



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)



碩士學位論文

터널링 기법에 적용한 IP-USN
연동 기법 연구



濟州大學校 大學院

컴퓨터工學科

탁부수릉절자야

2010年 8月

목 차

그림목차	iii
표 목차	iv
국문초록	v
영문초록	vii
약어표	ix
I. 서론	1
II. 관련 연구	3
1. USN	3
1) 네트워크 기술동향	4
2) OS 동향	6
(1) TinyOS	6
(2) SOS	8
(3) MANTIS	8
(4) ANTS	9
(5) 나노 Qplus	9
2. IP-USN	9
3. 관련 네트워크 프로토콜 기술	11
1) Zigbee	11
2) 6LoWPAN	13
4. IP-USN 연동 기술	16
1) 게이트웨이 기반 연동 기법	16
2) 오버레이 연동 기법	17
3) 터널링 연동기법	18
(1) 정적 및 동적 연동 기법	18
(2) PAT 연동 기법	19

III. 제안 시스템 설계	21
1. IP-USN을 위한 연동 설계	21
1)터널링을 위한 IP 프로토콜 재정의	22
2)터널링을 위한 베이스 스테이션	23
3)적용 시나리오	25
IV. 시스템 구현	27
1. 흐름도	28
1)센서 노드-> 베이스 스테이션 처리 모듈	29
2)LMT 검색 모듈	29
3)베이스 스테이션 -> 도착지 베이스 스테이션 정보 전송 모듈	30
4)도착지 베이스 스테이션 -> 도착지 노드 전송 모듈	31
2. 구현 결과	32
3. 결과분석	36
V. 결과	38
참고문헌	39

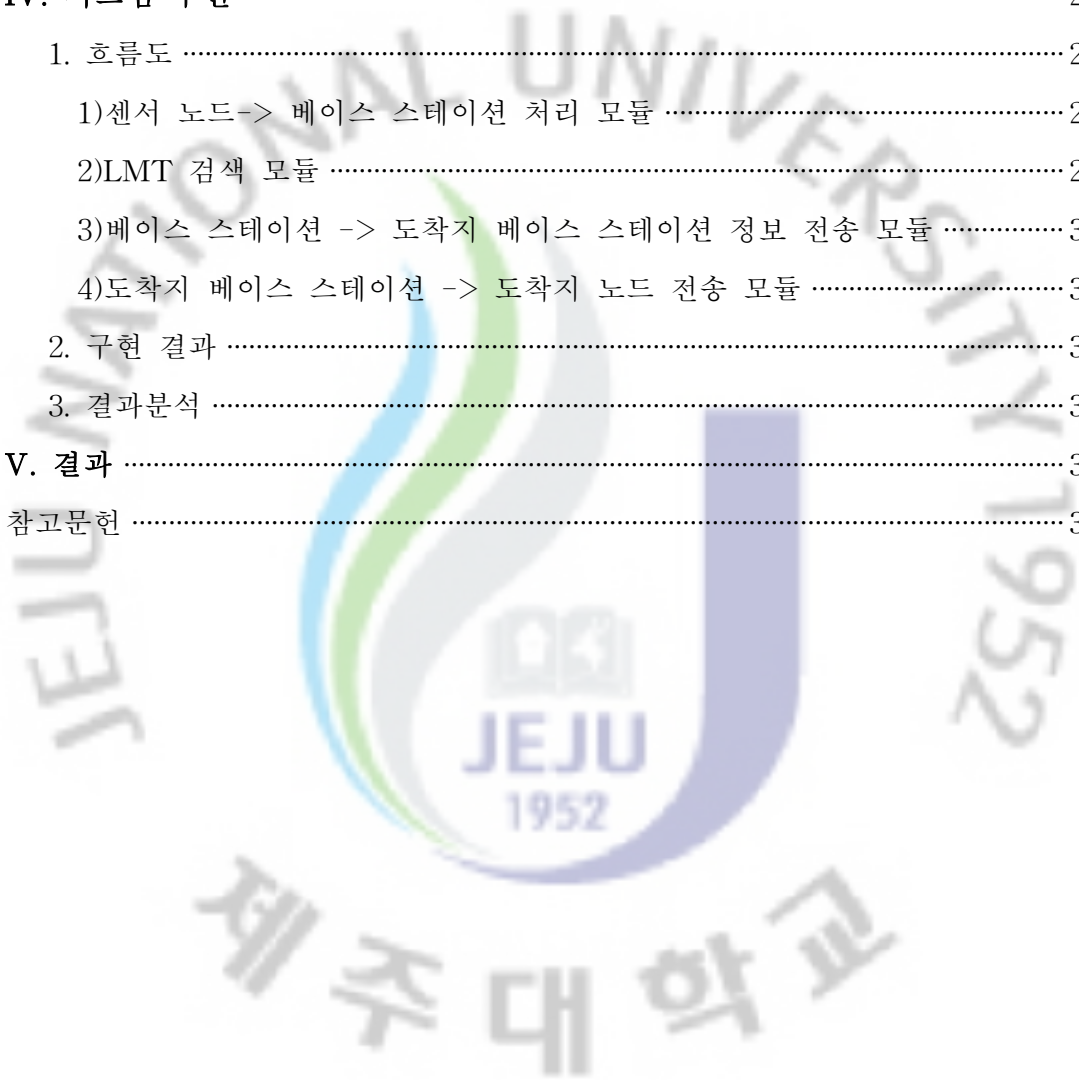


그림 목 차

그림 1 USN 기본 구성	3
그림 2. 컴포넌트 기반의 Tiny-OS 예제	7
그림 3. IP-USN 아키텍처	10
그림 4. Zibgee 프로토콜 스택	12
그림 5. Zigbee 네트워크 구조	12
그림 6. IEEE 802.15.4 MAC 프레임 포맷	14
그림 7. 802.15.4 와 6LowPAN 프레임 포맷	15
그림 8. DTN 기반 연동 방식	17
그림 9. 센서 네트워크에서의 IP 오버레이 기반 연동 방식	18
그림 10. 터널링을 이용한 연동 기법	19
그림 11. 터널링 스택 구조	22
그림 12. 터널링 구현을 위한 TCP 프로토콜 재정의 구조	23
그림 13. Deluge 매커니즘	25
그림 14. 다른 지역간 액츄레이터 제어 시나리오	26
그림 15. Kmote-S1센서	27
그림 16. 제안 연동 방법 흐름도	28
그림 17. 센서 노드 -> 베이스 스테이션 처리 모듈	29
그림 18. LMT 검색을 위한 모듈	30
그림 19. 베이스 스테이션 -> 도착지 베이스 스테이션 정보 전송 모듈	31
그림 20. 도착지 베이스 스테이션 -> 노드 전송 모듈	32
그림 21. Source Network내의 수집된 센서 노드 A의 값	33
그림 22. 호스트와 베이스 스테이션이 TCP 연결을 맺는 화면	33
그림 23. 센서노드가 데이터를 보내고 있는 화면	35

표 목 차

표 1. Zigbee와 6LoWPAN의 비교	16
표 2. 터널링 기반 적용 기법에 따른 비교	20
표 3. [그룹] 필드에 대한 주소 매핑 테이블	24
표 4. 구현 사양 및 요구조건	27
표 5. IP와 TCP의 비교	37



국문 초록

터널링 기법을 적용한 IP-USN 연동 기법 연구

컴퓨터공학과 절자야

지도교수 곽호영

USN(Ubiquitous Sensor Network)은 무선통신 기술을 기반으로 구축된 센서들 간의 자율적인 무선 네트워크를 의미한다. 이러한 USN을 기반으로 사물 및 환경에 관련된 다양한 센싱 데이터들을 전송, 취합, 저장 및 분석하여 사용자의 상황에 맞는 정보를 적절히 제공해 줄 수 있는 응용 서비스의 연구가 활발히 진행되고 있다.

그러나, 이러한 USN은 하나의 그룹을 형성하고, 그룹 안에서의 한정된 네트워크를 기반으로 정보 수집 및 교환이 이루어지고 있다. 이에 서로 다른 그룹들로 구성된 USN망들을 상호 연동하고, 이를 IP(Internet Protocol) 기반의 인터넷으로의 확장이 필요하다.

이러한 확장의 요구로 IP 기반의 인터넷망과 센서 기반의 USN을 서로 연동하기 위한 IP-USN(Internet Protocol-Ubiquitous Sensor Network)에 대한 연구가 활발히 진행되고 있다.

IP-USN은 기존의 IP 인프라를 기반으로 USN에 광범위한 확장성을 제공하고 센서 노드, 게이트웨이 및 싱크 노드의 이동성을 보장하는 USN 서비스이다. IP-USN 기술로 현재 IPv6를 바탕으로 6LoWPAN 이라는 표준화 기술이 개발되었지만 현재 IPv4를 기반으로 하는 인터넷 망에서의 적용이 쉽지 않다.

따라서 본 논문에서는 현재 IPv4 기반의 인터넷과 USN 망과의 연동을 위한 방법으로 터널링 기법을 적용하여 다른 그룹의 USN 망들을 연동할 수 있는 방법을 연구하고, 구현을 통한 실험 및 검증을 하였다.

본 논문의 제안 방법은 USN 데이터를 IP 데이터그램에 캡슐화 시켜서 보냄으로써 가능성을 확인하였는데, 캡슐화 시키는 방법으로는 상대적으로 어플리케이션

들과의 호환성을 높이고 신뢰성 있는 데이터 전송을 위하여 TCP프로토콜의 Destination Port 부분에 데이터를 캡슐화 시켜서 보냄으로써 가능하였다.

또한, 자체의 USN 환경에서는 별도의 TCP/IP 스택이 구현되지 않아도 TOS_Msg 등의 자체 프로토콜을 가지고 통신이 가능하다는 장점이 있다. 이에 따라 센서 노드에서의 TCP/IP 의 프로토콜을 처리하기 위한 오버헤드가 존재하지 않는다.

베이스 스테이션에서의 IP 및 TCP 프로토콜내의 캡슐화 및 역캡슐화, 다른 USN망의 LMT 테이블 정보를 통한 매핑 등의 처리만 있으면, 간단히 IP-USN망 연동을 구현할 수 있다.



ABSTRACT

An implementation of Protocol for IP-USN based on Tunneling

LKHAGVASUREN ZOLZAYA

Department of Computer Engineering

Graduate School

Jeju National University

USN(Ubiquitous Sensor Network) based on wireless communication between deployed sensors and autonomous means a wireless network. the USN is based on a variety of things and the environment related to the transmission of sensing data, collect, store and analyze the user's context-sensitive information that could provide an appropriate service has been actively studied.

But, the USN is the formation of a group, a group based in the network's limited collection and exchange. this consists of different groups that USN and IP-based expansion of the Internet, are required to study.

To studied extensively for the IP-USN(Internet Protocol-Ubiquitous Sensor Network).

6LoWPAN technology based on IP-USN have been for IPv6 standard internets. but is not apply IPv4-based Internet networks.

In this paper, the IPv4-based Internet networks will USN, and as a means of integration by applying a tunneling technique USNs of other groups working on ways to interact, implementation and verification of the experiment was over.

The proposed method of this paper, USN data encapsulated in IP datagrams were confirmed by sending it possible, include encapsulation, a way to improve

compatibility with applications is relatively reliable TCP protocol for data transmission in the data portion of the Destination Port was possible by sending the encapsulation. the USN environment itself, a separate TCP / IP stack does not implement its own protocol, such as TOS_Msg it is possible to communicate.

Accordingly, in sensor nodes, TCP / IP protocol for handling the overhead does not exist. in the base station in the IP and TCP protocol encapsulation and decapsulation. LMT(Location Mapping Table) through the mapping information, simply Interworking with IP-USN can be implemented.



약어표

USN	Ubiquitous sensor network
IP	Internet Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
IPv6	IP Version 6
IPv4	IP Version 4
IP-USN	Internet Protocol-Ubiquitous sensor network
WSN	Wireless Sensor Network
UDP	User Datagram Protocol
ICMP	Internet Control Message Protocol
6LoWPAN	IPv6 Low-power personal area network
NC	Network Coordinator
FFD	Full Function Device
RFD	Reduced Function Device
DTN	Delay Tolerant Networking
PAT	Port Address Translation
NAT	Network Address Translation
SMAC	Smart Media Access Control
OS	Operation System
WPAN	Wireless Personal Area Network
NEST	Network Embedded Software Technology
WG	Working Group
MAC	Media Access Control
PHY	Physical layer
IETF	Internet Engineering Task Force

I/O	Input Output
API	Application programming interface
UC	University of California
CPU	Central processing unit
MANTIS	Multimodal Networks of In-situ Sensors
ANTS	An evolvable Network of Tiny Sensors
FIFO	First In First Out
LMT	Location Mapping Table



I. 서론

USN(Ubiquitous Sensor Network)은 무선통신 기술을 기반으로 구축된 센서들 간의 자율적인 무선 네트워크를 의미한다. 이러한 USN을 기반으로 사물 및 환경에 관련된 다양한 센싱 데이터들을 전송, 취합, 저장 및 분석하여 사용자의 상황에 맞는 정보를 적절히 제공해 줄 수 있는 응용 서비스의 연구가 활발히 진행되고 있다.

USN에 있어 기본이 되는 센서 노드는 범용 목적의 컴퓨팅 시스템이 아니라 몇 가지 센서를 부착하여 환경 정보를 인식하고, 이를 전달하는 목적으로 동작하는 등 매우 제한적인 용도로 주로 활용되어 왔다.

이러한 제한적 용도로 의하여 센서 노드들은 시스템 자원을 최소화하였으며 많은 센서노드들을 설치 운용하는 특성을 지니게 되었다. 이를 자원하기 위한 프로토콜로 TCP/IP 기반의 프로토콜을 사용하기에는 부적절하여 센서 네트워크에 적용 가능한 프로토콜이 요구되었다.

센서 네트워킹 프로토콜은 non-IP 방식이 채택되었으며, Zigbee 프로토콜이 가장 널리 알려진 센서 네트워킹 프로토콜 방식이라 할 수 있다.

그러나 USN은 하나의 그룹을 형성하고, 그룹 안에서의 한정된 네트워크를 기반으로 정보 수집 및 교환이 이루어지고 있다. 이에 서로 다른 그룹들로 구성된 USN망들을 상호 연동하고, 이를 IP(Internet Protocol) 기반의 인터넷으로의 확장이 요구되었다. 그리고 TCP/IP의 모든 기능을 센서 노드에 적용하지 않아도 센서 노드가 통신하는데 지장이 없기 때문에 대표적인 센서노드의 시스템 사양에서 충분히 운용될 수 있는 TCP/IP 스택을 확보할 수 있다.

이러한 이유로 USN에 IP 기술을 접목하려는 연구가 진행되었으며, IP-USN의 개념이 등장하게 되었으며, 이를 이용한 표준화 기술로 6LoWPAN(IPv6 over Low power WPAN)을 들 수 있다.

