특집

센서네트워크 하드웨어 플랫폼 및 운영체제 개발 동향

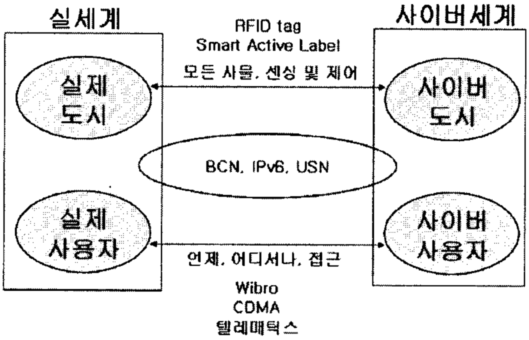
소선섭 (공주대학교), 은성배 ((주)옥타컴)

요 약

본 기고문에서는 센서네트워크 기술 중에서 하드웨어 플랫폼과 운영체제에 대한 현황을 기술한다. 하드웨어 플랫폼 및 운영체제의 경우 시장 초기진입 단계이므로 연구 개발 중인 플랫폼보다는 상용화된 플랫폼들을 소개하는데 주력한다. 또한, 상용화된 플랫폼을 이용하여 ZigBee 통신 성능을 실험하고 분석하였다. 먼저 ZigBee의 최대 전송률을 실험하였으며 무선 LAN과의 주파수 충돌시의 전송률을 실험하였다. 실험 결과, 칩수준에서의 최대 전송률은 250Kbps라고 하나 실제 성능은 이보다 낮은 150Kbps를 기록하였다. 무선LAN과의 충돌 시에는 10~20% 정도의 성공률만을 보이며 실내에서 무선 LAN의 영향을 더욱 많이 받는다는 결과를 얻었다.

Ⅰ. 서 론

최근 IT 산업에서 유비쿼터스 컴퓨팅에 대한 관심이 증가하고 있다. 유비쿼터스 컴퓨팅이란 소형 컴퓨터를 여기 저기 배치시켜 우리의 일상생활을 효율적으로, 편리하게 유지, 관리하는 것이다. 이러한 유비쿼터스 시대를 위한 기반 기술로서 무선통신을 지원하는 센서네트워크의 중요성이 커지고 있다. 그림 1은 유비쿼터스 시대에서 실세계와 사이버세계의 관계, 그리고 센서네트워크의 역할 및 기타 통신망들의 관련성을 보여준다.



〈그림 1〉 실세계와 사이버 세계의 관계

그림 1에서 실세계에는 실생활공간과 실사용자가 존재한다. 이 실세계의 현 상태를 RFID나 USN을 이용하여 사이버세계의 사이버도시나 사이버사용자에게 사상한다. 실사용자는 기존의 인터넷 등의 유선망 및

Wibro, CDMA, 텔레매틱스 등의 무선접근망을 이용하여 언제, 어디서나 사이버 사용자로 로긴한다. 사이버세계에서의 판단이나 결정이 센서네트워크의 구동기를 거쳐 실세계의 상태 변화에 반영된다. 그림 1처럼 센서네트워크는 유비쿼터스 사회에서 실세계와 사이버 세계와의 상태 변이를 상호 전달하는 중요한 역할을 수행한다.

본 기고문에서는 센서네트워크 기술 중에서 하드웨어 플랫폼과 운영체제에 대한 현황을 기술한다. 센서네트워크 하드웨어 및 운영체제 분야는 기반 기술 개발 단계를 지나 상용화된 통신 칩이나 하드웨어 플랫폼이 판매되는 초기 시장 진입 단계이다. 이를 고려하여 연구 개발 중인 플랫폼보다는 상용화된 플랫폼들을 소개하는데 주력한다.

