

유비쿼터스 환경에서 센서 노드의 관리와 망 구성을 위한 RFID 미들웨어 프로토콜에 관한 연구

최 용 식*, 김 성 선**, 신 승 호***

A study on RFID Middleware protocol for management of sensor node and network implementation in Ubiquitous environment

Yong-Sik Choi *, Sung-Sun Kim **, Seung Ho Shin ***

요 약

본 논문에서는 RFID/USN 기반의 유비쿼터스 미들웨어의 센서들은 통상 6개월에서 2년여 기간 동안 동작 가능하더라도 배치 환경에 따라 하루 만에 축전지의 전원을 모두 소모해 버릴 수 있다. 이는 생성된 자료를 목적지까지 전달하는 무선 통신이 센서 전원사용의 큰 비중을 차지하기 때문이다. 센서들이 데이터를 수집하기 위해 통신을 하는 과정에는 목적지에 대한 라우팅 경로 설계가 반드시 필요하다. 그러나 센서의 생존성 향상을 위해 내부 설정을 변경시키기 위해 외부에서 명령을 입력받는 행위는 센서노드의 단순한 경로검색법에 정면으로 대치된다. 그러므로 제한된 자원을 가진 센서노드는 아주 작은 상호작용능력을 부여하기 위해서는 별도의 경로 검색법과 명령체계가 필요하다. 따라서 1:n 형태의 센서배치를 통해 다른 센서들 간의 간섭 없이 명령과 데이터를 송수신 가능한지를 실험하고 시뮬레이션 하고 이형의 센서들로부터 받아들인 센싱(Sensing) 데이터를 실제 정보로 변환하기 위해 센서 데이터 분석과 변환 방법들을 연구하였다.

Abstract

In this paper, Though sensors of Ubiquitous Middle-ware System based on RFID/USN can usually be reacted from six mouths to two years, they can be exhausted their power of storage battery only one day by communication rates and ranges. That is, the lifetime of sensor node is depended on how much spending power under wireless communication that can communicate its sensing data to its destination. Therefore, it is necessary that each sensor should be designed the Routing path to its destination, in order to remote collecting data. But, in order to improve lifetime of sensor node and modify inner setting, it is opposite to simple searching path

• 제1저자 : 최용식

• 접수일 : 2007.5.31, 심사일 : 2007.6.8, 심사완료일 : 2007. 7.19.

* 인천대학교 컴퓨터공학과 박사과정 ** 가천의과학대학교 정보기술과 교수

*** 인천대학교 컴퓨터공학과 교수

method of sensor node by entering external commands. accordingly, 1:n sensor arrangement of n form command and data send-receive that is possible simulation do without interference other sensors and research to different sensor data analysis and conversion ways to convert Sensing data that accept from sensors to actuality information.

▶ Keyword : RFID/USN, Middleware, sensor node

I. 서론

유비쿼터스 컴퓨팅(Ubiquitous Computing)은 새로운 IT 패러다임으로 1988년 미국 제록스 팰로앨토연구소의 마크 와이저(Mark Weiser)가 제안한 개념이다. 이는 일상생활과 컴퓨팅이 접목된 지능화된 환경을 통해 제한 없이 접속하고 쉽게 서비스를 제공받을 수 있도록 발전되었으며, 기술의 비약적 발전을 통해 실현 가능성을 높여지고 있다. 또한 유비쿼터스 컴퓨팅 환경은 네트워크를 기반으로 한 장치간의 연결을 기본으로 하고 있다. 특히, 무선중심의 근거리 통신기술이 발달함에 따라 유비쿼터스 컴퓨팅 환경을 이루는 무수히 많은 개체들은 유기적으로 연결되어, 서로 데이터를 주고받고, 이를 통해 서비스를 제공하게 된다. Wi-Fi, Bluetooth, ZigBee 등의 급격한 통신 기술의 발전, 고성능 프로세서의 초소형화 및 에너지 효율성에 의한 임베디드 컴퓨터의 발전, 그리고 스마트폰, 인터넷 태블릿 Ultra-Mobile PC[1], 휴대용 정보 단말기(PDA, Blackberry[2]) 등의 발전은 공기처럼 주변 환경에 내재된 컴퓨터들이 사물 및 사람과 보이지 않는 네트워크로 연결되고 있다. 사용자에게 필요한 서비스를 제공해주기 위해서는 사용자 및 환경에 대한 동적인 모델을 수용하고 센서(Sensor)를 통해 상황 정보를 인식, 지능적 판단 및 가공을 하고 공유하여 적절한 서비스를 제공해주어야 한다[3]. 상황인식 및 처리 기술은 크게 환경 또는 사용자에게 관련이 있는 빛, 온도, 소리, 움직임 등의 물리적인 정보나 화학적인 정보를 검출하는 센서 기술과, 여러 가지 센서들 및 디바이스 간에 정보를 용이하게 공유할 수 있게 하는 센서 네트워크 기술, 그리고 인식 및 공유된 상황 정보에 따라 사용자에게 제공되는 인터페이스 기술이 있다.

