

지능형 홈을 위한 무선 센서 네트워크 구성

Construction of Wireless Sensor Network for Intelligent Home

황세희 · 장인훈 · 심귀보

Se-Hee Whang, In-Hun Jang, and Kwee-Bo Sim

중앙대학교 전자전기공학부

요 약

센서 네트워크는 많은 센서 노드들을 산개시켜놓은 다음 노드들끼리 통신망을 구성하고 주위 환경 정보를 수집하고 분석한다. 각 노드들은 제한된 자원 내에서 지정된 일을 수행해야 하기 때문에 효율성이 극대화되어야 하고 하드웨어 자체의 성능도 뛰어나야 한다. 지금까지 기술적인 제약이 있었지만 최근 IC 제조 기술과 무선 네트워크 기술의 급속한 발달로 인해 저렴하면서 전력소비가 적고 다양한 기능을 하는 소형 임베디드 시스템을 구성할 수 있게 되었다. 기술 발전과 함께 무선 센서 네트워크는 새로운 연구 분야로서 자리매김하였고 현재 여러 연구실에서 연구되어지고 있으며, 제각기 다른 이름으로 다양하게 불리고 있다 - Wireless Integrated Network Sensors (WINS), Mobile Ad hoc NETwork (MANET), Ubiquitous Sensor Network (USN). 현재 센서 네트워크 분야에서 TinyOS가 널리 이용되고 있다. 본 논문에서는 TinyOS와 그 플랫폼을 이용해서 가정 내에서 컨텍스트 인식을 위한 센서 네트워크를 구성하고자 한다.

Abstract

In the sensor network, a lot of sensor nodes are scattered sparsely and organizes a united communication network between each node. After that, environmental information around each sensor node are gathered and analyzed. Because each node operates under resource constraint, the efficiency and hardware specification of a node should be maximized. There exist technical constraints until now but recent technical progress in IC fabrication and wireless network enables to construct a tiny embedded system, which has the properties of low cost, low power consumption, multi functions. Wireless sensor network becomes a modern research field with technical improvements, is studied in numerous laboratories, and is called as diverse different project names - Wireless Integrated Network Sensors (WINS), Mobile Ad hoc NETwork (MANET), Ubiquitous Sensor Network (USN). TinyOS is one of leading project and is widely used. In this paper, we suggest a sensor network, which uses TinyOS platforms and aims for context awareness in a home environment.

Key words : sensor networks, home networks, tinyos, context awareness

1. 서 론

최근 IC 제조기술과 무선 통신 및 센서 기술의 발달로 소형 임베디드 시스템을 구축할 수 있게 되면서 무선 센서 네트워크가 각광을 받고 있다. 기존에는 제한된 자원 내에서 원하는 만큼의 성능이 나타나지 않는 등 기술적 제약으로 인해 센서 네트워크의 실현 여부가 불투명했었다. 하지만 고성능의 마이크로프로세서와 집적도가 높은 임베디드 시스템과 같은 관련 기술이 발전함에 따라 실제 구현이 가능해지면서 이에 따라 많은 연구가 진행되고 있다.

센서 네트워크는 다수의 센서 노드들을 이용해서 통신망을 구성하고 주위의 환경 정보를 수집하고 분석한다. 통신망을 구성해야하기 때문에 안정된 네트워크를 구성해주어야 한다. 소형 임베디드 시스템인 센서 노드는 여러 제약 사항을 가지며 이에 대해서 강인해야하고 안정된 동작을 보장해야

한다. 따라서 센서 네트워크는 다양하고 많은 제약사항이 존재하는 가정환경 내에 적용하기 알맞다고 사료된다.

본 논문에서는 무선 센서 네트워크를 홈 네트워크에 적용하고자 한다. TinyOS라는 센서 네트워크용 운영체제를 이용해서 시스템을 구축하고 하드웨어 플랫폼으로는 MICAZ와 Telos라는 플랫폼을 이용한다. 이러한 시스템을 바탕으로 네트워크를 구성하고 홈 네트워크에 어떻게 적용할 수 있는지에 대해 연구하고자 한다. 2장에서는 우리가 사용하고자 하는 센서 네트워크가 무엇인지 알아보고 3장에서는 홈 네트워크에 대해서 설명한다. 4장에서는 센서 네트워크가 적용된 홈 네트워크 시스템을 어떻게 구성할 것인지 또한 부가적인 서비스를 제공하기 위해서 추가적으로 필요한 것들이 무엇인지에 대해서 알아보고 센서 네트워크의 운용 방식과 이때 가용한 서비스에 대해서 논의한다.