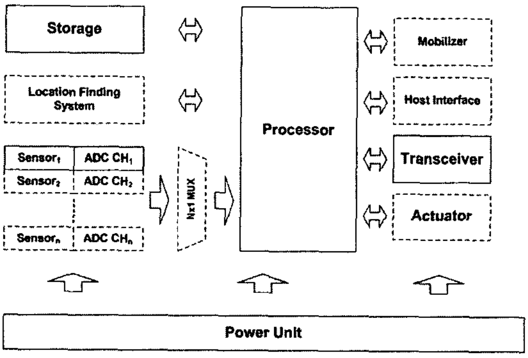
또한, 상용화된 플랫폼을 이용하여 ZigBee[1] 통신 성능을 실험하고 분석하였다. 먼저 ZigBee의 최대 전송률을 실험하였으며 무선 LAN과의 주파수 충돌시의 전송률을 실험하였다. 실험 결과, 칩 수준에서의 최대 전송률은 250Kbps라고 하나 실제 성능은 이보다 좀 낮은 150Kbps를 기록하였다. 무선LAN과의 충돌시 에는 10~20% 정도의 성공률만을 보이며 실내에서 무선 LAN의 영향을 더욱 많이 받는 결과를 얻었다.

Ⅱ장에서 센서 노드의 구조 중, 하드웨어와 운영체제의 구조를 기술한다. Ⅲ장에서 현재 상용화되어 있는 무선 통신 칩의 특성과 하드웨어 플랫폼, 그리고 운영체제에 대하여 기술한다. Ⅳ장에서는 최근, 각광을 받고 있는 통신 방식인 ZigBee MAC 프로토콜에 대한 실험을 기술하고 그 결과를 소개한다. Ⅴ장에서 결론 및 향후 연구 방향을 기술한다.

Ⅱ. 센서 노드 개념도

1. 하드웨어 플랫폼

센서노드를 구성하는 하드웨어 컴포넌트로는 그림 2 와 같이 센싱 유닛, 프로세싱 유닛, 송수신 유닛, 그리고 파워 유닛 등 크게 4 가지로 구분된다. 또한 응용 분야에 따라 추가로 장착할 수 있는 모듈 형태의 컴포넌트로는 위치인식 시스템, 전원 생성 장치, 모빌라이저, 호스트 인터페이스, 구동기(actuator), 외부 메모리 등이 있다.



〈그림 2〉 센서노드의 하드웨어 컴포넌트 구조

- 센싱 유닛

센싱 유닛은 센서와ADC(Analog to Digital converter)부로 나눌 수 있다. 센서에서 감지된 물체의 움직임이나, 온도, 습도, 빛 등의 아날로그 신호를 디지털로 ADC를 이용해서 바꾸는 역할을 수행한다.

- 프로세싱 유닛

센싱 유닛을 통해 수신변환된 디지털 신호는 프로세싱 유닛으로 전달이 된다. 프로세싱 유닛은 다른 센서 노드와의 통신이나 자료의 처리 등을 담당하며, 다른 센서 노드와 유기적

인 작업을 하는데 필요한 연산 등을 수행한다.

- 송수신 유닛

송수신 유닛을 통해서 센서 노드가 센서 네트워크에 연결이 된다.

- 파워 유닛

센서 노드에 전원을 공급해 주는 유닛으로, 응용 분야에 따라 전지나 연료셀, 태양전지 등이 파워 유닛으로 사용될 수 있다.

- 위치인식 시스템

위치인식 시스템은 센서노드의 절대적, 상대적 위치를 인식하는 모듈로서 GPS나 초음파, 전파 세기 등을 이용하여 다양한 방식으로 구현할 수 있다. 센서네트워크에서 위치인식은 매우 중요한데 위치인식을 통하여 다양한 트랙킹, 상황인지 및 가타 서비스를 제공할 수 있다.

- 전원 생성 장치

기본적으로 배터리를 사용하여 센서노드를 동작시키지만 태양광이나 풍력 같은 전원생성장치와 결합한다면 좀 더 유연한 응용을 제작할 수 있다.

- 모빌라이저

센서노드의 이동을 수행하는 장치이다.

- 호스트 인터페이스

센서노드의 소프트웨어 개발이나 상태 점검을 위하여 호스트 컴퓨터와의 연결을 담당하는 모듈이다. SPI, Infrared, USB, 이더넷 등 다양하다.

