

동적 설정 가능한 지그비 서비스 게이트웨이 구조

준회원 김성훈*, 강정석*, 정회원 박홍성**, 백승호***, 박재한***

ZigBee Service Gateway Enabling Dynamic Reconfiguration

Seong-hoon Kim*, Jeong-seok Kang*, Associate Members, Hong-seong Park**,
Sung-Ho Baeg***, Jae-Han Park*** Regular Members

요약

지그비(ZigBee)는 무선 통신 표준 기술로서 이를 이용한 많은 서비스 응용들이 개발되고 있다. 이러한 서비스 응용들은 인터넷과의 연결을 위한 서비스 게이트웨이를 필요로 한다. 이러한 지그비 서비스 게이트웨이를 효율적으로 활용하기 위해서는, 지그비 서비스 게이트웨이는 프로파일에 따른 다양한 응용 메시지 변환, 새로운 노드의 추가/삭제에 따른 지그비 서비스의 추가/삭제에 대한 처리, 지그비 네트워크로의 불필요한 메시지 전달 차단과 같은 기능을 제공해야 한다. 본 논문에서는 이러한 기능들을 제공하는 새로운 형태의 지그비 서비스 게이트웨이, ZiGate를 제안한다. 제안하는 ZiGate는 지그비 네트워크와 연동하여 메시지 변환 모듈을 재설정하거나 불필요한 패킷을 차단하는 제공한다. 더불어, ZiGate는 프로파일 기반의 지그비 응용 메시지를 SOAP과 같은 텍스트 형과 이진 형을 갖는 인터넷 서비스 메시지로 상호 변환할 수 있는 기능을 동시에 제공한다. ZiGate를 실제 구현하여 테스트베드에 설치 및 평가를 통하여, 본 논문에서는 제안하는 ZiGate가 효율적임을 보인다.

Key Words : ZigBee Gateway, ZigBee, Internet, Message Translation Module, Dynamic Reconfiguration

ABSTRACT

ZigBee is one of wireless communication standards and many applications based on Zigbee have been developing for commercial applications. Most of those applications require links to Internet like Zigbee service gateways. However, ZigBee service gateway should provides following functions to be effectively utilized itself: translation of messages between diverse ZigBee application profiles and message formats used in Internet, addition/removal of a new ZigBee service without stopping existing services according to addition/removal of a Zigbee node, and no transmission of unwanted packets to ZigBee network. Therefore, this paper proposes a new type of ZigBee Service Gateway, called ZiGate to provide the functions. The proposed ZiGate is capable of re-configuring message translation modules required in case of addition/removal of nodes and doesn't transmit unnecessary packets to the ZigBee networks by synchronizing itself with ZigBee network. And ZiGate provides translation modules between messages defined in Zigbee application profiles and both text-based and binary-based messages used in Internet. In this paper we implements the ZiGate and evaluates it on the test bed. This evaluations show that the proposed ZiGate works efficiently.

※ 본 연구는 BK21 및 산업자원부의 지원에 의해 수행되었습니다.

* 강원대학교 전자통신공학과 산업정보통신 연구실({hs99018, sleeper82}@control.kangwon.ac.kr),

** 강원대학교 IT특성화학부 전기전자공학부 산업정보통신 연구실(hspark@kangwon.ac.kr)(^o: 교신저자)

*** 한국생산기술연구원 로봇기술본부 ({shbaeg, hans1024}@kitech.re.kr)

논문번호 : KICS2007-08-373, 접수일자 : 2007년 8월 20일, 최종논문접수일자 : 2007년 10월 9일

I. 서론

지그비(ZigBee)^[1]는 저 전력, 저 비용을 지원하는 무선 통신 표준 기술로서 현재 이를 이용한 많은 응용들이 개발되고 있다. 이러한 응용에는 가정, 사무실, 빌딩에서 지능형 환경을 구축할 수 있도록 전등 제어나 온도 조절, 도어락 제어 혹은 정보 가전의 정보 전달^[2]의 제어 응용부터 온도, 조도 등의 센서 네트워크^[3] 응용 등 다양한 형태로 있다^[4]. 그리고 이러한 응용들은 지그비 연합(ZigBee Alliance)에서 정의한 프로파일을 기반으로 응용마다 독자적인 메시지 형식을 갖는다.

지그비 게이트웨이는 주로 WSDL 혹은 SOAP 등의 텍스트 혹은 이진 데이터 형의 인터넷용 응용 메시지를 지그비용 응용 메시지로 변환시키는 역할을 수행한다. 이러한 게이트웨이는 주소 매핑, 시그널링 매핑 등을 필요로 하는 일반 게이트웨이와 낮은 결합도를 지원하여 재사용성이 높은 특성의 서비스 지향 구조^[14]를 기반으로 한 서비스 게이트웨이로 구분된다. 본 논문에서는 후자의 방법을 통하여 지그비와 인터넷을 연결하는 방법을 선택하였다.

서비스 게이트웨이를 이용하면 낮은 결합도로 인하여 지그비 네트워크 단에서는 독자적으로 최적화된 기능을 사용할 수 있는 장점이 있다. 하지만 실질적으로 지그비 서비스 게이트웨이를 사용하기 위해서는 다음과 같은 3가지 문제를 해결해야 한다.

첫째, 지그비 서비스 게이트웨이는 다양한 인터넷용 메시지를 지그비 응용 프로파일 메시지 형식에 따라서 변환해야한다. 지그비 서비스의 종류와 인터넷 메시지 종류에 따라 여러 형태의 메시지 변환모듈을 구현해야 하고, 이들을 모두 한 게이트웨이 혹은 여러 게이트웨이에 설치해야 하는 문제가 생긴다. 전자의 경우, 지그비 게이트웨이는 변환을 수행하는 데 있어 다양한 지그비 응용의 메시지 변환을 수행하기 위하여 모든 메시지 변환 모듈을 항상 메모리에 적재해 놓는 것은 비효율적이다. 후자의 경우 비용이 많이 발생하게 된다.

둘째, 지그비 서비스 게이트웨이는 실제 지그비 네트워크와 연동하여 그 정보를 유지해야 한다. 인터넷 응용이 지그비 네트워크상에 존재하지 않는 지그비 노드에 명령을 내리거나 정보를 요청할 경우 불필요한 네트워크 부하 및 데이터 손실을 유발한다.