2. 무선 센서 네트워크

센서 네트워크에서는 작은 센서 노드들을 이용해서 주위 환경 정보를 얻어내고 분석한다. 무선 센서 노드들은 이웃한 노드들과 통신망을 구성하면서 서로의 정보를 교환한다. 각

접수일자 : 2005년 10월 17일

완료일자 : 2005년 11월 30일

감사의 글 : 본 연구는 산업자원부의 뇌신경 정보학 연구사업 (2005.4~2006.3)의 연구비 지원으로 수행되었습니다. 연구비 지원에 감사드립니다.

노드들은 소형 임베디드 시스템으로 제한된 자원을 가지기 때문에 많은 제약이 뒤따른다. 따라서 제한된 자원 내에서 효율적인 동작을 보장하고 다양한 분야에서 이용하기 위한 연구들이 진행 중이다. 본 논문에서는 무선 센서 네트워크 분야에서 가장 널리 사용되고 있는 TinyOS를 이용해서 시스템을 구성하고 어떻게 이용할 수 있는지에 대해 논의하도록 한다.

2.1 TinyOS

TinyOS는 오픈 소스 운영체제로 센서 네트워크와 같은 소형 임베디드 시스템을 위해 고안되었다. 현재 다양한 표준 플랫폼을 제공하고 다양한 플랫폼 라이브러리를 제공한다 [1][2].

1) Component-based architecture

TinyOS 응용 프로그램은 그림 1에서 보는 것과 같이 Configuration, Module과 같은 Component와 Component들을 연결해주는 Interface로 구성된다. Interface는 다시 Command와 Event 함수로 나뉜다. TinyOS에서는 네트워크 프로토콜, 센서 드라이버와 데이터 수집 프로그램 등 다양한 컴포넌트 라이브러리를 제공해주기 때문에 개발 기간이 단축될 수 있도록 한다.

2) Event-driven model

TinyOS는 평상시에는 거의 Sleep 모드로 동작하다가 이벤트가 발생했을 때만 정상 모드로 동작한다. 센서를 구동할 때만 동작하는 센서 네트워크 시스템에 TinyOS를 적용하면 저 전력을 보장할 수 있다.

3) Ambiguous boundary between OS and application

TinyOS 및 응용 프로그램들은 Event-Driven 방식을 지원하기 위해 기존의 C를 확장한 nesC로 작성되어 있다. 또한 TinyOS에서는 기본적으로 다양한 컴포넌트 라이브러리를 제공하기 때문에 TinyOS는 운영체제와 응용 프로그램이 혼재되어 있지만 둘 사이의 경계가 애매하다는 특징을 가진다.

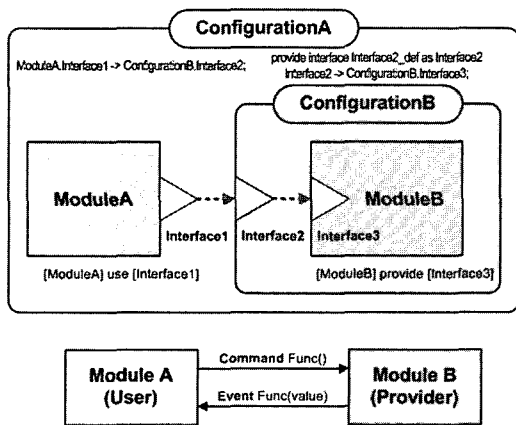


그림 1. TinyOS 프로그램 구성 요소
Fig. 1. Composition of TinyOS Application

2.2 Platforms

TinyOS의 플랫폼은 여러 가지가 있지만 이들 중 대표적

인 것으로는 MICA와 Telos 시리즈가 있다. 그림 2를 참고하면 모델에 따른 정확한 스펙을 알 수 있다. MICAz는 MICA2 기반으로 RF 부분만이 CC2420으로 바뀌었다 [3]. 본 논문에서는 MICAz와 Telos를 기반으로 사용한다. 두 플랫폼 모두 CC2420을 사용해서 상호간 통신이 가능하고 같은 프로그램을 사용할 수 있기 때문이다. 또한 호스트와 인터페이스를 할 때 여러 가지 방법을 이용할 수 있기 때문에 확장성도 보장된다.

