를 이용하여 무선인터넷에

접속하여 IoT 서비스를 하고 있다.

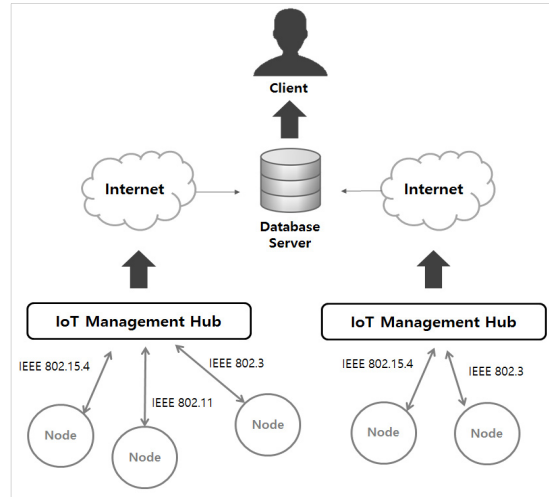


Fig. 2 Overview of home IoT Network System

[그림 2]와 같은 Home IoT Network 시스템에서 IoT 관리 허브가 AP에 접근하기 위하여 SSID의 검색과 선택, 그리고 보안키의 입력이 필요하다. 그렇기 때문에 SSID를 확인할 수 있는 디스플레이 장치와 보안키를 입력할 수 있는 입력 장치가 IoT 관리 허브에 장착되어야 한다. 하지만, 디스플레이 장치와 입력 장치가 IoT 관리 허브에 구성되어 있다면 IoT 관리 허브장치의 크기는 불필요하게 커질 것이고, 장치의 가격 또한 소비자에게 부담이 될 우려가 있다. 또한 이러한 제품이 사용자들에게 상품으로 판매되어 가정에 설치되어지는 경우 어려운 사용법 때문에 불편함을 야기할 우려가 있다.

본 논문에서는 IoT 단말의 소형화와 저가격화 그리고 편의성의 목표를 달성하기 위해 디스플레이 장치와 입력 장치가 없는 IoT 관리 허브를 구성한다. 그리고 스마트폰 애플리케이션을 이용하여 AP의 SSID와 보안키를 입력하는 네트워크 연결 처리 방법을 제안한다.

본 논문의 관련 연구는 사물인터넷, Web Application Server, Android, Shell Script로 이루어져 있고, 본론 구성은 Home IoT Network의 구성, IoT 관리 허브 단말, 연결 절차, 연결 기술 구현, 시스템 개선 효과 순서로 구성되어 있으며, 논문에 마지막 장에서 결론을 도출하고 향후 연구에 대해 논의한다.

## II. 관련 연구

### 2.1. 사물인터넷과 홈 네트워크

사물인터넷은 각종 사물에 컴퓨터 칩과 통신 기능을 내장하여 인터넷에 연결하는 기술을 의미하며, 약어로 짧게 'IoT'로 사용한다. 여기서 사물이란 가전제품, 모바일 장비, 웨어러블 컴퓨터 등 다양한 임베디드 시스템이 된다. 사물인터넷에 연결되는 사물들은 자신을 구별할 수 있는 유일한 아이디를 가져야 하고, 인터넷을 통한 통신 능력을 가져야 하고, 데이터를 처리하는 능력을 가져야 한다[4]. 대중적인 사물인터넷 기술로는 Home IoT Network System이 있다. 이 시스템에서는 IEEE 802.15.4, IEEE 802.3, IEEE 802.11의 프로토콜을 활용하여 IoT 관리 허브와의 네트워크를 구성하고 IoT 관리 허브가 WLAN에 연결되어 서비스 된다. 본 논문에서는 IoT 사물인 센서 단말과 액추에이터 단말, 소형 PC인 IoT 관리 단말 등을 구성한 Home IoT Network System 시스템을 활용하고, IoT 관리 허브가 WLAN에 연결되는 절차에 대하여 연구한다.