셋째, 지그비 서비스 게이트웨이가 실제 배치된 후 사용함에 있어, 새로운 종류의 서비스를 갖는 지그비 장치가 네트워크상에 추가될 경우 기존의 서비스에 대한 방해로 최소화하면서 새로운 변환 기능을 제공

할 수도 있어야 한다. 그렇지 않을 경우, 새로운 서비스가 추가 될 때마다 게이트웨이의 동작을 정지시켜야 하는데, 이는 곧 기존의 서비스가 일시적으로 중단되는 문제를 발생시키게 되므로 중요한 데이터의 손실을 가져올 수 있다. 또한 지그비 네트워크상에 특정 서비스가 존재하지 않는 경우 메모리 효율을 위하여 해당 메시지 변환 기능을 제거 할 수 있어야 한다.

인터넷과 지그비 네트워크를 연결하기 위한 지그비 일반 게이트웨이에 대한 연구는^[5,6,7,8,9]가 있다. [5]는 OSGi를 기반으로 센서와 액츄에이터를 서비스 드라이버로 추상화한 API로, [6,7]는 WSDL등의 표준웹서비스 응용으로, [8,9]는 특정 인터넷 응용으로 이들 각각 모두 지그비 메시지를 적합하게 변경하는 것만을 주 기능으로 하는 대리인으로서의 역할만을 수행한다. 하지만 아직까지 앞서 언급한 바와 같은 3 가지 문제를 해결하는 지그비 서비스 게이트웨이는 없다.

본 논문에서는 이러한 3가지 문제들을 해결하기 위한 지그비 서비스 게이트웨이, ZiGate (ZigBee Service Gateway)를 제안한다. 제안하는 ZiGate는 동적 재설정 가능한 메시지 변환 모듈을 제공한다. 메시지 변환 모듈은 지그비 메시지를 두 가지 형태의 인터넷 서비스 메시지로 상호 변경할 수 있는 기능을 제공한다. ZiGate는 코디네이터와 연결되어 실제 지그비 네트워크와 지속적으로 연동함으로써 지그비 장치 정보를 유지한다. 이를 기반으로 ZiGate는 필요한 경우에만, 예를 들면 해당 노드가 지그비 네트워크에 존재하는 경우에 지그비 노드로 데이터를 전송하게 하고, 지그비 네트워크에 장치 유무에 따라서 동적으로 해당 메시지 변환 모듈을 활성화 혹은 비활성화 시킨다. 또한 비활성화 시간이 일정시간이 지나면 해당 모듈을 메모리에서 삭제 가능하게 한다. 이 기능은 새로운 지그비 장치가 연결될 때 새로운 메시지 변환 모듈을 추가시킬 메모리가 없을 경우 삭제 가능한 모듈을 삭제시킴으로 새로운 모듈을 동작시킬 수 있도록 한다. 그리고 실제 테스트 베드를 통한 실험 및 평가를 통하여 제안하는 ZiGate가 효율적임을 보인다.

본 논문은 다음과 같이 구성된다. II절에서는 지그비에 대하여 간단히 설명하고 III절에서는 제안하는 지그비 서비스 게이트웨이인 ZiGate의 구조를 설명한다. IV절에서는 ZiGate에서 사용한 메시지 변환 방식과 관련된 동작들을 제시한다. V절에서는 ZiGate를 구현하여 실제 지그비 장치들과 함께 동작됨을 보이고 측정된 성능지표를 보여줌으로서 ZiGate의 유효성을 보여주고, 마지막으로 VI절에서는 결론으로 맺는다.

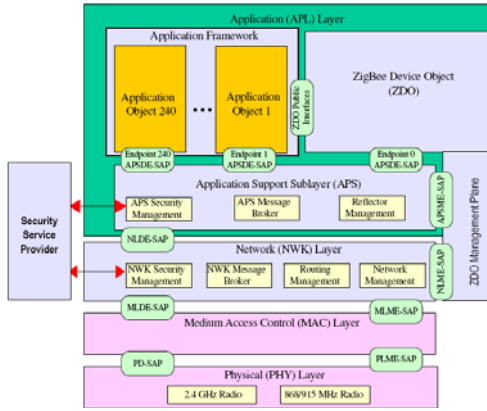


그림 1. 지그비 프로토콜 스택 구조

II. 지그비 개요

지그비 게이트웨이를 개발하는 데 있어 지그비 응용 계층의 특성을 알아야 한다. 그림 1과 같이 지그비는 IEEE 802.15.4^[10]의 매체접근제어계층과 물리계층기술을 바탕으로, 지그비 연합에서 정의한 네트워크 계층과 응용 계층과 같은 상위 계층의 프로토콜로 이루어져 있다

그림1의 구조를 기반으로 지그비는 응용 서비스를 위한 기능을 다음과 같이 두고 있다.

- 장치 기술(Device Description)
- 장치 발견과 서비스 발견
- 응용 레벨의 바인딩

장치 기술(記述)은 프로파일 내의 하나의 장치에 대한 논리적 정의를 말한다. 장치 기술은 프로파일 내에 정의된 입·출력 클러스터로 구성되며 이는 방향성 메시지 구조를 나타낸다. 예를 들면, 그림 2에서 OnOffSRC는 원격 스위치 제어기(Switch Remote Controller: 이하. SRC)의 출력 클러스터 식별자로 이는 동시에 스위치 부하 제어기(Switch Load Controller: 이하. SLC)의 입력클러스터가 된다. 이를 통하여 SRC의 OnOffSRC와 관련된 속성 데이터가 SLC의 입력으로 들어가는 논리적 데이터의 흐름을 알 수 있다.

장치 발견은 장치를 발견하기 위한 것으로 64비트 IEEE 주소 요청과 16비트 네트워크 주소 요청 두 가지가 있다. 서비스 발견은 네트워크상의 장치들을 연결하기 위한 핵심으로서 브로드캐스트 혹은 유니캐스트를 통하여 이루어진다. 이러한 서비스를 위한 질의는 여러 가지 중 다음의 6가지 형태만 간

단히 설명한다.

- 활성 종단점(Active EndPoint)-장치에 존재하는 종단점들에 대한 정보를 알기 위한 질의
- 단순 기술자(단순 기술자)-특정 장치의 종단점이 가지는 프로파일, 장치기술 정보를 알기 위한 질의
- 노드 기술자(Node descriptor)-지그비 노드의 기능 정보를 알기 위한 질의
- 전력 기술자(Power Descriptor)-노드의 전력 공급원 및 전력 상태 정보를 알기 위한 질의
- 복잡 기술자(Complex Descriptor)-장치에 관한 확장된 정보를 알기 위한 질의
- 사용자 기술자(User Descriptor)- 각각의 장치를 사용자가 인식하기 쉬운 사용자 기술자를 알기 위한 질의

응용 레벨의 바인딩은 응용 계층의 개념으로 상호 보완적인 응용 장치의 종단점 사이의 논리적 연결을 말한다. 예를 들면, 그림2의 경우 OnOffSRC는 SRC와 SLC의 논리적 묶음을, LightLevelLSM 단색 조도 센서(Light Sensor Monochromatic: 이하. LSM)과 SLC의 논리적 묶음을 표현한다.