Microcontroller	ATmega128	ATmega128	ATmega128	ATmega128
Type	ATmega128	ATmega128	ATmega128	ATmega128
Program memory (KB)	128	128	128	128
RAM (KB)	8	8	8	8
Active Power (mW)	15	15	15	15
Sleep Power (uW)	15	15	15	15
WakeUp Time (ms)	1000	1000	1000	1000
Nonvolatile storage				
Chip	ATmega128	ATmega128	ATmega128	ATmega128
Connection type	UART	UART	UART	UART
Size (KB)	128	128	128	128
Communication				
Radio	CC2420	CC2420	CC2420	CC2420
Data rate (kbps)	10	10	10	10
Modulation type	FSK	FSK	FSK	FSK
Receive Power (mW)	12	12	12	12
Transmit Power at 0dBm (mW)	10	10	10	10
Power Consumption				
Minimum Operation (V)	1.8	1.8	1.8	1.8
Total Active Power (mW)	15	15	15	15
Programming and Sensor Interface				
Expansion	UART, I2C, SPI, ADC, DAC, PWM, RTC, etc.	UART, I2C, SPI, ADC, DAC, PWM, RTC, etc.	UART, I2C, SPI, ADC, DAC, PWM, RTC, etc.	UART, I2C, SPI, ADC, DAC, PWM, RTC, etc.
Communication				
Integrated Sensors				

그림 2. TinyOS 플랫폼 간의 비교
Fig. 2. Comparison of TinyOS Platforms

3. 홈 네트워크

3.1 Home Networks

홈 네트워크란 가정 내의 유무선 네트워크 장비들을 상호 데이터 통신을 통해서 하나로 통합된 유기적인 네트워크를 일컫는다. 홈 네트워크에서 더 나아가 이 통신망을 기반으로 사용자에게 알맞은 서비스를 제공하는 것을 스마트 홈이라고 한다. 지능형 홈 공간 안에서는 사용자에게 맞춤 서비스를 제공하기 위해서 사용자가 원하는 것을 파악하고 가능한 범위 내에서 최대한 욕구를 충족시켜줄 수 있는 서비스를 제공해야 한다.

지능형 홈을 구축하기 위해서는 가정 내의 홈 네트워크 이외에도 외부의 콘텐츠나 솔루션, 그리고 이들을 홈 네트워크와 연결시켜주기 위한 외부 네트워크가 필수적이다. 이외에도 홈 네트워크를 외부 네트워크와 연결시켜주기 위해서는 홈 게이트웨이가 필요하다. 홈 네트워크 내의 장치 연결은 유선이나 무선으로 연결되기 때문에 여러 가지 장치들을 제어하기 위해서는 홈 서버가 필요하다. 지능형 홈 환경 중에서 본 논문은 가정 내의 홈 네트워크 망 구성하는 부분에 초점을 두고, MICAz와 Telos 같은 TinyOS의 플랫폼을 기반으로 가정 내에서의 무선 센서 네트워크 망을 통하여 홈 네트워크의 인프라를 구성한다.

3.2 Context Awareness

Context는 정해진 환경 내에서 사용자의 상태나 주위 환경 정보를 지칭한다. 센서 노드 주위의 물리적인 환경을 가리키는 Physical Context와 센서로부터 직접적으로 데이터를 얻어내지 않고 센서의 위치나 시간과 같은 상황을 가리키는 Situational Context로 구분된다. 다양한 센서를 통해서 Physical Context를 얻을 수 있지만 이들 중 특히 시각정보

와 청각정보에 관한 것을 Visual Context와 Auditory Context라고 구분한다.

인프라가 구축된 유비쿼터스 환경 내에서 다양한 센서들을 통해서 사용자의 상태나 주변 환경을 인식하고 이를 통해 사용자의 요구사항을 추론하는 과정을 Context Awareness라고 한다. 센싱 데이터의 수집과 분석이 어디에서 수행되는가에 따라 Direct Awareness와 Indirect Awareness로 구분된다. Direct Awareness에서는 센서 노드에서 데이터의 수집과 분석이 모두 이루어지고 호스트에서는 센서 노드로부터 컨텍스트 정보를 얻는다. 반면 Indirect Awareness에서는 호스트에서 직접 센서 데이터를 받아서 분석한다. 호스트에서 데이터를 수집하고자 할 때 센서 노드로부터 중간 노드를 몇 단계 거치는가에 따라서 Multi-Hop과 Single-Hop 방식으로 나뉜다. 그리고 중간 노드들의 역할에 따라서 계층 구조로 구분하기도 한다. 또한 컨텍스트 인식 추론 과정에서 정확도를 높이기 위해서 다수의 다양한 센서를 사용하는 것을 Multi-Sensor based Context Awareness라고 한다.

