

TinyOS 를 이용한 홈 네트워크용 실시간 조도 모니터링 시스템

김문기*, 한병희, 김지홍, 김용현, 이수용, 홍윤식
인천대학교 컴퓨터공학과

e-mail : {moonki2003,byunghee,riot999,yh-kim,bluegenii,yshong}@incheon.ac.kr

A Real-Time Monitoring System of Intensity of Illumination for Home Networks using TinyOS

Moon-ki Kim*, Byung-Hee Han, Ji-Hong Kim, Yong-Hyun Kim, Soo-Yong Lee
and Yun-Sik Hong
Dept. of Computer Science and Engineering, University of Incheon

요 약

TinyOS 기반 무선 센서 노드를 사용한 실시간 계측 데이터 측정 및 제어 기술은 특히 홈 네트워크 분야에 널리 적용되고 있다. 본 논문에서는 ATMega128L 을 장착한 최소 8 대 이상의 Micaz Mote 센서 노드 모듈을 사용하여 각 방의 조도 값을 실시간 측정하는 시스템을 구현하였다. 특히 TinyOS 에서 제공되는 OscilloscopeRF 의 메시지 구조를 분석하여, 각 노드의 ID 및 계측 데이터를 추출하였다. 또한 이렇게 추출된 계측 데이터를 센서 네트워크의 싱크 노드로부터 데이터 통제 센터(Doc)로 효율적으로 전송하기 위한 TCP 기반 네트워크 프로그래밍을 구현하였다. 실험 결과 센서 노드 수 및 샘플링 주기에 상관없이 안정적으로 계측 데이터 수신에 이루어짐을 확인하였다. 한편, IEEE802.11a/g 기반 무선 네트워크를 통해 실시간 계측 상황을 휴대용 단말기인 PDA 에서 확인할 수 있도록 이를 구현하였다.

1. 서론

'전화기를 들고 다닐 수 없을까?'라는 아이디어로 지금의 휴대폰이 탄생되었고, 이로 인하여 '네트워크'라는 단어는 일반 사용자들도 자연스러워 졌다. 보다 편리한 것을 추구 하자는 욕구를 채우기 위해 기술이 발전한다. 사용자의 현재 작업환경의 편의성을 제공하고자 하는 아이디어를 시작으로 발전하는 기술은 사용자의 작업을 대행해 줄 수 있는 에이전트 시스템이 개발되었다. 이러한 에이전트 시스템에 '센서 네트워크 기술'의 도입을 시도하게 되었다[1].

본 논문은 다음과 같이 구성되어 있다. 2 장에서는 연구배경으로 TinyOS 와 응용프로그램 및 NesC 에 대해 설명하고, 3 장에서는 개발환경을 설명하고, 4 장에서는 조도 모니터링 시스템의 하드웨어 및 소프트웨어

구현 및 안정적 계측 데이터 수신 실험을 보여주고, 5 장에서는 결론 및 향후 연구 방향에 대해 기술 하겠다.

2. 연구배경

2.1 TinyOS 와 NesC

TinyOS[2]는 미국 캘리포니아의 UC Berkeley 대학에서 센서 네트워크를 위해서 개발된 초소형 OS 이다. 개발을 빠르게 할 수 있도록 라이브러리가 컴포넌트 구조를 하고 있으며 개발 언어로는 C 언어와 비슷한 NesC 를 사용하게 된다. NesC 를 이용하여 컴포넌트 모듈을 개발하고 사용하여 센서 응용 프로그램을 만드는 것이다[4].

2.2 ListenRaw 응용프로그램의 분석

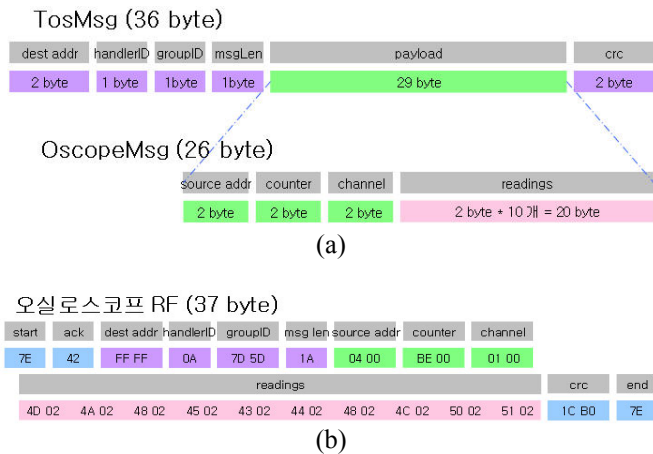
TinyOS 를 설치하면 기본으로 제공되는 Java 응용프로그램으로 Serial 포트(COMx)로 송신되는 데이터들을