다음 절에서는 이러한 특성을 갖는 지그비와 인터넷과의 연동을 위한 지그비 게이트웨이의 구조를 설명한다.

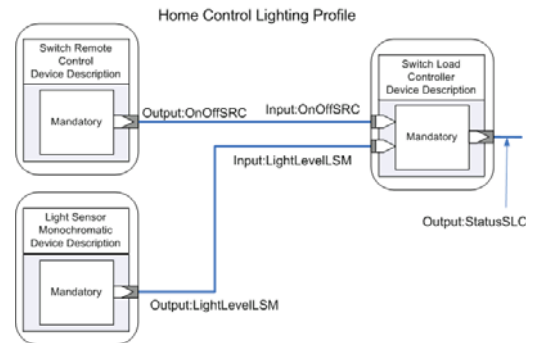


그림 2. 장치 기술을 이용한 장치간의 관계

III. ZiGate의 구조

3.1 ZiGate 개요

제안하는 ZiGate는 기본적으로 메시지 변환은 물론 실제 네트워크에 존재하는 서비스와 동기화하여 메시지 변환 모듈을 재설정 하거나 데이터 송신여부를 결정한다. 이를 위한 ZiGate의 구조는 그림 3

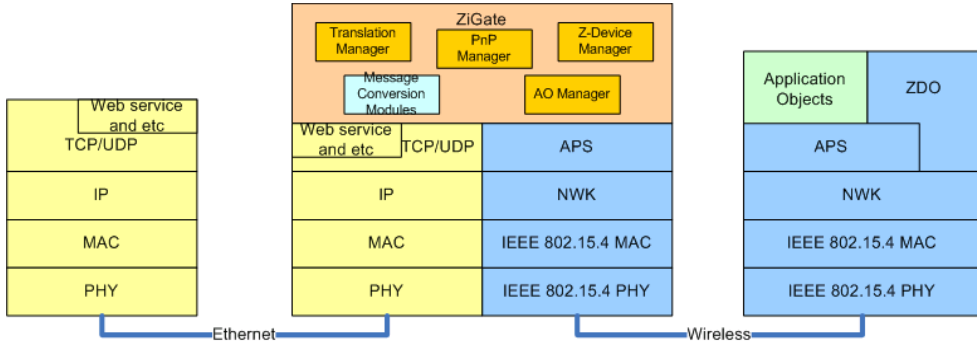


그림 3. ZigBee Service Gateway 구조

에 나타나 있으며 응용 객체 관리자, 지그비 장치 관리자, PnP 관리자, 메시지 변환 모듈, 변환 관리자의 5개 핵심 요소들로 이루어진다.

3.2 ZiGate 핵심 구성 요소

3.2.1 응용 객체 관리자 (Application Object Manager: AOM)

응용 객체 관리자는 ZiGate의 응용 객체들에 종단점 번호를 부여하고 이를 관리하는 역할을 수행한다. 지그비 게이트웨이의 응용 객체는 이 후에 설명될 메시지 변환 모듈의 구성요소로서 지그비의 종단점 번호를 포함하는 단순 기술자와 해당 단순 기술자의 고유 명칭을 가지고 게이트웨이 내에서 유일하게 식별된다. 또한 이 응용 객체들은 지그비의 응용 메시지 형식을 그대로 따르기 때문에 지그비 네트워크의 응용들과 서로 데이터를 주고받을 수 있다.

3.2.2 지그비 장치 관리자 (ZigBee Device Manager: ZDM)

지그비 장치 관리자는 지그비 네트워크에 존재하는 모든 노드들에 대한 서비스 정보를 관리한다. 이를 위해서 지그비 장치 관리자는 새로운 지그비 노드가 네트워크에 참여시 해당 노드의 장치 정보(IEEE주소, 네트워크 주소) 및 서비스 정보(노드 전력, 활성 종단점, 종단점 별 단순 기술자, 사용자, 복잡 기술자)를 수집하고 이를 저장하여 둔다. 그리고 각각의 수집된 지그비 노드가 존재 유무 상태를 지그비 네트워크와 동기화 하여 관리한다. 이 정보는 메시지 변환 모듈을 통하여 데이터 송신 시 목적지 주소에 해당하는 노드가 존재하는 경우에만 데이터를 전송하도록 하여 불필요한 데이터 전송을 없앤다.

3.2.3 PnP 관리자 (Plug and Play Manager: PPM)

PnP관리자는 ZDM에 의하여 제공되는 지그비 네트워크상의 장치 및 서비스 정보에 따라서 필요한 메시지 변환 모듈만 지그비 게이트웨이 내에 적재하도록 한다. 그리고 지그비 네트워크에 새로운 지그비 장치가 추가될 때 혹은 지그비 게이트웨이에 새로운 메시지 변환 모듈이 추가될 때 PnP 관리자는 그림 4와 같이 변환관리자, 지그비 장치 관리자, 응용 객체 관리자들과 협력하여 지그비 게이트웨이의 재설정을 수행하는 책임을 진다.

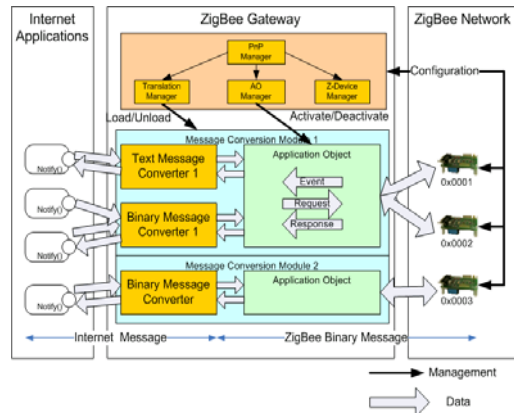


그림 4. ZiGate에서의 모듈 간 관계

3.2.4 메시지 변환 모듈 (Message Translation Module: MTM)

메시지 변환 모듈은 지그비 응용의 메시지 처리를 담당하는 지그비 응용 객체와 이 메시지를 받아들여 인터넷 응용에 맞게 메시지를 변환해 주는 메시지 변환자로 구성된다. 이는 메시지 흐름과 관련

메시지 변환의 수행은 물론 네트워크상의 지그비 장치들이 제공하는 서비스에 동기화 되어 실행시간에 추가/삭제가 가능한 소프트웨어 모듈이다.

