

스마트 홈서비스를 위한 UPnP기반 홈 게이트웨이와 지능형 가전기기 간 연동 시스템

한왕원* , 김영만**

*국민대학교 전자정보통신대학 컴퓨터공학부

Convergence System Between Home Gateway and Intelligent Home Appliance for Smart Home Service

Wang-Won Han* , Young-Man Kim

Communication Lab, School of Computer Science, Kookmin University

E-mail : (wwhan, ymkim)@kookmin.ac.kr

요 약

최근 정보통신 기술의 발전으로 인하여 어느 장소에서든 컴퓨터 또는 무선장치를 사용하여 집안의 지능형 가전기기들을 점검하고 제어 할 수 있는 수준까지 이르렀다. 이와 같은 서비스를 제공하기 위한 요소기술 중에 마이크로소프트사에서 발표한 UPnP(Universal Plug & Play)는 TCP/IP 프로토콜을 기반으로 하여 각 지능형 가전기기에 IP주소를 할당하여 어느 곳에서나 누구나 편리하게 지능형 가전기기의 서비스를 이용할 수 있도록 해준다. 본 논문에서는 UPnP 미들웨어 기반의 홈 네트워크에서 웹 기반의 인터페이스를 통해 지능형 가전기기를 쉽게 관리할 수 있는 홈 게이트웨이와 지능형 가전기기 간 연동 시스템을 설계하고 구현한다.

1. 서 론

우리나라의 컴퓨터 보급대수가 늘어나고 초고속 인터넷이 대부분의 가정에 보급됨에 따라 가정내 정보화도 가속화되고 있다. 또한 한대 이상의 컴퓨터를 보유한 가구가 증가하고, 스캐너, 프린터 등의 컴퓨터 관련 장비뿐만이 아니라 인터넷이 가능한 냉장고와 같이 네트워크 접속이 가능한 가전기기들이 출현함에 따라 홈 네트워크를 구축할 수 있는 기반이 빠르게 조성되고 있다. 이에 따라 홈

오토메이션에 대한 욕구가 증가하고, 홈 네트워크 기술에 대한 관심이 증가되고 있으며, 홈 네트워크 기술 분야의 표준 경쟁 또한 치열해지고 있다.

지능형 가전기기들을 네트워크로 연결하고 관리하기 위해서는 연결기술(Ethernet, Bluetooth, 무선 랜, IEEE1394, PLC)과 함께 장치에 대한 섬세한 제어와 원활한 관리를 지원하는 미들웨어가 필요하다. 이러한 미들웨어 기술로는 UPnP[1][2], Jini[3], HAVi[4] 등이 있는데, 이 중 UPnP

는 기존 컴퓨터상에서 주변기기들을 관리하기 위해 사용되던 기술인 Plug and Play 기능을 홈 네트워크에 확대 적용한 기술로서, 컴퓨터나 홈 게이트웨이와 같은 컨트롤 포인트의 역할을 수행할 수 있는 장치를 중심으로 가정 내 각종 지능형 가전기기를 제어한다.

본 논문에서는 UPnP 기술의 개념을 간단히 설명하고, 인텔사에서 UPnP 표준 명세에 따라 개발한 “Intel UPnP SDK for Linux” [5]를 기반으로 구현한 홈 네트워크 시스템을 설명한다. 마지막으로 홈 네트워크 서비스 중에서 홈 오토메이션 서비스를 UPnP 기반 홈 네트워크에서 제공하기 위해 센서 네트워크를 사용하여 가습기, 스탠드, 선풍기 등의 일반 가전기기를 연결하고, 센서 네트워크의 싱크노드를 UPnP 디바이스에 연결함으로써 UPnP 기반 홈 오토메이션 시스템을 구현한다.

2. UPnP 구조 및 프로토콜 소개

이번 절에서는 본 논문에서 설계하고, 구현하고자 하는 홈 오토메이션 시스템의 근간을 이루는 UPnP 기술에 대한 구조 및 프로토콜의 흐름에 대해 설명한다.

2.1 UPnP의 구성요소

UPnP를 이루는 구성요소는 크게 컨트롤 포인트, 디바이스, 서비스로 이루어져 있으며 각 구성요소의 역할은 아래와 같다.

UPnP 컨트롤 포인트는 다른 UPnP 디바이스를 제어하는 역할을 담당한다. UPnP 컨트롤 포인트는 자신이 제어할 수 있는 UPnP 디바이스가 네트워크에 연결되어 있는지를 검색하고, 검색된 UPnP 디바이스의 명세서를 참조하여 수행 가능한 서비스와 제어 방법을 파악한다.