컨텍스트 인식 과정이 끝난 다음 이를 기반으로 사용자의 요구사항에 따라 능동적인 서비스를 제공해주는 것이 Context-based Service이다. 하나 이상의 컨텍스트가 공존하는 환경을 Multi-Context Environment라고 한다. 사용자의 요구에 맞는 서비스를 제공하고자 할 때 다수의 컨텍스트들이 존재하기 때문에 컨텍스트들 간의 우선순위를 정하고 충돌을 관리해주어야 한다. 그림 3은 컨텍스트를 인식하고 컨텍스트에 기반을 둔 서비스를 제공하는 과정의 개념을 도식화하고 있다.

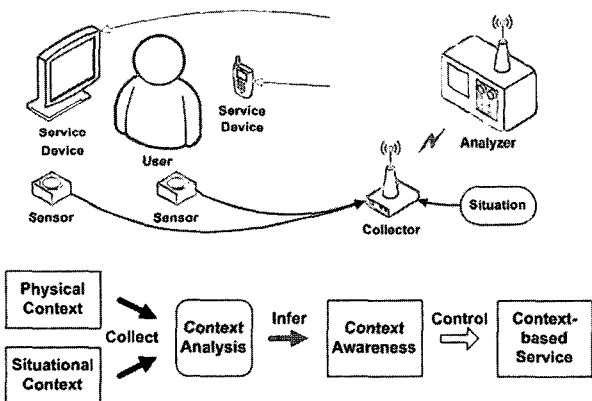


그림 3. Context Awareness 과정
Fig. 3. Process of Context Awareness

4. 홈 네트워크의 구성

4.1 System Configuration

본 논문에서는 TinyOS의 플랫폼인 MICAz와 Telos를 이용한다. 두 가지 센서 모트들은 모두 CC2420을 이용해서 2.4 GHz대역의 RF 통신을 지원한다. TinyOS를 통해서 다른 플랫폼 간에서도 같은 프로그램을 이용할 수 있고 이를 이용해서 정보를 교환할 수도 있다. Telos에는 기본적으로 조도 센서와 광량 센서 그리고 온·습도 센서를 내장하고 있다. 컨텍스트 인식에 있어서 정확도가 높은 추론을 통해서 다양한 서비스를 제공하기 위해서는 많은 종류의 센서가 필요하지만 본 논문에서는 개인의 생체 신호를 통한 생체 정보 인식 및

광량과 온습도에 따른 서비스 제공으로 제한하고 실험을 통해 가능성을 검증해보도록 한다.

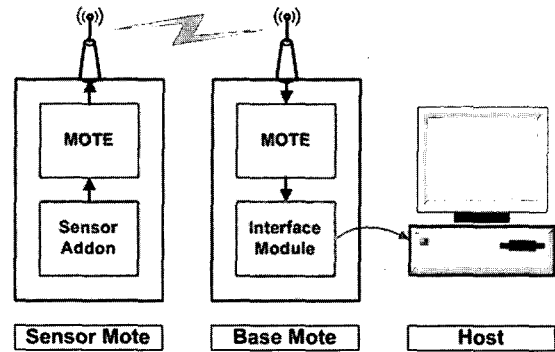


그림 4. 기본 네트워크 구성
Fig. 4. Basic Network Configuration

실험을 위한 기본적인 네트워크를 구성에는 그림 4와 같이 호스트와 베이스 모트, 센서 모트가 필요하다. 호스트로는 PC를 사용하며 베이스 모트로는 MICAz를 이용한다. MICAz는 호스트 PC와 시리얼 인터페이스인 MIB510을 이용해서 연결한다. 센서 모트로는 Telos나 MICAz를 이용하도록 한다.

그림 5는 기본 네트워크 구성을 통해서 센서 모트로부터 어떤 과정을 통해서 신호를 취득할 수 있는지 그 과정을 보여주고 있다. 센서 모트의 경우 초기화를 통해서 센서 초기화 및 센서 값을 받을 준비를 하고 베이스 모트와 무선 연결을 한다. 베이스 모트는 호스트와 인터페이스 방식에 따라서 데이터 전송 방식을 결정하고 관련 부분을 초기화한 다음 센서 모트와 무선 연결을 설정한다. 호스트는 인터페이스에 따라 데이터를 받을 수 있도록 해당 포트를 설정한다. 센서 모트에서 센서 데이터를 패킷의 형태로 연결된 무선망을 통해 베이스 모트로 보내면 베이스 모트에서는 패킷을 그대로 터널링하는 기능만 수행한다. 호스트에서 베이스모트와의 인터페이스를 통해서 받은 패킷에서 오버헤드 부분을 제외한 센서 값을 얻어내고 이를 그래프로 도식화한다.