3.2.5 변환 관리자 (Translation Manager: TM)

변환 관리자는 PnP 관리자의 명령을 받아서 메시지 변환 모듈을 게이트웨이의 실행 중에 추가하거나 혹은 제거하는 역할을 수행한다. 또 한 지정된 폴더를 감시하여 메시지 변환 모듈의 추가/삭제/변경 여부를 알아내고 이를 PnP 관리자에게 알림으로써 PnP 관리자가 메시지 변환 모듈의 재설정을 수행할 수 있게 한다.

IV. 메시지 변환 방법 및 메시지 변환 모듈 관리

4.1 메시지 변환 모듈 내부 구조

지그비와 인터넷 응용 사이의 메시지 변환을 위해서는 지그비의 응용 프로파일내의 장치기술과 인터넷의 서비스 메시지 형태 사이에 변환이 수반되어야 한다. 이 변환은 지그비 응용형식에 맞춰진 이진 메시지들을 인터넷 응용에서의 서비스 메시지로 표현하기 위해 지그비의 장치 기술을 소프트웨어 인터페이스와 그에 관한 함수로 표현하면 된다.

지그비의 장치 기술은 이진 데이터로 이루어진 프로파일 식별자 및 하나이상의 입출력 클러스터 식별자들로 이루어진다. 따라서 게이트웨이에서는 이러한 지그비 장치 기술을 인터넷에서 들어오는 메시지 종류에 맞게 변환 시켜야 한다. 그림 5는 지그비의 장치 기술과 인터넷 메시지와의 변환을 위한 메시지 변환 모듈 구조를 나타낸 것이다. 메시지 변환 모듈은 서비스 드라이버, 포트 및 ClusterPortPair로 구성되어 있다. 서비스 드라이버는 응용 프로파일에서 정의된 장치 기술내의 모든 클러스터 식별자들을 처리하는 응용 객체를 내장하고 있다. 포트(Port)는 인터넷에서 들어오는 메시지의 변환에 사용하기 위하여, 즉 서비스 드라이버의 기능을 수행하는 데 필요한 동작 정보를 문자열 형태로 보관한 것이다. ClusterPortPair는 클러스터와 포트간의 매핑을 유지하게 하는 일종의 표이다. 포트는 그림 6A와 같은 구조로 동작(Operation), 인수(Argument), 데이터 타입(Data Type)을 가지며 서비스 드라이버와 연결되어 관리된다. 즉, 서비스 드라이버를 수행하는 것은 포트에 연결된 동작을 수행시킬 수 있기 때문에 실제 지그비 장치를 동작하는 것과 동일하다.

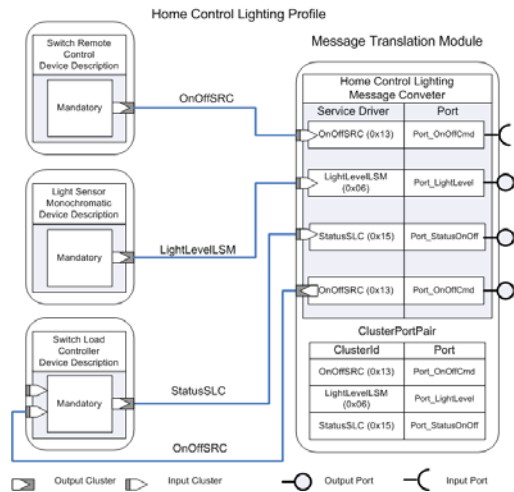
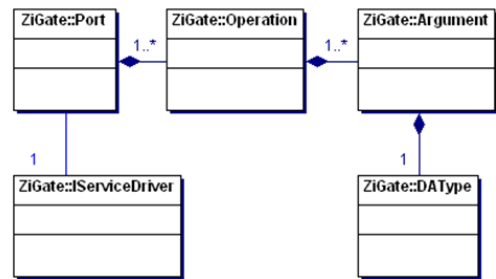


그림 5. 메시지 변환을 위한 메시지 변환 모듈 구조



(A)

```
ZIGate::ClusterPortPair ClusterPortPairList[HCL_CLUSTERPORT_PAIRS] =
{
    ClusterPortPair(CLUSTERID_OnOffSRC, &Port_OnOffCmd),
    ClusterPortPair(CLUSTERID_StatusSLC, &Port_StatusOnOff),
    ClusterPortPair(CLUSTERID_LightLevelLSM, &Port_LightLevel),
};
```

(B)

그림 6. (A) 메시지 변환을 위한 포트 클래스 다이어그램 및 (B) ClusterPortPair 리스트의 예

이는 실제 사용에 있어 다음과 같은 과정을 거치게 된다. 인터넷 응용이 지그비 장치에 데이터를 전송하면, 이진 형식의 경우에는 서비스 드라이버를 직접 사용하여 관련 동작과 인수 등을 추출한다. SOAP등과 같은 텍스트 형인 경우에는 요청패킷으로부터 동작에 해당하는 문자열을 추출한다. 추출한 문자열을 이용하여 그림 6B와 같은 ClusterPortPair 리스트로부터 이 동작에 연결되어 있는 서비스 드라이버를 찾아낸다. 그리고 관련 인수들을 타입에 따라 변환한 후 해당 서비스 드라이버의 함수를 호출함으로써 지그비 장치에 데이터를 전달하게 된다.

이와 반대로 지그비 네트워크에서 인터넷 응용으로 데이터가 전달될 경우의 변환은 지그비 메시지로부터 클러스터 식별자를 추출한 후 ClusterPortPair 리스트로부터 이에 해당하는 포트를 찾아낸 다음 해당되는 동작을 찾아 변환을 역으로 수행함으로써 이루어진다.

4.2 메시지 변환

4.2.1 이진 데이터 기반의 메시지 변환

이진 데이터 변환은 앞서 언급한 바와 같이 서비스 드라이버를 사용하여 변환이 이루어진다. 이에 대한 과정은 그림 7에 나타나 있다. 이 과정에서 중요한 점은 인터넷 응용에서의 하나의 요청은 이에 의해 발생하는 지그비의 요청 패킷의 시퀀스 번호와 맵핑되어 저장되었다가 지그비 응답 패킷이 오면 그 응답의 시퀀스 번호에 해당하는 인터넷의 응답 메시지를 인터넷 응용으로 보내는 점이다. 그림 8은 지그비 네트워크에서 인터넷으로 이벤트 데이터가 전달되는 경우를 나타낸 것이다. 이 과정은 초기화 과정에서 메시지 변환 모듈 내의 서비스 드라이버

이버에 관심 있는 이벤트를 구독함으로써 해당 이벤트에 대한 데이터를 통지 받을 수 있다.

