USN 기술 동향 분석

김귀현*, 최명수*, 김하정*, 김기영*, 정민아**, 김서균***, 이성로*
*목포대학교 전자공학과
**목포대학교 컴퓨터공학과
***정보통신연구진흥원
e-mail:srlee@mokpo.ac.kr

Analysis of Ubiquitous Sensor Network Technology

Gwi-Hyun Kim*, Myeong-Soo Choi*, Ha-Jeong Kim*, Ki-Young Kim*,
Min-A Jeong**, Seo-Gyun Kim***, Seong-Ro Lee*
*Dept of Electronics Engineering, Mokpo National University
**Dept of Computer Engineering, Mokpo National University
***Institute for Information Technology Advancement

요 약

센서 네트워크 기술은 센서와 무선 네트워크 기능을 이용하여 물리공간에서 측정한 아날로그 데이터를 디지털 신호로 변환하고, 인터넷 같은 전자공간에 연결된 루트노드로 전달하는 입력시스템 역할을 한다. 물리적 세계와 사이버 세계를 연결할 수 있는 특징 때문에 센서 네트워크의 개념은 새롭게 대두되고 있는 지능형 환경 모니터링, 위치인지 서비스, 지능형 의료시스템, 지능형 로봇 시스템 등 다양한 서비스분야에 적용되고 있다. 이러한 유비쿼터스 센서 네트워크에 대해 하드웨어 플랫폼, 소프트웨어 플랫폼 위주의 기술 동향을 분석해 보고자한다.

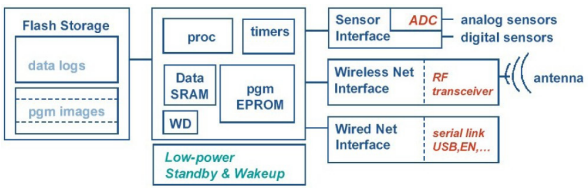
1. 서론

유비쿼터스의 언어적 의미는 라틴어에서 유래한 것으로 “도처에 널려있다”, “언제 어디서나 동시에존재한다”라는 의미로 사용되고 있으며 일반적으로 물이나 공기처럼 도처에 편재해 있는 자연자원이나 종교적으로 신이 언제 어디서나 시공을 초월하여 존재한다는 것을 상징할 때 사용하는 것이다. 이렇듯 다양한 컴퓨터가 현실세계의 사물과 환경 속으로 스며들어 상호 연결되어 언제, 어디서나 이용할 수 있는 인간, 사물, 정보간 최적의 컴퓨팅 환경을 유비쿼터스 컴퓨팅의 정의라 할 수 있겠다[1].

유비쿼터스 컴퓨팅의 개념이 마크 와이저(Mark Weiser)에 의해 정의된 이후 현재의 유비쿼터스 컴퓨팅의 종류는 웨어러블 컴퓨팅, Nomadic 컴퓨팅, Pervasive 컴퓨팅, Calm 컴퓨팅, 감지 컴퓨팅, Exotic 컴퓨팅, 1회용 컴퓨팅 등을 들 수 있다. 유비쿼터스 컴퓨팅은 크게 세 가지로 기술 분류를 할 수 있다. 첫째는 정보이용 관점에서 인식 및 센서기술, 네트워크 기술, 프로세싱(정보 처리)이다. 둘째는 정보 시스템 관점에서 보면 센싱모듈, 상황인지, 인간대 정보의 인공물 인터페이스(u-인터페이스), 백 엔드 시스템(u-응용시스템)이다. 셋째는 정보기술 체계면에서 기초 기술(인증, 보안), 하드웨어 기술(입출력, 기억장치, 통신, 전원), 소프트웨어 기술(OS, 미들웨어), 응용기술(www, JAVA, WAP, XML)이다.

