

## Mica2 mote 를 이용한 센서 네트워크 테스트 베드 구축

이윤경, 박영수, 전성익  
한국전자통신연구원 정보보호 연구본부

### Sensor Network Test Bed Construction using mica2 mote

Yunkyung Lee, Youngsu park, Sungik jun  
Information Security Research Division  
Electronics and Telecommunications Research Institute  
E-mail :neohappy@etri.re.kr

#### Abstract

Technological progress in integrated, low-power, CMOS communication devices and sensors makes a rich design space of networked sensors viable. These sensors can be deeply embedded in the physical world and spread throughout sensor network environment like smart dust. So ubiquitous computing will be come true. SmartDust project is the one of ubiquitous computing approach. It produces TinyOS, mote(mica, mica2, rene2, mica2dot, etc.), NesC, TinyDB, etc. We constructs sensor network test bed and tests to approach sensor network and ubiquitous computing.

#### I. 서론

post-pc 시대가 도래하면서 컴퓨터 시스템 디자인에 있어서 새로운 영역이 생겨나고 있다. 유비쿼터스(ubiquitous) 또는 퍼베이시브(pervasive) 컴퓨팅도 새로운 영역의 하나라 할 수 있는데, 그 이름에서 알 수 있듯이 편재(遍在)하는, 즉 도처에 존재하는 컴퓨팅 환경을 뜻한다. 현재의 데스크탑 PC 나 노트북 PC, PDA 등의 네트워크 환경을 넘어서서 휴대전화, 가전제품, 게임기 등의 전자제품뿐만 아니라 각종 센서와 라디오가 부착된 나무, 다리, 가로등, 컵 등의 사물이 네트워크화 되어 언제, 어디서나, 누구나 대용량의 통신망을 이용하여 다양한 데이터 수집이나 통신이 가능한 것을 의미한다 [1]. 이러한 유비쿼터스 컴퓨팅 시대를 위하여 미국, 일본, 유럽 각지에서 국가 또는 학교 차원의 프로젝트가 진행중에 있다. 이들 프로젝트들 중 미국 DARPA 의 지원을 받아 UC Berkeley 에서 진행된 프로젝트로 SmartDust 가 있다. SmartDust 프로젝트는 센서와 각종

통신모듈, 메모리 등이 탑재된 먼지처럼 작은 물체를 개발하기 위한 것으로 전쟁 시 생 화학전에 대비해서 전장에 smart dust 를 뿌려놓고 먼 거리에서 그 지역의 대기 상태를 측정하는데 이용할 수도 있고, 농장 관리에 이용할 수도 있다. 또한 숲의 상태, 강물이나 바닷물의 상태, 오염도 등을 측정하는데 이용할 수도 있을 것이다. 현재 개발된 smart dust 는 5 센트 크기의 동전보다 약간 더 큰 크기이지만 기술의 발달로 머지않아 smart dust 라는 명칭에 걸맞은 아주 작은 크기로 만들 수 있으리라 본다. 더 작고, 더 싸고, 더 적은 전력을 소모하는 시스템 개발이라는 현재의 추세를 생각하면 smart dust 의 개발은 더욱 가속화 될 것이다.

Smart dust 의 기본적인 마이크로 컨트롤러는 메모리와 프로세싱 모듈뿐만 아니라 DAC, ADC, UART, 인터럽트 컨트롤러, 카운터 등의 인터페이스 모듈까지도 포함한다. Smart dust 사이의 통신은 유선 및 무선이 모두 가능하고, 무선 통신의 방법에는 RF, infrared, optical 등의 방법을 적용할 수 있다. 사용하는 센서에는 빛, 열, 위치, 움직임, 화학 물질 등을 감지하는 센서가 있다.

본 논문에서는 센서와 RF 통신 및 PC 와의 유선 통신 모듈이 부착된 smart dust 인 mica2 를 이용하여 센서 네트워크 테스트 베드를 구축하고 이를 이용한 실험에 관하여 기술하고자 한다.

#### II. 센서 네트워크

UC Berkeley 의 SmartDust 프로젝트에서는 작은 센서 디바이스 및 이를 이용한 기본적인 센서 네트워크를

구축하고 있다. 센서 네트워크에 사용하는 센서 디바이스는 작고, 값이 싸고, 적은 파워를 소모하여야 하므로 센서 디바이스의 계산 능력과 통신 능력에는 한계가 있을 수 밖에 없다.

