

도시건물의 센서네트워크환경의 방재 및 피난대피 유도시스템에 관한 연구

Research on a Fire Detection and a Guide System in Building
on the base of Sensor Network

권창희*

한세대학교 IT학부, 컴퓨터공학과, 조교수

Chang Hee Kwon*

Dept. of Computer Engineering, Hansei University

e-mail : kwonch@hansei.ac.kr*

(031)-450-5254

<요약문>

본 연구는 재해를 줄이고 철저한 대비를 하기 위하여 재난에 관한 정보를 체계적으로 수집 및 분석하고 이를 활용 할 수 있는 소방방재 또는 정보시스템이 요구되고 있다. 도시방재관리시스템이란 관점에 있어서 도시건물의 센서네트워크 환경의 방재 및 피난대피 유도시스템 모델 구축을 통한 효과 및 개선방안을 고찰 하였다.

<Abstract>

To prevent and minimize damages from disasters, it is necessary to build a fire detection or a guide system that can gather and organize proper information and that can analyze about disasters. This research aims at suggestion of fire detection and guide system in building on the base of sensor network. To do so, it examines the effects of this suggesting model system based urban disaster management system.

Key words : fire detection, sensor network, urban disaster management system

1. 개요

Sensor Network에 대한 이론적 부분과 응용적 측면에 대한 접근에 있어서 건축구조물에 도입하여 건축구조물에 화재 발생시 수동적인 대응(소화기나 비상구 표시, 스프링 쿨러 등)의 기존의 문제점을 해결할 수 있는 방법을 제시하였다. 건축물의 각 층에 분산되어있는 Mote가 실내 환경정보(예를 들면 온도, 조도, 가스 등)를 수집하고, 이를 각 Mote간의 Routing을 통해 Gateway로 Data를 전송하여 Gateway 및 각 서버 프로그램들은 Mote로부터 수집된 데이터를 분석하여 평상시에는 실내 환경 안내 및 자동화 기능을 수행하게 한다. 화재 발생시 각 Mote에서 수집된 Data 분석 및 A star 알고리즘을 통해 보다 안전한 비상 통로를 계산하여 각 층에 설치된 안내 전광판으로 전송하여 비상상황에 대비하도록 하였다.

2. 시스템구축 내용

(1) 시스템 구성

그림 1에서 보는 바와 같이 각 층 양쪽에 Sensor Node(Mote)가 위치하고 각각의 Mote들은 Gateway와 RF(Radio Frequency)와 연결 되어있다. 또한 이를 통합관리하는 Host System이 있다. 이들의 역할은 평상시 Server로부터 기본적인 정보(예를 들면 온도, 습도 등)를 최적 경로를 계산하여 안내하는 역할을 하게 된다. 서버 시스템은 Gateway Server, Analysis Engine, Host Server, Monitoring Viewer로 구성되며, 각각의 서버는 Database와 상호 연결되는 구조로 설계되었다. 각 층마다 2개씩 총 6개의 Mote와 Gateway Server와 연동하여 Gateway역할을 할 Mote 1개를 합하여 총 7개를 설치하였고, Host System은 Hybus사의 X-Hyper255B 보드 3대를 이용하여 구현하였다. 본 시스템은 그림 2에서 보는 바와 같이 총 세 부분으로 구성되어 있다. 센서링 하는 센싱과 이를 RF패킷화 하여 전송할 Sensor Part와 RF 패킷수신과 이를 분석하고 모니터링 할 Server Part 및 분석된 정보를 바탕으로 각 외부 기기를 총괄 관리 및 컨트롤하는 Host System 등으로 설계하였다. Sensor Part는 Sensor Node(이후 Mote)는 온도와 조도 정보를 수집하여 RF Packet화 시킨 후 Gateway역할을 하는 Mote로 전송하게 된다. Gateway는Gateway Server와 Serial로 연결하여 RF Packet을 전달하는 기능을 하게 되는 데 Server Part의 구성은 Database를 중심으로 Gateway Server, Monitoring Viewer, Analysis Engine, Host Server로 구성되며, 각각의 서버는 Database를 통하여 통신하는 구조로 설계하였다. Host Part는 XScale(PXA 255B)기반의 Embedded System Board위에 구현하였으며, 각각의 기기들을 컨트롤 하기 위한 보드와 USB용 선풍기 및 USB용 전등으로 구성하였다.