### 2.2. Web Application Server

웹 애플리케이션 서버(Web Application Server, 약자 WAS)는 인터넷 상에서 HTTP를 통해 사용자 컴퓨터나 장치에 애플리케이션을 수행해 주는 미들웨어(소프트웨어 엔진)이다. 웹 애플리케이션 서버는 동적 서버 콘텐츠를 수행하는 것으로 일반적인 웹 서버와 구별이 되며, 주로 데이터베이스 서버와 같이 수행이 된다. 한국에서는 일반적으로 "WAS" 또는 "WAS S/W"로 통칭하고 있으며 공공기관에서는 "웹 응용 서버"로 사용되고, 영어권에서는 "Application Server" (약자 AS)로 불린다. 웹 애플리케이션 서버는 대부분이 자바 기반으로 주로 Java EE 표준을 수용하고 있으나, 자바 기반이지만 Java EE 표준을 따르지 않는 제품과 .NET이나 Citrix 기반인 비Java 계열도 존재한다. 범용적으로 사용되는 Apache Tomcat은 아파치 그룹에서 자카르타 프로젝트(JakartaProject)로 개발한 공개SW WAS로 Java Servlet 및 JSP(Java Server Page) 기술에 대한 공식적인 참조 구현에 사용되는 서블릿 컨테이너이다. Apache Tomcat은 웹 서버와 연동하여 실행할 수 있는 Java 환경을 제공하여 JSP와 Java Servlet이 실행할 수 있는 환경을 제공한다. 또한 Apache Tomcat은 관리들을 통해 설정을

변경할 수 있고 XML10) 파일을 편집하여 설정할 수도 있다. Apache Tomcat에 내장된 웹 서버로만 웹 시스템을 구성할 수 있지만, 대규모의 사용자가 사용하는 시스템을 구축하려면 웹 서버와 연동하는 안정적인 시스템을 구축해야 한다[5]. 본 논문에서는 IoT 관리 허브에 Tomcat WAS를 설치하여 안드로이드와 통신하는 애플리케이션을 설계한다.

### 2.3. Android와 Wifi Manager

안드로이드는 앱 개발자들이 자바 언어로 응용 프로그램을 작성할 수 있게 하였으며, 컴파일된 바이트코드를 구동할 수 있는 런타임 라이브러리를 제공한다. 또한 안드로이드 소프트웨어 개발 키트(SDK: Software Development Kit)을 통해 응용 프로그램을 개발하기 위해 필요한 각종 도구들과 API를 제공한다[6].

본 논문에서는 IoT 관리 허브와 스마트폰의 접속 가능한 AP가 같다고 추정한다. 따라서 안드로이드의 Wifi Manager API를 활용하여 스마트폰의 AP 목록을 저장한다. 간단하게 WifiManager 참조변수를 선언하고 getSystemService() 메소드로 초기화하여 사용할 수 있다. 또한 안드로이드의 웹뷰를 활용하여 WifiManager로 호출한 AP의 정보를 URL을 통해 IoT 관리 허브의 웹서버로 파싱하는 것이 가능하다.

### 2.4. Shell Script

셸 스크립트는 셸에서 사용할 수 있는 명령어들의 조합을 모아서 만든 batch 파일이다. 스크립트라고 하는 것은 인터프리터에 의해 해석/실행되는 프로그램을 말하는데 어떤 종류의 인터프리터를 사용하는가에 따라 어떤 스크립트인가에 대한 이름이 정해진다. Perl Script, Tcl/Tk Script 등의 이름에서 사용하는 인터프리터를 알 수 있다. 셸 스크립트는 인터프리터로 셸을 사용하는 스크립트를 가리킨다. 또한 어떠한 셸을 사용하는가에 따라서 본(bourne) 셸 스크립트, C 셸 스크립트, 콘(Korn) 셸 스크립트, TC 셸 스크립트 등으로 나뉜다. 스크립트의 특징 중 하나는 그 내용이 텍스트 형식이기 때문에 일반 편집기로 볼 수 있다는 것이다. 스크립트 중 어떤 내용도 cat, vi, emacs 등의 편집기로 볼 수 있다. 스크립트 파일을 열어 보면 여러 명령들과 스크립트 문법의 조합으로 되어 있는 것을 확인할 수 있다[7]. 본 논문에서는 IoT 관리 허브를 무선 AP 모드로 동작 시키기

위한 모든 작업들을 쉘 스크립트를 작성하여 실행파일로 활용하였다.