* 본 논문은 2005 년도 산업자원부지정 인천대학교 멀티미디어연구센터 지원에 의한 것임

수집하고 동시에 수집된 데이터를 화면에 보여주는 역할을 하는 프로그램이다. 센서가 보내온 메시지를 16 진수 값으로 표현을 한다.

2.3 OscilloscopeRF 의 메시지 구조

OscilloscopeRF 는 TinyOS 를 설치하면 기본적으로 제공되는 NesC 로 개발된 센서 응용 프로그램이다. 해당 응용프로그램은 센서 노드가 조도 값을 측정해서 컴퓨터와 연결된 Base 노드로 측정한 데이터를 무선 메시지로 전송을 한다. 그림 1 은 OscilloscopeRF 의 메시지 구조를 나타낸다.



(그림 1) (a) TosMsg 와 OscilloscopeMsg 의 구조
(b) ListenRaw 로 살펴본 OscilloscopeRF 메시지

ListenRaw 를 사용하여 OscilloscopeRF 메시지 구조를 분석하면 표 1 과 같다.

<표 1> OscilloscopeRF 메시지 분석

표시	크기 (byte)	내용
start	1	0x7E 메시지의 시작
ack	1	0x42 는 ack 신호를 다시 보낼 필요가 없음을 의미
dest addr	2	0xFFFF 는 브로드캐스트
handlerID	1	0x0A 는 OscilloscopeMsg
groupID	2	0x7D 는 디폴트 그룹
msgLen	1	메시지의 길이
counter	2	메시지를 보내면 카운터 증가
source addr	2	노드 번호
channel	2	채널
reading	20	노드가 센싱한 조도 값. 0x0242 와 같이 2byte 씩 데이터를 저장
crc	2	데이터의 유효성 판별
end	1	0x7E 로 데이터의 끝

3. 개발환경

실시간 조도 모니터링 시스템에 사용된 H/W 명세는 표 2 와 같다

<표 2> 조도 모니터링 시스템 H/W 명세

구분	CPU	OS	기타
서버(PC)	펜티엄 4(2.4GHz)	Windows XP Professional	RAM: 1GB LAN: 10/100Mbps
클라이언트	PC	Windows 2000 Server	RAM: 1GB LAN: 10/100Mbps DBMS: MS-SQL
	PDA	Windows CE 4.21	RAM: 128MB LAN: 802.11b
Node	Atmega 128L	TinyOS	RF Chip: CC2420 2.4GHz (IEEE 802.15.4) Data Transfer Rate : 250Kbps

4. 조도 모니터링 시스템 구현

4.1 하드웨어 구성

개발에 사용된 센서의 플랫폼은 micaz mote 를 기반으로 하여 국내에서 제작한 UStar2400 를 사용하였다. UStar2400[3]은 ATmega128L 마이크로컨트롤러를 장착하였고 통신모듈로는 Chipcon 사의 CC2420 을 사용하여 ZigBee(802.15.4) 무선 통신을 하게된다. 센서로는 온습도 센서(SHT11)와 조도 센서(S1087-xx)를 장착하였고 Base 노드의 경우 UART 인터페이스를 통한 시리얼 통신을 하게된다. 그림 2 는 시스템에 사용된 플랫폼이다.