UPnP 디바이스는 사용자에게 제공하기 위한 서비스와 부속 장치를 포함하고 있다. UPnP 디바이스는 제조사, 제품번호와 같은 디바이스 정보와 디바이스가 수행 가능한 서비스 목록을 각각 디바

이스 명세서와 서비스 명세서에 XML 형태로 저장하고 있으며, UPnP 컨트롤 포인트가 요구하면 해당 명세서를 제공한다.

서비스는 UPnP의 최소 제어 단위이며 상태 테이블과 제어 서버 및 이벤트 서버로 구성된다. 상태 테이블은 상태변수 값의 저장, 갱신등과 같은 관리 작업을 수행하고, 제어 서버는 UPnP 컨트롤 포인트의 제어 요청시에 해당 제어요청을 처리한다. 마지막으로 이벤트 서버는 상태 테이블이 갱신될 때마다 이를 등록된 컨트롤 포인트에게 이벤트로써 알리는 역할을 담당한다.

2.2 UPnP 네트워크 프로토콜과 동작 단계

UPnP 프로토콜은 TCP/IP, UDP, HTTP과 같은 기존의 프로토콜들을 기반으로 하여 동작하며, UPnP 디바이스 간에 교환하는 데이터는 HTTP 기반의 XML을 사용하여 통신한다. UPnP를 구성하는 핵심 프로토콜로써 네트워크상의 UPnP 디바이스와 서비스를 찾기 위한 SSDP(Simple Service Discovery Protocol), 이벤트 메시지의 송수신시에 사용하는 GENA(General Event Notification Architecture), UPnP 컨트롤 포인트가 UPnP 디바이스에 제어 명령을 보내기 위해 사용하는 SOAP(Simple Object Access Protocol)가 있으며, 이러한 UPnP 프로토콜을 사용한 컨트롤 포인트와 디바이스 간의 통신은 발견, 기술, 제어, 이벤팅, 프리젠테이션의 5단계로 나누어진다

3. UPnP 홈 오토메이션 시스템 설계 및 구현

이번 절에서는 앞 절에서 간략하게 설명한 UPnP 미들웨어를 사용하여 웹 기반으로 집안의 지능형 가전기기들을 제어하고, 점검할 수 있는 홈 오토메이션 시스템을 설계하고, 가전기기들이 연결된 센서 네트워크를 홈 오토메이션 시스템의 UPnP 디바이스에 연결함으로써 홈 오토메이션

시스템의 구현을 완성한다.

