SMART ENVIRONMENT

By
Mohsin Taufiq
Rameez Rehman
Umair Hussain Asad

Submitted to the Faculty of Computer Software Engineering Department,
Military College of Signals, National University of Sciences and Technology,
Rawalpindi in partial fulfillment for the requirements of a
B.E Degree in Computer Software Engineering

JULY 2011

ABSTRACT

Smart environment is a technological concept in which the physical world that is richly and invisibly interwoven with sensors to gather and process their sensed data. While using certain communication framework send this information to actuators which take certain actions to control the environment. The SMART ENVIORNMENT project is exactly based on the same idea. The two basic things of interests in our smart environment are intrusion and overheating detection for fire. The reason for the selection of these two entities is that security and safety are critical in all aspects of today's world. In order to ensure safety easily deployable and easy to afford solutions are required. The Smart Environment Project is just one of those solutions that can be used as a basic tool for the detection of intrusion with homes and offices along with the provision to detect overheating. For this purpose, Texas Instrument's Ez430-RF2500 devices are used which are equipped with temperature sensors while PIR sensors for intrusion detection are also attached to these devices for intrusion detection. Ez430-RF2500 are low power and resource constrained which support a star topology network. The basic aim of this project is to establish a multi-hop network with these devices to enhance the coverage area of the network for which IETF ROLL (Routing on Low power and Lossy networks) routing protocol has been implemented on these devices. A desktop application (Smart Monitoring System) is also developed to graphically represent the gathered information from different field sensors. This application generates visual, audio and sms based alerts to notify the end user in case of any critical event.

CERTIFICATE OF CORRECTNESS AND APPROVAL

Certified that work contained in this thesis “Smart Environment” carried out by Mohsin Taufiq, Rameez Rehman and Umair Hussian Asad under the supervision of Asst. Prof. Dr. Fasial Bashir for partial fulfillment of Degree of Bachelor of Software Engineering is correct and approved.

Approved by

____

(Supervisor Name)

_____ Department

MCS

Dated: _______________

DEDICATION

In the name of Allah, the Most Merciful, The Most Beneficent

To our teachers, without whose unflinching support and unstinting cooperation, a work of this magnitude would not have been possible

ACKNOWLEDGEMENT

We are eternally grateful to Almighty Allah for bestowing us with the strength and resolve to undertake and complete the project.

We gratefully recognize the continuous supervision and motivation provided to us by our supervisor, Assistant Professor Dr Faisal Bashir for his continuous and valuable suggestions, guidance, and commitment towards provision for the undue support throughout our thesis work. We are also grateful to the other professors and instructors of the CS Dept for their guidance and support throughout the project. It is because of their help and guidance that we were able to complete the project and thesis work in time.

We would like to offer our admiration to all out class mates and seniors who have been supporting, helping and encouraging us throughout our project.

Last but not the least we are deeply obliged to our families for their never ending patience and support for our mental peace and to our parents for the strength through their prayers.

TABLE OF CONTENT

List of Figures

List of Tables

1 Chapter

Introduction to Smart Environment

1.1 Introduction

Smart Environment is hardware (MSP430 rf2500) and software based project which aims at developing an automated environment with the ability to detect environmental changes such as temperature and intrusion. Hardware and software triggers are an essential part of the Smart Environment as these will initiate the generation of alerts based on the sensed values. The final product is an integration of the hardware which manages communication and the software application. Communication is achieved via the gradient routing protocol which has been developed to enable multi hop routing over a wireless network using the MSP430 rf2500. The routing protocol has been developed in C using IAR embedded workbench and the software in C#.NET. The hardware with the routing protocol is used to provide the routing and the software application used to process this data and generate alerts. This chapter describes the background, basic functionalities of the system which are required to initiate the idea and the specifications which involve the development of the system.

1.2 Background

Smart environments is a technological concept that, according to Mark Weiser is "a physical world that is richly and invisibly interwoven with sensors, actuators, displays, and computational elements, embedded seamlessly in the everyday objects of our lives, and connected through a continuous network" [14].

This project is based on the very same concept. It comprises of various elements that make up a smart environment as stated in Mark Weisers definition.

A rising concern in all aspects of life is security and no compromises can be made here. Human error is not something that can be tolerated in this case and if such error ocuurs the effects can be disastrous.

A well designed and tested automated system which reduces the chance of error is the solution to these problems. A Smart Environment is the solution. One which can sense and notify its users in time to take corrective action, in case of any anomaly in the sensed environment.

1.3 Problem Addressed

The problem of security in houses as well as in small organizations housed in a single home or small buildings is a major concern. Despite the availability of state of the art solutions, the problem persists. Not because these solutions don’t provide the required functionality but due to the cost that makes these solutions unaffordable for the common man and small organizations. Also companies provide solutions such as separate intrusion detection, fire alerts and environmental monitoring as separate modules. Some of the solutions use wired approach and consumes a lot of space and is dependent on electricity i.e. a power failure means automated security is down.

The above stated problems are addressed by the Smart Environment. The Smart Environment is for those users who want both safety and security to protect their environment from fire and intrusion. The product also provides information regarding temperature of the sensed environment and hence can be used in the industry/ agriculture where temperature monitoring is a critical issue.

1.4 Goals and Objectives

Smart Environment is developed to provide the safety and security from fire and intrusion respectively. This system is well suited for environments like small buildings, laboratories and storage places etc. The objective of the smart environment is to provide an application that monitors temperature and intrusion detection and alerts the user in case of overheating and intrusion. The goal is the development of an application i.e. Smart Monitoring System. The Smart Monitoring System has two main components. The first will enable wireless communication in a multi hop fashion to allow the successful sensing and transmission of sensed data. Major concern in this regard is that the sensing devices (sensor) are resource constrained i.e. battery powered, low computational capability and few kilobytes of memory. Deploying a self configuring multi hop routing solution over such resources constrained devices requires on demand routing with energy efficient routing metrics. The second will be the application which will use the information received from the wireless sensor network to allow monitoring and the generation of alerts in case critical situation. In this component, a user friendly GUI and capability to inform remote user regarding sensed data is important. This component provides visual, audio and sms alerts to the user.

1.5 Deliverables

The project deliverable is an application for the monitoring of temperature and intrusion with the ability to generate alerts based on the values sensed from the environment.

In its entirety it should be a self configuring multi hop wireless sensor network for sensing and transmitting information detected by the sensors to a common sink. This data will serve as input for the functioning of the above mentioned application.

1.6 Document Organization

The document includes the basic knowledge of the smart environment. The first chapter describes what project is and explains the basic idea, background, goals and objectives. In second chapter related work has been discussed that has been done in this field. Requirements and design specifications are discussed in the chapter 3 and 4. Implementation of the Smart Environment is elaborated in chapter 5. Testing strategies and results are discussed in Chapter 6 and 7 respectively. The project work is concluded in the last chapter along with the future directions.

1.7 Summary

This chapter starts with a brief introduction and background to the smart environment project. Initial sections are followed by an explanation of the problems that the project addresses and what the Smart Environment project aims to achieve. It then touches upon the final product the project will deliver. In the last section document organization is elaborated.

In a nutshell the Smart Environment project aims at the development of an automated system for the monitoring of temperature (overheating and fire) and detecting intrusion. Moreover in case of such events it defines a mechanism of user motification.

2 Chapter

Related Work

2.1 Introduction

In this chapter existing technologies and solutions which address problems similar to the one the Smart Environment project aims to solve are discussed in length. Also a comparison is done between these and the Smart Environment which is presented at the end of this chapter.

2.2 Burglar Alarm

Burglar or intrusion alarms are electronic alarms to alert user form the intrusion. It has been observed that these types of alarms are commonly connected to some control unit via wiring but in very rear cases radio frequency signals are used for communication. There alarm systems most of the time only detect the opening of the window or door. Few new burglar alarms systems are now using the PIR (Passive Infrared) [17] for detection purpose. PIR sensor detects the motion and generates the intrusion alert. Smart Environment also uses PIR sensors to detect the motion and generate alert for intrusion.

Conventional Alarms have a major drawback that the user has very little control over it and in most of the systems users can only turn it on or off [15], manually. Moreover, information regarding the intrusion is not provided to remote user who is outside the audible range of the alarm.

The figure 2-1 shows a wired burglar alarm which are commonly used in many places.

Figure 2 – 1 Burglar Alarm

2.3 Hold up and Panic Button System

This system is another type of intrusion detection system which has wired/wireless hold up and panic buttons which incorporate silent alarms. This system requires a person who presses the button to send the indication to the control room. These types of systems can only be used in retail and banking, commercial areas or any place where a company's employee is face to face to the customers or people from outside the company. The system is for some limited environments and also requires a person to generate the alert, thus informing the control room that something has gone wrong.

The figure 2-2 shows few hold up and panic button which are often used in banks and offices.

Figure 2 – 2 Hold up and Panic Button System

2.4 Fire Alarm Systems

Fire alarm systems that use sensor such as smoke sensor for the detection of fire. In the case of fire an audio alarm is sounded to alert people in surrounding areas [16].

2.4.1 Fire Alarms with Smoke Detector

Most of these systems use smoke detectors to sound an alarm. They are usually wired and generate false alarms. The system only detects smoke which in some cases generates false alarms like on cigarette smoke or smoke due to cooking. Whereas in case of the Smart Environment, temperature is continuously measured and if at any point in time the temperature exceeds the threshold set by the user, he will be notified.

In figure 2-3 fire alarms which detect smoke and generates alarm for fire.

Figure 2 – 3 Fire Alarm Smoke Detector

2.4.2 Fire Alarms with Pull Stations

Fire alarm systems which use pull station installed all over the building can be pulled by any person to generate alarm in case of fire. These pull stations are connected to main control via wires. The wires could burn in the case of fire and the whole system fails. In the case of Smart Environment the system automatically generates alerts. Also communication is done over a wireless network as opposed to a wired one.

Figure 2-4 show pull stations that are in use in large buildings.

Figure 2 – 4 Pull Station Fire Alarm

2.5 Summary

The existing technologies and systems that are providing similar solutions as Smart Environment have discussed in this chapter. Burglar alarms, Hold up and Panic button, Fire alarms with smoke detector and pull station are discussed. Although, considerable amount of work has been in the field of home/office security and environmental monitoring, however on this project the Smart Environment is developed with very resource constrained and cheap devices. But at the same time, the developed system is not only comparable but also affective than numerous existing solutions.

3 Chapter

Requirement Specification

3.1 Introduction

The requirement specifications of the Smart Environment project are documented in this chapter. The sensor motes (eZ430 RF2500) will sense the environment for temperature and intrusion detection. This data will be delivered to the base station or sink node over a wireless network. On sink this data will be processed and displayed in case of an event such as fire or intrusion alerts will also be generated. Therefore, hardware triggers such as intrusion and software triggers such as sudden increase in temperature are handled by the system.

3.2 Project Scope

The Smart Environment is a self configuring multi hop wireless sensor network that will use temperature sensors and PIR sensors to sense temperature and intrusion respectively. There will only be a single sink i.e. the final destination no matter what route is followed will be the sink node. It is aimed for small buildings and closed environments. The software application will allow the monitoring, setting application parameters, visual and audio alerts and sms to the remote user.

3.3 Overall Description

The overall descriptions of the system including the product features, perspective and network perspective are elaborated in this section.

3.3.1 Product Features

The final product is intended to have a number of features. These features are provided using different modules in a way to achieve the overall functionality. The basic product features are elaborated in the remaining of this section.

3.3.1.1 Multi Hop Routing Protocol

The Smart Environment must allow all active devices (motes) to communicate with each other in a multi hop fashion i.e. each node should forward data packets to the next node and this process should continue until the packet reaches the sink node. This is a major requirement of the system as the current routing protocol implemented one the sensor motes does not support multi hop routing (over 2 hops).

This requires the selection and implementation of a multi hop routing protocol that can be implemented on the resource constrained meZ430 rf2500 hardware. Multi hop routing will enable these motes to establish a mesh topology and will increase the range f observed phenomena.

3.3.1.2 Data Transmission

The Smart Environment must periodically sense the environment for any anomaly. Moreover, it is also required to transmit this sensed information to the base station using the multihop routing mechanism.

3.3.1.3 Smart Monitoring System Application

The Smart Monitoring System is a desktop application. The application should be able to visually display the sensor motes data which they are sending from their observed environment. The data representative should be made both graphically and on the console based output. The system should also be capable of dynamically updating its GUI in order to accommodate for the new nodes joining the network and for deleting the node which are on more active. In the GUI the application should display the motes in the form of a network with connected lines showing the current available modes and the routes taken by packets to reach the sink.

This application will also generate alerts in the form of visual alerts, audio alerts and also sms alerts. In order to send sms alerts the application must communicate with and send messages through a USB GSM modem.

The user must be allowed to give nodes which appear in the graphical mode a custom name to assist him in identifying the devices. Moreover interface for customizing application related parameters should also be a part of Smart Monitoring System.

3.3.1.4 Alert Generation

The Smart Monitoring System must notify the user in the case of overheating as well as in the case of an intrusion. The user should be given the option of choosing one or all of the three kinds of alerts: Visual alerts, Audio alerts and sms alerts.

Visual alerts should be such that a user monitoring the system is immediately informed of an intrusion by means of some visual change e.g. a node that has over heated should be highlighted or its color changes. If the temperature value again falls below the critical value it should return to its original color.

Audio alerts can be in the form of a sound or alarm which is triggered on overheating or intrusion depending on the settings made by the user.

If the user has selected to receive sms alerts then he must be sent a message notifying him of the event that triggered the alert and the name of node on which alert is triggered.

3.3.2 Product Perspective

Smart Environment from product perspective include sensor motes (eZ430 RF2500), the mote software which provides improved multihop routing and a desktop application. eZ430 RF2500 sensors have a RTOS implemented over them. This is responsible for getting the temperature reading from temperature sensor, routing it to destination and providing medium access. However, modification have the implemented routing protocol provides one to two hop network support. Therefore, from product perspective Smart Environment should support a ‘n’ hop network. In this regard, a new routing strategy is to be implemented. This software is similar to firewall and is to be implemented in low level C language.

This will control all the functionality of the target boards or motes which includes the temperature measuring, intrusion detection and transmission of information in multi hop fashion.

The second software is developed for the base station which can be a laptop/ desktop computer with which the sink node is connected. The software will receive the information through sink via COM port and will display the information in human understandable and readable format. The information will be displayed on the Visual Display Unit (VDU).

3.3.3 Network Perspective

The Smart Environment uses EZ430 RF2500 motes which form the network infrastructure. Every mote communicates its sensed data to the closest mote based on the gradient protocol using the wireless medium.

The farthest mote sends the data to the closest mote in the network based on the gradient protocol as proposed by IETF ROLL [1][2] and the transfer of the data continues until the data is reached to the sink node. Figure 3.1 shows how motes communicate with each other in the network.

