

동적 재구성이 가능한 데이터 지향적인 양방향 통합 리모컨의 소프트웨어 프레임워크

신 영 술⁺ · 이 우 진^{**}

요 약

대부분의 기존 통합 리모컨은 동일한 제조회사의 특정 제품군만을 제어할 수 있다. 또한, 시그널 기반의 단방향 통신을 이용하므로 가전기기의 상태를 모니터링 할 수 없다. 본 연구에서는 Wireless Personal Area Network 상에서 가전기기의 상태를 관찰할 수 있는 동적 재구성이 가능한 양방향 통합 리모컨의 소프트웨어 프레임워크를 제안한다. 동적 재구성이 가능한 통합 리모컨은 제어 대상인 가전기기의 프로필에 따라 제어를 위한 GUI를 동적으로 생성함으로써 해당 가전기기를 제어할 수 있도록 재구성될 수 있다. 즉, 하나의 통합 리모컨을 통해 프로필을 가진 모든 가전기기를 제어하고 모니터링 할 수 있다.

키워드 : 통합 리모컨, 동적 재구성, 데이터 지향적, 양방향

Software Framework of Reconfigurable Data-oriented and Bi-directional Universal Remote Controller

Youngsui Shin⁺ · Woo Jin Lee^{**}

ABSTRACT

Most of existing integrated remote controllers can control only a group of home appliances made by the same manufacturer. And they cannot monitor the status of home appliances since they are based on analog RF (Radio Frequency) signal and operate in a uni-directional manner. In this paper, we propose a software framework for a bi-directional universal remote controller (URC) which monitors the status of home appliances in Wireless Personal Area Network (WPAN). The URC can handle the control and status messages which consist of a command with related arguments. When a user wants to control any home appliance, the URC can be dynamically configured to its dedicated remote controller by generating a GUI according to its profile. Any other devices which have a capacity for understanding the profile from the controlled devices can play the same role of the URC.

Key Words : Universal Remote Controller, Reconfigurable, Data-Oriented, Bi-direction

1. 서 론

다양한 종류의 리모컨을 하나의 리모컨으로 단일화하기 위한 연구와 가정 내의 모든 가전기기를 제어하기 위한 통합 리모컨에 대한 연구가 진행 중이다. 유비쿼터스의 한 분야인 홈네트워킹에서 가전기기는 점차 정보가전기기화 되어 가고 있으며, 이를 통합 관리하기 위한 제어기기를 필요로 하는 추세이다[7,8,9,10]. 리모컨에 대한 이상적인 요구사항으로서, 통합 리모컨은 가전기기의 종류에 상관없이 모든 가

전기기를 제어할 수 있어야 하며, 복잡한 제어도 원격에서 단순한 절차를 통해 수행할 수 있어야 한다. 그리고 제어 대상이 되는 가전기기의 상태를 통합 리모컨을 통해 모니터링하고 가전기기의 상태에 따라 적절한 제어 명령을 전달하는 기능을 가져야 한다.

기존의 리모컨은 제어 대상인 가전기기에 제어 명령 시그널을 전송한다. 리모컨이 데이터가 아닌 시그널을 전송하기 때문에 가전기기가 수행해야 할 행위의 종류만 전달할 수 있다. 리모컨의 각 버튼마다 코드가 할당되어 있고 버튼을 누르는 즉시 코드가 시그널로 전달되기 때문이다. 명령 코드와 인자 값을 함께 제어 메시지로 구성하여 가전기기로 전달 할 수 있다면 여러 차례에 걸친 시그널 전송 과정이 간단해지고 한 번의 정보전달로써 가전기기는 하나의 제어 메시지를 이용하여 완전한 명령을 실행할 수 있다.

※ 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT 연구센터 육성지원 사업의 연구결과로 수행되었으며, 두뇌한국(BK) 21 사업 지원에 의해 수행되었습니다.

⁺ 준 회 원 : 경북대학교 전자전기컴퓨터학부 박사과정

^{**} 정 회 원 : 경북대학교 전자전기컴퓨터학부 조교수

논문접수: 2007년 4월 16일, 심사완료: 2007년 9월 10일

가전기기와 단방향 통신 방법을 사용하는 리모컨은 가전기기의 상태를 모니터링 할 수가 없다. 따라서 가전기기의 상태를 고려한 제어가 불가능하다. 리모컨은 가전기기의 상태가 자신이 제어한 상태로 변경되어 있으며, 리모컨이 다른 명령을 전달하지 않은 이상 가전기기는 리모컨이 예측하는 최종 상태에 머물러 있을 것으로 예측한다. 하지만 가전기기 자체적인 요인으로 인한 가전기기의 상태 변화가 발생할 수 있다. 양방향 통신을 이용하여 가전기기가 전달한 상태 정보를 처리할 수 있다면, 리모컨은 가전기기의 상태를 고려한 제어가 가능해진다.

일반적인 통합 리모컨은 미리 제어 대상이 되는 가전기기의 제어 정보를 미리 고정적으로 가지게 된다. 생산 단계에서 고려되지 않은 가전기기는 리모컨을 이용하여 제어할 수가 없다. 하지만 가전기기의 제어 인터페이스 정보가 포함된 프로필을 이용한다면 소프트웨어 프레임워크 상에서 리모컨의 행위를 동적으로 재구성할 수 있다. 재구성이 가능한 리모컨은 가전기기의 제어 정보를 미리 가질 필요 없으며, 가전기기 각각에 해당되는 제어 프로그램을 설치할 필요도 없다. 또한 제어 대상이 되는 가전기기의 종류도 제한되지 않는다.