### III. 본 론

#### 3.1. Home IoT Network 구성

본 논문에서 활용하고 있는 Home IoT Network에서는 크게 데이터서버, IoT 관리 허브 역할을 하는 소형 PC 그리고 IoT 단말인 센서/액추에이터 단말들로 구성된다. 통신은 저속 전송 속도와 근거리 통신을 위하여 Zigbee Alliance에서 개발한 무선 네트워크 기술인 Zigbee를 사용한다. Zigbee는 작은 크기로 전력 소모량이 적고 값이 저렴하여, 홈 네트워크 등 유비쿼터스 구축 솔루션으로 각광받고 있다. IoT 단말들은 통신 기능을 하는 Zigbee 모듈을 장착하고 있고 IoT 관리 허브 역시 Zigbee 모듈로 센싱데이터를 수집하고 액추에이터 단말에 명령을 내린다. 소형 PC로는 Raspberry Pi를 활용하였다. Raspberry Pi는 성능이 뛰어나면서도 저렴한 가격이 특징이고 데비안, 아치 리눅스, QtonPi등의 리눅스 배포판을 제공하고 있어 사용이 편리하다. 또한 USB 이더넷이나 Wifi 어댑터를 활용하여 외부와 네트워크가 가능하기 때문에 IoT 관리 허브로 사용한다.

본 논문에서 활용하는 Home IoT Network 환경에서는 IoT 관리 허브를 중심으로 스타형 토폴로지 네트워크를 구성한다. 각 End Device인 센서 노드를 보유한 단말들은 온습도 센서, 이산화탄소 센서, 적외선 감지센서 등 다양한 센싱 기능을 담당하고 있으며, 이 단말들은 소형 PC와 연결되어 데이터를 저장한다. IoT 관리 허브는 Coordinator로서 액추에이터 노드를 보유한 단말 노드에게 패킷을 보내 단말들을 적절하게 제어한다. 이 네트워크는 유선통신, Zigbee, Bluetooth, WLAN 등의 근거리 무선통신을 지원한다.

#### 3.2. IoT 관리 허브 단말

##### 3.2.1. IoT 관리 허브 단말의 역할 및 구성

IoT 관리 허브는 일반 IoT 단말들을 관리하는데 센서 노드를 보유한 단말(센서 단말), 액추에이터 노드를 보유한 단말(액추에이터 단말), 센서노드와 액추에이터 노드를 복합 보유한 단말(복합 단말) 등을 관리한다. IoT 관리 허브와 일반 단말들은 저전력, 저가격 통신 모듈인

zigbee를 이용하여 자체적으로 설정된 연결 절차를 통해 통신 가능한 상태이다. [그림 3]과 같이 IoT 관리 허브는 센서 단말과 복합단말에게서 센싱 데이터를 수집하고, 이 데이터를 통하여 액추에이터 단말과 복합 단말에게 명령을 보내 특정한 액추에이팅 기능을 명령한다.

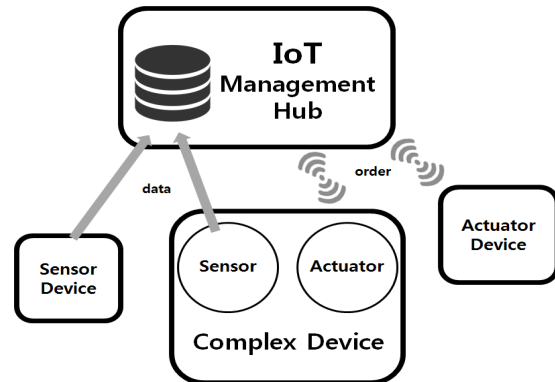


Fig. 3 A role of IoT management hub

IoT 관리 허브의 단말은 Raspbian OS, Java, MySQL Database, Tomcat Web Application Server, Application으로 이루어져 있다. 본 논문에서 활용하는 IoT 관리 허브 단말은 MySQL Database, Tomcat WAS, Java를 이용하여 IoT 단말에서 오는 센싱 데이터를 수신하는 역할을 한다.

WISoft IAQ(Indoor Air Quality)						
Date	SEQ.	Name	CO	CO2	Temp.	
2015-03-18 14:11:52.0	230326	SMS#11	-1.00	2080.00	12.20	
2015-03-18 14:11:51.0	230325	SMS#11	-1.00	2057.00	12.20	
2015-03-18 14:11:39.0	230324	SMS#11	-1.00	2053.00	12.20	
2015-03-18 14:11:38.0	230323	SMS#11	-1.00	2080.00	12.20	
2015-03-18 14:11:26.0	230322	SMS#11	-1.00	2089.00	12.20	
2015-03-18 14:11:25.0	230321	SMS#11	-1.00	2099.00	12.20	
2015-03-18 14:11:15.0	230320	SMS#11	-1.00	2039.00	12.20	
2015-03-18 14:11:13.0	230319	SMS#11	-1.00	2094.00	12.20	
2015-03-18 14:11:00.0	230318	SMS#11	-1.00	2048.00	12.20	
2015-03-18 14:10:59.0	230317	SMS#11	-1.00	2081.00	12.20	
2015-03-18 14:10:48.0	230316	SMS#11	-1.00	2081.00	12.20	
2015-03-18 14:10:47.0	230315	SMS#11	-1.00	2090.00	12.20	

Fig. 4 The received data from IoT device

본 논문에서는 IoT 관리 허브의 Tomcat WAS와 Java를 이용하여 안드로이드에서 파싱되는 SSID와 보안키를 받는 Application을 구현한다.

