

유비쿼터스 센서 네트워크 기반의 생활환경 자동제어 시스템 설계 및 구현

The Design and Implementation of Automatic Control System of Living Environment Based on Ubiquitous Sensor Network

윤 지 훈, 문 승 진*
(Ji-Hoon Yun and Seung-Jin Moon)

Abstract : The ubiquitous sensor network technique is widely applied to variety of information fields such as home automations, logistics, traffic controls, public administrations, health and environment monitoring and etc. It is particularly useful in the areas where energy consumption is minimal and where continuous monitoring of the surrounding environments, which generates streams of data, are required. In this study, we have designed and implemented a living environment automatic control system which collects the streams of temperature, humidity, light and noise data of a simulated house setting in real-time fashion, then controls the home environment based on the collected data according to the users favorites. In order to differentiate the proposed system from the currently existing similar system, we have demonstrated not only the feasibility of collecting data using sensor network in the controlled environment but also the ability to control the various household equipments through wireless communications.

Keywords : sensor network, embedded systems, living environment, automatic control, ubiquitous computing

I. 서론

특정 기기에 구애받지 않고(any-device), 누구라도(anyone), 언제(anytime), 어디서나(anywhere), 다양한 미디어(any-media)를 즐길 수 있는 환경을 말하는 유비쿼터스 센서 네트워크(ubiquitous sensor network)는 주변 환경의 변화를 감지할 수 있는 센싱 기능과 센싱된 데이터를 처리할 수 있는 프로세서 등을 추가 하여 이들 간 무선 네트워킹이 가능하도록 이루어져 실시간으로 통신이 가능하게 하는 형태를 가진다[15].

유비쿼터스 센서 네트워크는 설치가 용이하고 범위의 확장성이 뛰어나며 고장이 나더라도 주변의 유사한 데이터를 수집하는 노드가 있으므로 네트워크가 담당하는 영역 내 정보를 얻을 수 있고 센서들이 배터리에 의해 동작되므로 이동이 가능하여 재배치가 가능하다는 장점을 가지고 있다. 이러한 다른 기술이 가지지 못한 장점 때문에 유비쿼터스 센서 네트워크는 스마트 태그, 스마트 홈, 위치추적, 인명구조 등의 위치추적 분야, 원격 센서 및 위치 인식 장치 등을 기반으로 하는 각종 제어 분야, 신체 관리 모니터링 및 의료진 위치 파악 등을 요구하는 의료 관련 분야 등 새로운 핵심적인 응용 분야 개발 및 관련 시장에서의 높은 선점 가능성을 예상하고 있다[13,14,17].

본 논문에서는 유비쿼터스 센서 네트워크 기반 기술을 적용해 가정에서 온도, 습도, 조도, 소음 등 실시간으로 발생된 데이터를 수집 및 분석 후 임베디드 장비인 X-Hyper 270에서 모니터링하고 사용자가 원하는 최적의 환경 조건

에 맞게 그와 관련된 가전 기기의 제어가 가능한 생활환경 자동제어 시스템을 구현하였다[1-4].

본 논문에서 제시된 생활환경 자동제어 시스템은 각 방마다 온도, 습도, 조도, 소음에 대해 변화를 줄 수 있는 가전기기를 두었다. 기존의 유비쿼터스 센서 네트워크 역할인 환경 정보에 대한 측정에 뿐만 아니라 가전 기기 작동을 조정하는 환경 제어 노드 한 개를 두어 환경 정보를 측정하는 노드를 포함 방마다 두 개씩의 노드를 설치하였다. 환경 정보 노드를 통해 측정된 데이터 값이 X-Hyper270으로 전달이 되면 X-Hyper 270에서는 데이터 값을 계산하여 LCD창에 환경 정보를 디스플레이 하게 된다. 다시 X-Hyper270을 통해 사용자가 설정해 놓은 온도, 습도, 조도, 소음의 최적 상태를 환경 정보 노드로 부터 전송 받은 데이터 값과 비교하여 상태에 벗어나게 되면 환경 제어 노드에게 그에 맞는 가전기기의 제어 값을 전송하게 된다. 환경 제어 노드는 연결 되어 있는 컨트롤 모듈을 통해 가전기기를 작동하게 명령을 내리게 된다[11,12].