본 논문은 다음과 같이 구성된다. 현재까지 이루어진 통합 리모컨 연구의 접근 방법에 대해 2장에서 정리한다. URC의 소프트웨어 프레임워크 구조를 3장에서 제시하고, URC와 가전기기 간의 상호작용 메커니즘을 4장에서 설명한다. 5장에서는 계층적인 GUI에 대해 기술하고, 구현과 그에 대한 평가를 6장에서 서술한다. 마지막으로 7장에서 결론을 맺으며 앞으로의 연구 과제를 기술한다.

2. 관련 연구

기존의 통합 리모컨에 대한 연구를 <표 1>에 보인다. <표 1>은 통합 리모컨의 개발 동향과 각각의 통합 리모컨에 대한 특징을 보인다. 각 통합 리모컨을 주요 특징에 따라 3개의 그룹으로 분류하였다. 분류 기준은 가전기기와 통합 리모컨 간의 상호작용과 정보 교환 방식이다.

상호 작용 방법은 단방향 통신과 양방향 통신으로 나눌 수 있다. 단방향 통신을 이용하는 리모컨은 가전기기의 상태를 모니터링 하지 못한다[1, 2, 3, 4, 5, 6, 7, 8, 9]. 정보 교환 방식은 단순한 시그널 방식, 명령 코드와 인자 값으로 구성되는 데이터 지향적인 메시지 방식이 있다. 데이터 지향적인 메시지 방식은 제어 과정을 단순화 시킨다[5, 6, 7, 8, 9, 10].

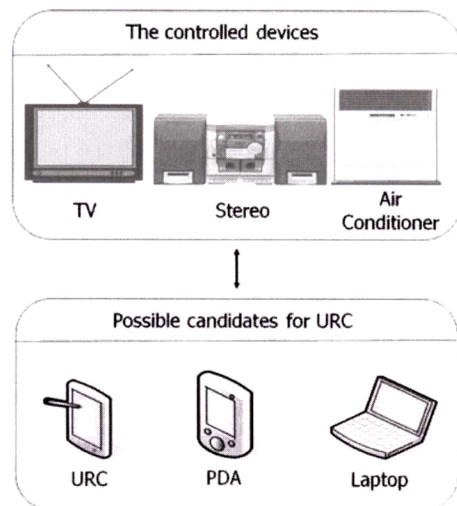
그리고 재구성이 가능한 통합 리모컨에 대한 연구가 이루어지고 있다[2, 3, 4, 6, 7, 8, 9]. 재구성이 불가능한 리모컨은 두 가지로 나뉘어진다. 가전기기의 제어 정보를 미리 고정적으로 가지고 있는 통합 리모컨과 가전기기를 제어하기 위한 전용 프로그램을 가전기기 수만큼 모두 설치하는 통합 리모컨이다[10]. 제어용 전용 프로그램을 설치하는 리모컨은 동적으로 재구성된다고 볼 수 없으며 프로필을 이용하는 리모컨보다 많은 리소스를 필요로 한다.

<표 1> 기존 통합 리모컨 간의 비교

| | Uni/Bi-directional | Signal/Data-oriented | Reconfigurable | Where to get profiles |
|--|--------------------|----------------------|----------------|-----------------------|
| BUTRC [1] | Uni-directional | Signal-oriented | Impossible | The RC |
| BCP/ECP RC [2] GRC [3] | Uni-directional | Signal-oriented | Possible | Appliances |
| Agent-based RC [4] | | | | |
| Mobile Agent RC [5] | Uni-directional | Data-oriented | Impossible | Appliances |
| ICrafter RC [6] BTRC [7] PUC [8] | Uni-directional | Data-oriented | Possible | Appliances |
| Context-aware Controller[9] | Uni-directional | Data-oriented | Possible | Server |
| ECHONET RC [10] | Bi-directional | Data-oriented | Impossible | Server |
| Our URC | Bi-directional | Data-oriented | Possible | Appliances |

추가적으로 가전기기의 프로필이 서버에 저장되어 있는 경우는 서버의 상태에 의존적이게 된다[9, 10]. 가전기기의 프로필은 각각의 해당 가전기기에 저장되는 것이 바람직하다. 프로필 표준화에 대해서는 아직 공식적으로 이루어지고 있지는 않다. 국내외 대기업들은 자체적으로 프로필을 정의하고 있으며, 다른 업체나 회사 간의 표준화는 진행되지 않고 있다. Zigbee의 경우, 프로필을 제작하여 상용으로 판매하고 있지만 GUI 정보에 관한 프로필은 아니다.

본 연구에서는 이상적인 통합 리모컨의 요구사항을 만족시키기 위해 데이터 지향적인 메시지를 이용하여 양방향으로 가전기기와 상호작용하고, 가전기기의 인터페이스 정보로 구성된 프로필을 이용하여 동적으로 재구성이 가능한 통합 리모컨(Universal Remote Controller, URC)을 지원하기 위한 소프트웨어 프레임워크를 제안한다. 제안된 소프트웨어 프레임워크는 WPAN 기반의 URC가 양방향 통신을 이용하여 가전기기와 데이터 지향적인 제어 메시지를 교환한다. URC는 제어 대상이 되는 가전기기의 상태를 관찰하고 가전기기의 상태 변화로 인해 발생하는 이벤트를 처리할 수 있다. 가전기기로부터 인터페이스 정보가 담긴 프로필을 얻어 해당 가전기기를 제어하기 위한 인터페이스를 생성하여 URC를 재구성한다. 따라서, URC 기기 외에 가전기기로부터 전달된 프로필을 통해 재구성될 수 있는 모든 기기는 URC와 같은 역할을 할 수 있다. (그림 1)에서 URC와 같은 역할을 할 수 있는 기기의 예를 보인다.