4.2.2 텍스트 기반의 데이터로 변환

지그비 이진 데이터는 또한 사람이 쉽게 인지할 수 있는 텍스트 기반의 데이터로 변환되는 것이 필요하다. 그리고 이러한 텍스트 형의 메시지 변환은 지그비 노드에 존재하는 응용 객체 각각에 맞추어 이루어져야 한다. 이 예로서 SOAP과 ZigBee간의 메시지 변환 과정이 그림 9에 나타나 있다. 이 과정은 4.1절에서 설명한 것처럼 응용 객체가 제공하는 오퍼레이션을 문자열 기반의 동작 표현으로 바꾸고 이를 다시 텍스트 기반의 SOAP 메시지로 변경함으로써 이루어진다. 이 과정은 또한 이진 데이터의 변환에서처럼 지그비 패킷의 시퀀스 번호와 인터넷의 요청 메시지를 맵핑시키고 시퀀스 번호에 해당하는 요청의 응답 메시지를 생성한다.

위 두 가지 메시지 변환과정과 더불어 인터넷의 응용은 지그비 장치상의 응용객체까지 구분할 수 있어야 한다. 이 과정은 새로운 지그비 노드가 추가되면 메시지 변환 모듈이 8바이트 IEEE 주소와 1바이트 종단점 번호를 순서대로 조합한 9바이트의 유일한 식별자를 만들어냄으로써 이루어지게 된다. 따라서 인터넷 응용의 관점에서는 하나의 서비스 동작을 호출 시 하나의 파라미터로서 조합된 9바이트의 유일한 식별자로 사용하면 이를 받은 메시지 변환 모듈은 앞의 8바이트와 뒤의 1바이트를 분리하여 지그비 주소로 변경하여 메시지를 전달하게 된다.

위에서 언급한 두 경우 모두 정적인 환경에서 초기에 설정 시 지그비 네트워크를 구성하는 응용에 맞추어 ZiGate의 메시지 변환자들을 내장하면 된다. 하

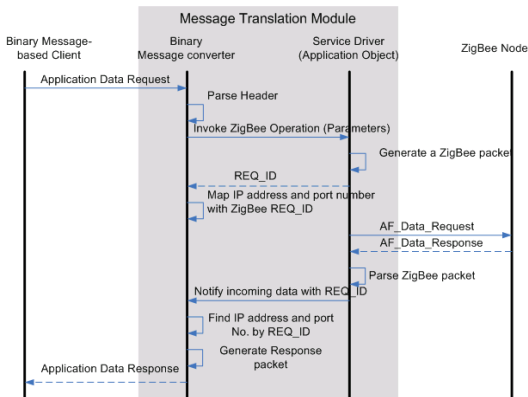


그림 7. 이진 데이터 기반의 인터넷 응용과 지그비 노드와의 요청/응답 메시지 변환 시퀀스

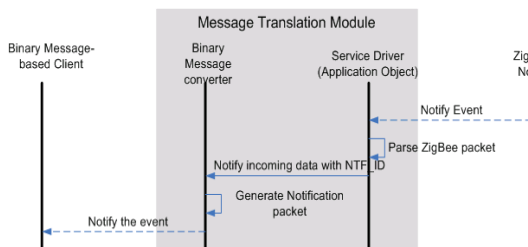


그림 8. 이진 데이터 기반의 인터넷 응용과 지그비 노드와의 이벤트 통지 메시지 시퀀스

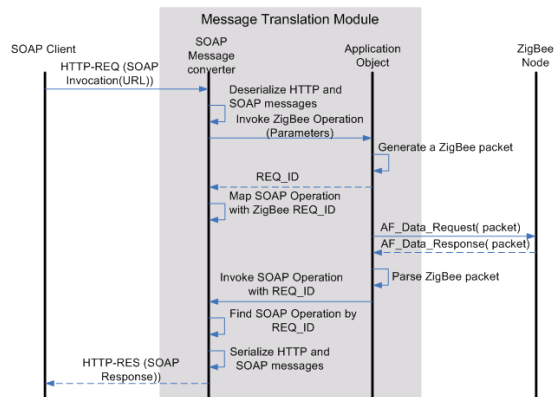


그림 9. 지그비 노드와 텍스트 기반의 SOAP 메시지 사이의 변환

지만, 초기설정 이 후 지그비 네트워크에 새로운 지그비 응용을 추가하거나 변경할 경우에 지그비 게이트웨이도 이러한 메시지 변환기들은 함께 추가하여야 한다. 그리고 이러한 과정에 메시지의 변환은 기존의 통신에 대한 방해가 없이 이루어 져야 한다.

4.3 메시지 변환 모듈 동적 추가/삭제

ZiGate가 배치가 된 이 후에 새로운 장치의 추가에 따라서 게이트웨이의 메시지 변환 모듈은 추가 혹은 변경될 수 있어야 한다. 이를 위한 과정은 다음의 3개의 과정에 따라 수행된다. 초기화, 노드 추가 및 메시지 변환 모듈 추가, 메시지 변환 모듈 추가 및 삭제.

4.3.1 초기화

초기화 시에 TM은 미리 설정된 폴더로부터 모든 MTM들을 로딩하고 단순 기술자를 추출하여 이를 MTM 모듈의 이름(DLL 파일명)과 함께 TM이 내부의 저장소에 저장한다. 그리고 이 후 새로운 메시지 변환 모듈이 추가되는 여부를 감지하기 위하여 디렉토리를 감시한다.

4.3.2 재설정 과정

ZiGate에서 재설정하는 과정은 지그비 노드가 네트워크에 참여함에 따라 메시지 변환 모듈이 메모리에 적재되는 것으로 크게 다음의 2가지 방법이 있다.

첫 번째 방법은 메시지 변환 모듈이 지정 폴더에 있지만 메모리에 적재되지 않은 상태에서 새로운 지그비 장치가 ZiGate에 등록되는 경우이다. 이 경우는 그림 10에 나타나 있다. 지그비 네트워크상에 지그비 장치가 등록되면, 이는 지그비 코디네이터를 통하여 PnP 관리자가 이를 알게 된다. 그러면 PnP 관리자는 ZDM에게 새 노드가 추가되었음을 알리고 (1) ZDM은 장치의 서비스 정보(활성 중단점, 단순 기술자)를 ZDO가 제공하는 질의 인터페이스를 통하여 수집한다(2). 다음으로 관련 정보의 수집이 완료되면 ZDM은 PnP 관리자에서 장치의 추가가 완료되었음을 알린다(3). PnP관리자는 추가된 정보 중 단순 기술자 내용을 가지고 AOM에 처리할 수 있는 메시지 변환 모듈이 존재하는 지 검사한다(4). 없다면, TM에게 새로운 단순 기술자를 처리할 수 있는 MTM을 요구한다(5). TM은 현재 단순 기술자와 정합되는 MTM이 모듈이 존재하는 지를 확인하고(6) 이에 해당하는 MTM의 이름을 가지고 MTM을 로딩한다(7). 그리고 로딩한 MTM을 PnP 관리

