

논문 2010-4-2

# 안드로이드 모바일 플랫폼을 이용한 홈 네트워크 응용

## Home Network Application using Android Mobile Platform

최진엽\*, 이상정\*\*, 전병찬\*\*\*

Jin Yeop Choi\*, Sang-Jeong Lee\*\*, Byoung-Chan Jeon\*\*\*

**요약** 최근 모바일 플랫폼과 홈 네트워크 시스템과의 융합이 널리 보급되어 보편화되고 있다. 또한 모바일 기술의 발달로 손안의 작은 PC라고 불리는 스마트폰이 보급이 되면서 기존 핸드폰에서의 활용을 넘어 스마트폰에 최적화된 홈 네트워크 응용 개발이 요구되고 있다. 구글에서 개발한 안드로이드 모바일 플랫폼은 운영체제, 미들웨어 및 주요 응용프로그램을 포함한 오픈소스로 다양한 응용을 개발 할 수 있고, 모바일 기기뿐만 아니라 셋톱박스과 가전제품 등에도 탑재가 가능하다. 본 논문에서는 안드로이드 플랫폼의 스마트폰 상에서 주택 방안의 상태(온도, 습도, 실시간 영상 등)에 관한 정보의 실시간 모니터링과 맥내의 가전기기 등을 제어하는 홈 네트워크 서비스를 설계 구현한다.

**Abstract** Recently, there are many home network applications using the convergence between mobile platforms and home network system. As smart phones which are called small PC in the hand becoming popular, the development of home network application beyond the one of cellular phones is required. Android mobile platform of Google includes OS, middleware and primary applications. Also, in addition to smart phones, it can be mounted on various devices such as set-top box and household appliances. In this paper, home network services on smart phones of Android platform are designed and implemented. The services provide real-time monitoring information (temperature, humidity, real-time video) of rooms and appliance control in a house.

**Key Words**: 홈 네트워크 (Home Network), 안드로이드 플랫폼 (Android Mobile platform), 마이크로프로세서 (Microprocessor), 영상 처리 (Imaging Processing), 가전 기기 (Appliance)

### 1. 서 론

현재의 IT 환경은 인터넷과 모바일 기술의 발전과 대중화로 모바일을 통한 IT 생활 단계로 진입하고 있다. 모바일을 이용한 서비스의 개발은 90년대 중반부터 업무의 효율성이나 재고의 관리 등을 위해 물류 관리, 영업 자원, 그룹웨어에서 시작되었고 이것이 오늘날의 텔레매틱스 서비스나 위치 기반 서비스의 기반이 되고 있다.<sup>[1]</sup> 특히 유비쿼터스 시스템에 대한 관심이 높아지면서 다양한 응

용 가능성을 가진 스마트폰이 유비쿼터스의 중심으로 급부상 하고 있으며 각 산업 분야에서 스마트폰을 이용한 응용 서비스에 대한 연구가 활발히 이루어지고 있다. 유/무선 네트워크의 통합의 대표적인 사례가 홈 네트워크 서비스이며 홈 네트워크 서비스의 경우 모바일의 융합을 통해 방 안의 상태를 실시간으로 확인하며 방 안에 설치된 카메라를 통해 실시간으로 영상을 확인하는 서비스를 제공한다. 실제로 여러 홈 네트워크 업체에서 기존의 모바일 플랫폼 기반의 응용이 출시가 되었다.

모바일 기술의 발달로 손 안의 PC라고 불리는 스마트폰의 응용 범위가 점점 넓어지고 있으며 각종 핸드폰 제조업체나 세계 유수의 IT기업들이 자사만의 특화된 스마트폰용 OS를 출시하였다. 기존에 출시된 윈도우 모바일

\*준회원, 순천향대학교 컴퓨터학과

\*\*정회원, 순천향대학교 컴퓨터학과

\*\*\*정회원, 청운대학교 방송영상학과 (교신저자)

접수일자 2010.5.17, 수정일자 2010.7.16

게재확정일자 2010.8.11

플랫폼은 플랫폼 자체가 폐쇄적이고 단일 플랫폼에 종속되는 단점을 가지고 있다. 따라서 응용 범위가 좁고 다양한 응용 서비스를 제공 하는데 제약이 따른다. 반면 안드로이드 플랫폼은 개방형 OS로 다양한 기기에 탑재가 가능하고 응용의 범위도 기존의 플랫폼보다 넓다. 이에 따라 홈 네트워크 응용도 기존의 모바일 플랫폼 응용 보다 안드로이드 플랫폼의 특성에 맞는 응용의 개발 필요성이 요구 되고 있다.<sup>[2]</sup>