(그림 1) URC의 역할을 할 수 있는 기기의 예

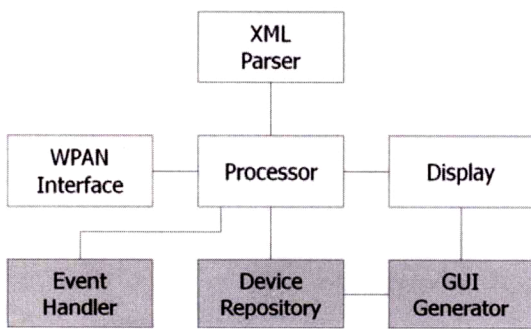
3. URC 소프트웨어 프레임워크 구조

URC는 가전기기의 프로필을 이용하여 인터페이스 정보를 얻고 데이터 지향적인 제어 메시지를 전송함으로써 가전기기에 제어 명령을 전달한다. 또한 URC는 가전기기의 상태를 모니터링하고 가전기기에서 발생한 이벤트를 처리할 수 있다. URC의 소프트웨어 구조는 (그림 2)에서 보인다. 주요 컴포넌트는 다음과 같다.

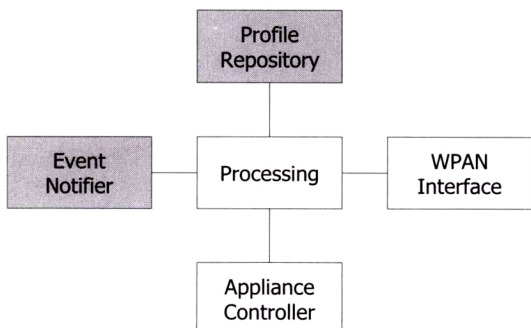
- 이벤트 핸들러는 가전기기의 상태 변화로 인해 가전기기로부터 전달된 이벤트 메시지를 처리한다. 비상 상황에 발생하는 긴급 이벤트 메시지는 가장 높은 우선권을 가지며, 일반 이벤트 메시지는 URC의 상태에 따라 처리되거나 무시된다.
- 디바이스 저장소는 가전기기로부터 전달된 프로필을 저장한다.
- WPAN 인터페이스는 무선 통신을 위한 매체이다.

가전기기로부터 전달된 프로필은 XML 파서를 통해 파싱되고 디바이스 저장소에 저장된다. 저장된 프로필에 해당되는 가전기기를 나중에 다시 제어할 때는 프로필 전송 과정이 필요하지 않다. 프로필에 포함된 가전기기의 인터페이스 정보는 GUI 생성자가 이용한다.

이벤트 핸들러는 가전기기로부터 전달된 모든 이벤트 메시지를 처리하지는 않는다. 현재 제어 중인 가전기기로부터 일반 이벤트 메시지를 전달받았을 경우, URC는 가전기기의 컨텍스트를 전송 받고 가전기기의 상태 정보를 갱신한다.



(그림 2) URC의 소프트웨어 프레임워크 구조



(그림 3) 가전기기의 소프트웨어 구조

제어하고 있지 않은 가전기기로부터 전달된 일반 이벤트 메시지는 처리되지 않는다. 만약 임의의 가전기기에서 발생한 긴급 이벤트 메시지는 URC의 상태에 관계없이 즉시 처리되어 URC는 긴급 상황에 효과적으로 대처할 수 있다.

가전기기가 자신의 상태 정보를 URC에 알리기 위해서는 기존의 가전기기에 새로운 모듈을 추가해야 한다. 하지만 기존 가전기기 구조의 변경을 최소화하는 것이 바람직하다. (그림 3)에서 보이는 가전기기를 구성하는 주요 컴포넌트는 다음과 같다.

- 프로필 저장소는 가전기기의 프로필을 저장한다. 프로필은 자신의 제어 API와 GUI 정보를 포함한다.
- 이벤트 통지 컴포넌트는 가전기기에서 이벤트가 발생하면, 이벤트에 대한 구체적인 정보가 아닌 이벤트 발생 사실을 알린다.

URC로부터 프로필 전송 요청이 전달되면 가전기기는 프로필 저장소에 있는 자신의 프로필을 전송한다. 중앙 처리 컴포넌트는 URC의 제어 메시지를 분석하여 제어 컴포넌트에게 분석된 제어 명령을 전달한다. 제어 명령이 수행된 후, 가전기기는 이벤트 통지 컴포넌트를 통해 변경된 상태 정보를 제공한다.

가전기기가 동작하는 동안에 이벤트가 발생하면, 이벤트 통지 컴포넌트는 이를 인지하고 URC에 통보한다. 기존의 가전기기 구조에 변경을 최소화 하기 위해 가전기기에서 발생한 이벤트에 대한 구체적인 정보가 아닌 이벤트 발생 사실만을 통지하는 것으로 이벤트 통지 컴포넌트의 역할을 제한시킨다.