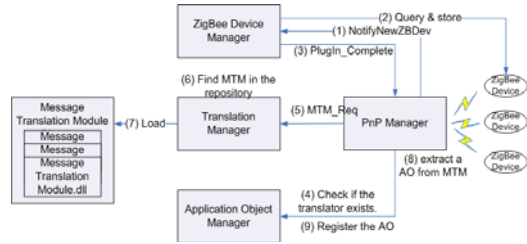


그림 10. 메시지 변환 모듈이 지정 폴더에 있지만 메모리에 적재되지 않은 상태에서 새로운 지그비 장치가 추가되는 경우의 재설정 과정

자에게 돌려준다(8). PnP 관리자는 받은 MTM에서 AO를 추출하고 이를 AOM에 등록 시킨다(9). AOM에 등록된 이 후에는 AO는 중단점 번호를 부여받고 이 시점부터 새로 추가된 지그비 장치에 대하여 동적인 메시지 변환 기능 제공함으로써 지그비 네트워크와 인터넷 간의 연동이 가능하게 한다.

두 번째 방법은 메시지 변환 모듈이 지정 폴더에 존재하지 않는 상태에서 지그비 노드가 등록된 후 메시지 변환 모듈을 추가하는 방법이다. 이는 새로운 MTM이 추가되는 경우에는 그림 11과 같은 과정을 수행한다. 그림 11에서, 먼저 TM은 새로운 MTM이 추가되었음을 디렉토리 감시를 통하여 알아낸다(1, 2). 그러면 TM은 이 모듈을 로딩하여 단순 기술자 정보를 추출하고 MTM 모듈의 이름(DLL 파일 명)과 함께 단순 기술자를 MTM 내부에 저장한다(3). TM은 이 정보를 PnP관리자에게 단순 기술자와 함께 알리고(4) 이를 알게 된 PnP관리자는 단순 기술자를 가지고 ZDM에게 현재 이 MTM과 변환 가능한 단순 기술자를 갖는 비활성화된 장치가 존재하는 지를 검사한다(5). 만일 존재한다면 로딩 후 MTM에서 AO를 추출하고(7) AOM에 등록 시킨다(8). 그렇지 않은 경우에는 해당 장치를 언로딩 한다(6). 그리고 이 장치의 실질적 로딩은 추 후 실제 장치가 추가될 경우에 수행된다.

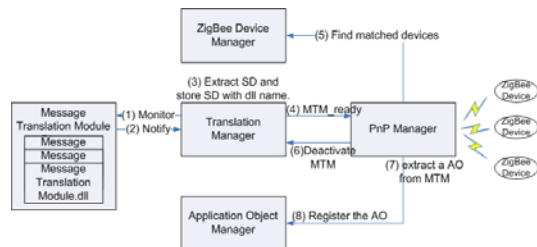


그림 11. 메시지 변환 모듈 존재하지 않는 상태에서 지그비 노드가 등록된 후 메시지 변환 모듈을 추가하는 경우 재설정 과정

4.3.3 메시지 변환 모듈 갱신 및 삭제

메시지 변환 모듈의 삭제 및 갱신은 위의 2의 과정에서 지그비 노드가 제거 및 갱신됨을 지그비 스택을 통하여 PnP 관리자가 알게 되고 이를 ZDM에 알려 동일한 단층 기술자를 갖는 지그비 장치가 더 이상 존재 하지 않는 경우에만 일정시간동안 지난 후에 MTM을 제거하게 한다. 그리고 디렉토리의 변환 모듈을 제거 및 갱신하는 경우 TM이 디렉토리 감시를 통하여 제거 및 갱신된 MTM 정보를 알게 되고 이를 PnP관리자에게 알려 내부적으로 변환 모듈이 제거 및 갱신되었음을 알리고 PnP관리자는 AOM으로 하여금 이에 해당하는 AO를 제거 및 갱신하도록 한다.

4.4 게이트웨이와 지그비 네트워크의 연동

지그비 네트워크와 ZiGate간의 연동은 지그비 장치 프로파일(ZigBee Device Profile: 이하 ZDP)에서 제공하는 서비스를 사용함으로써 가능하다. 이는 기본적으로 게이트웨이에서 표준에서 제공하는 ZDP에서 주기적인 질의를 통한 방법과 지그비 네트워크의 라우터에서 ZiGate로 통지하는 방법이 있다. 제안하는 게이트웨이는 2가지의 경우를 모두 지원한다. 하지만 전자에 경우에는 주기적인 질의에 의한 네트워크 부하가 심하다는 단점이 있기 때문에 이 논문에서는 후자의 방법을 사용한다.

통지에 의한 방법은 코디네이터 혹은 라우터 노드가 자신에게 참여/떠남 등의 지그비 노드에 대한 정보를 ZiGate에 알림으로서 수행된다. 이 과정은 새로운 노드가 참여할 경우 부모 라우터 노드는 ZDP의 발견 등록 요청(Discovery Register request)을 해당 노드의 네트워크 주소 및 IEEE주소와 함께 ZiGate로 통지함으로써 ZiGate는 새로운 노드의 추가 혹은 제거 정보를 알 수 있다. 여기서 새로운 장치의 추가에 대하여 ZiGate는 ZDP의 서비스 질의를 이용하여 필요한 장치 정보를 수집한다. 그리고 기존의 장치가 이동에 의한 경우에는 ZDP의 종단 장치 알림(End device announcement)을 통하여 ZiGate는 노드의 이동을 알 수 있다. 이러한 과정에서 수집되거나 변경되는 정보들은 모두 ZiGate 내의 지그비 장치 관리자의 내부 저장소에서 관리된다. 이러한 과정을 통하여 ZiGate는 지그비의 네트워크에 존재하는 장치 및 서비스 정보와 동기화하여 연동 될 수 있다.

V. 구현 및 성능 평가

4.1 구현

본 논문에 제안하는 ZiGate는 C/C++를 이용하여 구현하였다. 그리고 제안하는 ZiGate의 기본적 동작 능력을 검증하기 위하여 ZiGate내의 메시지 변환 모듈로서 Chipcon^[11]에서 제공하는 홈 전등 제어 (Home Control Lighting) 프로파일에 정의된 장치 기술들의 일부인 단색 조도 센서(Light Level Monochromatic: 이하LSM), 스위치 원격 제어기 (Switch Remote Controller: 이하 SRC), 스위치 부하 제어기(Switch Load Controller: 이하 SLC)들을 두었다. 그리고 응용 객체들을 텍스트 기반인 SOAP메시지로 변환할 수 있는 메시지 변환 모듈을 구현하였다. 더불어 이진형태의 메시지 변환을 수행할 수 있도록 하기 위한 것으로 LSM의 이진 메시지 변환 모듈을 구현하였다.