본 논문은 RFID/USN(Radio Frequency IDentification /Ubiquitous Sensor Network) 기반의 유비쿼터스 미들웨어를 설계 구현하기 위한 선행 연구로서 센서와 및 RFID 배치, 센서의 생존성 보장을 위한 접근, 배치된 센서들의 관리와 망 구성을 제안하였다. 센서의 배터리는 통상 6개월에

서 2년여 기간 동안 동작 가능하더라도 배치 환경에 따라 하루 만에 축전지의 전원을 모두 소모해 버릴 수 있다. 이는 생성된 자료를 목적지까지 전달하는 무선 통신이 센서 전원사용의 큰 비중을 차지하기 때문이다. 그렇기 때문에 배치상황에 맞는 센서의 통신 및 자료처리 방식을 원격으로 설정하여 관리상의 이점을 가지게 된다. 센서들이 데이터를 수집하기 위해 통신을 하는 과정에는 목적지에 대한 라우팅 경로 설계가 반드시 필요하다. 그러나 센서의 생존성을 향상 위해 내부 설정을 변경시키기 위해 외부에서 명령을 입력받는 행위는 센서 노드(Sensor node)의 단순한 경로 검색법에 정면으로 충돌한다. 그러므로 제한된 자원을 가진 센서노드는 아주 작은 상호작용능력을 부여하기 위해서는 별도의 경로검색법과 명령체계가 필요하다. 따라서 1: n 형태의 센서배치를 통해 다른 센서들 간의 간섭 없이 명령과 데이터를 송수신 가능한지를 선행하여 실험하고 시뮬레이션하고 이형의 센서들로부터 받아들인 센싱(Sensing) 데이터를 실제 정보로 변환하기 위해 센서 데이터 분석과 변환방법들을 추가로 연구하였다.

II. 관련연구

2.1 MANET

바일 애드-혹(Ad-hoc) 네트워크는 이동 노드들이 동적으로 임의의 네트워크 토폴로지를 구성할 수 있는 네트워크 시스템이다. 최근 그 중요성이 부각되어 IETF의 MANET 워킹그룹에서 연구가 활발히 진행 중이다. MANET은 노드들의 움직임에 따라 토폴로지가 동적으로 변화하는 특성을 가지며, 이에 따른 다양한 토폴로지 제어 알고리즘이 등장하게 되었다. 이는 기존의 유선 네트워크에서의 라우팅 알고리즘을 적용하기에 많은 문제점의 원인이 되고 있으며, 네트워크 토폴로지를 효과적으로 구성하고 이를 적절히 유지하기 위한 알고리즘의 연구가 활발히 진행되고 있다. 그림 1은 애드-혹 네트워크의 구성형태를 나타내며 (a)는 종점간(Peer to Peer) 통신을 수행하는 독립 애드-혹 네트워

크 구성을 나타내며, (b)는 이러한 종점간 통신이 기간 망에 연결되어 기간망의 장점과 독립 애드-홉 네트워크의 장점을 동시에 가질 수 있도록 구성한 것이다[10].

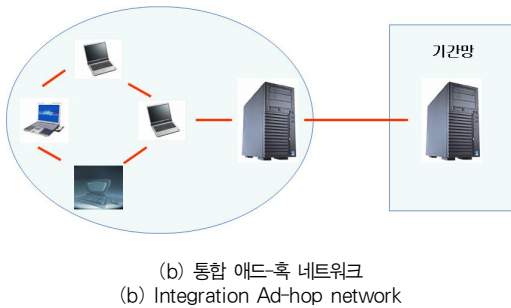
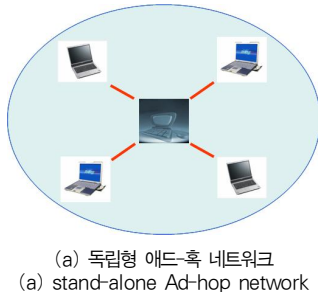


그림 1. MANET의 구성
Fig 1. MANET Architecture

2.2 무선 센서 네트워크

WSN(Wireless Sensor Network)은 국내에서 USN(Ubiquitous Sensor Network)이라는 용어로 많이 사용되어지고 있다. 유비쿼터스 센서 네트워크는 광범위하게 설치되어 있는 유무선 네트워크 인프라에 상황 인식을 위한 다양한 센서들을 장착하여 센싱 된 정보를 이용하여 응용서비스를 제공하는 경우는 그림 1의 (b)와 같다. USN은 지능형 물류관리 시스템과 시큐리티, 침입 탐지, 군사, 방재 시스템 등에 사용된다. 기존 네트워크와 달리 센서 네트워크는 다음과 같은 특징을 갖는다[5][6].

- 센서 노드는 애드홉 망에 비하여 매우 많은 노드를 가진다.
- 센서 노드는 매우 적은 전력을 소비해야하며, 장시간 작동되어야 한다.
- 센서 노드는 고장이 나기 쉽다.
- 센서 노드는 매우 낮은 연산 능력과 저장 공간을 갖는다.
- 센서 노드는 Point to Point 통신이 아닌 브로드캐스트 통신을 한다.