2. USN 하드웨어 플랫폼

유비쿼터스 센서 네트워크에 사용되어지는 센서노드의 Mote 형태는 WeC, Rene, Rene2, Dot, Mica, Mica2Dot, Mica2, Telos의 순으로 발전되어 왔다. 이중 Telos는 산업계 표준으로 채택되어 사용되고 있으며 IEEE 802.15.4 표준을 따르고 있다. 센서 노드는 센서와 무선 네트워크로 구성되어 있으며 무선 네트워크는 RF전송, MCU 등을 포함하고 있어야 하고 Ad-hoc, 토폴로지, 라우팅 등을 지원해야 한다. 또한 무선 프로토콜을 효율적으로 적용할 수 있어야 하고, 센서 인터페이스의 유연성, 저전력의 전원대기, 빠른 wakeup, Watchdog과 모니터링을 지원해야 한다. 아래의 그림 1은 이러한 센서노드의 디자인 요소를 보여주고 있다.



(그림 1) 센서노드 디자인 요소

센서 노드(mote) 측면에서 IEEE 802.15.4의 플랫폼을 살펴보면 기본적으로 저전력에 중심을 맞추어 구성되어 있다. 2.4GHz 대역의 ISM대역의 주파수를 사용한 CC2420 무선통신 방식을 사용하며 250Kbps의 전송률을 갖으며, TinyOS를 지원한다. TI의 MSP430은 초저전력을 구현한 것으로 sleep 1.6mA, active 460mA, 동작전압은 1.8V이다.

아래 표 1은 Telos와 MicaZ의 성능을 비교한 것이다.

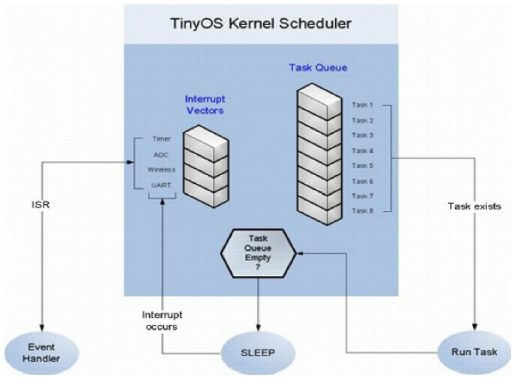
<표 1> Telos와 MicaZ의 성능비교

구분	Telos	MicaZ
Sleep	2.4mA	30mA
Wakeup	290ns, 최대 6ms	60ms, 최대 4ms
Process	4MHz 16bit	8MHz 8bit