본 논문에서는 Google의 모바일 플랫폼 안드로이드를 이용하여 각 방의 상태를 실시간으로 확인하고 방에 설치된 각 가전을 제어하는 시스템과 각 방의 영상을 스마트폰으로 실시간 확인 할 수 있는 홈 네트워크 시스템을 설계하고 이를 위한 응용을 설계 구현한다. 각 방에 설치된 Zigbee 센서를 이용하여 방안에 대한 정보를 전송, 데이터베이스에 저장하고 WIFI나 3G망을 통해 사용자가 언제든지 스마트폰을 통해 온도, 습도, 조도에 대한 정보를 표시한다. 또한 마이크로 컨트롤러 시스템을 통해 각 가전의 전원을 연결하고 사용자가 스마트폰을 통해 가전을 원격으로 제어하며 사용자가 적정 온도/습도를 설정함으로써 자동으로 온도를 조절하는 시스템을 개발한다. 또한 각 방에 카메라를 설치하고 영상 데이터를 서버를 통해 전송하여 스마트폰에서 방안의 상태를 영상으로 실시간으로 확인하는 기능을 구현한다.<sup>[9]</sup>

## II. 이론적 배경

### 1. 안드로이드 모바일 플랫폼

구글에서 2007년 11월에 발표한 안드로이드 플랫폼은 모바일 운영체제로 운영체제, 미들웨어 및 주요 응용프로그램을 포함하는 모바일 기기용 소프트웨어 모음을 의미한다. 안드로이드는 컴포넌트의 재사용과 재배치가 가능한 응용 프레임워크와 모바일 디바이스에 최적화 된 Dalvik 가상머신, Webkit 엔진 기반의 통합 브라우저, 구조화된 데이터 스토리지를 위한 SQLite를 가지고 있다. 디바이스 에뮬레이터, 디버깅 툴, 메모리 및 퍼포먼스 프로파일링 그리고 이클립스 IDE를 위한 플러그인을 포함하는 풍부한 개발환경을 지원한다. 모바일 플랫폼으로서의 안드로이드 플랫폼의 장점은 자기가 개발한 응용을 안드로이드 마켓을 이용해 자유롭게 올리고 다른 사람이 개발한 응용을 받을 수 있는 자체 모바일 마켓 플레이스를 운영하고 있다는 점이다. 안드로이드 마켓에는 개발

자가 안드로이드 응용을 개발해 자유롭게 올릴 수 있으며 안드로이드 마켓을 통해 사람들에게 무료나 유료로 응용을 다운로드 할 수가 있다. 안드로이드 마켓에는 약 5만 여개의 유료, 무료 응용이 업로드 되고 있으며 응용 수는 빠르게 증가하고 있다.

그림 1. 은 안드로이드 시스템의 구조를 나타내고 있다. 안드로이드의 시스템 구조를 보면 상위 계층으로 응용, 응용 프레임 워크가 존재하며 하위 계층으로는 라이브러리, 안드로이드 런타임, 리눅스 커널로 이루어져 있다. 2.6 기반의 리눅스 커널은 카메라, 키보드와 같은 각 장치를 제어하고 전원 관리와 IPC Binder 드라이버를 담당한다. 안드로이드 런타임은 Dalvik을 포함하는 가상 머신과 Core 라이브러리를 포함하고 있다. 안드로이드 라이브러리는 안드로이드를 구성하고 있는 C와 C++ 라이브러리로 이루어져 있으며 아래와 같은 라이브러리로 구성되어 있다. System C Library는 BSD 기반의 Standard C 시스템으로 이루어져 있고 Media Library는 MPEG4, H264를 지원하는 OpenCore 라이브러리를 포함한다. Surface Manager는 2D와 3D의 여러 개의 화면 레이어를 화면에 표시하기 위한 매니저 역할을 담당하고 LibWebCore는 웹 브라우저 라이브러리를 포함하며, 안드로이드 자체 데이터베이스인 SQLite와 OpenGL 라이브러리를 가지고 있다. 응용 프레임 워크 층은 응용을 지원하는 각 매니저로 구성 되어 있다. 안드로이드 화면을 구성하는 View를 관리하는 View Manager, 응용과 응용 사이의 데이터 공유를 관리하는 Content Provider, 그래픽과 레이아웃 파일을 관리하는 Resource Manager, 안드로이드 상태를 알리는 Notification Manager, 안드로이드 응용을 구성하는 Activity를 관리하는 Activity Manager 등으로 구성 되어 있다.<sup>[3]</sup>

