

# 무선 센서 네트워크 기반의 빌딩 조명 제어 시스템

## Intelligent building light control system based on wireless sensor network

강정훈, 유준재, 윤명현, 이명수, 임호정, 이민구, 장동섭  
전자부품연구원

Jeonghoon Kang, JunJae Yoo, Myunghyun Yoon, Myungsoo Lee,  
Hojung Lim, Mingoo Lee, Dongseop Jang  
Korea Electronics Technology Institute

**Abstract** - Sensor network supports data delivery from physical world to cyber space. Sensors get physical events then wireless network transfers sensor data to service server. We use sensor network technology to light control system for intelligent building. In ubiquitous computing environment, wireless sensor network is basic tool for intelligent service. In this paper, we propose intelligent building light control system based on wireless sensor network. It is implemented using previous light control product, can be adopted to present building light system.

**Key Words** : wireless sensor network, light control, intelligent building system, ubiquitous computing

### 1. 서 론

센서 네트워크(sensor network)는 물리공간의 상태인 빛, 소리, 온도, 움직임 같은 물리적 데이터를 센서 노드에서 감지하고 측정하여 중앙의 기본 노드(base-station or sink)로 전달하는 센서 노드들로 구성되는 네트워크이다. 센서 네트워크는 일반적으로 멀티-홉(multi-hop) 무선 네트워크 형태의 다수의 분산 센서 노드들로 구성된다. 센서 노드들은 하 나이상의 센서(온도, 소리, 빛, 가속도, 위치, 자기장 등), 액추에이터(actuator), 마이크로 컨트롤러, 수 십 KB 크기의 EEPROM, 수 KB의 SRAM, 수백 KB 크기의 플래시 메모리, 근거리 무선 통신 모듈로 구성된다. 센서 네트워크 기술은 센서와 무선 네트워크 기능을 이용하여 물리공간에서 측정한 아날로그 데이터를 디지털 신호로 변환하고, 인터넷 같은 전자공간에 연결된 루트(root) 노드로 전달하는 입력시스템의 역할을 한다. 물리적 세계와 사이버 세계를 연결할 수 있는 특징 때문에, 센서 네트워크의 개념은 새롭게 대두되고 있는 지능형 서비스들의 지능형 환경 모니터링, 위치인지 서비스, 지능형 의료시스템, 지능형 로봇 시스템 등 다양한 분야에 적용되고 있다.

센서 네트워크의 장점은 낮은 사양의 하드웨어를 이용하여 무선 애드-혹(ad-hoc) 네트워크를 구성할 수 있는 점이다. 예를 들어, 지금까지 개발된 블루투스(bluetooth), 무선랜(wireless LAN)등의 무선 네트워크 기술들은 반드시 컴퓨터, PDA 같은 고급 컴퓨터 장치를 필요로 하는데, 센서 네트워크 노드는 독자적으로 네트워크를 구성한다. 이런 네트워크 구성의 용이성 때문에 유비쿼터스(ubiquitous) 컴퓨팅 환경의 기반기술로 사용될 수 있을 것이다. 인텔도 기술 로드맵 중에 센서 네트워크를 기반 기술로 예측하고 있다[1].

근래 센서 네트워크와 관련된 칩과 시스템들이 상용화되기

시작했으나 널리 보급되고 산업화되기 위해서는 해결되어야 할 문제들이 많이 남아있다. 네트워크 프로토콜, 적절한 응용 서비스 등의 부재로 인해 빠른 시간 내에 상용화되기 힘들 것으로 예측되기도 한다.

TinyOS는 새로운 컴퓨팅 패러다임인 저가의 소형 컴퓨터 시스템, 즉 네트워크 임베디드 시스템 또는 센서 네트워크에 사용될 수 있는 시스템 기술이다. 버클리 대학에서 주도적으로 개발을 추진하고 있으며, 현재 센서 네트워크 분야에서 실제 구현 및 적용에 가장 빠르게 대응하고 있다. 인텔 및 크로스보우 등의 회사가 이 기술의 확산을 위해 지속적으로 투자하고 있으며 마이크로소프트사도 관련 기술을 보유하기 위해 노력 중이다.

TinyOS가 표면적으로는 지그비(Zigbee) 표준과 다른 성격으로 보일 수도 있지만, 주요 요소기술들은 동일한 기능을 지향하고 있다. 이런 이유로, 근래 본격적인 구현 및 산업화가 시도되면서 두 기술에 대한 업계의 관심이 높아지고 있다.