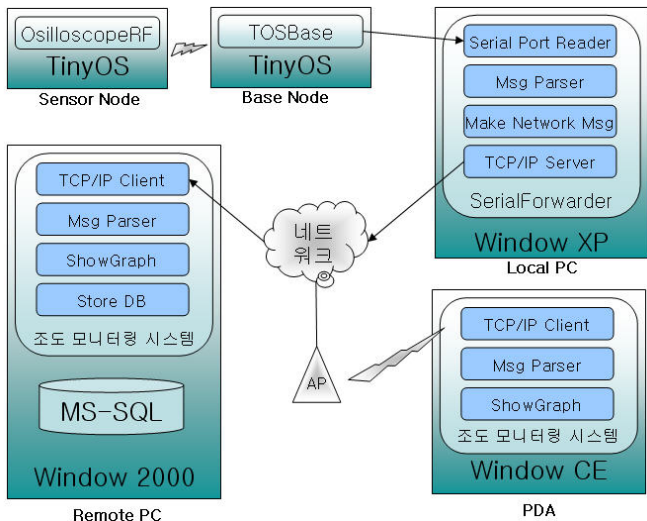
(그림 2) UStar2400 하드웨어 구성

4.2 소프트웨어 구성

그림 3 은 시스템 구성을 도식으로 나타낸 것이다. 조도 모니터링 시스템은 서버/클라이언트 구조를 가진다. 이는 원격지에서 클라이언트를 통해서 실시간으로 센서가 설치된 지역을 모니터링 할 수 장점을 지닌다. 또한 PDA 를 이용하여 AP 가 설치된 지역이라면 이동 중에도 언제든지 센서가 설치된 지역을 모니터링 할 수 있는 또 다른 특징을 가진다.

조도 모니터링 시스템은 크게 세 부분으로 나눌 수 있다. 센서 노드 부분, Local PC 의 서버 부분, 그리고 원격지의 클라이언트로 이루어져 있다. 센서 노드 부분에는 TinyOS 에서 기본적으로 제공해주는 응용프로그램은 OscilloscopeRF 및 Base 노드용 TOSBase 가 사용되어 센서 노드에서 조도 값을 센싱 후 무선으로 전송하게 된다. 그리고 서버 부분은 시리얼 포트에서 데이터를 읽어오는 모듈, 읽어온 데이터를 분석하고

쪼개는 모듈, 필요한 데이터만 모아서 네트워크 패킷으로 만드는 모듈, 그리고 TCP/IP 를 이용한 패킷을 전송하는 모듈로 이루어져 있다. 클라이언트 부분은 PC 와 PDA 로 나뉘는데 공통적으로 TCP/IP 를 통해서 서버로부터 패킷을 받는 모듈, 받은 패킷을 분석하는 모듈, 분석한 데이터를 그래프로 표현하는 모듈을 가지며 PC 의 경우 추가로 Database 에 저장하는 모듈을 가지고 있다.



(그림 3) 조도 모니터링 시스템 도식

그림 4 는 실제 조도 모니터링 시스템을 적용한 모습을 보여주고 있다. 각 Room 에 설치된 센서 노드를 통해서 센싱된 데이터는 Base 노드로 전송하게 된다. 데이터를 서버에서 받아 네트워크 패킷으로 재구성한 후 클라이언트에게 전송을 한다. 클라이언트는 PC 와 PDA 등이 있으며 수신된 데이터를 분석하여 해당 Room 의 상태 및 조도 값을 파악하여 출력을 한다. 또한 센서 노드의 상태를 파악하여 사용자에게 센서 노드 상태 정보를 알려준다.

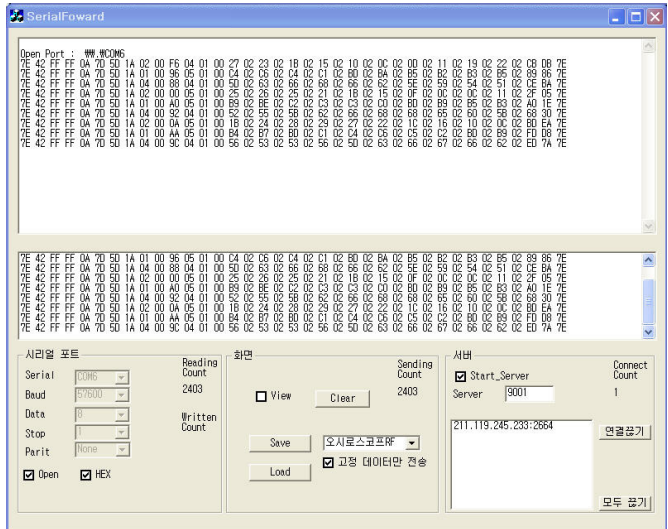


(그림 4) 전체 시스템 구성 모형

4.2.1 서버

TinyOS 패키지 중 Java Program 인 SerialForwarder 를 바탕으로 해서 Visual C++을 이용하여 유사한 프로그램을 개발하였다. 그림 5 는 Visual C++형태로 서버에서는 ListenRaw 와 같은 출력 결과를 얻을 수 있다. 서버는 네트워크로 데이터를 전송하는 역할을 한다.