4. 상호작용 메커니즘

4.1 WPAN 설정

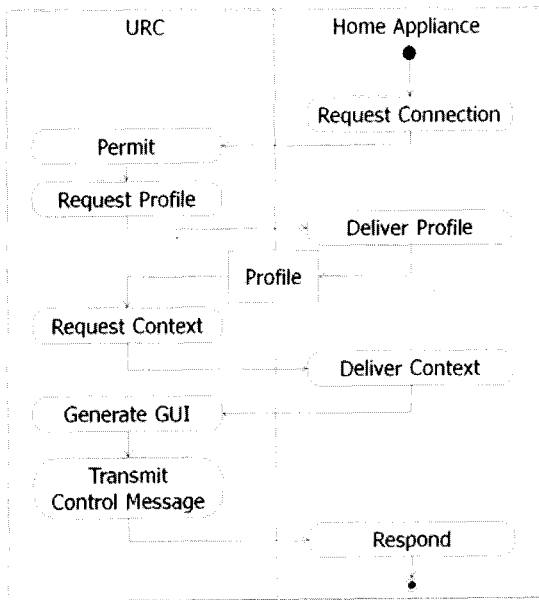
WPAN을 설정할 경우, URC는 마스터 노드 혹은 코디네이터의 역할을 하고, 가전기기는 단말 노드가 된다. 코디네이터는 WPAN을 시작하고 관리하며, 단말 노드인 가전기기가 WPAN에 접속할 수 있는 권한을 제공한다. 권한을 얻은 가전기기는 URC에 의해 논리적 주소를 할당 받는다[11, 12, 13].

URC는 가전기기가 URC에 연결 요청을 하면 가전기기의 고유 IEEE 주소를 확인한다. URC는 접속 권한을 받을 수 있는 IEEE 주소인지 검증을 하고 가전기기에 논리적 주소를 할당한다. 만약 가전기기의 IEEE 주소가 허용된 범위에 있지 않다면 URC는 연결 요청을 처리하지 않는다.

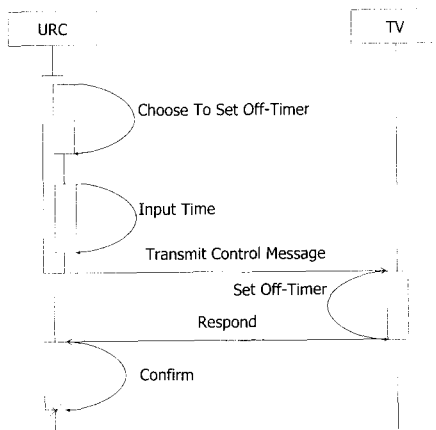
가전기기는 URC에 직접 연결되거나, 라디오 주파수 에너지가 가전기기에 도달하지 않는다면 라우터를 통해 연결될 수 있다[13, 14]. 다시 말해, 라우터를 이용하여 WPAN을 확장하여 모든 가전기기를 URC와 연결할 수 있다.

4.2 URC와 가전기기의 기본적인 동작 방법

URC가 가전기기를 제어하는 과정은 연결 단계, 프로필과



(그림 4) URC와 가전기기의 상호 동작 과정



(a) 꺼짐 예약 기능의 설정 과정

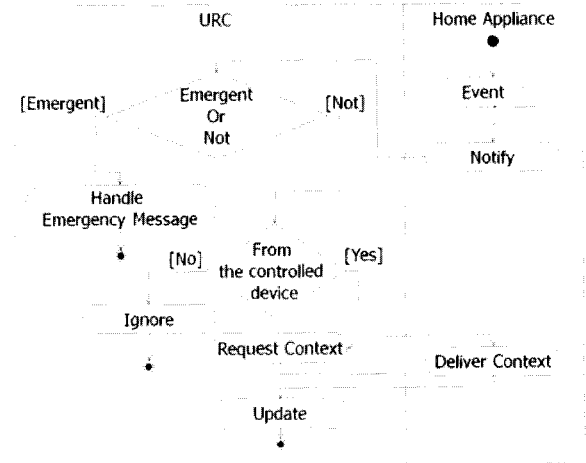
| | | | |
|------------------|-------------------|------------------|--------------------|
| Src. ID : | 0x1111 | Src. ID : | 0x2222 |
| Dest. ID : | 0x2222 | Dest. ID : | 0x1111 |
| Transaction ID : | 0x3333 | Transaction ID : | 0x3333 |
| Cmd : | req_set_off-timer | Cmd : | resp_set_off-timer |
| Hour : | 23 | Hour : | 23 |
| Min : | 30 | Min : | 30 |

(b) 요청 메시지

(c) 응답 메시지

(그림 5) TV 제어를 위한 제어 메시지의 예

컨텍스트 전송 단계, 제어 메시지 전송 단계로 나뉘어진다. (그림 4)는 URC가 가전기기를 제어하는 세 단계를 보인다. 우선 가전기기는 URC에 연결 요청을 하고 논리적 주소를 할당 받는다. 가전기기가 URC와 연결되면, URC는 가전기기에 프로필을 요청한다. XML 형태로 전달받은 프로필은 파싱되어 디바이스 저장소에 저장된다. URC가 하나의 가전기기를 제어한 후에 다른 가전기기를 제어하려 한다면 디바이스 저장소에 저장된 프로필을 이용할 수 있다. URC가 가전기기의 프로필을 가지고 있을지라도 가전기기의 상태 정보는 알 수 없다. 따라서 프로필을 이용하여 GUI를 생성하기 전에 가전기기에 컨텍스트 정보를 요청한다. GUI 생성자



(그림 6) 이벤트 메시지 처리 과정

는 프로필에 포함된 가전기기의 인터페이스 정보를 이용하여 GUI를 생성한다.