전체적인 구성은 2장에서는 본 논문에서 사용된 유비쿼터스 센서 네트워크 하드웨어 플랫폼과 플랫폼에 들어간 TinyOS, NesC 그리고 모니터링을 한 Hyper270에 대해서 설명하였고 3장에서는 생활환경 자동제어 시스템을 설계하였다. 4장에서는 3장을 바탕으로 시스템을 구현하였고 마지막 5장에서는 구현한 시스템에 대하여 실행 및 분석 하였다.

* 책임저자(Corresponding Author)

논문접수 : 2007. 9. 29., 채택확정 : 2007. 10. 26.,

윤지훈, 문승진 : 수원대학교 컴퓨터학과

(messblues@suwon.ac.kr/sjmoon@suwon.ac.kr)

※ 본 논문은 2007년 산학 협력실 사업에서 지원하였음.

II. 관련연구

1. TinyOS와 NesC

TinyOS는 UC버클리에서 진행해온 WEBS(Wireless

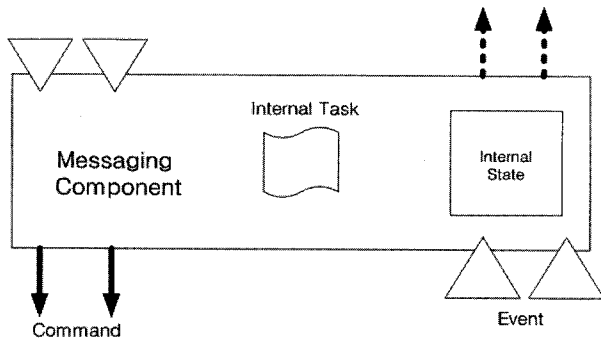


그림 1. TinyOS 컴포넌트 모델.
Fig. 1. TinyOS component model.

Embedded System) 프로젝트의 일환으로 스마트 더스트 (Smart Dust)를 사용하기 위해 개발된 컴포넌트 기반의 임베디드 운영체제이다. 매우 작은 메모리로 멀티 태스킹을 지원할 수 있는 이벤트 기반 모델로 초당 4만 번의 컨텍스트 스위칭을 지원하고 이벤트 발생이 없을 시에 CPU를 슬립 모드로 전환함으로써 초소형 모트의 중요한 특징인 적은 전력 소모가 가능하게 한다[5,6].

TinyOS는 재사용이 가능한 일련의 시스템 컴포넌트를 제공하여 각각의 애플리케이션에서 사용하고자 하는 컴포넌트만을 이용하여 구성할 수 있고 서로 선점하지 않는 태스크와 이벤트를 통해서 보류된 동작은 즉시 리턴하고 스케줄러가 태스크를 후에 수행할 때까지 컴퓨터이션을 뒤로 미룬다[7].

그림 1은 TinyOS의 컴포넌트 모델로 커맨드 핸들러, 이벤트 핸들러, 고정 크기의 프레임, 태스크로 구성된다.

NesC는 임베디드 네트워크 시스템을 위한 프로그래밍 언어로 제한된 메모리와 저 전력의 제한 조건에서 이벤트 기반의 동작 및 유연한 동시 동작 모델 컴포넌트에 기반한 애플리케이션의 특성에 맞는 프로그램 모델을 제공한다.

NesC는 C언어에 기반 하여 확장되었으나 동적 메모리를 할당 하지 않는 정적 언어, 안정성을 위해 전체 프로그램에 대한 분석을 통한 최적화 수행, 컴포넌트 개념에 기반하여 직접적으로 TinyOS의 이벤트 기반 동시성 모델 지원등의 새로운 특징을 지니고 있다[10].

2. Sensor network hardware platform

1999년 버클리 대학에서 Wec라는 모트가 개발된 이후 미국의 크로스보우사(Crossbow)에 의해 상업화 되어 시장에 공급되고 있다. 표 1은 그와 관련된 TinyOS 모트의 종류이다.