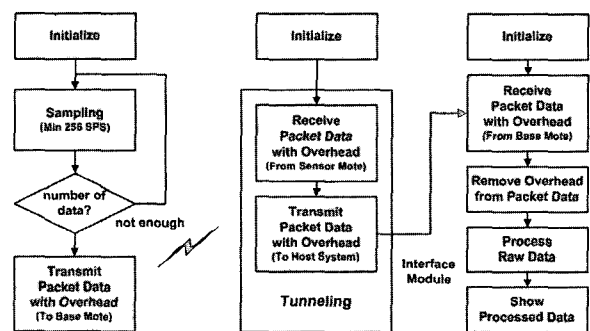


그림 5. 기본 네트워크 구성에서 각 부분의 동작
Fig. 5. Operation of Each Part in Basic Networks

위와 같은 네트워크 구성은 여러 가지 센서 데이터를 호스트에서 직접 이용하기 때문에 멀티 센서 기반 간접 컨텍스트 인식에 해당한다. 하지만 이런 구성만으로는 제한적인 서비스에 제공해주지 못하기 때문에 좀 더 다양한 컨텍스트를 인식할 수 있도록 해야 한다. 사용자의 생체 정보를 얻기 위해서 센서 모트는 사용자의 몸에 부착하는 방식으로 구동

하고 모트의 위치에 따른 상황적 컨텍스트를 이용할 수 있도록 각 모트에 아이디를 부여하고 모트가 설치된 위치 정보를 미리 입력함으로써 좀 더 다양한 컨텍스트 인식을 가능하도록 한다.

본 논문에서 목표로 하는 것은 가정 내에서 센서 네트워크를 구현하고 이를 홈 네트워크의 개념에서 이용할 수 있도록 하는 것이다. 무선 통신은 거리에 한계가 있고 가정은 집집마다 다르며 여러 가지 제약이 존재하는 환경이다. 따라서 다양한 가정환경에 적용이 가능하고 강인한 특성을 가지기 위해서는 1:1의 네트워크 구성 방식보다 정해진 환경에 유연하게 적용할 수 있는 구조가 필요하다. 본 논문에서는 이런 조건을 만족시키기 위해서 센서 모트와 베이스 모트 사이에 중계 모트를 두어서 이 문제를 해결하도록 한다. 중계 모트와 베이스 모트가 가정 내에서 하나의 통신망을 구성하고 센서 모트는 이들 중 연결이 가능한 어느 모트들과도 연결을 할 수 있도록 한다. 이러한 구조를 사용하면 다양하고 제약이 많은 가정환경에도 하나의 단일 네트워크 구조를 형성할 수 있다.

4.2 Service-Oriented Design

어떠한 서비스를 제공하려고 하는가에 따라서 시스템의 인프라가 구축되어야 한다. 본 논문에서는 기본적으로 사용자의 위치를 파악하고 상태를 인식하도록 할 수 있는 일종의 헬스 모니터링 시스템 서비스를 제공하도록 한다. 따라서 센서 모트에는 생체 신호를 인식하기 위한 센서가 필요하다. 위치를 파악하기 위해서는 베이스 모트와 중계 모트들의 위치를 미리 알아야 하며, 실험 환경 내에서 센서 모트와 다른 모트들과의 연결 가능 여부를 통해서 모트들의 통신 범위를 미리 알고 있어야 한다. 이외에 모트에 기본적으로 포함되어 있는 광량 센서와 온·습도 센서를 이용하면 각 위치에서의 환경을 파악할 수도 있다.

그림 6은 중계 모트라는 개념을 도입한 후 구성된 네트워크의 예를 보여준다. 베이스 모트와 중계 모트가 미리 네트워크를 구성을 하고 이 네트워크 환경 내에 센서 모트를 부착한 사람이 들어오게 된다. 센서 모트가 신호를 보내고 기존의 네트워크 망은 이 신호를 받게 되는데 이때 각 모트들의 무선 통신 반경과 미리 알고 있는 모트들의 위치에 따라 상대적으로 센서 모트의 위치를 결정할 수 있다. RF 통신 소자들의 Received Signal Strength Indicator (RSSI)를 지원할 경우 보다 정확한 위치를 파악할 수 있는데, 현재 MoteTrack과 같은 프로젝트가 진행 중이다 [4].