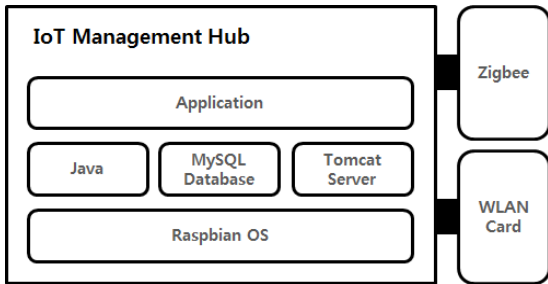


Fig. 5 The Specification of IoT Management Hub

3.2.2. 관리 허브의 WLAN AP 역할

IoT 관리 허브는 WLAN Card를 사용하여 네트워크 커버리지에 있는 AP를 잡아 인터넷에 접속한다. 또한 AP 모드를 지원하는 WLAN Card를 사용하여 IoT 관리 허브를 WLAN AP로 활용하였다. IoT 관리 허브를 WLAN AP로 사용하는 경우에 스마트폰으로 IoT 관리 허브에 접속하는 것이 가능하다. 위에서 언급했던 것과 같이 WLAN Card가 AP의 역할을 지원할 경우에 설정 파일을 변경하여 IoT 관리 허브를 AP로 사용하는 것이 가능하다.

Table. 1 Description of IoT Management Hub

Application	The Parsing the SSID and security key on your smartphone, sensing data transfer and actuator control
Java	The Basic language
MySQL	The Database of IoT management hub
Tomcat	The web application server of IoT management hub
Raspbian OS	The operating system of IoT management hub
Zigbee	The communication between IoT management hub and sensor/actuator device
WLAN Card	The wireless local area network card for IoT management hub

[그림 6]은 본 논문에서 사용한 IoT 관리 허브와 IoT 단말의 실제 모습이다. IoT 관리 허브는 대중적으로 널리 사용되는 Raspberry Pi를 사용하였고, IoT 단말은 자체 제작하였다. [그림 6]의 IoT 단말은 이산화탄소의 양을 측정하는 센서 단말이다.

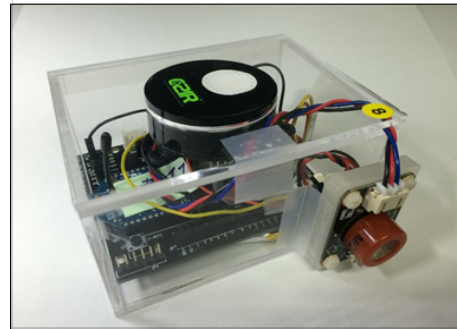
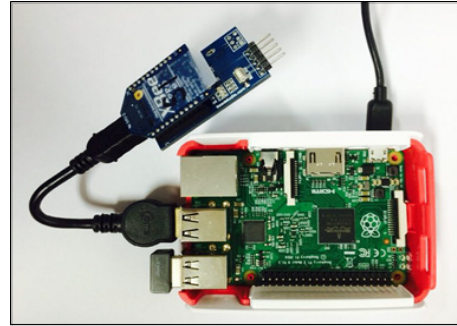


Fig. 6 The photos of IoT management hub and device

3.3. 연결 절차

3.3.1. 연결 과정

다음 과정은 사용자가 IoT 관리 허브를 특정 AP에 연결시키기 위한 과정이다. 먼저 스마트폰을 이용하여 WLAN AP 모드로 동작하는 IoT 관리 허브에 접속한다. 스마트폰과 IoT 관리 허브의 네트워크 커버리지는 거의 같기 때문에, 스마트폰이 접속 가능한 AP 목록과 IoT 관리 허브가 접속 가능한 AP 목록은 같다고 판단할 수 있다. 따라서 스마트폰의 AP 목록에 있는 특정 AP의 SSID와 보안키를 IoT 관리 허브의 네트워크 설정에 저장한다. 그 후에 원격이나 수동으로 IoT 관리 허브를 Reboot 해주면, IoT 관리 허브는 설정에 등록되어 있는 특정 AP를 찾아 커넥션을 완료한다.