센서 시스템은 센서 및 데이터 추출보드, 프로세서/무선 보드 및 게이트웨이/네트워크 인터페이스 보드로 구성되어 있다. 프로세서/무선통신 보드는 저 전력 단거리 무선 통신을 지향하는 8비트 마이크로 컨트롤러와 저속 RF CC2410 칩을 기반으로 구성되어 있다. 그림 2는 크로스보우사의 MPR2400의 구조이다. 이 모트는 지그비(IEEE 802.15.4)에 호환되며 MCU는 Atmel사의 Atmega128로 집적 되어 있다. 센서 및 데이터 추출 보드는 조도, 마이크, 온도, 가속도 및 자기장 센서 등으로 구성되어 있고 실내 온도 및 조도 감

표 1. TinyOS 모트의 종류.

Table 1. The family of TinyOS motes.

Mote	Wec	rene	dot	mica	mica2	iMote	btMote	Telos
Released	1999	2000	2001	2002	2003	2003	2003	2003
Clock	4MHz	4Mhz	4MHz	4Mhz	7MHz	12MHz	12MHz	20MHz
CPU	Atmel	Atmel	Atmel	Atmel	Atmel	ARM	Atmel	Motorola
Flash (code,KB)	8	8	16	128	128	512	128	60
RAM (KB)	0.5	0.5	1	4	4	64	4	4
Radio (kBaud)	10	10	10	40	40	460	460	250
Radio Type	RFM	RFM	RFM	RFM	Chip Con	Zeevo BT	Ericsson BT	ZigBee

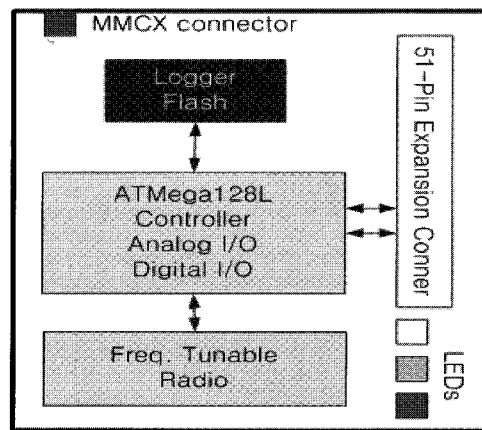


그림 2. 프로세서/무선 보드 MPR2400 구조.
Fig. 2. Processor/Wireless board MPR2400 structure.

지, 미세 동작의 진동감지, 음향감지 등에 다양하게 응용될 수 있다. 게이트웨이/네트워크 인터페이스 보드는 현재 MICA 패밀리 플랫폼을 장착하여 사용하는 것이 대부분이며 Serial Port, USB, Ethernet Port, Wireless 등과 연결 할 수 있게 되어 있다[8,9,16].

본 논문에서 사용한 프로세서/무선 보드는 Hybus사에서 개발한 H-mote로 Telos의 플랫폼과 호환성 및 CC2420 RF 트랜시버를 통해 임베디드 시스템과 연동이 가능하고 USB 방식으로 되어 있어 별도의 게이트웨이/네트워크 인터페이스 보드가 필요하지 않다는 장점을 가지고 있다. 센서 및 데이터 추출보드는 Hybus사의 H-sensor를 이용하였고 초음파, 온도, 조도, 습도, 소음 등의 측정이 가능하다[18].

3. X-Hyper270-TKU

본 논문에 사용된 X-Hyper270-TKU는 Hybus사에서 개발된 임베디드 툴킷으로 인텔 벌버디(Bulverde) PXA270 기반으로 되어있다. SDRAM 128MB, NOR Flash 32MB, NAND Flash 64MB로 되어 있으며 CPLD 사용으로 데이터/어드레스 버스의 Over Shoot/Under Shoot를 줄이고 안정적인 코어 전원설계 및 CPU 발열 방지 기능을 추가하여 안정적이고 고성능의 실험 실습이 가능하게 하였다. GPS, 블루투스, 유비쿼터스 무선 센서 네트워크 장비 연동이 가능하다[19].

III. 생활환경 자동제어 시스템

이번 장에서는 II장에서 본 관련 연구를 바탕으로 생활 환경 자동제어 시스템 설계에 관해서 설명하였다. 먼저 전체 시스템의 흐름에 관해 설명하였고, 3.2절에서는 환경 측정 노드와 제어 노드 모듈의 설계에 관해 설명하였다. 임베디드 장비에 올라가는 생활환경 시스템의 구현에 관해서는 3.3장에서 나타내었다.