6LoWPAN은 센서 노드에 IPv6 기반의 IP 주소를 부여하여, IP기반의 인터넷과

의 자연스러운 연동이 가능하도록 하는 기술이다. 즉, USN을 구축하는데 있어 중간지점에 게이트웨이 같은 변환 및 중계 장치가 없어도 통신 개체들 사이에 1:1 직접 통신이 가능하도록 하는 장점을 지니고 있다.

그러나 현재의 인터넷은 대부분이 IPv4기반의 프로토콜을 이용하고 있어, IPv6 기반의 6LoWPAN을 직접 이용하는데 번거로움이 존재한다. 또한 IPv6의 스택 구조가 IPv4보다 크며, 복잡하게 이루어져 있어, 비교적 단순한 센서네트워크를 위한 헤더로 이용하기에는 부적절하다.

따라서 본 논문에서는 현재 IPv4 기반의 인터넷과 USN 망과의 연동을 위한 방법으로 터널링 기법을 적용하여 다른 그룹의 USN 망들을 연동할 수 있는 방법을 연구하고, 구현을 통한 실험 및 검증을 하였다. 그 결과 터널링 기법을 적용한 IP-USN 간의 연동이 이루어짐을 확인할 수 있었다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구들을 설명하고, 3장에서는 터널링 기법을 적용한 IP-USN 연동 방법을 설계하고, 4장에서는 제안 방법에 대한 구현 및 분석을 하며, 5장에서는 결론을 맺는다.

II. 관련 연구

1. USN

USN[1][4][7]은 무선통신 기술을 기반으로 구축된 센서들 간의 자율적인 무선 네트워크를 의미한다. USN의 대표적인 특징은 무인으로 동작되거나 사람이 접근하기 힘든 환경에서 센서 자율적으로 네트워크를 형성하여 데이터를 수집하고, 전송하는데 있다. 이로 인하여 에너지와 데이터 처리 능력의 제한을 갖고 있다. 그림 1.은 USN의 기본 구성을 나타낸다.

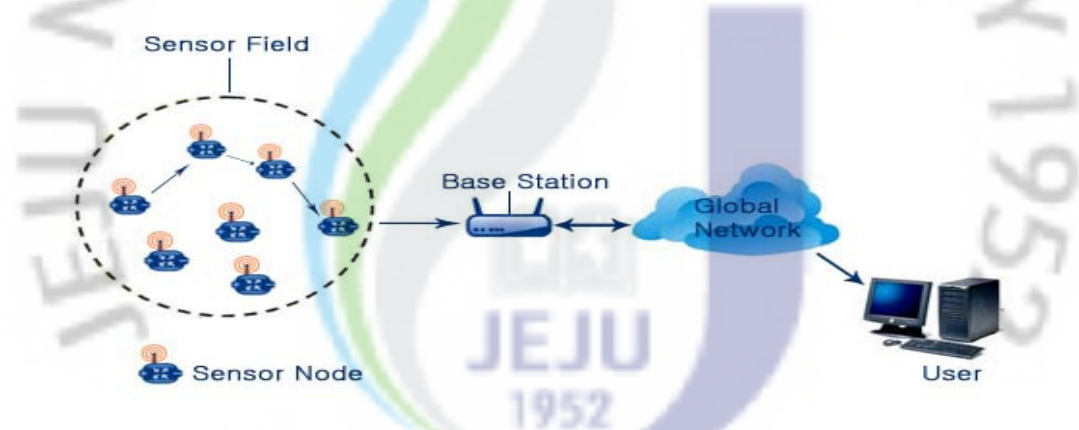


그림 1. USN 기본 구성

센서노드는 사물의 인식정보나 주변의 환경정보(온도, 습도, 오염정보, 균열정보 등)를 감지하고, 감지된 정보를 통합적으로 처리한 데이터를 유무선 통신기술로 전달하는 역할을 한다. 베이스 스테이션은 센서 태그 또는 센서 노드가 외부 네트워크와 통신하기 위해 접속하는 중계 역할을 한다.

센서노드는 센서 네트워크로 전달된 서비스요구 또는 이미 설정한 조건에 따라 생성된 정보를 감지된 초기 데이터 또는 주변 센서 노드 간의 커뮤니케이션에 의해 가공된 형태로 싱크 노드에 전달된다. 베이스 스테이션으로 전달된 정보는 사

용자의 서비스에 대한 응답으로 사용되거나 통계적인 자료로 활용된다.

센서 네트워크는 네트워크를 구성하는 일정 지역에 크기가 1mm³ 정도의 작은 노드들이 수 백 개에서 수천 개까지 설치하여 통신하는 구조를 갖는다[1].

센서 네트워크는 유비쿼터스 환경 구현을 위한 기반 분야로써 현재 유통, 물류, 환경제어, 홈 네트워크, 교통, 국방 등 다양한 형태의 서비스에 활용되고 있다. 이러한 서비스들은 서로 다른 요구사항에 따라 다양한 형태의 개발 및 관리가 이루어지고 있으며, 제한된 네트워크 안에서만 정보 수집 및 분석이 이루어져 서로 다른 네트워크간의 상호 정보공유가 어려운 실정이다.

USN에 대한 네트워크 기술동향 및 운영체제의 동향에 대해 간단히 살펴보고자 한다.

1) 네트워크 기술동향

최근 USN을 위한 프로토콜은 에너지 소모를 줄이는 방법에 대한 여러 가지 연구들을 진행하고 있다. 에너지 소모의 원인은 크게 3가지로 분류할 수 있다.

전송된 패킷이 오염되었을 때 그 패킷은 버려지므로, 재전송에 의한 에너지 소모 및 전송 시간 또한 증가된다. 두 번째 다른 노드 목적지로 전송하는 패킷을 중간에서 엿듣는(Overhearing)것이다. 세 번째 제어 신호 오버헤드이다. 송·수신되는 제어 신호는 에너지 소비가 크다. 마지막으로 비효율적인 원인은 idle listening이다. 전송되지 않는 데이터를 수신하기 위해 듣는 것으로 많은 시간 동안 일어난다. 이러한 에너지 소모를 줄이는 방법으로 데이터 링크계층에서 충돌을 줄이는 것이 기본이라 할 수 있다[1].

IETF를 중심으로 Ad-Hoc 네트워크를 위한 라우팅 프로토콜의 표준화는 상당한 수준으로 발전되었다. 기본적인 특성은 기본이고 이동성 관리와 QoS지원 방안까지 연구되고 있다. 하지만 센서 네트워크의 특징인 에너지 관리와 셀 수 없이 많은 노드 수에 대한 특별한 대책이 없어 전혀 다른 방식의 접근을 시도하는 기법이 발표되고 있다. IEEE 802.15.4와 함께 연구되고 있는 ZigBee Alliance의 라우팅 기법은 기존 라우팅 방식을 뒤엎는 전혀 새로운 기법으로, 많은 수의 노드에 주소를 미리 할당하지 않고 네트워크 구성에 따라 적절히 변화시켜 라우팅을 실시하는 알고

리즘을 적용한다.

또 다른 접근으로 차세대 인터넷 기술인 액티브 네트워크(Active Network)를 적용시키는 방안도 제안되고 있다. 액티브 네트워크는 노드간에 주고받는 패킷에 실행이 가능한 코드를 포함시켜 특별한 용도의 기능을 수행 하도록 하는 기술이다. 센서 네트워크에 적용하기에 오버헤드가 큰 단점이 있지만 사용자의 요구에 능동적으로 대응하여 새로운 서비스를 제공할 수 있는 기술로, 네트워크 구성이 쉽게 바뀌는 센서 네트워크의 특성에 적절하게 적용할 경우 엄청난 시너지효과를 볼 수 있다.

데이터 링크 계층의 프로토콜에 적용될 것으로 인정받는 S-MAC의 주기적인 동작(Periodic Listen/Sleep)은 센서 노드의 에너지 소모를 줄이는 매우 좋은 방식이다. 센서 노드 각각은 전송할 데이터의 양이 매우 적으므로 노드의 transceiver를 계속 켜두고 사용하는 것은 많은 낭비를 가져온다. 특히 데이터의 송수신을 하지 않는 동안 다른 노드비를 통신을 계속 듣고 있는 것은 transceiver를 꺼두는 것에 비해 1000배 정도의 에너지를 더 이용하는 것으로 밝혀져 송수신이 필요 없는 시간동안에는 transceiver를 꺼두는 방식이 큰 효과를 보인다.

라우팅 기법은 센서 네트워크 field의 전체적인 에너지 소모량을 감소시키거나 균등한 에너지 소비를 유도해야 한다. 특히 Sink 노드에 가까운 센서 노드들은 전송할 데이터의 양이 많아 에너지 소모가 심해지는데 이를 막기 위해 노드의 송신 에너지를 더 소모해서 더 먼 거리로 전송하는 방식으로 에너지 소모량을 균등하게 하는 방식도 제안되고 있다. 먼 거리로 전송할 경우 그만큼 적은 수의 노드가 통신하게 되므로 전체적인 에너지 소모량을 균등하게 조절할 수 있다.

또한 센서 네트워크의 특성상 인터넷에 비해 간단한 프로토콜을 요구하고 Sink 노드가 반드시 존재하는 점을 이용하여 네트워크 프로토콜 구조를 바꾸는 방식도 발표되고 있다. 인터넷의 5계층, 7계층 구조와 달리 측정 계층을 제거하거나 2,3계층을 통합하여 하나의 프로토콜로 센서 네트워크에 필요한 만큼의 네트워크 기능을 제공하는 방식이다. 이렇게 함으로써 일부 기능에 제한을 받더라도 기존 방식의 한계를 벗어난 에너지 효율성을 보이기도 한다[1-3].

2) OS 동향

USN 환경에서 센서 노드는 센싱, 프로세싱, 무선통신 기능을 갖추어야 하므로 센서 OS는 이러한 기능을 충분히 지원하면서 소형화 되어야 한다. 센서 노드를 위한 OS는 크기가 작아야 하며, 자원을 효율적으로 관리할 수 있어야 하고, 저전력 소모, 안정적 운영, 네트워크 접속기능, 분산처리, 개발자 및 이용자를 위한 편리한 인터페이스를 제공하여야 한다[1-3].

센서 네트워크용 OS는 저전력 무선 네트워크 지원 및 소형 하드웨어 채택 등을 요구하고 있다. 이를 위하여 설계 시 요구되는 기술 동향은 다음과 같다.

첫째, 센서 네트워크에서는 자원이 제한적이기 때문에 센서 노드의 저전력 통신 특성을 갖추어야 하며, 프로세서의 메모리 영역의 효율적인 관리를 수행하여야 한다.

둘째, 센서 네트워크의 전체적인 에너지 효율을 높이기 위하여 센서 노드들 간의 시간 동기화가 이루어져야 한다.

셋째, 센서 노드는 일반적으로 제한된 처리 능력과 크기가 작은 하드웨어 구조를 가지며, 또한, I/O 장치를 지능형 제어가 아닌 원시적인 접근 방식에 의해 접근하여야 하기에 이러한 I/O의 하드웨어적인 제약 사항들을 고려해야 한다.

넷째, 센서 노드들이 실제 환경에 뿌려진 이후로 유지 보수가 어렵고, 환경의 영향을 많이 받을 수 있기 때문에, 이러한 문제점을 제공하기 위하여, 동적으로 환경에 대응할 수 있는 특성을 지녀야 한다.

다섯째, 센서 네트워크에서의 한 센서 노드의 통신 거리상의 제약으로 인하여, 센서 네트워크 운영체제를 설계할 때는 멀티 홉 라우팅 지원을 위한 프로토콜이 지원되어야 한다.

여섯째, 센서 네트워크 응용 프로그래머들의 손쉬운 프로그래밍을 위하여 용이한 운영 체제의 API를 제공하여야 한다[4].

(1) TinyOS

TinyOS[4-7]는 UC 버클리에서 진행해온 스마트 더스트(smart dust) 프로젝트에 이용하기 위하여 개발된 이벤트 발생 중심의 상태 변화 방식을 채택한 센서 네트워크용 운영체제로써, 동시적인 프로세싱 및 제한된 하드웨어 메모리 공간에서

의 효율적인 성능을 지원해주는 운영체제이다.

TinyOS의 특징은 다음과 같다. 상태 머신의 기반의 구조를 가지는 운영체제로, 각각의 상태는 컴포넌트가 해당된다. 응용 프로그램은 그림 2.에서처럼 컴포넌트 구현에 독립적인 연결 방법을 사용하여 각각의 TinyOS의 컴포넌트를 연결한다. 그리고 각 컴포넌트에 명령이 내려지고 이 명령을 처리하는 이벤트 처리기는 그 명령에 따른 상태변화를 신속하게 일으켜, 필요로 하는 일을 수행하는 특징을 가진다. 다음으로는 센서 네트워크의 특징인 저전력 파워 소비를 지원하기 위하여 센서 노드들의 일이 요구되지 않을 경우 저전력 모드인 슬립 모드로 전환함으로써 효율적인 CPU의 사용을 이룰 수가 있는 것이다.

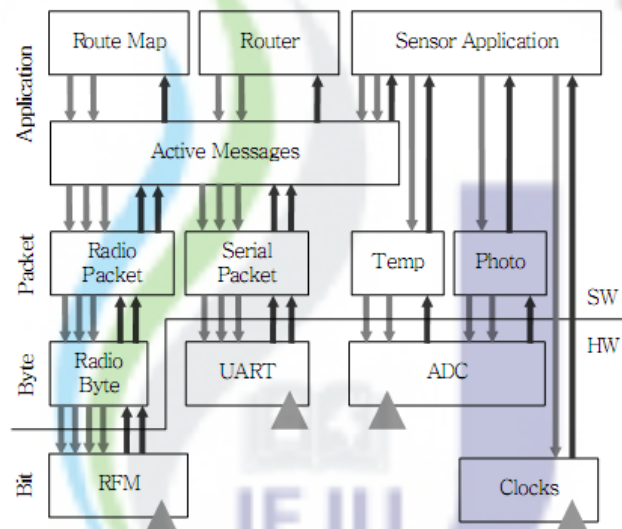


그림 2. 컴포넌트 기반의 Tiny-OS 예제

또한, nesC라는 동적 메모리를 할당하지 않는 정적인 언어를 통하여 센서 네트워크용 응용프로그램을 작성할 수가 있으며, 소프트웨어가 필요로 하는 동작의 효율적인 코드를 만든다. nesC는 안정성을 위하여 전체 프로그램에 대한 분석을 통해 최적화를 수행하고, 컴포넌트 기반의 언어이며, 직접적으로 TinyOS의 이벤트 기반의 동시성 모델을 지원한다.

즉 이 언어를 이용하여 메모리 반도체 칩들과 메모리 컨트롤러를 정해진 컴포넌트와 양방향으로 갖는 인터페이스를 이용하여 프로그램 메모리 모듈을 만든다.