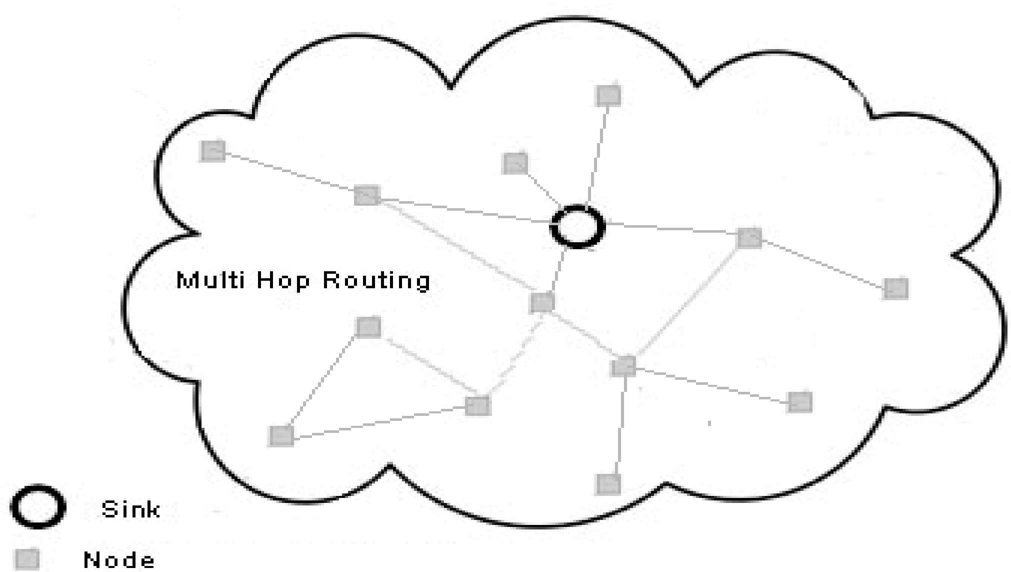

Figure 3 – 1 Smart Environment Network

3.4 Assumption and Dependencies

In the process of development of smart environment few assumptions have been made. In this section these assumptions have been explained.

3.4.1 Basic Assumptions

The first and most important assumption is that the system targets closed environments such as small buildings and homes and that there will be a single sink. However the Smart Environment project can also be deployed in large buildings but this would require a large number of motes and depending on the situation may also require multiple sink nodes. Also these sink nodes must communicate with each other to establish an effective network.

3.4.2 Operating Conditions

The Smart Environment hardware i.e. MSP430 rf2500 sensor motes are powered by two AAA batteries. Based on the operation of the motes these cells will have an average life of 4-6 months. The batteries must be replaced after that to ensure proper functioning.

The radio range of motes is severally effected by obstacles such as walls and objects within buildings and homes. Based on the tests which were conducted indoor radio range of motes across rooms is approx 20-30 feet while in corridors where line of sight communication might be possible radio range is 50-60 feet.

The Smart Environment project is intended for small closed environments such as houses and small buildings. However it can be used outdoors as well, with possibly a little deterioration in communication.

The targeted operating system for base station application i.e. SMS (Smart Monitoring System) is Microsoft Windows XP, Vista and Windows 7. Windows was selected as the first choice for the operating system because it is the most common OS and easy to use.

3.5 System Features

This section gives details of the system features as documented in the functional requirements. All the functional requirements have been met and transformed into system features. Common system features includes the following:

3.5.1 Temperature Measurement

Each node measures the environment temperature continuously while this information is periodically transmitted over the network and upon reaching the sink node, this temperature value is used by the Smart Monitoring System, the system makes a decision based on user settings that whether an alert should be generated or not.

3.5.2 Intrusion Detection

PIR (Passive Infrared) sensor [13], [17] is used for intrusion detection. However, the original eZ430 RF2500 devices are not equipped with PIR sensor. Therefore, PIR sensors are attached on the serial ports of the devices.

The detection range of the PIR sensor is 5-7 meter. Whenever an object moves within this range a trigger is reported by the sensor and it is delivered to the sink.

3.5.3 Smart Monitoring System (Desktop Application)

All nodes can be monitored using the Smart Monitoring System. It shows all nodes that are connected and displays them graphically. The user is given the option to view the data in a console as well, which shows the received data as plain text.

The Smart Monitoring System also allows the user to view the path the data has taken to reach the sink and which node has sent this data. This path is continuously updated and displayed in the form of connected nodes in the GUI window.

In both the graphical and console mode the user can view the temperature, whether an intrusion has been detected at a node, the voltage of respective node batteries and the route adapted by the data to reach the sink.

The Smart Monitoring System can alert the user in the form of visual alerts which is the default alert method, audio alerts and sms alerts. The user is given the liberty to customize these alert settings i.e. when to be notified. The user can set the maximum temperature value on which he should be informed. Also, he can select the type of alert in addition to visual alerts i.e. audio and sms. Along with that he can also specify whether he only wants to receive overheating alerts or only intrusion alerts or both.

3.5.4 A Multi Hop Routing Protocol

This is the backbone of the Smart Environment project. It is responsible for routing of data from the sender to the sink node. Each time a node transmits data it will use the developed protocol to create and send the data packet. The protocol defines the basis for selecting the next node in the hop sequence. Design and implementation details of the protocol are included in chapter 4 and chapter 5.

3.6 Non-Functional Requirements

Requirements that are required by the Smart Environment project to ensure smooth functioning and to achieve best performance are explained in this section. These are requirements that are not tangible but necessary for the proper operation of Smart Environment.

3.6.1 Performance Requirement

Smart Environment should be efficient with respect to response time, performance and operation. As the product deals with safety and security issues, which are both critical in nature, it is important for the system to be efficient and accurate. The minimalistic approach should be adopted to preserve limited amount of memory. Therefore, the size of the program must not exceed the memory limit of the devices. Since the network topology can change with the addition and deletion of sensors the system should respond to any change within appropriate amount of time.

3.7 Software Quality Attributes

Smart Environment project has to follow some important quality requirements that should cause positive impact on the quality. The quality requirements are discussed in this section.

3.7.1 Runtime System Qualities

The Smart Environment is a product which performs its functionalities in real time. So on runtime some function are adopted in order to fulfill the user required functionalities. The runtime quality attributes are as follow.

3.7.1.1 Functionality

Smart Environment must have to perform the functionalities like measuring the temperature, detecting the intrusion no matter wherever the devices are placed within the network. Moreover, displaying the changes on the VDU of the computer to which the sink mote is connected and the generation of alerts in case of abnormal behavior in the environment is also implemented.

3.7.1.2 Availability

The availability of the Smart Environment is very important because in case of overheating or fire and intrusion all this have to be informed to the user by generating alerts which can only be possible if the system is available.

3.7.1.3 Usability

The Smart Environment has to be user friendly and easy to install for the user. The user should understand right away how it will work after the system has been installed. The help menu option is available on the desktop application which guides user to all the details of the SMS.

3.7.2 Non-Runtime System Qualities

Non-Runtime qualities of Smart Environments are those requirements which can benefits other developers to enhance the code and improve any functionality of the system according to the environment.

3.7.2.1 Modifiability

Smart Environment should also have the ability to accommodate the changes and modifications. The product has the ability to incorporate both the hardware and software functionality if anyone wants to modify it. New sensors like pressure or presence of any toxic gas can be attached with the sensor device. Likewise, some changes can be incorporated in the software.

3.7.2.2 Reusability

The modules to be developed in Smart Environment should be reusable e.g. a module to access COM port data is developed in the project, the same can be used in some other application. Therefore, the implementation and working of the Smart Environment should be understandable so that it can be reused.

3.7.2.3 Integrate-ability

The Smart Environment has separately developed components which are required to collectively perform all the required functionalities at its best. These modules are integrated in such a way that all the requirements are fulfilled.

3.7.2.4 Testability

Smart Environment must be able tested to remove all the faults and bugs from the final product. Different testing techniques can be applied for example integration testing and beta testing. These testing techniques make the system to work according to the requirements specified.

3.8 Other Requirements

The Smart Environment system is should be developed with available devices. Also the cost of modification required within devices (like PIR sensor) should be low. It should be well documented, understandable, fault free at the time of delivery. It must allow the interfacing of additional sensors if required in the future.

3.9 Summary

In this chapter the requirements of the smart environment project are described. These requirements include the application, functional and non functional requirements and also the important features of the system. These requirements will serve as a basis for checking back if we have completed all that was decided at the start and also for testing purposes.

4 Chapter

Software Design

4.1 Introduction

The design and architecture of the system which is to be developed is very important because whole system qualities are based upon the design. The good Software design includes all requirements given by the user or client and managed considering all non functional requirements. Hence it is very important to have a good design for the development of the system. The design and architecture specification of the Smart Environment are explained in this chapter.

4.2 System Overview

Smart Environment project is based on the concept of sensing, processing and constructing. In other words, sensors like temperature/ motion sensors are used to gather information while these sensors are attached to MSP430 microcontrollers which process the data and send it to the base station. The base station then triggers an alarm or SMS in case of a critical activity which is the action portion of the system.

In more detailed overview, the sensors will sense the environment for temperature and intrusion. This information is used to build a packet that will be transmitted over the network. The packet will be transmitted using the IETF ROLL [1] [2] to the node closest to the transmitting node. The packet is forwarded until it is received by the sink node.

Once received by the sink the data is read by the Smart Monitoring System and processed to make further decisions. The data is also displayed to the user. Details of the Smart Monitoring System can be found in previous sections. Based on the values the Smart Monitoring System will generate alerts to notify the user using VDU, alarms and SMS.

4.3 Assumption and Dependencies

The basic assumption for Smart Environment is that the system is for homes or one floor of a building. It should not be placed in open environments. The hardware will not be exposed to direct sun light or direct wind from a heater or air conditioner. The hardware will be placed in places where intrusion is expected within 5-7 meters of the placement. The different nodes will not be placed more than 20 feet apart.

4.4 System Architecture

This section provides a detailed and comprehensive architectural overview of the system, using different architectural views to depict different aspects of the Smart Environment project. The significant architectural decisions which were made are also described herein.

4.4.1 Architectural Constraints and Goals

There are two main constraints in relation to the hardware being used and thus are also the constraints that were faced in the development of the multi hop routing protocol. These are memory and power constraints. The first i.e. the memory limitation, that comes with the EZ430 RF2500 i.e. there is a only 16KB of memory. The second also imposed by the hardware is that it is ultra low powered and so any hardware and sensors that are added must be able to function on a power supply provided by two AAA batteries. External power supply could have been used but that would add to the size of the node and so this option was discarded.

The goal is to design and implement a self configurable multi hop routing protocol for a wireless sensor network using the above mentioned hardware, within the above mentioned constraints. So that these low cost devices can be used in Smart Environment for data collection and routing purposes. Software application that reads and display the information graphically for the purpose of monitoring and generating alerts.

4.4.2 Serial Port Communication

The communication between the target computer and the sensor network nodes is via sink node. The sink is the final destination for any packet that is transmitted. The sink is connected to the target computer using USB interface. The Smart Monitoring System reads this data via a serial COM port.

4.4.3 Sensor Network Architecture

This section describes the architecture used for the wireless sensor network. This is shown below in figure 4-1.

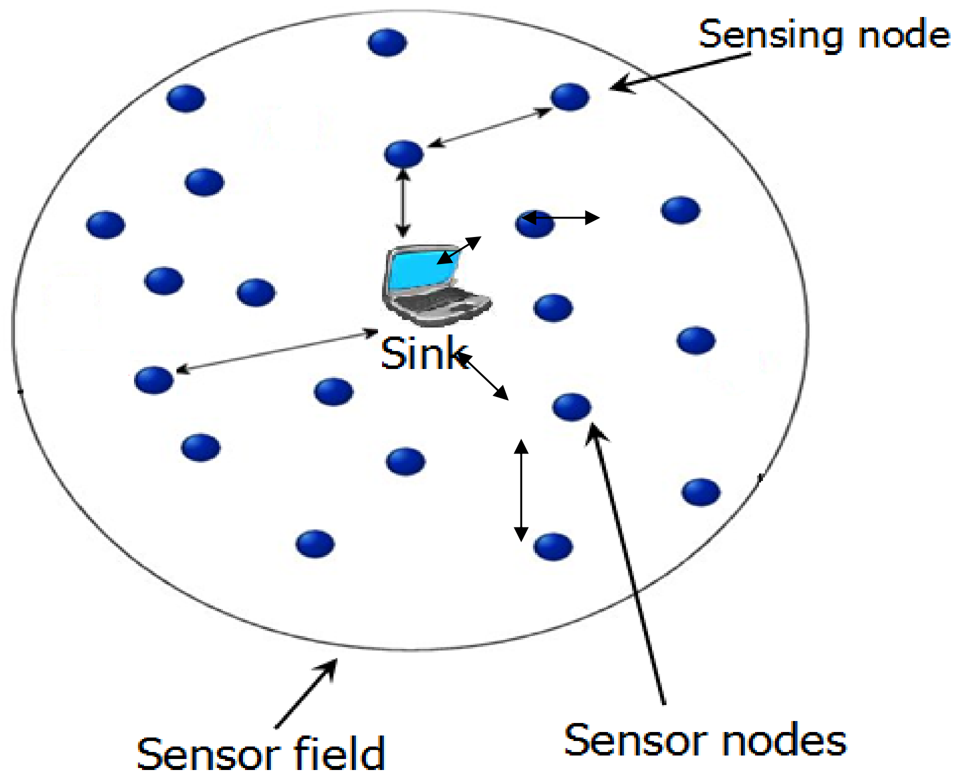

Figure 4 – 1 Communication System of Smart Environment

4.4.3.1 Hardware

This represents the EZ430 RF2500 sensor board which will host the temperature and PIR sensors. The sensors will sense from the environment, the sensed data will then be inserted into a data packet, the format of which is defined by the gradient protocol that has been developed. This packet is then transmitted by the transceiver. Once the packet has been transmitted it will be received by the transceiver of other nodes.

4.4.3.2 Multi Hop Routing Protocol

This protocol will take the sensed information and transmit it via the transceiver. This process will take place on all the nodes until the data reaches the sink. The architecture for the sink is given and explained in next section. The protocol design can be found in the System Design section.

4.4.4 Node Architecture

The following figure 4-2 is the architecture for a single node. Each node consists of hardware, and software.

The main hardware components of each node are the transceiver which accepts and transmits the data, the Micro Controller which carries out the processing on the data, and the sensors, which sense data from the environment.

The sensed data is used to construct a data packet which will be transmitted and finally received at the sink. We are responsible for the portion shaded in the below architectural figure 4-2 i.e. the routing protocol.

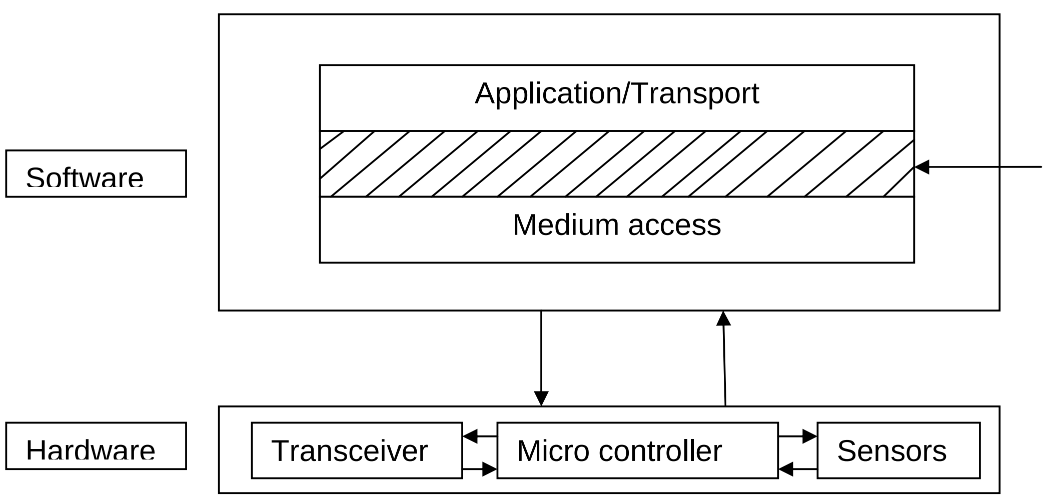

Figure 4 – 2 Node Architecture

4.4.5 Sink Architecture

This represents the architecture of the sink. The functionality of the sink varies slightly from that of other nodes. It must accept the data and instead of forwarding to the next node, it must be sent to the USB port, from where it will be read by the desktop application.