1. 생활환경 자동제어 시스템 개발 환경

기존의 무선 센서 네트워크의 모니터링 기능에서 임베디드 장비 X-Hyper270을 이용하여 모니터링 뿐만 아니라 제어도 가능하게 구현하였다. 그림 3은 생활환경 자동제어 시스템 구조를 보여준다. 가정의 각 방에는 한 개씩의 환경 측정 노드를 두어 온도, 조도, 소음, 습도 값을 측정하여 RF 통신을 이용하여 X-Hyper270에 전송하게 된다.

X-Hyper270에는 RF 통신이 가능하게 하는 CC2420이 탑재되어 있어 센서 노드를 통해 측정된 환경 측정값을 X-Hyper270에 수신 받는 것이 가능하다. X-Hyper에서는 각 방의 상태를 LCD를 이용해 그래프 형식으로 보여주게 되며 사용자가 선택한 최적의 상태에 벗어나면 다시 RF 통신을 이용하여 다른 환경 제어 노드에게 각각 온도, 조도, 소음, 습도에 맞는 기기를 동작 시키도록 한다.

가정에서는 각 방마다 환경 측정 노드 1개와 환경 제어 노드 1개, 제어 노드와 연결 되어 필요한 전압을 제공하기 위한 별도의 AC, DC 보드 1개, 그리고 AC, DC 보드와 연결되어 있는 기계를 둔다.

2. 센서 노드 모듈

센서 노드 모듈은 환경 측정 노드와 제어 노드 각각 1개씩의 다른 모듈을 가지고 있다.

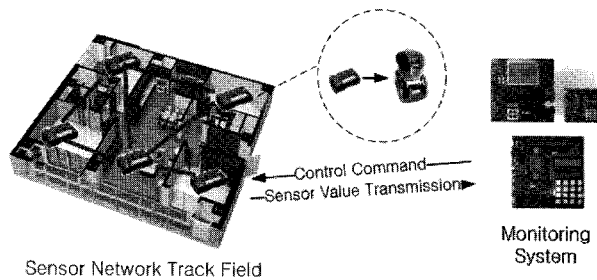


그림 3. 생활 환경 자동 제어 시스템 구조.
Fig. 3. Living environment automatic control system structure.

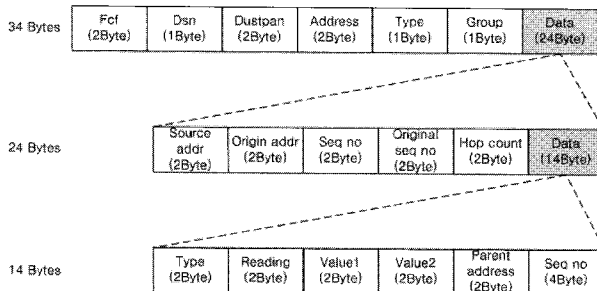


그림 4. 환경 측정 노드 패킷 구조.
Fig. 4. Environment measurement node packet structure.

표 2. 센서 데이터 계산식.

Table2. Sensor data calculation formula.

Temperature	Result = (double) Value1 * 0.01 - 40
Illumination	Result = (double) Value1 * 489 / 45333
Humidity	Temp1 = (double) Value1 * 0.01 - 40 Temp2 = -0.000028 *(double) Value2 + (double) Value2 + 0.0405 *(double) Value2 - 4 Result = (Temp1 - 25) * (0.01 + 0.00008 * (double) Temp2) + Temp1
Noise	Result = (double)Value1 / 30

환경 측정 노드의 흐름은 전원이 켜지는 시점에서 환경 센서(조도, 온도, 습도, 소음)를 초기화로 시작하고 센서의 초기화 이후 환경 센서를 이용한 각 방의 주변 환경정보를 센싱한다. 환경 측정 노드에서는 시간의 간격을 두어 한가지의 센서 값에 대해서만 측정하고 시간이 흐르고 다른 센서 값을 측정한다. 일정한 시간이 지난 후에 변화된 값이 없으면 해당 되는 센서 값의 측정하는 시간의 간격을 넓혀준다. 만일 시간의 간격을 넓힌 상태에서 변화가 있다면 다시 처음의 시간의 간격 상태에서 측정하게 한다.