끝으로 TinyOS는 센서 네트워크의 필수적인 요소라 할 수 있는 멀티 홉 라우팅

기술을 제공함으로써 거리 제한을 해결하였으며, 여러 가지 그 외에 필요한 엔진인 TinyDB, 그리고 보안을 위하여 Tiny-Sec과 같은 모듈을 지원한다[4-6].

(2) SOS

SOS는 Mote 계열 기반의 센서 네트워크를 지원하기 위한 UCLA에서 개발된 운영체제이다. SOS는 메시지 패싱, 동적 메모리 할당, 모듈을 자율적인 적재와 제거를 지원하는 공동 커널(common kernel)을 지원하며, 동적 재구성이 하나의 목표이다. 첫번째 특징은 동적 재구성 기능을 지원한다는 것으로 이 기술은 센서 네트워크에서의 각각 개개의 센서노드들의 소프트웨어를 무선 네트워크를 통하여 수정하고 업데이트 하는 기능이다. 이것은 센서 네트워크의 새로운 모듈 업데이트가 필요할 때, 사용될 수 있다. 두번째 특징은 센서 네트워크 프로그래머들이 필요로 하는 서비스에 맞게 응용프로그램을 제작할 수 있도록 하는 데 목적이 있다. 이를 위하여 OS는 전통적인 운영체제의 기술을 센서 네트워크에서의 자원 제약 특성을 만족하기 위하여 설계되었으며, 그 결과로 유연한 센서 네트워크 시스템 구축을 지원하게 해준다. 세번째 특징은 응용 애플리케이션은 하나 이상의 모듈로 구성되어 있으며 이러한 모듈들은 비동기 메시지 및 함수 호출을 통하여 서로 동작하게 된다 [4].

(3) MANTIS

MANTIS는 콜로라도 대학에서 개발된 멀티 스레드를 지원하는 센서 네트워크용 임베디드 운영체제이다. 특징은 초소형 스레드에 기반한 멀티 스레드 구조를 채택하여 TinyOS와는 달리 일반 프로그래머들이 익숙한 구조에 의하여 프로그래밍이 가능하도록 하였다. 특징은 다음과 같다. MANTIS는 레이어 기반의 운영체제 기반이며, 이것은 멀티 스레딩, preemptive 스케줄링 기법, 그리고 mutual exclusion을 통한 I/O 동기화, 그리고 하드웨어를 추상화시키는 디바이스 드라이버의 특징을 가진다. 또한 애플리케이션 프로그래머가 프로그래밍을 손쉽게 할수록 C언어 방식의 API를 지원하며, 센서 네트워크에서의 멀티 홉 통신을 위한 네트워크 스택을 지원한다. 그리고 리모트 셸 기능을 통하여 운영체제를 컨트롤하며, 프로그래밍의 재설정 또한 가능하다[4].

(4) ANTS

ANTS[8]는 ICU(한국정보통신대학교)에서 센서 네트워크 환경의 쉬운 구성을 위해 개발된 운영체제이다. 특징으로는 유연한 멀티 스레딩 엔진을 탑재하였고 충분한 API를 통한 손쉬운 프로그래밍이 가능하다[4].

ANTS는 시스템의 실행 시간 중 소프트웨어의 재구성과 진화에 초점을 맞추고 있다. 진화형 센서 네트워크의 개념은 미래의 센서 네트워크가 어떤 용도로 구성되는가에 따라 그 구조가 다를 수 있으며, 상황이나 환경 변화에 효율적으로 대처해야 한다는 요구사항에서 대두되었으며, ANTS는 센서 네트워크 응용이 실행환경에 필요한 구성요소들을 자동으로 디자인하여 소프트웨어의 구조를 재구성할 수 있다[7].

(5) 나노 Qplus

나노 Qplus 운영체제는 한국전자통신연구원(ETRI)에서 개발된 센서 네트워크용 운영체제로써 다음의 특징을 가진다. 첫째, 에너지 소모를 최소화하기 위하여, 센서 네트워크를 구성하는 노드들 간의 시간 동기화 기법을 제공하며, 또한, 슬립 모드와 활성 모드를 반복함으로써 저전력 파워 소비를 지원한다. 둘째, 제한된 메모리의 사용을 최소화하기 위하여 멀티 스레드 간의 스택을 공유한다. 셋째, 멀티 스레드 스케줄러 방식(FIFO, 시간 기반 라운드 로빈 방식, 우선 순위에 따른 스레드 선점형 방식)으로, 실시간 운영체제의 특성을 지원한다. 넷째, 응용프로그래머들이 손쉽게 센서 프로그램을 개발할 수 있게 C 기반의 프로그램 작성 기법을 제공한다[4].

2. IP-USN

앞 절에서 설명된 것과 같이 USN은 하나의 그룹을 형성하고, 그룹 안에서의 한정된 네트워크를 기반으로 정보 수집 및 교환이 이루어지고 있다. 이에 서로 다른 그룹들로 구성된 USN망들을 상호 연동하고, 이를 IP(Internet Protocol) 기반의 인

터넷으로의 확장이 필요하다. 그리고 TCP/IP의 모든 기능을 센서 노드에 적용하지 않아도 센서 노드가 통신하는데 지장이 없기 때문에 대표적인 센서노드의 시스템 사양에서 충분히 운용될 수 있는 TCP/IP 스택을 확보할 수 있다.

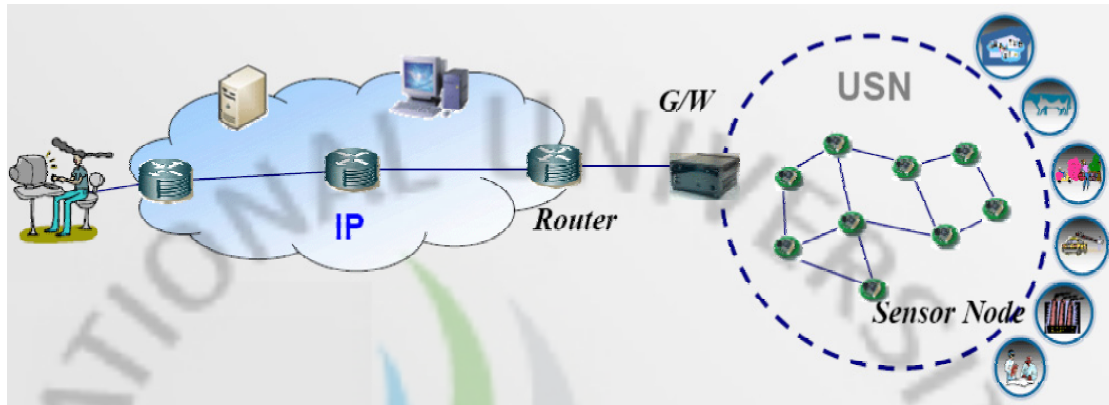


그림 3. IP-USN 아키텍처

이러한 이유로 USN에 IP 기술을 접목하려는 연구가 진행되었으며, IP-USN의 개념이 등장하게 되었다. IP를 사용함으로써, 기 구축된 통신 및 응용서비스의 인프라를 활용할 수 있어 비용의 절감 및 기 검증된 IP기술을 이용하여 신뢰성과 안정성을 보장할 수 있다. 또한, IP-USN[9-11]을 통하여 다른 네트워크에 있는 센서와 센서간 혹은 센서와 호스트간의 End-to-End[9] 커뮤니케이션이 가능하다. 그림 3.은 IP-USN의 아키텍처를 나타낸다.

기존의 센서 네트워크와 비교되는 점은 멀티라우터가 가능하여 센서 노드의 데이터가 베이스 스테이션 또는 게이트웨이로 집중되어 발생하는 데이터 트래픽 부하 및 베이스 스테이션과 게이트웨이의 고장에 따른 문제점이 해결된다. 또한 멀티라우터의 지원은 대규모, 확장성, 신뢰성이 보장되어 도시기반의 대규모 인프라가 가능해짐을 의미한다.

3. 관련 네트워크 프로토콜 기술

1) Zigbee

ZigBee Alliance[11][12]는 2002년 8월에 신뢰성 및 기기간의 상호 연동을 보장하며, 저비용으로 저전력을 소모하는 무선통신의 구체적인 활용 및 응용을 목적으로 세계적인 산업체 간에 결성된 비영리 협력 모임이다.

특히, 유수의 반도체 제조 업체 및 기술 파급의 역할이 큰 기관들이 참여하여 현재 급속한 성장을 보이고 있으며, 2003년 IEEE 802.15.4의 제정과 함께 IEEE 802.15.4를 ZigBee Alliance의 표준으로 승인하였다. PHY 및 MAC 계층에 대한 표준을 기반으로 MAC 계층 이상의 프로토콜 스택 및 응용 영역에 대하여 표준화를 추진하는 동시에 관련 칩 생산 및 제품화에 박차를 가하면서 더욱 활기를 띠고 있다.

그림 4.는 ZigBee의 프로토콜 스택을 나타내었다. ZigBee의 통신 모드가 마스터-슬레이브 방식을 기본으로 하고 있지만 'Mesh Mode'라 불리는 점대점 방식의 네트워킹이 가능하고 네트워크 안에서 하나의 기기를 네트워크 조정자(Network Coordinator)로 명하여 다른 기기들을 제어하는 일련의 방식을 채택하였다. 또한 이러한 네트워크 조정자 들간의 통신이 가능하며 특정의 기기가 Mesh 모드의 네트워크 상의 다른 모든 기기를 인식하지 못할 때에도 네트워크를 스스로 구성할 수 있다[13].

Zigbee 프로토콜은 IEEE 802.15.4를 바탕으로 Zigbee Alliance[12-14]에서 상위 네트워크, 보안, 어플리케이션의 표준을 제정하고 있다. Zigbee의 특징은 원격감지, 감시, 제어, 모니터링의 응용과 가격을 최대한 낮추고 전력소모를 최소화하는 방향으로 설계되었다.[15]

WPAN[16](Wireless Personal Area Network) 기술을 정의한 IEEE 802.15.4 표준을 기반으로 하는 Zigbee는 저속 근거리 무선통신 기술로 2.4GHz 대역과 868/915MHz 대역을 사용하는 2개의 주파수 할당대역에서 동작한다.

그리고 저전력의 Zigbee 송수신기를 빛이나 기온, 압력과 같은 다양한 기능의 센

서들과 결합하여 보다 큰 규모의 센서네트워크를 구성할 수 있는 기술이다.

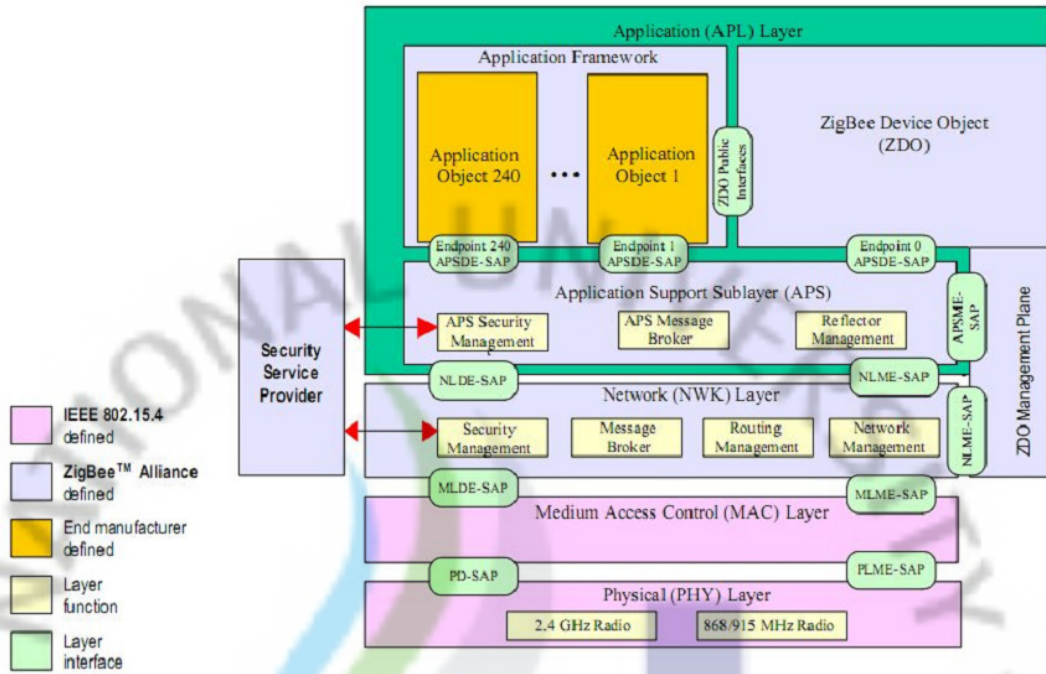


그림 4. Zigbee 프로토콜 스택

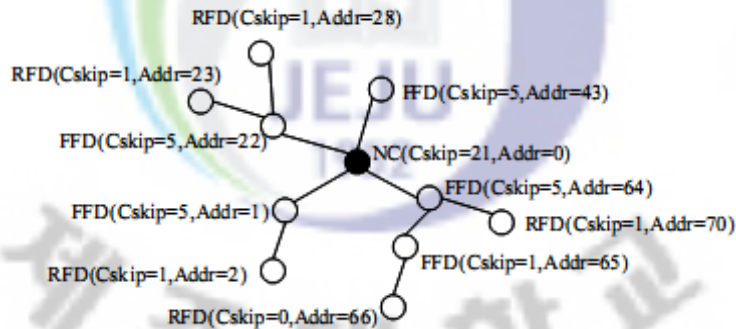


그림 5. Zigbee 네트워크 구조

Zigbee 기반의 센서네트워크 특징은 전력소모를 최소화시켜서 전력 효율성을 향상시킨 것으로 이를 지원하는 Zigbee 디바이스는 NC(Network Coordinator), FFD(Full Function Device)와 RFD(Reduced Function Device)로 구성된다. NC는 대용량 메모리와 컴퓨팅 파워를 가지며 네트워크 내부의 센서디바이스들을 제어

및 관리하고 초기화 기능을 수행한다. FFD는 센서의 일반적인 센싱 기능뿐만 아니라 센서네트워크에서 NC와의 통신을 담당하는 라우터 기능도 수행할 수 있다. RFD는 FFD보다 제한된 기능을 가지며 주로 네트워크의 종단센서를 구성하는데 사용된다. 다양한 센서디바이스로 구성되는 센서네트워크는 일정 지역을 중심으로 클러스터를 형성하고 클러스터내의 대표 센서가 NC와 통신을 할 수 있는 클러스터 트리 형태의 토폴로지를 형성한다[16].

그림 5.는 Zigbee 프로토콜을 이용한 센서 네트워크의 구성시 생성되는 트리 토폴로지를 나타내며, 이러한 토폴로지를 바탕으로 각 센서 노드를 식별할 수 있는 ID를 부여받아 센서 노드간 통신이 이루어진다. 따라서 Zigbee 프로토콜 기반의 USN 내에서의 센서 식별은 고유 ID 값으로 식별이 이루어짐을 알 수 있다.

