특집

무선 센서 네트워크를 위한 임베디드 소프트웨어 기술

김대영, 양진영, 이인선, 유성은, 성종우, Tomás Sánchez López (한국정보통신대학교), 도윤미 (한국전자통신연구원)

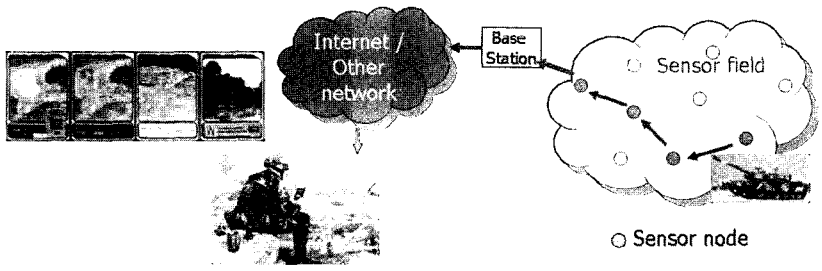
I. 서 론

최근 유무선 통신 기술의 발전 및 모바일 정보기기의 보편화로 새롭고 다양한 서비스를 제공할 수 있는 유비쿼터스 컴퓨팅에 대한 관심이 고조되고 있다. 유비쿼터스 컴퓨팅을 실현시키는 핵심기술 중의 하나로서, 최근에는 외부 환경의 감지와 제어 기능을 수행하도록 사물에 컴퓨팅, 센싱, 통신 기능을 내장하는 센서 네트워크 기술이 각광받고 있다. 특히, '비즈니스위크지'는 '미래의 기술'이라는 특집에서 '센서 혁명(The Sensor Revolution)'을 비즈니스 관점에서 주목해야할 네 가지의 기술 중 하나로, MIT의 '테크놀로지 리뷰'지는 'Brain-Wireless Sensor Networks'를 '세상을 바꿀 10가지 떠오르는 기술' 중 하나로 선택하였다는 점을 주목할만하다.[1,2]

미국에서는 미과학재단(NSF : National Science Foundation)과 DARPA가 중심이 되어 버클리와 UCLA를 포함한 여러 대학과 Intel, Motorola, Crossbow 등 다양한 기업에서 센서네트워크 연구가 활발히 진행되고 있다. 한편, 유럽에서는 센서 네트워크를 "사용자에 친근한 정보사회 창출"을 위한 IST (Information Society Technologies) 분야로 인식하여 지식 기반 사회의 건설을 위한 요소 기술로 개발하고 있다. 또한 일본에서는 세계적 기술 우위를 선점하고 있는 센서, 배터리 분야에 대한 기술을 바탕으로 일본 총무성의 주도로 다양한 응용 서비스를 정의하고 개발에 힘쓰고 있다.

국내에서는 정보통신부의 IT 839전략에서 RFID/USN(Ubiquitous Sensor Network)을 3대 IT 인프라 중 하나로 지정하고, RFID/USN 협회의 설립과 함께 연구 개발 및 확산을 추진하고 있다. 또한 산업자원부의 RFID기반 물류 및 한국형 u-SCM, 과학기술부의 뉴프런터어 과제 등 국가의 주도적인 정책하에 센서 네트워크에 대한 연구 개발이 산학연 협력으로 활발히 진행되고 있다.

센서 네트워크 응용은 지금까지 주로 군사용이나 과학, 공공 분야에 제한되어 왔으나, 최근들어 타이어 압력 센서(TPMS:Tire Pressure Monitoring System)와 유압 센서 등을 무선 센서 네트워크로 연결하여 자동차의 감시와 제어에 사용하는 산업 응용이나, 제품의 재료 입고로부터 생산 및 상품의 판매, 재고 관리 등 선진 물류 관리 시스템에 센서 네트워크의 적용으로 높은



〈그림 1〉 무인정찰 무선 센서 네트워크

투자 대비 생산비 절감을 기대하고 있다.

센서 네트워크에서는 저전력/저가격의 무선 통신, 초소형 마이크로 프로세서 및 운영체제, 자동 구성이 가능한 ad-hoc 네트워킹, 상황인지(Context-aware) 미들웨어, MEMS, 고성능의 다양한 센서, 배터리 등 여러관련 기술의 개별적인 발전을 요구하고 있다. 또한, 이들 기술에 대한 표준화 노력뿐만 아니라, 초소형 센서 시스템의 네트워킹으로 형성되는 센서 네트워크의 특성으로 인하여 임베디드 시스템 기술이 센서 네트워크의 발전을 가늠하는 기술로 평가된다. 특히, 센서 네트워크를 이루는 기술들 중 운영체제, 개발 환경, 네트워크 프로토콜, 시큐리티, 위치인식 프레임웍, 클럭동기 프레임웍, 미들웨어, 가상머신 등이 센서 네트워크를 위한 임베디드 소프트웨어 기술이라고 정의할 수 있다. 본 논문에서는 센서 네트워크를 위한 임베디드 소프트웨어 핵심 기술을 정의하고, 현재까지의 관련 기술의 연구 성과들에 대하여 정리한다.

이어지는 II 장에서는 무선 센서 네트워크 개요에 대하여 소개하며, III 장에서는 센서 네트워크를 위한 임베디드 소프트웨어의 기술을 정의하고, 기술 개발을 위한 요구사항 및 설계 시의 고려사항, 현재 기술의 동향에 대하여 기술하며 마지막으로 결론을 맺는다.

II. 무선 센서 네트워크 개요

무선 센서 네트워크는 넓은 영역에 설치되는 네트워크 인프라에서 다양한 센서 디바이스가 감지한 센서 데이터를 결합하여 응용 서비스에 전달하는 기술이라고 할 수 있다. 이처럼 물리 환경 혹은 상황 정보를 감지하기 위한 센서들과 무선통신, 네트워킹, 센서 데이터 처리 기능들의 결합이 무선 센서 네트워크 기술이라 정의하면 단순하게 기존의 무선 네트워킹 기술과 센서기술을 결합하는 특별히 새로울 것이 없는 것처럼 보일 수도 있다. 그러나 무선 센서 네트워크는 단순히 몇 개의 센서를 네트워킹하는 것이 아니고 배터리 전원에 의해 동작하는 수백 혹은 수천 개의 센서 노드들이 네트워크를 형성하여 때로는 가혹한 환경에서 가능한 오랜 기간동안 보다 정확하고 믿을 수 있는 정보를 수집하는 기능과 무선 통신과 네트워킹 기능이 있는 센서가 임의의 배치에도 불구하고 자율적으로 네트워크를 형성하는 것을 기본으로 하고 있으므로 지금까지와는 다른 고려사항을 가진 새로운 연구 분야 중 하나로 떠오르고 있다.