- 센서 노드는 센서의 수와 오버헤드로 인해 전역적인 유일한 ID를 가질 수 없다.

즉, 부분적인 서브 망을 구성하게 된다. 이러한 센서 네트워크의 성능은 다음의 세 가지 범주로 분류되어 평가된다. 첫째, 에너지 효율성, 둘째, 전송 데이터의 정확성, 셋째, 서비스 품질이다. 이들 중 센서 노드의 동작 시간을 최대화하기 위해 에너지 효율성에 대한 연구가 집중되었으며, 최근에는 에너지 효율성과 더불어 데이터에 대한 안정성과 서비스 품질에 대한 연구도 활발히 이루어지고 있다 [7][8][9].

2.3 RFID 미들웨어

이미 선진각국은 이러한 관련 기술을 다양한 각도에서 연구해왔다. 연대 중반에 제록스의 연구소의 마크 와이저는 80년대에 유비쿼터스의 개념을 정립하고 현재의 컴퓨팅 환경의 문제점을 지적했으며 이후 다양한 연구를 진행했다.

미국 선 마이크로 시스템 사는 전자기기들을 서로 연동해 분산동작하게 하는 시스템인 지니(JINI)를 이용해서 네트워크에 연결된 모든 정보기기에 서비스를 제공하는 연구 중이다[1].

마이크로소프트는 지난 년에 복수의 운영체제와 응용 프로그램들이 플랫폼에 관계없이 상호 연동해 모든 장치에 접근할 수 있는 분산 환경 구축전략인 닷넷(.NET) 전략을 발표했다.

네트워크에 연결된 무수한 기기를 어디서나 언제라도 네트워크에 접근해 e-비즈니스까지 수행할 수 있는 환경을 의미하는 IBM의 퍼베이시브 컴퓨팅(pervasive computing) 또한 유비쿼터스 컴퓨팅 개념에 상당히 근접해 있다. IBM은 기술의 실현을 위해 복잡한 데이터 관리, 확정성 있는 컴퓨팅, 전송 기술과 알고리즘 등이 통합된 미들웨어 개발에 주력하고 있다.

MIT는 90년대 중반부터 컴퓨터 사이언스 랩이 주도하는 옥시젠(Oxygen)이란 프로젝트를 통해 퍼베이시브 컴퓨팅 개념을 정립했으며 이미 상용화 되어 있는 다양한 제품을 통합하는 방법으로 유비쿼터스 컴퓨팅의 미래상을 제시했다.

독일의 프라운호퍼 연구소의 경우 Roomware라는 프로젝트를 통해 기존의 공간과 사용자가 임베디드 장치를 통해 효율적으로 통신하는 방법을 연구했으며 산출물의 일부는 이미 상용화되어서 판매되고 있다.

일본 동경대학의 사카무라 켄 교수의 주도하에 진행된 TRON 프로젝트는 초기 가전에 장착되는 저기능의 마이크

로프로세서용 임베디드 운영체제로 시작되어서 현재 일본출시 가전의 약 60%를 상회하는 제품에 채택되었으며 이러한 통합 플랫폼을 바탕으로 유비쿼터스 컴퓨팅의 개념으로 진화하고 있다.

프랑스의 INRIA ObjectWeb 에서 주관하는 컨소시엄이 1999년에 시작되어 지금도 WWW, Enterprise Java Bean Server, J2EE CORBA-compliant 응용 서버에서 효율적으로 통합될 수 있는 오픈소스 분산객체용 미들웨어에 대한 연구가 활발히 진행되고 있다.

국내에선 유비쿼터스 지향 응용서비스의 부재로 업체에서의 개발보다는 연구소나 학교에서 다양한 시도가 이루어지고 있다. 또한, 한국의 경우 기술을 인식하고 개발의 필요성을 인지한지 얼마 되지 않았지만, MIT Lab에서 이미 상용화되어 있는 다양한 제품을 요소로 하는 개념의 COTS(Commercial Off the Shelf)접근방식의 개발과 이미 구축되어 있는 유·무선 인터넷 인프라를 활용할 경우 상용화에 상당한 우위를 차지할 수 있는 계기가 될 것이다.

(재)유비쿼터스 컴퓨팅 사업단은 유비쿼터스 전반적 원천기술 개발을 목표로 하고 있다. 유비쿼터스 서비스의 하부 센서노드로부터 상위 서버 그리고 외부 망까지의 글로벌한 유비쿼터스에 사용될 다양한 미들웨어들이 개발되어지고 있다. 특히, 유비쿼터스 환경의 분산 지향형 미들웨어와 컨텍스트 어웨어 미들웨어(context aware middleware)에 지식기반의 멀티 모달 내용들이 연구되어 지고 있다. 정통부는 컨트롤 미들웨어(HAVi, Jini, UPnP, Lonworks)의 단순 기능을 제공하는 기존 미들웨어의 한계를 극복하여 유·무선 통합 네트워크에 연결된 다양한 기기들의 상호 운용성을 보장하고 이들 기기를 동적으로 연동하여 다양한 유비쿼터스 서비스에 적합한 미들웨어가 개발되어 지고 있다.

