

# UPnP와 LonWorks의 상호연동을 위한 브리지 구조

°배대호\* 박준호\* 강순주\* 최광호\*\*

\*경북대학교 전자공학과, \*\*전자 부품 연구원

{baedo°, zec}palgong.knu.ac.kr sjkang@ee.knu.ac.kr choikh@nuri.keti.re.kr

## Bridge Architecture for Interoperability between UPnP and LonWorks

°Dae-ho Bae\*, Jun-ho Park\*, Soon-ju Kang\*, Kwang-ho Choi\*\*

\*School of Electrical Engineering and Computer Science, Kyungpook National University

\*\*KETI(Korea Electronic Technical Institute)

### 요 약

다양한 서비스들이 필요한 홈 네트워크 환경은 하나의 프로토콜로 구현 되는 것은 부적합 하다. 따라서 본 논문에서는 실용적인 홈 네트워크를 구축하기 위해 UPnP와 LonWorks 네트워크를 상호 연결하는 브리지 구조를 제안한다. 제안하는 브리지 구조는 제어 네트워크에 적합한 LonWorks 네트워크를 가전기기를 제어하기 위해 사용하고 UPnP 네트워크를 미들웨어 프레임워크로 사용하여 홈 네트워크를 관리하는 구조를 가지고 있다.

### 1. 서 론

홈 네트워크[1]의 가전기기는 더욱 지능화 되고 있으며, 통신망 기술은 매우 빠르게 발전하고 있다. 홈 네트워크에서 가전기기를 연결하기 위해, IEEE1394와 LonWorks [2] 같은 다양한 프로토콜이 개발 되고 있으며, 이러한 프로토콜들을 지원하는 HAVi, CEBus, UPnP[3] 같은 미들웨어의 표준이 제시되고 있다.

이러한 표준들 중, 애설론사가 개발한 LonWorks (LonTalk 프로토콜 기반) 네트워크는, 홈 오토메이션 분야에 적합한 시장 표준으로 네트워크 장치들 사이의 컨트롤 데이터나 상태 데이터를 안전하게 전달할 수 있는 필드 버스 컨트롤 네트워크이다.

그러나 LonWorks는 컨트롤 네트워크에서 기술적인 우수함을 가지지만, 센서나 액츄에이터 등의 장치들 뿐만 아니라 PC, PDA, DTV등의 가정 내의 모든 장치를 네트워크에 포함시킬 수 있는 미들웨어 프레임워크에 대한 부분은 부족하다.

반면, 마이크로소프트에서 제안된 UPnP 네트워크는 홈네트워크 기기들의 연결과 다양한 하부 프로토콜의 지원을 제공하는 미들웨어 프레임워크를 제시하고 있다. 그러나, UPnP는 복잡한 프레임워크로 인해 작은 장치에 내장시키는 것은 어려우며, 장치들 간의 실시간 통신 및 동기화와 같은 필드 버스 컨트롤 네트워크의 특성은 만족시키지 못한다.

따라서, LonWorks의 프로토콜 특성을 이용하여 가정내의 작은 장치를 제어하기 위한 네트워크를 구성하고,

UPnP의 개방적이며 여러 프로토콜을 지원할 수 있는 구조를 이용하여 이러한 필드 장치들을 홈 네트워크에 포함시킨다면, 더욱 실용적인 홈네트워크를 구성할 수 있을 것이다.

본 논문에서는 홈 네트워크의 다양한 디바이스를 제어하고 관리하기 위해, 디바이스 연결을 위한 LonWorks 네트워크를 구성하고 이를 UPnP 네트워크를 통해 제어하는 UPnP와 LonWorks의 브리지 구조[4]를 제안한다.

### 2. 요구 분석

LonWorks 장치에서 LonTalk 프로토콜을 사용하기 위하여, 네트워크 통신 칩(뉴런 칩)[5,6]을 장치에 내장하고 있다. 각 LonWorks 장치들은 다른 장치와 통신을 하기 위해 표준화된 네트워크 변수들을 가지고 있으며 각 변수들은 바인딩(binding) 방식을 통해 가상의 통신 채널을 설정한다. 또한 다양한 어플리케이션 프로그램을 위해 LonWorks는 호스트 어플리케이션 구조를 제시한다. 이는 어플리케이션 프로그램이 뉴런 칩이 아닌 호스트 프로세서에서 동작할 수 있는 방식으로써 UPnP 프레임워크의 연동을 위해서 꼭 필요한 구조이다.

UPnP는 IP, TCP, UDP, HTTP, XML 등과 같은 인터넷 프로토콜을 기반으로 하고 있으며, SOAP[7]을 이용하여 각 장치들의 컨트롤 명령을 주고 받으며, GENA[8]를 이용하여 각 장치들의 상태정보를 전달 하게 된다. UPnP 장치들은 장치를 제어하는 컨트롤 포인트와 제어 대상인 장치로 나누어 진다.