- 구동기

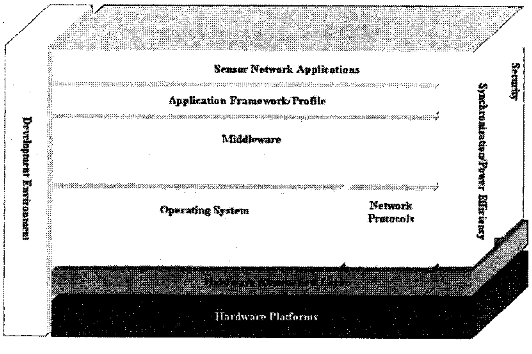
센싱된 데이터를 처리하여 제어대상의 상태를 변경할 수 있는 디바이스를 말한다. 모터, AC/DC 스위치, 디지털 출력, 음성 출력 등 다양하다.

- 외부 메모리

센서노드의 프로세싱 유닛에는 프로그램 메모리 및 데이터 메모리, 그외 플래시, EEPROM 등이 구비되어 있으나 데이터 로깅 등의 특정 응용의 경우 외부 확장 메모리를 요구할 수 도 있다. 전원이 없어도기억을 유지하는 비휘발성 메모리 등이 적절하다.

2. 센서노드 소프트웨어 구조

센서노드를 구성하는 소프트웨어 컴포넌트로는 그림 3 과 같이 운영체제, 네트워크 프로토콜, 미들웨어, 전력관리, 보안, 위치인식 및 클럭 동기화 등이 있다. 그러나 이러한 소프트웨어 컴포넌트 중에는 하드웨어적으로 구현될 수 있는 부분이 있으므로 하드웨어와 소프트웨어의 구분이 엄격하지는 않다.



〈그림 3〉 센서노드의 소프트웨어 컴포넌트 구조

- 센서노드 운영체제

센서노드는 마이크로 컨트롤러를 내장한 소형 컴퓨터 시스템으로 다양한 센싱 응용처리와 노드들간 통신등을 위하여 운영체제를 필수적으로 요구한다. 센서노드에 있어서 운영체제는 다음과 같은 기능을 포함하여야 한다.

- 제한된 메모리와 CPU 자원을 가지는 센서 노드에 적합한 태스크 구조 및 관리 기능
- 센서노드 내부 및 센서노드 간 에너지 효율적이며, 센서 네트워크 응용에 적합한 메시지 처리 기능
- 표준화된 HAL 및 디바이스 드라이버 기능
- 클럭 동기화 및 위치인식 기능
- 저전력 관리 기능
- 표준 센서노드 운영체제 API 기능
- 센서 응용 개발 환경

특히 유비쿼터스 센서 네트워크에서 각각의 센서는 다목적으로 사용되고, 장소를 가리지 않고 다수의 센서를 설치하기 위하여 소형화, 상호 간섭의 방지, 센서끼리의 연계 등이 요구되므로, 센서노드 운영체제 설계시 다음 사항이 고려되어야 한다.

- 자원이 제한적인 센서노드 하드웨어에서 수행되어야 하므로, 작은 크기여야 하고, 전력 소모가 적어야 하며, 센서노드간에 저전력 통신을 제공하면서도 프로세스와 메모리를 효율적으로 관리하여야 한다.
- 다수의 하위레벨 이벤트 발생과 상위레벨의 프로세싱이 상존하므로, 이들을 동시 처리할 수 있는 정밀한 동시성 (Concurrency) 처리 기능이 요구된다.
- 센서노드는 일반적으로 제한된 병렬 처리 능력과 하드웨어 제어 구조를 가지며, 센서와 같은 하드웨어 I/O 장치를 지능형 제어기가 아닌 원시적인 직접 접근 방식에 의해 제어를 하므로, 이러한 I/O 제약 사항이 운영체제 설계시 고려되어야 한다.
- 다양한 응용 분야를 가지는 센서 네트워크에서는 범용 하드웨어와 소프트웨어가 존재하는 것이 아니라 응용 분야에 따라 크게 달라질 수 있으므로, 운영체제는 가능한 한 어떠한 하드웨어 구조에서도 효과적으로 사용할 수 있도록 유연성과 모듈성을 갖추고 있어야 한다.
- 센서 노드들은 한번 배치가 되고 나면, 유지 보수가 어렵고, 운용 환경 또한 열악할 수 있으므로, 이들을 고려하여 강인한 구조로 설계되어야 한다.
- 센서 네트워크의 핵심 프로토콜인 Ad-hoc 라우팅 프로토콜과 프로그래밍 용이성이 고려되어야 한다.