2) 6LoWPAN

IETF 6LoWPAN WG는 L2 Layer에 IEEE 802.15.4를 기반으로 하는 센서 네트워크에 IPv6를 지원하기 위한 이슈를 다루는 그룹으로서 저전력, 20~250kbps의 데이터 전송률, 900~2400MHz의 주파수 대역에서 최소형 메모리와 최소형 프로세서만을 장착한 센서 응용을 대상으로 하면서 IP기반의 센서 네트워킹에 대한 표준 기술을 다루고 있는 그룹으로 LoWPAN 상에서 IPv6 패킷 전송 방안을 정의하는 것을 목표로 하고 있다[17][18].

6LoWPAN[19][20]는 IETF 인터넷 영역(Area)의 LoWPAN 워킹그룹(WG)에서 표준화되고 있는 기술로서 LoWPAN(Low-power Wireless Personal Area Networks), 즉 IEEE 802.15.4 표준의 MAC 및 PHY 계층의 상위 계층에 IP 계층을 올려 센서 네트워크상에서 IPv6 패킷을 전송하고자 하는 기술이다. 2004년 11월 61차 IETF 회의에서 BOF가 열렸으며 2005년 3월부터 정식 워킹그룹으로 시작하였다. 6LoWPAN 기술은 기존의 IP 인프라를 재사용할 수 있다는 장점과 더불어 IPv6의 특징으로 들 수 있는 많은 수의 주소 공간과 자동 주소 할당 등의 기능이 센서 네트워크에 특징에 부합하는 점 등으로 인해 가능성을 인정받고 있다[19].

6LoWPAN 기술은 어떻게 하면 데이터 전송 속도가 느린 IEEE 802.15.4 (250kbps/ 2.4GHz, 40bps/915MHz, 20kbps/868MHz) 기술을 통해 헤더 사이즈가

큰 IPv6 패킷을 효율적이고 안전하게 전달하고, 또 전달하고자 하는 장치들을 어떻게 검색할 것인지에 대한 내용을 연구하는 기술이다[17].

LoWPAN은 제한된 전력 하에서의 저비용 무선 통신을 위한 네트워크로서 IEEE 802.15.4 표준을 따른다. 그림 6.은 IEEE 802.15.4의 MAC 프레임구조를 나타내었으며, LoWPAN의 주요한 특징[19][20]은 다음과 같다.

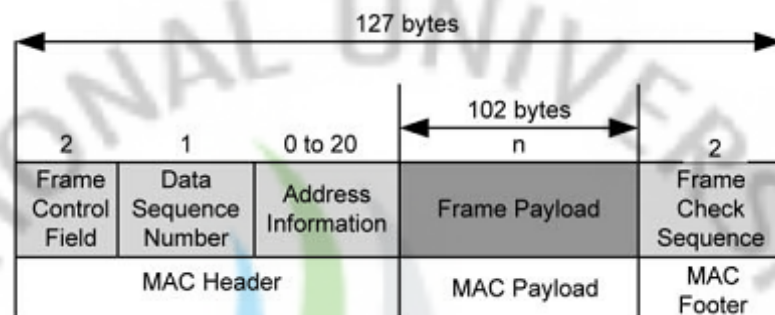


그림 6. IEEE 802.15.4 MAC 프레임 포맷

1. LoWPAN의 최대 패킷 크기는 127 바이트에 불과하다. 그림 5에서 보듯 이중 PHY 및 MAC에서 사용하는 부분을 제외하면 약 102 바이트 정도만이 상위 계층에 의해 사용될 수 있다. 또한 링크 계층 보안 정보가 포함되면 사용가능한 크기는 최하 81 바이트까지 줄어들 수 있다. IPv6 패킷의 크기가 40 바이트인 것을 고려해보면 상당히 작은 크기인 것을 알 수 있다.
2. 낮은 대역폭 및 저전력, 저비용의 특징을 가진다. 2.4 GHz 대역에서 약 250 kbps 정도의 대역폭을 가지며, 장치는 주로 센서 또는 스위치 등에 부착되어 낮은 처리 속도 및 적은 메모리를 가질 것이다. 또한 몇몇 장치는 배터리 모드로 동작할 수도 있다. 이 경우 전력 소비를 줄이기 위해 장치가 주기적으로 슬립 모드에 들어감에 따라 통신이 불가능한 시간이 발생할 수 있다.
3. 16비트의 짧은 주소와 IEEE 64비트 주소를 둘 다 지원한다.
4. 토폴로지는 스타, 클러스터-트리 그리고 메시 형태를 구성할 수 있다.

5. 저가격, 소형의 특징을 감안할 때 LoWPAN에서는 PC나 기존의 무선 네트워크에 비해 많은 수의 장치들이 배치될 가능성을 고려해야한다.
6. 장치는 배터리의 소모, 물리적 훼손, 무선 환경의 불안정 등으로 신뢰성이 떨어질 수 있다.

6LoWPAN은 IPv6를 기반으로 네트워크 계층, 전송 계층, 어플리케이션 계층을 End-to-End로 연결할 수 있다. 따라서 게이트웨이에서 별도의 라우팅이 필요하지 않으며, Zigbee에 비해 데이터전송 지연이 적다는 장점을 가진다. 또한 센서 노드를 식별하기 위해 IP를 사용함으로써 인터넷과 같은 기존 인프라를 그대로 활용할 수 있기 때문에 비용절감의 효과 및 안전성 및 신뢰성을 보장할 수 있다.

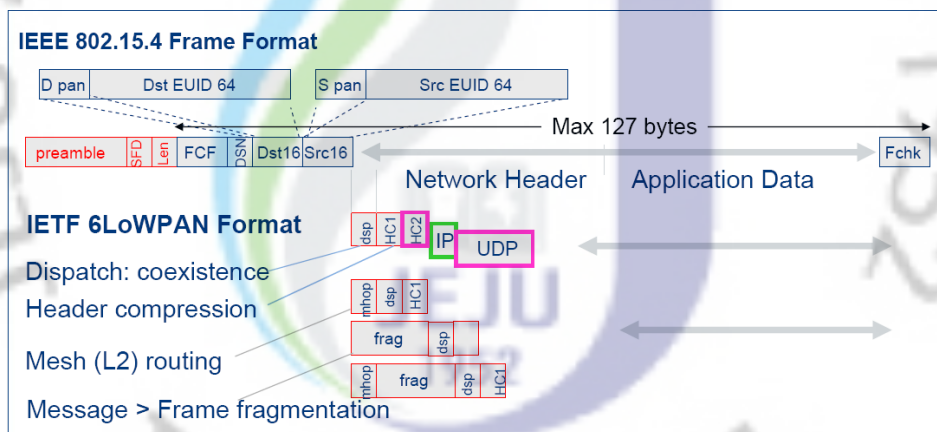


그림 7. 802.15.4 와 6LoWPAN 프레임 포맷

그림 7.은 6LoWPAN의 패킷 단축에 따른 그림을 나타내고 있다. PHY/MAC이 사용하는 최대정보를 고려하면, 80Byte안에 IPv6/TCP 또는 IPv6/UDP 헤더 및 센서 응용을 위한 데이터가 표현되어야 한다. 이를 위하여 6LoWPAN에서는 dispatch를 사용하여 6LoWPAN 패킷임을 표현하고, IPv6/UDP 헤더 압축방법을 제시하고 있다[17].

그러나 이러한 6LoWPAN은 최악의 경우 IEEE 802.15.4 프레임을 통한 IP 패킷

전송할 때 최대 가용 크기는 81Byte며, IPv6 옵션 헤더를 제외한 IPv6헤더 40Byte를 적용하면 UDP나 TCP와 같은 상위 계층 프로토콜에게는 41Byte만 남는다. UDP의 8Byte헤더 또는 TCP의 20Byte 헤더를 적용하면 UDP와 TCP를 위한 데이터 전송 공간은 각각 33Byte와 21Byte만 남게 된다[17]. 추가로 단편화와 재조립을 위한 계층에서 사용될 공간까지 생각한다면 응용 데이터 전송을 위해서는 매우 적은 공간만이 남게 된다. 따라서 이러한 프로토콜 형태로 사용한다면 데이터가 수십 Byte에 불과하더라도 과도한 단편화와 재조립이 일어나게 되는 문제점이 있다.

표 1은 Zigbee와 6LoWPAN의 비교를 나타낸 것이다.

표 1. Zigbee와 6LoWPAN의 비교

	Zigbee	6LoWPAN
PHY/MAC	IEEE 802.15.4	IEEE 802.15.4
프로토콜 스택범위	전체	Adaptation 계층
네트워크 주소범위	16비트, 하나의 PAN만 적용	128비트, 전체적 적용가능
게이트웨이 기능범위	응용계층	네트워크 계층
서비스 발견	로컬	글로벌
통합물관리	어려움	용이함

4. IP-USN 연동 기술

1) 게이트웨이 기반 연동 기법

게이트웨이를 이용하는 연동기법의 경우 응용 계층에서의 게이트웨이 기법과 DTN(Delay tolerant Networking)[21][22] 게이트웨이를 이용하는 기법으로 다시 분류할 수 있다[24].

응용 계층에서의 게이트웨이 기법은 센서 네트워크와 인터넷의 연동을 위해 응용 계층에서 두 네트워크의 특성에 맞도록 프로토콜을 변화시켜주는 게이트웨이를 의미하며, DTN은 그림 8에서 나타난 것과 같이 동질한 형태로 통신이 이루어지는 지역 네트워크 간의 응용 계층 게이트웨이 기법을 이용하여 서로 망을 연동하는 방법을 의미한다.

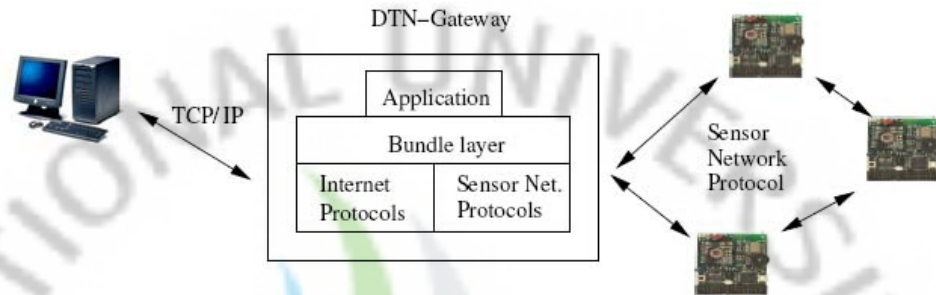


그림 8. DTN 기반 연동 방식

이러한 연동 기법들은 구조적으로 제안 및 소개되고 있는 실정이며, 설계 및 구현들은 미비한 실정이다.

2) 오버레이 연동 기법

센서네트워크 또는 인터넷상의 오버레이 네트워크를 구성하는 기법의 경우 센서 네트워크 위에 가상 IP 네트워크를 구현하는 방법과 인터넷 위에 가상 센서 네트워크를 구현하는 방법으로 분류할 수 있다.

IP 오버레이 네트워크 기법은 센서네트워크 상의 센서 노드들에게 IP 프로토콜 스택을 구현하고 IP주소를 부여하여 IP 패킷을 통해 데이터를 주고받기 위한 오버레이 계층을 구성하는 방법이며, 오버레이 센서네트워크 기법은 인터넷에 위치한 호스트에 센서 네트워크와 연동할 수 있는 오버레이 센서 네트워크를 구성하는 방법이다[24][25].

센서 노드에 IP 주소를 할당하는 것과 IP의 주소 중심적(address-centric) 라우팅 및 센서 네트워크의 데이터 중심적(data-centric) 라우팅 방식 간의 차이 문제를 해결하기 위해 클러스터 헤드들을 선별하고 이들 간의 터널을 구축해서 IP 기반 오

버레이 방식으로 동작하는 방법의 제안되었으나, 클러스터 헤드를 제외한 나머지 센서 노드들은 IP 기반으로 동작하지 않기 때문에 클러스터 헤드가 결국은 게이트웨이 역할을 가지고 있어야 한다[21][23].

그림 9.는 센서 네트워크에서의 IP 오버레이 기반 연동 방식을 나타낸다.

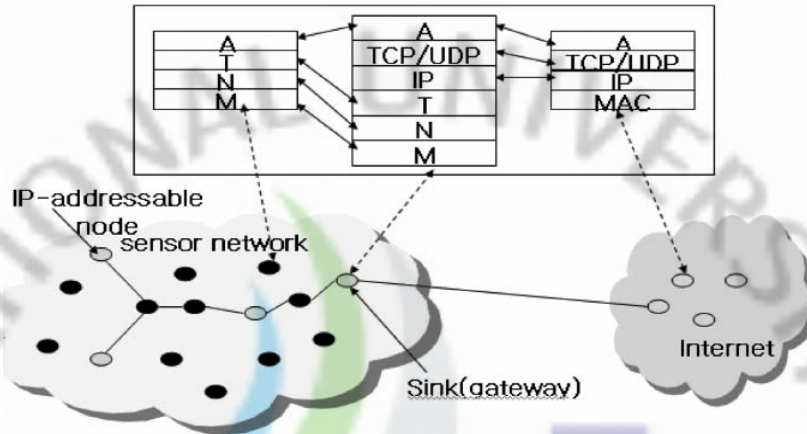


그림 9. 센서 네트워크에서의 IP 오버레이 기반 연동 방식

3) 터널링 연동 기법

터널링은 인터넷을 사적이며 안전한 네트워크의 일부로서 사용하는 것으로서, 한 네트워크에서 다른 네트워크의 접속을 거쳐 데이터를 보낼 수 있도록 하는 기술을 말한다.

터널링을 이용한 연동 기법은 센서 네트워크와 터널링을 위한 주소변환기(Translator)가 존재하게 된다. 이때 주소 변환기(Translator)간에는 Zigbee기반의 통신으로 이루어지며, 주소변환기와 인터넷 간에는 IP기반이다. 또한 주소 변환기는 센서 네트워크 인터페이스와 인터넷 인터페이스를 가진다[24].

(1) 정적 및 동적 연동 기법

주소 연동을 위해 제안된 주소 변환기는 Zigbee ID와 자신이 보유하고 있는 인터넷 IP 주소 사이에서 주소를 응용 계층상에서 상호 연동하는 기능을 제공한다. 따라서 센서네트워크내의 센서는 주소 변환기의 기능을 통해 인터넷과 통신을 할

수 있게 된다.

주소 변환기의 기본 기능은 센서 네트워크내의 센서가 센싱한 정보를 인터넷으로 전송해주는 역할이며, 추가적으로 TCP/IP 패킷을 해석하고 TCP/IP 프로토콜의 필드들을 변경할 수 있어야 하며 그 밖의 필요한 요구사항들을 NAT(Network Address Translation)처럼 가질 수 있다[24][26-28].