The data is received by the sink is transferred to OS. This is in turn taken by the .NET API. Finally it is read by the desktop application and the necessary processing is done. This is shown in the following architecture figure 4-3.

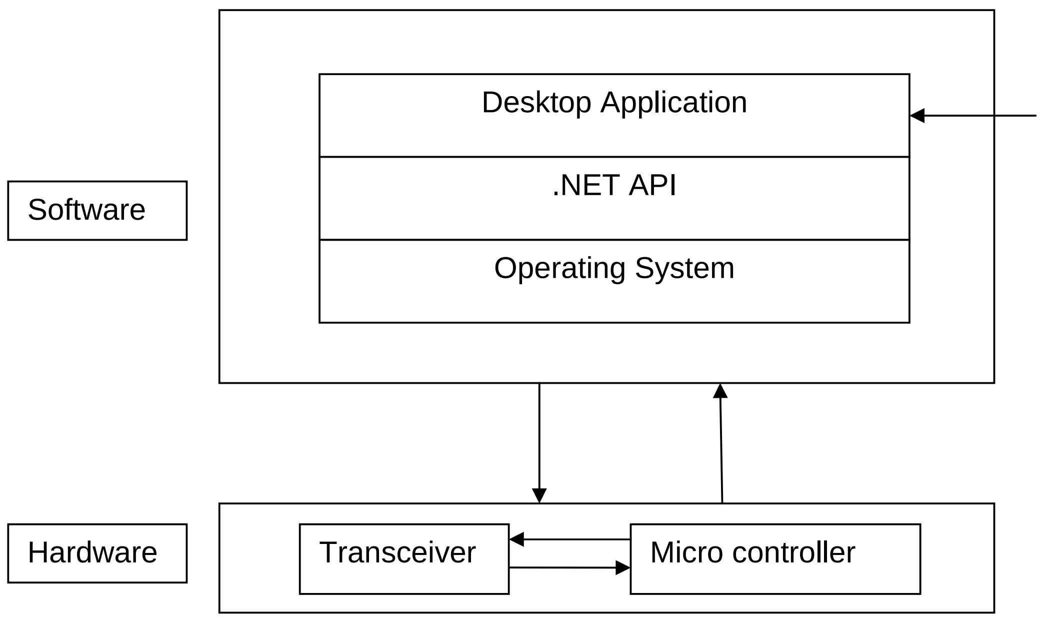

Figure 4 – 3 Sink Architecture

4.5 System Design

The design of the smart environment consists of two separate modules. The first is the routing protocol and second is the Smart Monitoring System. These have been designed separately but in a fashion which will allow easy integration and to achieve the complete functionality of the Smart Environment.

4.5.1 Routing Protocol

The protocol provided with the eZ430 RF2500 devices is SimpliciTi [10], [11]. The SimpliciTi protocol provides communication for star topology networks i.e. all motes communicate directly with the sink. This limits the coverage range of the network which in this case would be of 20 feet radius. To overcome this limitation a multi hop routing protocol has been implemented. This protocol is the gradient routing protocol which is proposed by the International Electrical and Telecommunication Forum (IETF) and is named as ROLL (Routing Over Low power and Lossy networks) [1], [2]. In the remaining of this section, we present a comprehensive overview of the design of the implemented gradient routing protocol.

4.5.1.1 What is gradient routing?

The gradient routing protocol is a stateless routing protocol. It assigns a height to each node. This height depends on the relative distance of that node from the sink. Height increases as the distance of the node increases from the sink. A cost function is used to calculate the distance. The cost function can be based on energy consumption, signal strength, hop count or any combination of these. The node with the largest gradient i.e. with lowest height is selected as the destination [3].

In figure 4-4 the node labeled R,3 (read as Node R with height 3) wants to send message, it sends it to the neighbor with least height, in this case Q. Similarly Q relays the message to N and N to A. A is the sink node. It is important to note that whatever path a packet takes the final destination will always be the sink.

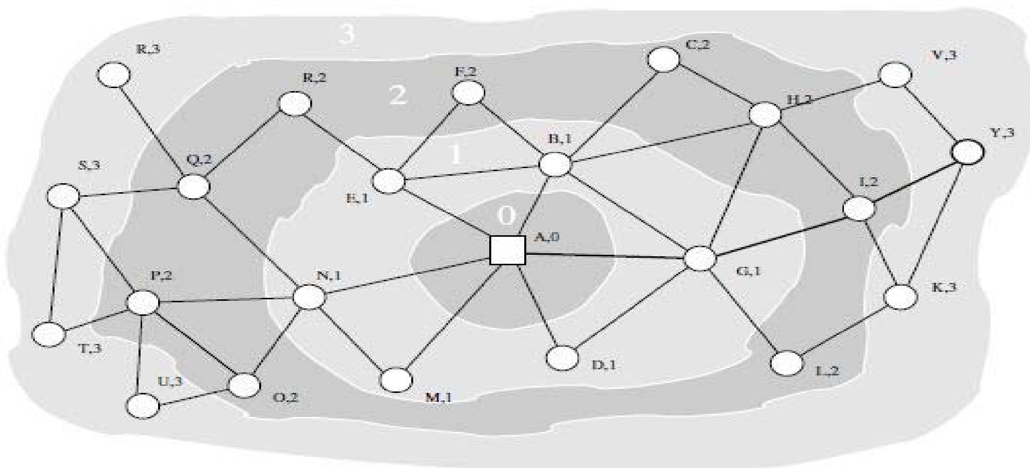

Figure 4 – 4 illustrating gradient routing [1], [2]

Since the available sensor nodes are resource constrained in nature and the network does not require strict QoS requirements, therefore we have used simply hop count as the basic routing metric or gradient.

4.5.1.2 Operation of Gradient based Routing

The proceeding sections include an explanation of the implementation the gradient routing protocol that is developed.

4.5.1.2.1 Height Calculation

A node sets its height to a non scalar value as soon as it is turned on except the sink which sets it height to 0. When a node sends a message, it learns its neighbors and updates its height by the minimum height of its neighbor's, incremented by one. The sink always keeps its height as 0 [4], [5].

4.5.1.2.2 Message forwarding

A node sends the message to the neighbor with the smallest height. The advantage is that a node is not limited to the single routing parent and sends the message to the best available parent thus increasing network reliability.

4.5.1.2.3 Neighbor discovery

A node creates a neighbor table each time a message is sent. When a node wants to send a message, it broadcasts a request and starts listening. As acknowledgements are received, the node creates a neighbor table. After the packet has been sent, the table is discarded [3], [4].

4.5.1.2.4 Packet format and parameters

The figure 4-5 shows the packet format. Header will be same for all the frames.

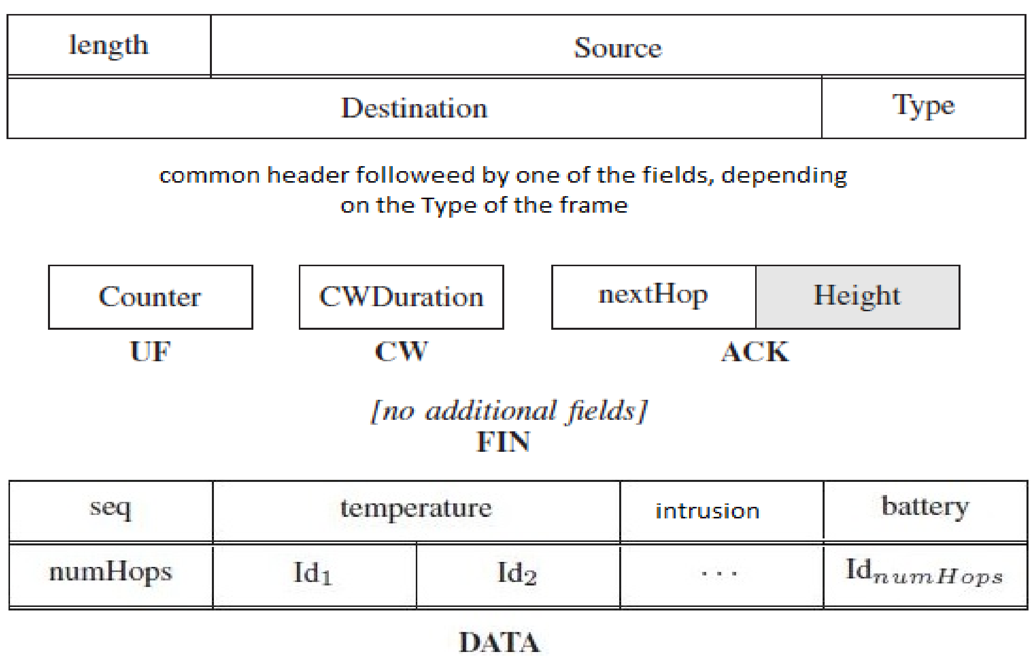

Figure 4 – 5 formats of the five frames types [6]

Type in the common header indicates the type of frame. The DATA frame contains a sequence number (seq) incremented by the sender at each new DATA frame. In the DATA packet temperature will hold the sensed value of temperature. The value of the intrusion will indicate whether there has been an intrusion or not. Battery hold the battery voltage of the sending node.

4.5.2 Smart Monitoring System (SMS) Design

This section provides details overview of the design of the Smart Monitoring System, using UML diagrams such as Use case diagrams and Class diagrams. This will depict the different aspect of the Smart Monitoring System.

4.5.2.1 Use Case Diagram of the Smart Monitoring System

The use case diagram shown in figure 4-6 describes how the user and system will interact with each other. The diagram explains how a user can interact with the system and what types of functionality he/she can perform on the system. The use cases in the diagram show what the system can do and the relationship of different use cases with the user has been shown according to the requirements. A detailed description of the use cases is given below.

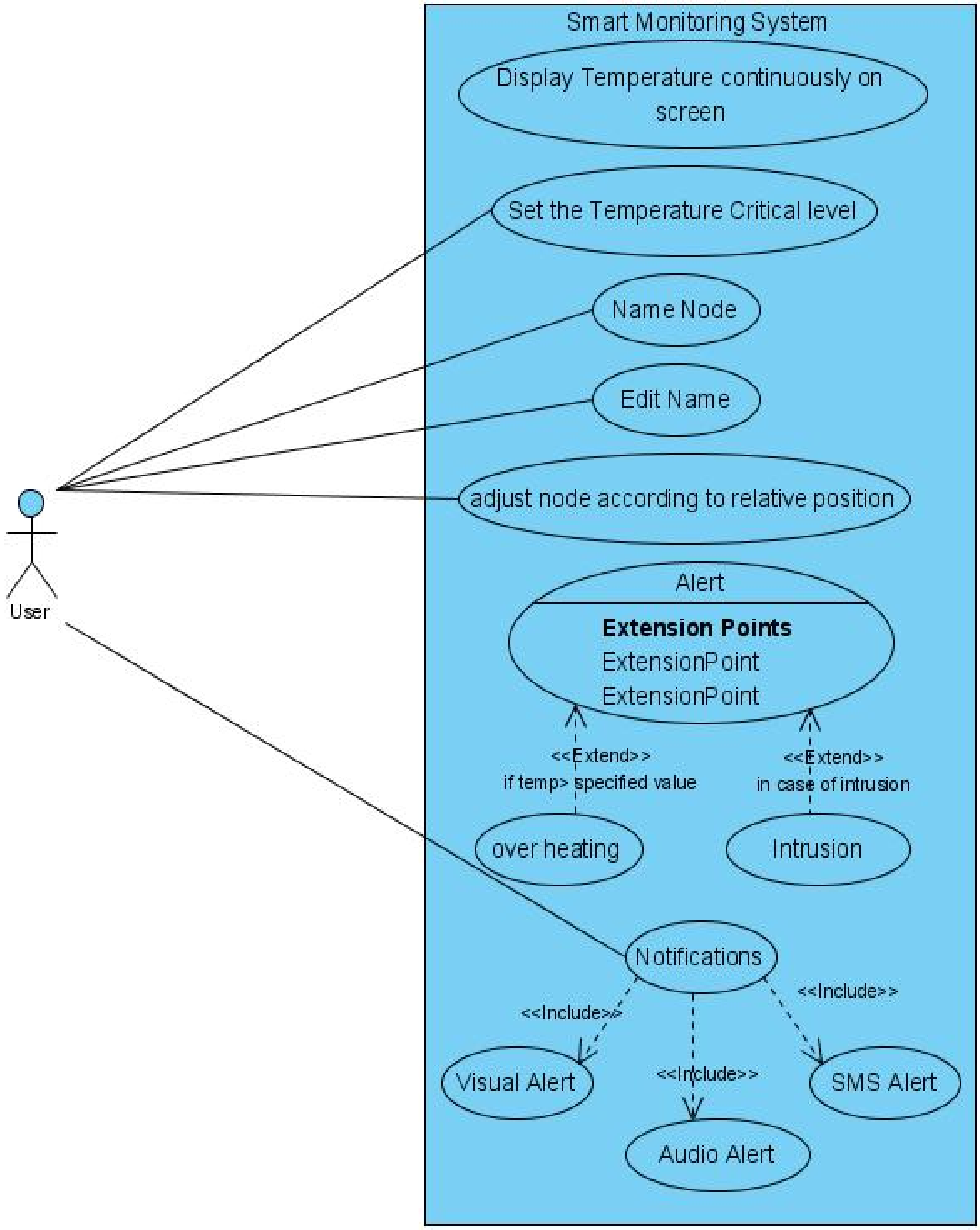

Figure 4 – 6 Use Case diagram

4.5.2.1.1 Display temperature continuously on screen

As data is received at the sink it is simultaneously read by the Smart Monitoring System via the COM port. The temperature is extracted from the received packet and is displayed. This allows for continuous monitoring of temperature changes. The figure 4-7 shows the system sequence diagram of use case.

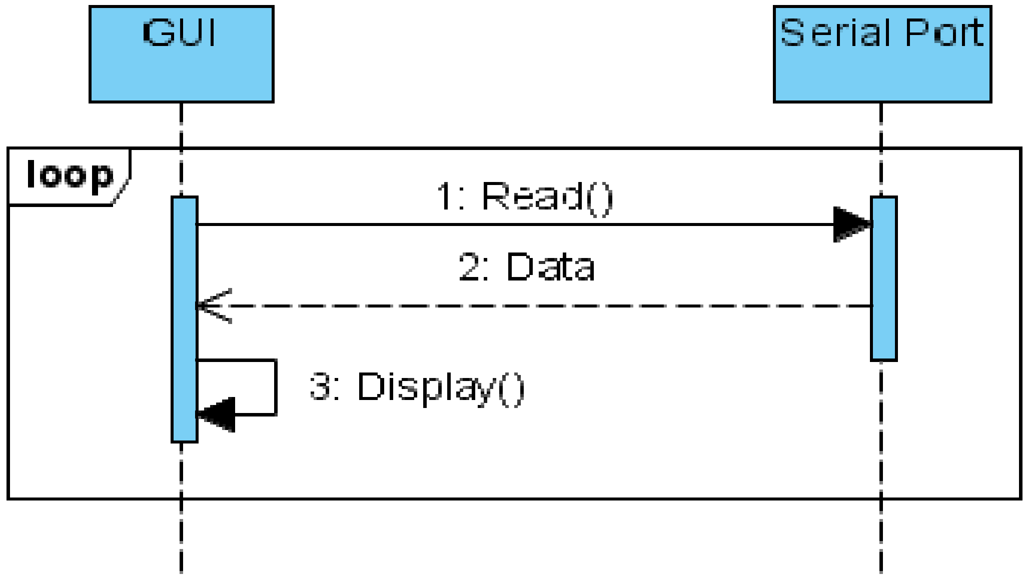

Figure 4 – 7 Display temp continuously on screen

4.5.2.1.2 Set temperature critical level

The user is provided with the option to set the temperature value which when exceeded will result in the Smart Monitoring System notifying the user in the form of an alert. The system sequence diagram in figure 4-8 provides the better understanding of the use case.