이 과정들을 수행하기 위하여 첫 번째로 IoT 관리 허브를 WLAN AP 모드로 동작시킨다. WLAN AP 모드를 동작시키기 위하여 Shell Script를 만들어 Raspbian OS에 탑재한다. 두 번째로 안드로이드 어플리케이션으로 스마트폰의 AP 목록을 받아 저장한다. 이 부분에서는 안드로이드의 Wifi API를 사용하여 AP 목록을 list view에 출력한다. list view에 있는 특정 AP를 클릭하면 URI를 통하여 SSID값이 파싱된다. 세 번째로 IoT 관리

허브에 WAS를 설치하여 안드로이드 웹뷰를 통하여 SSID와 보안키를 받아 IoT 관리 허브에 저장한다.

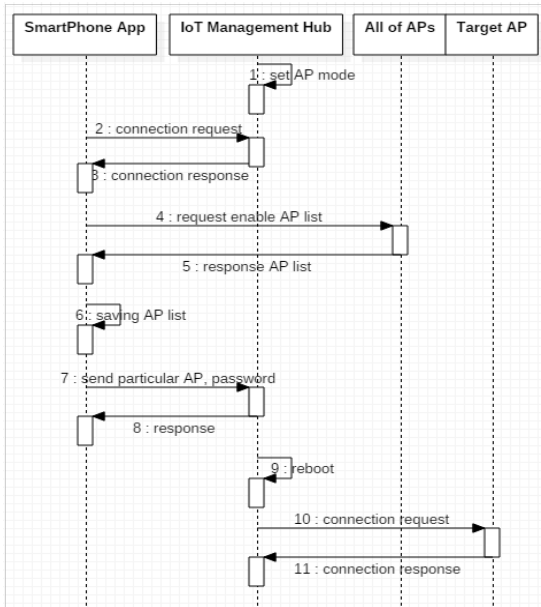


Fig. 7 System sequence chart

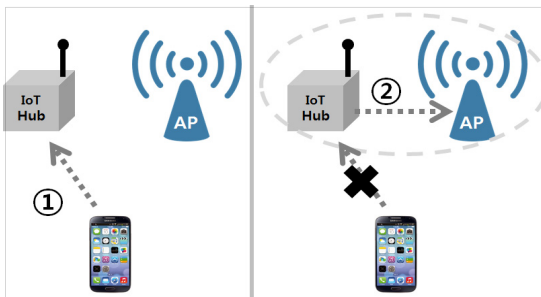


Fig. 8 Overview of the system

[그림 8]에 1번 과정은 WLAN AP 모드로 동작하는 IoT 관리 허브에 스마트폰이 접속한 상황이다. 2번 과정은 스마트폰으로 SSID와 보안키를 IoT 관리 허브에 저장하고 원격으로 Reboot되는 상황이다. Reboot시에 설정 값이 등록되면서 특정 AP에 접속하는 것이 가능하다. 연결 과정을 다음 문단에 정리한다.

1. 무선 AP 모드로 동작하는 IoT Hub에 스마트폰이 접속한다.

2. 어플리케이션을 실행하여 IoT Hub에서 동작하는 WAS 프로그램에 접속한다.
3. 스마트폰 바운더리의 AP 리스트를 어플리케이션에 출력한다.
4. AP를 선택하고 비밀번호를 입력한다.
5. IoT Hub의 설정파일 정보에 AP 정보가 등록된다.
6. 재부팅되어 AP 모드가 종료되고 설정파일 정보를 읽어 선택한 AP에 접속한다.

이 과정에서 스마트폰과 IoT 관리 허브의 바운더리가 비슷하기 때문에 IoT 관리 허브가 스마트폰 바운더리에 있는 AP를 설정한다 하더라도 큰 문제가 없다. 연결 기술에 관한 자세한 내용은 다음 절에서 설명한다.

### 3.4. 연결 기술 구현

#### 3.4.1. WLAN AP 모드

IoT 관리 허브를 WLAN AP 모드로 동작시키기 위해서 hostapd와 dhcp 서버 그리고 wpa\_supplicant 패키지를 설치해야 한다. 그 후에 hostapd.conf 파일에 무선 랜 카드 정보, WPA 설정, SSID 및 보안키 설정 등을 통하여 WLAN AP 모드를 설정한다. 그 후 hostapd 파일에 hostapd.conf 설정 파일의 경로를 참조하여 hostapd 설정을 완료한다. dhcp 서버로는 isc-dhcp-server 패키지를 설치하여 활용한다. hostapd와 마찬가지로 dhcpd.conf 파일에 dhcp 설정을 할당해 줄 private IP의 범위 지정 및 서브넷 마스크, 라우터 설정등을 기입하고 경로 설정을 지정하여 완료한다.