그림 10. 터널링을 이용한 연동 기법

그림 10.은 터널링을 이용한 연동 기법을 그림으로 나타내었으며, 이러한 주소 변환기를 이용한 연동 기법으로 Zigbee기반의 센서 ID와 인터넷 IP주소가 매번 동일하도록 하는 정적 연동 기법과 연결 설정이 발생할 때마다 주소 풀에서 유효한 IP 주소를 배정하는 동적 연동 기법을 고려할 수 있다. 즉 정적 연동 기법은 센서 네트워크와 인터넷간의 1:1 주소 매핑을 매번 동일한 주소로 수행한다.

(2) PAT 연동 기법

PAT은 NAT 기법의 NAT Pool에 할당할 수 있는 인터넷 IP 주소가 한 개인 경우에 인터넷 IP 주소를 항상 일정하게 할당되고 TCP/UDP의 포트 번호를 새로 할당하는 기능을 갖는다. 즉 전달되는 패킷의 Payload에 사설망의 호스트 정보인 IP 주소와 TCP/UDP 포트 정보가 전달되는 경우에 각각의 프로토콜에 따라서 할당된 IP 주소나 TCP/UDP 포트 정보를 변경하여 최소의 인터넷 IP 주소로 사설망에 인터넷 접속을 가능하게 한다.

따라서 PAT 기반 연동 기법은 Translator가 센서네트워크내의 센서디바이스 수 만큼 IP 주소를 확보하지 않아도 센서네트워크와 Internet간을 TCP 또는 UDP 포

트를 사용하여 연동시킬 수 있다. 또한 이 기법은 센서디바이스들에게 미리 포트번호를 할당시키는 정적 처리와 상황에 따라 할당시키는 동적 처리가 가능하다.

포트번호를 정의하기 위해 현재 사용할 수 있는 포트 번호의 종류, 즉 포트 번호는 Well Know Ports(0-1023), Registered Ports(1024-49151), Dynamic and/or Private Ports(49152-65535)가 있으므로 중복되지 않은 새로운 포트번호 부여 체계가 요구된다. 따라서 시스템이 사용할 수 있는 포트 번호의 최대값을 구하고 여기에 Offset(센서 노드 Addr)값을 더한다. 또한 센서디바이스의 Offset값으로 “0”을 부여할 수 있어서 사용할 포트번호가 “65535”가 될 수 있고 이는 시스템 포트번호와 구분할 수 없게 되는데 이러한 중복을 피하기 위한 추가로 “1”을 더하여 사용할 포트번호를 구하는 방법을 적용한다. 표 2.는 터널링 연동 기법에서 사용되는 세부 기법들 중 NAT 기반과 PAT기반 연동 기법의 차이를 나타내고 있다.

표 2. 터널링 기반 적용 기법에 따른 비교

기 준	NAT 기반	PAT 기반
복잡도	단순함	복잡함
주소 확보	한 개 이상	한 개 (Overloading)
Offset(port)	필요치 않음	필요함
서비스 방법	정적, 동적	동적
Pool 관리	필요함	필요치 않음
토폴로지 변경	불가능	불가능
End-to-End	가능	불가능
주소 매핑	일대 일	일대 다

III. 제안 시스템 설계

1. IP-USN을 위한 연동 설계

앞 장의 관련연구에서 기술한 것 같이, 터널링은 인터넷을 사적이며 안전한 네트워크의 일부로서 사용하는 것으로서, 한 네트워크에서 다른 네트워크의 접속을 거쳐 데이터를 보낼 수 있도록 하는 기술을 말한다.

기존의 터널링 기법은 단순히 USN 망의 데이터를 IP 망으로 나눠 보내주기 위한 방안으로 제시되었으며, 하나의 망을 대상으로 연구가 진행되었다. 또한 IP-USN에서의 연동 기술은 노드 자체에 IPv6 스택을 적용함으로써, 센서 노드의 부하를 초래하여, 노드 자체의 성능 향상이 뒷받침되어야 한다는 전제 조건이 요구되었다.

USN내의 노드들로부터 수집된 정보는 동일 네트워크에서는 자유롭게 데이터의 수집 및 분석이 가능하나, 다른 네트워크로 전달되기 위해서는 터널링 기법을 통해 IP기반의 인터넷망을 통해서 전달을 할 수 있다. 따라서 본 제안 논문에서는 터널링 기법을 이용하여 서로 다른 USN의 데이터 전송을 위해 IP 망과의 연동 및 이를 위한 기법을 설계하고, 구현하였다.

그림 11.은 USN과 IP망 사이에 서로 다른 스택을 지원하기 위한 터널링 스택 구조를 나타낸다.

그림 11.과 같은 스택 구조 지원하는 터널링을 구축한 베이스 스테이션을 이용하여, 해당 베이스 스테이션에서 바로 다른 USN의 베이스 스테이션으로 데이터를 전송하게 된다. 이때, USN 내부의 패킷이 외부의 망으로 나갈 때에는 베이스 스테이션에서 TCP/IP 패킷 안에 캡슐화 하여 보내게 되는데 중착점 호스트에서는 TCP/IP 패킷을 역캡슐화 시켜서 노드정보에 대한 패킷을 추출하게 된다. 반면 패킷을 받게 될 때에는 노드(Node) 정보를 싱크 노드에서 노드에 대한 정보를

Multiple TCP/IP Stack을 이용하여 구현 한 뒤에 베이스 스테이션에서 받은 패킷을 해당 노드로 포워딩(forwarding) 시켜준다. 이를 위해 베이스 스테이션은 도착지에 대한 IP 주소 및 해당 노드 ID에 대한 정보를 이용하여야 한다. 이를 위해 싱크 노드에는 도착지에 대한 IP와 해당 센서 노드들에 대한 정보를 테이블로 지니게 된다.

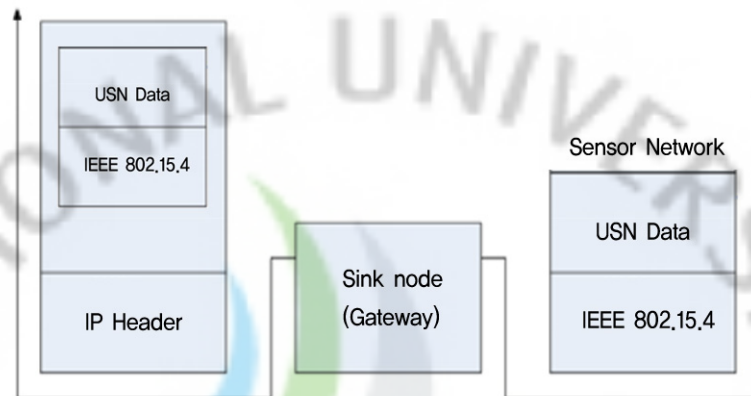


그림 11. 터널링 스택 구조

1) 터널링을 위한 IP 프로토콜 재정의

서로 다른 USN과 IP 망 사이의 데이터를 주고 받기 위해서는 데이터 패킷의 새로운 정의가 요구된다. 하지만, 본 논문에서는 기존의 망과의 호환성을 유지하기 위해서 TCP 프로토콜 내 필드 역할을 재정의 하여 사용하였다.

그림 12.에 나타낸 것과 같이 터널링을 위하여 TCP 프로토콜을 재 정의하여 구현하고자 한다. 재 정의된 부분은 USN 내에서의 센서 노드를 식별하기 위한 센서 ID와 서로 다른 망을 구별하기 위한 그룹 ID를 별도의 구조체로 재 정의하였다.

또한, 변경된 내용은 기존의 TOS 메시지 구조를 기반으로 이루어져 기존의 TOS 메시지 기반의 데이터를 그대로 활용할 수 있는 장점이 있다.

```

typedef struct _dst_node { /* 도착지 노드에 대한 정보를 표현하기 위해
                           재정의하는 destnode 구조체 */

    u_int8_t dst_group;
    u_int8_t dst_node;
} destnode;

struct tcphdr
{
    u_int16_t th_sport; /* source port */
    /* u_int16_t th_dport; destination port ; 원래의 TCP 헤더 structure */
    destnode dnode; /* 재정의된 부분 ; 노드 id와 그룹 id를 표현 */
    tcp_seq th_seq; /* sequence number */
    tcp_seq th_ack; /* acknowledgement number */
# if __BYTE_ORDER == __LITTLE_ENDIAN
    u_int8_t th_x2:4; /* (unused) */
    u_int8_t th_off:4; /* data offset */
# endif
# if __BYTE_ORDER == __BIG_ENDIAN
    u_int8_t th_off:4; /* data offset */
    u_int8_t th_x2:4; /* (unused) */
# endif
    u_int8_t th_flags;
# define TH_FIN 0x01
# define TH_SYN 0x02
# define TH_RST 0x04
# define TH_PUSH 0x08
# define TH_ACK 0x10
# define TH_URG 0x20
    u_int16_t th_win; /* window */
    u_int16_t th_sum; /* checksum */
    u_int16_t th_urp; /* urgent pointer */
};

```

그림 12. 터널링 구현을 위한 TCP 프로토콜 재정의 구조

2) 터널링을 위한 베이스 스테이션

터널링을 구현하기 위해 베이스 스테이션에는 각 노드들의 정보, 수집된 데이터, 도착지의 USN에 대한 정보들을 서로 연결시켜주기 위한 각각의 정보들이 요구된다.

첫 번째로 도착지의 USN 망을 구별하고, 연결된 도착지 베이스 스테이션의 IP 기반의 주소를 매핑하기 위한 테이블이 필요하며, 이는 제안 패킷 구조의 [그룹] 필드를 참고하여 표 3.의 [그룹]필드에 대한 주소 매핑 테이블과 같이 나타낼 수 있다.

표 3. [그룹] 필드에 대한 주소 매핑 테이블

IP Address	Group ID	dPort
117.17.102.100	1	8801
117.17.102.101	2	8802
⋮		
117.17.102.n	n	xxxn

표 3.의 매핑 테이블을 이용하여 도착지의 USN을 구분하고, 도착지의 베이스 스테이션의 주소를 알아낼 수 있다. 또한 dPort는 해당 도착지의 노드 ID를 식별하기 위한 수단으로 사용한다.

두 번째로 도착지 내의 센서 노드의 ID를 패킷내의 destination ID를 통해 알아낼 수 있으며, 이를 이용하여 노드로의 데이터를 전송할 수 있다. 그러나 도착지의 그룹 ID 정보가 출발 USN내의 그룹 ID와 서로 다르게 되어 도착지에서의 노드 전송이 제대로 이루어지지 않을 가능성이 존재한다.

이러한 경우에 대비하여 본 제안 논문에서는 destination ID를 통한 정보 전달을 기본으로 하되, 그룹 ID 정보의 상이로 인한 도착지 센서 노드로의 패킷 전송이 되지 않을 경우에 대비하여 destination ID를 dPort로도 구분하여 전송하도록 설계하였다.

3) 적용 시나리오

본 제안 연동 방법을 이용한 적용 시나리오는 크게 두가지로 나타낼 수 있다.

첫 번째, 서로 다른 망의 데이터 정보를 동기화하기 위하여 적용할 수 있다. 현재, TinyOS내에는 동일 USN에서의 노드별 프로그램 동기화를 위하여 Deluge라는 프로그램을 제공하고 있다.

Deluge는 컴파일된 프로그램의 Binary와 같은 큰 데이터 객체를 무선 센서 네트워크내에 있는 많은 노드들에 퍼트리는 유용한 방법으로, 그림 13.에 나타낸 것과 같이 Bootloader 및 command와 같은 매커니즘을 결합하여 해당 그룹 안에 있는 노드를 원격으로 프로그래밍 할 수 있다.

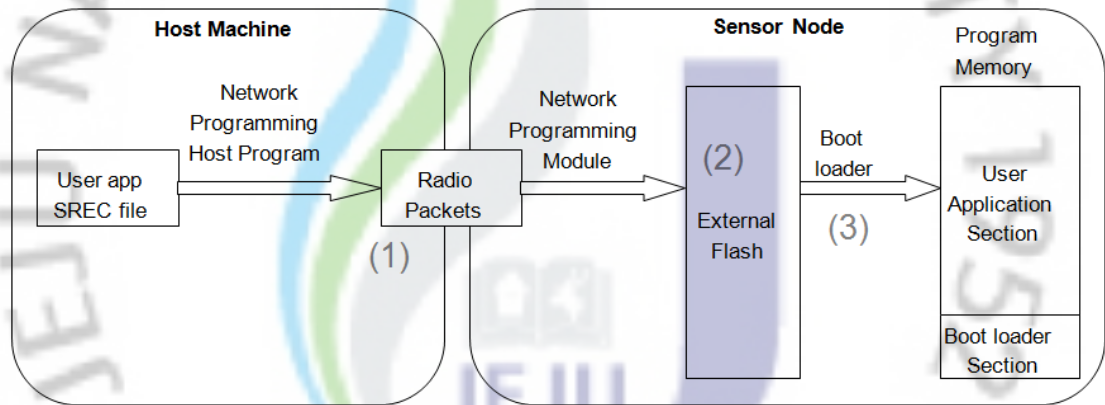


그림 13. Deluge 매커니즘

이러한 Deluge 프로그램은 동일 USN에서의 여러 센서노드들의 각 역할이 부여된 프로그램을 일시에 수정하기 위하여 널리 사용되어지고 있다. 그러나 Deluge 프로그램은 동일 USN에서의 프로그램 업데이트에는 용이하나, 서로 다른 USN에서는 동일한 프로그램을 업데이트하거나 동기화가 불가능하다.

이러한 Deluge 프로그램을 본 제안 연동 방법을 이용하여 전송하게 된다면, 간단히 서로 다른 USN에서도 똑같은 프로그램으로 동기화 업데이트가 가능하다.

두 번째, 서로 다른 지역간의 정보 공유를 위하여 본 제안 연동 방법은 유용하게

사용될 수 있다. A라는 지역에서의 온도, 강수량 정보의 이상이 발생한 경우, 다른 지역인 B 지역으로 그 정보를 바로 전송할 수 있으며, B 지역에서는 그 정보를 이용하여 미리 액츄레이터를 제어하여 예방할 수 있는 여러 가지 시나리오 적용 모델들을 생각해 볼 수 있다.