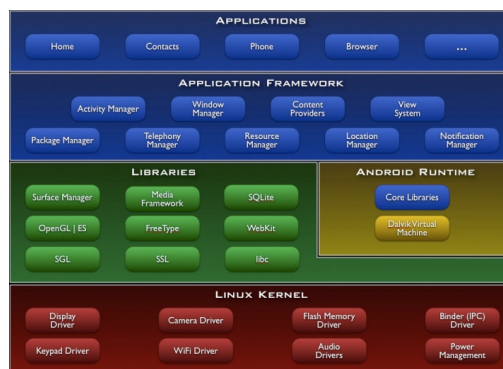


그림 1. 안드로이드 시스템 구조  
Fig 1. Android System Architecture

## 2. 안드로이드 플랫폼과 홈 네트워크

구글은 안드로이드 시스템 부분을 담당하는 코어에 대한 대부분의 소스 코드를 공개함으로써 오픈 모바일 플랫폼을 지향하고자 하였다. 이것은 안드로이드 플랫폼이 모바일에 국한되지 않고 셋톱박스나 인터넷 전화 등과 같은 태블릿에 다양한 가전에 탑재가 될 수 있음을 의미한다. 향후 홈 안드로이드 기반의 통합 홈 네트워크 시스템을 구성하게 되면 사용자는 안드로이드 단말기를 통해 TV를 원격으로 동작하게 하거나 셋톱박스를 통해 프로그램을 미리 예약 녹화 할 수 있고 세탁기 타이머를 설정하는 등 안드로이드가 탑재된 모바일 플랫폼으로 모든 정보 가전을 제어 할 수 있다. 그림 2는 TouchRevolution사에서 제작한 안드로이드 기반에서 동작하는 인터넷 전화기와 세탁기 정보가전을 통해 다양한 안드로이드의 응용 범위를 보여주고 있다.



그림 2. 안드로이드 기반의 정보 가전  
Fig 2. Android-based Appliance

Zigbee는 저가, 저전력, 무선 매쉬 네트워킹의 표준이고 저가이기 때문에 무선 제어 컨트롤 및 모니터링 응용 분야에 광범위하게 배치될 수 있다. 또한 저 전력이기 때문에 배터리 수명이 길어진다. 메시 네트워킹 기능을 하기 때문에 넓은 영역 범위에서 고 수준의 신뢰성을 제공한다. Zigbee는 산업, 과학, 의학용 무선 주파수 내에서 작동하며 유럽에서는 868MHz, 미국과 오스트레일리아에서는 915MHz, 범세계적으로는 2.4GHz 무선 주파수를 사용한다. Zigbee는 다른 무선 개인 통신망들과는 달리 비교적 저렴하고 간단한 기술로 이루어져 있다. 따라서 적은 데이터 전송량, 적은 전력 소모량, 보안성을 갖춘 네트워킹 등에 최적화 되어 있다. Zigbee의 기본 기술인 IEEE 802.15.4는 상호 운용성이나 무선 RF 기능 응용에서 많은 이점을 제공하기 때문에 홈 네트워크나 빌딩 자동화, 공장 자동화, 헬스 케어에 응용 될 수 있다.<sup>[4]</sup> 홈 네트워크 시스템에서 가장 핵심적인 기술이며 실내에 대한 환경 정보들, 예를 들면 온도나 습도, 조도와 같은 정보들

습득한다. 그림 3은 Zigbee 프로토콜 기반의 Crossbow Tmote 모듈이다.



그림 3. Zigbee 센서  
Fig 3. Zigbee Sensor

## III. 홈 네트워크 시스템 설계

### 1. 시스템 구성도

본 논문의 시스템은 기존의 홈 네트워크 시스템을 기반으로 한 안드로이드 홈 제어 구성 시스템을 4가지 부분으로 나뉘었다. 첫 번째는 각 방안의 정보를 수집 부분인데 각 방에 Zigbee 무선 센서와 카메라가 설치되어 있다. 둘째는 각 가전의 동작을 제어할 마이크로 컨트롤러 시스템이다. 세 번째는 이들 정보를 수집하여 데이터베이스에 저장하고 마이크로 컨트롤러를 제어하는 홈 게이트웨이 부분이며, 네 번째는 홈 게이트웨이에 설치된 무선 공유기나 외부의 인터넷 망을 통해 홈 서버에 접속하여 방에 대한 상태를 조회하고 각 방의 가전을 컨트롤 하는 스마트폰 응용 부분으로 나뉜다.<sup>[5,6]</sup> 각 방에 설치된 무선센서는 방의 온도, 습도, 조도 등의 정보를 측정하여 홈 서버에 설치된 무선센서 노드로 보내고 홈 서버는 각 방에 대한 정보를 데이터베이스로 저장한다. 안드로이드 홈 네트워크 응용은 인터넷 망을 이용하여 홈 서버에 접속해 데이터베이스에 저장된 DB를 읽어오고 각 방의 정보를 스마트폰을 통해 조회 한다. 또한 스마트폰은 각 방에 설치된 각 가전의 동작 상태를 조회하고 원격으로 동작을 제어할 수 있다. 스마트폰에서 제어 버튼을 누르면 홈 서버는 동작 상태에 대한 데이터를 업데이트 시키고 마이크로컨트롤러에 신호를 보내 각 가전의 동작을 제어한다. 각 방에 설치된 카메라는 홈 서버를 통해 영상 정보를 전달하고 스마트폰을 통해 TCP/IP로 영상 정보를

수신하여 각 방의 영상을 원격으로 확인할 수 있다. 그림 4는 본 논문에서 제안한 홈 네트워크 시스템 구성도를 보여주고 있다.