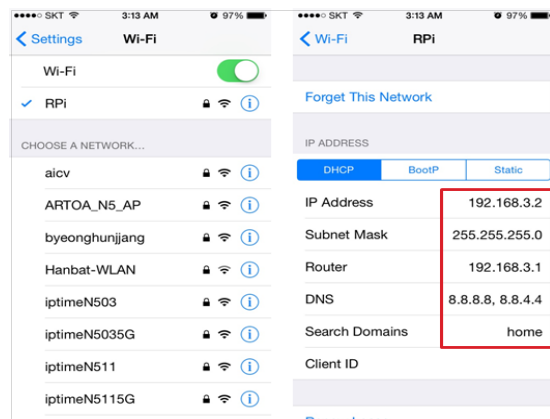


Fig. 9 The Screen of AP mode Connection



그 후 IoT 관리 허브에 설치된 OS의 네트워크 설정 파일에서 192.168.x.1의 주소를 고정하도록 설정하고 Reboot시에 AP와 dhcp 가 실행되도록 rc.d 파일을 수정한다. IoT 관리 허브가 WLAN AP 모드와 dhcp 서버로 동작하고 있는 경우, 스마트폰으로 접속하여 AP 정보를 확인해 보면 [그림 9]과 같은 화면을 볼 수 있다. [그림 9]과 같은 경우는 192.168.3.1의 주소가 dhcp 서버로 동작함을 알 수 있다. IoT 관리 허브를 WLAN AP로 동작시키기 위한 모든 과정들을 Shell Script를 이용하여 하나의 파일로 구성하였다.

```

GNU nano 2.2.6 File: wisoft_ap.sh
#!/bin/bash

echo "apt-get update"
apt-get update

echo "Install hostapd isc-dhcp-server iptables"
apt-get -y install hostapd isc-dhcp-server iptables

echo "Check Wifi Chipset"
if [ $(lsusb | grep "RTL8188CUS\|RTL8192CU" | wc -l) -eq 0 ]; then
    exit 1
fi

conf_file="/etc/hostapd/hostapd.conf"
read -p "input SSID : " SSID
read -s -p "input password : " password

echo "Create ${conf_file}"
echo "interface=wlan0" > ${conf_file}
echo "ssid=${SSID}" >> ${conf_file}
echo "channel=1" >> ${conf_file}
echo "macaddr_acl=0" >> ${conf_file}
echo "auth_algs=1" >> ${conf_file}
echo "ignore_broadcast_ssid=0" >> ${conf_file}
echo "wpa=2" >> ${conf_file}
echo "wpa_passphrase=${password}" >> ${conf_file}
echo "wpa_key_mgmt=WPA-PSK" >> ${conf_file}
echo "wpa_pairwise=CCMP" >> ${conf_file}
echo "driver=rtl871xdrv" >> ${conf_file}
    
```

Fig. 10 The Part of Shell Script

Shell Script 파일에는 패키지 설치, WLAN AP 모드를 지원하는 무선랜카드의 여부확인, echo명령어를 이용한 설정파일 변경, 권한 설정 등의 기능을 탑재하였다. 개발자들은 이 Shell Script 파일을 실행함으로써 손쉽게 IoT 관리 허브를 AP 모드로 설정하여 배포할 수 있다.

### 3.4.2. 데이터 수신을 위한 JSP Web page

WAS 에 안드로이드 Application에서 파싱될 데이터를 받을 수 있는 페이지를 jsp를 이용하여 구현하였다. 런타임 API를 이용하여 설정파일을 열고, 스트림을 이용하여 안드로이드에서 파싱되는 SSID와 보안키를 쓸 수 있도록 구현하였다. 그 후에 사용자가 입력 버튼을 누르면 Reboot 될 수 있도록 구현하였다. tomcat 서버에

서 자동으로 Reboot 명령을 실행하기 위한 권한을 주기 위하여 /etc/sudoers에 tomcat 권한을 추가하였다.

```

pi@raspberrypi ~ $ sudo vi /etc/sudoers
#includedir /etc/sudoers.d
pi ALL=(ALL) NOPASSWD: ALL
tomcat7 ALL=(ALL) NOPASSWD: ALL
    
```