<그림 1>의 무인정찰용 무선 센서 네트워크는 탱크의 움직임을 감지하는 센서, 컴퓨팅, 무선통신, 그리고 네트워킹 기능을 탑재한 센서노드(Sensor node)와 이들 센서노드들을 관리하면서 유무선 네트워크 인프라를 통해 예제의 정찰병과 같은 사용자에게 수집된 정보를 제공하는 기능을 수행하는 베이스 스테이션 (Base Station)으로 구성된다. 센서 노드에 탑재되는 센서 또는 컴퓨팅 능력에 따라 무선 센서 네트워크의 응용이 광범위해 지는데, 탱크의 움직임을 포착하는 간단한 기능뿐만 아니라, 탱크의 움직임 영상을 전송하는 기능, 영상 프로세싱에 의하여 피아를 구별할 수 있는 복잡한 기능도 수행할 수 있다. 또한, 센서 노드들 간의 협력을 통해서, 탱크의 이동 방향과 속도와 같은 고급 정보까지도 추출해 낼 수 있으며, 이동성을 가진 센서 노드를 사용하여 탱크를 추적하는 경우에는 이동성에 따라 네트워킹의 기술도 다르게 요구된다. 무선 센서 네트워크는 예제의 군응용 뿐만 아니라, 홈 오토메이션, 물류, 지능형 교통 시스템, 헬스 케어, 공정 자동화, 환경 감시, 지능형 로봇, 자동차, 그리고 공장 자동화와 같은 산업 전반 응용에 활용될 수 있다.

다음 장에서 무선 센서 네트워크의 임베디드 소프트웨어 기술을 논하기 전에, 임베디드 소프트웨어가 탑재되는 센서 노드 하드웨어에 대해서 간단히 언급하도록 한다. 1999년 미국 버클리 대에서 WeC라는 첫 번째 Mote 플랫폼을 개발한 후, 매년 rene, dot, mica, mica2, mica2 dot, micaZ와 같은 센서 노드 하드웨어가 버클리대에 의해서 공개되었다. 대부분의 센서노드 하드웨어는 저전력 단거리 무선 통신을 지향하는 8비트 마이크로 컨트롤러와 저속 RF 칩을 기반으로 구성되어 있고, 제한적인 메모리 (Mote 시리즈의 경우, ATMEGA128L을 사용하며 128KB의 플래쉬 메모리와 4KB의 RAM을 가짐)와 배터리 용량으로 인하여 복잡한 기능구현에 제약이 따르므로 자원의 활용과 기능 구현 간에 적절한 균형이 요구된다.

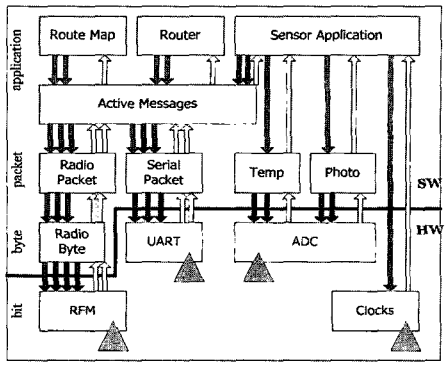
III. 무선 센서 네트워크 임베디드 소프트웨어 기술

1. 센서 네트워크 운영체제 및 개발환경

센서 네트워크 운영체제는 센서네트워크의 응용 개발을 지원하며, 제한된 센서 노드 자원을 관리하는 기반 소프트웨어이다. 자원 제약 사항과 센서 네트워크의 응용을 고려하면, 센서 네트워크 운영체제를 위한 설계 요구사항은 다음과 같다.

우선 센서 네트워크의 운영체제는 작은 크기여야 하는데, 특히 RAM의 사용을 많이 줄여야 한다. 이 때문에 센서 운영체제의 태스크들은 스택을 공유하거나, 심지어는 TinyOS 경우처럼 컨텍스트 오버헤드를 줄이기 위해 태스크간 선점(preemption)을 허용하지 않는 경우도 있다. 센서 운영체제는 전력 소모를 관리하기 위해서 전력 관리자를 두어 마이크로컨트롤러와 라디오 칩의 다양한 슬립모드와 세부 기능의 전력 공급을 제어하고, 그리고 센서 자체의 전력 공급을 제어함으로써 저전력 운영을 달성하게 된다. 센서 노드는 일반적으로 제한된 병렬 처리 능력과 하드웨어 제어 구조를 가지며, 센서와 같은 하드웨어 I/O 장치를 지능형 제어기가 아닌 원시적인 직접 접근 방식에 의해 제어를 수행하므로, 이러한 I/O 제약 사항이 운영체제 설계에 역시 고려되어야 한다. 또한 다양한 응용 분야를 가지

는 센서 네트워크에서는 범용 하드웨어와 소프트웨어가 존재하는 것이 아니라 응용분야에 따라 크게 달라질 수 있으므로 운영체제는 가능한 어떠한 응용에서도 효과적으로 사용할 수 있도록 유연성과 모듈성을 갖추고 있어야 한다. 또한 센서 노드들은 한번 배치가 되고 나면, 유지 보수가 어렵고, 운용 환경 또한 열악할 수 있으므로, 이들을 고려하여 강인한 구조로 설계되어야 한다. 마지막으로 센서 네트워크의 핵심 프로토콜인 저전력 ad-hoc 라우팅 프로토콜, 위치 인식, 클럭동기 기능을 지원하며 프로그래밍 용이성이 고려되어야 한다.



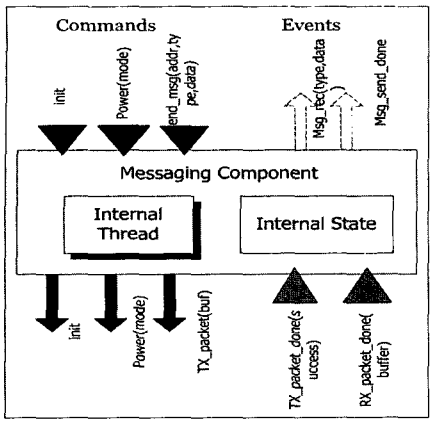
〈그림 2〉 컴포넌트로 구성된 TinyOS 응용예

버클리의 TinyOS 프로젝트는 이 분야에서 가장 진보된 연구 중의 하나이다.[3] TinyOS는 운영체제를 포함하고 있을 뿐만 아니라 다양한 미들웨어와 개발 환경을 가지고 있다. 또한 오픈 커뮤니티 활동을 통하여 표준 플랫폼으로 성장하는 것을 목표로 하고 있다. 그 외 콜로라도 대학의 Mantis나 유럽의 EYES OS, 그리고 ETRI에서 개발중인 나노OS 플랫폼이 있다. 나노 OS는 정부의 RFID/USN과 관련된 다양한 국책 사업에 활용될 수 있도록 다양한 요구사항 하에 개발되고 있으며, 향후 한국형 센서 네트워크 표준 플랫폼으로 자리 잡을 전망이다.