다음으로 어떤 신호를 이용할 것인지에 대해 살펴보아야 한다. 몇 가지 고려해야 할 점이 있는데 반응에 대해서 피드백 반응이 크고 빨라야 한다. 변화율이 둔한 생체 신호는 그 원인을 파악하기가 힘들기 때문이다. 또한 측정이 용이한 동시에 소형화가 가능해야 한다는 점이다. 사용자가 생체 센서를 부착했을 때 거부감을 가지지 않고 생체 신호의 오차율도 줄어줄 수 있을 것이다. 현재로서 심전도(ECG), 용존 산소량 측정(Oximetry), 혈압 등이 가장 조건에 부합하다고 볼 수 있다.

심전도는 피부에서 나타나는 미세한 전위를 측정해서 LNA를 통해 증폭시키는 것이다. 용존 산소 측정법은 적외선과 적색 가시광선이 헤모글로빈의 농도에 따라 투과율이 다르게 측정되는 것을 이용한다. 또한 맥박을 알아야만 혈압을 측정할 수 있다. 따라서 위에서 언급한 방법들은 공통적으로 R-R Interval을 측정할 수 있고, R-R Variability를 이용하면 자율신경계의 활성화도와 밸런스, 스트레스 정도를 측정할

수 있다 [5].

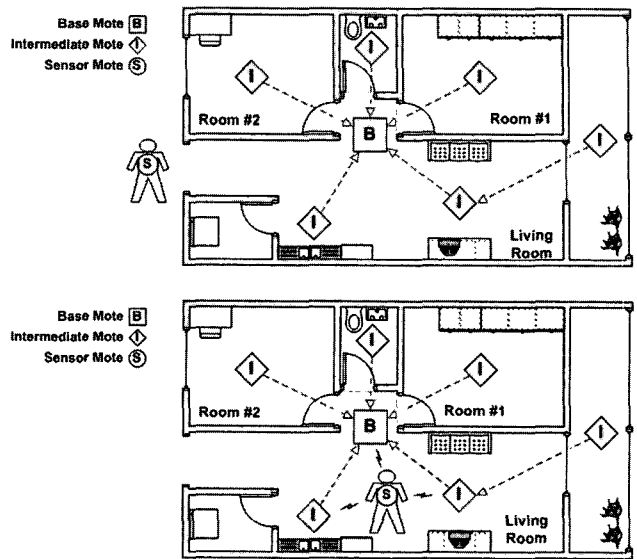


그림 6. 가정 내에서의 네트워크 구성
Fig. 6. Organization of Networks in a Home

4.3 Operation of Home Networks

홈 네트워크의 구성은 센서 모트, 중계 모트, 베이스 모트와 호스트로 이루어진다. 본 논문의 홈 네트워크는 TinyOS에서 제공하는 Multi-Hop 트리구조 네트워크를 이용하도록 한다. 각 부분들의 동작은 네트워크를 구성하기 전 단계, 구성하는 단계, 유지하는 단계로 나누어 살펴보도록 한다.

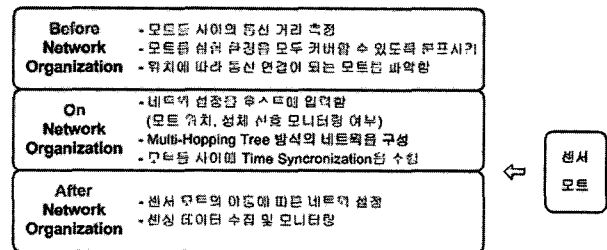


그림 7. 네트워크 동작 과정
Fig. 7. Operation of Home Networks

첫 번째로 네트워크를 구성하기 전에 미리 중계 모트와 베이스 모트 사이의 통신 거리를 측정하고 가정 내의 전 영역을 커버할 수 있도록 실험을 통해 모트들의 위치를 정하고 그 위치의 좌표를 알아야 한다. 또한 센서 모트의 위치를 추론하기 위해서는 센서 모트의 위치에 따라 연결이 되는 중계 모트나 베이스 모트를 파악해야 한다.

두 번째로 네트워크를 구성하는 단계에서는 호스트에서는 미리 측정된 모트들의 위치를 알고 있고, 생체 신호의 모니터링 여부를 결정한다. 각 모트들은 TinyOS의 Multi-Hop 트리구조를 이용해서 네트워크를 구성한다. 네트워크가 구성되고 나면 모트들 간에 Time Synchronization 과정을 수행한다.