(그림 5)는 제어 메시지를 이용하여 TV를 제어하는 예를 보인다. 꺼짐 예약 기능을 선택하더라도 곧바로 해당 명령 코드를 TV에게 전달하지 않는다. URC는 꺼짐 예약 시간을 사용자로부터 모두 입력 받은 후에, 꺼짐 예약 기능을 설정하는 명령 코드와 시간을 나타내는 인자 값으로 구성된 제어 메시지를 전송한다. 따라서 여러 차례에 걸친 데이터 전송 과정이 간략해진다. TV는 제어 메시지를 전달받아 URC의 요청을 수행한 후, 수행 결과를 URC에 알린다.

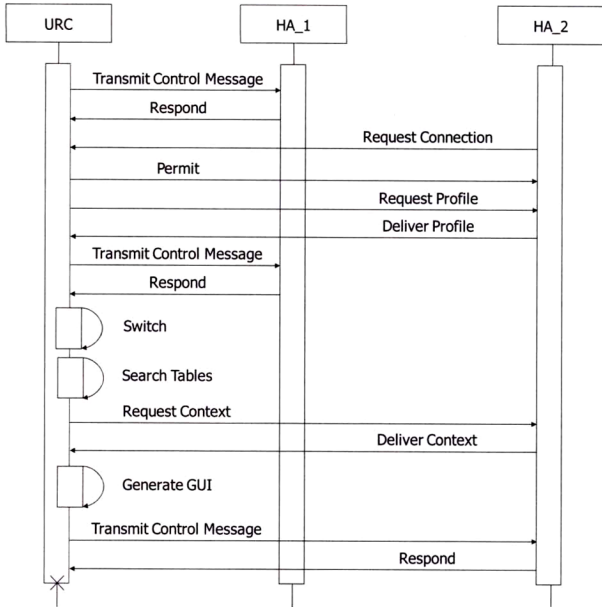
4.3 양방향 통신

이벤트 메시지를 처리하는 과정은 (그림 6)에서 보이는 것과 같다. 일반 이벤트가 가전기기에 발생하면, 이벤트 통지 컴포넌트는 URC에게 이벤트 발생 사실을 알린다. URC가 일반 이벤트 메시지를 받으면, 어느 가전기기가 그 메시지를 전송한지 판단한다. 가전기기가 현재 제어 중인 대상이면, URC는 메시지를 보낸 가전기기에 컨텍스트 전달을 요청함으로써 가전기기의 상태를 업데이트 하여 가전기기의 변경된 상태를 알 수 있다. 가전기기가 URC의 제어 대상이 아니면, 이벤트 메시지는 무시된다.

긴급 메시지는 URC가 즉시 알아야 할 정보를 나타낸다. URC는 곧바로 긴급 이벤트 메시지를 처리하고 이를 디스플레이 장치에 표시한다. URC는 가전기기에 컨텍스트를 요청하고 긴급 상황에 대처할 수 있다.

4.4 프로필 테이블

URC는 장치 저장소에 가전기기의 프로필을 저장한다. 프로필에 포함된 가전기기의 ID, 상세 설명, IEEE 주소를 디바이스 목록 테이블에 저장한다. 프로필에 포함된 제어 API와 GUI 정보는 인터페이스 정보 테이블에 저장된다. 디바이스 목록 테이블은 인터페이스 정보 테이블에 의해 참조된다. 두 테이블의 사용 예는 (그림 7)에 보인다. URC가 HA_1을 제어하는 동안, 새로운 가전기인 HA_2가 URC에 연결 요청을 한다. HA_2는 논리적 주소를 URC에 의해 할당 받고,



| ID | Description | IEEE Address | Interface Reference | Index | Interface Information |
|-------|-----------------|----------------|---------------------|-------|----------------------------|
| HA_1 | TV_Living Room | (unique addr.) | 1 | 1 | Interface Information HA_1 |
| HA_2 | Air Conditioner | (unique addr.) | 2 | 2 | Interface Information HA_2 |
| | | | | | |

(그림 7) 프로필 테이블의 예

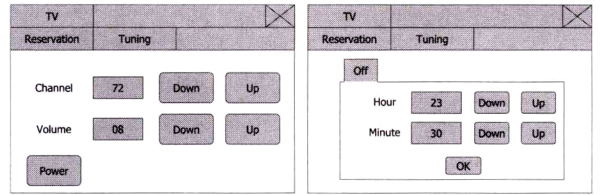
자신의 프로필을 URC로 전송한다. URC가 제어를 HA_2로 전환할 경우, HA_2의 디바이스 목록 테이블에서 IEEE 주소를 검색하여 논리적 주소를 알아낸다. 그 다음, HA_2에게 컨텍스트 정보를 요청하고, 인터페이스 정보 테이블에 저장된 HA_2의 인터페이스 정보와 전달받은 컨텍스트를 맵핑하여 HA_2의 GUI를 생성한다.

5. 계층적 GUI

일반적으로 URC는 포터블 기기이므로 디스플레이 공간이 제한되어 있다. 계층적 GUI는 URC가 제한된 디스플레이 공간을 효율적으로 사용할 수 있게 하고, 직관적인 GUI를 제공할 수 있다. 주요 GUI 요소는 쉽게 사용될 수 있도록 스크린의 전면부에 나타난다. 자주 사용되지 않는 기능이나 제어 명령의 인자 값을 입력 받는 역할을 하는 GUI 요소는 계층적으로 생성된다. 예를 들어 CD 플레이어의 전원 버튼이나 재생 버튼은 빈번히 사용되는 반면 세부 음향 설정 기능은 상대적으로 사용 빈도가 낮으므로 계층적인 메뉴 형식으로 표현할 수 있다. 세부 음향 설정 기능은 계층적 GUI를 통하여 단계적인 상호작용으로 접근 할 수 있다.