따라서, LonWorks - UPnP 브리지 구조에서 LonWorks 장치들은 UPnP 컨트롤 포인트 및 장치들과 통신하기 위해 가상의 네트워크 변수를 제공하는 호스트 어플리케이션이 필요하며, SOAP을 통하여 들어온 메시지를 LonTalk에서 사용하는 메시지로 변경하는 부분과, LonTalk에서 들어온 메시지를 UPnP에서 사용하기 위해 GENA로 변경하는 부분이 필요하다. 마지막으로 UPnP와 LonWorks 네트워크 데이터의 일관성을 유지하기 위하여 UPnP - LonWorks 브리지는 컨트롤 요청에 대한 트랜잭션 처리구조를 제공하여야 한다.

### 3. 시스템 설계 및 구조

이러한 요구 사항을 고려한 UPnP - LonWorks 브리지는 그림 1과 같이 구성된다.

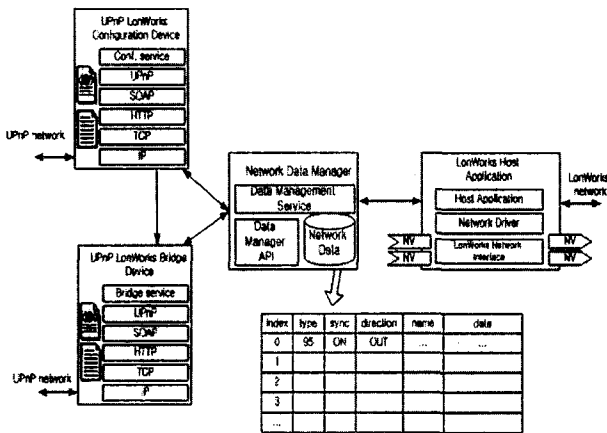


그림 1 UPnP - LonWorks 브리지 구성

UPnP - LonWorks 브리지는 LonWorks 호스트 어플리케이션, 네트워크 데이터 관리자, UPnP 설정 장치, UPnP 브리지 장치로 구성된다.

LonWorks 호스트 어플리케이션은 LonWorks 디바이스로서의 기능을 수행하며, 다른 LonWorks 장치들과 통신하기 위한 가상 네트워크 변수를 가지고 있다. LonWorks 호스트 어플리케이션에서 관리하는 가상 네트워크 변수들의 상태정보는 UPnP 네트워크와 연동을 위해 네트워크 데이터 관리자에게 전달된다.

네트워크 데이터 관리자는 LonWorks 네트워크와 UPnP 네트워크와의 연결을 담당하는 실질적인 객체이다. 네트워크 데이터 관리자는 LonWorks 네트워크 장치의 데이터 구조를 따르고 있다. 이는 LonWorks 데이터 구조가 UPnP 데이터 구조를 포함할 수 있으며 실질적인 장치 관리를 위한 구조를 가지고 있기 때문이다. 이 데이터는 LonWorks 호스트 어플리케이션 뿐만 아니라 UPnP 설정 장치 및 UPnP 브리지 장치에서 이용된다. 또한 네트

워크 데이터 관리자는 네트워크 데이터의 변경을 UPnP - LonWorks 브리지로 전달되는 컨트롤 요청 단위로 제한하여 UPnP와 LonWorks 네트워크 데이터에 대한 일관성을 유지한다.

UPnP 설정 장치는 UPnP 컨트롤 포인트에서 LonWorks 장치를 컨트롤 하거나 모니터링 하기 위해서 필요한 가상 네트워크 변수를 설정할 수 있게 한다.

UPnP 브리지 장치는 GENA와 SOAP을 이용한 실질적인 명령을 주고 받는 역할을 한다. 즉, UPnP 설정 장치가 설정한 데이터를 기반으로 UPnP 네트워크에서 통신을 위한 장치 기술 문서, 서비스 기술 문서, 프리젠테이션 문서등을 생성하며, 이를 바탕으로 LonWorks 네트워크의 컨트롤 명령을 GENA로 변환하여 UPnP 네트워크에 전달하고, UPnP 네트워크의 SOAP 메시지를 LonWorks 네트워크 프로토콜로 변환하는 역할을 한다.

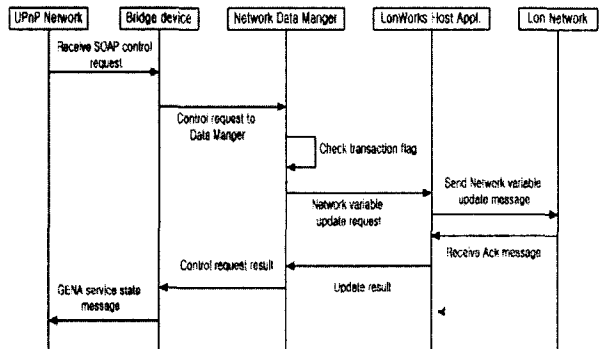


그림 2 UPnP 컨트롤 요청시 흐름도

그림 2는 앞서 설명한 각 컴포넌트들의 흐름도를 보여준다. UPnP 브리지 장치가 UPnP 컨트롤 포인트로부터 SOAP 형태의 컨트롤 요청을 받으면, SOAP 메시지를 네트워크 데이터 관리자가 사용하는 형태로 변환한 뒤 네트워크 데이터 관리자로 전달한다. 그리고 네트워크 데이터 관리로부터 LonWorks 장치의 상태정보를 받은 경우, UPnP 브리지 장치는 이벤트 메시지를 요청하는 컨트롤 포인트에게 상태정보를 GENA 형태로 변경하여 전달해준다.