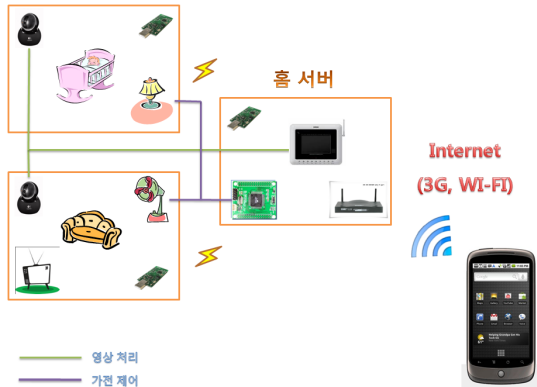


그림 4. 홈 네트워크 시스템 구성도  
Fig 4. Home Network System Architecture

## 2. 홈 네트워크 시스템 흐름도

그림 5은 본 시스템에서 제안한 안드로이드 홈 네트워크 응용에서 각각의 기능에 대한 시스템의 동작 흐름도를 보여주고 있다. 그림 5에서 보여주는 바와 같이 3가지 부분 동작으로 구분되며 홈 서버에서 Zigbee 센서의 정보를 얻어와 스마트폰에 표시하는 과정과 마이크로 컨트롤러를 통해 가전을 제어하는 부분, 홈 서버에서 각 방의 카메라 영상을 받아 스마트폰에 전송하는 과정으로 나누어진다.

각 방에 설치된 Zigbee 센서는 방 안의 온도, 습도, 조도 등의 상태 정보를 측정하여 홈 서버에 설치된 베이스 노드에 보내고 각 방에 대한 정보를 홈 서버의 데이터베이스에 저장한다. 스마트폰은 WIFI나 3G 인터넷 망을 이용하여 홈 서버에 접속하고 홈 서버는 데이터베이스에 저장된 정보를 읽어온 후 상태 정보를 스마트폰에 전송 후 표시한다. 표시되는 정보는 각 방의 상태 정보와 가전 동작 상태에 대한 정보를 화면에 표시한다. 가전 제어의 경우 스마트폰에서 ON/OFF 버튼을 누르면 홈 서버는 각 가전의 상태 정보를 확인한 후 업데이트하고 마이크로 컨트롤러에 제어 신호를 보내 각 가전을 ON/OFF 시킨다. 영상 처리의 경우 각 방에 설치된 카메라를 통해 홈 서버에 화상 이미지 데이터를 실시간으로 전송한다. 스마트폰은 홈 서버에서 각 방의 화상 이미지를 전송 받아 화면에 표시한다.

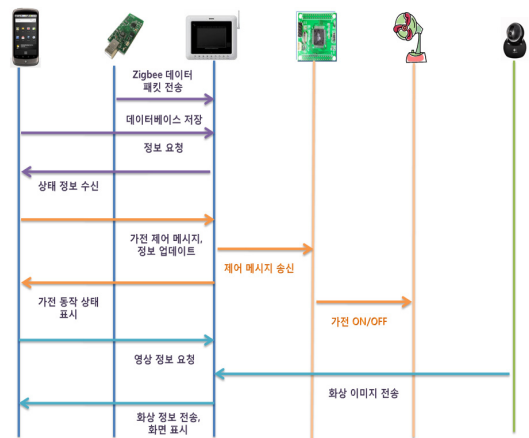


그림 5. 홈 네트워크 시스템 동작 흐름도  
Fig 5. Home Network System Process chart

## 3. 센서 데이터 수신 및 처리

각각의 노드는 OscilloscopeRF라는 TinyOS 응용을 포함하고 있고 홈 서버와 통신하는 베이스 노드에는 TOSBase를 포함 한다. 각 노드는 베이스노드에게 자기 자신에 대한 모듈 정보와 ID, 노드 번호, 각 채널의 데이터 값을 베이스 노드로 송신한다. 베이스 노드에는 각 노드들로부터 들어오는 데이터를 수신하고 들어온 데이터를 판별하여 모트 정보와 각 정보의 채널, 데이터를 판별한다.<sup>[7]</sup> 표 1은 베이스 노드로부터 들어오는 각 노드의 전체 데이터의 크기와 각각의 바이트가 의미하는 값을 설명한다. 노드의 데이터 총 크기는 44byte고 각각 2byte의 크기를 가진다.