3.1 UPnP 홈 오토메이션 시스템 개요

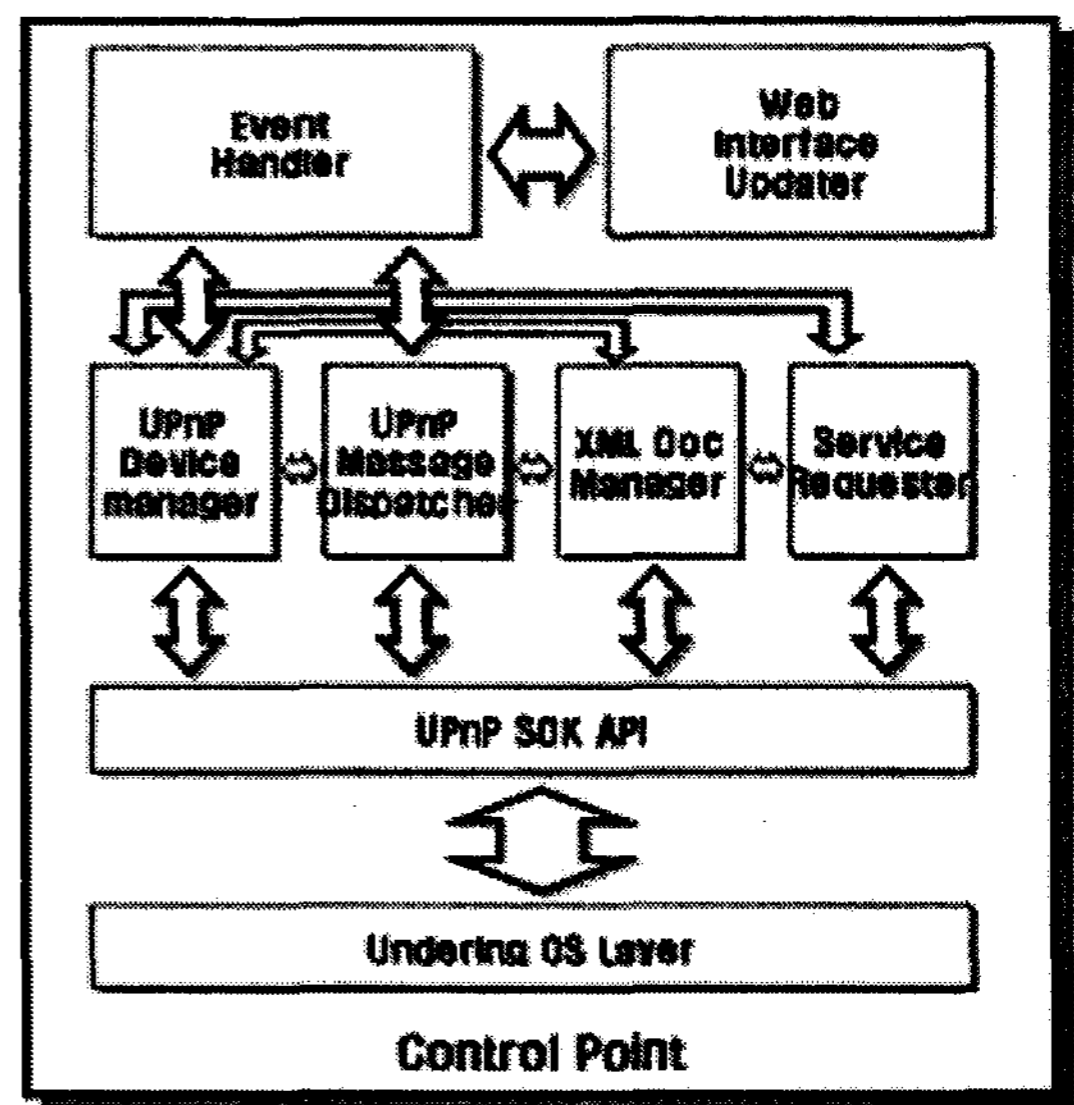
UPnP 기반의 홈 오토메이션 시스템을 구현하기 위해 본 논문에서는 Fedora core 4 리눅스 환경에서 동작하는 인텔사의 "UPnP SDK for Linux 1.0.4"을 사용하였다. UPnP SDK는 개발자들이 리눅스 환경에서 컨트롤 포인트와 UPnP 디바이스를 개발할 수 있도록 프로토콜과 XML DOM 라이브러리, 그리고 기타 웹 관련 API를 제공한다.

본 논문에서는 UPnP SDK를 사용하여 홈 게이트웨이와 UPnP 디바이스를 위한 소프트웨어를 개발하였으며, UPnP 디바이스의 내장형 장치로써 센서 네트워크를 구축하여 검증하였다. 센서 네트워크를 구성하는 센서 노드는 Crossbow사의 Mica2[6] 모드를 사용하여 조도, 습도 값을 측정하여 홈 게이트웨이(컨트롤 포인트)에 보고하도록 선정하였다. 홈 게이트웨이는 UPnP 디바이스가 보고한 조도, 습도 값을 사용하여 센서 네트워크에 연결된 가전기기의 전원을 제어한다. 즉 측정된 조도값이 사용자가 이전에 설정한 특정 기준치 이하가 된다면 스탠드의 전원을 On하고, 기준치 이상이 된다면 스탠드의 전원을 자동으로 Off해준다. 또한 수동으로 가전기기의 전원을 제어하거나 가전기기의 상태를 점검하기 위해 홈 오토메이션 시스템은 웹 인터페이스를 제공한다.

3.2 홈 게이트웨이(컨트롤 포인트) 응용 프로그램 구조

홈 오토메이션 시스템의 홈 게이트웨이는 [그림 1]과 같이 XML 문서 관리자, UPnP 메시지 해석기, UPnP 디바이스 관리자, 웹 인터페이스 업데이터, 서비스 요청기, 이벤트 핸들러로 이루어진다. 홈 게이트웨이 응용 프로그램을 구성하는 모듈들은 모두 UPnP SDK API를 사용하여 구현되었다. XML 문서 관리자는 기술(Description)단계에서 사용되는 모듈로써 UPnP 디바이스로부터