센서 디바이스 하나 하나는 센서 네트워크에서 하나의 노드가 된다. 센서 노드는 최초의 PC 와 비슷한 정도의 계산 능력과 저장 용량을 갖고 있는데, CPU는 RISC 머신과 비슷한 특성을 갖는 16 비트 어드레스의 8 비트 프로세서이고 32 개의 레지스터가 있다. 이 프로세서는 2 개의 CPI 를 가지고 3.0V, 4MHz 에서 동작한다. 그리고 명령어의 구조도 아주 제한적이어서 비트 또는 바이트 레벨의 I/O 오퍼레이션은 잘 지원하지만 산술, 논리 연산의 지원에는 부족함이 있다. 총 저장공간은 8KB 의 명령어 플래쉬 메모리, 512 바이트의 데이터 RAM, 512 바이트의 EEPROM 이 있고, 모든 노드는 10Kbps 의 대역폭을 갖는 916 MHz ISM 밴드 라디오로 동작하고, TinyOS 라 불리는 이벤트 중심(event-driven)의 오퍼레이팅 시스템에서 동작한다.

일반적으로 센서 노드는 RF 를 이용하여 통신하기 때문에 데이터의 브로드캐스팅은 아주 기본적인 통신 방법이 된다. 아래 그림 1 은 센서 네트워크의 개념도를 보여준다.

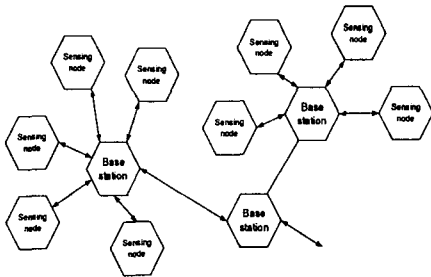


그림 1. 센서 네트워크의 예

네트워크는 하나 이상의 베이스 스테이션(base station) 주위에 형성되는데, 베이스 스테이션은 센서 네트워크와 컴퓨팅 기반시설 사이의 인터페이스가 된다. 센서 노드들은 라우팅 숲(routing forest)을 형성하고 각 트리(tree)의 루트는 베이스 스테이션이 된다. 각 노드는 베이스 스테이션으로 메시지를 전송할 수 있고, 자신에게 어드레스된 패킷을 인지하여 패킷의 명령에 따라 적절한 행동을 할 수도 있으며, 메시지를 브로드캐스트 할 수도 있다. 베이스 스테이션은 다른 센서 노드와 동일한 기능은 물론이고, 다른 센서 노드들의 수명(lifetime)을 증가할 만큼 충분한 전력이 공급되어야 하고, 암호

화 키를 저장할 수 있을만큼 충분한 메모리를 갖고 있어야 하며, 외부 네트워크(PC, 서버 등)와 통신하기 위한 수단을 갖고 있어야 한다. 센서 네트워크 내에서 주로 사용하는 통신 방법에는 다음 세 가지가 있다:노드에서 베이스 스테이션으로의 통신, 베이스 스테이션에서 노드로의 통신, 베이스 스테이션에서 모든 노드로의 통신.

### III. 테스트 베드 구축과 실험

센서 네트워크의 테스트 베드를 구축하기 위하여 UC Berkeley 에서 개발하고 cross bow 사에서 제작한 센서 네트워크 개발 킷을 구입하였다. 센서 네트워크 개발 킷은 센서 노드에 해당하는 여러 개의 mica2 와 mica2dot, 각종 센서가 부착된 센서 보드, PC 와 센서 노드 사이의 통신을 위한 프로그래밍 보드로 구성되어 있다. 그림 2 에는 이들 센서 네트워크 개발 킷의 구성 요소들을 찍은 사진을 보여준다.