TinyOS는 이벤트 발생에 의한 상태 천이 방식을 채택한 state machine 기반의 프로그래밍 개념을 사용한 운영체제로써, 제한된 메모리 공간의 효율적인 이용과, 프로세싱의 동시성 등을 지원해 주는 운영체제이다. TinyOS에서는 시스템 자원의 제약들 때문에 기존의 IP 프로토콜, 소켓, 쓰레드 개념들을 사용하지 않는다. 이러한 TinyOS의 특징을 크게 아래 3가지로 구분해 볼 수 있다.

첫째, 재사용 가능한 소프트웨어 컴포넌트 기반의 운영체제이며, 응용 프로그램은 하드웨어 컴포넌트의 입/출력을 연결하듯 소프트웨어 컴포넌트의 입/출력 인터페이스를 연결함으로써 작성된다. 둘째, 상태 머신(state machine) 기반의 구조를 가지는 운영체제로, 각각의 상태는 TinyOS의 컴포넌트가 해당된다. 각 컴포넌트의 명령(command)과 이벤트 처리기는 한 상태에서 다른 상태로 빠르게 전이를 수행하며, 기본적으로는 하드웨어의 신호 처리와 같은 특성을 가지므로 적은양의 부가처리와 논블럭킹의 특성을 지닌다. 셋째, 센서노드의 중요한 요구사항의 하나인 저전력 소모를 구현하기 위해 사용되지 않는 CPU의 사이클 동안은 휴지 상태로 들어가 전력소모를 줄인다.

내부적으로 TinyOS 응용 프로그램은 커맨드와 이벤트를 이중 우선 순위에 의해 스케쥴하는 경량 스케쥴러와 함께 그림 2와 같이 컴포넌트의 상태 천이 그래프로 구성이 된다. 예의 응용은 센서 노드가 온도와 조도를 센싱하여 무선 인터페이스를 통해 베이스노드로 전송하는 단순한 경우이며, 실제 3450바이트의 코드와 226바이트 데이터만을 사용하는 초소형 프로그램이



〈그림 3〉 메시지 컴포넌트 예

다. 여기서 Active Messages 컴포넌트는 센서노드와 베이스 노드와의 통신을 담당한다. 그림3에서와 같이 각 컴포넌트의 인터페이스는 입력 커맨드과 출력 커맨드, 입력 이벤트와 출력 이벤트 4가지로 구성된다. 입력 커맨드는 자신의 컴포넌트에서 커맨드 핸들러를 통해 서비스하는 함수로 상위 컴포넌트가 해당 컴포넌트의 기능이 필요할 때 호출하는 것이며, 출력 커맨드는 하위 컴포넌트의 서비스를 호출할 때 사용한다. 입력 이벤트는 해당 컴포넌트의 이벤트 핸들러에 의해 처리되며 하위의 컴포넌트가 해당 컴포넌트에 신호를 넘겨주기 위하여 사용되는 것이며, 출력 이벤트는 해당 컴포넌트가 상위의 컴포넌트에 신호를 전달하기 위해 사용된다.

TinyOS는 코어 운영체제 이외에도 센서 네트워크용 소형 데이터 베이스 엔진인 TinyDB와 태스크 툴킷, PC상에서 시뮬레이션 할 수 있는 TOSSIM, 센서 노드로의 바이트 코드 프로그램의 자동적인 포워딩과 업로드를 가능하게 하고 TinyOS 프로그래밍을 쉽고 간결하게 해주는 가상 머신의 일종인 BOMBILLA, 시큐리티 기능을 제공하는 TinySec을 포함한 많은 다른 유틸리티들이 제공된다.

2. 센서 네트워크 스택

센서 네트워크에서 프로토콜 기술은 저가격의 저전력 무선 기술, 저전력 라우팅 및 센서 네트워크 특성에 적합한 데이터 기반 라우팅 기술, 그리고 타 센서 네트워크 및 상위 네트워크과의 상호 운용성 등을 필요로 한다. 이들 프로토콜을 구현한 물리계층에서부터 네트워크 계층까지의 센서 네트워크 스택은 기본적으로 관리와 제어를 위해 소프트웨어 기능이 요구되며, 특히 MAC의 상당 부분과 네트워크 계층은 소프트웨어로 구현되기 때문에 임베디드 소프트웨어의 범주에 포함된다.

가) 물리 계층

물리 계층은 통신 채널 상으로 데이터 비트를 송수신하는 것을 담당하는 계층으로, 무선통신 환경에서 이 계층은 주파수 선택, 반송파 주파수 생성, 신호 검출, 변복조, 채널 코딩 등을 수행한다. 센서네트워크 응용에 적합한 주파수로는 900MHz, 2.4GHz의 ISM대역의 주파수들이 많이 사용되고 있다. 기존의 물리 계층을 설계할 때 주된 고려사항은 데이터 전송 속도 및 처리율(throughput)에 있었다. 하지만, 센서 네트워크에서는 많은 양의 데이터를 주고받지 않기 때문에 기존의 고려사항은 부차적일 수밖에 없다. 보통, 센서노드들은 그 특성상 배터리로 동작하며, 배터리를 충전하거나 교환할 수 없는 경우가 많기 때문에 배터리의 수명이 센서노드의 수명과 직결된다. 따라서, 무선 센서 네트워크에 적합한 물리 계층을 설계할 때 가장 중요하게 고려해야

할 사항은 전력 소모를 최소화하는 것이다.

나) 데이터 링크 계층

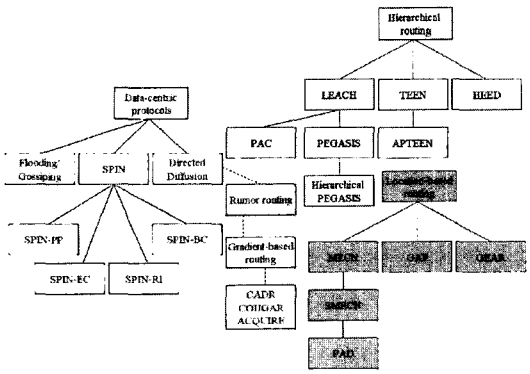
데이터 링크 계층은 물리 계층의 비트들과 데이터 링크 계층의 프레임 사이의 상호 변환, 전송 에러 처리, 매체 접속 제어, 그리고 흐름제어 등을 담당하며, 점대점 또는 점대다수의 네트워크를 구성하는 데 있어 신뢰성을 보장해야 한다.