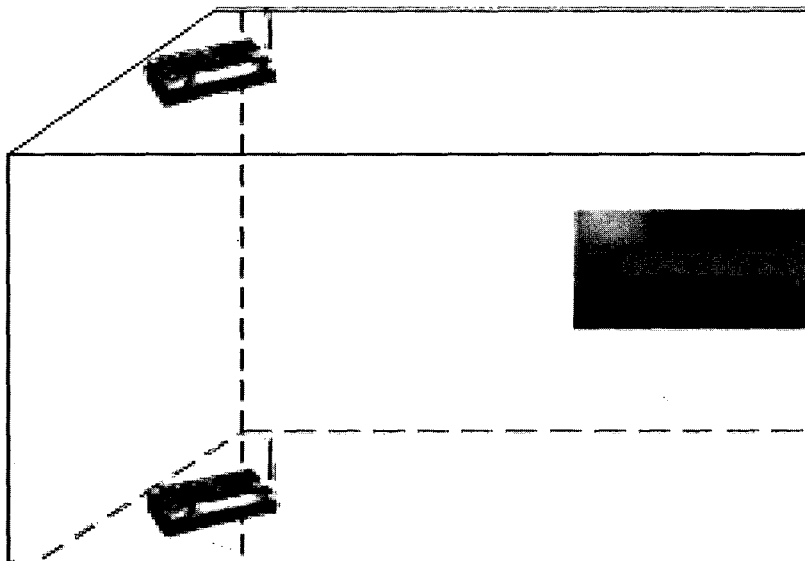


그림 1 시스템 개념도

Server Program의 구성은 기본적으로 ADO(ActiveX Data Object)로 구성된 모듈이 들어가며 이를 통해Database로 통신한다. Gateway Server는 RF Packet을 분석하기 위한 Parser모듈과 RF Packet을 받기 위한 Serial Communication부분으로 구성된다. Monitoring Viewer는 건물의 상태 정보를 보기 위한 3D Viewer부분과 Oscilloscope Viewer부분, ListViewer부분 그리고 Mote정보, 화재 정보 등을 입력하기 위한 Options 부분으로 구성된다. 위험상황(화재발생)시 길안내 서비스를 담당할 Analysis Engine은