유비쿼터스형 미들웨어의 정형적 모습은 현재 존재하지 않는다. 이는 상기에서 언급한 유비쿼터스 서비스 환경의 다양성에 의존하기 때문이다. 이러한 이유로 현재까지 연구 개발 사례를 유비쿼터스 미들웨어가 아닌 유비쿼터스형 미들웨어로 표현을 하고 있다. 결과적으로 유비쿼터스형 미들웨어를 추구하기 위한 기본 요소들은 확장성, 신뢰성, 이기종, 가용성, 견고성, 이동성, 보안성 그리고 상호 운용성 등 기술의 특징으로 정의한다. 향후 다양한 응용 서비스에 공통적 프레임워크를 기본으로 한 유비쿼터스형 미들웨어의 개발이 진정한 유비쿼터스의 구축에 지름길이 될 것이다 [4].

III. 연구 결과 및 고찰

3.1 센서노드 데이터송수신/명령을 위한 미들웨어 프로토타입(UIC V.0.3.0.4)

센서노드 대한 데이터 송수신 주기를 제어하고 이를 화면상에 표시하기 위해 그림 2와 같다. 구현된 결과는 RFID 및 센서노드를 제어하기 위해 다음 5개의 구현부로 나눈다. 시리얼 통신부, 노드명령부, 토폴로지 뷰, 데이터베이스 로그, statistic view이다.

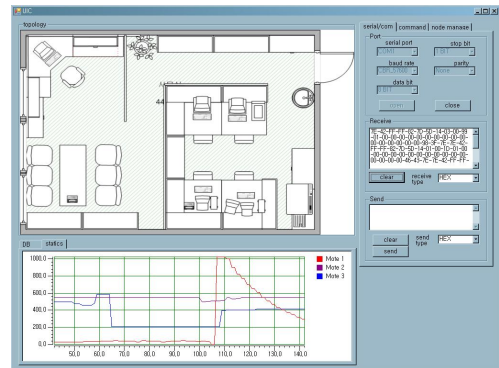


그림 2. UIC V.0.3.0.4 구현물
Fig 2. Implementation of UIC V.0.3.0.4

그림 3은 어떠한 흐름으로 센서데이터를 수집하고 처리하는지를 보여주고 있다. 최초 원격지에서 생성된 센싱된 데이터는 자신의 라우팅 경로를 통해 이동되어 베이스 노드에 집중된다. 베이스 노드에 모인 데이터는 시리얼통신을 통해 RFID/센서노드의 데이터를 처리하는 미들웨어인 UIC v3 의 시리얼 단자에 이벤트를 발생시킨다. 미들웨어에 데이터 수집되었음을 알리는 이벤트가 발생되면 데이터베이스에 데이터 로깅을 위해 복사되어 입력되고 statistics /analysis module 과 센서 토폴로지 망(Sensor topology mesh)을 형성하기 위해 필터링 된다. 이후에 사용자는 센서 커맨더를 통해 해당 센서에 특정한 명령을 내려 센서의 설정 등을 변경한다.

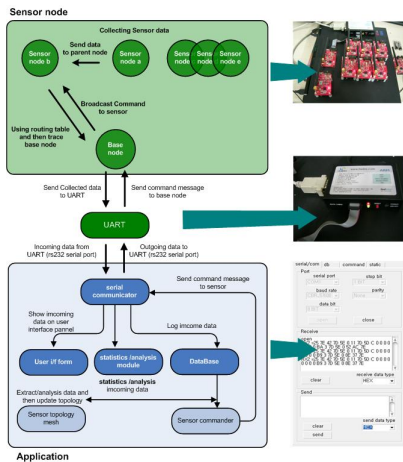


그림 3. UIC 데이터 흐름도
Fig 3. Data flow of UIC

그림 4는 UIC v3 의 클래스 다이어그램을 나타내고 있다. RFID/센서노드 데이터 수집 미들웨어 수집을 위한 퍼이럿프로젝트의 프로토타입 클래스이다. 데이터 수집을 위해 별도의 에이전트를 사용하는 범위를 벗어나는 패턴들은 고려하고 있지 않으며, 소수의 단순한 1:n 형태의 망구조를 만족하고 어플리케이션을 통해 명령을 브로드캐스트 하기 위해 단순화된 구조를 바탕으로 설계하였다.