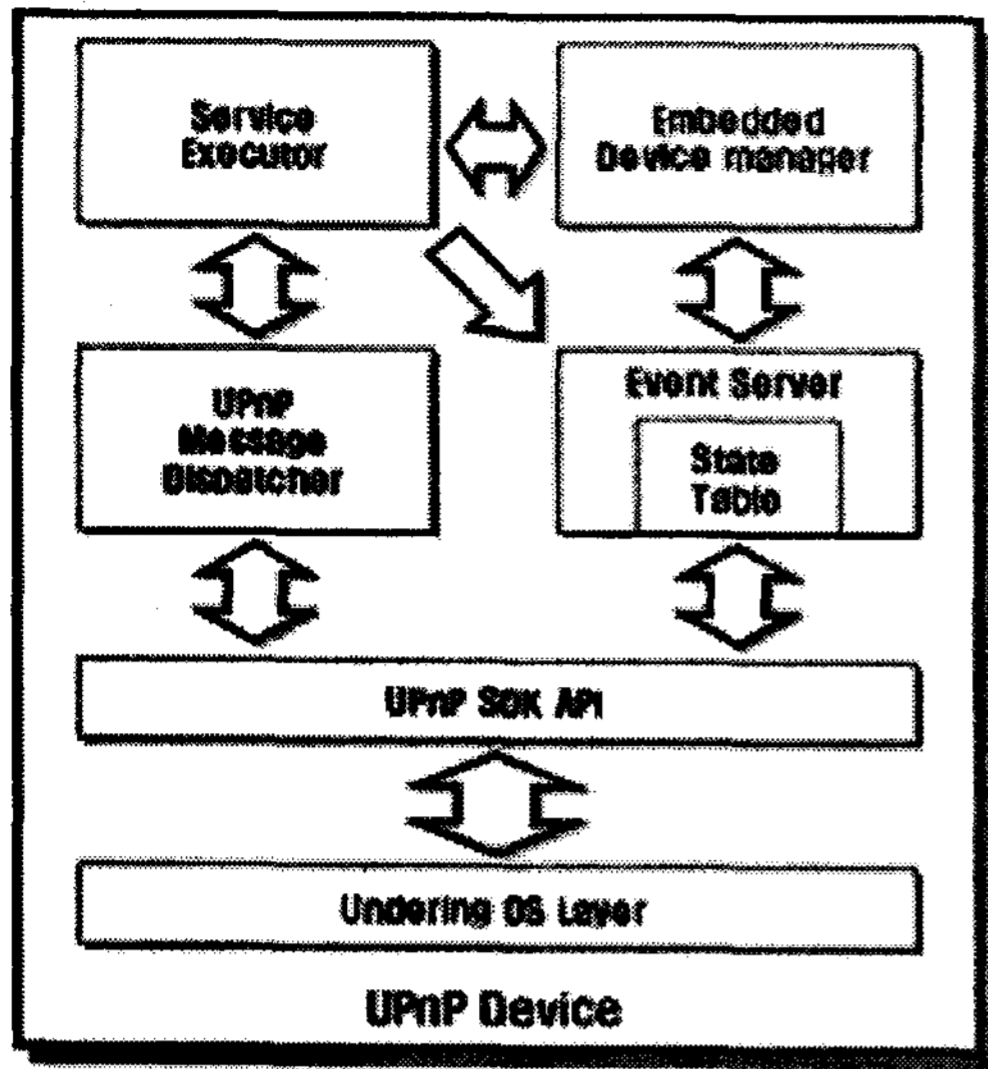
얻어온 명세서에 포함된 서비스 목록 및 상태 변수 정보등 UPnP 디바이스와 관련된 중요 정보들을 관리하고, 다른 모듈들이 이러한 정보를 요구하면 제공하는 역할을 담당한다. UPnP 메시지 해석기는 UPnP 디바이스가 보내온 UPnP 메시지를 해석하고, 해석한 메시지의 내용에 따라 해당되는 모듈을 호출해주는 역할을 담당한다. UPnP 디바이스 관리자는 발견(Discovery) 단계에서 새롭게 발견된 UPnP 디바이스들을 테이블로 관리하고, UPnP 디바이스 제어와 관련된 간단한 정보들을 저장하고 있다. 웹 인터페이스 업데이터는 UPnP 디바이스가 UPnP 네트워크에 새롭게 추가되거나 제거되었을때 홈 게이트웨이의 프리젠테이션 페이지를 갱신하는 역할을 담당한다. 그리고 서비스 요청기는 UPnP 디바이스를 제어하기 위한 명령을 생성하고 UPnP 디바이스에게 전달하는 모듈이다. 마지막으로 이벤트 핸들러는 서비스에 대한 응답이나 UPnP 디바이스의 상태변경과 같은 이벤트 발생시에 이벤트에 대한 적절한 처리를 해준다.