본 논문에서는 무선 센서 네트워크를 이용한 빌딩 조명 제어 시스템 구현에 필요한 TinyOS 및 멀티-홉 센서 네트워크 기술을 분석하고, 응용 서비스를 위한 시스템의 구조를 제안한다. 그리고 실제 구현된 결과와 향후 고려해야 할 사항을 정리한다.

### 2. 관련 기술 동향

#### 2.1 TinyOS

TinyOS의 개발 목적은 세 가지로 설명할 수 있다[2]. 첫째, 미래의 센서 네트워크 노드에 사용될 수 있는 소프트웨어 구조를 설계, 둘째, 여러 하드웨어 플랫폼과 소프트웨어 라이브러리를 사용하여 운영체제와 서비스 애플리케이션을 구현할 수 있도록 설계, 셋째, 제한된 자원, 동시성, 강인성, 응용

서비스별 요구사항을 만족할 수 있는 설계이다.

이런 목표의 달성을 위해 TinyOS는 구성이 모듈화 되었으며, 제한된 자원에서 동작할 수 있는 이벤트 구동 방식으로 구현되었다. 모듈구성에 따라 운영체제가 다양한 하드웨어에 적용이 가능하며, 서비스 애플리케이션도 동일한 추상화를 유지할 수 있다.

TinyOS에서는 시스템과 사용자 부분을 나누어 놓지 않았다. 그러나 대부분의 애플리케이션에서 자주 사용되는 모듈들이 있는데, 타이머, 데이터 수신, 전력 제어, 네트워크 모듈들이 대표적인 예이다. 이 모듈들의 사용은 센서 네트워크 노드가 한정된 전력으로 동작하고 주기적으로 데이터를 수집하며, 간단한 데이터 처리와 주위 노드로 데이터를 전송한다는 주요 동작에 대한 특징을 나타내고 있으며, 다른 센서 네트워크 플랫폼인 맨티스(MANTIS)[3], 센서심(SENSORSIM)[4], 엠스타(EMSTAR)[5]등도 동일한 특징을 갖고 있다. TinyOS의 장점은 누구에게나 공개되어 사용할 수 있다는 점인데, 기존의 공개 소프트웨어와 동일한 장점을 갖는다. 리눅스(LINUX)나 비에스디(BSD)와 같은 기술 공개의 원칙은 산업화 및 기술 확산을 가속화 시킬 것으로 예측된다.

TinyOS는 컴포넌트 기반, 이벤트 기반의 운영체제이다. 블로킹되는 호출은 존재하지 않는다. 대신 블로킹이 필요할 정도의 장시간의 동작을 수행하는 호출은 즉시 리턴을 보내 주고, 리턴을 보내준 후후에 동작을 수행하고 수행이 종료되었을 때, 종료 상황을 알려준다. 따라서 호출은 두개의 방향으로 진행되는데, 이것을 split-phase라고 한다. 이 방법에서는 모든 컴포넌트들이 하드웨어 블록처럼 동작을 한다. 컴포넌트들은 명령을 받고, 이 명령이 종료되면 종료되었다는 이벤트를 상위로 알려준다. TinyOS의 concurrency 모델은 non-preemptive, 지연된 함수 호출 방식의 태스크에 기반하고 있다. 컴포넌트들은 추후의 실행을 위해 스케줄러에게 태스크를 호출하여 등록할 수 있다. TinyOS 스케줄러는 FIFO 방식의 고정 길이의 큐를 기반으로 한다.

모든 TinyOS의 코드들은 nesC로 작성되어 있다. nesC는 컴포넌트 형태의 C 언어이다. 프로그래밍을 할 때, 애플리케이션을 만들기 위해 TinyOS의 부트 시퀀스에 필요한 컴포넌트들을 연결한다.

## 2.2 멀티-홉 네트워크

MultiHopRouter, TinyDiffusion, GPSR, BVR등의 멀티-홉 네트워크 알고리즘들은 경로로 정해질 가능성이 있는 이웃 노드 검색과 관리 테이블 구성이 공통적인 동작이다. 이 정보들은 처음 라우팅 경로를 결정할 때와 주위 노드의 생성 또는 소멸 등의 네트워크 환경이 변화했을 때 경로 재설정에서 사용된다. 이웃 노드 테이블에는 노드의 주소, 연결 상태, 홉수 같은 라우팅 메타데이터를 포함한다. 링크 상태는 라우팅 경로를 결정할 때 사용되며, 홉수는 테이블을 관리하는데 사용된다.