표 1. Zigbee 메시지 형식  
Table 1. Zigbee Message Form

Byte order	Field length	Field name	Sample data
0,1	2 byte	Dest addr	7E 42
2	1 byte	Msg len	1A
10	1 byte	Handler ID	0A
11	1 byte	Group ID	7D
13, 14	2 byte	Mote ID	02 00
15, 16	2 byte	Counter	D2 00
17, 18	2 byte	Channel	01 00
19, 20	2 byte > 10	Data	3F 1B
21, 22			41 1B
23, 24			40 1B
25, 26			41 1B
27, 28			40 1B
29, 30			41 1B
31, 32			43 1B
33, 34			42 1B

- Dest addr : 목적지 주소를 가지고 SYNC\_BYTE 를 포함. 7E로 데이터 시작 판별
- Msg Len : 메시지의 총 길이
- Handler ID : 메시지 핸들러 ID
- Group ID : 무선으로 통신을 할 때 동종의 기기와 데이터를 구분하기 위해 그룹별로 바인딩 하여 통신 할 수 있도록 그룹을 지정하는 ID로 부여
- Mote ID : 송신한 노드의 ID
- Counter : 해당 노드가 보낸 데이터 양
- Channel : 온도, 조도, 습도 등의 각 채널
- Data : 실제 센서 노드가 수집한 데이터. 2byte씩 총 10개의 데이터로 구성

표 2 는 각 채널 별로 들어오는 데이터의 타입을 정의한다.

표 2. 각 채널 별 데이터 타입  
Table 2. Chanel Data Type

Channel Number	Data
0	습도
1	온도
2,3	조도
4	내부 온도

그림 6 은 센서로부터 들어온 데이터 패킷 내용을 표시하고 각 채널에서 추출한 온도, 습도, 조도 등의 데이터를 출력한 모습이다. 그림 7 과 같이 하나의 버퍼는 7E, 42, 1A, 01, 08의 값으로 시작한다. 들어오는 2개의 스트림 데이터를 buf\_1, buf\_2에 저장하고 각각의 버퍼값을 비교해 시작 값이 일치하지 않은 경우에는 들어오는 버퍼의 값을 버리고 다음 스트림 데이터를 가져온다. 2개의 버퍼 값이 일치 하는 경우 buf\_2값을 저장한다. Data에 해당하는 부분은 센서의 ID 값과 메시지 길이, 데이터 채널, 센싱 데이터 등을 포함한다.

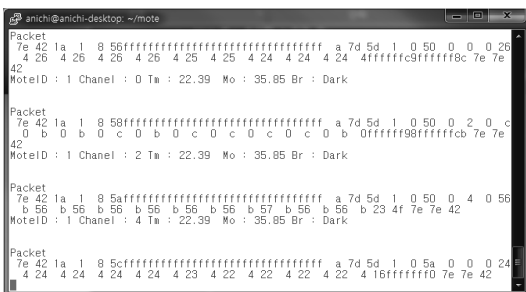


그림 6. 각 노드에서 수신된 데이터  
Fig 6. Received data from each node

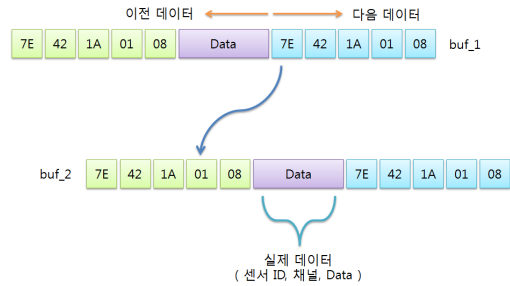


그림 7. 센서 데이터 처리  
Fig 7. Sensor data processing

센서로부터 들어오는 연속적인 데이터를 원하는 데이터로 변환하기 위해 표 2 의 각 채널 별로 해당되는 바이트 값의 데이터를 읽어 들여 온도, 습도, 조도의 값으로 변환한다. 해당되는 데이터는 표 1 의 19번째와 20번째 데이터를 표본 데이터로 설정하고 아래의 계산식에 의해 온도, 습도, 조도의 값으로 계산된다.

- 표본 데이터 :  $\text{byte}[20] * 256 + \text{byte}[19]$
- 습도 계산 :  $\text{humidity} = -4 + (0.0405 * \text{avrdata}) + (-2.8 * 1.0E-6) * (\text{avrdata} * \text{avrdata})$
- 온도 계산 :  $\text{tempature} = 0.00355 * \text{avrdata}$
- 조도 계산 :  $\text{solor} = (\text{avrdata} * 1500) / 4096$

온도, 습도의 경우에는 위 계산식에서 바로추출 되며 조도의 경우 solor 값이 30 이하인 경우 “어두움”, 30에서 75 사이 이면 “밝음”, 75 이상이면 “매우 밝음”으로 표시한다. 각각의 값은 데이터베이스에 저장되는데 먼저 센서의 MoteID를 인식 한 뒤 각 센서의 번호에 해당되는 데이터베이스에 값이 저장된다.

홈 서버의 데이터베이스 테이블은 총 3개로 구성되어 있다. 첫 번째는 각 방에 상태 정보를 저장하는 테이블이고, 두 번째는 각 가전의 작동 상태를 저장 하는 테이블이고 세 번째는 각 방에 대한 적정 환경 설정 테이블이 있다. 표 3 의 테이블은 각 방에 대한 상태에 대한 테이블이다. 각 방 마다 하나의 테이블을 가지고 있으며 센서에서 값이 들어올 때 센서 번호에 따라 방 테이블에 온도, 습도, 조도 값이 연속적으로 저장된다. 안드로이드 단말기에 표시되는 값은 각 필드에 해당되는 전체 값의 평균을 계산하여 표시 한다.