[그림 1] 홈 게이트웨이 응용 프로그램 구조

3.3 UPnP 디바이스 응용 프로그램 구조

PnP 디바이스의 응용 프로그램은 [그림2]에서



[그림 2] UPnP 디바이스 응용 프로그램 구조

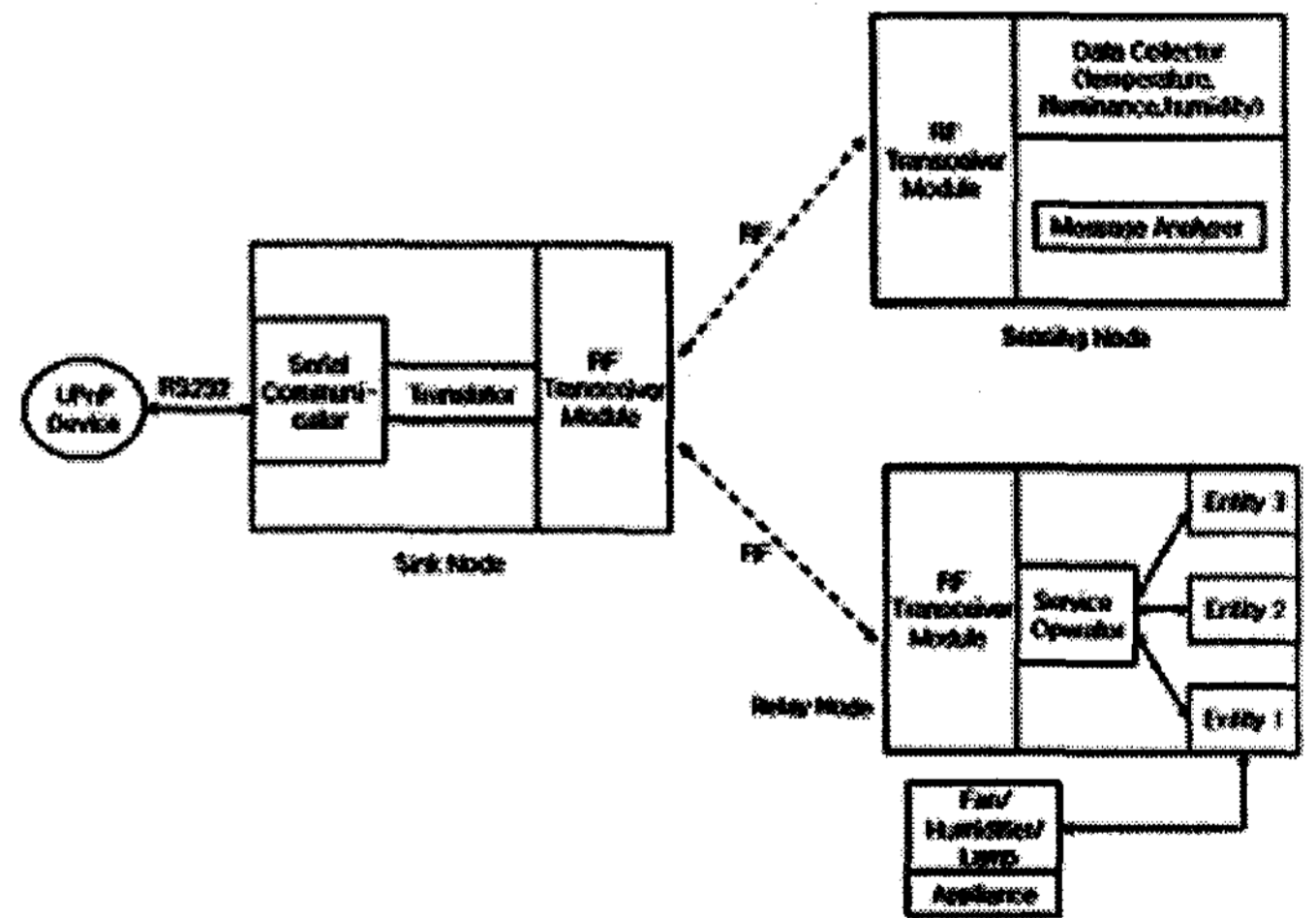
보이는 바와 같이 UPnP 메시지 해석기, 상태 변수 테이블, 이벤트 서버, 내장형 디바이스 관리자, 서비스 실행기로 이루어진다. UPnP 디바이스 응용 프로그램을 구성하는 모듈들은 홈 게이트웨이의 응용 프로그램을 구성하는 모듈들과 마찬가지로 UPnP SDK API를 사용하여 구현되었다. UPnP 메시지 해석기는 홈 게이트웨이와 UPnP 메시지를 주고 받기 위해 UPnP 메시지를 생성하거나 홈 게이트웨이로부터 받은 UPnP 메시지를 해석하여 적절한 모듈에 메시지를 전달하는 역할을 담당한다. 상태 변수 테이블은 UPnP 디바이스와 관련된 상태 변수들을 관리하는 테이블이며, 이벤트 서버는 이러한 상태 변수 테이블에 변화가 생기거나 홈 게이트웨이에게 보고해야만 하는 이벤트 발생 시에 이벤트 서버에 등록된 홈 게이트웨이들에게 생성된 이벤트를 전송한다. 내장형 디바이스 관리자는 UPnP 디바이스에 연결된 센서 네트워크를 관리하기 위한 모듈로써 센서 네트워크에 연결된 가전기기들을 제어하거나 센서 노드들에 의해 측정된 값들을 처리한다. 이러한 작업들을 수행하기 위해 내장형 디바이스 관리자는 센서 네트워크의 메시지를 해석할 수 있는 기능을 포함하고 있다. 마지막으로 서비스 실행기는 UPnP 메시지 해석