ZiGate에 연결되어 사용될 베이스 스테이션으로서는 Chipcon의 CC2430EB^[11]를 사용하였으며 이는 RS232C(115200bps)를 통하여 ZiGate가 설치된 PC와 연결하였다. 그리고 지그비 노드로는 SLC와 LSM이 탑재된 CC2430DBK를 사용하였고 이 지그비 노드들과 ZiGate간의 통신을 확인하고자 패킷스니퍼를 PC와 연결하여 사용하였다. 그리고 인터넷의 응용으로는 텍스트 기반의 메시지를 사용하는 클라이언트로서 CyberLink의 UPnP^[12]를 수정하여 HTTP기반의 SOAP클라이언트를 구성하였고 이진 데이터를 사용하는 응용으로는 Internet Communication Engine(Ice)^[13]을 사용하였다. 그림 12는 이러한 테

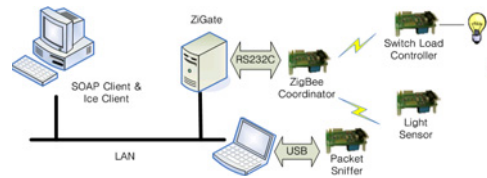


그림 12. 테스트 베드 구성

표 1. 테스트 베드 사양

| Client PC | Gateway | ZigBee Base Station and nodes |
|------------------|------------------|-------------------------------|
| Pentium 2.13 GHz | 4 Pentium 1.8GHz | 4 8051(32MHz) |
| 1 GB RAM | 256 MB RAM | 4KB RAM |
| Windows XP | Windows XP | - |

스트 베드 구성을 나타내고 표1은 그에 따른 사양들을 나타낸다.

4.2. 성능 평가

이 논문에서 제안하는 지그비 게이트웨이의 성능을 검증하기 위하여, 그림 13의 테스트베드에서 노드 설정에 걸리는 시간 및 인터넷 응용에서 지그비 노드까지 명령을 수행하여 응답을 받을 때까지 소요되는 응답 시간을 각 경우에 대하여 50회씩 측정하였다. 그림 13는 ZiGate의 실행 화면을 나타내며, 그림 14는 ZiGate실행 후 새로운 지그비 노드의 추가에 따른 데이터 수집을 패킷 스니퍼로 캡처한 화면을 나타낸다.

새로운 지그비 노드 추가에 따른 설정 시간은 ZiGate가 최초 새로운 노드의 추가를 통보 받은 이후 필요한 지그비 서비스 정보 수집 및 관련 메시지 변환 모듈을 적재하는 데 까지 걸리는 시간으로 Tsetup으로 표시하며, 설정 과정은 다음과 같다.

```

IEEE Address: 17:16:15:14:13:12:00:00:
ZDApp::ProcessEvent is called
AF_MGMT_SAP_CMDHandleJoinIndication
ShortAddress:0001
aa:15:01:00:00:4b:12:00:
CapabilityInformation:8e
ZDApp::ProcessEvent is called
ACT_EndPoint Request to 0001
ZPNP_START_PLUGIN_EVT: Collection Procedure
ShortAddress:0001
aa:15:01:00:00:4b:12:00:
CapabilityInformation:8e
ZDApp::ProcessEvent is called (A)
Response Line 0.062136
SimpleDescription Request to 0001
ZDApp::ProcessEvent is called
!=====!Handle Plug-in Complete
Text::LighteningMessageConverter::Init
A New Application Object is added
End Point Number: 240 Task Id: 2 (B)
Plugin Complete Line 0.142240
GetPortbyClusterId
ENTER =====> SLC Device Type WSDL
    
```

그림 13. ZiGate 실행화면(A: 장치 추가에 데이터 수집 시작, B: 데이터 수집완료에 따른 새로운 응용 객체 추가)

| APS Dest Endpoint | APS Cluster Id | APS Profile Id | APS Src Endpoint | APS Counter | APS Payload | LOI |
|--|----------------|----------------|------------------|-------------|-------------------------------------|-----|
| 0x000 | 0x0005 | 0x0000 | 0x000 | 7 | 00 01 00 | 228 |
| Active Endpoint Request(0x0005)/Response(0x8005) | | | | | | |
| 0x000 | 0x0005 | 0x0000 | 0x000 | 0 | 00 00 01 00 01 06 | 224 |
| Simple Descriptor Request(0x0004)/Response(0x8004) | | | | | | |
| 0x000 | 0x0004 | 0x0000 | 0x000 | 1 | 01 00 01 00 04 00 FE FF 00 01 13 00 | 228 |
| Node Descriptor Request(0x0002)/Response(0x8002) | | | | | | |
| 0x000 | 0x0002 | 0x0000 | 0x000 | 2 | 02 00 01 00 01 00 00 50 00 00 00 00 | 236 |

그림 14. 그림 10의 (A)과정에 따른 장치 수집과정의 패킷 캡처 장면

지그비가 추가 되면 ZiGate는 해당 노드에 대한 정보를 수집한다. 그리고 필요에 따라서 관련된 MTM을 동적 로딩한다. 이 과정에서 메시지 변환을 위하여 최소한으로 필요한 것은 활성 종단점 리스트와 각 종단점에 따른 단순 기술자이다. 이것을 수집하는 데 걸리는 시간은 활성 종단점의 수에 따라 조금씩 차이는 나지만 나머지는 동일하다.

표 2는 SLC를 하나의 활성 종단점으로 갖는 지그비 장치에 대하여 기본 설정 시 걸리는 시간과 전체 설정에 걸리는 설정 시간을 나타낸다. 여기서 기본 설정은 활성 종단점, 단순 기술자 정보만을 수집하는 것을 말하며 전체 설정은 최소 설정은 물론 노드, 전력, 사용자 기술자를 모두 수집하는 것을 의미한다. 결과를 보면 알 수 있듯이 전체 설정의 경우 기본 설정보다 노드, 전력, 사용자 기술자에 대한 요구를 수행하기 때문에 약 116ms 정도 더 긴 설정 시간이 소요됨을 볼 수 있다.