센서 네트워크의 대표적인 MAC 프로토콜인 S-MAC을 소개하면, 크게 periodic listen and sleep, collision and overhearing avoidance, message passing의 세 가지 기법을 제공한다.[4] 대부분의 센서 네트워크 응용들은 특별한 센싱 이벤트가 발생하지 않을 때는 휴지(idle) 상태가 된다. 그리고 이벤트 보고 주기도 길기 때문에 센서 노드의 라디오 수신기를 항상 켜놓는 것은 전력을 낭비하게 되는 요인이 된다. 그래서 S-MAC을 사용하는 센서 노드들은 동기화되어 listen 과 sleep 모드를 반복하여, 전력 소모를 줄이는 방법을 사용한다. Collision을 피하기 위해서 S-MAC은 802.11과 매우 유사한 방법을 사용하며, carrier sense 후 노드간의 RTS(Request To Send)/CTS(Clear To Send)를 교환하여 hidden node 문제를 해결하며, 최종 송신 노드와 수신 노드를 결정한다. 또한 overhearing 문제를 해결하기 위해서, 다른 노드들 사이의 RTS/CTS 패킷을 인식한 주변노드들은 그 노드들의 데이터 패킷 전송 시간동안 sleep 모드로 바뀌어 데이터를 수신하지 않는다. 송신하고자 하는 메시지의 길이가 길 경우에는, fragmentation에 의해 생기는 RTS/CTS 제어 패킷의 수가 증가함에 따라 에너지 소모가 증가하게 된다. 이를 해결하기 위해서 S-MAC에서는 긴 메시지를 짧은 여러 개의 패킷으로 fragmentation 하고, 한 개의 RTS/CTS 만으로 채널을 예약한 후 패킷들을 보내게 된다. 이로써 제어 패킷에 의한 에너지 소모를 줄일 수 있다. 다만 다른 노드들이 전체 메시지의 전송이 끝날 때까지 기다려야 하기 때문에 공정성(fairness)에서 문제점이 있을 수는 있다.

다) 네트워크 계층

센서 네트워크에서의 네트워크 계층은 센서 노드들간 통신을 위한 애드 혹(ad-hoc) 라우팅 기능과 외부의 타 센서 네트워크나 기존의 인터넷과의 통신을 위한 상호 운용성을 제공하여야 한다. 또한 수천 내지 수만 개의 센서노드가 분산되어 있는 센서 네트워크에서 하나의 노드에 생기는 문제가 전체 네트워크의 기능에 영향을 미치지 않도록 고장 감내성(fault tolerance)을 고려해야 한다. 그밖에도 센서 네트워크라는 새로운 분야는 네트워크 계층의 설계에 몇 가지 추가적인 사항을 요구한다. 에너지 효율성(energy efficiency), 데이터-중심(data-centric), 데이터 통합수집(data aggregation), 속성 기반의 주소(attribute-based addressing), 위치 인식(location awareness)등이 대표적이다.

에너지 효율성은 센서 네트워크와 분리할 수 없는 근본적인 것으로 어떤 한 요소만의 문제가 아니라 센서 네트워크를 이루는 전체 요소의 전력소모를 고려하여 연구되어야 한다. 이는 또한 네트워크 계층에서는 저전력 라우팅과 깊은 관련이 있다. 데이터-중심기법이란 센서노드들이 센서 네트워크 내에서 IP 주소와 같은 어떤 고유한 주소를 가지고 통신을 하기보다는, 싱크(베이스 노드)가 관심사항(interest)을 센서 노드들에게 알리면, 각 센서 노드는 센싱한 데이터가 관심사항과 일치하는지 판단하여, 일치하면 데이터를 싱크에게 전송하는 기법으로, 속성 기반의



〈그림 4〉 센서 네트워크 라우팅 프로토콜 분류

주소화를 필요로 한다. 데이터 통합 수집은 한 노드가 다른 여러 노드의 데이터를 모아 의미있는 하나의 데이터로 만들고, 이를 싱크 노드로 전송함으로써 전송 횟수와 전력소모를 줄여주며 데이터-중심 라우팅에서 생길 수 있는 임플로젼(implosion)과 오버랩(overlap) 문제도 해결할 수 있다. 임플로젼은 동일한 데이터가 여러 경로를 통해 여러 번 수신되는 것이고, 오버랩은 다수의 노드가 동일한 지역을 감시하면서 같은 데이터가 불필요하게 많이 생성되는 문제이다. 속성 기반의 주소 방식에서는 노드가 네트워크 안에서 고유하게 정해진 주소를 가지지 않으므로, 싱크가 관심 있는 데이터(속성으로 지칭됨)를 기술한 쿼리를 센서 노드들에게 전송하여, 이 쿼리와 일치하는 데이터를 가지는 특정 노드들만이 데이터를 전송한다. 마지막으로 센서 노드들의 위치를 활용하여, 최소 전력을 소모하는 라우팅 경로를 찾는 기법들이 많이 활용되고 있다.

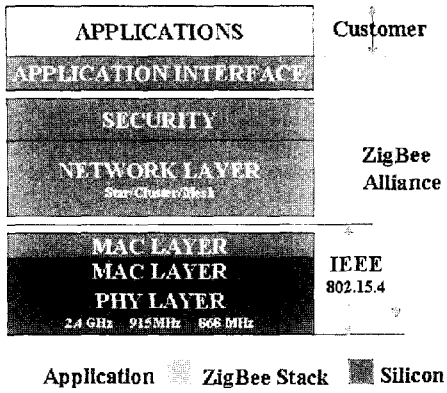
지금까지 많은 연구들이 센서 네트워크 라우팅 프로토콜과 관련하여 진행되어 왔다. 이러한 연구의 결과로 개발된 라우팅 프로토콜들은 그 구조적 특성과 이용 정보에 따라 크게 데이터-중심 라우팅 프로토콜, 계층적 라우팅 프로토콜, 위치기반 라우팅 프로토콜로 분류될 수 있다. 라우팅 프로토콜은 이외에도 급격한 토폴로지의 변화, 고도의 확장성, QoS 등에 대한 연구들이 네트워크 계층에서 활발히 연구되고 있다. 그림 4에 대표적인 센서 네트워크 라우팅 프로토콜들을 분류하였다.[5]

라) 표준화 동향

센서 네트워크를 위한 표준 프토토콜로서 기대를 받고 있는 저속개인무선네트워크(LR-WPAN)를 위한 표준화 단체로서는, IEEE 802.15의 TG4가 있는데, IEEE 802.15.4는 물리 계층에서 MAC 부계층까지를 정의하며, 이 표준의 한 응용으로 무선 센서 네트워크가 포함되어 있다. 그리고 ZigBee Alliance는 신뢰성 및 기기 간의 상호 연동을 보장하며, 저비용으로 저전력을 소모하는 무선통신의 구체적인 활용 및 응용을 목적으로 산업체간에 결성된 비영리 협력 모임이다. 특히, 유수의 반도체 제조업체 및 기술 파급 역할이 큰 기관들이 참여하여 현재 ZigBee Alliance는 급속한 성장을 하고 있으며, 그림 5와 같이 IEEE 802.15.4를 기반으로 네트워크 계층을 포함, 보안 및 응용 프로파일까지를 정의한다. ZigBee의 물리 계층과 MAC 부계층인 IEEE 802.15.4는 이미 2003년도 10월에 표준화되었으나, 그 외의 네트워크 계층을 포함한 ZigBee 표준은 아직 표준화 작업 중에 있다.