그림 4는 환경 측정 노드의 패킷 구조이다. 패킷은 총 34Byte로 구성되어 있고 value1과 value2에는 각 상황에 맞는 센서 값을 가지고 있다. 습도의 경우 두 개의 값(온도 값이 필요)이 필요하기 때문에 2개의 필드로 구성 하였다.

제어 노드의 흐름을 보면 각 센서는 AC, DC 보드와 연결되어 있고 AC, DC 보드는 각각 가전기와 연결되어 있다. 임베디드 장비에서 보내는 신호를 받아서 현재 동작 상태이면 멈추게 하고 멈춘 상태이면 동작하게 한다. 임베디드 장비에서 아무런 신호를 보내지 않으면 휴면 상태에 있게 된다.

3. 자동제어 시스템 설계

자동제어 시스템은 패킷 데이터를 수신하거나 송신하게 되며 패킷 데이터는 byte코드로 되어 있으며 판독할 수 있도록 패킷을 하나씩 나눠서 처리되어진다.

환경 측정 노드의 값을 받아오는 부분과 화면에 그래프를 표시 해주는 부분은 쓰레드를 이용하여 다른 동작과 관계없이 실행된다. 조도, 습도, 온도, 소음으로 따로 메뉴가 나누어져 있고 각 값에 따라 최대값과 최소값이 다르게 정해지게 된다. 표 2는 각 측정값에 따른 데이터 계산식을 나타낸다. 센서를 통해 측정된 값을 데이터 계산식을 이용해 화면에 보여주게 된다.

최대값과 최소값은 처음에 정해진 상태에서 사용자가 자신의 최적의 상태에 맞게 조절 할 수 있다. 사용자가 조절 한 최적의 상태에 벗어나게 되면 메시지를 보여주며 각 가전 기기가 연결 되어 있는 제어 보드에 패킷 데이터를 송신하게 된다.

IV. 생활환경 자동제어 시스템 구현

이번 장에서는 III장에서 설계한 내용을 바탕으로 자동제

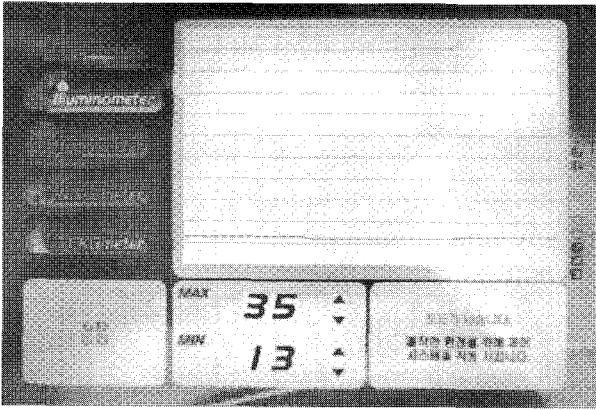


그림 5. 생활 환경 자동 제어 시스템.

Fig. 5. Living environment automatic control system.

어 시스템의 구현에 대해 설명하였다. 4.1절에서는 센서 노드의 모듈 구현에 대해 설명하였고 4.2절에서는 자동제어 시스템 어플리케이션 구현에 대해 나타내었다.

1. 센서 노드 모듈 구현

3.2장에서 설계한 센서 노드 모듈의 구조를 바탕으로 모듈을 구현한다. 환경은 Windows 상에서 Cygwin, TinyOS 1.1.13 package를 사용하였다.

2. 자동제어 시스템 구현

3.3장에서 설계한 자동제어 시스템을 바탕으로 어플리케이션을 구현한다. 프로그램은 리눅스 기반의 GDK 1.0을 사용하여서 임베디드 리눅스 상에서 실행 가능하도록 하였다.

그림 5는 자동제어 시스템을 구현한 화면이다. 메뉴는 온도, 조도, 습도, 소음, EKG(아직 구현 되지 않음)로 나누어져 있다. 메뉴의 밑 부분에는 각 방을 상태를 확인 할 수 있게 되어 있다. 전체적으로는 현재 상태를 그래프로 표현해 주고 있고 밑에 부분에는 최대값과 최소값을 선택 할 수 있게 되어 있다. 그 옆에는 최대값이나 최소값의 상태에서 벗어 날 때 상태를 보여주는 메시지 창이다.