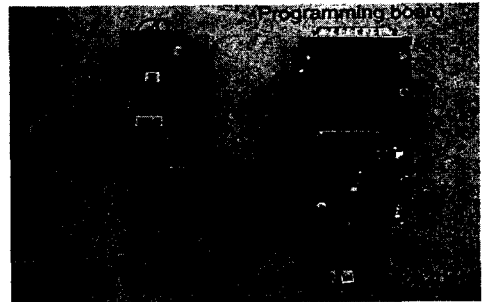


그림 2. 센서 네트워크 개발 킷의 구성

프로그래밍 보드는 병렬 포트(parallel port)와 직렬 포트(serial port)와의 연결이 가능하고, mica2, mica2dot, 센서 보드를 연결하기 위한 연결 잭이 달려 있다. 따라서 mica2 또는 mica2dot 를 프로그래밍 보드의 연결 잭에 꽂고 PC 의 병렬 포트와 프로그래밍 보드의 병렬 포트를 연결한다. PC 에는 TinyOS 가 설치되어 있어야 하고, TinyOS 에서 mica2 또는 mica2dot 의 어플리케이션 프로그램을 mica2 또는 mica2dot 에 인스톨한다. Mica2 는 라디오 송/수신장치, LEDs, UART, 프로세서, 메모리 등으로 구성되어 있고, 센서 보드를 연결 잭에 연결함으로써 센싱 기능을 갖게 된다. Mica2dot 는 간단한 센서가 포함되어 있고, mica2 와 마찬가지로 라디오 송/수신 장치, LED, UART, 프로세서, 메모리 등이 포함되어 있다. 센서 노드의 LED 는 출력장치로 이용되는 때, 간단한 연산 결과 값이나, 센싱 정보를 나타낸다.

센서 네트워크는 단순히 센서 노드와 프로그래밍 보드만으로 이루어진 것은 아니다. 센서 노드를 구동하기 위하여 TinyOS 라는 오퍼레이팅 시스템이 있어야 하고, TinyOS 에 적절한 NesC 라는 C 언어와 비슷한 형태의 새로운 프로그래밍 언어를 이용하여 구현한 어플리케이션 프로그램도 필요하다. TinyOS 와 NesC 는 모두 센서 노드처럼 메모리와 전력이 제한된 시스템에 적용할 수 있도록 간단하고, 프로그램 크기가 작으며, 전력 소모가 적게 만들어졌다. TinyOS 를 PC 에 설치하고, TinyOS 상에서 어플리케이션 프로그램을 인스톨하면 센서노드의 플래쉬 메모리에 프로그램이 업로드 되었음을 알리는 메시지가 뜬다. 그런데 프로그램 업로드 과정에서 "flash error"가 생길 수 있는데, 이는 정확한 원인은 알 수 없으나 주로 센서 노드와 프로그래밍 보드가 연결되지 않았거나, 센서노드와 프로그래밍 보드 모두에 전원을 공급하거나, 전원 공급이 전혀 없을 때 생긴다. 그리고 PC 의 오퍼레이팅 시스템으로 Windows XP 보다는 Window 2000 을 사용하는 것이 센서 네트워크가 더욱 안정적으로 동작함을 확인하였다.

### 3.1 센서 노드 사이의 통신

센서 노드들 사이의 통신을 확인하기 위하여 mica2 에 센서 보드를 연결하고, 포토센서가 수집한 정보를 다른 mica2 노드로 전송하는 실험을 수행하였다. 센서 보드가 연결된 mica2 에는 포토센서가 센싱한 데이터를 ADC 를 이용하여 디지털로 바꾸고 이 값을 mica2 의 LED 로 출력한 후 RF 를 이용하여 다른 mica2 로 전송하는 프로그램을 인스톨한다. 이 프로그램은 물론 NesC 로 구현된 것이고 mica2 의 플래쉬 메모리에 저장된다. 또 다른 mica2 에는 RF 를 이용하여 데이터를 받고, 받은 데이터를 LED 로 출력하는 프로그램을 인스톨한다. 포토 센서가 어두운 곳에 노출되면 붉은색 LED 에 불이 들어오고, 밝은 곳에 노출되면 노란색 LED 에 불이 들어오는데, 이를 센서 보드가 연결된 mica2 보드에서 확인하고, 센서 보드가 연결 되지 않은 mica2 노드에서도 동일한 현상을 확인한다. Mica2 는 900MHz 에서 동작하기 때문에 충분히 먼 거리에서도 포토 센서가 수집한 정보를 받을 수 있다. 그림 3 은 이 과정을 실험하는 사진이다. 그림 3 에서 왼쪽의 센서 보드가 연결된 mica2 에 나타나는 LED 와 오른쪽의 mica2 들에 나타나는 LED 가 일치하는 것을 확인하였다. 또한 데이터를 보내는 mica2 노드에 카운터 값을 LED 로 출력하고 이를 RF 로 전송하는 프로그램을 설치하였을 때에도 오른

쪽의 데이터를 받는 mica2 들에 카운터 값이 LED 로 출력되는 것을 확인 하였다.