Fig. 11 The Setting of Authorization

### 3.4.3. 데이터 파싱을 위한 스마트폰 Application

안드로이드 개발환경에서는 Wifi API를 활용하여 list view에 접속 가능한 AP 목록을 나열하고 클릭할 수 있도록 구현하였다. list view에 있는 SSID를 클릭하면, URI 파싱을 통하여 WAS에서 실행되고 있는 페이지에 SSID 값이 고정으로 입력된다. 이는 사용자가 실수로 SSID를 잘못 입력하는 상황을 미연에 방지하는 효과가 있다. 사용자는 안드로이드 어플리케이션에서 보안키를 입력하는 것만으로도 IoT 관리 허브와 AP를 연결시키는 것이 가능하다.

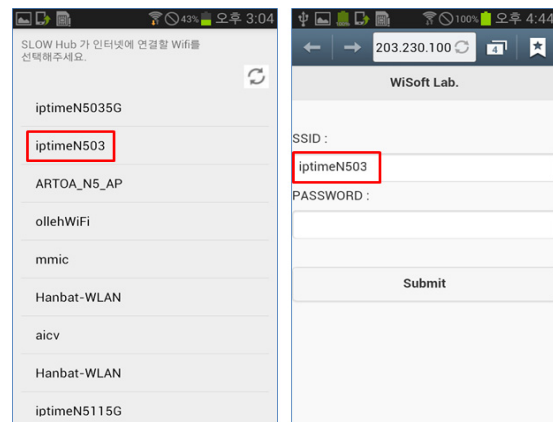


Fig. 12 The List of AP and Web View

## 3.5. 시스템 개선 효과

### 3.5.1. 사용자 편의성

편리한 IoT 서비스를 확인하기 위한 일환으로, 사용자에게 IoT 관리 허브를 무선 AP에 연결하는 테스트가 진행되었다. IoT 관리 허브의 재부팅 시간은 항상 같기 때문에, 사용자가 스마트폰 Application을 사용하여 SSID 선택과 비밀번호 입력까지 걸리는 시간을 측정하였다. Application을 처음 사용하는 10명의 일반 사용자

에게 테스트를 진행한 결과 제일 빠른 연결 시간은 18.66초, 제일 느린 연결 시간은 28.27초로 사용자 전원 모두가 30초 내에 IoT 관리 허브를 네트워크에 연결하였다. 기존 디스플레이 장치와 입력장치를 이용하여 걸리는 시간과 비교한다면 상당히 만족스러운 결과를 보여주고 있다. 본 연구의 결과로 IoT 서비스에 있어서 사용자에게 친숙하고 관리가 편한 환경을 제공한 것을 증명하였다.

### 3.5.2. 소형화 및 저가격화

사물인터넷 기술도 점점 대중화됨에 따라서 가격 경쟁을 해야 할 것이다. 이 목적을 달성하기 위해서는 단말들의 부피를 소형화시키는 것이 필수적이다. 위에 언급했던 바와 같이, 기존 기술에서는 IoT 관리 허브에 디스플레이 장치와 입력 장치가 같이 구성되었다. 이 장치들은 비교적 고가이고 부피가 크기 때문에 문제점이 되고 있다. 실제로 IoT 관리 허브에 일반 모니터와 키보드를 연결했을 때, 일반적인 소형 디스플레이 장치 모듈(가로 약 53mm, 세로 약 56mm, 높이 약 21mm)과 소형 입력 장치(가로 약 100mm, 세로 약 80mm, 높이 약 30mm) 모듈을 장착했을 때와 비교했을 때, 소형화 및 저가격화의 목적을 달성한 것을 알 수 있다.

## IV. 결 론

### 4.1. 결론

기존 Home IoT Network 시스템에서는 IoT 관리 허브가 WLAN AP에 접근하기 위해서 AP의 SSID를 보여줄 디스플레이 장치와 AP에 걸려있는 보안키를 입력하기 위해서 입력 장치가 필요하였다. 이 디스플레이 장치와 입력 장치가 IoT 관리 허브에 구성된다면 본 연구의 목적인 저가격화 그리고 소형화의 목표를 달성하기가 어려웠다. 하지만 본 연구를 통하여 디스플레이 장치와 입력 장치를 구성할 필요가 없어졌고 연구의 목적인 저가격화, 소형화의 목적을 달성하였다. 또한 IoT 서비스를 함에 있어서 기존의 방식은 사용자들에게 불편함을 줄 수 있었다. 하지만 사용자에게 친숙한 스마트폰을 이용한 연결 방법을 제시함으로써 IoT 서비스의 편리성을 도모하였다.