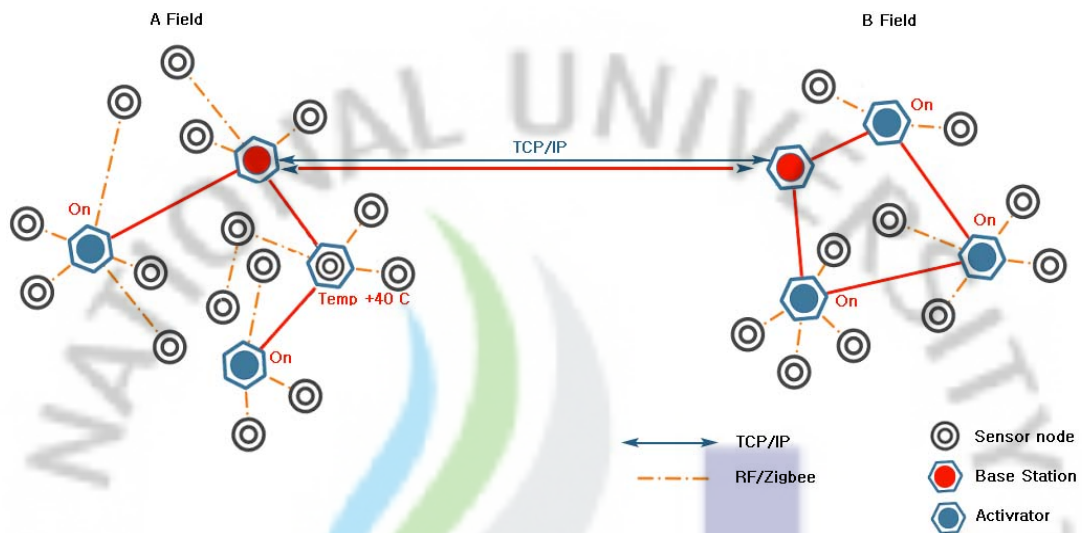


그림 14. 다른 지역간 액츄레이터 제어 시나리오

이러한 시나리오는 서로 다른 USN 망에서의 정보 공유 및 액츄레이터 제어가 필요한 모든 곳에서 이용될 수 있으며, 이에 유용한 연동 방법으로 적용될 수 있다.

IV. 시스템 구현 및 분석

본 논문에서 제안하는 연동 방법을 구현을 위해 전자부품연구원에서 제작된 Kmote와 Kmote-S1 등의 하드웨어를 사용하여 각 센서 노드와 베이스 스테이션을 구축하였다.

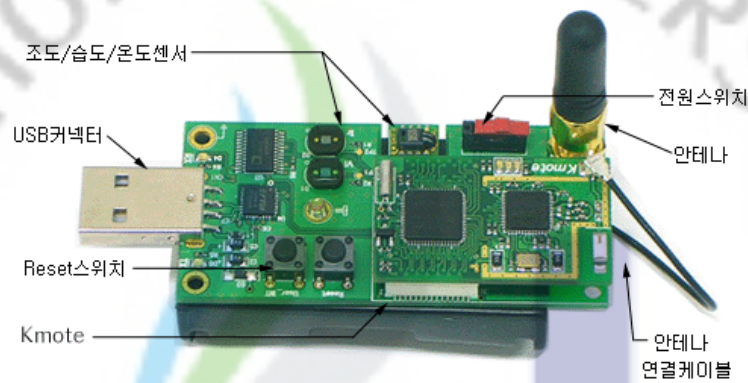


그림 15. Kmote-S1 센서

표 4. 구현 사양 및 요구조건

항목	사양 및 조건
센서 노드	Kmote-S1 4EA - SN : 기본 온도값 센싱 - BS : 정적 그룹 매핑 테이블 내장 UART 통신 및 TCP 패킷 분석 프로그램
노트북	2EA - OS : Fedora Core 5, VMWARE - TinyOS 1.1.x
nesC 구현	- UART, RF, MSP430interrupt 등 각종 컴포넌트를 이용하여 구현 - Osilloscope 및 TOSbase 프로그램

그림 15.는 Kmote-S1 센서를 나타내고 있으며, Kmote-S1 노드는 SHT11 통합 센서를 이용하여 조도/습도/온도 정보를 센싱할 수 있는 역할을 가지고 있다.

구현을 위한 사양 및 요구조건은 표 4.에 나타내었으며, 센서 노드들은 기본적으로 온도값을 센싱하도록 구성되었다.

1. 흐름도

구현을 위한 소스노드에서 도착지 노드까지의 흐름을 그림16.에 나타내었다.

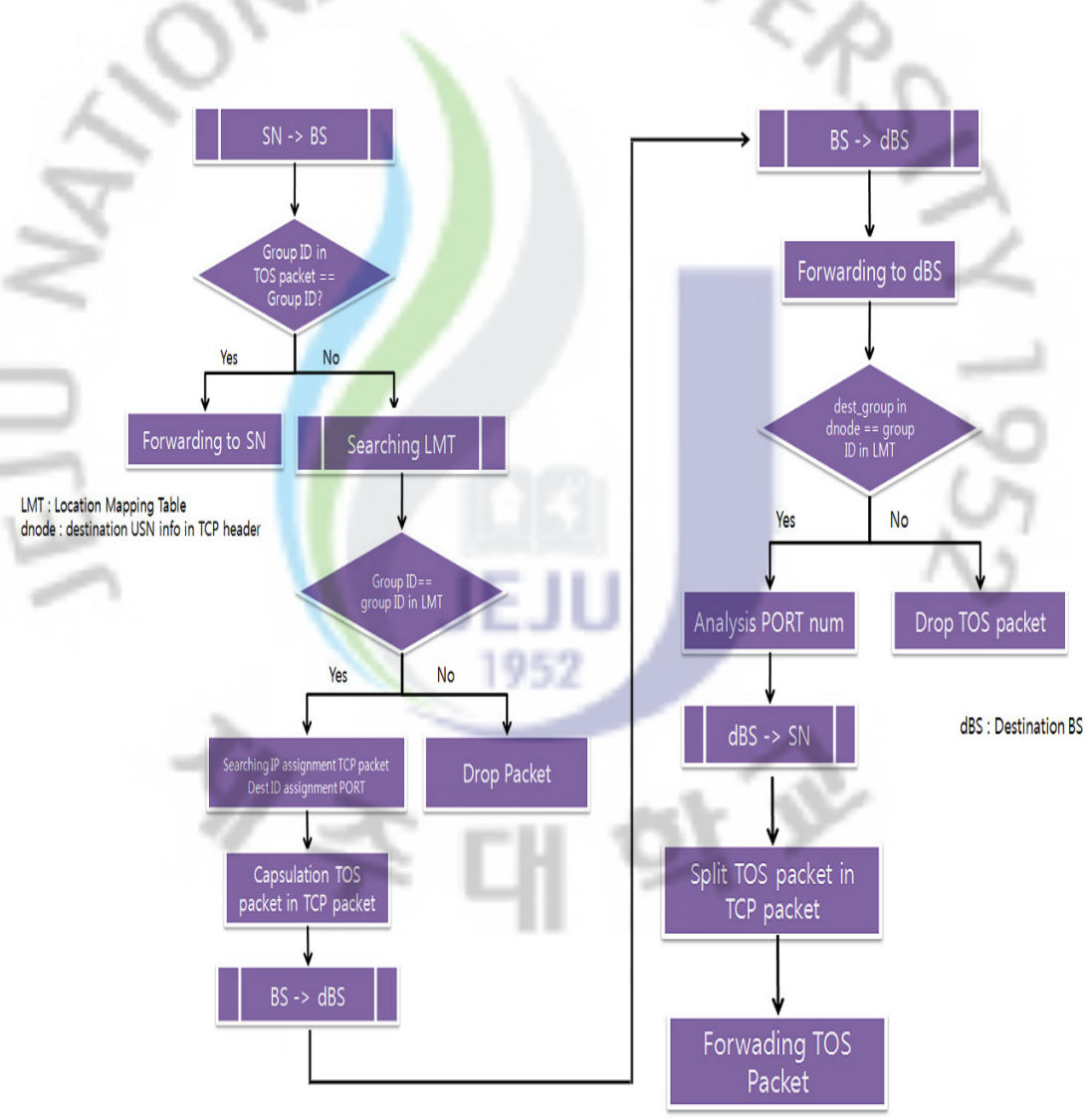
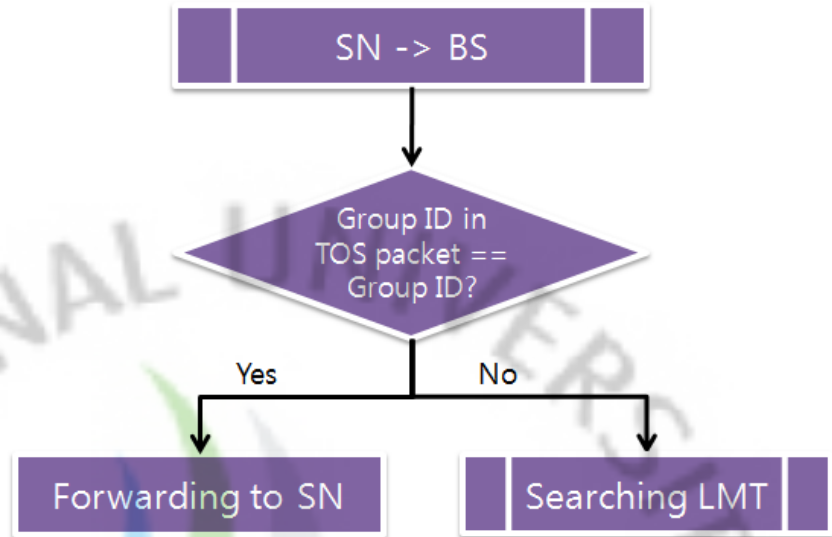


그림 16. 제안 연동 방법 흐름도

1) 센서 노드 → 베이스 스테이션 처리 모듈



LMT : Location Mapping Table
dnode : destination USN info in TCP header

그림 17. 센서 노드 -> 베이스 스테이션 처리 모듈

그림 17.은 센서노드에서 베이스 스테이션으로의 데이터 처리 모듈을 나타낸 것이다. 이 모듈은 데이터를 보내고자 하는 센서노드로부터 동일 USN에 존재하는 베이스 스테이션으로 데이터를 보내며, 베이스 스테이션에서 처리하는 동작과정을 나타낸다.

베이스 스테이션은 센서 노드로 부터 전달받은 TOS 패킷내의 그룹 ID 정보를 현재 USN의 그룹아이디와 동일한지 확인하고, 외부로 데이터를 보내게 될 경우에는 LMT을 검색하는 모듈로 처리가 진행되게 되며, 내부로 데이터를 보내게 될 경우에는 바로 내부의 센서 노드로 데이터를 전송하게 된다.

2) LMT 검색 모듈

LMT 검색을 위한 모듈에서는 TOS 패킷안의 그룹 ID와 위치 매핑 테이블에 존재하는 그룹 ID를 검색하여, 해당 정보가 없는 경우는 도착지 USN 베이스 스테이

션의 정보가 존재하지 않으므로, 패킷을 버리게 된다.

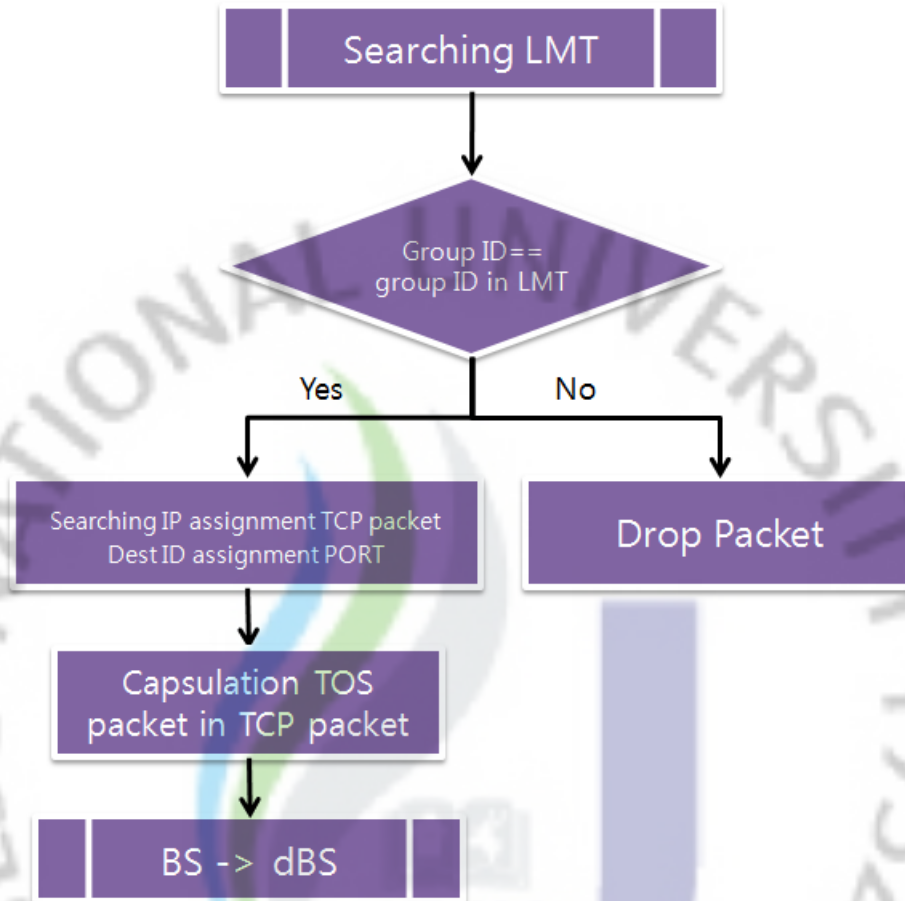


그림 18. LMT 검색을 위한 모듈

그러나 해당 정보가 존재하는 경우에는 그룹 ID와 매핑된 도착지 USN 베이스 스테이션의 IP 주소를 IP 헤더에 삽입하고, TOS 패킷안의 Dest ID 정보를 LMT에서 검색하여, dPORT 정보를 같이 TCP 헤더에 삽입하게 된다.

또한, 원래의 TOS 패킷은 TCP 프로토콜의 데이터 부분에 그대로 캡슐화하게 되며, 이를 베이스 스테이션에서 도착지의 베이스 스테이션 IP를 통해 정보를 전달하게 된다.