다음으로는 ZiGate가 이진데이터와 텍스트 데이터 모두를 동시에 효율적으로 처리함을 보이고자 이들 각각에 대하여 ZiGate에서 조도 센서(LightSensor)로 조도의 값을 요청하는 데 걸리는 응답 시간, Trsp를 측정하였다. 표 3은 응답 시간에 대한 측정 결과를 나타낸다. 결과를 보면 Ice기반의 응용이 더 빠르게 처리됨을 알 수 있는데 이는 SOAP의 경우 HTTP까지 거쳐 갈 뿐 만 아니라 문자열을 일일이 처리해야 하기 때문에 상대적으로 긴 응답 시간을 보임을 알 수 있다.

결론적으로 이는 제안하는 ZiGate가 실제 지그비 네트워크와 연동을 통하여 메시지 변환 모듈을 효율적으로 설정 및 재설정할 뿐만 아니라 이진 형식의 메시지 및 텍스트 기반의 메시지를 모두 원활히 처리함을 보여 준다.

표 2. 설정에 걸리는 시간: Tsetup(A: 전체 설정 B: 기본 설정)

| | A (전체 설정) | B(기본 설정) |
|------------|-----------|----------|
| 평균(sec) | 0.245 | 0.129 |
| 표준 편차(sec) | 0.009 | 0.012 |

표 3. 이진 메시지 및 텍스트 메시지 처리에 걸리는 응답 시간: Trsp

| | Binary(Ice) | Text(SOAP) |
|------------|-------------|------------|
| 평균(sec) | 0.063 | 0.142 |
| 표준 편차(sec) | 0.012 | 0.010 |

VI. 결론

본 논문에서는 지그비 인터넷 게이트웨이를 사용할 때 생기는 문제점들을 제기하고, 이러한 문제점들을 해결하기 위한 새로운 구조의 지그비 게이트웨이(ZigBee Gateway: ZiGate)를 제안하였다.

제안하는 ZiGate는 동적 재설정 가능한 메시지 변환 모듈을 제공하고 이를 통하여 지그비 메시지를 두 가지 데이터 형의 인터넷 메시지로 쉽게 상호 변경할 수 있음을 보였다. 그리고 ZiGate는 코디네이터와 연결을 통한 실제 지그비 네트워크와 지속적으로 연동하여 지그비 장치 정보를 저장하게 함으로써 필요한 경우에만 지그비 노드로 데이터를 전송하게 하고, 지그비 네트워크에 장치 유무에 따라서 동적으로 해당 메시지 변환 모듈을 활성화 혹은 비활성화 시킴을 보였다. 제안한 ZiGate를 실제 구현하고 이를 테스트 베드를 통해 시험함으로써 제안하는 ZiGate가 효율적임을 보였다.

앞으로는 제안한 ZiGate를 적용하는 데 요구되는 범위성(Scalability)과 시스템의 재설정 시간을 분석하는 연구가 필요하다.

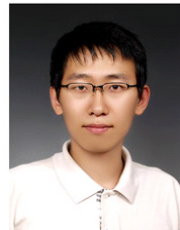
참고 문헌

- [1] Zigbee Alliance, "Zigbee specification: Zigbee document 053474r06 Version 1.0", 14 Dec. 2004. Web site: www.zigbee.org
- [2] E. Petriu, N. Georganas, D. Petriu, D. Makrakis, and V. Groza. Sensor-based information appliances. IEEE Instrumentation and Measurement Mag., 3:31--35, 2000.
- [3] Wheeler, A. "Commercial Applications of Wireless Sensor Networks Using ZigBee", IEEE Communications Magazine, Vol: 45, Issue: 4, pp: 70-77, April 2007
- [4] Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E., "Wireless Sensor Networks: A Survey." Computer Networks Journal, 38, 4 (2002), pp. 393 - 422.
- [5] King, R. Bose, S. Pickles, A. Helal, S. Vander Ploeg and J. Russo, "Atlas: A Service-Oriented Sensor Platform," the 4th ACM Conference on Embedded Networked Sensor Systems (Sensys), Boulder, Colorado, USA

- [6] Mark Gaynor, Steven L. Moulton, Matt Welsh, Ed LaCombe, Austin Rowan, John Wynne, "Integrating Wireless Sensor Networks with the Grid," IEEE Internet Computing, vol. 08, no. 4, pp. 32-39, Jul/Aug, 2004
- [7] <http://www.arcx.com/archronix/>
- [8] <http://www.zigbee.org/en/documents/SensorsExpo/7-Sensors-Expo-kinney.pdf>
- [9] <http://www.cirronet.com/zg2400e.htm>
- [10] IEEE Computer Society, "Part 5.4: Wireless Medium Access Control and Physical Layer Specifications for Low-Rate Wireless Personal Area Networks(LR-WPANs)", 1 Oct. 2003.
- [11] <http://www.chipcon.com/>
- [12] <http://www.cybergarage.org/net/upnp/cc/index.html>
- [13] <http://www.zeroc.com/>
- [14] Michael N. Huhns and Munindar P. Singh , "Service-Oriented Computing: Key Concepts and Principles," IEEE Internet Computing, vol. 9, no. 1, pp. 75-81, Jan-Feb 2005.

김 성 훈(Seong-hoon Kim)

준회원



2006년 2월 강원대학교 전기전자
정보통신공학부 졸업
2006년 3월~현재 강원대학교 전자
통신공학 전공 석사과정
<관심분야> 무선 통신, 미들웨어

강 정 석(Jeong-seok Kang)

준회원



2006년 8월 강원대학교 전기전자
정보통신공학부 졸업
2006년 9월~현재 강원대학교 전
자통신공학 전공 석사과정
<관심분야> 미들웨어, 분산 컴퓨팅

박 흥 성(Hong-seong Park)

정회원



1983년 2월 서울대학교 제어계측
공학과 졸업
1986년 2월 서울대학교 제어계측
전공 석사
1992년 2월 서울대학교 제
어계측전공 박사
1992년~현재 강원대학교 전기전
자정보통신공 학부 교수

<관심분야> 실시간 통신, 무선데이터통신, 미들웨어

박 재 한(Jae-Han Park)

정회원



2000년 2월 부산대학교 전자 공학
과 석사
2000년~2003년 (주)신성전자 부
설연구소 근무
2004년 2월~현재 한국생산기술
연구원 로봇기술본부 제어인식
연구팀 연구원

<관심분야> 로봇 비전, 로봇 작업, 스마트 환경

백 승 호 (Seung-Ho Baeg)

정회원



1993년 2월 고려대학교 재료공학
과 석사
1993년~2003년 한국생산기술 연
구원 나노표면기술 연구팀 선임
연구원
2004년 1월~현재 한국생산기술 연
구원 로봇기술본부 제어인식연

구팀 수석연구원

<관심분야> 머신 비전, 센서네트워크, 스마트 환경