각각의 메뉴를 선택할 때마다 그래프의 색이 바뀌면서 다시 그래프를 그려주게 되고 최대값과 최소값도 온도, 조도, 습도, 소음에 따라 다른 값의 최대값과 최소값을 나타내게 된다.

V. 데이터 측정 및 분석

IV장에서 구현된 프로그램을 이용하여 실제 집과 비슷한 모형에서 데이터 측정 및 그 결과에 대해서 분석을 하였다.

그림 6은 집과 유사한 형태로 테스트 베드를 구현한 것을 보여준다. 온도, 습도, 조도, 소음의 상태에 따라 반응하는 모습을 보여주기 위해 각 방마다 AC, DC 보드와 제어 노드를 두고 거기에 맞는 컴퓨터 파워 팬(온도), 자동차용 가습기(습도), 자동차용 장식품(조도), 경보 알람기(소음)을 배치하였다. 본 논문의 테스트 베드가 실제 주거환경을 축소한 모형이므로, 온도, 습도의 변화와 제어에 대한 반응이 실제와는 다를 수 있다.

테스트 베드를 이용하여 구현된 생활환경 자동제어 시스템의 몇 가지 데이터 측정 및 분석을 하였다. 먼저 환경 측

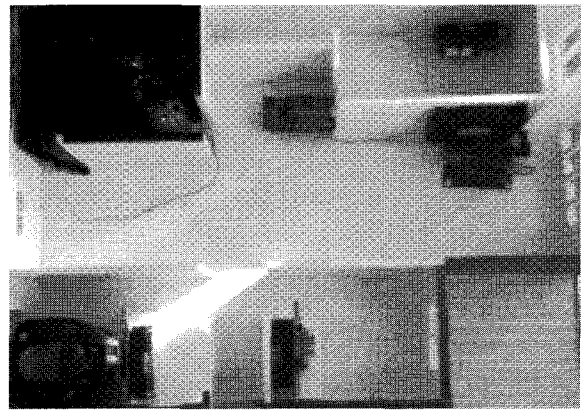


그림 6. 생활 환경 자동 제어 시스템 테스트 베드.

Fig. 6. The test bed of living environment automatic control system.

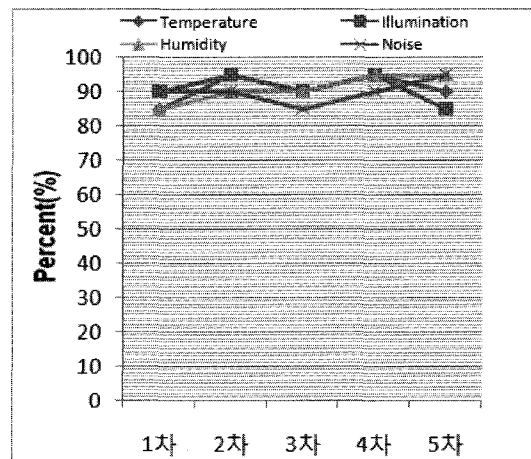


그림 7. 환경측정노드와 임베디드 장비 사이 데이터 전송의 정확성.

Fig. 7. Accuracy of data transfer between environment measurement node and embedded device.

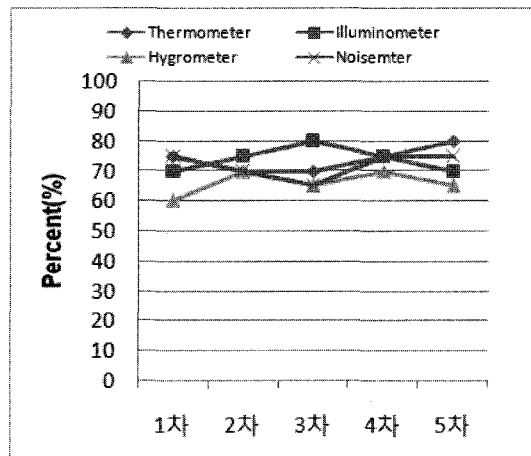


그림 8. 상황에 따른 가전기기 제어의 정확성.

Fig. 8. Accuracy of the home appliance machinery and tools control which according to context aware.