TV의 계층적인 GUI 예를 (그림 8)에 보인다. 채널과 볼륨, 전원 버튼은 화면 전면부 패널에 나타난다. 특수한 기능은 계층적 메뉴를 통해 표시된다. 메뉴를 구성하는 메뉴 아이템이 선택되면 팝업 윈도우가 나타난다. (그림 8(b))는 TV의 꺼짐 예약 기능을 설정하는 팝업 윈도우를 보인다.

본 논문에서는 계층적 GUI를 생성하기 위한 태그를 <표 2>



(a) TV의 전면 패널 (b) 꺼짐 예약 시각 설정 패널

(그림 8) TV의 계층적인 GUI

<표 2> 계층적 GUI 표현을 위한 태그

| 태그 | 기능 |
|-------------------------|-------------------------|
| <front>, </front> | 전면부 패널에 그래픽 요소를 표현한다. |
| <pair>, </pair> | 관련 그래픽 요소를 인접하게 위치 시킨다. |
| <menu>, </menu> | 메뉴바에 메뉴를 추가한다. |
| <menuitem>, </menuitem> | 메뉴에 메뉴 아이템을 추가한다. |
| <pop>, </pop> | 팝업 윈도우를 생성한다. |

```

...
<front>
...
<pair>
<label="Channel"/>
<textfield id="chtext", editable="true", type="int"/>
<button id="ch-up", label="Up"/>
<button id="ch-down", label="Down"/>
</pair>
<pair>
<label="Volume"/>
<textfield id="volttext", editable="true", type="int"/>
<button id="vol-Up", label="Up"/>
<button id="vol-down", label="Down"/>
</pair>
...
<button id="power", type="Power"/>
</front>
...
<menu id="reserve" label="Reservation">
<menuitem id="reserv-off" label="Off">
<pop id="off-time" calledby="reserv-off">
<pair>
<label="Hour"/>
<timefield id="offTime"/>
</pair>
</pop>
</menuitem>
<menuitem id="reserv-on" label="On">
<pop id="on-timer" calledby="reserv-on"/>
...
</pop>
</menuitem>
</menu>
<menu id="tune" label="Tuning">
...
    
```

(그림 9) TV 프로필을 구성하는 GUI 정보

에 보이는 것과 같이 정의한다. 태그 <front>, </front>는 전면부 패널에 GUI 요소를 나타낸다. 그리고 채널과 볼륨 버튼은 서로 인접하여 위치하는 것이 사용상의 편의를 제공할 수 있다. 따라서 <pair>, </pair>를 사용하여 관련된 기능들이 인접한 곳에 위치하도록 한다.

특수한 기능은 계층적으로 나타난다. 세부 설정 기능은 <menu>와 </menu>를 이용하여 메뉴 형식으로 표현할 수 있다. <menuitem>, </menuitem>은 메뉴 아이템을 추가하

는 역할을 한다. <pop>와 </pop>는 메뉴 아이템이 선택되었을 시에 해당 기능을 위한 팝업 윈도우를 생성하는데 이용된다. (그림 9)는 (그림 8)에 보이는 GUI를 생성하는 TV 프로필의 예이다.

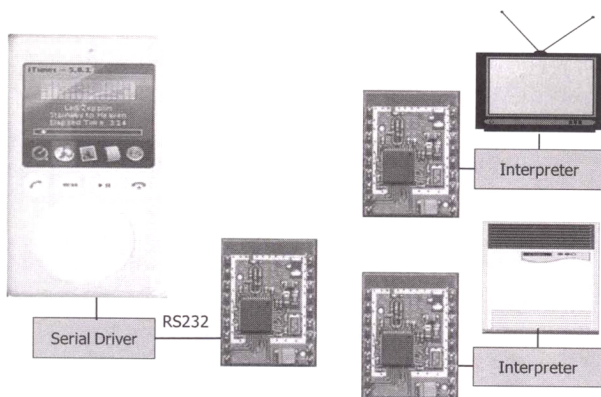
실제로 생성되는 버튼이나 상태 표시창 등의 GUI 요소들은 위한 태그도 정의되어야 한다. 예를 들어, TV 프로필에 사용된 <label>, <button>, <textfield> 등이 실제의 GUI 요소를 나타낸다. URC는 가전기기의 프로필에 포함된 이러한 태그들을 자신이 가진 그래픽 라이브러리에서 지원하는 GUI 요소로 맵핑시킨다. 따라서 동일한 가전기기의 프로필이라도 URC 역할을 하는 기기에 따라 자신에 맞는 GUI를 생성하므로 각기 다른 GUI를 보일 수 있다.

6. 구현과 평가

URC와 가전기기의 프로그램을 (그림 10)과 같이 각각의 컴퓨터에 설치한다. 가전기기 프로그램이 설치된 각각의 호스트는 가전기기 역할을 대신하는 에뮬레이터이다. 시스템은 윈도우 환경에서 자바를 사용하여 구현되었으며, 통신 프로토콜은 Bluetooth v1.1 Class 3[15]을 사용하였다. Bluetooth 모듈을 URC와 가전기기 역할을 하는 호스트에 직접 내장이 불가능하여, Bluetooth 모듈은 호스트와 RS232C 프로토콜을 통해 데이터 교환을 하며, 보드레이트는 115200bps로 설정하였다. 가전기기 에뮬레이터에는 URC로부터 무선을 통해 전송된 메시지를 이해하기 위해 인터프리터가 필요하다.

우선 URC 기본 화면의 메뉴에서 (그림 11)과 같이 현재 URC와 연결되어 있는 가전기기의 목록을 볼 수 있다. 그 목록에서 실내 온도 조절기를 선택한다.