### 4.2. 향후 연구

현재 시스템에서 사용자가 실수로 보안키를 잘못 입력하고 Reboot할 경우에 IoT 관리 허브가 WLAN AP 모드로 다시 돌아와 재설정을 할 수 없다. 또한 보안키가 다르기 때문에 네트워크 설정 값이 달라 특정 AP에 접속도 하지 못한 상태가 된다. 이 상황을 배제하기 위하여 선택할 AP의 SSID와 보안키가 일치하는지 여부를 확인한 후에 SSID와 보안키 값을 둘 다 파싱하여 사용자의 실수를 100% 방지하는 것이 향후 연구의 목적이다.

## ACKNOWLEDGMENTS

This research was supported by the research fund of Hanbat National University in 2013

## REFERENCES

- [1] <http://www.gartner.com/technology>
- [2] J.-Y Park and N.-H Park, "Entity Authentication Scheme for Secure WEB of Things Applications", *The Korean Institute of Communications and Information Sciences*, v.38 no.5B, pp.394-400, 2013.
- [3] S.-H Kim, Y.-D Kim, J.-H Lee, C.-S Oh, M.-S Seo, C.-S Lee and H.-J Park, "A Study on The Procedure That Sensor Devices Are Connected to Home Server in IoT Environment", *The Korean Institute of Communications and Information Sciences*, v.6 no.6, pp.1263-1265, 2014.
- [4] C.-S Oh, M.-S Seo, J.-H Lee, S.-H Kim, Y.-D Kim and H.-J Park "Indoor Air Quality Monitoring Systems in the IoT Environment" *The Journal of Korean Institute of Communications and Information Sciences '15-05* Vol.40 No.05.
- [5] E.-S Jo, "A Study on Migration Process of Open Source Software WAS", Soongsil University, Software Engineering, 2014.
- [6] H.-K Kim, "Design and Implementation of Android-based Smart Application Platform for Controlling Various Home Appliances and Low Power Consumption", Pusan National University, Department of Electrical and Computer Engineering, 2013.
- [7] <http://www.linuxlab.co.kr/docs/10-4.htm>





**김상현(Sang-Hyun Kim)**

2014년 2월 : 한밭대학교 전파공학과 (공학사)  
2014년 3월~현재 : 한밭대학교 정보통신전문대학원 전파공학과 석사과정  
※관심분야 : 사물인터넷, 웹 프로그래밍, 데이터베이스



**김영돈(Young-Don Kim)**

2000년 2월 : 대전산업대학교 도시공학과 (공학사)  
2004년 2월 : 한밭대학교 정보통신전문대학원 정보통신공학과 (공학석사)  
2013년 2월 : 한밭대학교 정보통신전문대학원 전파공학과 (공학박사)  
※관심분야 : 사물인터넷, 자연어처리, 자료처리, 데이터베이스



**이창석(Chang-Seok Lee)**

1986년 2월 : 한국과학기술원 전기 및 전 공학과 (공학석사)  
1996년 2월 : 한국과학기술원 전기 및 전자공학과 (공학박사)  
1998년 4월~현재 : 국립 한밭대학교 정보통신공학과 교수  
※관심분야 : 반도체 및 통신 소자, 고주파회로



**이동호(Dong-Ho Lee)**

2000년 2월 : 한국과학기술원 전기 및 전자공학과 (공학사)  
2002년 2월 : 한국과학기술원 전기 및 전자공학과 (공학석사)  
2007년 2월 : 한국과학기술원 전기 및 전자공학과 (공학박사)  
2007년 4월 ~ 2009년 9월 : 미국 조지아텍 박사 후 과정  
2009년 10월 ~ 2010년 10월 : 미국 Skyworks Solutions, Inc., Senior Engineer  
2010년 11월 ~ 현재 : 국립 한밭대학교 정보통신공학과 부교수  
※관심분야 : RFIC, 전력증폭기, 레이더



**박현주(Hyun-Ju Park)**

1990년 2월 : 서울시립대학교 전산통계학과 (공학사)  
1992년 2월 : 서울대학교 전산학과 (공학석사)  
1997년 2월 : 서울대학교 전산학과 (공학박사)  
1998년 4월~2000년 : 3월 대전산업대학교 정보통신공학과 전임강사  
2000년 4월~현재 : 국립 한밭대학교 정보통신공학과 교수  
※관심분야 : 프로그래밍 언어, 운영체제, 데이터베이스