- 센서 네트워크 프로토콜

센서 네트워크 프로토콜은 센서노드간 네트워크를 구성하여 메시지를 송수신하고 각 노드의 역할에 따른 센서 네트워크 작업 및 조정 역할을 수행하기 위하여 저전력 Ad-hoc 라우팅[2] 기능, 고장 대처 및 자가 복구 기능, 이동성 지원 기능, 그리고 다른 네트워크 프로토콜과의 상호연동 기능 등을 포함한다. 이러한 센서 네트워크 프로토콜은 주로 1-3 계층을 다루며 물리계층은 주로 RF를 사용하며 데이터 링크 및 네트워크도 저전력을

고려하여 구현된다. 문제는 인터넷 프로토콜과 연동되지 않는 구조라는 점이다.

III. 센서노드 개발 동향

1. 무선 통신 칩 개발 동향

센서노드를 위한 통신 프로토콜은 아직 표준이 정해지지 않았다. RF를 사용한다지만 주파수도 확정된 것이 없고, 통신 방식도 확정된 것은 없다. 다만 통신거리, 저전력 특성, 응용의 성격에 따라 RF, Infrared, UWB[3] 등이 거론되고 있을 뿐이다.

그 중에서 ZigBee는 참여하는 업체의 영향력, 상용화된 통신 칩, 데이터링크, 네트워크 계층의 프로토콜의 완비 등, De Facto 표준으로서 관심을 집중하고 있다. 본 기고에서는 ZigBee 프로토콜을 지원하는 상용 칩들의 특징을 기술하고 비교 분석한다.

표 1은 ZigBee 상용 통신 칩의 제원을 표시하고 있다. 공통적으로 3 볼트 구동전압을 사용하며 Jennic 사의 제품을 제외하면 모두 별도의 MCU를 사용한다. cc2420과 em2420이 동일한 제원을 갖는데 이는 Ember사가

〈표 1〉 ZigBee 무선 통신 칩 제조 동향

항목 \ 모듈		CC2420 (Chipcon사)	EM2420 (Ember사)	MC13192/3 (Freescale사)	JS24Z121 (Jennic사)
RX sensitivity		-94dBm	-94dBm	-92dBm	-93dBm
Tx Power		Min:-3dBm Typ: 0dBm	Min:-3dBm Typ:0dBm	Min:-3dBm Typ:0dBm Max:3.6dBm	+1dBm
Current Consumption	RX	20mA	20mA	42mA	50mA
	TX	17mA	17mA	35mA	40mA
Operating Voltage		2.1~3.6V	2.1~3.6V	2.0~3.4V	2.2V~3.6V
CPU ON Chip 유/무		무	무	무	On Chip (32Bit CPU)
MCU와 Interface		SPI	SPI	SPI	On Chip
Package type		QLP-48 package, 7 x 7 mm	QLP-48 package, 7 x 7 mm	QFN-32	8x8mm 56 lead QFN

Chipcon사의 소프트웨어 개발팀이 분사한 회사이며 em2420이 cc2420과 동일한 칩이라는 점을 보여준다.

전력소모면에서 cc2420이 여타 칩들보다 적은 전력을 소모하는 것을 보여준다. 대신에 송신 출력면에서 상대 칩들보다 낮다. (주)옥타컴이 상용화한 Nano-24 모듈(cc2420 장착)을 이용하여 최적의 조건 - 외부 장착 안테나를 장착하여 개활지에서 실험 - 에서 실험했을 때 500미터 이상의 전송거리를 보여주었는데 Freescale사의 칩이나 Jennic사의 칩이 더 먼 거리를 통신할 수 있을 것으로 추정할 수 있다.

패키지 크기는 제품 개발에서 매우 중요한데 Freescale사의 제품이 32 핀으로서 매우 작으며 Jennic사 제품의 경우 32비트 CPU를 내장하면서도 8mm * 8mm 크기인데 소형의 통신 모듈을 제작하는 데 매우 적절하다는 것을 보여준다.

결론적으로 저전력 소모면에서는 Chipcon사 제품이 가장 우수하나 전송거리가 다소 짧으며 전력소모를 고려하지 않을 때에는 Freescale사의 제품이 가장 우수한 통신 성능을 보인다고 말할 수 있다. Jennic사의 제품은 단일 칩에 32비트의 MCU와 통신 칩을 내장했다는 장점을 갖는다.