대표적인 802.15.4 RF 칩인 Chipcon사의 CC2420은 하드웨어로 구현된 물리계층 및 MAC 부계층의 일부 기능을 제외한 모든 기능을 마이크로컨트롤러에서 실행하는 임베디드 소프트웨어로 구현하고 있다.

ZigBee의 네트워크 계층은 현재 표준화 진행



〈그림 5〉 ZigBee 프로토콜 스택

중이며 스타와 메시 방식의 토폴로지를 지원한다. 모든 ZigBee 디바이스들이 라우팅 기능을 가지고 있을 필요는 없는데, 라우팅 기능을 가지고 있는 디바이스를 ZigBee 라우터라 부른다. 네트워크 계층은 네트워크에 참여하고 떠나는 데 사용되는 절차를 기술하며, 프레임들에 보안을 적용하는 방법, 프레임들을 원래 목적지로 라우팅하는 방법 또한 정의하고 있다. ZigBee 조정자(coordinator)는 적당한 때에 새로운 네트워크를 시작하는 기능을 가지며 새로이 가입된 디바이스에게 주소를 할당하는 역할을 담당한다.

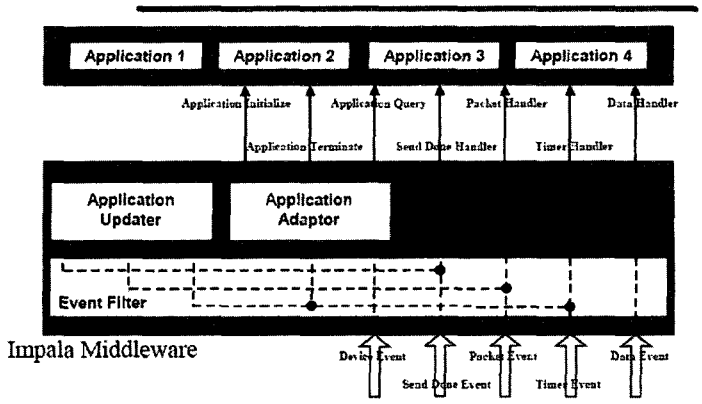
지금까지 살펴보았듯이, IEEE 802.15.4 및 ZigBee는 무선 센서 네트워크를 한 응용으로 하는 첫 표준화 활동이라는 점에서 의미가 있다. 하지만, 현재 IEEE 802.15.4를 개선하기 위한 IEEE 802.15.4.a, IEEE 802.15.4b와 같이 새로운 표준화 작업이 진행되고 있고, ZigBee 네트워크 계층에 대하여는 센서 네트워크에 적합한 표준화를 위해서 여전히 연구할 사항이 남아 있다.

3. 센서 네트워크 미들웨어

가) 센서 네트워크 미들웨어 요구사항 및 기능

센서 네트워크 미들웨어는 일반적인 미들웨어와 마찬가지로 다양한 센서 노드로 구성된 하드웨어 계층과 운영체제 상에서 존재하며, 응용 소프트웨어에 추상화된 인터페이스를 제공하는 역할을 한다. 하지만 기존에 연구된 많은 미들웨어들을 센서 네트워크에 그대로 적용시킬 수는 없다. 이는 센서 네트워크는 일반적으로 데스크탑 환경이나 모바일, 무선 환경과는 또 다른 특징을 가지고 있기 때문이며, 센서 네트워크 미들웨어는 이러한 센서 네트워크의 특징을 만족시킬 수 있도록 설계되어야 한다. 센서 네트워크는 제한된 컴퓨팅 능력과 에너지 그리고 낮은 대역폭뿐만 아니라, 설치되는 환경의 영향을 많이 받으며, 노드 위치의 변화, 센서 네트워크 일부의 유실 등 센서 네트워크의 전체 혹은 부분이 동적인 변화를 겪기 쉽다. 심지어는 전력의 완전 소모나 환경의 영향으로 노드가 동작하지 못하더라도 해당 노드를 수리할 수 없는 경우가 많기 때문에 센서 네트워크 미들웨어는 이러한 환경을 최대한 고려해서 설계되어야 한다. 다음은 센서 네트워크 미들웨어 설계시에 고려해야 할 요구사항이다. [6]

중앙 집중적인 알고리즘을 기반으로 하는 네트워크는 센싱된 정보를 중앙 노드로 집중시키고 중앙 노드에서 복잡하고 지능적인 태스크를 수행하도록 한다. 따라서 중앙 집중 알고리즘은 단일 중앙 노드의 결함등에 대처할 수 없고 효율적인 에너지 사용이 어려우며 대형 네트워크 구조에 적합하지 않다. 이와는 반대로 센서 네트워크를 위한 구조는 각각의 센서 노드들이 근접한 센서 노드들과 협력을 하는 분산된 알고리즘을



〈그림 6〉 Impala 미들웨어 구조

바탕으로 한다. 이로써 센서 네트워크 규모가 커짐에 따라 증가하는 네트워크 파티션(network partition)과 특정 노드의 에러(failure)에도 효과적으로 대처할 수 있게 한다.

센서 네트워크를 통해서 더 많은 정보를 획득하고 이에 따른 태스크의 질을 높이기 위해서는 일반적으로 더 많은 자원을 사용해야만 한다. 적응형 알고리즘은 결과의 질과 이를 획득하기 위한 투입 자원의 수준 사이에서 적절한 트레이드오프(trade-off)를 제공하며, 제한된 자원을 효과적으로 사용하기 위한 필수적인 접근 방법이다.

센서 네트워크 미들웨어는 응용에 따라 다양한 기능들을 수행하는데, 현재 많이 연구되고 있는 미들웨어 기능으로는 전력관리, 위치인식 프레임웍, 소프트웨어 배포와 업데이트, 센서 데이터베이스, 데이터의 적절한 분배와 복제와 관련된 기능들이 있다. 그 외에도 센서 네트워크의 상황 인식 미들웨어에 대한 연구도 활발히 이루어지고 있다. 본 논문에서는 지면 관계상 이들에 대한 설명은 생략한다. 다음 절에 대표적인 미들웨어 프로젝트의 예를 들었다.

나) 대표적인 센서 네트워크 미들웨어 연구

대표적인 센서 네트워크 미들웨어로서 Cornell 대학의 Cougar, Delaware 대학의 SINA, Rochester 대학의 MiLAN, Virginia 대학의 DSWare, UCLA의 SensorWare, 프린스턴 대학의 Impala, UCB의 Bombilla 와 UCLA의 Middleware Techniques in PADS, Virginia의 SAMANTA, SCADDS등이 있다. 그중 대표적인 몇 가지 연구 내용은 다음과 같다.