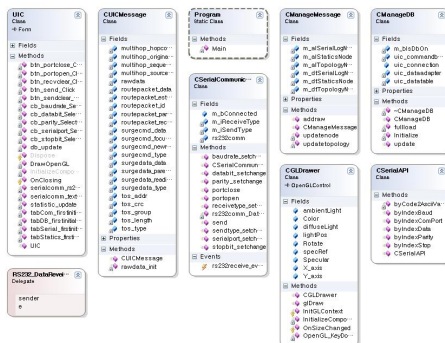


그림 4. UIC v3 클래스 다이어그램
Fig 4. UIC v3 class diagram

3.2 모듈별 기능 설명

3.2.1 데이터베이스(DataBase) 로깅

시리얼 단자를 통해 들어온 데이터는 CManageDB의 인스턴스에 삽입되기 전에 통신 부호를 걸러낸다. 데이터베이스

스에 입력은 history 노드와 info 노드로 구분되며 발생 데이터를 저장하는 순수한 로그를 위해 history 노드를 사용하고 동적인 토폴로지를 유지하기 위해 info 노드를 사용하여 저장한다. 그림 5와 같이 저장된다. 저장된 데이터는 센서 노드간의 데이터 송수신 상태와 네트워크 상태 분석에 사용될 수 있다.

DB	[statics				
Master node	addr	type	group	MetaID	data
	511	130	125	1	112
	511	130	125	1	112
	511		125	1	122
Info node					

그림 5. 데이터베이스 로그
Fig 5. DataBase log

3.2.1 시리얼 통신

센서로부터 수집된 데이터는 PC의 rs232 케이블을 통해 직렬통신을 수행하여 미들웨어에 모인다. 그림 6의 시리얼 통신 폼에서 시리얼 통신의 해당 PC의 데이터 수집 방식을 설정 한다. PC의 가변적인 포트를 제외한 센서모듈의 기번 설정은 1 stop bit, 57600 baud rate, none parity, 8 data bit 이다.

serial/com | command | node manage |

Port

serial port COM1 ▾	stop bit 1 BIT ▾
baud rate CBR.57600 ▾	parity None ▾
data bit 8 BIT ▾	
open	close

Receive

```
7E-42-FF-FF-82-7D-5D-14-01-00-7A
-00-00-00-00-00-00-00-00-00-00-
00-00-00-00-00-00-31-AD-7E
```

clear

receive type

HEX ▾

Send

clear

send type

HEX ▾

send

그림 6. 시리얼 통신 폼
Fig 6. Serial communication form

3.2.3 static view

센서 데이터가 데이터베이스에 수집이 될 때 해당 센서에 해당하는 데이터 종류는 사전에 정의되어 있어야 한다. 이는 센서노드의 통신부와 센서부가 서로 독립적으로 동작하기 때문이며, 해당센서의 수집 데이터가 무엇인지 알고 있을 때 그림 7의 데이터 통계표가 유용하게 사용될 수 있다. 그림 7은 이형의 데이터 센서들이 하나의 데이터 입력 통계 뷰에 보여주는 것을 보여주고 있다. 그림에서 y축은 데이터의 크기를 나타내고, x축은 0.0 을 기준으로 데이터의 입력 주기를 나타내고 있다. 그러므로 해당주기 x축의 값은 특정시간이 지난후의 센서 데이터 값의 변화를 나타낸다. y축의 값은 최대 10비트(Decimal 1024)의 값을 가진다. 이 값은 센서에서 측정한 아날로그 값을 ATmega128 MCU의 ADC를 통해 디지털 값으로 변환된 값이다. ADC가 변환을 완료하면 그 결과는 ADC 데이터 레지스터 ADCH:ADCL에 저장된다. 단극성 입력 채널을 사용하는 경우의 변환결과는 $ADCH:ADCL = V_{in} / V_{ref} * 1024$ 와 같이 표현된다. 여기서 V_{in} 은 아날로그 입력전압이고, V_{ref} 는 선택된 기준전압이다. 아래의 그래프를 예를 들어 설명하면, 파란색 Mote3의 x축 20.0의 y축 값은 600이다. 이것을 식으로 나타내면, $600 = \frac{x}{3.3V} \times 1024$ 이 된다. 그러므로 입력전압 x

는 1980mV가 된다. 이 전압을 해당 센서에 식을 이용하면 수치화 할 수 있다. 이 그래프에서 y축의 값은 ADC의 출력 값을 나타내고 있다.

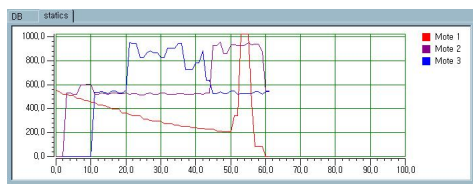


그림 7. static view 데모
Fig 7. static view demo

3.2.5 센서 커맨드

센서 커맨드는 특정 센서의 설정을 원격으로 변경시키기 위해 사용되는 명령부이다. 대상과 수행 명령의 종류를 결정하면 명령어 byte code가 조합되어 생성되고, 브로드캐스트 형태로 이를 전송하는 명령을 수행하게 된다. 그림 8과 같다. 명령어 바이트 코드는 메시지를 받는 MoteID와 커맨드의 종류로 Tos msg 구조체의 데이터 부분에 매핑되어 전송된다.