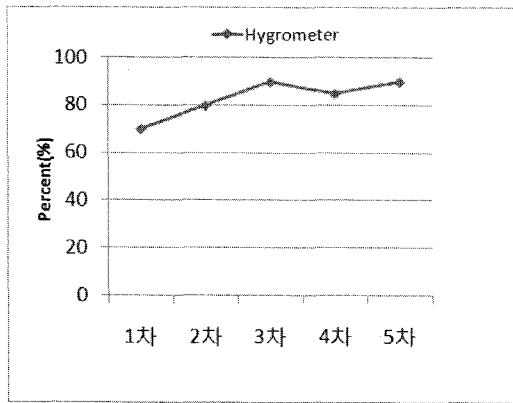


그림 9. 수정된 가전기기 제어의 정확성(습도).

Fig. 9. Accuracy of the home appliance machinery and tools control which is amended.

정 노드와 임베디드 장비 간의 데이터 전송 시에 데이터의 손실 여부를 파악해 정확성이 어느 정도 인지 확인 하였다. 테스트는 각 차별로 20회, 5번 반복하여 총 100회 측정으로 데이터 전송의 성공과 실패를 파악 하였다. 그림 7는 데이터 전송의 성공 여부를 백분율로 나타낸 것으로 측정 결과는 80%~90% 이상의 데이터 전송의 정확성을 확인 할 수 있다.

그림 8은 임베디드 장비에서 가전기기를 제어하기 위한 제어 노드로 데이터 전송의 데이터 손실 여부를 파악한 결과이다. 환경 측정 노드와 임베디드 장비간의 데이터 전송과 마찬가지로 각 차별로 20회, 5번 반복하여 총 100회 측정을 하였다. 결과는 평균 71.5%로 낮은 정확성을 나타내었다. 낮은 정확성을 나타내는 것은 임베디드 장비에서 여러 개의 환경 측정 노드와 환경 제어 노드를 한 개의 칩으로 센서 패키지를 받음으로 이로 인한 데이터 손실이 생겼다. 이에 따라서 임베디드 장비와 가전기기 상의 데이터 전송의 데이터 손실을 줄이는 방법으로 임베디드 장비에서 제어 노드의 데이터 전송 횟수를 기존의 1회에서 2~5회로 높여서 각각 1차에 1번 2차에 2번 각 차별로 20회, 5번을 시도하였다. 결과는 그림 9와 같이 전송 횟수를 높일수록 정확성이 높아지는 결과를 나타내었다. 3번 이상 데이터를 전송의 경우에는 정확성의 차이가 없고 전송 횟수가 많을수록 임베디드 장비에서의 자동제어 시스템의 속도가 느려지는 결과가 있어서 3번의 전송 횟수가 가장 적합한 것을 확인 할 수 있었다.

VI. 결론

본 논문에서는 유비쿼터스 센서 네트워크를 이용한 거주 환경 모니터링 시스템을 제안하였다. 제한된 방법은 노드를 이용하여 여러 가지의 환경 정보를 확인하여 임베디드 장비에 디스플레이 해주고 사용자가 원하는 최적의 상태를 다시 임베디드 장비에서 노드에게 동작을 지시하여 노드와 연결되어 있는 가전기기가 그에 맞는 동작을 취하도록 되어 있어 주거지역에서 통합적인 환경 제어를 가능하도록 한다.

본 논문에서 제안한 자동제어 시스템은 센서 네트워크의 데이터 수집의 역할 뿐만 아니라 센서 네트워크를 통해 가전기기의 제어가 가능하도록 개선하였다. 이러한 내용을 바탕으로 실생활에서 가전기기 내부에 장착하게 된다면 사용자가 하나의 임베디드 장비를 통해 주거지 내부에 있는 모든 기기 등을 쉽게 제어 할 수 있을 것으로 예상된다.

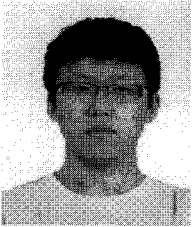
따라서, 향후 연구과제로서 본 연구에서 제안한 방법을 통해 기존에 제공 되었던 가전기기뿐만 아니라 환경에 필요한 방법 시스템, 오염도 측정 등 실생활에 더 심층적인 부분에 대한 연구 및 개발을 바탕으로 새롭게 도출될 수 있는 문제점과 개선점을 찾아 주거 환경의 전반적인 관리가 가능하도록 이루어져야 한다. 이러한 과정에서 도출된 결과는 앞으로의 센서 네트워크를 통한 주거환경 모니터링 및 제어분야에서 이용될 것으로 보인다.