TinyOS에서 초기에 구현된 멀티-홉 라우팅 프로토콜들은 전송 오류가 심했지만[6], 연결 상태 예측기(link state estimator)를 사용한 MultiHopRouter, BVR, TinyDiffusion, TinyOS DSDV에서는 이 문제들이 해결 되었다[7]. 그리고 이들 멀티-홉 라우팅에서는 Send와 Intercept 인터페이스를

사용하기 시작했는데, 패킷 생성에 효율적인 getBuffer 커맨드와 재전송해야 하는 패킷을 수신했을 때 호출되는 Intercept 이벤트가 포함되어 있어서, 노드들은 데이터의 재전송 전에 데이터 처리가 가능해 졌다. 또한 네트워크 스택 하부에 이웃 노드의 패킷을 수신할 수 있는 기능을 추가하여 더 정확한 주변 노드들의 정보를 수집할 수 있다. 전송 큐, 출력 큐, 생성 큐, 전달 큐 등을 추가하여 전송 효율도 높였다.

TinyOS에 구현되어 있는 라우팅 프로토콜들은 초기와는 달리 주변 노드에 대한 테이블을 생성, 다른 노드의 패킷을 수신, 메시지 큐의 사용으로 지연시간, 네트워크 구성 실패, 패킷 손실 등의 단점을 극복하였다.

## 3. 무선 센서 네트워크 기반 조명제어 시스템

### 3.1 무선 스위치

무선 스위치는 TinyOS를 지원하는 Telos 플랫폼을 기반으로 구현하였다. Telos 플랫폼은 CC2420을 기반으로 하고 USB 인터페이스가 지원되는 개발의 효율이 높은 플랫폼이며, 현재 다양한 프로토콜들이 개발 중인 하드웨어이다. 4개의 프로그램 스위치를 지원하도록 설계하였으며, 각 스위치는 인터럽트 방식으로 동작한다. 스위치의 안테나는 보드에 내장된 패치 안테나를 사용하였으며, 거리는 실외에서 100m 내외의 통신거리를 지원한다.

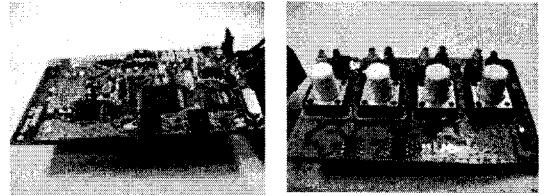


그림 1. 무선 스위치

### 3.2 Light Control Unit

빌딩 조명 제어 시스템은 LCU(Light Control Unit)을 사용한다. 빌딩 내에 설치된 스위치는 유선으로 LCU로 연결된다. LCU는 각 셀 또는 전등을 제어하는 스위치 역할을 한다. 스위치와 LCU는 PLC(Power Line Communication)을 이용하여 통신한다.

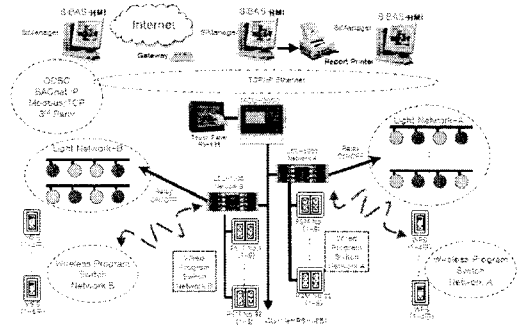


그림 2. LCU 기반의 유선 스위치 네트워크

### 3.3 Access Point

기존에 LCU와 스위치에 사용되던 PLC(Power Line Communication)와 무선 센서네트워크를 적용하기 위해서는 access point가 필요하다. AP(Access Point)의 기능은 무선 센서네트워크와 PLC 간의 데이터 연동이다.

아래 그림의 우측 상단의 AP는 Telos 플랫폼을 기반으로 하고 있으며, UART를 통해 PLC 통신부와 연결된다. 기존의 스위치들은 PLC 통신부를 포함하고 있었지만, 무선 스위치들은 이 AP를 이용하여 PLC와 연결된다.



그림 3. LCU / WSN 연동 시스템

### 3.4 서버 GUI

그림 3의 좌측 상단 기기는 LCU의 설정과 LCU와 컨트롤 서버간의 데이터 연결을 지원하는 마스터 기기이다. 마스터 기기는 이더넷을 통해 LCU와 전등의 상태를 서버로 전송한다. 그림 4는 마스터를 통해 전달 받은 데이터를 GUI로 표현한 것이다. 서버에서는 각 전등 셀의 상태를 모니터링 할 수 있으며, 제어 명령을 내릴 수 있다. 각 건물의 평면도에 표시 도니 전등의 점등 상태 모니터링과 제어가 가능하다.