Chipcon사의 경우, cc2430이 8051 코어와 cc2420을 포함한 단일칩 솔루션을 곧 출시한다고 하며 국내의 RadioPulse사가 올해 말까지 단일 칩 솔루션을 출시한다고 한다. 아직 시장 초기단계이므로 각사의 제품들 간의 경쟁력은 좀 더 지켜봐야 할 것이다.

2. 플랫폼 개발 동향

2000년 이후로 센서네트워크 연구가 활성화되면서 세계적으로 많은 하드웨어 플랫폼 연구가 진행되었다. 미국의 경우, UC Berkeley 대학에서 Mems 기술을 활용하여 먼지처럼 작게 만든 Smart Dust[4] 연구가 진행되고 있으며, 그 중간 결과로서 mote, mica[5] 등이 만들어져서 Crossbow[6]에서 상용화하였다. 일본의 경우에는 사카무라 켄 교수의 주도하에 TRON 프로젝트[7]가 진행되었고 크기별로 표준 T-엔진, m-T엔진, n-T 엔진, p-T엔진 의 4 종류를 지정하여 산업체에서 생산하도록 하고 있다. 유럽에서도 Smart-ITS[8] 프로젝트의 결과 다양한 하드웨어 플랫폼들이 개발되었다.

국내의 경우, ICU에서 ANTS 플랫폼[9]이 개발중에 있으며 ETRI에서 2004년 ZigBee기반의 플랫폼을 개발완료 하였다. 그 결과를 (주)옥타컴에서 상용화하여 판매중에 있다. 표 3은 상용화된 하드웨어 플랫폼의 형황을 보여준다. 2003년 이후 ZigBee를 지원하는 플랫폼들이 개발되어 판매되고 있다. 옥타컴의 Nano-24의 경우 하드웨어 제원은 Crossbow와 유사하나 운영체제로서 TinyOS[5]와 자체 개발한 Sensos, ETRI가 개발한 Nano-Q+까지를 모두 지원하는 것이 특징적이다.

〈표 3〉 하드웨어 플랫폼의 상용화 현황

제품명 (회사명)	mica2 (Crossbow)	imote	Telos (Crossbow)	Nano-24 (옥타컴)
출시년도	2003	2003	2003	2004
클럭스피드	7MHz	12MHz	20MHz	12MHz
CPU	ATmel	ARM	Motorola	ATmel
플래시(KB)	128	512	60	128
RAM(KB)	4	64	4	4
무선통신(KBaud)	40	460	250	250
Radio Type	Chipcon	Zeevo BT	Zigbee	Zigbee
주파수 대역	433MHz	433MHz	2.4GHz	2.4GHz
운영체제	TinyOS	TinyOS	TinyOS	TinyOS, Nano-Q+, 및 Sensos

3. 운영체제 개발 동향

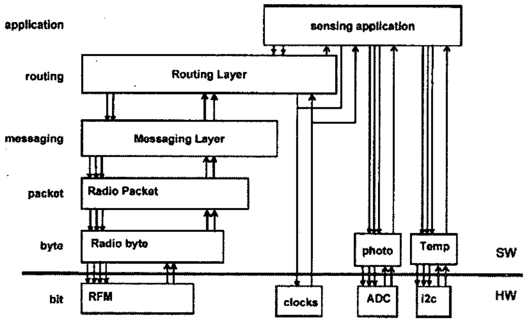
1) TinyOS

2001년 6월부터 버클리 대학은 Networked Embedded Software Technology(이후 NEST)라는 DARPA 프로젝트를 수행했다. NEST 프로젝트의 목적은 네트워크 임베디드 시스템들을 위한 다음과 같은 특징을 갖는 플랫폼을 개발하는 것이며 그 결과물이 바로 TinyOS[9]이다.