### 4. 구현 및 결과

UPnP - LonWorks 브리지를 구현하고 검증하기 위하여 그림 3과 같은 테스트 환경을 구축하였다.

UPnP 네트워크는 일반적인 TCP/IP 네트워크 상에서 구축되었다. UPnP 장치는 무선랜을 사용하는 PDA를 이용하였으며, 컨트롤 포인트로 PC를 이용하였다. LonWorks 장비로써 전동, 스위치 및 리모콘을 사용하였다. 이러한 장비들은 제안하는 UPnP - LonWorks 브리지를 이용하

여 연결되며, 컨트롤 포인터를 이용하여 제어될 수 있다.

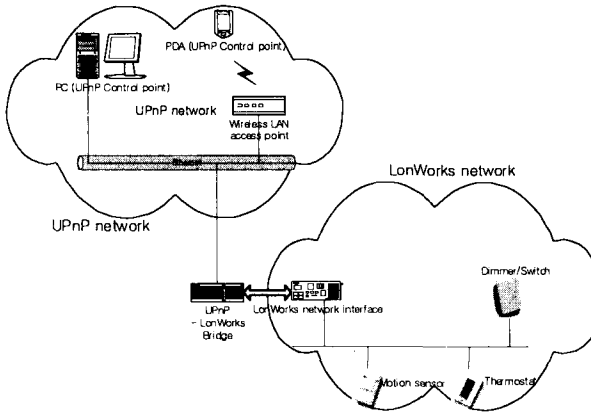


그림 3 UPnP - LonWorks 브리지 테스트 환경

그림 4는 컨트롤 포인터에 나타나는 UPnP - LonWorks 설정 장치를 보여준다. 그림 4는 LonWorks 장비들을 UPnP에서 이용하기 위해 설정하는 웹 문서로써, 그림 1의 UPnP 설정 장치에 의해 생성되는 문서이다.

| Index | Type        | Direction | Name |
|-------|-------------|-----------|------|
| 0     | SNVT_amp(1) | IN(O)     |      |

| Index | Type | Direction | Name            |
|-------|------|-----------|-----------------|
| 0     | 95   | 0         | TEST_SWITCH0000 |
| 1     | 95   | 1         | NV_switch_out10 |
| 2     | 95   | 1         | NV_switch_out20 |
| 3     | 95   | 1         | NV_switch_out30 |
| 4     | 95   | 1         | NV_switch_out40 |
| 5     | 95   | 1         | NV_switch_out50 |
| 6     | 95   | 1         | NV_switch_out60 |
| 7     | 95   | 1         | NV_switch_out70 |

그림 4 UPnP 설정 장치의 프리젠테이션 문서

그림 5는 설정장치에서 설정한 데이터를 바탕으로 구성된 UPnP - LonWorks 브리지 장치를 보여준다. 그림 5의 웹 문서는 그림 1의 UPnP 브리지 장치에 의해 생성되며, 실제적인 UPnP 장치 및 LonWorks 장치 제어를 위해 사용된다.

UPnP LonWorks Bridge Service Page

Network Variable0  
type : SNVT\_switch  
name : TEST\_SWITCH0000  
dimmer value(0 ~ 200): 0  
switch state(0 ~ 1): 0

Submit command

Network Variable1  
type : SNVT\_switch  
name : NV\_switch\_out10  
dimmer value(0 ~ 200): 0  
switch state(0 ~ 1): 0

Submit command

Network Variable2  
type : SNVT\_switch

그림 5 UPnP 브리지 장치의 프리젠테이션 문서

본 논문에서는 UPnP 네트워크와 LonWorks 네트워크의 연동을 위한 브리지 구조를 제시하였다. 이를 위해 UPnP 및 Lonworks 네트워크의 컨트롤 명령들을 변환하고 전달하는 구조를 제시하였으며, 실제적인 구현을 통해 UPnP와 LonWorks 네트워크의 장치들을 제어하고 감시할 수 있는 방법을 제시하였다.

### 5. 참고 자료

- [1] Gerard O'driscoll, The Essential Guide to Home Networking Technologies, Prentice Hall PTR, 2001.
- [2] LonTalk Protocol Specification Version 3.0, 1994.
- [3] Universal Plug and Play Device Architecture Version 1.0, 08 Jun 2000.
- [4] S. H. Park, S. J. Kang, D. H. Seo, and D. H. Park, "Real-Time Middleware Supporting Interoperability between LonTalk-based Control Network and IEEE 1394-based A/V Network in Home," PDPTA '02, 2002.
- [5] Richard Greenane and Simon Dobson, "Integrating LonWorks into an open systems control environment," LonWorld'99, Echelon, 1999.
- [6] LonWorks Host Application Programmer's Guide, Echelon cooperation, 2003.
- [7] Simple Object Access Protocol (SOAP) Version 1.1, 08 May 2000.
- [8] General Event Notification Architecture Base (GENA), 09 July 1998.