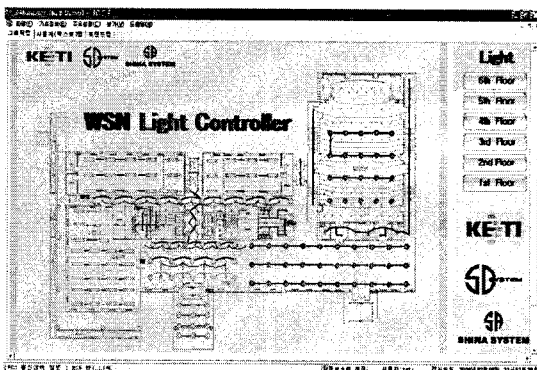


그림 3. 조명 컨트롤 GUI

## 4. 결론

본 논문에서는 유비쿼터스 컴퓨팅 환경에서 물리 공간의 이벤트를 입력받는 센서 노드들의 기능을 추가하여 사용자의 입력을 받아 조명을 제어하는 조명제어 시스템의 구현방법을

제안하였다. 제안한 시스템에서는 Telos 플랫폼을 기반으로 무선 스위치, AP를 구현하였으며, AP는 PLC 통신모듈과 통합되었다.

빌딩용 조명제어 시스템에서 LCU는 추가되어 확장될 수 있는 구조이다. LCU와 AP간의 네트워크 구성은 관리자에 의해서 쉽게 관리될 수 있지만, AP와 스위치는 센서네트워크의 특성상 자동적으로 설정되고 관리 되어야 한다. 네트워크 구성에 있어서는 센서 네트워크가 장점을 갖지만, 대규모 시스템일 경우에 발생할 수 있는 지연 시간 등의 문제를 해결하기 위한 기술이 앞으로 제시되어야 한다. 또한 현재 시스템은 LCU와 무선 스위치 사이만 무선으로 적용하였으나, 향후에는 각 전등과 스위치간의 무선화가 요구된다. TinyOS의 무선 통신 네트워크에서 상대적으로 고려가 많이 되지 않고 있는 신뢰성 있는 전송도 이런 제어를 위한 홈 네트워크 또는 빌딩용 네트워크에서 향후 지원되어야 할 기능이다.

센서 네트워크는 낮은 가격과 무선 통신의 장점 때문에 다양한 분야에 적용될 수 있다. 앞으로 다양한 응용 서비스 환경 구현을 통해, 새로운 기술적 요구사항을 찾아내고 개발하여, 효과적인 상용 서비스를 도출하는 시도가 필요하다.

### 참고문헌

- [1] <http://www.intel.com/research/exploratory>
- [2] J. Hill, R. Szweczyk, A. Woo, S. Hollar, D. E. Culler, and K. S. J. Pister. System architecture directions for networked sensors. In Proceedings of ASPLOS, pages 93-104, Boston, MA, USA, Nov. 2000.
- [3] H. Abrach, S. Bhatti, J. Carlson, H. Dai, J. Rose, A. Sheth, B. Shucker, J. Deng, and R. Han. MANTIS: system support for Multimodal NeTworks of In-situ Sensors. In Proceedings of the 2nd ACM International Conference on Wireless Sensor Networks and Applications, pages 50-59. ACM Press, 2003.
- [4] S. Park, A. Savvides, and M. B. Srivastava. SensorSim: a simulation framework for sensor networks. In Proceedings of the 3rd ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems, pages 104-111. ACM Press, 2000.
- [5] J. Elson, S. Bien, N. Busek, V. Bychkovskiy, A. Cerpa, D. Ganesan, L. Girod, B. Greenstein, T. Schoellhammer, T. Stathopoulos, and D. Estrin. Emstar: An environment for developing wireless embedded systems software. Technical Report 0009, CENS, Mar. 2003.
- [6] D. Ganesan, B. Krishnamachari, A. Woo, D. Culler, D. Estrin, and S. Wicker. Complex behavior at scale: An experimental study of low-power wireless sensor networks. Technical Report 02-0013, UCLA Computer Science Division, Mar. 2002.
- [7] A. Woo, T. Tong, and D. Culler. Taming the underlying challenges for reliable multihop routing in sensor networks. In SenSys '03, Los Angeles, California, Nov. 2003.