표 3. 온도, 습도, 조도 테이블

Table 3. Temperature, Humidity, Bright Table

num	온도	습도	조도
1	20.55	30.33	Bright

표 4 와 표 5 의 테이블은 가전 상태에 대한 테이블인 데 각 방에 설치된 가전이 현재 어떤 상태인가를 확인 하는 상태 데이터를 저장한다. 가전 상태 테이블은 조명 작동 상태를 확인하는 테이블과 히터, 에어컨과 같은 가전 작동 상태를 확인하는 테이블로 구성 되어 있다.

표 4. 조명 상태 테이블

Table 4. Bright Status Table

방 번호	중앙등	측면등	미등
1	On	Off	Off
2	Off	On	On
3	Off	Off	Off

표 5. 가전 상태 테이블

Table 4. Appliance Status Table

방 번호	히터	에어콘	가습기
1	On	Off	Off
2	Off	On	On
3	Off	Off	Off

표 6 의 테이블은 사용자가 원하는 온도나 습도로 설정하면 그 환경 설정에 맞추기 위해 홈 서버가 가전들을 자동으로 제어하기 위한 데이터이다. 응용에서 적정 온도를 알맞게 설정하면 홈 서버는 자동적으로 사용자가 설정한 최적 환경 값을 맞추기 위해 가전들을 자동 제어 하도록 한다.

표 6. 적정 온도 테이블

Table 6. Optimum Temperature

방 번호	온도	습도
1	23	30
2	25	36
3	18	22

#### 4. 영상 처리

영상 데이터 처리 부분은 카메라의 영상을 스마트폰

으로 전달하기 위해 TCP/IP socket 프로그램으로 구현 하였다. 홈 서버와 연결된 웹 카메라는 클라이언트의 요청 신호가 들어오면 접속 요청을 수락하고 카메라의 영상 데이터를 비트맵으로 변환하여 클라이언트에 전송 하게 된다. 클라이언트는 비트맵 정보를 연속적으로 수신하고 화면에 표시한다. 그림 8 은 스마트폰과 홈 서버 사이의 데이터 전송 흐름을 설명한다.

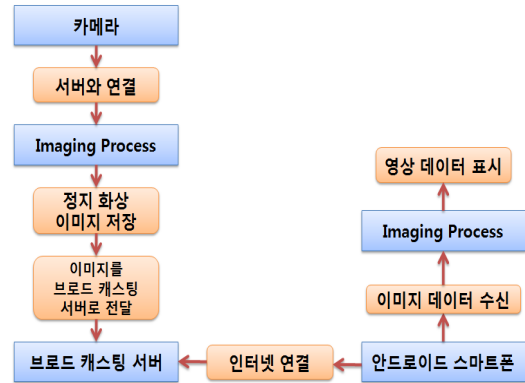


그림 8. 영상 데이터 전송 흐름

Fig 8. The flow of imaging processing transmission

카메라가 동작하는 동안 카메라에서 캡처한 정지 화상 이미지를 RAW나 JPEG 로 압축하여 버퍼에 저장한다. 서버 측은 일정시간 마다 카메라로부터 동영상을 연속적으로 받아 버퍼에 저장하며 브로드 캐스팅 서버는 동작하는 동안 압축된 이미지 버퍼를 읽고 서버에 접속된 클라이언트에게 이미지 버퍼를 전송하게 된다. 안드로이드 스마트폰은 3G 네트워크나 WI-FI를 통해 인터넷에 접속해 브로드 캐스트 서버로 연결하고 서버가 클라이언트의 요청을 수락 하면 서버로부터 압축된 화상 이미지를 수신하게 된다. 안드로이드 스마트폰의 영상 처리는 서버로부터 받은 화상 이미지를 안드로이드 스마트폰의 화면에 표시하게 된다.<sup>[8,9]</sup>

## IV. 구현 및 테스트

### 1. 구현 환경

안드로이드 응용 개발은 Android SDK 1.6에서 진행 하였으며, 개발 도구는 Eclipse의 Android ADT Plugin

을 사용하여 개발 하였다. 영상 처리 부분을 구현하기 위해 안드로이드 NDK(Native Development Kit)의 OpenCV 라이브러리를 사용하였고 센서 값을 저장하기 위한 데이터베이스는 오픈소스 기반의 CUBRID 2.0 버전을 사용하였다. 테스트 장비로 센서는 Moteiv사의 Tmote Sky 모듈을 사용하였고 마이크로 컨트롤러 보드는 ATMEGA128 칩이 내장된 청과 이엠티 MP-100 보드를 사용하였다. 카메라는 Logitech의 QuickCam Pro 4000을 사용하였다. 서버 환경은 Ubuntu 9.04 Linux를 기반으로 작성하였다.