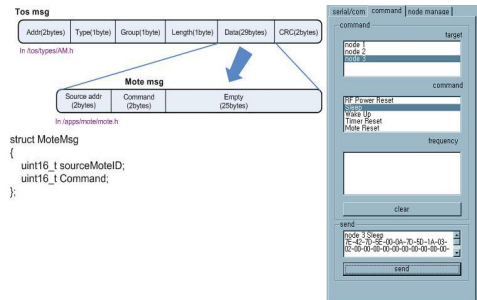


그림 8. 커맨드 뷰
Fig 8. Command View

3.3 센서 데이터 덤프

센서 데이터 덤프는 센서로부터 생성된 데이터를 cygwin 상에서 인터럽트하여 보여주고 있다. 이러한 데이터들은 구현물에 따라 사이즈와 구조가 상이하나, 본 논문에서는 편의를 위해서 31byte로 제한하여 송수신 한다. 그림 9와 같다. 각각의 바이트 값이 의미하는 것은 그림 8에 나타나 있는 Tos msg 구조와 동일하다.

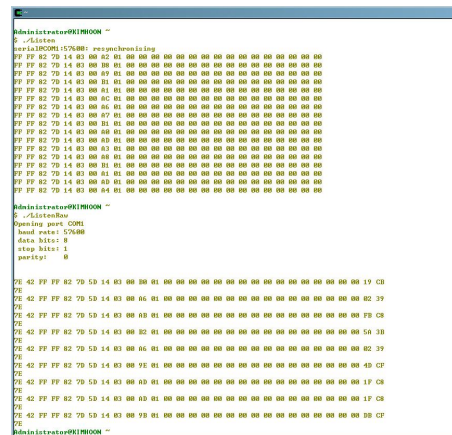


그림 9. 샘플 데이터
Fig 9. Sample Data

Listen 명령은 수신된 메시지를 Tos 메시지 형태로 보여준다. 반면 ListenRaw 명령은 시리얼 포트로 들어오는 모든 메시지 패킷을 보여 주고 있다. 바이트 스트림의 맨 처음과 끝의 바이트는 RS232통신의 FLAG 값이다.

3.4 센서모듈 송수신 메시지 구조

서지(surge)의 Tos msg를 개량한 경량화된 형태의 mote msg를 구성하였다. 센서모듈은 기본적으로 라우팅

될 경로와 데이터를 담아야 한다. 대개의 경우 데이터들은 베이스 노드로 집중되어 모이도록 라우팅 테이블이 구성된다. 그러나 연구결과 특정센서를 제어하기 위해 미들웨어에서 명령을 재처리하여 송출하는 경우 명령은 베이스 노드에서 폐기되기 때문에 이를 피하기 위해 자체적으로 메시지 구조와 라우팅 방법을 구성하였다. 그림 10과 같이 Tos msg를 31byte로 제한하였다. 이는 사용 센서모듈의 업체의 스펙에 종속된다.

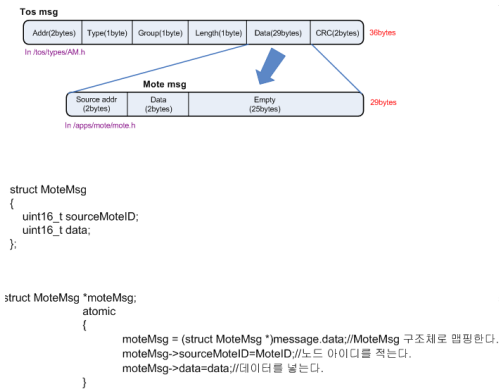


그림 10. Mote msg
Fig 10. Mote msg

3.5 센서모듈 Load 코드

TinyOS의 mote msg는 범용적으로 사용되는 것이기 때문에 필요 없는 정보의 전송이 이루어질 수 있다. 이에 경량화 된 lite mote msg를 제안하였습니다. lite mote msg를 송수신하기 위해 센서모듈에 탑재될 Mote hex image를 추가로 개발하여 각각의 센서모듈에 기록 하였다.

그림 11은 PC에서 생성된 메시지들이 어떻게 전파되는지 보여주고 있다. 센서로부터의 데이터 수집과정 이후의 단계는 수집된 데이터를 분해하고 분석하여 의미 있는 자료로 변환한다. 이후 사용자가 정의한 몇몇 기준에 의해 특정 센서의 설정을 변경하게 된다. 그러나 단순히 센싱 데이터를 수집하는 과정과 명령을 주고받는 행위는 센서노드의 제한된 자원사용 측면에서 상이한 구조를 가지게 된다. 애드-혹 기반의 이동형 통신 장비와 단말PC의 통신상 유사점은 상대 장비를 인식하기 위해 고유한 ID를 부여한다는 점이다. 이러한 ID로는 IP 주소나 MAC 주소를 들 수 있다. 그러나 이마저도 부족하여 IPv6로의 확장을 통해 부족한 주소 체계를 해결하고 있다. 센서노드들은 이와는 비교할 수 없이 열악한 주소체계로 통신한다. 28여개 미만의 채널과

각 채널 당 216개의 ID로 구분된다. 센서 노드와 단말 PC가 서로 상호작용할 수 있도록 명령어 체계와 망구성에 대한 프로토타입 실험을 진행하고 망 구성 이전단계인 1:N 형태로 구성 하였다.