참고문헌

- [1] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk and J. Anderson, "Wireless sensor networks for habitat monitoring," *WSNA '02*, September 2002.
- [2] T. Arici and Y. Altunbasak, "Adaptive sensing for environment monitoring using wireless sensor networks," *WCNC*, March 2004.
- [3] C. D. Kidd, R. Orr, G. D. Abowd, Christopher G. Atkeson, Irfan A. Essa, Blair Macintyre, Elizabeth Mynatt, Thad E. Starner and Wendy Newstetter, "The aware home : A living laboratory for ubiquitous computing Research," *CoBuild '99*, October 1999.
- [4] M. Leoncini, G. Resta and P. Santi, "Analysis of a wireless sensor dropping problem in Wide-area environmental monitoring," *IPSN '05*, April 2005.
- [5] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler and K. Pister, "System Architecture Directions for Networked Sensors," *ASPLOS 2000*, November 2000.
- [6] C. Srisathapornphat, C. Jaikaeo, and C.-C. Shen, "Sensor Information Networking Architecture," *ICPPW '00*, 2000.
- [7] P. Levis, S. Madden, D. Gay, J. Polastre, R. Szewczyk, A. Woo, E. Brewer, and D. Culler, "The emergence of networking abstractions and techniques in TinyOS," *NSDI '04*, March 29-31, 2004.
- [8] Ian F. Akyildiz, Y. Sankarasubramanian, W. Su, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, August 2002.
- [9] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "The design of an acquisitional query processor for sensor networks," *SIGMOD*, June 2003.
- [10] D. Gay, P. Levis, R. V. Behren, M. Welsh, E. Brewer, and David Culler., "The nesC language: A holistic approach to networked embedded systems" *PLDI 2003*, June 2003.
- [11] 이기욱, 성장규, "유비쿼터스 센서 네트워크 기반의 상황 정보 모니터링 시스템 구현," 한국 컴퓨터정보학

회 제 11 권 제 5 호, 2006.

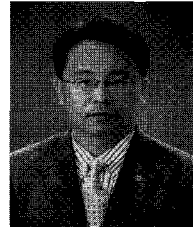
- [12] 성하준, 김대진, “지능형 임베디드 홈 오토메이션 서버의 구현,” 전자통신기술연구소 논문지 제 8 권 제 1 호, 2005.
- [13] 김대영, 김재인, 성종우, 이강우, “센서 네트워크 운영 체제/미들웨어 기술동향,” IITA 주간기술동향제 1221 호, 2005.
- [14] 강정훈, “센서 네트워크 기술 동향과 응용서비스 구현 사례”, 한국전자부품연구원, EIC기술 세미나, 2005.



윤 지 훈

2007년 수원대 컴퓨터학과 졸업. 2007년 동 대학 석사 과정. 관심분야는 실시간 센서 네트워크 시스템, 임베디드 시스템, 유비쿼터스 컴퓨팅.

- [15] 휴인스 기술연구소, “유비쿼터스 무선 센서 네트워크,” 홍릉과학출판사, 2006.
- [16] 남상엽, 하이버스, “유비쿼터스 센서 네트워크 응용,” 상학당, 2007.
- [17] 박승창, 남상엽, 류영달, 이기혁, 김완석, “유비쿼터스 센서 네트워크 기술,” 진한 M&B, 2005.
- [18] Hybus, “H-Mote Manual,” Hybus, 2007.
- [19] 남상엽, 노성동, 노태상, 우종정 “PXA270을 이용한 임베디드 시스템 구조 및 응용,” 상학당, 2006.



문 승 진

1986년 The University Of Texas At Austin 졸업. 1991년 The Florida State University 석사. 1997년 동 대학 박사. 1997년~현재 수원대학교 컴퓨터 학과 부교수. 관심분야는 실시간 멀티미디어 리눅스, 실시간 모바일 데이터베이스, 실시간 임베디드 시스템, 실시간 센서 네트워크 시스템.