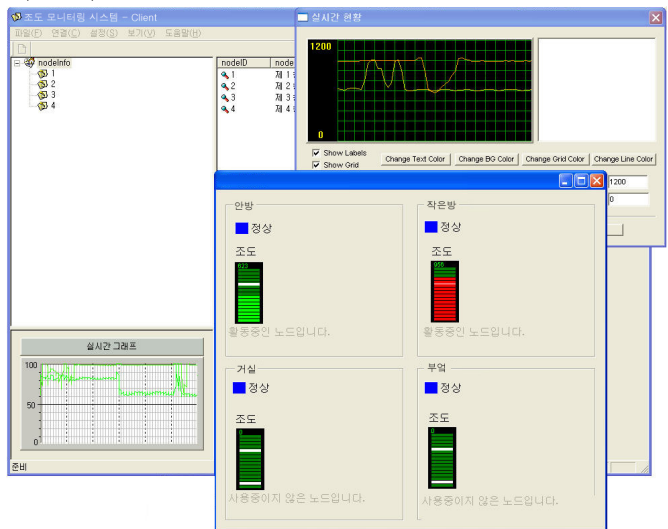
센서 노드가 보내는 데이터를 스스로 가공하여 원격지에 있는 클라이언트로 전송을 한다. 또한 현재 패킷 전송 현황을 볼 수 있다. 그리고 파일저장/불러오기 기능을 통해서 패킷의 구성을 파일로 저장할 수 있도록 개발하였다. 이렇게 함으로써 수신된 데이터에 대한 분석 및 센서 노드의 상황을 서버에서도 원시적으로 모니터링이 가능하도록 만들었다.



(그림 5) 서버

4.2.2 클라이언트

클라이언트는 Local PC 에 있는 서버에 접속을 하여 실시간으로 네트워크를 통해서 센싱된 데이터를 받는다. 네트워크로 받은 패킷을 노드의 ID 및 센싱 데이터를 10 진수로 바꾸어 사용자가 현재 상황을 볼 수 있도록 그래픽으로 표현하였다. 이러한 클라이언트의 기능을 PC 뿐만 아니라 PDA 등 휴대용 기기에서도 수신할 수 있도록 만들어 이동 중에도 원격지의 상태를 파악할 수 있도록 하였다. 만약에 원격지에 있는 센서 노드의 전원이 끊기거나 기타 사정으로 인하여 일정시간 이상 데이터를 전송 받지 못하면 클라이언트는 이를 분석하여 센서 노드의 상태 이상을 경고하게 된다.



(그림 6) PC 용 클라이언트

그림 6은 원격지에 있는 PC용 클라이언트이다. PC용 클라이언트는 각 노드별 한계 값을 적용하여 한계 값 이상/이하가 되었을 경우 붉은색으로 그래프가 표시되도록 하였다. 그리고 시간의 흐름에 따라 변화하는 조도 값을 나타내는 그래프를 추가하였다.

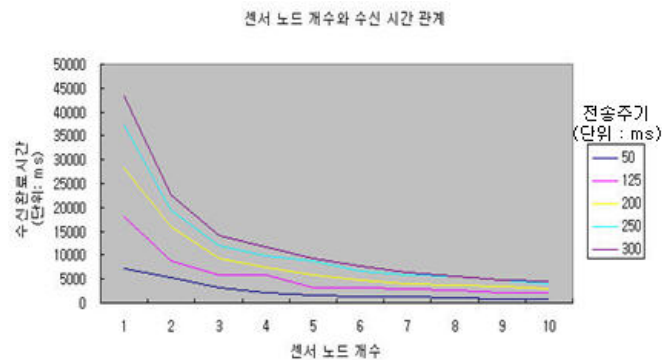
PC용 클라이언트는 웹과 연동을 위하여 센서의 상태 및 데이터 값을 데이터베이스화 하는 모듈이 존재한다. 이에 반해 상대적으로 사양이 낮은 PDA의 경우 그림 7처럼 필수적인 모니터링 기능만을 넣어두었다. 간단히 서버에서 받은 데이터를 표시하면서, 노드의 상태를 표시하는 것으로 그 역할을 제한하도록 하였다.