Impala (Princeton)

얼룩말과 같은 야생 동물들의 이동과 번식 연구에 센서 네트워크를 활용하기 위한 ZebraNet 프로젝트의 일환으로 시작되었다. Impala는 응용의 모듈화(modularity), 적응력(adaptivity), 복구력(repairability)에 연구 초점을 맞추고 있으며 그림 6과 같은 계층적 시스템 구조를 가지고 있다. 응용 프로토콜과 프로그램들은 상위 계층에 위치하며 그 아래로 시스템 자원의 현재 상태에 최적인 행동을 선택하는데 사용되는 응용 어댑터, 소프트웨어 버전 업그레이드등에 사용되는

응용 업데이터 등의 미들웨어 에이전트가 상위 계층을 지원한다.

Bombilla (Berkeley)

Bombilla는 TinyOS 상에 구현된 작은 가상 머신 (virtual machine)으로 제한된 자원을 가진 노드들로 구성된 센서 네트워크에서 프로그램을 동적이면서 효율적으로 수행하는 것을 도와준다. Bombilla는 최대 24개의 바이트 명령으로 구성되는 캡슐이 각각의 노드에 올려져서 태스크를 수행할 수 있는 구조로 되어 있으며 캡슐은 스스로 이동할 수 있다. Bombilla는 이 외에 이웃하는 노드들로부터 소프트웨어를 자동으로 갱신할 수 있는 기능과 악의적인 코드의 실행을 방지하는 보안 기능을 포함하고 있다. 하지만 Bombilla는 극도로 자원이 제한된 상황에서의 사용을 가정하기 때문에 프로그램을 더 쉽고 효율적으로 만들기보다는 함축적인 명령어 세트를 사용하는 방법을 채택했으며 따라서 프로그램의 구현이 어렵다는 단점을 가진다.

SensorWare (UCLA)

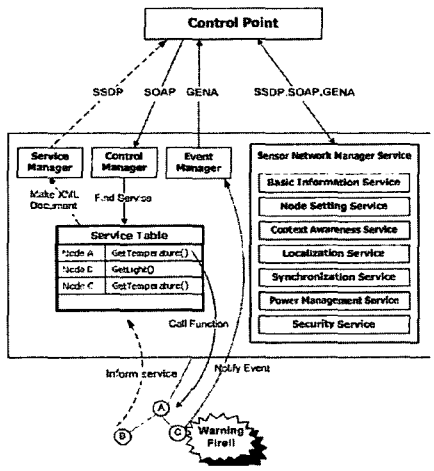
SensorWare는 분산된 센서 노드의 제어를 위해서 경량의 실행프로그램인 모바일 스크립트를 사용한다. 스크립트는 다른 노드로 복사되거나 이동할 수 있으며, 내장된 수행 알고리즘에 따라 자율적으로 동작하게 된다. 따라서 사용자 또는 베이스 노드가 센서 네트워크의 제어를 위해 일일이 모든 센서와의 통신을 통해 정보를 처리해야 하는 부담을 줄일 수 있다. SensorWare는 또한 개발자로부터 자원 관리와 같은 하드웨어와 관련된 부분을 숨겨주고, 여러 응용 프로그램 사이에서 센서 노드끼리의 자원을 공유하는 방식을 제공해줄 뿐만 아니라, 고수준의 스크립트 언어를 사용하기 때문에 센서 네트워크 환경에서의 편리한 개발환경을 제공해준다. 하지만 SensorWare는 180Kbyte라는 초소형 임베디드 환경에서는 적지 않은 크기를 가지고 있기 때문에, mote처럼 제한된 메모리를 가진 센서 노드에는 적합하지 않다.

다) 센서 네트워크 미들웨어와 기존 네트워크과의 연동

센서 네트워크를 기존 네트워크 환경에 통합하기 위한 연구도 진행되고 있는데, 예를 들어 센서 네트워크 서비스를 UPnP(Universal Plug and Play)를 통해 제공하는 것이 그 예이다. BOSS(Bridge Of the Sensors) 구조의 경우, 제한된 사양의 센서 노드들이 UPnP를 직접 동작시키는 것이 어렵기 때문에 베이스 노드에게 모든 센서 노드들의 UPnP 에이전트 기능을 담당하도록 한다.[7] 그림 7과 같이 BOSS에서는 베이스 노드에서 UPnP의 모든 기능 (addressing, discovery, description, control, eventing presentation)을 이용할 수 있도록 서비스, 컨트롤, 그리고 이벤트 매니저를 구현하였다.

4. 센서 네트워크 시큐리티 및 프라이버시

센서 노드들은 도청에 취약한 무선 환경으로 연결되기 때문에, 인증 된 노드만이 데이터 통신에 참여하며 그 내용이 타인에게 노출되지 않아야 하는 등의 데이터 보안과 프라이버시 정보 보안이 요구된다. 그러나 8비트 급의 소형 CPU, 저용량의 배터리, 적은 메모리 공간, 높은 지연시간 등 센서 노드가 가지는 여러 제약 사항은 보안 기능의 충분한 제공에 심각한 제한을 준다. 그리고 무선 환경의 특성상 RF 잡음과 다중경로



〈그림 7〉 BOSS 구조 및 기능

페이딩으로 인한 패킷 손실, 센서의 이동으로 인한 재경로 설정과 같은 특성들로 인하여 보안 보장이 더욱 어려워진다. 특히 넓은 지역에 설치된 센서 노드 그 하나하나가 공격의 대상이 될 수 있음에도 이들을 관리하기가 어렵기 때문에, 안전한 센서 네트워크를 구성하는 것은 보안 분야에서의 새로운 연구 주제가 될 수 있을 것이다.

안전한 데이터 통신을 위해서 기본적으로 요구되는 기능들은 우선 데이터 인증, 기밀성, 무결성, 신규성(Freshness) 등이 있다. 즉, 데이터는 합법적인 사용자로부터 온 것이어야 하며, 타인에게 노출되거나 타인에 의해서 변조되지 않아야 하고 전달된 데이터는 최근의 것이어야 한다. 센서 네트워크에서도 이러한 기본적인 보안이 유지되어야 한다.[8]

센서 네트워크에서 발생 가능한 보안적 문제점으로는 센서 노드의 포섭, 도청, 사적인 데이터로의 접근, 서비스거부 공격, 기존 센서 네트워크의 악용 등이 있다.

센서 네트워크는 수천 또는 수 만 개의 노드들로 구성될 수 있으므로 노드들을 개별적으로 모니터하고 공격으로부터 방어하는 것은 사실상 불가능하다. 공격자들은 네트워크내의 소수의 노드들에 대한 공격만으로도 센서 데이터를 변조하거나, 감지 된 정보를 빼 낼 수가 있다. 이런 센서 노드 포섭 공격을 방지하기 위해서 강력한 위조 방지 하드웨어를 사용하는 것은 비용문제로 인하여 좋은 해결책이 될 수 없으므로 노드 간 인증을 이용하여 노드를 확인하거나, 노드 폐지를 이용해서 악의적인 노드를 제거하는 등의 소프트웨어 수준의 해결방안을 확보해야 한다.