기로부터 서비스 요청을 전달받으면 UPnP 디바이스가 제공할 수 있는 서비스인지를 판단하여 자신이 제공할 수 있는 서비스라면 서비스를 담당하는 내장형 디바이스에게 메시지를 전달한다.

3.4 UPnP 디바이스와 센서 네트워크간 연동

앞에서 설명한 UPnP 디바이스 소프트웨어는 센서 네트워크를 제어할 수 있는 내장형 디바이스 관리자를 가지고 있으며, 컨트롤 포인트에 의한 센서 네트워크 제어 요청을 받으면, UPnP 디바이스는 시리얼 케이블로 연결된 센서 네트워크의 싱크 노드에게 제어 명령을 전달하여 센서 네트워크에서 해당 동작을 수행될 수 있도록 한다.

센서 네트워크는 UPnP 디바이스와의 연결을 위한 싱크 노드, 주변환경 정보를 수집하는 역할을 담당하는 센싱 노드, 가전기기와 연결되어 가전기기에 대한 제어를 담당하는 릴레이 노드로 이루어지며 각 구성요소의 소프트웨어 구조는 [그림3]과 같다.



[그림 3] 센서 네트워크 노드 소프트웨어 구조

3.4.1 싱크노드 소프트웨어 구조

센서 네트워크의 싱크노드는 UPnP 디바이스와 센서 네트워크를 연결해 주는 역할을 담당한다. 본 논문에서 사용한 싱크노드의 하드웨어는

Crossbow 사의 MIB510과 Mica2[6] 모트를 사용하였으며, 싱크 노드의 소프트웨어는 시리얼 통신 모듈, 메시지 변환기, RF 통신 모듈로 이루어진다. 시리얼 통신 모듈(Serial Communicator)은 UPnP 디바이스와 싱크 노드 사이의 메시지 교환을 담당하며, RF 통신 모듈(RF Transceiver Module)은 센싱 노드와 릴레이 노드가 송신한 메시지를 받거나 센싱 노드와 릴레이 노드에게 UPnP 디바이스의 메시지를 전달하는 역할을 담당한다. 마지막으로 메시지 전환기(Translator)는 시리얼 통신 모듈에서 전달 받은 메시지를 RF 통신 모듈의 형식으로 변환하거나 RF 통신 모듈의 메시지를 시리얼 통신 모듈이 이해할 수 있는 메시지 형식으로 변환하는 역할을 담당한다.

3.4.2 센싱 노드 소프트웨어 구조

센싱 노드는 온도, 조도, 습도와 같은 주변 환경 정보를 수치화하여 홈 오토메이션 시스템에 제공하는 역할을 담당한다. 수치화된 환경 정보들은 사용자가 설정한 기준값과 비교하여 가전기기들을 자동으로 On/Off 하는 기능을 위해 사용된다. 센싱 노드의 하드웨어는 Mica2 모트를 사용하였으며, 소프트웨어는 데이터 수집기(Data Collector), 메시지 분석기(Message Analyzer), RF 통신 모듈(RF Transceiver Module)로 이루어진다. 데이터 수집기는 주변 환경정보들을 수집하는 센서들을 관리하고, 센서들로부터 데이터를 전달받아 RF 통신 모듈에 전달하는 역할을 담당한다. 메시지 분석기는 싱크 노드로부터 센싱 노드를 제어하기 위해 전달 받은 메시지를 해석하고 적절한 처리를 수행한다. 마지막으로 RF 통신 모듈은 싱크 노드와 마찬가지로 센서 네트워크상에서 노드간의 RF 통신을 담당한다.