실내 온도 조절기가 선택된 후, URC는 해당되는 프로필



| | |
|------------------|---|
| Operating System | Windows XP |
| Language | Java |
| Library | JDK 5.0 Update 6 JavaComm 2.0 Win32 |
| Bluetooth Module | Bluetooth v1.1 Class 3 Baud rate : 115200 bps |

(그림 10) 시스템의 구성과 환경

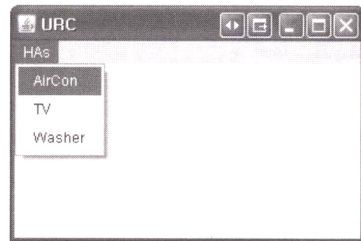
을 통해 GUI를 생성한다.

(그림 12)는 URC가 실내 온도 조절기를 제어하는 예를 보인다. URC가 실내 온도를 섭씨 25도로 설정하는 제어 명령을 전달하고, 실내 온도 조절기는 제어 명령을 수행하고 온도가 섭씨 25도로 설정되었다는 수행 결과를 보고한다.

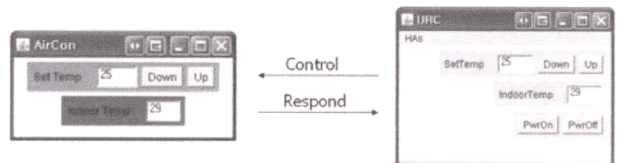
URC가 세탁기로 제어를 전환하고자 할 경우, (그림 13)과 같이 URC와 연결 되어 있는 가전기기들의 목록을 가진 왼쪽 상단의 HAs라는 메뉴에서 세탁기를 선택한다.

(그림 14)와 같이 세탁을 위해 물의 온도를 섭씨 42도로 설정하고, 탈수 레벨은 보통으로 설정한다. 세탁을 완료하는데 소요되는 시간은 35분이다.

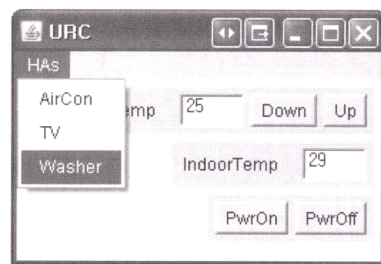
세탁기 제어를 끝내고 TV를 보기 위해 가전기기 목록에서 TV를 선택한다. URC는 TV에 프로필을 요청하고, 전달 받은 프로필을 사용하여 TV 제어를 위해 URC 자신을 재구성한다. URC는 (그림 15)와 같이 계층적 GUI를 통하여 TV의 꺼짐 예약 시각을 설정한다.



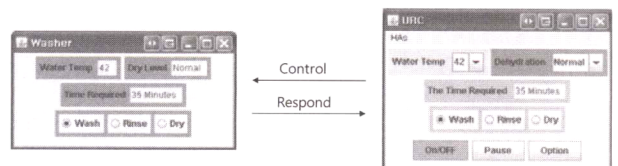
(그림 11) URC와 연결되어 있는 가전기기의 목록



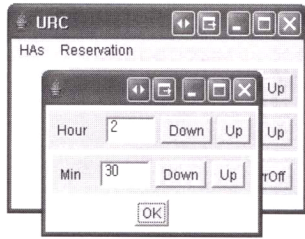
(그림 12) URC에 의한 실내 온도 조절기 제어



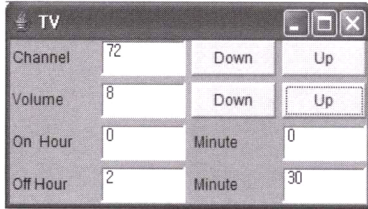
(그림 13) 세탁기 선택 화면



(그림 14) URC를 통한 세탁기 제어

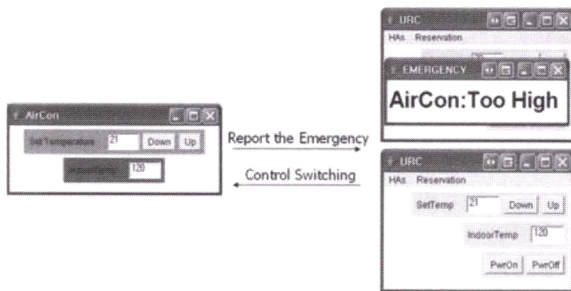


(a) TV 꺼짐 예약을 위한 URC의 GUI



(b) TV

(그림 15) URC와 TV의 상호작용



(그림 16) 긴급 이벤트 발생의 예

<표 3> 성능 측정

| | Elapsed Time |
|--|--------------|
| Transmission of a packet between hosts through RS232C (baudrate : 115200bps) | 10 |
| Registration of a home appliance | 21 |
| Transmission of a profile | 832 |
| Transmission of a context | 292 |
| GUI generation | 32 |
| Home appliance control | 290 |
| Event notification | 31 |

(Time Unit : ms)

URC가 TV를 제어하는 동안 실내 온도 조절기에서 주위 공기 온도가 급격히 상승하는 긴급 이벤트가 발생하였다. 실내 온도 조절기는 URC에 이 사실을 보고한다. URC는 실내 온도 조절기로 제어를 전환하고 긴급 상황에 대처한다. (그림 16)은 이러한 긴급 이벤트의 예를 보인다.

Bluetooth v1.1를 활용한 URC가 TV를 제어하는 과정에서, 제어 단계 별로 성능을 측정하였다. 성능 측정 결과는 <표 3>에서 보이는 것과 같다. 프로파일 전송과 컨텍스트 전송 과정이 많은 수행 시간을 요구하였다.