- 극소형 프로세서에 적용가능
- 대규모 네트워크 구성 가능
- 저전력 동작이 가능
- 모듈성을 제공
- 안정적으로 동작하는 시범 플랫폼

TinyOS의 특징은 컴포넌트 기반의 모듈이 프로그램의 근간이며 모듈간의 Command 및 Event를 통해서 데이터의 전송이 이루어지는 Event Driven 방식의 운영체제라는 점이다. 컴포넌트의 구성은 Frame (Storage), Tasks, Command 및 Event의 Interface로 이루어져 있다. 또한 이런 컴포넌트구성 및 연결은 nesC(Network Embedded System C)언어코드로 작성되어져 있다. ncc는 이 nesC언어의 컴파일러이고 nesC란, 소프트웨어를 마치 하나의 하드웨어처럼 기술하는 언어라 말할 수 있다. 예들 들면 메모리 칩들과 메모리 컨트롤러를 정해진 인터페이스로 연결하여 메모리 모듈을 만든다고 보면 된다.

TinyOS 컴포넌트 모델들은 소프트웨어적으로 하드웨어를 추상적으로 표현하였기 때문에 개발자들이 쉽게 모듈들을 짜맞추어 쉽게 하나의 Sensor Mote 응용을 만들 수가 있다. 다시 말해 통합 및 교체 등이 용이한 객체 관점에서 기술되어진 운영체제라 볼 수 있다.



〈그림 4〉 SensorMote에서 동작하는 모듈간의 통신들

그림 4에서와 같이 상자로 묶여있는 것이 하나의 모듈들이고, 이들 간에 이루어지는 통신들이 모듈 간 인터페이스이면서 실제로 이들 간의 데이터는 Command 및 Event가 발생하게 되는 것이다.

TinyOS 기능들 중에서 가장 두드러진 점은 저전력 소비를 위하여 설계되었다는 것인데 이 때문에 전세계의 많은 개발 그룹들이 이용하고 있다. 또한, 데이터베이스 엔진인 TinyDB, Virtual Machine인 Bombillia, 시뮬레이터TOSSIM, TinySec과 같은 시큐리티 모듈들이 제공되고 있다.

2) Nano-Qplus 및 Mantis OS

2004년 말, ETRI의 편재형미들웨어 연구팀에서 Nano-Qplus 라는 센서네트워크 운영체제를 개발하였다. Nano-Qplus의 특징은 다음과 같다.

- 센서 및 구동기의 종류에 따라 OS 커널을 최적화하여 재구성 가능한 scalable OS를 제공(TinyOS 보다 향상된 S/W 구조 지원)
- 멀티스케쥴러를 지원하여 다양한 무선통신 방식(433MHz RF, Zigbee)을 지원
- 표준형 및 마이크로 임베디드 OS와 동일한 API set을 지원(POSIX 표준 기반)

Nano-Qplus는 (주)옥타컴에서 협력하여 개발한 Nano-24라는 플랫폼에서 동작하고 있으며 2005년 6월말에 소프트웨어를 공개할 예정으로 있다.

〈표 2〉 Nano-Qplus와 TinyOS의 비교

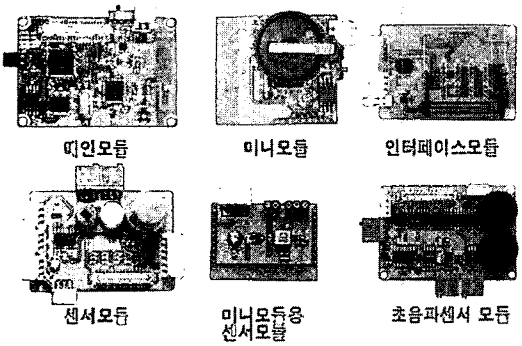
	Nano-Qplus	UCBerkely TinyOS
크기조정이 가능한 재구성 기능	O	X
무선통신 모듈	O	O
에너지 효율적인 스케쥴러	O	O
시간동기화 기능	O	X
스마트센서노드 기술	O	X

콜로라도 대학의 MANTIS[10]는 응용개발 프로세스의 단순화를 통하여 센서네트워크를 효율적으로 구축하기 위한 Multi-Modal 개발환경을 제공하려는 목적으로 C-Style 프로그래밍이 가능하고 X86 및 ATmega128에서 수행가능한 Mantis 라는 OS를 제공하고 있다.