마지막으로 네트워크가 구성된 이후에는 센서 모트의 데이터를 수집하고 모니터링 하는 단계이다. 센서 모트들이 이동함에 따라 미리 구성된 네트워크에 연결 설정을 하고 연결된 모트에 따라 센서 모트의 위치를 추론한다. 본 논문의 시

시스템에서 측정하는 신호는 생체 신호와 모트들의 온·습도 정보가 있다. 센서의 데이터의 양이 많고 네트워크의 규모가 커질수록 네트워크 내에 중복되는 데이터들로 인해서 많은 부하가 걸리게 된다. 네트워크 내의 트래픽을 줄이기 위해서 중복된 데이터를 제거하기 위한 방법을 고려해야 한다.

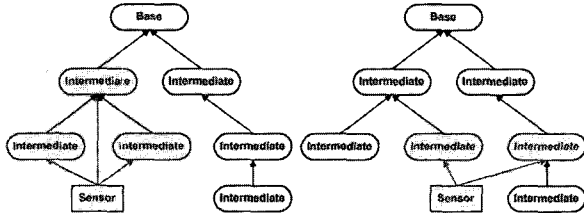


그림 8. 홈 네트워크와 센서 모트의 연결

Fig. 8. Connection between Sensor Mote and Home Networks

센서 모트에서 브로드캐스팅을 통해 센싱된 패킷을 보낼 때, 센서모트의 ID와 패킷 시퀀스 번호, 데이터 수집 시간과 센서 데이터 값을 보내게 된다. 이때 센서 모트와 통신 연결이 된 부분에서는 센서의 데이터 값을 제외한 나머지 정보들을 바탕으로 중복된 패킷을 제거하고 통신 연결이 된 모트들을 커버할 수 있는 상위 모트에 의해서 중복 패킷을 제거하게 된다. 또한 패킷을 연속적으로 받을 경우 지나간 패킷을 검사할 수 없기 때문에 FIFO 버퍼를 통해서 최소한의 패킷을 저장하고 이러한 구조를 통해 패킷 중복 검사를 수행한다. 위의 과정을 거쳐서 중복되지 않은 패킷을 베이스 모트에서 호스트로 보내게 된다.

Sensor ID	Sequence Number	Acquisition Time	Raw Sensor Data
-----------	-----------------	------------------	-----------------

그림 9. 패킷의 구성

Fig. 9. Composition of Packet

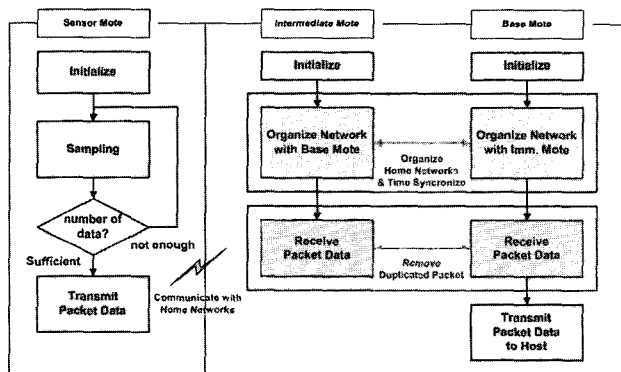


그림 10. 모트들의 동작 과정

Fig. 10. Progress of Each Mote

호스트에서는 우리가 목표로 했던 생체 신호 데이터와 통신 연결 설정을 바탕으로 컨텍스트 인식을 할 수 있다. 다음 그림은 생체 신호 데이터와 통신 연결 설정을 바탕으로 사용자의 위치와 사용자의 상태에 대한 컨텍스트의 연관성을 보여준다. 홈 네트워크를 설정하기 전의 실험을 통해 얻어낸 상황적 컨텍스트와 네트워크상에서 센서 모트의 연결 상태를 통해서 사용자의 위치를 파악할 수 있고, 센서 데이터를 통

해서 사용자의 상태를 파악할 수 있다. 이러한 컨텍스트 인식을 통하여 다양한 서비스를 제공할 수 있는데, 사용자가 위급한 상황에서 응급 연락을 취하고 위치를 알려줄 수 있는 기본적인 서비스를 제공할 수 있을 것이다. 또한 사용자의 생체 데이터를 통해서 기분이나 스트레스 정도를 유추하고 사용자에게 맞는 서비스를 제공할 수 있을 것이다.