3) 베이스 스테이션 -> 도착지 베이스 스테이션 정보 전송 모듈

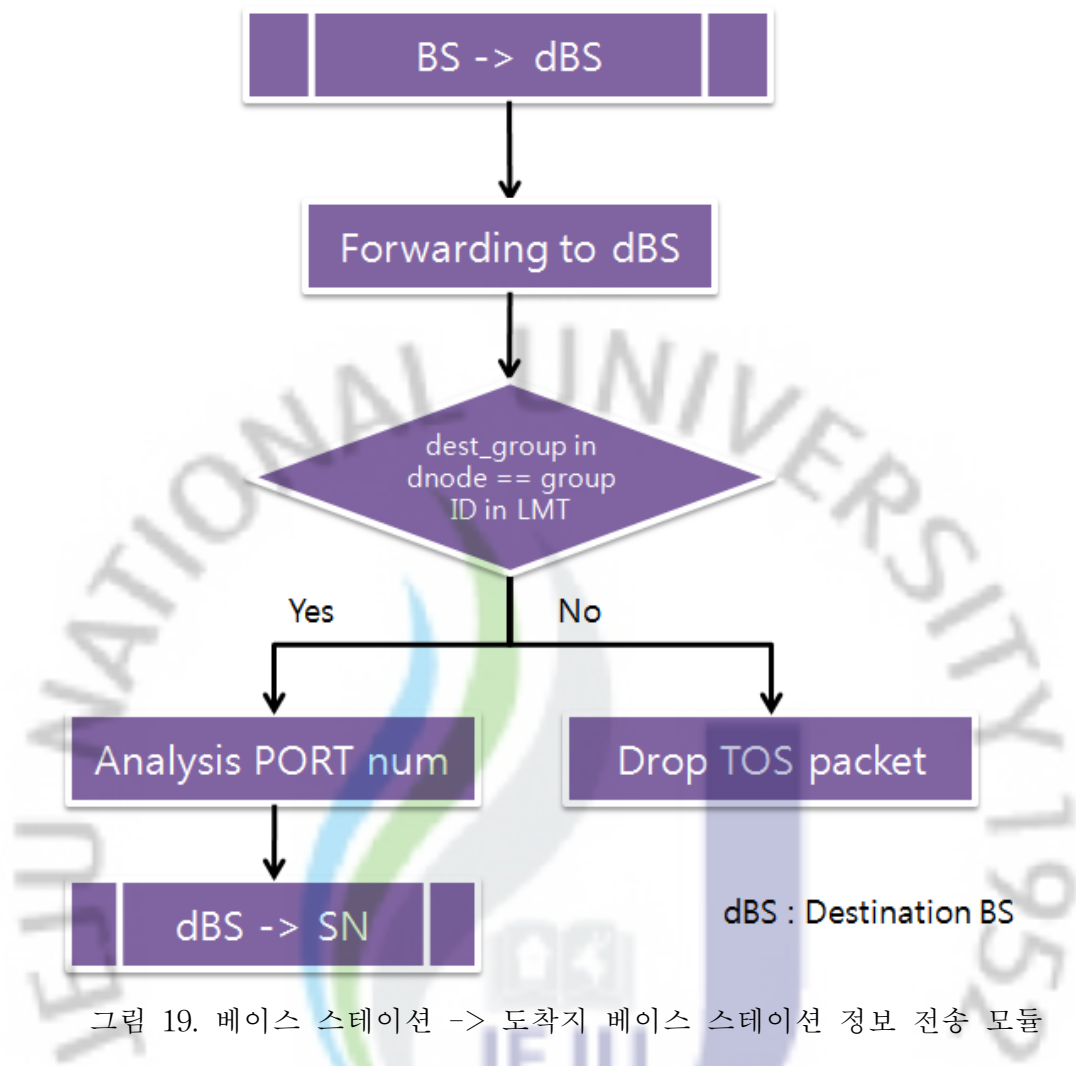


그림 19. 베이스 스테이션 -> 도착지 베이스 스테이션 정보 전송 모듈

베이스 스테이션에서 도착지 베이스 스테이션으로 캡슐화된 정보를 TCP 헤더에 적재하여 포워딩 하게 되면, 정보를 받는 도착지 베이스 스테이션에서는 몇가지 작업을 수행하게 된다.

도착지 그룹 ID와 TCP 헤더에 존재하는 dest_group ID와 동일한지를 판단하여, 동일하지 않다면, 잘못 전송된 패킷정보이므로, 이를 버리게 되고, 동일하다면, TCP 헤더에 존재하는 PORT 정보를 이용하여, 도착지 USN 내에 존재하는 센서 노드의 정보를 분석하게 된다.

4) 도착지 베이스 스테이션 -> 도착지 노드 전송 모듈

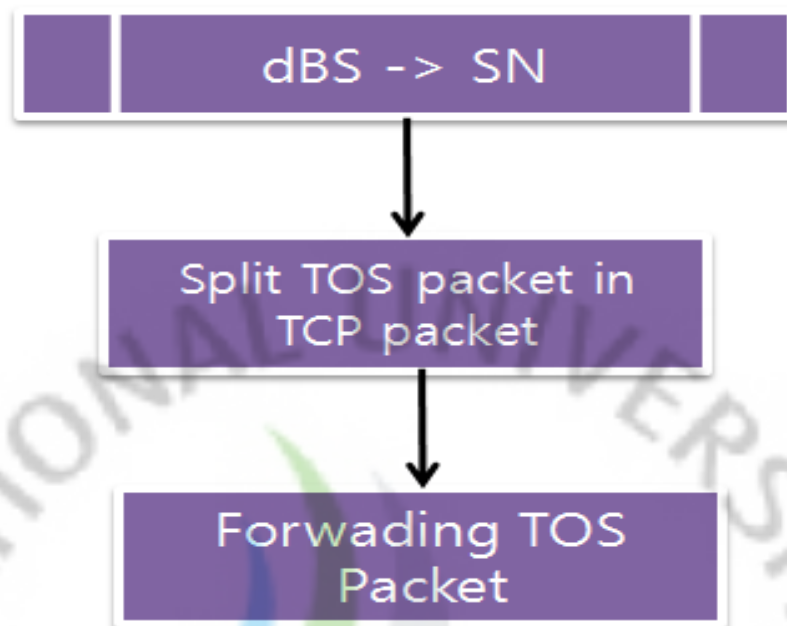


그림 20. 도착지 베이스 스테이션 -> 노드 전송 모듈

도착지 베이스 스테이션에서는 캡슐화된 TCP 패킷내에서 TOS 패킷을 쪼개어, 이를 네트워크내의 해당 센서 노드로 정보를 전송하게 된다.

이러한 과정을 통하여, 터널링 연동 방법을 이용한 서로 다른 USN의 연동이 가능하다.

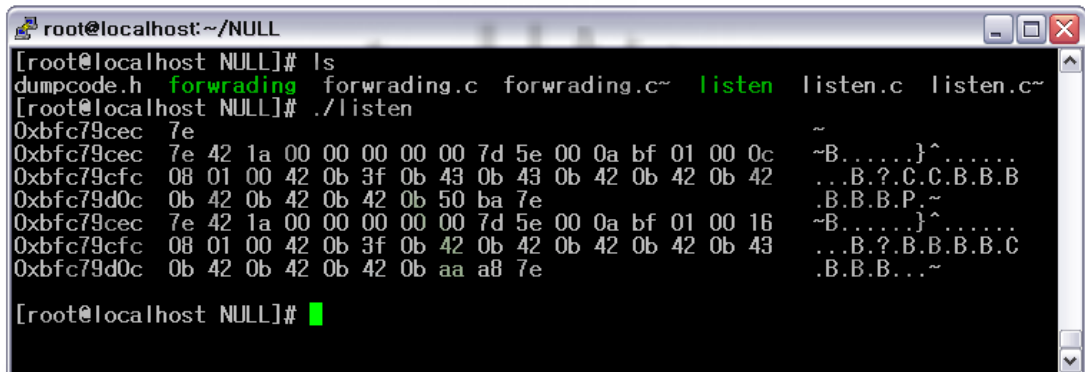
2. 구현 결과

데이터를 보내고자 하는 USN(Source Network) 내의 센서노드 A에서 RF통신을 통하여 베이스 스테이션으로 데이터를 보내면, 베이스 스테이션에서는 그 데이터를 IP 및 TCP 프로토콜안에 캡슐화하여 도착지의 서로 다른 USN(Destination Network) 내의 센서노드 B에 패킷이 도착함을 확인함으로써, 터널링을 통한 연동 방법의 구현을 확인하였다.

그림 21.은 Source Network의 베이스 스테이션에서 센서노드 A로부터 RF 통

신으로 넘겨받은 값을 나타내었다.

이렇게 전송받은 값을 베이스스테이션에서는 IP 프로토콜 헤더에 캡슐화 하게 되는데, 캡슐화 하게 될 때는 TCP 도착지 포트 필드에 도착지 센서노드의 정보를 저장하여 보내게 된다.



```
root@localhost:~/NULL
[root@localhost NULL]# ls
dumpcode.h forwading forwading.c forwading.c~ listen listen.c listen.c~
[root@localhost NULL]# ./listen
0xbf79cec 7e ~
0xbf79cec 7e 42 1a 00 00 00 00 7d 5e 00 0a bf 01 00 0c ~B.....}^.....
0xbf79cfc 08 01 00 42 0b 3f 0b 43 0b 43 0b 42 0b 42 0b 42 ...B?.C.C.B.B.B
0xbf79d0c 0b 42 0b 42 0b 42 0b 50 ba 7e .B.B.B.P.~
0xbf79cec 7e 42 1a 00 00 00 00 7d 5e 00 0a bf 01 00 16 ~B.....}^.....
0xbf79cfc 08 01 00 42 0b 3f 0b 42 0b 42 0b 42 0b 42 0b 43 ...B?.B.B.B.B.C
0xbf79d0c 0b 42 0b 42 0b 42 0b aa a8 7e .B.B.B...~

[root@localhost NULL]#
```

그림 21. Source Network내의 수집된 센서 노드 A의 값

외부에서 센서노드에 접근을 하게 될 때에도 데이터를 받는 USN의 IP와 함께 도착지 포트 필드에 도착지 노드 정보를 삽입함으로써 해당 노드에 접근이 가능하다.

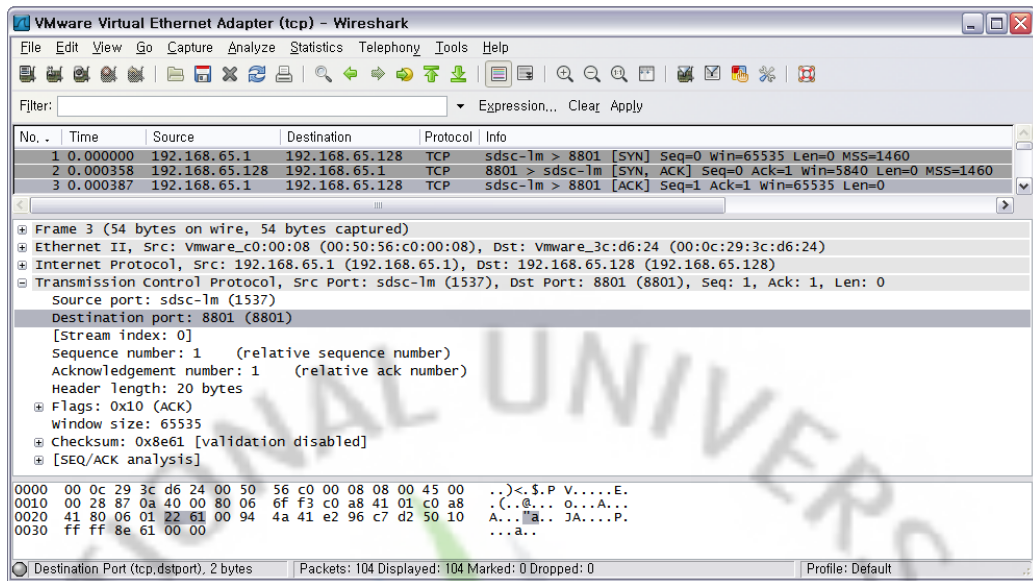


그림 22. 호스트와 베이스 스테이션이 TCP 연결을 맺는 화면

그림 22.은 Source Network의 베이스스테이션과 Destination Network의 베이스 스테이션이 서로 IP 망을 통하여 TCP 연결을 이루고, 이를 통해, 데이터를 전송하는 것을 나타낸다.

Source Network의 베이스스테이션(192.168.65.1)에서 Destination Network의 베이스 스테이션(192.168.65.128)의 센서노드 1번과 3단계 핸드셰이크(three-way handshaking)방법을 맺는 패킷을 Wireshark 라는 프로그램으로 캡처하였다.

3단계 핸드셰이크(three-way handshaking)는 센서노드 상의 TCP/IP 스택이 아직 구현전이기 때문에 베이스 스테이션에서 연결을 맺는 것으로 베이스 스테이션에서 연결을 맺은 뒤에 실제 노드와 자료 교환이 이루어지게 된다. 노드와 Source Network의 베이스 스테이션간의 자료 교환은 Sequence Number 와 Acknowledgment Number 를 통해서 확인하므로 신뢰성 있는 자료 교환이 이루어지게 된다.

Source Network와 Destination의 베이스 스테이션간의 연결은 다음의 절차를 통하여 이루어진다.

- 1) Source Network의 베이스 스테이션에서 Destination Network의 베이스 스테

이선으로 연결하기 위하여 SYN Flag가 설정된 패킷을 보냄.

- 2) Destination Network의 베이스 스테이션에서 다시 SYN, ACK Flag가 설정된 패킷을 보냄.
- 3) 최종적으로 Source Network의 베이스 스테이션에서 ACK Flag가 설정된 패킷을 보냄으로써 연결이 확립됨.

이렇게 연결이 되면, 그림 23.와 같이 데이터를 전송하게 된다.

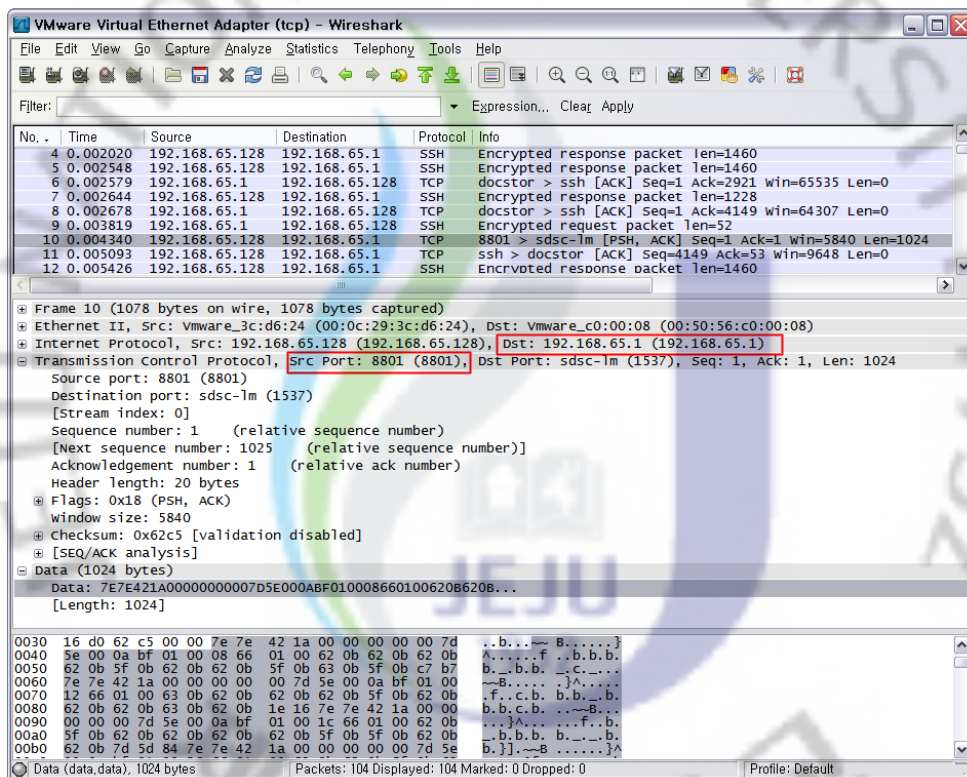


그림 23. 센서노드가 데이터를 보내고 있는 화면

TCP의 포트번호 뒤의 1Byte는 "01"로 1번 노드를 의미하며 Source Network의 베이스 스테이션과 Destination Network의 센서노드는 서로 데이터를 교환하기 전에 3단계 핸드셰이크(three-way handshaking) 방법을 이용하여 연결을 맺는다. Source Network의 베이스 스테이션에서 Destination Network의 베이스 스테이션으로 연결을 맺은 뒤에 Source 센서 노드가 Source Network의 베이스 스테이션으로

로 데이터를 보내게 된다. Destination Network의 베이스 스테이션이 IP가 192.168.65.1이고 Source Network의 베이스 스테이션에서 Port번호 8801으로 전송하고 있음을 그림 23.에서 확인할 수 있다. Port번호 8801번은 도착지 센서 노드의 1번으로 보낸다는 것을 의미한다. 이를 통하여 IP주소와 포트번호를 통하여 원하는 센서 네트워크의 센서노드로 접근이 가능하며, TCP의 포트번호를 통하여 기존의 어플리케이션과의 쉽게 호환이 가능하다.