IV. 하드웨어 플랫폼 성능 실험 및 분석

본 절에서는 ZigBee MAC 프로토콜의 성능 실험 및 결과에 대하여 기술한다. 실험에 쓰인 장비는 옥타컴의 Nano-24[11]이며 그림

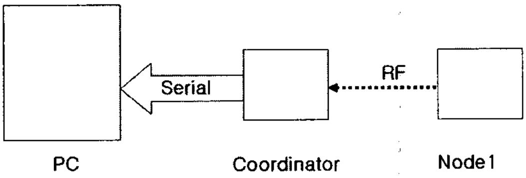
5에서 볼 수 있다.



〈그림 5〉 Nano-24 하드웨어 플랫폼

NANO-24는 Atmel사의 ATmega 128L을 MCU로 사용하고 RF는 Chipcon사의 CC2420을 사용한다. 조도, 온도, 습도, 적외선, 가스, 초음파, 가속도 등 다양한 센서를 제공하며, 확장성 또한 좋아서 여러 가지 응용에 적용할 수 있다. NANO-24는 Zigbee (802.15.4)를 지원하고, TinyOS S/W 솔루션과 ETRI에서 개발한 센서 네트워크 개발 환경인 NANO-Q+와 호환이 가능함에 따라 센서 네트워크를 구성하고 개발하는데 강력한 개발환경을 지원해준다. 성능 실험을 통하여 최대 전송률을 구하였으며 같은 주파수 대역을 사용하는 무선 LAN과 동시에 데이터 전송을 할 때 전송률을 구하였다.

1. MAC 최대 전송률



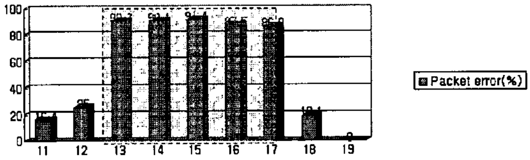
〈그림 6〉 최대 전송률 실험 설정

MAC 최대 전송률 측정을 위한 설정은 그림 6과 같이 각각 한 개의 Coordinator와 Node, 그리고 PC로 구성된다. Node1은 127Byte(1 패킷의 최대 크기)의 데이터를 RF를 이용하여 계속해서 Coordinator로 전송한다. Coordinator는 Node1 으로부터 전송되는 데이터를 카운트하여 전송률을 계산한다.

실험의 조건은 1) cc2420 칩에 직접 명령을 내렸을 때, 2) Nano-Q+ 상에서 통신할 때, 3) TinyOS상에서 통신할 때, 인데 결과는 각각 220Kbps, 153Kbps, 132Kbps의 결과를 얻었다. 하드웨어 칩에 직접 명령을 내렸을 때에는 충돌 방지를 위한 MAC 프로토콜의 임의 지연시간이 없으므로 더 좋은 결과를 낸 것이다. 2)와 3)의 실험 결과로부터 Nano-Q+가 TinyOS 보다 통신 지연시간이 약간 적다는 것을 알 수 있다.

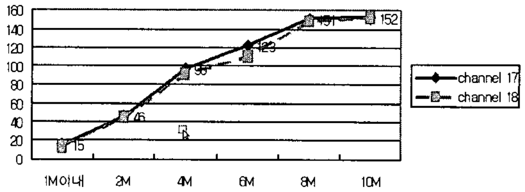
2. 무선 LAN과 충돌 시험

실험 방식은 두 대의 컴퓨터를 이용하여 무선 LAN(IEEE802.11)을 ad-hoc으로 연결하고 ftp를 이용하여 5Mbps의 트래픽을 발생시킨다. ZigBee(IEEE802.15.4)는 무선LAN 연결 바로 옆에서 coordinator와 device간에 연결과정 이후의 통신으로, 127byte 패킷을 최대 전송 속도로 1000 패킷 단위로 데이터를 보낼 때의 시간과 error율을 20회 반복 실험하였다.

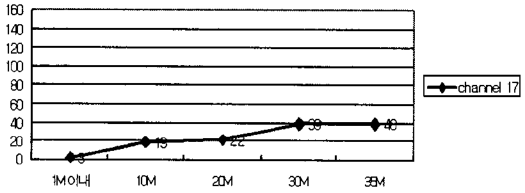


〈그림 7〉 채널별 에러율

그림 7에서 ZigBee의 채널 11, 12, 18, 19는 큰 영향을 받지 않으나 13~17까지의 채널은 거의 전송이 이루어지지 않는 것을 확인할 수 있다.