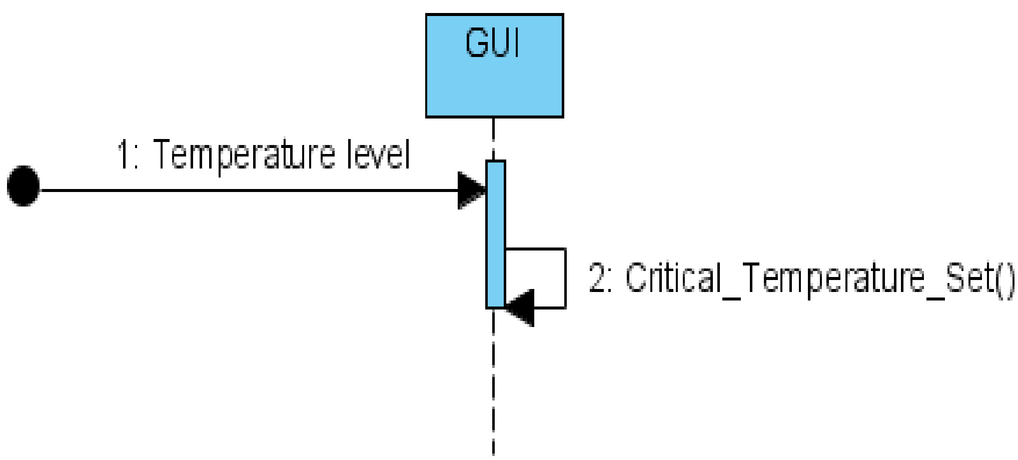

Figure 4 – 8 Set temperature critical level

4.5.2.1.3 Name Nodes

To assist the user of the Smart Monitoring System in monitoring the various nodes that will be placed at different locations in a building, the user is given the option to name the nodes. As a good practice this should be a sensible name, allowing the viewer to identify the location of the node. For example a node named as Node1 doesn't help in identifying which node this is or where it is placed. However, if the name is "Computer Lab 1", anyone who sees this can easily understand that this is a node placed in Computer lab 1. Adapting a good naming convention is a pre requisite for using this facility to its best. The use case can be better understood using the system sequence diagram shown in the figure 4-9.

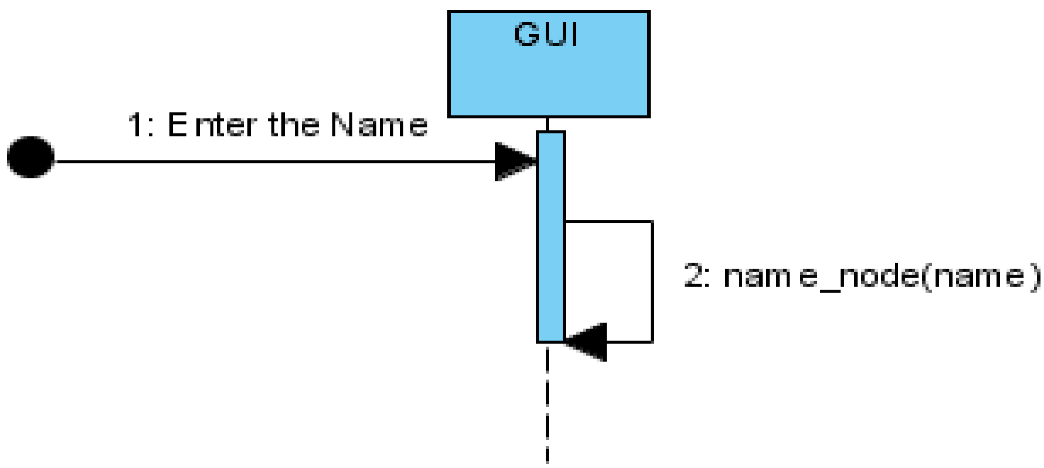

Figure 4 – 9 Name Nodes

4.5.2.1.4 Edit Name

The location where a node is placed can be changed by simply placing the node at the new location, as the network is self configuring. Based on this the user is provided with the rename option. All that needs to be done to rename a node is double click the name and it will ready for editing. So now the user can change the name of the node with respect to the new location with great ease. The figure 4 -10 is the system sequence diagram of the edit name.

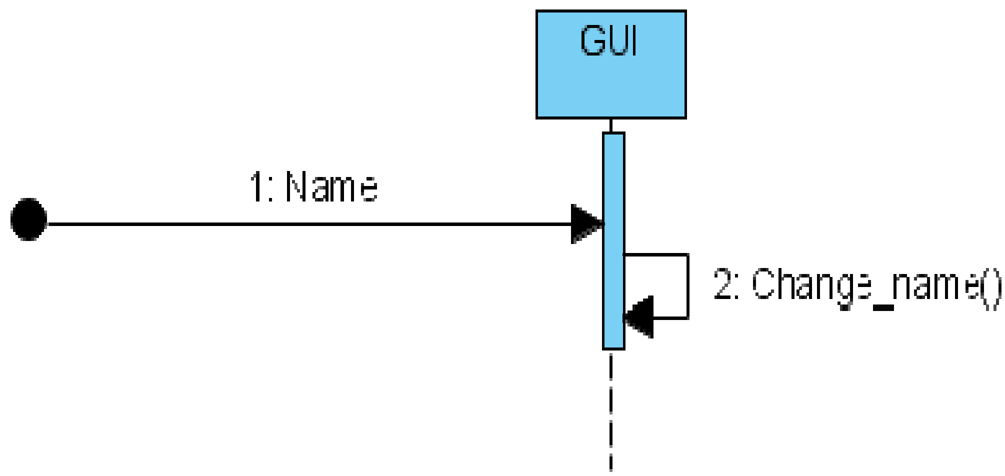

Figure 4 – 10 Edit Name

4.5.2.1.5 Adjust Node According to Relative Position

The user can adjust the position of the node relative to the position of the location where the node is placed on the screen of the monitor for their own convenience. The adjusting of the node according to relative position can be better understood by the help of system sequence diagram in the figure 4-11.

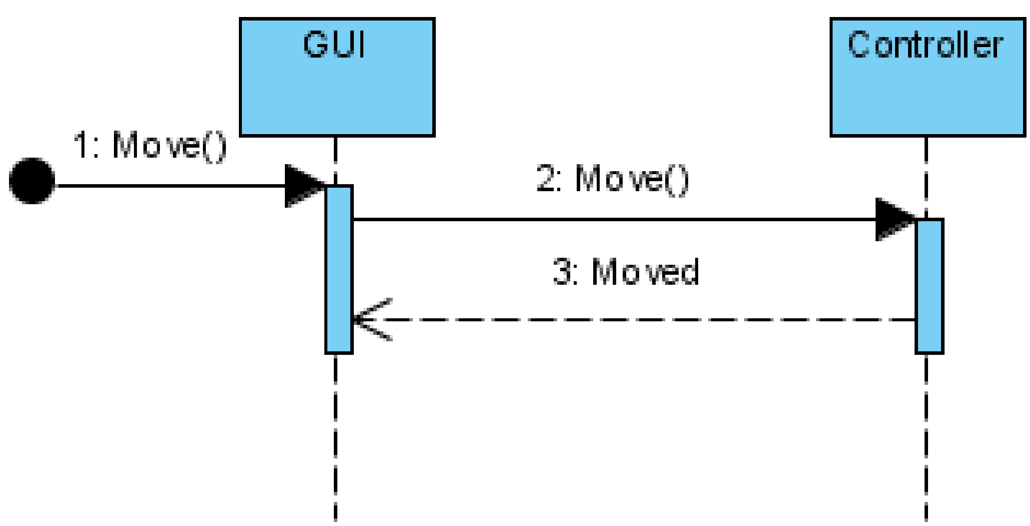

Figure 4 – 11 Adjust node according to relative position

4.5.2.1.6 Alerts

In the case that an intrusion is detected or the temperature value exceeds that set by the user, the user is notified in the form an alert. The alerts generation includes two conditions which are shown in the figure 4-12.

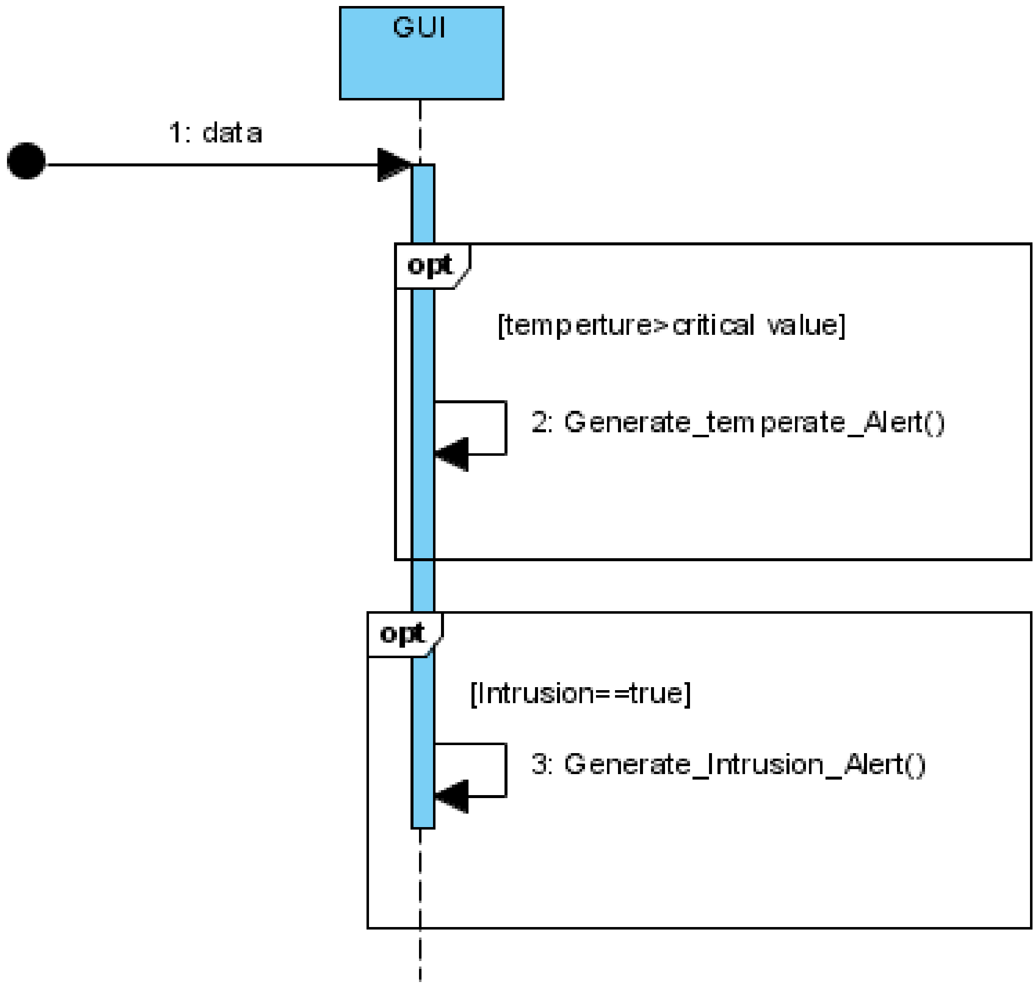

Figure 4 – 12 Alerts

4.5.2.1.7 Overheating

This use case is extended from Alerts. This use case explains that an overheating alert is generated when temperature of certain node increases from critical temperature level which is set by the user. Figure 4-12 gives better understanding of this extended use case.

4.5.2.1.8 Intrusion

This is the second extended use case of alert use case. This explains that intrusion alert will only be generated when an intrusion occurs. Figure 4 – 12 the system sequence diagram explains this extended use case.

4.5.2.1.9 Notification

The user will be notified as mentioned above. These notifications are in the form of alerts. Three kinds of alerts are provided by the Smart Monitoring System. These are explained below and the figure 4-13 shows the system sequence diagram.

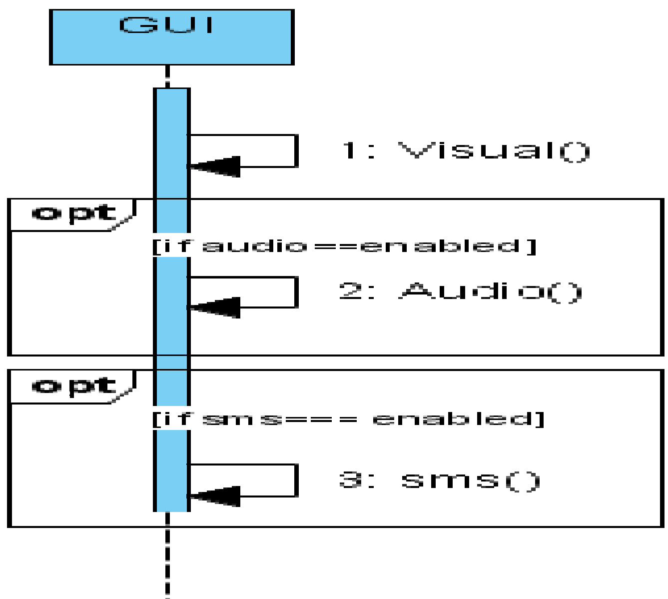

Figure 4 – 13 Notification

4.5.2.1.10 Visual Alert

Visual alerts are always on by default and can't be turned off by the user. These are in the form of color changes or change of appearance of the representation of the node.

4.5.2.1.11 Audio Alert

The second alert option is that of audio alerts. This used for sounding of an alarm to notify the user for the event.

4.5.2.1.12 SMS Alert

This is also an optional alert. If the user chooses to be notified via a text message then a sms will be sent to the number specified in the settings window. Simply clicking a check box and entering a number activates this option.

4.5.3 Class Diagram

The class diagram of Smart Environment has been shown in the figure 4-14 and the detail about the classes is discussed in this section.