3. 결과 분석

본 논문에서의 제안 연동 방법을 이용하여 IP-USN의 연동이 가능함을 확인할 수 있었다. 서로 다른 USN을 연동하기 위하여 터널링 방법을 사용하였고 TCP/IP 망을 가상의 터널로 IP 및 TCP 프로토콜내에 USN 데이터를 캡슐화 하여 보내는 방법을 이용하였다. IP 헤더의 상위 프로토콜인 TCP 헤더의 Destination Port 에 목적지 USN 및 노드 정보를 저장하여 데이터를 캡슐화 하여 전송하였으며, TCP/IP 스택이 구현 되어있을 경우, USN 외부의 망에서도 TCP의 Destination Port에 해당 목적지 정보를 저장해서 보냄으로써 외부에서도 센서노드에 접근이 가능하다. 베이스 스테이션은 게이트웨이 역할을 하며 USN 데이터를 캡슐화/역캡슐화 과정 통하여 망과 망 사이 데이터 전달을 중개하는 역할을 한다.

본 제안 연동 방법을 이용하면, 자체의 USN 환경에서는 별도의 TCP/IP 스택이 구현되지 않아도 TOS_Msg 등의 자체 프로토콜을 가지고 통신이 가능하다는 장점이 있다. 이에 따라 센서 노드에서의 TCP/IP의 프로토콜을 처리하기 위한 오버헤드가 존재하지 않는다.

베이스 스테이션에서의 IP 및 TCP 프로토콜내의 캡슐화 및 역캡슐화, 다른 USN망의 LMT 테이블 정보를 통한 매핑 등의 처리만 있으면, 간단히 터널링을 구현할 수 있다.

센서 노드 데이터를 캡슐화 하는 방법에는 두 가지 방법에 의해 구현이 가능하다.

첫째 방법은 IP 헤더의 Identification 부분을 재정의 하여 USN 그룹 정보와 센서 노드 정보를 저장해서 보내는 방법이다. 이 방법의 경우에는 TCP의 헤더의 용량만큼 데이터그램의 크기를 줄일 수 있다는 장점이 있지만 줄일 수 있는 데이터의 크기는 옵션 부분을 제외한 TCP 헤더의 20바이트이며, 이 정도 크기면 센서 네트워크가 아닌 외부 망에서는 부담이 상대적으로 적은 크기이다. 그리고 IP 데이터그램에 노드정보를 저장해서 보냈을 경우 Raw Level 에서 처리해야 되므로 어플리케이션과의 호환성이 상대적으로 어렵고 데이터 전송 중에 발생하는 데이터의 오류 검출 등이 어렵다.

둘째 방법은 TCP 헤더의 Destination Port부분을 재정의 하여 USN 그룹 정보와

센서 노드 정보를 저장해서 보내는 방법이다. 이 방법의 경우에는 패킷의 크기가 IP 헤더만 보내는 것보다 상대적으로 커지게 되나, Port 번호에 센서노드 정보를 저장하여 보냄으로써 어플리케이션과의 호환성이 높다. 그리고 TCP 연결지향 프로토콜로 데이터를 전송하기 전에 3단계 핸드셰이크(three-way handshaking) 방법을 통하여 연결을 맺기 때문에 확인응답과 순서번호를 통해서 데이터 송수신 과정에서 확인 가능하여, IP 헤더에 목적지 정보를 저장해서 보내는 것보다 안정적으로 보낼 수 있다는 장점이 있다.

따라서 제안 방법에서 캡슐화를 IP 헤더에 구현하지 않고, TCP 헤더를 통하여 구현한 것은 표 5.와 같은 TCP의 장점을 고려하였기 때문이다.

표 5. IP와 TCP의 비교

	IP	TCP
패킷의 크기	적다	상대적으로 크다
어플리케이션과의 호환	상대적으로 어렵다	호환성이 높다
데이터 전송의 신뢰성	낮다	높다

IV. 결과

본 논문에서는 Tunneling 기법을 이용하여 IP-USN 연동 방법에 대한 연구를 수행하였다. 프로토콜이 서로 다른 망을 융합하기 위한 방법에는 Dual Stack 구현, Header Translation 그리고 Tunneling 기법이 있는데 본 논문에서는 USN의 데이터를 현재 사용 중인 IPv4를 적용한 IP망과의 연동을 위하여 Tunneling 기법을 이용한 IP-USN을 구축하는 방법에 대하여 제안하였다.

본 논문의 제안 방법은 USN 데이터를 IP 데이터그램에 캡슐화 시켜서 보냄으로써 가능함을 확인하였는데, 캡슐화 시키는 방법으로는 상대적으로 어플리케이션들과의 호환성을 높이고 신뢰성 있는 데이터 전송을 위하여 TCP프로토콜의 Destination Port 부분에 데이터를 캡슐화 시켜서 보냄으로써 가능하였다.

또한, 자체의 USN 환경에서는 별도의 TCP/IP 스택이 구현되지 않아도 TOS_Msg 등의 자체 프로토콜을 가지고 통신이 가능하다는 장점이 있다. 이에 따라 센서 노드에서의 TCP/IP의 프로토콜을 처리하기 위한 오버헤드가 존재하지 않는다.

베이스 스테이션에서의 IP 및 TCP 프로토콜내의 캡슐화 및 역캡슐화, 다른 USN망의 LMT 테이블 정보를 통한 매핑 등의 처리만 있으면, 간단히 IP-USN망 연동을 구현할 수 있다.

본 제안 방법을 적용시 Deluge 프로그램의 동일 USN에서의 업데이트 프로그램의 한계를 극복하여 서로 다른 USN에서도 동일한 업데이트가 가능하다.

또한, 서로 다른 USN 망에서의 정보 공유 및 액츄레이터 제어가 필요한 모든 곳에서 이용될 수 있으며, 이에 유용한 연동 방법으로 적용될 수 있다.

본 제안 방법에 대한 향후 연구로는 정적 테이블을 가정하여 LMT를 구축하였으나, 동적 환경에서의 서로 다른 USN 갱신을 파악할 수 있도록 이를 보완하기 위한 연구가 수행되어야 할 것으로 사료된다.

참고문헌

- [1] 이재용 “유비쿼터스 센서 네트워킹 기술” TTA 저널 제95호
- [2] Ian F.Akyildiz, Weilian Su, Yogesh Sankarasubramaniam, "A Survey on Sensor Networks", IEEE Communications Magazine, August 2002.
- [3] W.Ye, H Heidemann, D.Estrin, "An Energy-efficient MAC protocol for wireless sensor networks" In 21 st conference of the IEEE computer and Communications Societies(INFOCOM), volume 3,pages 1657-1576, June 2002.
- [4] 박승민, “센서네트워크 노드 플랫폼 및 운영체제 기술동향” 전자통신동향분석 제21권 제1호. 2006. 2
- [5] <http://tinyos.keti.re.kr/>
- [6] Jason Hill, Robert Szewczyk, Alec Woo, Seth Hollar, David Culler, Kristofer Pister, "System Architecture Directions for Network Sensors" ASPLOS 2000, Cambridge, Nov.2000
- [7] 김대영, 김재연, 유성은, 성종우, “USN 센서 네트워크 기술” OASIS Standard & Technology Review, 제 25권 1호. 2005. 12
- [8] Daeyoung Kim, Tomas Sanchez Lopez, Seongeun Yoo, Jongwoo Sung, Jaeon Kim, Youngsoo Kim, Yoonmee Doh, "ANTS:An Evolvable Network of Tiny Sensors" EUC2005, LNCS 3824, pp.142-154, 2005.
- [9] 박준성, 김기형, “IP-USN 기술 및 표준화 동향”, 전자교환기술, 제 23권 2 호 2007.
- [10] 허의남, 홍충선, 추현승, 강석봉,정현철 “IP-USN 연동을 위한 핵심기술 개발 및 KOREAN에서의 활용”, 한국 정보 사회 진흥원 2009.01.10.
- [11] 김학범 “IP-USN 최신 기술 동향 및 보안요구사항 분석” 통신정보보호학회 지 제16권 6호 [2006] pp.64~73 (10pages)
- [12] Zigbee Alliance 홈페이지, <http://www.zigbee.org>.
- [13] 김대영, 도윤미, 박노성, 서창우, 유성은, “센서네트워크 저전력 네트워크 프

- 로토콜 기술” 한국인터넷정보학회 제5권 제4호. 2004.12
- [14] 원관호, 김재호, 유준재 “지그비” TTA 저널 제94호, 2004.7
- [15] 김계원, 서재완, 황대준, 추현승, “구글맵을 이용한 위치 추적 서비스를 제공하는 6LoWPAN 테스트베드 구현” 한국 인터넷 정보학회 (10권 5호) 2009.10
- [16] 김정희, 권훈, 김도현, 곽호영, 도양희, 변영철, “센서 네트워크와 인터넷과의 정적 주소 연동 방안 구현” 한국콘텐츠학회논문지 '06 Vol. 6 No. 12
- [17] 김은숙, 김용운, “6LoWPAN 기반의 IP-USN 기술 표준화 동향” 전자통신동향 분석 제 22기반의 6 호 2007.12
- [18] K.Kim, G.Montenegro, S.Daniel Park, I.Chakeres, and S.Yoo, “Dynamic MANET On-demand for 6LoWPAN(DYMO-Low) Routing,” draft-montenegro-6lowpan-dymo-low-routing-00.IETF, 2007.6
- [19] 임채성, Waleed Mansoor, 김기형, 박수홍, 이재호, “IPv6기반 센서 네트워크 (6LoWPAN)을 위한 라우팅 프로토콜 기술” 전자공학회지 제33권 제8호. 2006.8
- [20] 정석, 신운섭, 유승화, 노병희, 김기형, “6LoWPAN을 이용한 기상관측 시스템 구현 및 결과 분석” 2007 한국컴퓨터종합학술대회 논문집 Vol.34, No.1 (D)
- [21] 유상근, 안상현, 임유진, 김용운, “이기종 센서 네트워크간의 연동을 위한 센서 네트워크 구조” 정보과학회지 제25권 제12호. 2007.12
- [22] Adam Dunkels, Juan Alonso, Thiemo Voigt, Hartmut Ritten and Jochen Schiller, “Connecting Wireless Sensor Networks with TCP/IP Networks”, WWIC 2004, Feb.2004
- [23] M.Zuniga and B.Krishnamachari, “Integrating Future Large-Scale Wireless Sensor Networks with the Internet”, USC Technical Report CS03-792, 2003
- [24] 김정희, 권훈, 김도현, 곽호영, 도양희, 변영철, “센서 네트워크와 인터넷과의 정적 주소 연동 방안 구현” 한국콘텐츠학회논문지 '06 Vol. 6 No. 12
- [25] Adam Dunkels, Juan Alonso, Thiemo Voigt, Hartmut Ritten and Jochen

Schiller, "Making TCP/IP viable for wireless sensor networks",
Proc.EWSN 2004. Jan. 2004.

[26] <http://www.ietf.org/rfc/rfc1631.txt>

[27] <http://www.ietf.org/rfc/rfc2663.txt>

[28] C.Milner, "Network address translation(nat)," Cisco Systems, Inc



감사의글

공학도의 꿈을 안고 설레임반 두려움반으로 한국이라는 낯선 나라에서 와서 모든 것이 새롭고 낯설기만 한 문화와 언어를 익혀가고, 새로운 친구를 만들고... 이제 겨우 한국이란 나라가 몸에 와 닿을만하니 어느덧 대학원 생활을 마무리 하고, 졸업 을 하게 되었습니다.

여러 가지 낯설음 속에서도 제가 대학원 생활을 잘 마무리 할 수 있었던 것은 멀리 있지만 마음속에선 항상 나를 지켜주고 응원해주었던 저의 가족들, 많은 배려와 격려, 아낌없는 가르침을 주신 지도 교수님을 비롯한 연구실의 선후배, 친구들 덕분에 이 자리까지 올 수 있었다고 생각합니다.

지도 교수님으로써 많이 부족한 제게 애정어린 관심으로 자신감을 심어주시고, 격려 해 주시며 오늘의 이런 결실을 얻게 큰 도움을 주신곽호영 교수님께 제일 먼저 감사 를 표합니다.

그리고 저의 논문 심사를 맡아주시고, 소중한 충고와 조언으로써 논문을 보완할 수 있게 해주신 이상준 교수님, 김도현 교수님께 깊은 감사를 드리며 김장형 교수님, 안기중 교수님, 변상용 교수님, 송왕철 교수님 그리고 변영철 교수님께도 감사 의 마음을 전하고 싶습니다.

제 논문 대해 같이 고민해 주고, 바쁜 일과에도 기꺼이 많은 시간을 내어 자신의 일처럼 신경을 써주신 권훈 선배님과 부족함이 많았던 나를 끝까지 믿고 도와준 익선이에게 한없이 많은 고마움을 전합니다.

연구실 생활을 함께 하며 낯선 곳에서도 웃음을 잃지 않고, 즐겁게 학교 생활을 잘 마무리 할 수 있게 해준 미경이언니, 동균이와 진우, 에르텐토야, 효주, 진수, 영철이에게도 고마운 마음을 글로 대신합니다.

그리고 용기를 잃지 않게 힘이 되어주신 노영식 선배님, 창영이와 지윤, 이정하 조교선생님, 정은경 조교선생님께도 그동안의 격려와 배려에 감사드립니다.

한국에서의 생활, 특히 제주대학교 컴퓨터공학과 교수님, 친구, 선후배.. 많은 분들과 함께했던 시간은 평생 잊을 수 없는 소중한 기억이 될 것입니다.

다시 한번 머리 숙여 깊이 감사드립니다.

2010년 7월