무선 환경에서 센서 노드들 간의 통신을 도청함으로써 사적인 정보에 접근하는 공격에 대응하기 위하여, 일반적으로 사용할 수 있는 방법은 데이터 암호화이다. 그러나 강력한 암호화를 제공할 수 있는 공개키 기반 암호화 시스템이나 종단간 암호화 기법 등은 저 전력, 저 메모리의 센서 노드에 적합하지 않다. 따라서 바로 인접한 이웃 노드와 비밀키를 공유하는 홉간 암호화 기법이나 다중경로 라우팅과 같은 방식이 적합할 수 있다.

센서 네트워크에서 정상적으로 수집된 사적인 데이터를 공격자가 불법적으로 획득하고, 또한 이들 데이터들 간의 상관관계를 알게 되는 경우에 프라이버시가 보호되지 못한다. 센서 네트워크는 대부분 원거리에서 사적인 데이터를 수집하기 때문에, 특히 데이터의 기밀성이 유지되어야 한다. 이를 위하여 데이터 암호화와 접근 제어 등의 방법을 사용한다거나 네트워크의 데이터 수집에 제한을 두는 방식 또는 감지된 데이터를 익명 방식으로 데이터베이스에 저장함으로써 데이터의 비밀성을 지킬 수 있다. 또는 쿼리 처리를 분산적으로 적용해 하나의 노드가 전체 쿼리에 대한 결과를 알 수 없도록 할 수도 있다.

네트워크의 기능마비를 목적으로 하는 서비스

거부(DoS) 공격은 센서 네트워크에서도 고려해야할 요소이다. 라디오 재밍 공격이나, 불필요한 데이터를 계속 전송함으로써 전력을 소모시키는 공격, 하나의 포섭 된 노드를 이용하여 불필요한 라우팅 정보를 발생시키는 방법 등의 서비스거부 공격이 가능하다. 각각의 공격의 유형에 따라 그 대응 방법도 달라져야 하는데, 중요한 것은 역시, 에너지 효율적인 방법을 사용해야 한다는 것이다.

센서 네트워크는 센서 노드를 이용하여 정보를 수집하는 목적으로 쓰인다는 특성 때문에, 센서 네트워크의 악의적인 사용을 방지하는 것도 고려되어야 하는 사항이다. 외부인이 센서 노드를 이용하여 사생활 정보를 빼내는 것을 원하는 사람은 없을 것이다. 센서 노드의 불법적인 사용을 방지하기 위하여, 불법적인 통신을 방지하고, 적법한 데이터만을 허용할 수 있는 센서 탐지기(detector) 등의 개발이 필요하다.

SPINS 는 센서 네트워크의 인증과 접근 제어를 위한 소프트웨어 솔루션 중의 하나로, U.C. Berkeley에서 제안되었다. SPINS는 SNEP(Sensor Network Encryption Protocol) 과 μTESLA(Micro Timed Efficient Stream Loss-tolerant Authentication)로 구성된다. SPINS에서의 베이스 노드는 신뢰할 수 있으며, 에너지 사용에 제한이 없는 것으로 가정한다.

SNEP은 의미적 보안(Semantic security), 데이터 인증(Data authentication), 재생 방지(Replay protection) 기능 등을 제공한다. SNEP은 제한된 계산/통신 자원을 사용해야하기 때문에, DES-CBC의 대칭 키를 사용한 암호방식을 사용하고, 암호화, 메시지 인증 코드(Message Authentication Code), 해쉬(hash), 무작위 수 발생(random number generation) 등을 단일 블록 암호화기(single block cipher)로 계산한다.

μTESLA는 브로드캐스트 시의 인증, 즉 브로드캐스트하는 데이터의 신뢰성을 제공한다. 데이터 브로드캐스트 시에 대칭 키 시스템을 사용하면, MAC 키를 탈취한 악의적인 노드에 의한 노드 모방(impersonation) 혹은 메시지 위조가 가능하다. 그러나 공개키 시스템은 제한된 자원을 가지는 센서 네트워크에 적합하지 않기 때문에, μTESLA는 대칭키 시스템을 이용한 공개키 에뮬레이션 시스템을 사용한다. 즉, 지연된 키 방출과 단방향 함수 키 체인(one-way function key chain)을 이용하여, 비대칭성을 이루며 브로드캐스트 인증을 가능하게 한다.

이 밖에도 Carnegie Mellon 대학의 Aura, George Mason 대학의 LEAP, U.C.Berkeley 의 TinySec, University of Illinois 의 Mist등의 프로젝트들에서도 센서 네트워크를 위한 임베디드 소프트웨어 기반의 보안 솔루션 개발이 진행되고 있다.

5. 센서 네트워크 위치 인식 및 클럭 동기화 프레임워크

센서노드의 위치인식은 위치기반 라우팅과 같이 에너지 효율적인 센서 네트워크 동작과 위치기반 응용 서비스를 위해 필수적인 기술이다. 또한 센서 네트워크에서 동기 기술은 S-MAC과 같은 동기 기반 통신 프로토콜 개발뿐만 아니라, 암호화 기술에서의 타임 스탬프, 다른 노드들로부터의 같은 이벤트의 중복 감지에 대한 인식등 여러 가지 응용을 위하여 필요한 기술이다. 그리고 이 두 기술은 최소한의 안정된 하드웨어 지원하에 알고리즘과 프로토콜의 대부분이 임베디드 소프트웨어로 구현된다.

위치인식만을 설명하면, 이는 센서 네트워크

내에서 개별 센서노드의 위치를 중앙 집중적이거나 분산적인 방법으로 알아내는 기술을 의미한다. 이는 또한 넓은 지역이나 많은 수의 센서노드에서도 안정적으로 작동하여야 하고, 하드웨어의 제약을 가진 장치에서 저 전력 요구사항을 만족시켜야 한다. 일반적으로 위치 인식은 아래의 3가지 방법에 의해서 구현이 되는데, 일반적으로 센서 네트워크에서는 삼각측량법을 많이 사용한다. [9]

- **삼각측량법** (Triangulation) : 삼각형의 기하학적인 성질을 이용하여 대상의 위치를 계산하는 방법으로, 절대위치를 알고 있는 3개 이상의 기준점(Anchor)으로부터 해당 센서노드와의 거리를 측정하여 위치를 계산하는 Lateration 기법과 기준점에 대한 상대적인 각도를 측정하여 거리를 알아내는 Angulation법이 있다. 실제 거리 측정 방법의 경우, MS의 RADAR에서는 직접 RF 신호의 세기 (Signal strength)를 측정하고, GPS, AT&T Cambridge Lab의 Active Bats 그리고 MIT의 Cricket에서는 전파나 초음파의 이동 시간 (Time-of-Flight)을 측정하며 University of Washington의 SpotON은 신호 감쇄 (Attenuation) 를 측정하여 거리를 추정한다.
- **접근법** (Proximity) : 알고 있는 기준 위치들 중 대상이 어디에 가까이 있는가를 찾아내어 대상의 위치를 알아내는 방법으로 GIT의 Smart Floor, Olivetti의 Active Badges 등에서 사용한다.
- **Scene 해석법** (Scene analysis) : 특정한 위치에서 관측된 장면의 특징을 사용하여 관찰자의 위치를 파악하는 방법이다.