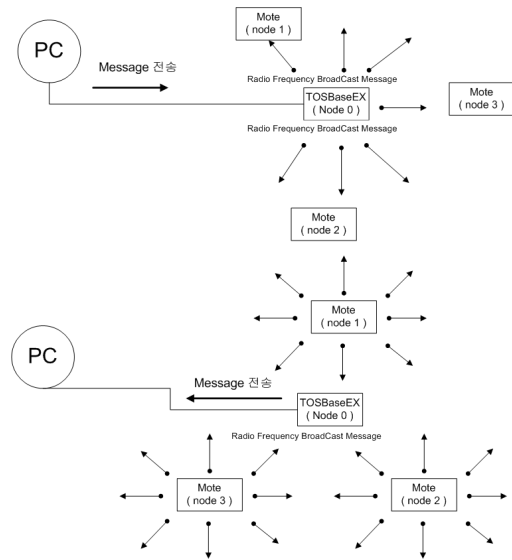


그림 11. Mote msg 운용도
Fig 11. Mote msg operability

3.6 센서 데이터 출력

3.6.1 동종 다수 센서 데이터 수집

동종의 3개의 조도 센서들을 이용하여 데이터를 수집하여 PC에 저장하였다. 그림 12은 PC 데이터에 수집되는 상황을 보여주고 있다. 조도인 빛의 밝기는 AD(analog to digital) 변환하여 이진형태로 표시된다. 그래프에서는 일정한 빛을 밝기가 유지되는 것을 알 수 있다. 논문에서는 각 조도 센서의 데이터 값을 빛의 밝기인 조도(Lux) 형태로 변환하지 않고 보여주고 있다. 그 이유는 조도 이외에 다른 유형의 센서와 혼용하여 보여주어야 하는 상황 때문인데, 동종의 센서가 아닌 경우 서로 유형의 센서들은 자신만의 독특한 데이터 변환 체계를 가지기 때문이다. 대표적으로 조도센서는 Lux로 표현되고, 가속도 및 기울기 센서는 중력가속도인 G로 나타내지며, 동작감지센서는 범위 내 감지에 대해 Yes/No의 형태로 표현된다.

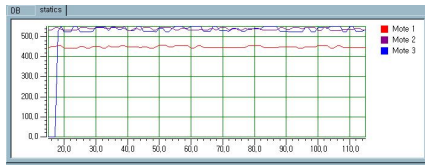


그림 12. 조도 센서 데이터 수집
Fig 12. Lux sensor data capture

3.6.2 이종 다수 센서 데이터 수집

그림 13에서 mote 1 은 습도 데이터의 수집을 보인다. mote 2는 가속도 및 기울기 센서의 데이터 변화를 보이고 있다.

마지막으로 mote 3 은 인체 감지 센서를 통해 적외선 감지 데이터를 수집하고 있다.

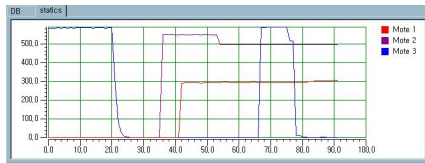


그림 13. 습도, 가속도, 기울기 및 인체감지 센서 데이터 수집
Fig 13. Moisture, acceleration, inclination and motion sensor data capture

3.6.3 배치 및 실험

센서 데이터를 수집하기 위해서 센서 노드들을 그림 14와 같이 배치하였다. 실험은 각 센서에 맞는 환경을 가정하여 배치하였다. 온도 센서의 경우 실내의 전열기 에 배치하고 습도센서의 경우 정수기 근방에 배치하여 실험 데이터를 산출하였다.

실험을 통해 제한된 영역에서 센서들은 망을 구성하고 데이터를 수집하였다. 그리고 Mote hex image 들은 정상적으로 Load 되었다. 즉 각각의 센서들에 상호 망을 구성하여 하고 개별적인 동작하고 각 센서들은 적절한 배치 위치와 패턴에 대한 데이터를 수집하였다.

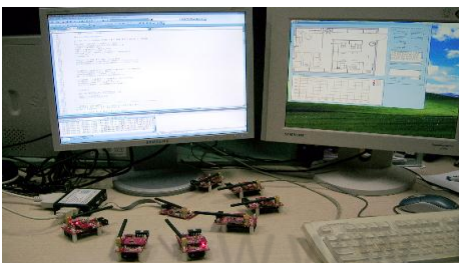


그림 14. 센서 배치 및 데이터 수집
Fig 14. layout sensor and data capture

IV. 결론

유비쿼터스 컴퓨팅 환경은 네트워크를 기반으로 한 장치 간의 연결을 기본으로 하고 있다. 특히, 무선중심의 근거리 통신기술이 발달함에 따라 유비쿼터스 컴퓨팅 환경을 이루는 무수히 많은 개체들은 유기적으로 연결되어, 서로 데이터를 주고받고, 이를 통해 서비스를 제공하게 된다. 사용자에게 필요한 서비스를 제공해주기 위해서는 사용자 및 환경에 대한 동적인 모델을 수용하고 센서를 통해 상황 정보를 인식, 지능적 판단 및 가공을 하고 공유하여 적절한 서비스를 제공해주어야 한다[3].