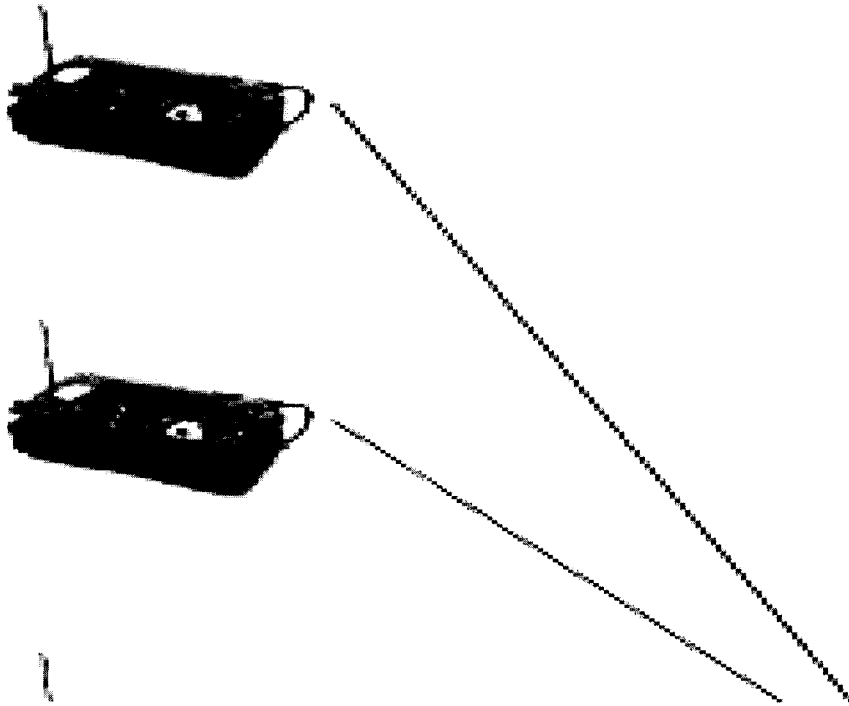


그림 2 시스템 아키텍처

초기화 부분, Path-Finding, Sensing 부분 등으로 나누어 진다. Host Part부의 기동과 동시에 Host Server로 접속이 이루어 진다. Host Server내부는 Client접속을 담당할 Accept Thread부와 Host당 하나씩 생성될 데이터 전송을 담당할 Client Thread부로 구성된다. Host Part부의 Device Control Module은 Host Server로부터 전송된 데이터를 Device Driver를 통하여 Screen Control 및 Machinery Control을 담당한다. 한편, Sensor Part의 구성 요소인 Mote는 NesC로 프로그래밍된 TinyOS(RTOS)기반 위에서 동작한다. Multi Sensing(온도와 조도)이 가능하도록 수정된 Multi-Sensor Base와 RF Packet을 받는 TOSBase로 구성된다. 소프트웨어로는 크게 운영체제와 컴파일러로 나뉘는데, Sensor Part에서 Mote의 운영체제로 TinyOS 1.1.7를 사용하였으며, Server Part는 Microsoft사의 Windows 2000 Family로 구성 하였다. Host Part의 경우 Embedded Linux 2.4.18 for ARM을 사용하였다. 컴파일러의 경우 GNU Toolchain을 사용하였으며, Tiny OS상의 Application을 작성하기 위해 cygwin을 사용하였다. ARV Studio는 Mote에 실행이미지를 다운로드 하기 위한 프로그램으로써 사용하였다. Mote는 (주)Maxfor사의 TIP30CM과 을 사용하였고, Host System은 (주)Hybus사의 X-Hyper255B를 사용하여 구성하였다.

(2) 프로그램 세부 내용

1) Server Part

TOSBaseM은 TinyOS의 커널 부분이라 할 수 있지만 TinyOS는 이것만이 커널이라 할 수 없고 가장 기본적인 부분은 Main 컴포넌트에 있다고 할 수 있다. Sensing Part는 TOSBase의 하위 컴포넌트로 연결되어야 한다. Main 컴포넌트는 CPU포트를 초기화하는 코드가 있으므로 CPU의존적인 부분이라 할 수 있다. HPLUART0M은 PC에 연결되어 Serial통신을 할 수 있게 하는 Serial컴포넌트이다. Comm은 RF(무선통신)를 가능하게 하는 컴포넌트이

다. RF를 지원해야 할 경우 이 컴포넌트를 Main과 연결하게 되는 것이다. CC1000RadioC 컴포넌트는 Chipcon사의 CC1000이란 무선통신을 가능하게 하는 칩을 컨트롤한다. CC1000RadioC는 CC1000RadioM모듈을 연결하고, CC1000RadioM은 CC1000ControlM과 연결되고 CC1000ControlM은 HPLCC1000M과 연결됨으로써 RF(무선통신)가 가능하게 된다. CC1000RadioM에 LedC가 연결되는데 이는 무선통신 패킷이 입출력 시 Led가 토글되는 것을 확인 할 수 있게 하는 컴포넌트이다. 소스(Sensing.nc)는 TOSBase로 센싱한 데이터(온도, 조도)를 보내는 SensingRF이다. Multi-Sensing(온도, 조도)를 위해서 ADC에 연결될 컴포넌트 두개가 있다. 아래 소스는 Serial통신 프로그램으로 중요한 부분만 보기로 하겠다. read_scc()함수는 블록화 되는 함수로 Serial Port에서 대기하고 있다가 포트에 데이터가 들어오면 받아오는 함수이다.

```

if (SCC::serialCtl().read_scc(mess,lenBuff,lenMessage) == TRUE)           ①
{
    if (lenMessage > 0)
    {
        m_moteld = (unsigned int)mess[8];
        if( 0 < m_moteld && m_moteld < MOTE_NUMBER+1){
            m_guid = (unsigned int)mess[5];
            if(m_guid == GROUP_ID){ //GROUP_ID Exception           ②
                m_dstAddr[0] = (unsigned char)mess[2];           ③
                m_dstAddr[1] = (unsigned char)mess[3];
                m_handlerID = (unsigned char)mess[4];
                m_channel = (unsigned char)mess[12];

                if(m_handlerID == HANDLR_ID && m_channel == CHANNEL){
                    result = AbstractData(mess);           ④
                    if(result != FALSE){
                        Display();           ⑤
                        SetSaveDatabase();           ⑥
                    }
                }
            }
        }
    }
}

```

- ① Serial통신하는 ReadFile함수를 호출한다.
- ② ReadFile에서 가져온 데이터를 GUID비교하여 필터링 한다.
- ③ GUID가 맞으면 패킷정보를 변수로 가져온다
- ④ Data구조를 10진수 정수 값으로 변환하는 함수이다.
- ⑤ 앞에서 데이터를 가져온 변수 값으로 GUI로 출력한다.
- ⑥ Database에 저장하는 함수이다.

2) Analysis Engine

분석엔진(Analysis Engine)의 역할은 화재 발생시 Mote와 Exit(출구)의 상태 정보를 변경

하여 미리 저장된 Map데이터와 함께 최적의 탈출 경로를 계산하는데 있다. 분석엔진의 구조는 그림11과 같다. 전체 5개의 Thread를 구성하여 각각의 Thread가 주기적으로 Database 및 Data Struct를 갱신하는 구조로 되어있다. 다음은 각 스레드별 수행과정을 설명한다.