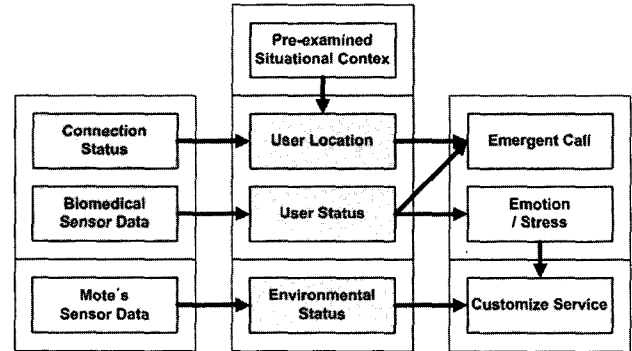


그림 11. 컨텍스트 인식 과정

Fig. 11. Process of Context Awareness

5. 결 론

본 논문에서는 TinyOS의 플랫폼인 MICAz와 Telos를 이용하여 기본적인 홈 네트워크를 구현할 수 있다는 사실을 보였고, 원하는 컨텍스트 인식을 위해서 센서를 선정하고 이를 통해 사용자에게 서비스를 제공할 수 있다는 사실을 보여주었다. 간단한 생체 신호와 플랫폼에서 기본으로 제공하는 센서들만을 이용했기 때문에 인식하고 이용할 수 있는 컨텍스트가 제한이 있을 수밖에 없다. 코드블루와 같은 PMA를 필두로 홈 네트워크 인프라를 구축하고, 앞으로 좀 더 세분화된 컨텍스트를 이용하기 위해서 다양한 센서를 포함하는 시스템을 구상해야 할 것이다.

참 고 문 헌

- [1] TinyOS Community Forum || An open-source OS for the networked sensor regime, [Online] Available: <http://www.tinyos.net>, 2005.
- [2] Motes, Smart Dust Sensors, Wireless Sensor Networks, Cross Bow, [Online] Available: <http://www.xbow.com>, 2005.
- [3] Zigbee, RF transmitter, transceiver and receiver from Chipcon, [Online] Available: <http://www.chipcon.com>, 2005.
- [4] MoteTrack, [Online] Available: <http://www.eecs.harvard.edu/konrad/projects/motetrack>, 2005.
- [5] Electrocardiography-ECG, Laxtha, [Online] Available: <http://www.laxtha.com>, 2005.
- [6] CodeBlue : Wireless Sensor Networks for Medical Care, [Online] Available: <http://www.eecs.harvard.edu/mdw/proj/codeblue>, 2005.

저 자 소 개



황세희 (Se-Hee Whang)
2004년 : 중앙대학교 전자전기공학부
공학사
2004년 ~ 현재 : 동대학원
전자전기공학부 석사과정

관심분야 : 임베디드 시스템
E-mail : just4you@unitel.co.kr



장인훈 (In-Hun Jang)
1993년 : 중앙대학교 제어계측공학과
공학사
1999년 : 동대학원 제어계측공학과
공학석사
1999년 ~ 2000년 : (주)신도리코 기술연
구소 연구원
2000년 ~ 현재 : (주)아로마솔루션 기술연
구소 책임연구원

2004년 ~ 현재 : 중앙대학교대학원 전자전기공학부 박사과정

관심분야 : 지능로봇, 인공두뇌, 지능형 홈 및 홈네트워킹
E-mail : inhun@wm.cau.ac.kr



심귀보 (Kwee-Bo Sim)
1984년 : 중앙대학교 전자공학과 공학사
1986년 : 동대학원 전자공학과 공학석사
1990년 : The University of Tokyo 전자
공학과 공학박사
1991년 ~ 현재 : 중앙대학교
전자전기공학부 교수

2003년 ~ 2004년 : 일본계측자동제어학회(SICE) 이사
2000년 ~ 2004년 : 제어자동화시스템공학회 이사 및
(현) 지능시스템연구회 회장
2003년 ~ 2004년 : 한국퍼지 및 지능시스템학회 부회장
2002년 ~ 현재 : 중앙대학교 산학연컨소시엄센터 센터장
및 기술이전센터 소장
2005년 ~ 현재 : 한국퍼지 및 지능시스템학회 수석부회장

관심분야 : 인공생명, 지능로봇, 지능시스템, 다개체시스템
학습 및 적응알고리즘, 소프트 컴퓨팅(신경망
퍼지, 진화연산), 인공면역시스템, 침입탐지시스
템, 진화하드웨어, 인공두뇌, 지능형 홈 및 홈네
트워킹, 유비쿼터스 컴퓨팅 등

Phone : +82-2-820-5319
Fax : +82-2-817-0553
E-mail : kbsim@cau.ac.kr
Homepage URL : <http://alife.cau.ac.kr>