본 논문은 RFID/USN 기반의 유비쿼터스 미들웨어를 설계 구현하는데 있다. 선행 연구로서 센서와 및 RFID 배치, 센서의 생존성 보장을 위한 접근, 배치된 센서들의 관리와 망 구성을 제안 하였다. 배치된 센서들이 낮은 주기로 외부의 전력 지원 없이 단독으로 사용되는 경우 효율적인 통신 주기와 범위는 센서의 생존성과 직접적으로 영향을 받는다. 특정 센서가 통상 6개월에서 2년여 기간 동안 동작 가능하더라도 배치 환경에 따라 하루 만에 축전지의 전원을 모두 소모해 버릴 수 있다. 이는 생성된 자료를 목적지까지 전달하는 무선 통신이 센서 전원사용의 큰 비중을 차지하기 때문이다. 그렇기 때문에 배치상황에 맞는 센서의 통신 및 자료처리 방식을 원격으로 설정하여 관리상하여야 한다. 센서들이 데이터를 수집하기 위해 통신을 하는 과정에는 목적지에 대한 라우팅 경로 설계가 반드시 필요하다. 그러나 센서의 생존성을 향상을 위해 내부 설정을 변경시키기 위해 외부에서 명령을 입력받는 행위는 센서노드의 단순한 경로 검색법에 정면으로 충돌한다. 그러므로 센서노드는 아주 작은 상호작용능력을 부여하기 위해서는 별도의 경로검색법과 명령체계가 필요하다. 따라서 1: n 형태의 센서배치를 통해 다른 센서들 간의 간섭 없이 명령과 데이터를 송수신 실험을 하여 이형의 센서들로부터 받아들이는 센싱 데이터를 실제 정보로 변환하여 데이터를 수집하고 변환 하였다.

참고문헌

- [1] Project Orgami, Microsoft.
http://origmiproject.com/default.aspx/ resource.
- [2] Blackberry, Research In Motion.
http://www.blackberry.com/ resource.
- [3] Schilit, B., Adams, N., and Want, R. "Context-Aware Computing Application." Proceeding of the Workshop on Mobile Computing Systems and Application, Santa Cruz, CA., pp. 85, 1994. Schilit95a.
- [4] (주)휴인스 기술연구소, '유비쿼터스 무선센서 네트워크 [구조 및 응용]', 홍릉과학출판사, 2006.
- [5] I. F. Akyildiz, S. Weilian, Y. Sankarasubramaniam, and E. Cayirci, "A Survey on Sensor Networks", Communication Magazine IEEE. vol. 40, no. 8, pp. 102-114, Aug. 2002.
- [6] National Research Council, Embedded, Everywhere: A Research Agenda for Networked Systems of Embedded Computers, National Academy Press, 2001.
- [7] H. Saito and H. Minimi, "Performance Issues and Network Design for Sensor Networks", IEICE Trans. On Commun., vol.E87-B, no. 2, pp. 294-301, Feb. 2004.
- [8] D. Ganesan, A. Cerpa, W. Ye, Y. Yu, J. Zhao, and D. Estrin, "Networking Issues in Wireless Sensor Networks", Journal of Parallel and Distributed Computing(JPDC), Special issue on Frontier in Distributed Sensor Networks, Dec. 2003.
- [9] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, "Next Century Challenges: Scalable Coordination in Sensor Networks", Proc. of the 5th Annual International Conference on Mobile computing and Networks (MobiCOM '99), Seattle, WA., pp. 263-270, Aug. 1999.
- [10] "A Comparison of Topology Control Algorithms for Ad-hoc Newworks"

저 자 소 개



최 응 식

2001년 인천대학교
컴퓨터공학과(공학사)
2003년 인천대학교
컴퓨터공학과(공학석사)
2003년 ~ 현재 : 인천대학교 컴퓨
터공학과 박사과정 재학 중
<관심분야> 컴퓨터통신,
임베디드 시스템, 암호학



김 성 선

1976년 인하대학교
전자공학과(공학사)
1981년 인하대학
전자공학과(공학석사)
2005년 인천 대학교
컴퓨터공학과(공학박사)
1993년~현재 가천의과학대학교
기술정보과 교수
<관심분야> 데이터베이스,
e-CRM



신 승 호

1979년 경희 대학교
전자공학과(공학사)
1981년 경희 대학교
전자공학과(공학석사)
1985년 경희 대학교
전자공학과(공학박사)
1986년~현재 인천대학교
컴퓨터공학과 교수
<관심분야> 컴퓨터 통신,
신호처리, 암호학