〈그림 8〉 개활지에서의 충돌 실험



〈그림 9〉 실내에서의 충돌 실험

그림 8은 개활지에서의 충돌 실험결과를 보여준다. 그림에서 볼 수 있는 것처럼 ZigBee 통신 모듈이 무선 LAN 통신 장비들과 8미터 이상 떨어질 경우 거의 정상 동작하는 것을 알 수 있다. 반대로 그림 9는 실내에서의 충돌 실험을 보여주는데 거리가 40미터 이상 벌어져도 무선LAN의 영향을 꾸준히 받는 것을 볼 수 있다. 실내에서는 무선 LAN의 전파가 반사등에 의해 실내에 충만해져서 먼 거리에서도 영향을 받는 것으로 판단된다.

V. 결 론

본 고에서는 상용화된 하드웨어 플랫폼과 운영체제에 대하여 조사하였다. 미국에서는 Crossbow사의 제품이 많이 판매되고 있으며, 일본에서 TRON 프로젝트의 T-엔진 들이 주력으로 자리잡고 있다. 국내에서도 옥타컴에서 ZigBee를 지원하는 Nano-24라는 플랫폼을 생산, 판매하고 있다. 운영체제는 UC Berkeley의 TinyOS가 공개소스로서 자리를 잡아가고 있으며 국내에서는 ETRI의 Nano-Q+가 개발되어 6월말 공개 예정으로 있다.

또한, 상용화된 플랫폼을 이용하여 ZigBee 통신 성능을 실험하고 분석하였다. 먼저 ZigBee의 최대 전송률을 실험하였으며 무선 LAN과의 주파수 충돌시의 전송률을 실험하였다. 실험 결과, 칩수준에서의 최대 전송률은 250Kbps라고 하나 실제 성능은 이보다 낮은 150Kbps를 기록하였다. 무선LAN과의 충돌 시에는 10~20% 정도의 성공률만을 보이며 실내에서 무선 LAN의 영향을 더욱 많이 받는다는 결과를 얻었다. 향후, 무선 LAN과의 충돌을 회피할 수 있는 기법이 개발되어야 할 것으로 판단한다.

참고 문헌

[1] http://www/zigbee.org/, ZigBeeTM Alliance

[2] 송점기, 배한석, 정홍종, 김동균, "이동 애드 혹 네트워크의 계층별 연구 동향," 정보과학회지, 제23권, 제4호, pp.11-20

[3] http://www.uwbforum.org/

[4] http://robotics.eecs.berkeley.edu/~pister/SmartDust

[5] http://webs.cs.berkeley.edu/tos/, The TinyOS and Mote Project

[6] http://www.xbow.com/

[7] http://www.tron.org/, The TRON Project

[8] http://www.smart-its.org/, The Smart-Its Project

[9] 김대영, T. S. Lopez, 유성은, 성종우, 도윤미, "유비쿼터스 센서 네트워크를 위한 ANTS 플랫폼," Telecommuncations Review, 제15권, 2호, 2005년 4월

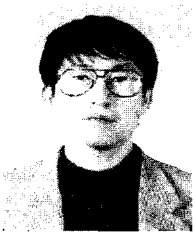
[10] H. Abrach, et-al, "MANTIS: System Support for MultimodAl NeTworks of In-situ Sensors," WSNA'03, Sep. 19, 2003, California, USA

[11] http;//www.octacomm.net/product/nano-24/, Nano-24

저자소개

소 선 섭

1988년 5월 - 1995년 2월 국방과학연구소 연구원
1995년 3월 - 현재 공주대학교 영상보건대학 컴퓨터 멀티미디어 전공
주관심 분야 소프트웨어 테스팅, 임베디드 시스템, 실험적 소프트웨어공학



은 성 배

1987년 3월 - 1990년 2月 : 전자통신연구원 TDX개발단 연구원
1995년 3월 - 현재 한남대학교 정보통신공학과 교수
2000년 10월 - 현재 (주)옥타컴 연구소장 겸임
주관심 분야 실시간 시스템, 센서네트워크