현재 위치인식에 대한 많은 연구가 진행되고 있으나 아직까지 현실적으로 경쟁력있는 알고리즘이나 시스템은 나오지 않고 있다. 이는 위치인식이 수행되는 실내 전파환경의 열악성과 거리 측정방법의 부정확성, 그리고 센서 네트워크의 기본 구조인 멀티홉 기반의 Ad-Hoc 네트워크 환경으로 인해 정밀한 위치를 계산해내기 어렵기 때문이다. 정밀한 위치인식을 위한 소형, 저전력의 하드웨어와 열악한 환경에서도 동작할 수 있는 알고리즘 개발이 기대된다.

Ⅳ. 결 론

무선 센서 네트워크 기술은 다양한 산업 응용 및 유비쿼터스 컴퓨팅 서비스를 실현시키는 핵심 기술로써, 대부분의 세부 기술들이 제한된 자원을 가지는 컴퓨팅 환경에서 동작하면서 저전력 요구 사항을 만족시켜야 하는 임베디드 소프트웨어의 형태로 구현 된다. 그런데, 자원의 제약 사항이 기존의 임베디드 시스템보다 훨씬 심하기 때문에 운영체제부터 네트워크 스택, 미들웨어, 보안에 이르기까지 모든 세부 기술들이 새로이 개발되어져야하는 매우 도전적인 분야이다. 본 논문에서는 이러한 센서 네트워크의 요소 기술들의 정의와 최근 이슈들, 그리고 연구 동향에 대해서 살펴보았다. 센서 네트워크는 RFID/USN IT 인프라의 중심 기술로써, 또한 다양한 응용의 기반 기술로 사용이 되기 때문에, 이러한 세부 임베디드 소프트웨어를 표준화된 플랫폼 형식으로 개발하여, 공개 소프트웨어 형식으로 발전시키며, 이로써 시장성 있는 응용 개발에 도움을 주어야 한다.

참고문헌

[1] Business Week, "Future of Tech, Tech Wave 2: The Sensor Revolution," available at http://www.businessweek.com/magazine/content/03_34/b3846622.htm

[2] Nicholas Negroponte, "10 Emerging Technologies That Will Change the World," available at http://www.globalfuture.com/mit-trends2003.htm

[3] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, "System Architecture Directions for Networked Sensors," ASPLOS IX, Nov. 2000.

[4] W. Ye, J. Heidemann, and D. Estrin. "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," 21st International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2002), Jun. 2002.

[5] K. Akkaya and M. Younis, "A Survey of Routing Protocols in Wireless Sensor Networks, " in the Elsevier Ad Hoc Network Journal (to appear)

[6] K. Römer, O. Kasten, and F. Matter, "Middleware Challenges for Wireless Sensor Networks," ACM Mobile Computing and Communication Review, Vol. 6, No. 4, pp. 59-61, Oct. 2002.

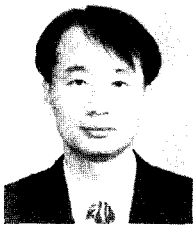
[7] 송형주, 김대영, "UPnP를 이용한 센서 네트워크 관리 시스템", 한국정보과학회 추계학술발표대회, 2004년 10월

[8] H.Chan, A.perrig. "Security and privacy in sensor networks," IEEE Security &Privacy Magazine, pp.103-105, Oct. 2003.

[9] J. Hightower, "Location Systems for Ubiquitous Computing", IEEE Computer, Aug. 2001

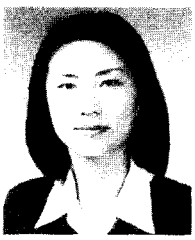
[10] 김대영, "센서네트워크 기술 I,II,III,IV,V,VI," FA Journal, Mar. - Aug. 2004.

저자소개



김 대 영

1990년 부산대학교 전산통계학과 학사
1992년 부산대학교 전산통계학과 석사
2001년 University of Florida 컴퓨터공학 박사
1992년 - 1997년 한국전자통신연구원 연구원
1999년 - 1999년 AlliedSignal Aerospace 연구소 방문연구원
2001년 - 2002년 Arizona State University 컴퓨터공학과 연구 조교수
2002년 - 현재 한국정보통신대학교 조교수
주관심분야 Sensor Networks, Real-Time and Embedded Systems, Ad-Hoc Networks



도 윤 미

1989년 경북대학교 전자공학과 학사
1991년 경북대학교 전자공학과 석사
2003년 University of Florida 컴퓨터공학 박사
1991년 - 1997년 한국전자통신연구원 선임연구원
2001년 - 2003년 Arizona State University 컴퓨터공학과 방문 연구원
2003년 - 2004년 한국정보통신대학교 연구교수
2004년 - 현 재 한국전자통신연구원 텔레매틱스 연구단 RFID/USN연구팀
주관심분야 RFID, Sensor Networks, Power Aware and Low Power Computing, Real-Time and Embedded Systems, Ad-Hoc Networks

저자소개

양 진 영

1988년 연세대학교 전자공학과 학사
1999년 한국과학기술원 전기 및 전자공학과 석사
1988년-2001년 삼성전자 중앙연구소
2004년-현 재 한국정보통신대학교 컴퓨터공학과 박사과정
2001년-현 재 삼성종합기술원 C&N Lab 전문연구원
주관심분야 RFID, Sensor Networks, Ad-Hoc Networks, Multiple Access Schemes, Localization

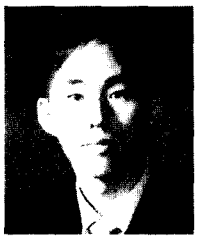
이 인 선

1995년 서강대학교 수학과 학사
1998년 Purdue University 수학과 석사
2004년-현재 한국정보통신대학교 컴퓨터공학과 박사과정
1998년-현재 삼성종합기술원 전문연구원
주관심분야 4G, Wireless Networks, Sensor Network Security & Key Management

유 성 은

2003년 한양대학교 전자전기공학부 학사
1999년-2002년 (주)파인디지털 근무
2003년-현 재 한국정보통신대학교 컴퓨터공학 석사과정
주관심분야 Sensor Network Protocols, WPAN, Real-Time Scheduling

저자소개



성 종 우

2002년 성균관대학교 정보통신공학부 학사
2004년 한국정보통신대학교 컴퓨터공학 석사
2004년-현 재 한국정보통신대학교 연구원
주관심분야 Home Network, Sensor Network Middleware

Tomás Sánchez López

2004년 Polytechnic University of Valencia, 스페인, 컴퓨터공학 학/석사
2004년-현 재 한국정보통신대학교 컴퓨터공학 박사 과정
주관심분야 Sensor Network OS, Evolvable Real-Time Systems