3.4.3 릴레이 노드 소프트웨어 구조

릴레이 노드는 가전기기에 연결되어 가전기기에 대한 제어(On/Off)를 담당한다. 릴레이 노드의 하

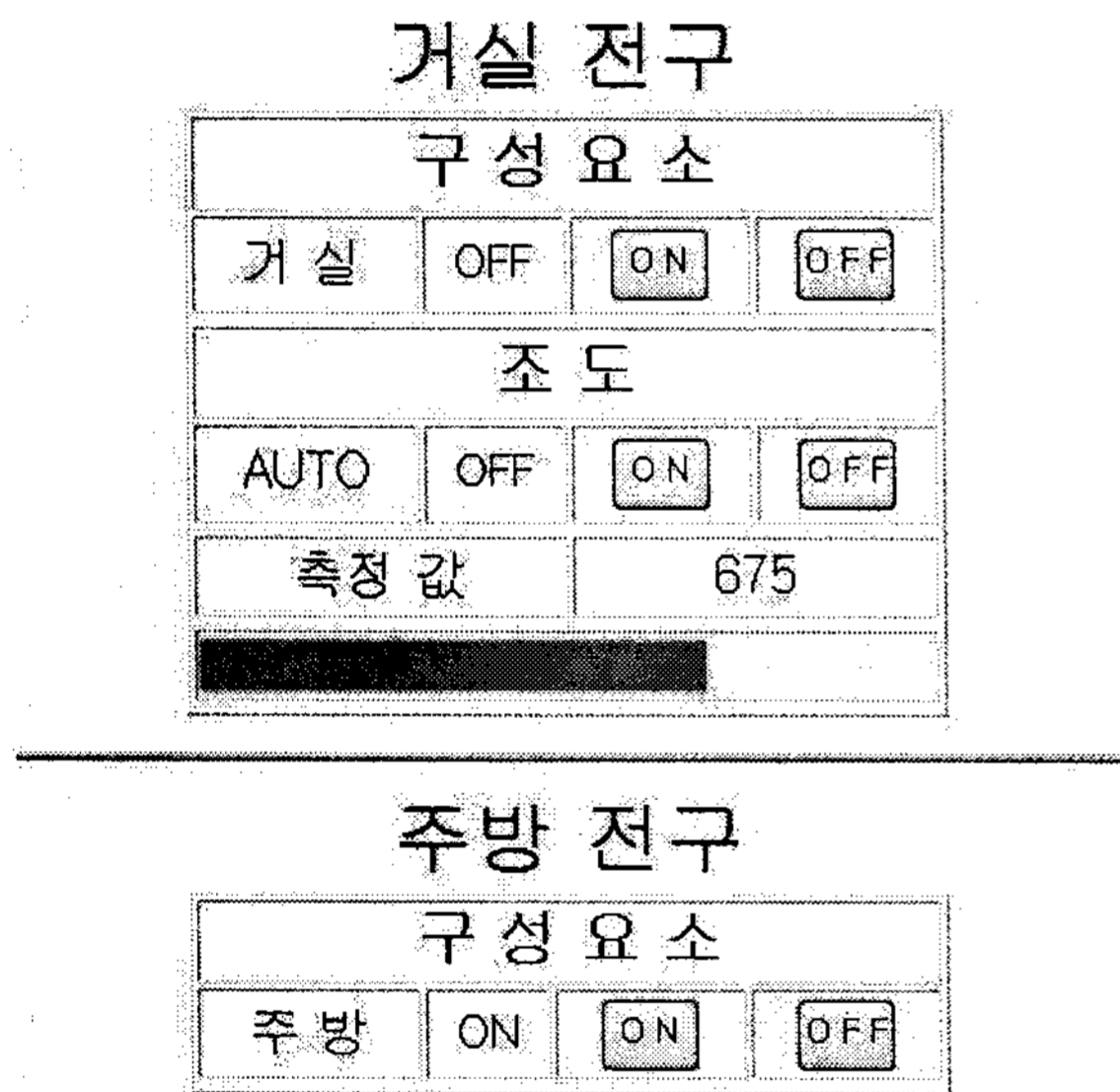
드웨어는 싱크 노드나 센싱 노드와 마찬가지로 Mica2 모트를 사용하였으며, 가전기기의 제어를 위해 트랜지스터와 릴레이로 이루어진 간단한 회로를 Mica2 모트에 추가하였다. 릴레이 노드의 주요 소프트웨어는 RF 통신 모듈(RF Transceiver Module), 서비스 실행기(Service Operator)로 이루어진다. RF 통신 모듈은 싱크 노드나 싱크 노드와 마찬가지로 RF 통신을 담당하며, 서비스 실행기는 싱크 노드로부터 전달받은 메시지를 해석하여 가전기기에 대한 적절한 서비스를 수행한다.