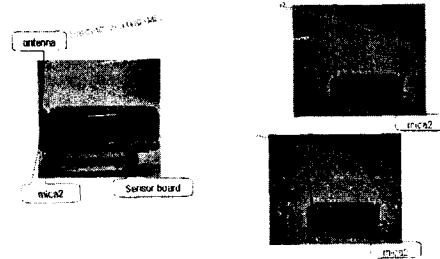


그림 3. 센서의 수집 정보를 다른 mica2 로 전송

### 3.2 센서보드의 수집 데이터를 PC 로 전송

센서 보드가 수집한 정보를 LED 로 출력해서 본다는 것은 아주 원시적인 방법이다. 따라서 PC 와 연결하여 PC 에서 센서 보드가 보내는 정보를 확인하여 보았다. Mica2 노드의 정보를 PC 로 전송하는 데에는 프로그램 보드의 직렬 포트를 이용한다. Mica2 노드와 센서 보드를 프로그래밍 보드에 연결하고, mica2 노드에는 포토센서가 수집한 아날로그 데이터를 디지털 데이터로 바꾸어 이를 LED 및 RF 로 전송하는 프로그램을 인스톨한다. 그리고 PC 에는 시리얼 포트로 들어오는 데이터를 받아서 이를 출력하는 프로그램을 실행한다. "SerialForwarder"라는 프로그램을 이용하여 PC 의 시리얼 포트로 들어오는 데이터를 받고, "SerialForwarder"가 받은 데이터를 모니터에 디스플레이 하기 위해서 "Oscilloscope"라는 프로그램을 이용한다. 이 두 프로그램 모두 자바로 구현된 프로그램이고, TinyOS 에 이미 설치되어 있는 프로그램이다. 그림 4 는 포토 센서가 수집한 데이터를 PC 에서 디스플레이 하는 실험 장면을 찍은 사진이다.

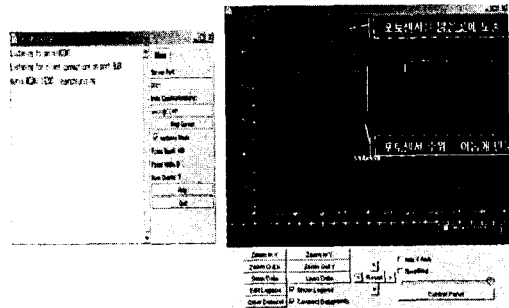


그림 4. 포토 센서의 수집 정보 디스플레이 장면

그림 4 의 왼쪽 창은 프로그래밍 보드의 직렬 포트와 PC 의 COM1 포트가 연결되어 있고, 1 개의 mote로부터

데이터를 받고 있으며, 통신 속도는 19200 baud rate 이고, 읽은 패킷의 수가 증가하고 있는 모습을 보여준다. 오른쪽 창은 시간이 흐름에 따라서 "SerialForwarder"가 받은 데이터를 디스플레이 하고 있는데, 포토센서가 노출된 주위의 밝기에 따라서 화면에 디스플레이 되는 데이터 값이 변화한다. 그래프의 가로축은 시간을, 세로축은 데이터를 나타낸다.

### 3.3 센서 노드로의 명령어 주입과 브로드캐스트

센서 노드가 보내온 데이터를 받아서 출력하는 단계를 넘어서서 PC 를 이용하여 센서 노드로 명령어 패킷을 전송하고, 명령을 받은 센서 노드는 명령을 수행하면서 주위의 다른 노드들로 받은 명령을 브로드캐스트 하는 실험을 하였다. 이를 위해서는 PC 와 센서노드 사이의 게이트웨이 역할을 하는 베이스 스테이션(base station)으로 동작하는 센서 노드가 있어야 한다. 즉, PC 에서 명령어 패킷을 직렬 포트에 보내면 이를 받은 베이스 스테이션은 받은 명령어 패킷을 다른 센서 노드들로 RF 를 이용하여 전송한다. 그림 5 는 이를 실행하는 과정을 찍은 사진이다.