- ① 센서 이상 유무체크 스레드(Sensor Thread)
 - A. Database의 View에서 SensingDataPosView에서 MotelD, Photosensor,, Tempsensor값을 읽어온다.
 - B. 현재 한계값과 비교 후 한계값 이상 올라가면 다음을 수행한다.
 - C. 온도의 이상 체크 카운트 변수를 만들어서 횡수를 체크한다.
 - D. A를 반복한다.
 - E. 만일 지정한 횡수 이상 체크가 되면 다음을 수행한다.
 - F. 관련 Mote Data Struct에 SensorEventType을 SN_EVENT_DANGER상태로 바꾼다.
- ② 온도 이상 유무 체크 스레드(Temp Event Thread)
 - A. Mote Data Struct에서 횡수를 기준 한계횡수와 다시 한번 비교한다.
 - B. 만일 기준 횡수 이상이면 현재 상황을 SensorEvent Table에 등록한다.
 - C. SensorInformation Table의 해당 Mote의 상태 필드를 SN_EVENT_DANGER상태로 갱신한다.
- ③ 방향 계산 결정 스레드(MapPathState Thread)
 - A. Host의 위치를 Source로 Exit의 위치를 Target으로 결정한다.
 - B. Exit에서 수정된 Map데이터를 가지고 A Star알고리즘을 수행한다.
 - C. 수행된 결과를 저장한다.
 - D. STB의 위치를 기준으로 하여 계산된 결과를 가지고 방향 표시를 결정한다.
 - E. 결정이 완료되면 SensorSTBInformation 테이블에 저장한다.
- ④ Map블록화 스레드(MapBlock Thread)
 - A. SensorInformation에서 상태 필드인 SensorTempState이 SN_EVENT_DANGER상태인 Mote를 읽어온다.
 - B. 읽어온 Mote의 위치에 해당하는 Map의 위치값이 위험 지역이 아니면 위험 지역을 바꾼다. (SN_MAPSTATE_UNWALKABLE, SN_MAPSTATE_SENSORDANGERS)
 - C. 각 Mote에 대하여 반복 처리한다.
- ⑤ Exit 차단관련 스레드(ExitState Thread)
 - A. SensorExitInformation 테이블에서 각 층의 출구의 위치를 읽어온다.
 - B. 출구의 위치에 해당하는 Map의 위치 값이 SN_MAPSTATE_UNWALKABLE상태이면 갈수 없는 상태 이므로 ExitState상태값을 1로 바꾼다.
 - C. 각 출구마다 반복한다.

3) 분석엔진

분석엔진(Analysis Engine)에서 사용한 길 찾기 알고리즘은 A Star Path-Finding Algorithm이다. 본 알고리즘을 적용하기 위해 두 가지 측면에서 접근하였다. 우선 2차원 평면상에서 적용된 알고리즘을 확장하지 않고 각 층별 Host에서 시작하여 출구에 이르는 Path를 계산하였다. 3차원 적용을 위해 출구에 대한 상태를 두어 현재 층에서 바로 위층과 아래층의 출구 상태를 판단하여 아래로 내려 갈수 있는지 위로 올라갈 수 있는지를 판단하

는데 사용하였다.

4) Monitoring Viewer

Monitoring Viewer의 역할은 Database에 등록된 Mote의 상태를 보는데 있다. 건물의 입체 정보와 각 층의 Mote의 상태를 보기 위한 Building View와 현재 Sensing 중인 Mote의 자세한 정보를 보기 위한 List View, Sensing한 정보를 그래프로 나타내기 위한 Oscilloscope View와 긴급상황 발생시 Event를 표시하기 위한 List View로 구성되어 있다. 그림 3은 Monitoring Viewer의 실행화면의 일예이다. Monitoring Viewer에서는 Monitoring기능 이외에 Mote, 출구 정보, Host 추가, 삭제를 수행 할 수 있다.