3.5 홈 오토메이션 시스템 수행과정

UPnP 네트워크상의 홈 게이트웨이가 SSDP 프로토콜을 사용하여 UPnP 디바이스를 발견하게 되면, UPnP 디바이스와 관련된 명세서를 UPnP 디바이스에게 요청한다. 이러한 요청에 대한 응답으로써 UPnP 디바이스는 XML 형식으로 구성된 명세서를 홈 게이트웨이에게 넘겨주게 된다. 홈 게이트웨이는 새로운 UPnP 디바이스와 관련된 명세서에 포함된 정보를 해석하여 얻은 정보를 UPnP 디바이스 매니저에 등록함으로써, 이후에 UPnP 디바이스를 제어하거나 상태를 점검할 수 있는 정보들(서비스 목록, 제어 URL 등)을 확보하게 된다. 또한 홈 게이트웨이의 웹 인터페이스 업데이트는 새로운 UPnP 디바이스의 프리젠테이션 URL을 홈 게이트웨이의 프리젠테이션 페이지에 추가한다.

사용자는 홈 게이트웨이의 프리젠테이션 페이지에 링크된 UPnP 디바이스의 프리젠테이션 페이지를 통해 특정 디바이스를 제어하거나 상태를 점검할 수 있다. 본 논문에서 설계하고 구현한 시스템에서는 UPnP 디바이스로서 내장형 장치의 역할을 담당하는 센서 네트워크를 구축하였으며, UPnP 디바이스의 프리젠테이션 페이지를 통해 센서 네트워크에 연결된 가전기기를 제어하거나 조도, 습도 센서의 값을 점검할 수 있다.

[그림4]는 센서 네트워크에 연결된 UPnP 디바이스의 프리젠테이션 페이지를 나타낸다. 현재 UPnP 디바이스에는 센서 네트워크를 사용하여 3개의 가전기기가 내장형 디바이스로써 연결되어 있으며, 이 가전기기를 제어하거나 현재 상태를 표시하기 위한 인터페이스를 제공한다. 거실에 있는 전구는 조도 센서와 연동되어 센서에서 감지된 측정값을 확인하여 자동 제어 할 수 있다. 자동 제어 기능은 사용자가 설정한 조도의 기준값에 따라 자동으로 거실 전구를 제어하는 자동화 시스템이다.



[그림 4] UPnP 디바이스 프리젠테이션 페이지

4. 결론

본 논문에서는 UPnP 포럼에서 정의한 표준에 맞는 기능들을 제공하는 인텔사의 "UPnP SDK for linux"의 API들을 사용하여 홈 네트워킹을 지원하는 UPnP 미들웨어를 기반으로 홈 네트워크 시스템을 구성하고 가전기기를 연결한 센서 네트워크를 UPnP 디바이스의 내장 장치로써 포함하여 구현하고 시험하였다.

본 논문에서 구성한 UPnP 미들웨어 기반 홈 네트워킹 시스템은 전문가의 도움 없이도 새로운

지능형 가전기기를 쉽게 설치할 수 있을 뿐만 아니라 웹 인터페이스를 통하여 사용자는 어디서나 집안의 가전기기를 제어하고 점검할 수 있는 장점을 가지고 있다. 따라서 UPnP 미들웨어는 앞으로 홈 네트워크 분야에서 커다란 비중을 차지할 것으로 예상된다.

UPnP는 현재 마이크로소프트사의 주도하에 많은 기업들이 표준으로 지지하고 있는 홈 네트워크 미들웨어 개념이지만 Jini, HAVI 등 다른 홈 네트워크 표준들과 호환되지 않고 있다. 홈 네트워크 표준간의 호환을 위해서는 OSGi[7]와 같은 미들웨어가 필요하다. 향후 연구로는 OSGi를 사용하여 UPnP, Jini, HAVI 홈 네트워크 표준들을 통합할 것이다.

[참고문헌]

- [1] UPnP Forum, "UPnP News Letter, Second Quarter", 2002, <http://www.upnp.org/news/default.asp>.
- [2] A.M. Brent, N. Toby, T. Charlie, and D.W. Mark, "Home networking with Universal Plug and Play", IEEE Communications Magazine, Vol.39, No.2, Dec. 2001, pp.104-109.
- [3] Sun microsystems, "Jini™ Architecture Specification", Oct. 2000.
- [4] Jussi Teirikangas, "HAVI: Home Audio Video Interoperability", 2001.
- [5] Intel, "UPnP SDK for Linux", <http://upnp.sourceforge.net/>
- [6] Crossbow, "Mica2 Datasheet", <http://www.xbow.com>.
- [7] OSGi Alliance, "About the OSGi Service Platform Technical Whitepaper", July. 2004.