(그림 7) PDA용 클라이언트

4.3 센서 노드 개수에 따른 수신 시간 관계 분석

센싱된 데이터가 안정적으로 수신되었는지를 알아보기 위한 실험을 그림 8과 같이 하였다. 그림 8을 살펴보면 센서 노드에서 센싱하여 전송하는 시간을 50, 125, 200, 250, 300 ms 단위로 설정하고 10개씩 센싱 데이터를 묶어서 보내는 OscilloscopeRF 응용프로그램을 사용하였다. 서버에서 100개의 데이터(20Byte * 100개)를 수신 완료했을 시간을 측정하였다. 데이터의 신뢰성을 위하여 10회 이상 실험한 데이터 값을 사용하였다.



(그림 8) 센서 노드 개수와 수신 시간 관계 분석

그림 8에서 알 수 있듯이 총 2MByte의 센싱 데이

터를 수신하는데 있어서 노드의 개수가 많으면 많을 수록 수신 완료 시간이 계속 줄어드는 것을 볼 수 있다. 이것은 센서 노드의 수가 노드의 데이터 전송에는 영향을 주지 않는다는 것을 의미하게 된다. 하지만 노드의 개수가 증가함에 따라서 수신 완료 시간의 감소 폭이 줄어드는 것을 볼 수 있는데 이는 Base 노드의 처리 한계 및 대역폭 한계에 가까워져 더 이상 데이터 효율이 증가하지 않는다는 것을 나타낸다. 따라서 센싱된 데이터가 안정적으로 수신을 위해서는 센싱을 하는 노드와 Base 노드가 일정 개수를 유지하는 것이 유리함을 알 수 있다.

5. 결론

유비쿼터스 사회 구현을 위한 핵심 기술 중 하나인 RFID/USN 기술 중 유비쿼터스 센서 네트워크에 관한 기술에 대한 이해를 하기 위해서 먼저 센서네트워크에 필수적이라고 할 수 있는 TinyOS의 구조를 파악하는 것이 중요하다. 본 논문에서는 TinyOS를 이용한 무선 센서 네트워크용 에이전트 시스템인 실시간 조도 모니터링 시스템을 구현하고 그 특징을 살펴 보았다. 이는 실시간 원격 모니터링이 필요한 곳에 본 시스템을 적용할 수 있다는 것을 의미한다. 특히 병원에서 환자의 상태를 모니터링 하거나, 물류 보관소에서 RFID와 연동을 예상하면 물품관리뿐만 아니라 물류 보관소 모니터링까지 되므로 효과적인 시스템이 된다.

또한 센서 노드의 개수와 수신 시간 관계 분석을 통하여 센싱 데이터가 안정적으로 수신 됨을 확인하였다.

향후 연구과제로는 대량의 센서 노드에서의 효과적인 데이터 수신 방법을 연구하고, 에너지 효율적인 ad-hoc 라우팅 연구와 함께 다양한 센서들간의 통합된 메시징 형태를 만들고자 한다.

참고문헌

- [1] 정완영, 이현창, 권성열, 유비쿼터스 컴퓨팅을 위한 센서&인터페이스, 성안당, 2005.
- [2] TinyOS Korea 포럼 : <http://www.tinyos.or.kr>
- [3] ㈜휴인스, <http://www.huins.com>
- [4] <http://www.tinyos.net>
- [5] 박종현 외, “센서네트워크를 이용한 실시간 원격 모니터링 시스템 구현”, 한국인터넷정보학회, 제 6권, 제 2호, 2005. 11
- [6] 신경호 외, “RFID/USN 기반의 능동형 창고 상태 관리 시스템의 설계”, 한국정보처리학회, 제 12권, 제 2호, 2005. 11
- [7] 이상훈 외, “Tiny-DB를 응용한 센서 네트워크 기반의 Web Server 설계 및 구현에 관한 연구”, 한국정보과학회, Vol.32, No.2, 2005. 11