## 2. 시나리오 테스트

본 논문에서 제안한 시나리오는 먼저 각 방에 여러 개 무선 센서를 설치하면 각각의 노드들은 RF 통신을 통해 노드와 통신하고 룸에서 수집한 데이터 패킷을 PC에 설치된 베이스노드에 송신한다. 베이스 노드는 각각의 데이터 패킷을 파싱하여 각 노드의 번호와 데이터 채널을 판별하고 해당되는 데이터의 바이트 값을 각각의 온도, 습도, 조도 변환 식에 맞게 계산 한다. 각 노드별로 수신된 온도, 습도, 조도는 노드들에 해당되는 데이터베이스에 저장된다. 스마트폰에서 각 룸에 대한 정보 조회요청이 들어오면 홈 서버는 데이터베이스에 저장된 각 룸에 대한 정보를 조회한 뒤 메시지를 안드로이드 단말기로 전송한다. 마찬가지로 단말기에서 가전 상태를 조회하고자 할 때는 홈 서버에 접속해 데이터베이스 값을 홈 서버로부터 수신 받고 응용 화면에 표시된다. 단말기에서 가전을 제어하고자 할 때 응용에서 가전제어 신호를 홈 서버에 전송 한 뒤 데이터베이스의 가전 상태 정보를 업데이트하고 마이크로 컨트롤러에 제어 메시지를 송신한다. 마이크로 컨트롤러는 홈 서버로부터 들어오는 제어 메시지를 판별한 뒤 각 메시지 별로 어떤 가전을 제어 할 것 인지를 결정하고 해당 가전 제어 정보를 수신한 뒤 화면에 디스플레이 한다. 각 방의 설치된 카메라의 영상을 확인하고자 할 때 홈 서버는 각 방에서 들어오는 영상 정보를 스마트폰에 전송하고 실시간으로 카메라의 영상이 화면에 표시된다.

사용자는 스마트폰을 통해 각 방에 대한 실내 환경과 가전 동작 상태를 확인 한다. 그림 9에서 보여주는 바와 같이 먼저 메인 메뉴에서 Room Status를 클릭하면 전체 방 리스트가 출력이 되고 사용자는 전체 방 리스트에서 원하는 방을 선택한 후 리스트 밑에 실내 상태 버튼을 클

릭한다. 가전 상태 버튼을 누르면 가전 상태에 대한 정보를 표시한다.



그림 9. 실내 환경, 가전 동작 상태  
Fig 9. Room status, Appliance operation

사용자는 실내 상태를 항상 최적으로 유지하기 위해 실내 적정 온도와 습도를 설정할 수 있다. 사용자는 방 리스트에서 적정 환경을 설정할 방을 선택한 뒤 적정 환경 버튼을 클릭하거나 상태 확인 화면에서 메뉴를 클릭하여 적정 환경 설정을 클릭한다. 적정 온도와 습도는 +, - 버튼을 이용해 설정 할 수 있으며 설정이 끝난 뒤 확인을 누르면 메시지 창이 출력된다. 그림 10 은 서재의 적정 온도를 23℃로, 적정 습도를 50%로 설정한 내용을 메시지로 출력한 화면이다.



그림 10. 적정 온도 설정  
Fig 10. Setting temperature

그림 11과 그림 12는 메인 메뉴에서 Light나 Appliance 버튼을 클릭하면 조명 및 가전 제어 창으로 바뀐다. 먼저 가전을 제어할 방의 리스트 박스를 클릭하고 제어를 원하는 방을 선택한다. 화면을 보면 각 가전의 On / Off 버튼이 있는데 사용자는 버튼을 눌러 조명 및 가전을 제어한다. 통합 제어는 모든 가전을 On / Off 시킨다.



그림 11. 조명 제어  
Fig 11. Lighting control



그림 12. 가전 제어  
Fig 12. Appliance control

실제 가전 제어 테스트는 그림 13과 같이 테스트를 하였다. 안드로이드 단말기를 통해 제어 신호를 전달하게 되면 마이크로 컨트롤러의 연결된 가전이 제어 신호와 동시에 ON/OFF 되는지를 테스트 한다.



그림 13. 가전 제어 테스트  
Fig 13. Appliance control test

메인 메뉴에서 카메라 버튼을 클릭하면 카메라가 설치된 방이 나온다. 각 방의 탭을 누르면 각 방에 설치된

카메라를 통해 각 방의 영상을 홈 서버로부터 받아 화면에 출력하게 된다. 그림 14는 각 방의 카메라 영상을 스마트폰을 통해 보여주고 있다.