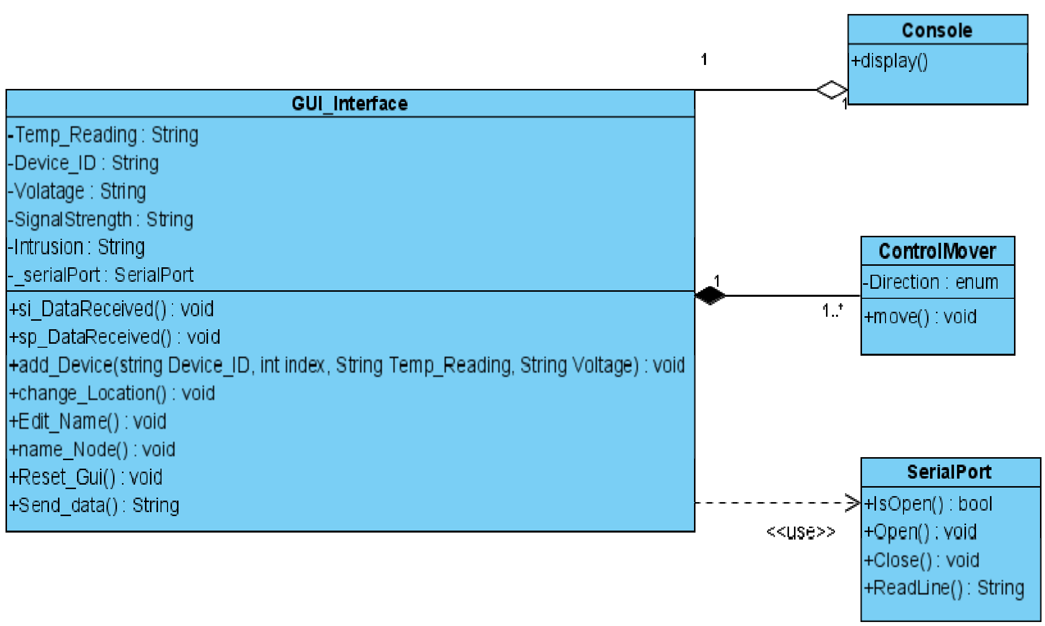

Figure 4 – 14 Class diagram Smart Monitoring System

4.5.3.1 GUI Interface

This class is the main class of the Smart Monitoring System which includes the maximum functionalities. The class controls and interacts with all other classes. This class is responsible for receiving data and adding a new node to the display (when a new node is activated) and few other features such as changing of location, naming the nodes and reset the GUI. The alerts generation is also the functionality of this class.

4.5.3.2 Console

The console class is the class which is dependent on the GUI interface class. There is the relationship of containership between the GUI interface and console. The functionality of this class is to display the information to the user in plain text.

4.5.3.3 Control Mover

The control mover class has the relationship of containership with the GUI interface. This class includes the functionality of moving the nodes on the screen to the relative position.

4.5.3.4 Serial Port

This class is responsible for serial port communication. It handles the reading of data from the serial port and making it available for use by Smart Monitoring System.

4.6 Summary

Design details and some of the design decisions along with the reasons for selection of these design decisions have been provided in this chapter. The design of the complete system has been divided into two parts i.e., the design of the routing protocol for the devices and the design of the desktop application (Smart Monitoring System) which allows monitoring, settings for alerts and generation of alerts.

5 Chapter

Implementation

5.1 Introduction

The previous chapter discussed the design; inner details of the class attributes and methods, Use case diagram etc. Based on that design this chapter will concentrate on the implementation details of the system.

We have categorized implementation details into temperature monitoring, Intrusion detection, routing protocol and Smart Monitoring System (SMS). Therefore, in the remaining of this section implementation details regarding these modules have been presented.

5.2 Temperature monitoring

Temperature measurements have been obtained using the on board temperature sensor on the devices. However, this sensor does not provide a temperature value as output to the system. Therefore, a function is used to calculate the temperature value [12] and then this value is transmitted to the sink node. The details of implementation of this function were taken from the data sheet of msp430 x2xx family. A code snippet to briefly explain this function is given below

```
degC = (((temp - 673) * 4230) / 1024);
 If (tempOffset! = 0xFFFF)
{
        degC += tempOffset; }
```

In the above code, tempOffset is a value set at the time of production and is used to adjust the temperature and then to convert the temperature value to centigrade.

5.3 Intrusion Detection

There are a number of sensors available which can be sued for the purpose of intrusion detection. We have used PIR sensors due to ease of availability and low price. Moreover, most of the motion detection sensors continuously emit radiation, such as light, radar, or ultrasonic waves and sense changes in the environment caused by a passing object. PIR sensors also sense changes in the environment, but they do not emit radiation. Instead, they detect changes in the temperature of objects and people that are within their range of surveillance [17].

The PIR sensors selected for this purpose are SB0061 and SB0081. These have been soldered onto the rf2500 motes. The PIR sensors are powered by soldering it to PIN 1(GND) and PIN2 (VCC+). The output pin of the PIR sensor is attached to the PIN3 of the mote. When an intrusion is detected by the PIR sensor, it sends a voltage signal at its output pin. When the PIN3 of the mote detects a high signal, it triggers an interrupt. This informs the system that an intrusion has been detected.

5.4 Routing Protocol

This is the back bone of the Smart Environment. This routing protocol (IETF ROLL) is responsible for sending sensed data to the sink node. The protocol has been implemented in C using IAR embedded workbench. The reason for choosing IAR was that it supports the msp430 family of devices (including rf2500). Also provided by IAR is an easy to use UART (Universal asynchronous receiver/transmitter) interface for the downloading and debugging of the C code on the motes. The reason for choosing low level C as the programming language is ez430-RF2500 motes have limited on board memory available i.e. 16KB.

Common Ad hoc routing protocols such as AODV [7], DSR [8] and DSDV [9] cannot be implemented on these devices due to their resource constrained nature. As a result, IETF ROLL has been selected for implementation. This IETF standard is currently being developed for resource constrained sensor nodes. However, it has not been standardized yet and no free implementation of this protocol is currently available by the IETF.

All nodes sense information from the environment and forward it to the next node. This forwarding process is repeated by each node until the information reaches the sink. The implemented routing protocol defines the steps that each node must take to achieve this.

The node periodically measures temperature values and continuously checks for an intrusion. When an intrusion is detected or a temperature value measured, a packet is locally generated on the node and then forwarded. In order to forward a message the node must start off by sending a preamble. The length of this preamble is set to the length of the check interval (CI) i.e. 104ms to make sure all the neighbors hear this preamble. In order to improve the efficiency of the radio in handling packets, the preamble is divided into a series of micro frames (UF). Each UF contains a counter which indicates the number of UF still to come. A total of 53 UF are transmitted. Each node that receives a UF will turn off its radio and switch to receive mode after receiving the last UF. Upon successful transmission of all UFs the sender transmits a CW (Contention Window) packet that will indicate the duration of a neighbor announcement window.

Once the neighbors receive the CW packet, each chooses a random back off time after which it sends an ACK message (within the neighbor announcement window). The node will sleep for the remaining time. The sender will listen for the complete duration of the neighbor announcement window and will populate the neighbor table as it receives the ACK messages. The neighbor table is initially empty.

In order to select the next hop (neighbor to which the data will be sent) the sender, after the completion of the neighbor announcement window will update its height value. This will be set to the minimum value received from the neighbors + 1. Along with this it will select this neighbor i.e. the one with the minimum height as the next neighbor. This information will be inserted into the header of the data packet to be transmitted. The destination node i.e. the one to whom the packet is sent to will accept the whole packet where as the others will simply switch to sleep after receiving and reading the header. The destination node, upon receipt of the whole packet will reply with a final acknowledgement i.e. FIN.

The devices by default perform preamble sampling. When a node wants to send a message, it starts by sending a preamble as long as the check interval (CI) to make sure that all nodes hear the preamble. For efficient handling the preamble is cut into a series of micro frames (UF), each containing a counter to indicate the number of micro frames still to come. Upon hearing a UF, a receiving node turns its radio off and sets a timer to switch into receiving mode after the last UF. At that moment, the sender indicates the duration of the neighbor announcement window to follow in a Contention Window (CW) packet.

Receivers choose a random back off (time) for sending an acknowledgment (ACK) message inside the neighbor announcement window and sleep the rest of the time, the sender listens for the complete announcement window and populates the initially empty neighbor table as it receives ACK messages.

After the neighbor announcement window, the sender updates its height by the minimum value of its neighbor, incremented by one, and selects its neighbor with smallest Height. It inserts this information into the DATA packet header which it transmits. The destination node receives the whole packet while the non destination neighbor switches to sleep after the header. The destination replies with a final acknowledgment FIN and all nodes resume preamble sampling [6].

The above sequence is shown in the Figure 5-1 for 3 nodes. The top most (A) is the one transmitting and the bottom most (C) is the one that ends up receiving the packet.

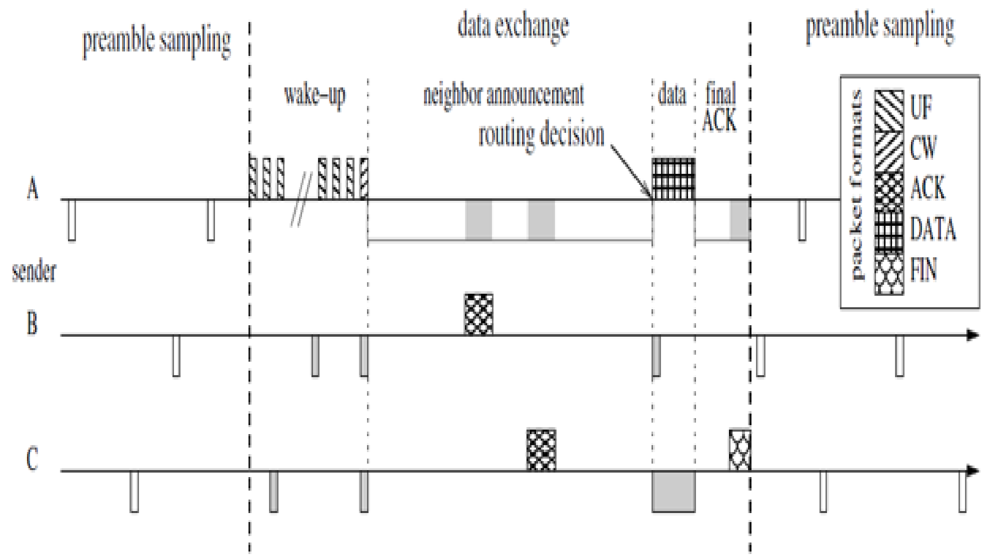

Figure 5 – 1 execution of protocol. The box above the axes means the radio is sending and the box below the axes means the radio receiving/ listening. No box means the radio is turns off [6].

5.5 Smart Monitoring System (SMS)

The communicating motes are not enough for the complete functioning of the Smart Environment. The user will be provided alerts based on the information provided by the motes, via Smart Monitoring System. This is an application developed in C# .NET. This will assist the user in monitoring the system as well as customizing alert settings. Data from all the motes is communicated to the sink node. The Smart Monitoring System continuously listens for data received at the sink. The Smart Monitoring System reads this information from the sink as it arrives via the COM port (The sink is connected to the computer via USB interface). Each time the message is received at the sink, a software trigger is generated which informs the SMS that data has arrived. In the next step SMS reads this data and processes it, displays it in a user friendly manner on the Visual Display Unit (VDU). Different implementation details of SMS components are explained in the remaining of this section.

5.5.1 Graphical User Interface

The Smart Monitoring System allows the user to view the received information as graphical objects or a console. Each time a new device is connected, it appears on the screen (depicted as a graphical object) and similarly when a mote is removed, it disappears from the screen. The route that is taken to arrive at the sink node is shown by lines connecting them on the SMS interface. The SMS application also, provides customized settings option to its user. This allows the user to set different thresholds for the generation of alerts. The alerts can be visual, audio or sent to the user as an SMS.

In order to make the Smart Monitoring System user friendly and allow for easy monitoring, the user is allowed to name the motes as they appear (each time a motes is connected) to identify where this motes is placed in the environment. e.g. if a mote is placed in the bedroom, the user might want to name it as name “Bed Room”. The user is also given the liberty to drag the objects (representing motes) to any place on the screen.

A screen shot of the application is shown in fig 5-2 which represents three nodes along with their names.

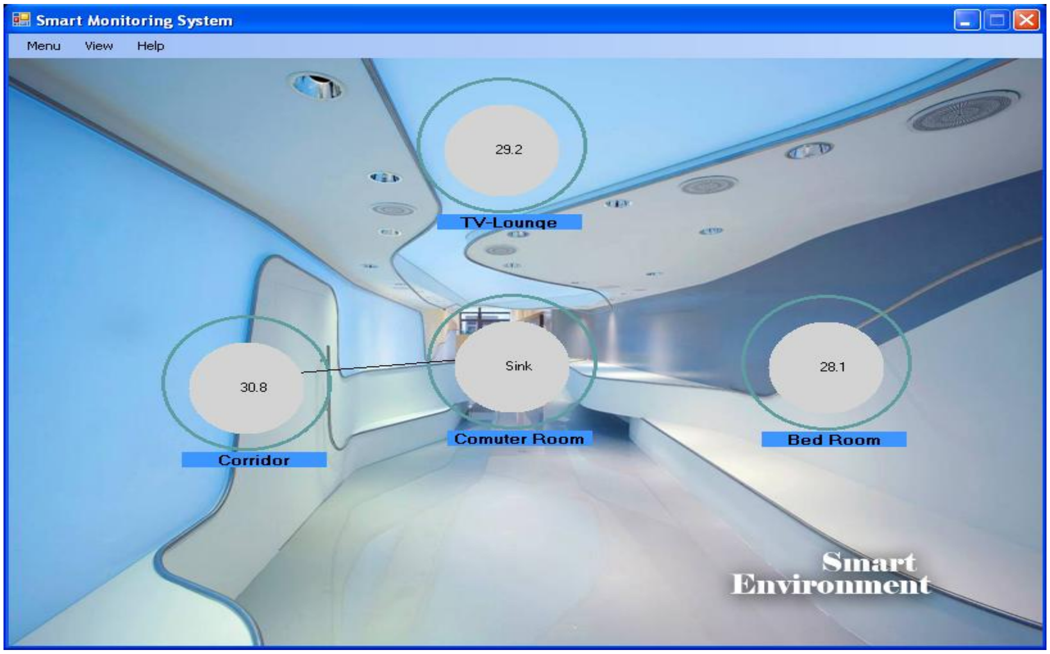

Figure 5 – 2 Smart Monitoring System

5.5.2 Console

The console mode displays the received information to the user in the form of text. Screen shot of the console output is shown in figure 5-3 which represents path taken by node and battery power.

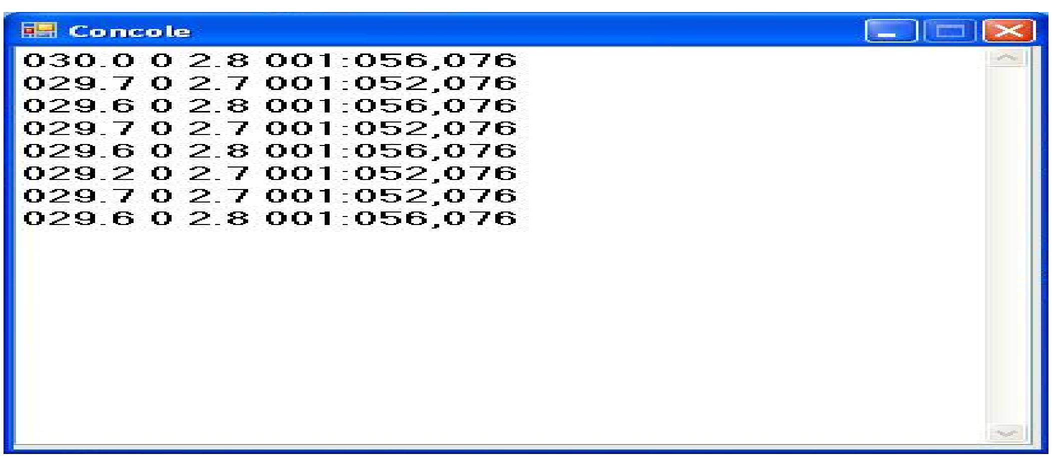

Figure 5 – 3 Console output of SMS

The text is to be read as: the first value is the temperature, second indicates whether an intrusion has been detected, 0 means no intrusion and 1 means an intrusion. Next is the battery voltage of the sending node. This is followed by the id of the sending node followed by the path taken by the packet to reach the sink.