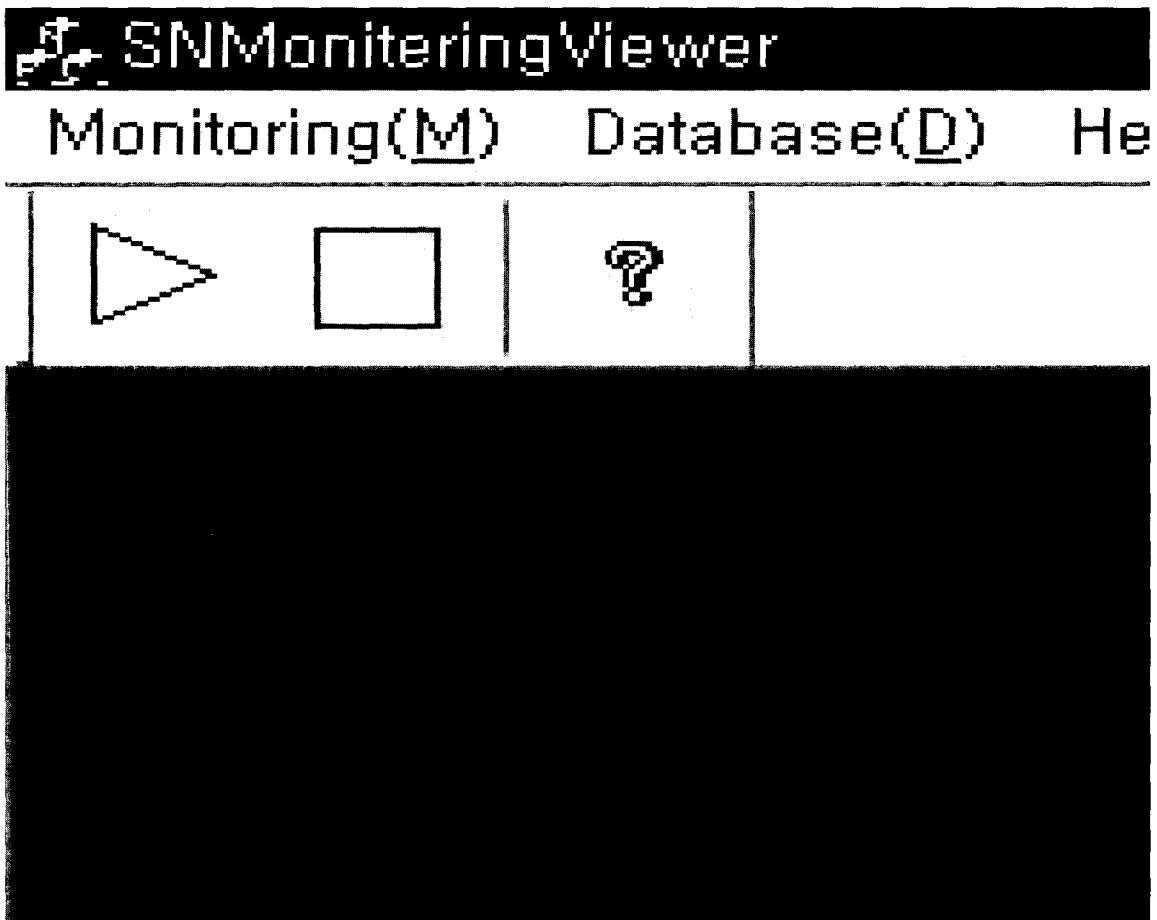


그림 3 Monitoring Viewer 실행 화면

4. 맺음말

본 논문은 Sensor Network를 이용하여 유비쿼터스의 기본 개념인 생각하는 사물(Think thing)을 건물에 적용하였고, Sensing한 정보를 이용하여 비상 상황(화재발생)시 보다 안전한 대피 경로를 자동으로 산출해 낼 수 있는 시스템을 제안하여 보았다. 앞으로 다가올 유비쿼터스 시대에 일반 가정집, 지하 구조물 등 모든 사물에 적용 가능하며 이를 통해 모니

터링, 상태 파악을 통해 또 다른 서비스를 구축할 수 있는 모델을 제시하였다. 본 연구에서 구축한 시스템은 크게 Sensor Part와 Server Part, Host Part로 구성되어있다. 현재 모든 Mote가 하나의 Gateway로 물려있는 상태로 구현이 되었다. 그러나 실생활에 적용 시 Sensor Network의 중요한 부분 중 하나인 Routing 기법을 구현해야 한다.

참고문헌

- [1] Aalen, O. (1978). Nonparametric Inference for a Family of Counting Process, The Annals of Statistics, Vol. 6, 701-726.
- [2] Cox, D.R. and Hinkley, D.V.(1974).Theoretical Statistics, Chapman and Hall, London.
- [3] Kaplan and Meier.(1958). Nonparametric Estimation from Incomplete Observations, Journal of the American; Statistical Association, Vol. 53, 457-481.
- [4] Nelson, W.(1972). Theory and Application of Hazard Plotting for Censored Failure Data, Technometrics, Vol. 14, 945-965.
- [5] Woolf, B.(1955). On Estimating the Relationship between Blood Group and Disease, The Annals of Humanity and Genetics, Vol. 19, 251-253.