3. USN 소프트웨어 플랫폼

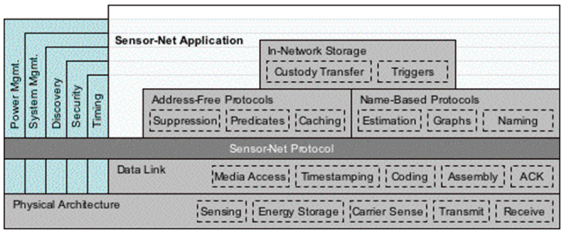
TinyOS의 개발 목적은 세 가지로 설명할 수 있다. 첫째로 미래의 센서 네트워크 노드에 사용될 수 있는 소프트웨어 구조를 설계, 둘째로 여러 하드웨어 플랫폼과 소프트웨어 라이브러리를 사용하여 운영체제와 서비스 어플리케이션을 구현할 수 있도록 설계, 마지막으로 제한된 자원, 동시성, 강인성, 응용 서비스별 요구사항을 만족할 수 있는 설계이다. 이러한 목표의 달성을 위해 TinyOS는 구성이 모듈화 되었으며, 제한된 자원에서 동작할 수 있는 이벤트 구동 방식으로 구현되었다. 모듈 구성에 따라 운영체제가 다양한 하드웨어에 적용이 가능하며, 서비스 어플리케이션도 동일한 추상화를 유지할 수 있다. TinyOS는 이벤트 발생 중심의 상태 전이 방식을 채택한 센서 네트워크용 운영체제로 동시적인 프로세싱 및 제한된 하드웨어 메모리 공간에서의 효율적인 성능을 지원해주는 운영체제이다. 상태 머신 기반의 구조를 가지며 응용 프로그램은 각 독립적인컴포넌트를 연결하는 방식으로 이루어진다. 이 명령을 처리하는 이벤트 처리기는 그 명령에 따라 상태변화를 일으켜 해당 작업을 처리하게 된다. 간단한 FIFO 스케줄러를 사용하기 때문에 실시간성 등을 고려하기 어렵고 복잡한 응용을 작성하는 데 어려움이 있다. 동적 메모리를 할당하지 않지만 NesC라는 언어를 통하여 컴파일 시간에 컴포넌트가 요구하는 메모리 크기를 정적으로 할당하여 센서 네트워크용 응용프로그램을 작성할 수 있다. NesC는 안정성을 위해 전체 프로그램에 대한 분석을 통해 최적화를 수행하는 컴포넌트 기반 언어로 TinyOS의 이벤트 기반 동시성 모델을 가능하게 해준다. TinyOS의 큰 장점으로는 센싱한 데이터를 효과적으로 관리하기 위한 TinyDB나 센서 네트워크에서의 보안을 위한 TinySec 등의 모듈을 지원함으로써 개발자가 손쉽게 다양한 기능을 가진 센서 네트워크를 구현할 수 있게 지원하는 것이다. TinyOS 2.0 이후 버전의 경우 인터페이스와 추상화 과정에 많은 변화가 있어 이전 버전과의 호환성은 지원하지 않지만, 코드를 업그레이드하기가 더 쉬워졌으며 안정성이 향상되었다.

OS의 스케줄러 측면에서 보면 인터럽트 처리를 기본을 하며 복잡한 기능 구현을 위해 기본적으로 스케줄링을 제공한다. 무선 센서 및 네트워크 지원을 하면서도 적은 리소스를 사용하도록 가벼운 OS 코어로 구현되어야 한다. 무선 MAC, 네트워크 프로토콜 구현을 위해 Task Scheduling을 지원해야 하며 단순한 처리는 일반적인 방법으로 처리하고 무선통신, 네트워크 기능 수행에 필요한 스케줄링을 제공하여 리소스 관리를 지원해야 한다. 아래 그림 2는 TinyOS의 커널 스케줄러를 보여주고 있다[2].



(그림 2) TinyOS의 커널 스케줄러

구조적 측면에서 보면 하버드대학에서 연구한 Narrow waist 프로젝트를 예로 들 수 있다. 이는 일반적이고 효율적 측면에서 데이터링크 계층과 네트워크 계층에서 구조를 변경하였고, 싱글 홉(single hop)을 지원한다. 아래의 그림 3은 하버드대학의 Narrow Waist 프로젝트의 소프트웨어 구조를 보여주고 있다[3].



(그림 3) 하버드대학의 Narrow Waist 프로젝트

4. 결론

새로운 컴퓨팅 환경의 도래와 유비쿼터스 컴퓨팅의 핵심기술 등의 발전을 통한 USN의 많은 이점과 다양한 응용 분야를 통해 많은 연구들이 진행되고 있으며, 기술시장에 뛰어들고 있는 상황이다. 이는 USN기술과 표준화가 이제 시작단계라고 할 수 있기 때문에 그만큼 기술의 선점이 중요하다고 할 수 있다.

감사의 글

본 연구는 교육과학기술부와 한국산업기술재단의 지역혁신인력양성사업과 지식경제부 및 정보통신연구진흥원의 대학IT연구센터 지원사업(IITA-2009-C1090-0902-0010)의 연구결과로 수행되었습니다.

참고문헌

[1] 김태형, 전광일, 최종무, 홍성수, “유비쿼터스 컴퓨팅을 위한 통합 소프트웨어 구조”, 정보과학회지 제21권 5호, 2003년 5월, pp.51-60.

[2] http://www.tinyos.net

[3] http://www.eecs.harvard.edu