5.5.3 Settings window

The settings window can be opened by going to the file menu and clicking on settings. This will open the setting window as shown in the figure 5-4

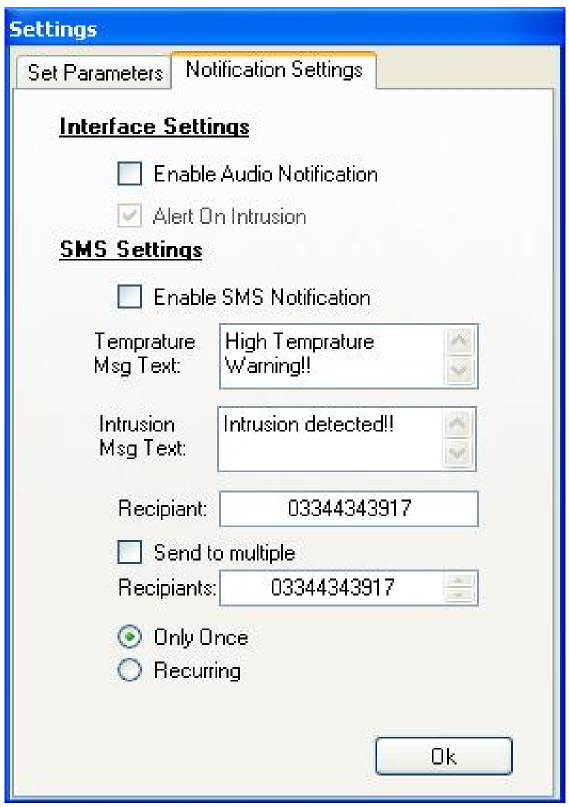

Figure 5 – 4 Settings

The user is provided with a number of options. Based on the parameters set by the user, the Smart Monitoring System will generate alerts. The user can set the value of the temperature, which when exceeded will result in a trigger which will generate an alert. Also the user can select whether he wants to receive alerts via SMS or audio (visual alerts are by default and can't be disabled). Also he can select when to generate an alert i.e. only upon intrusion, only upon over heating or both.

5.6 Summary

The implementation of final product has been explained in this chapter. Implementation of the gradient routing protocol and the Smart Monitoring System has been explained in detail.

6 Chapter

Testing

6.1 Introduction

Most important issue related to any project is to ensure that the quality of the product is high and it is efficient therefore testing of the software is conducted. In Software projects special importance is given to the testing and it is one phase of Software development life cycle, no software project is completed without testing. The Smart Environment project is no exception. Testing techniques which are used in testing of Smart Environment to obtain the high quality product are discussed in this chapter.

6.2 Testing Techniques and Levels

Testing of the software projects involve different levels of testing to make sure that the software which is being developed is error and fault free. Smart Environment has different modules which were developed separately depending up on the functionalities. Therefore testing of all the modules has to be done and testing while integrating all the modules. The different levels at which testing was done are discussed here.

6.2.1 Unit testing

Unit testing involves the testing of each module at the completion and sometimes during the development of the module. The testing of each module is carried out on the basis of the defined data sets. The Smart Environment was divided in to four different modules that are temperature measuring, Intrusion detection, Multi hop routing protocol and Smart Monitoring System (Software Application).

6.2.1.1 Temperature Module

The unit testing of the module is done by taking the original temperature of the location in which device was placed as the data set. There was a slight variation in the expected and actual output. The result of unit testing showed that the temperature measured was accurate up to +3 degree centigrade of the original temperature. The reason for variation in the temperature sensor reading is that the sensor is not attached to the ez430-Rf2500 device on its serial port as an external sensor. Instead it is built with the microcontroller therefore it value is generally 2-3 degree more than the actual value.

6.2.1.2 Intrusion Module

In this module of Smart Environment, testing was performed to ensure that the signal generated by the PIR sensors which is soldered on the eZ430-RF2500 done so only when something came within the range of surveillance. This was done by setting up the hardware, movement of objects at varying distance from at different angles to the PIR sensor and observing the value that indicated intrusion. After a rigorous testing process it was concluded that over 90-95% of the results accurately detected intrusion when the object was in the surveillance range. A very small set of results were false alarms due to the nature of the PIR sensors. The effective detection range of the PIR sensor is 4-5 meters.

Once turned on for the first time the PIR takes 1 to 3 minutes to familiarize itself with the environment. During this time there is a possibility of false alarms.

Testing has revealed that the PIR can detect an intrusion within 5 meters normally, 3-4 meters at an angle

6.2.1.3 Multi Hop Routing Protocol

The multi hop routing protocol is the most important module of the Smart Environment because it deals with the transfer of data from the different nodes to the sink node. The testing was carried out by placing the nodes at varying distances from each other and the sink. The contents of the packets received were observed to ensure that correct data reached the sink. The testing process revealed that communication deteriorated if the distance between nodes was above 20 feet. Also communication faltered when there were many obstructions between communicating nodes.

Certain delays are caused in the cases of node relocation, adding a new node to the network, removing a node from the network. Also a delay is caused during a multi hop transmission at intermediate nodes. All of the above mentioned delays were in the range of 3-4 seconds.

6.2.1.4 Smart Monitoring System

This is the software application of the system which displays the data coming from different devices to the sink node. It also process data to generate alerts depending upon the incoming data.

In order to test the application the network was set up and then it was tested for different events. The application was tested for fire, intrusion, changes in network configuration and for renaming of sensor nodes.

Moreover, message sending was tested and revealed that the messages are successfully delivered to the user whose number has been specified in the settings window. Audio alerts were tested to check if an alarm is sounded if the user has checked the audio alerts option. Also, visual alerts were tested in case of different events.

6.2.2 Integration Testing

Integration testing is the type of testing which is performed when different modules of the system are combined to form a whole system. This testing level ensures that the system is stable and performs as expected after integrating the individually tested modules. The modules as mentioned above and the precious chapters were integrated and tested. The results produced were satisfactory to the fact that the system correctly reported events and multi-hop routing was operational. All modules communicated with each other as expected. The nodes successfully measured values from the environment. These were correctly inserted into packets and transmitted by the gradient protocol. Upon receipt by the sink, the data was successfully read by the Smart Monitoring System and respective alerts were generated.

6.2.3 System Testing

System testing is the level of testing which comes when the whole system has been developed and integrated. The complete system was tested in different places with different conditions to verify that those conditions do not disrupt the performance of the system. The testing was carried out both indoors and outdoors. The only significant finding was that the coverage area of the network was more outside than inside the buildings.

6.3 Summary

Testing is one of the most important phases of software development life cycle because by testing one can ensure the quality and performance of the product as expected. Different testing approaches were applied on the Smart Environment to make system which lead to a more stable and correct version of the system.

7 Chapter

Result and Analysis

7.1 Introduction

Smart Environment is a real time application to provide safety and security. It measures the temperature and keeps the user informed about the changes in temperature and generates alerts. It also detects intrusion and alerts the user. The system provides all these services by using Texas Instruments devices eZ430-RF2500 which communicates wirelessly to a central device were the Smart Monitoring Application is deployed.

7.2 Result

Smart Environment deals with safety and security issues which makes this system sensitive and important. The basic concept of the project is to facilitate people with temperature measuring to provide safety against fire and intrusion detection to alert user about the intrusion in restricted or protected area. The main objective of the project was achieved which is discussed in this section.

The main objectives of project was to develop a self configuring multi hop routing protocol for wireless communication, temperature and intrusion detection and generation of audio, visual and SMS alerts.

Each of the modules has been tested and results gathered. These are documented below

7.2.1 Temperature measuring and Intrusion detection

Temperature measuring and intrusion detection is the first objective of Smart Environment which was very important to be achieved. Temperature measurement was achieved by using eZ430-RF2500 in built temperature sensor. Figure 7-1 below shows the temperature read by two different devices.

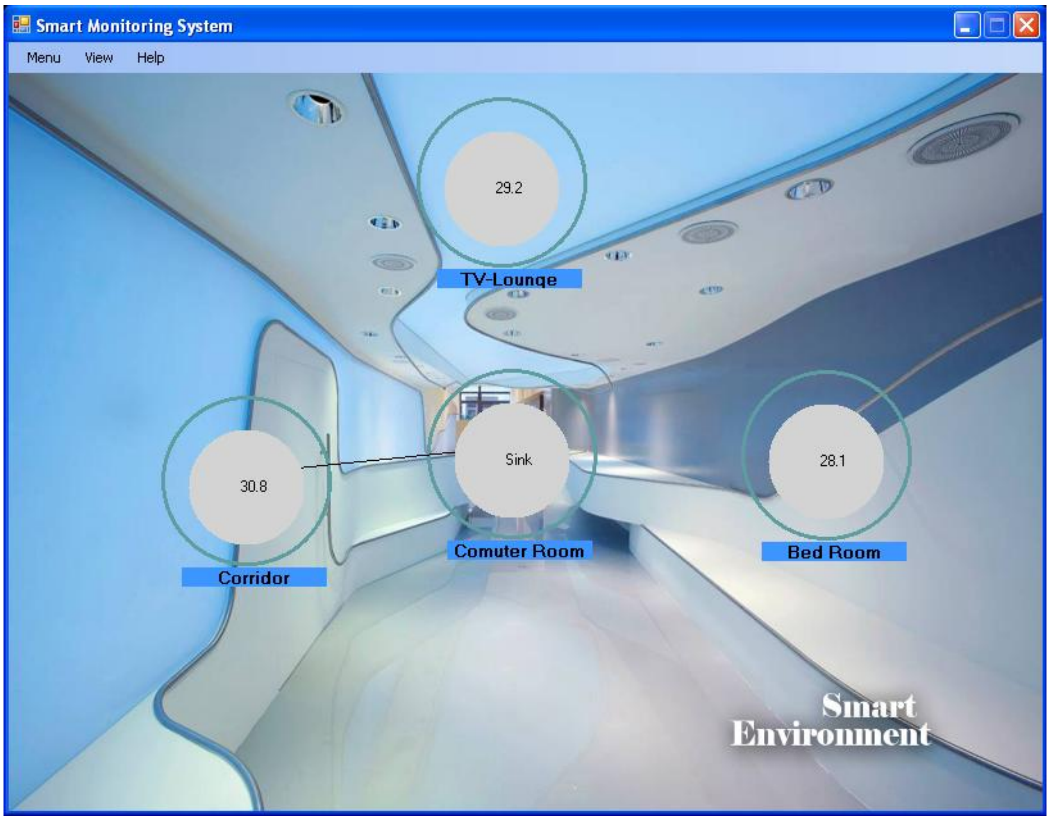

Figure 7 – 1 Temperature reading

The figure 7-1 is showing three devices displaying the temperature of the TV-lounge, Bedroom and corridor. The temperature that is being displayed varies to +3 degree centigrade.

The intrusion is detected by the help of PIR sensor soldered on the devices eZ430-RF2500. The PIR sensor detects the intrusion and generates the signals which alert the user on the SMS. Figure 7-2 gives the better understanding of the concept.

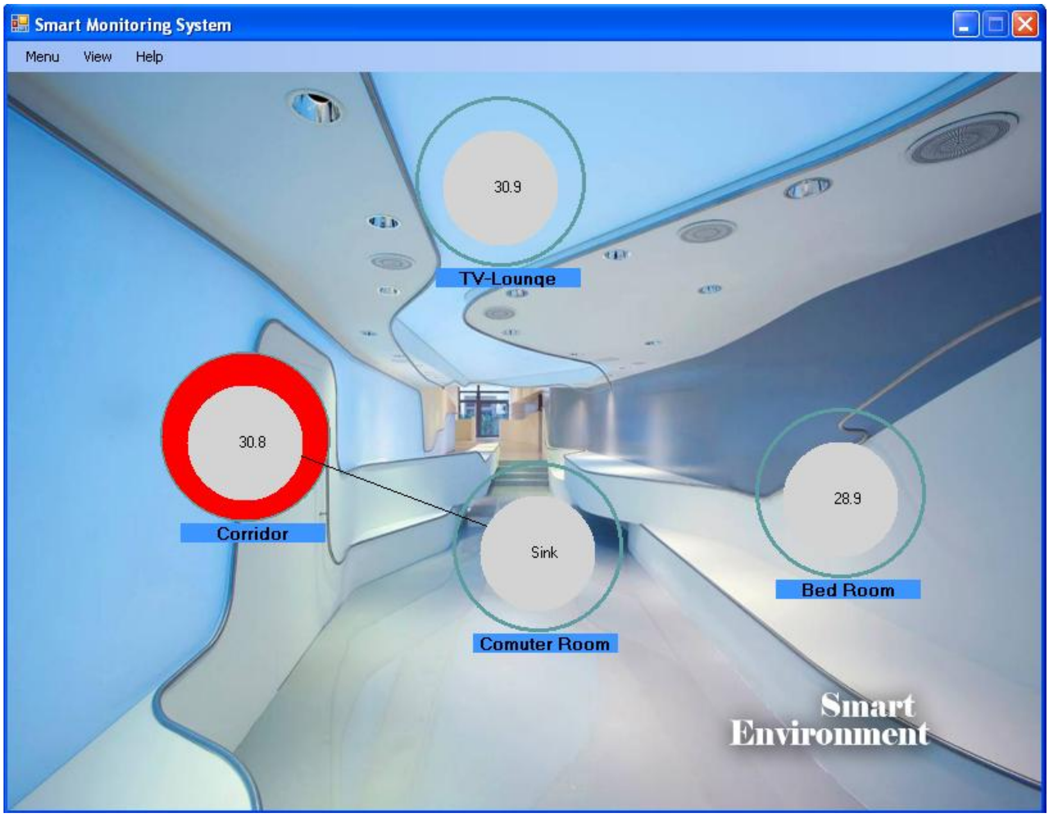

Figure 7 – 2 Intrusion detection

7.2.2 Multi Hop routing protocol

Multi hop routing is an important objective of the system because it involves the transfer of data from the end node to the sink node. The data is sent in packet and the protocol is based on the gradient protocol presented by IETF ROLL (Routing on Lo power and Lossy networks) [1], [2]. The result of multi hop can be observed by the help of figure 7-3 in which different devices are connected to each other via a line which is representing that is the path of data coming to the sink.