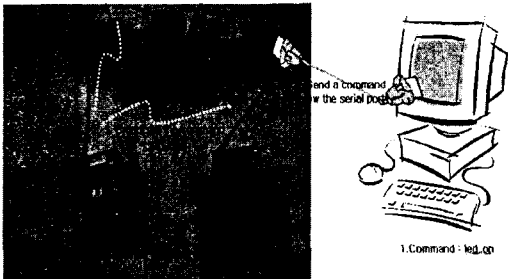


그림 5. PC 에서 red LED 를 ON 하라는 명령어 패킷을 전송하였을 때 센서 노드의 반응

PC 에서 명령어 패킷을 보내서 센서 노드들이 이 명령을 따르도록 하기 위해서 한 개의 mica2 에는 베이스 스테이션으로 동작하는 프로그램을 인스톨하고, 다른 mica2 들에는 RF 로 데이터를 받아서 이를 실행하는 프로그램을 인스톨한다. 베이스 스테이션으로 동작하는 프로그램은 직렬 포트로부터 받은 명령어 패킷을 RF 신호로 전송하고, 또 RF 로 받은 데이터를 직렬 포트를 통해서 PC 로 전송하는 기능이 있다. 그림 5 에서 볼 수 있듯이 PC 에서 빨간색 LED 를 ON 하라는 명령어 패킷을 직렬 포트에 보내면 프로그래밍 보드에 연결된 베이스 스테이션은 이 명령어 패킷을 RF 를 이용하여 다른 센서 노드로 전송한다. 명령어 패킷을 먼저 받은 센서

노드는 붉은색 LED 를 ON 하고 주위의 다른 센서 노드로 명령어 패킷을 전송한다. 그림 5 의 4 번 과정에서 mica2 의 붉은색 LED 가 ON 되었으며, 4 번의 mica2 로부터 명령어 패킷을 받은 5 번 과정의 mica2 는 한 개의 홉을 거쳤음을 보여주기 위하여 초록색 LED 가 ON 되어 있고, 명령어를 수행하여 붉은색 LED 가 ON 되었다. 그리고 4 번과 5 번 과정에서 두 mica2 는 RF 로 데이터 수신을 기다리고 있다는 것을 나타내기 위해서 노랑색 LED 가 깜박이도록 프로그램 되어 있다.

## IV. 결론

본 논문에서는 UC Berkeley 대학에서 구현한 센서 네트워크 테스트 베드를 구축하고 이를 실험한 결과에 관하여 기술하였다. TinyOS 를 PC 에 설치하고 mica2 에 적절한 어플리케이션 프로그램을 업로드 하여 mica2 사이의 통신, mica2 와 PC 사이의 통신, 명령어 주입 등의 과정을 테스트 하였다. 900MHz RF 통신을 이용하기 때문에 충분히 먼 거리에서도 통신이 가능하고, 센서가 모은 정보를 다른 노드로 또는 PC 로 전송하는 것이 가능하다. 또한 PC 에서 센서 노드로 명령어 패킷을 보내고 이 명령어를 받은 센서 노드는 명령어를 실행하는 것이 가능함을 확인하였다. 아직은 mica2 와 mica2dot 모두 크기가 크고 자바와 C 언어에 근거를 둔 NesC 언어를 이용하기에 프로그램 실행 시간이 빠르지 않다는 단점이 있지만, 파워 소모가 크지 않고, 비교적 정확한 RF 통신을 하는 등의 장점도 많다. 크기가 크다는 단점은 반도체 공정 기술의 개발을 통해 수년 내에 해결이 되리라 본다. 또한 프로그램 실행이 느리다는 단점은 NesC 가상 머신의 개발로 해결할 수 있으리라 본다.

## 참고문헌

- [1] 노무라총합연구소, "유비쿼터스 네트워크와 시장창조," 전자신문사, 2002.
- [2] A. Perrig, R. Szewczyk, V. Wen, D. Culler, J.D. Tygar, "SPINS: Security Protocols for Sensor Networks," Mobile Computing and Networking, 2001.
- [3] J. Hill, R. Szewczyk, S. Hollar, D. Culler, K. Pister, "System Architecture Directions for Networked Sensors," ASPLOS, 2000.
- [4] P. Levis and D. Culler, "Mate: A Tiny Machine for Sensor Networks," ASPLOS, 2002.
- [5] <http://webs.cs.berkeley.edu>