사용된 Bluetooth의 전송 속도에 비해 프로파일 전송과 컨텍스트 전송 과정이 오래 걸리는 원인은 Bluetooth와 RS232C 간의 속도 차이이다. Bluetooth v1.1에 비해 RS232C가 6배 이상의 느린 속도를 가지므로 RS232C가 병목 작용을 하게 된다. 또한 데이터 전송 과정 상에서의 패킷 손실과 구현에 사용된 자바 언어 자체의 느린 패킷 처리 속도도 성능 저하의 요인이 된다. 가장 큰 성능 저하의 요인인 병목 현상을 해결하기 위해서는, Bluetooth 모듈과 각각의 호스트 간에 RS232C 통신을 사용하지 않고 호스트에 Bluetooth 모듈을 직접 내장시켜야 한다.

7. 결론 및 향후 연구 과제

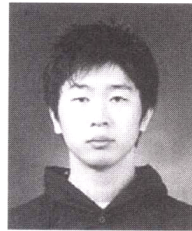
통합 리모컨의 이상적인 요구 사항을 만족시키기 위해 본 논문은 가전기기의 상태를 모니터링 할 수 있는 동적 재구성이 가능한 데이터 지향적 양방향 통합 리모컨의 소프트웨어 구조를 제시하였다. URC와 가전기기의 구조는 양방향으로 상호작용이 가능한 메커니즘을 제공하며 기존의 가전기기 구조로부터 변경을 최소화 시킨다. 또한 URC는 프로필을 가진 모든 가전기기를 제어할 수 있다. 특정 URC 기기 대신, 가전기기의 프로필을 이용하여 GUI를 생성할 수 있는 모든 기기는 URC와 같은 역할을 할 수 있다.

현재 프로필에서 가전기기의 인터페이스를 효과적으로 표현할 수 있는 방법과 URC의 리소스를 고려하여 GUI를 생성하는 방법에 대한 연구가 진행되고 있다. 앞으로 상용 URC 기기 개발과 이를 실제 가전기기에 적용하는 연구가 필요하다.

참고 문헌

- [1] Jia Ren Chang Chien and Cheng-Chi Tai, "The Information Home Appliance Control System - A Bluetooth Universal Type Remote Controller," Networking, Sensing and Control, IEEE International Conference, vol.1, pp.399-400, March 2004.
- [2] Masahito Tezuka et al., "Development of bi-directional remote controller protocol and systems for domestic appliances," IEEE Transactions on Consumer Electronics, vol.46, no.3, August 2000.
- [3] Yu-Chung Yang and Fan-Tien Cheng, "Development of an Autonomous and Generic Remote Control Scheme," in Proc. of ICRA IEEE, 2004.
- [4] Takuo Osaki et al., "An Agent-Based Bidirectional Intelligent Remote Controller," IEEE Transactions on Consumer Electronics, vol.47, no.3, August 2001.
- [5] Soko Aoki et al., "Autonomous and Asynchronous Operation of Networked Appliances with Mobile Agent," in Proc. of ICDCSW 2002.
- [6] Shankar R. Ponnkanti et al., "ICrafter : A Service Framework for Ubiquitous Computing Environments," in Proc. of UBIComp, 2001.

- [7] Fridtjof Feldbusch et al., "The BTRC Bluetooth remote control system," *Personal and Ubiquitous Computing*, vol.7, no.2, July 2003.
- [8] Jeffrey Nichols et al., "Generating Remote Control Interfaces for Complex Appliances," *UIST*, vol.4, pp.161-170, October 2002.
- [9] Youngjae Kim, and Dongman Lee, "A Personal Context-aware Universal Remote Controller for a Smart Home Environment," *Advanced Communication Technology*, vol. 3, pp. 1521-1525, 2006.
- [10] Yosuke Tajika et al., "Networked Home Appliance System using Bluetooth Technology Integrating Appliance Control/Monitoring with Internet Service," *IEEE Transactions on Consumer Electronics*, vol.49, no.4, November 2003.
- [11] Specification of the Bluetooth System Version 1.1.
- [12] ZigBee Specification v1.0
- [13] IEEE Standard 802.15.4
- [14] Nick Baker, "Bluetooth Strengths and Weaknesses for Industrial Applications," in *Computing & Control Engineering Journal*, vol.16, pp.20-25, April-May 2005.
- [15] Ed Callaway et al., "Home Networking with IEEE 802.15.4: A Developing Standard for Low-Rate Wireless Personal Area Networks," in *IEEE Communications Magazine*, August 2002.



신 영 술

e-mail : youngsulshin@msn.com

2005년 경북대학교 전자전기컴퓨터학부
(학사)

2007년 경북대학교 컴퓨터학과 (이학
석사)

2007년~현 재 경북대학교 전자전기
컴퓨터학부 박사과정

관심분야: 소프트웨어 시스템 모델링 및 분석, Embedded S/W
기술



이 우 진

e-mail : woojin@knu.ac.kr

1992년 경북대학교 컴퓨터학과(학사)

1994년 한국과학기술원 전산학과(공학석사)

1999년 한국과학기술원 전산학과(공학박사)

1999년~2002년 한국전자통신연구원
S/W공학연구부 선임연구원

2002년~현 재 경북대학교 전자전기컴퓨터학부 조교수

관심분야: 임베디드 실시간 시스템 모델링 및 분석, Requirements
Engineering, Petri nets, 웹 서비스 기술 등