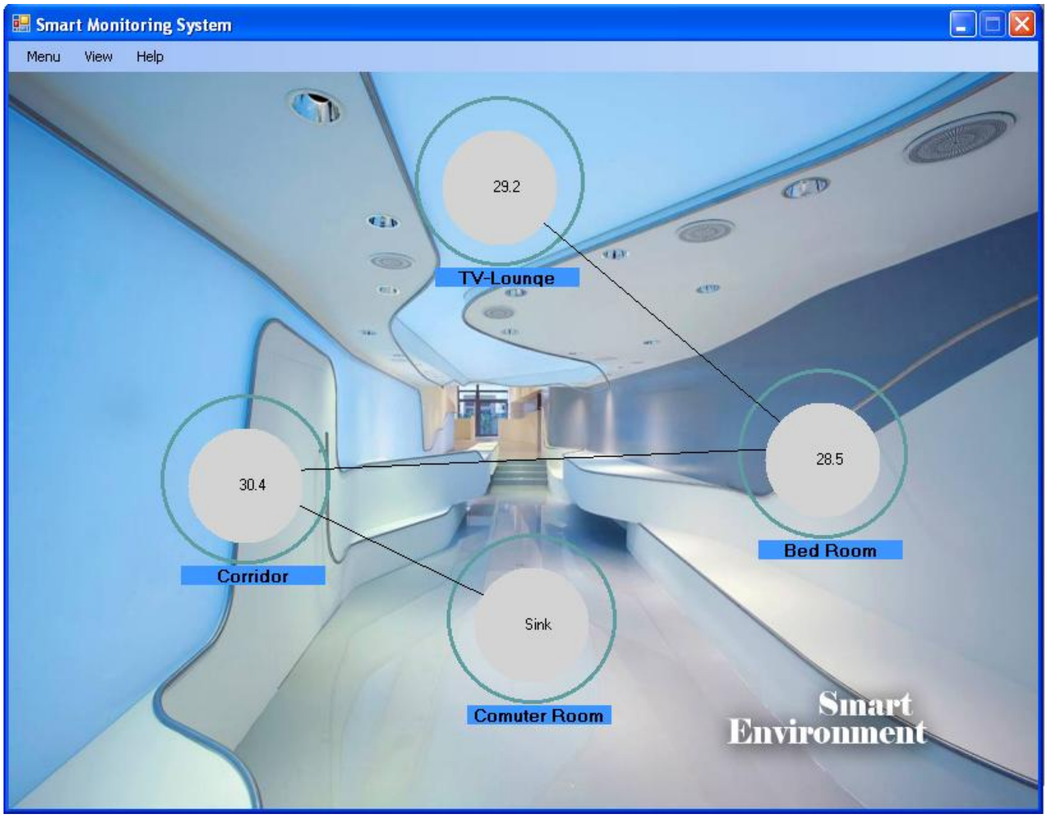

Figure 7 – 3 Multi Hop routing protocol

7.2.3 Alerts

The alerts are generated in case of fire, overheating and intrusion to inform the user about the occurring of undesired event. There are three types of alerts audio, visual and text message.

The audio alert is in form of an alarm using the host computers audio device and the visual alerts are shown in the figure 7-4 and figure 7-5 for better understanding.

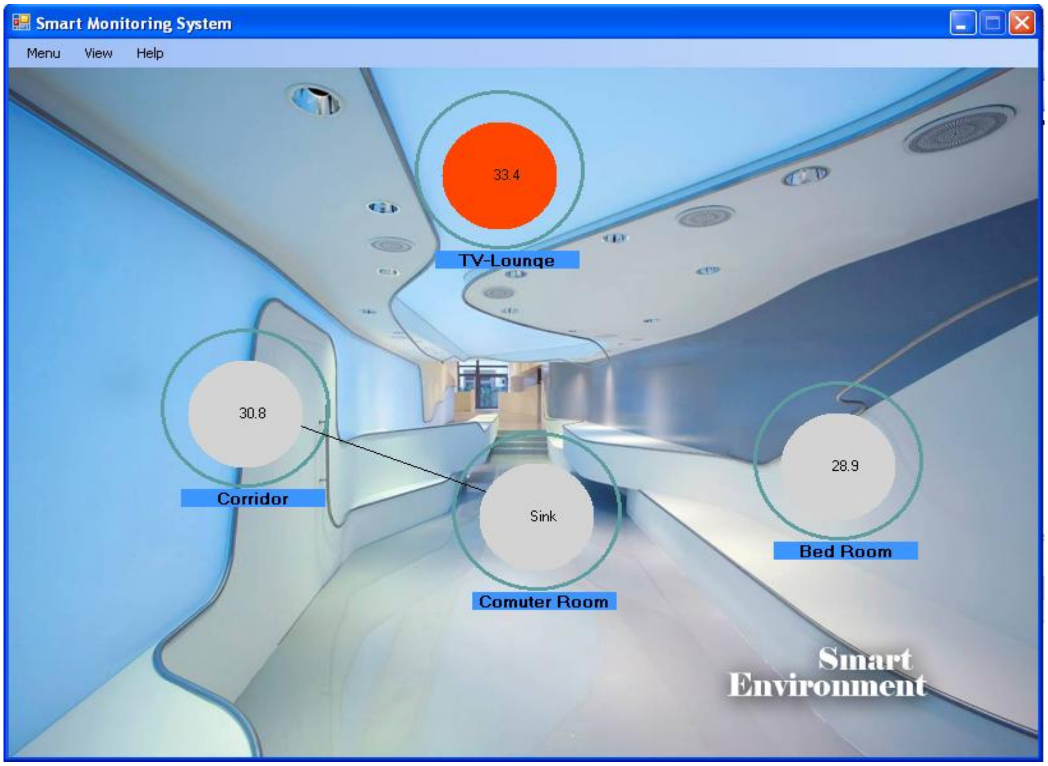

Figure 7 – 4 Temperature Alert

There is also option available for text alerts which can be enabled and text message will be sent whenever an undesirable event takes place. For sending text message, a USB GSM modem is used. These alerts are very accurate and timely sent to the user on his cell number.

Note: the Smart Monitoring System dispatches the message immediately. Network delays could delay message delivery to the user.

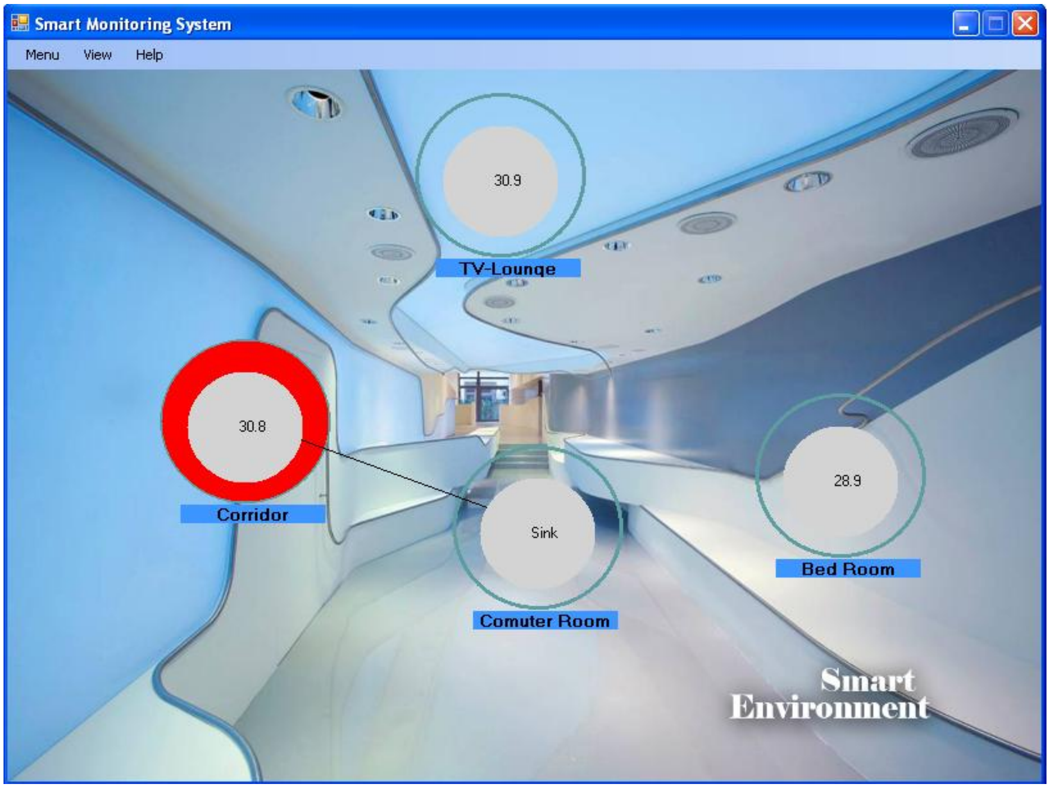

Figure 7 – 5 Intrusion Alert

7.3 Analysis

The analysis of the results of Smart Environment can be made by comparing it with those systems which are also providing similar functionalities. One of the major advantages of the Smart Environment is its ability of self configuration. There is no need for lengthy installations. Just turn on the devices and install the application.

Based on the results, analysis has been carried out to discover the bounds of the software. Also to detect and resolve conflicts if any with the requirements. Also this will help to establish certain things that must be avoided and others that are necessary while using the software and the underlying hardware.

The results of the analysis are summarized in the following table 7-1.

Table 7 – 1 Analysis

Sr.#	Parameters	Analysis & Results
1.	Outdoor VS Indoor radio range	**Outdoor:**30-35 feet **Indoor:**15-20 feet
2.	Temperature Measurement (Documented vs. Tested)	**Documented:** -40°C – 85°C **Tested:** 8°C – 60°C Variation of +/-3°C with actual temperature
3.	Intrusion Detection (Documented vs. Tested)	**Documented:** 5-7 meters normally, 3-4 meters at an angle **Tested:** 5 meters normally, 3-4 meters at an angle
4.	Delay when new node added to the network	3-4 seconds
5.	Delay caused by relocation of a node	3-4 seconds
6.	Delay at each node during multi hop transmission	3-4 seconds
7.	Delay after node removal	3-4 seconds

Based on the above results and analysis the following things can be concluded:

Once turned on for the first time the PIR takes 1 to 3 minutes to familiarize itself with the environment. During this time there is a possibility of false alarms.

The system is well suited for small homes and buildings. The system is easy to use and install and no lengthy procedures are needed to setup the hardware or the network. Delays will not affect the performance of the system as they are small and the maximum delays observed have been documented.

The system fulfills all the requirements that were mentioned in the requirements document.

7.4 Summary

The Smart Environment is an important system providing two essential functionalities in a single unit i.e. temperature monitoring and intrusion detection. An application has been developed to assist in monitoring i.e. Smart Monitoring System which responds to different events. The wireless sensor network used to monitor the environment is cost effective as it uses low cost and low power hardware (TI's Ez430-RF2500). In order to increase the coverage area of the network, a multi-hop routing protocol was implemented on these devices.

8 Chapter

Conclusion and Future Work

8.1 Introduction

This chapter describes the future scope of the project and the overall conclusion of the project. The project can be extended and few ideas are given in the chapter for the up gradation of the concept.

8.2 Conclusion

Security and safety are critical in all aspects of todays world, irrespective of whether human lives are involved or not. Even though stakes are higher if human life is at risk. Whatever the case compromise is not an option. In order to ensure safety easily deployable and easy to afford solutions are required. The Smart Environment Project is just one of those solutions. What makes this solution an effective one is that it is easily deployable with low cost. It can be used as basic tool for the detection of intrusion with homes and offices along with the provision to detect overheating resulting into fire. The developed solution has been rigorously tested and the end results are satisfactory.

8.3 Future work

A true Smart Environment is one that not only detects fire or intrusion but also automates and controls the whole environment. Therefore, this project is the foundation for a big and more practical solution.

For this purpose, more sensors can be attached with the existing nodes to monitor the effectively. These sensors may include humidity sensors, occupancy sensors, light sensors, carbon monoxide sensors, glass-breakage detectors, Smoke detectors, automatic meter reading: gas meters, water meters, e-meters, Active RFID applications etc.

Another extension of this project is that the current coverage area of developed solution is limited to homes or a single office floor. In order to enhance the coverage area of the network more base stations or sinks can be incorporated within the network which can be at different floors of the buildings. Therefore, sink-to-sink communication must have to developed in this regard.

8.4 Summary

This chapter documents the conclusions drawn after the completion of the Smart Environment Project. A section has also been dedicated to some of the future plans.

APPENDIX

SMART MONITORING SYSTEM

USER'S MANUAL

WELCOME

Thank you for using the Smart Monitoring System.

This user's manual is designed to be a reference for the operation of the Smart Monitoring System (SMS).

Here you can find detailed operation information related to the SMS.

Overview and Environment

The following sections give a brief overview of the Smart Monitoring System and the minimum requirements needed to run the software

Overview

It is software that comes with hardware required for the monitoring of temperature and intrusion.

It provides the user with a way to monitor the environments temperature changes. It shows temperature and intrusion information for each node that is added to the network. It can display this information graphically or in simple text format using the console mode. SMS can notify the user of an anomaly in the environment using different types of alerts. Visual which require the user to be at the computer, audio and SMS notifications if the user wants to be alerted via a text message.

It allows the user to customize settings and the graphical environment to fit their needs.

Environment

OS: Windows XP /Vista/7.

CPU: 1 GHz or higher.

Memory: 1GB or higher for XP OS.

Displayer: 1024*768 or higher.

Installations

This section documents the installation process.

Installation

Please check the installation CD and make sure it includes the following file:

setup.exe

Double click the setup.exe to begin installation. See Figure 2-1.

Figure 2-1 Welcome Screen for Setup

Click next button to go to License Agreement. See Figure 2-2.

Figure 2-2 License Agreement

Select the first option and click the next button, you can see an interface is shown, as in Figure 2-3. Please input user name and organization name.

Figure 2-3 Customer Information

Click next button, you can see an interface is shown, as below. Please select the Complete option to install all features of the Smart Environment System. See Figure 2-4.

Figure 2-4 Installation Type

Click next button, you can see there is an interface asking you to confirm the installation. See Figure 2-5.

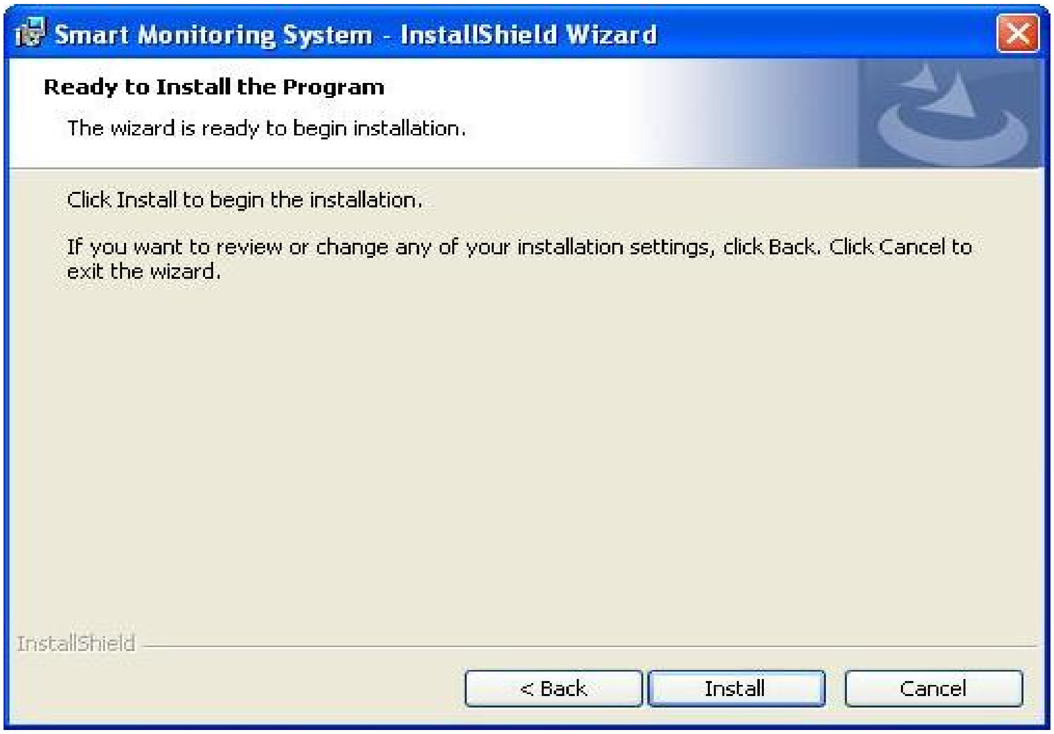

Figure 2-5 Ready to Install SMS

Click next button, system begins installation. The interface is shown as in Figure 2-6.