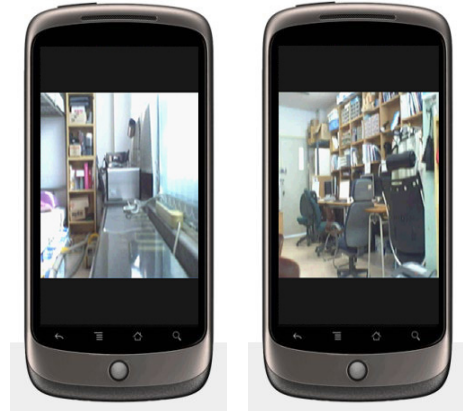


그림 14. 각 방의 카메라 영상  
Fig 14. Video of each room

## V. 결론

본 논문에서는 안드로이드가 탑재된 모바일 단말기를 통해 각 방의 상태를 확인하고 가전을 원격 제어 할 수 있는 응용을 설계 및 구현 하였다. 각 방에 설치된 센서를 통해 방안의 상태를 스마트폰으로 확인하고 조명 및 가전을 원격으로 제어하였다. 또한 방안에 설치된 카메라를 통해 각 방에 대한 영상화면 출력을 구현 하였다.

향후 연구 방향으로는 단순히 상태를 모니터링하고 가전을 제어하는 기능을 넘어택내의 이상 징후를 감지할 수 있는 방안에 대한 연구가 이루어 질 것이다.택내에 도둑이 들거나 화재 등의 이상 징후를 홈 서버에서 감지해 스마트폰에서 표시하는 부분을 추가 할 것이며 돌발 상황에 대처하기 위해 스마트폰에서 사용자에게 알리고 대처할 수 있게 하는 방안에 대한 연구가 이루어 질 것이다. 또한, 향후 안드로이드 플랫폼이 모바일을 넘어 셋톱박스나 정보 가전에 안드로이드가 탑재가 되면 향후 안드로이드 플랫폼을 중심으로 한 홈 네트워크가 구축이 될 것으로 예상되며 앞으로 스마트 폰을 중심으로 한 홈 네트워크에서 어떠한 역할을 할 것인지에 대한 방안을 제시하도록 할 것이다.



## 참 고 문 헌

- [1] 박정규, 서승호, 김양남, 이궁해, “유비쿼터스 컴퓨팅 환경을 위한 모바일 서비스 모델”, 한국정보과학회, 2006
- [2] 한국소프트웨어 진흥원, “모바일 플랫폼 경쟁에서 S/W(애플리케이션/콘텐츠)경쟁으로 확대”, 2008
- [3] Paolo Meroni, Elena Pagani, Gian Paolo Rossi, “An Opportunistic Platform for Android-based MobileDevices”, MobiOpp '10, ACM, 2010
- [4] 장한식, 이상도, “Zigbee 기술을 이용한 디지털 홈 네트워킹”, 한국통신학회지(정보와통신), p. 50-62, 2005
- [5] 김동균, 송재훈, 이상정, “모바일 단말기의 정보제어가전을 위한 홈 게이트웨어 설계”, 한국정보처리학회 2003 추계학술발표대회 논문집 제 10권, 제 2호, p. 967-970, 2003
- [6] CHIH-LIN HU, WEI-SHUN LIAO\*, YEN-JU HUANG, “Mobile Media Content Sharing in UPnP-Based Home Network Environment“, JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 24, 2008
- [7] 나선웅, 오홍록, 이해각, 이상정, “무선 센서 네트워크를 이용한 홈 네트워크 개인화 서비스”, 한국정보과학회 한국컴퓨터종합학술대회 논문집(D) 제 33권 1호, p.244-246, 2006
- [8] Jonathan J. Hull, Xu Liu, Berna Erol, Jamey Graham, Jorge Moraleda, “Mobile Image Recognition: Architectures and Tradeoffs”, HotMobile 10, ACM, 2010
- [9] 최진엽, 이상정, 전병찬, “안드로이드를 이용한 홈 네트워크 영상처리 설계 구현”, 한국인터넷방송통신학회 2010 춘계 학술발표대회 논문집 제8권 1호, p254-257, 2010

## 저자 소개

### 최진엽(준회원)



- 2010년 순천향대학교 컴퓨터학과 학사 졸업
- 2010년 ~ 현재 순천향대학교 컴퓨터학과 (석사 과정)
- <관심분야 : 안드로이드, 임베디드 시스템, 스마트폰 프로그래밍 >

### 이상정(정회원)



- 1983년 한양대학교 전자공학사
- 1985년 한양대학교 전자공학석사
- 1988년 한양대학교 전자공학박사
- 1988년-현재 순천향대학교 컴퓨터학부 교수
- 1999년~2000년 미국 University of Minnesota 방문교수
- <관심분야 : 네트워크 응용, 컴퓨터 구조 >

### 전병찬(정회원)



- 순천향대학교 대학원 전산학과 박사
- 현 정운대학교 방송영상학과 교수
- <관심분야 : 컴퓨터구조, 홈 네트워크, 모바일, 마이크로 프로세서 등 >