Figure 2-6 Installing

During the installation process, you can click cancel button to exit.

After installation, you can see an interface is shown as below. See Figure 2-7.

Figure 2-7 Completed Installation

Click close Finish button to complete the installation.

How to Use Software

This section describes the various features offered by the software and how to use these.

Menu

The figure 3-1 shows the dropdown menu button which includes the option of settings, Reset, disconnect and exit.

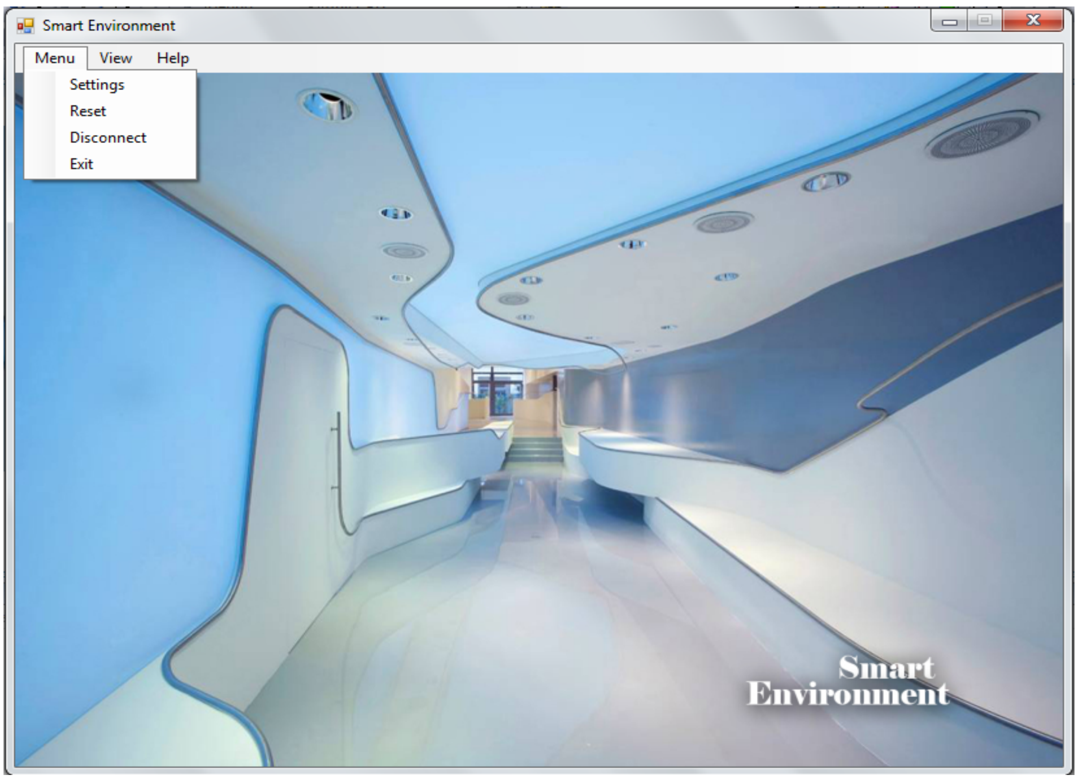

Figure 3 – 1 Dropdown Menu

Settings

The settings button is to access the settings of the system which includes the setting of critical temperature as shown in the figure 3-2. In figure 3-3 notification settings are displayed. From where you can enable/disable audio alerts and SMS notification and also set the text message content and the recipients cell phone number.

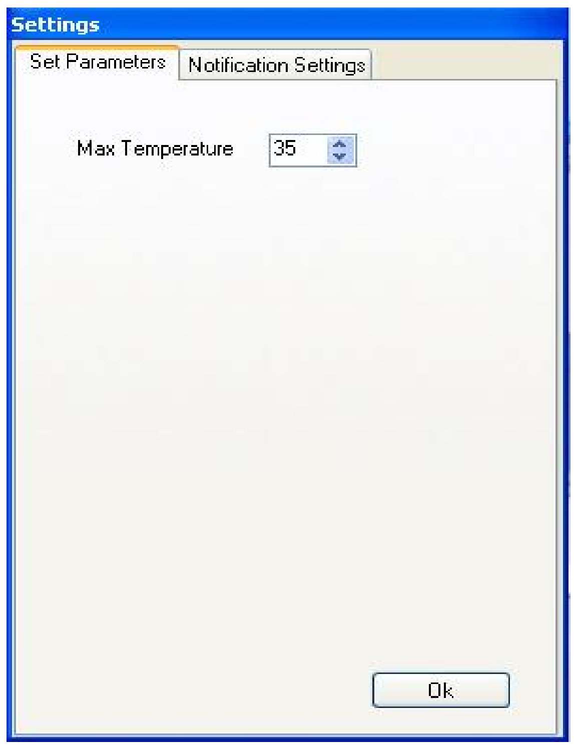

Figure 3 – 2 Set critical Temperature

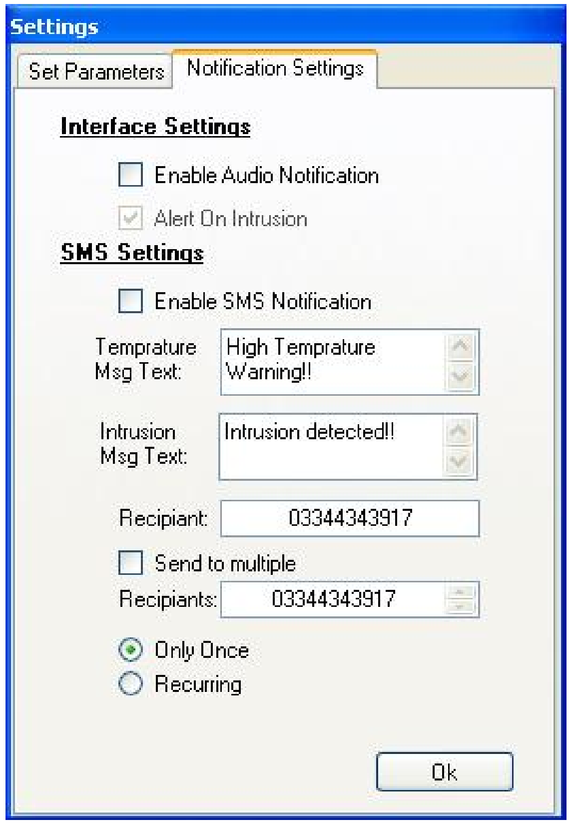

Figure 3 – 3 Notification Settings

Max. Temperature

This is the temperature value which when exceeded will result in the generation of an alert.

Notification Settings

This will enable/disable audio and SMS notifications. User can set the notification message text and select options to send notifications to a single number or multiple numbers.

Reset

The reset button is to reset the graphical user interface.

Disconnect

The disconnect tab is for disconnecting the software from all the sensors in the system.

Exit

Exit tab is to close the Smart Monitoring System.

View

The view dropdown includes Console, full screen and temperature view option in different units such as Celsius or Fahrenheit as shown in figure 3-2.

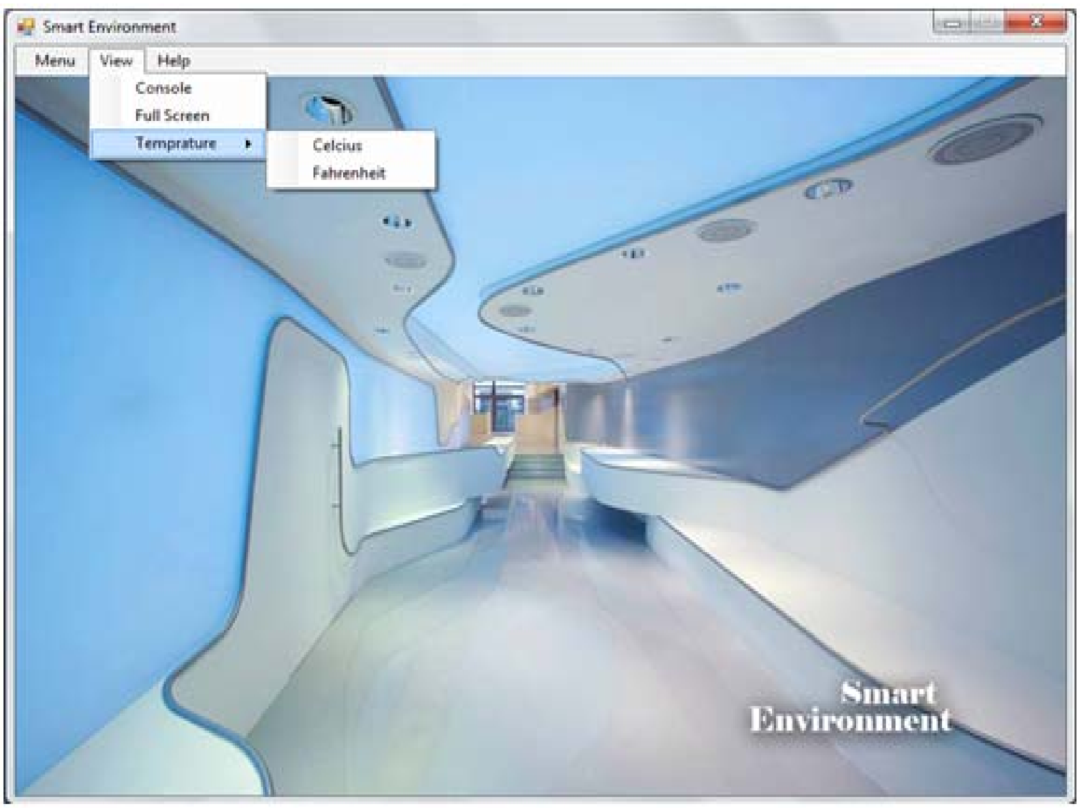

Figure 3 – 4 View dropdown with temperature option

Console

The console mode displays the received information to the user in the form of text. The following screen shot shows the console window.

The text is to be read as: the first value is the temperature, second indicates whether an intrusion has been detected, 0 means no intrusion and 1 means an intrusion. Next is the battery voltage of the sending node. This is followed by the id of the sending node followed by the path taken by the packet to reach the sink.

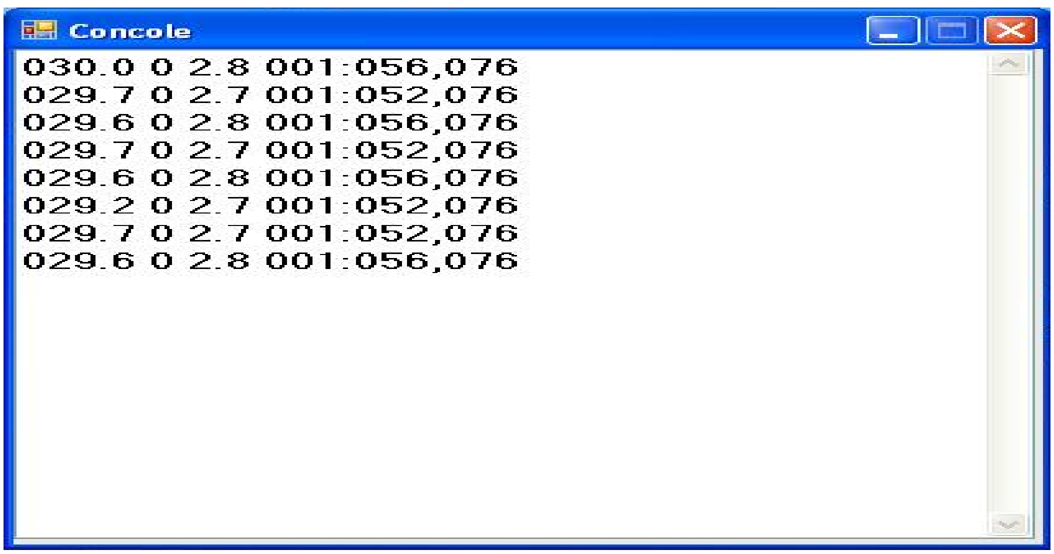

Figure 3– 5 Console

Temperature Unit

This causes the application to switch the output between Celsius and Fahrenheit.

Description of Graphical Elements

For each node which is connected to the network, a circular graphical element (bubble) is added to the Smart Monitoring System. Each element conveys the following information.

Temperature

Each node measures the temperature using the internal MSP430 temperature sensor. The sensed value of temperature is shown as text in the centre of the bubble. The user can choose whether the temperature be displayed in Celsius or Fahrenheit. In addition, the color of the bubble changes to red if the temperature value exceeds the threshold set by the user.

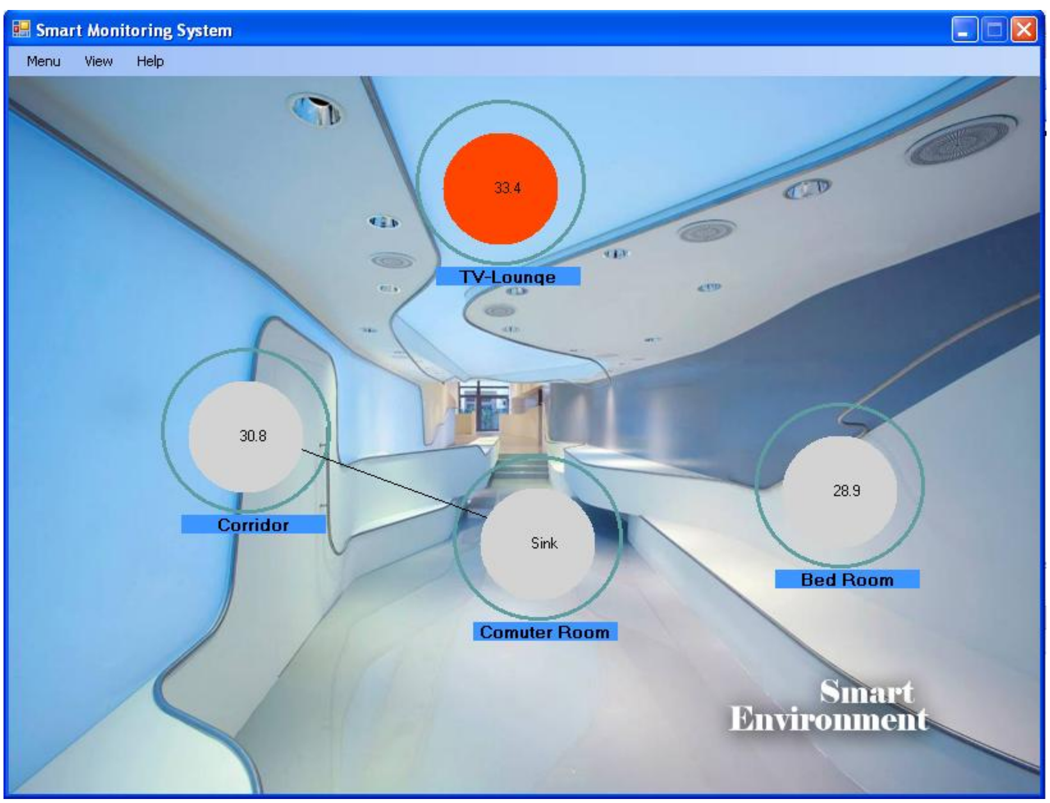

Figure 4– 1 Temperature Reading

Intrusion

Each node detects the intrusion using the PIR sensor soldered on eZ430 RF2500. The sensed intrusion is shown on the screen by displaying the red circles around the node in case of intrusion.

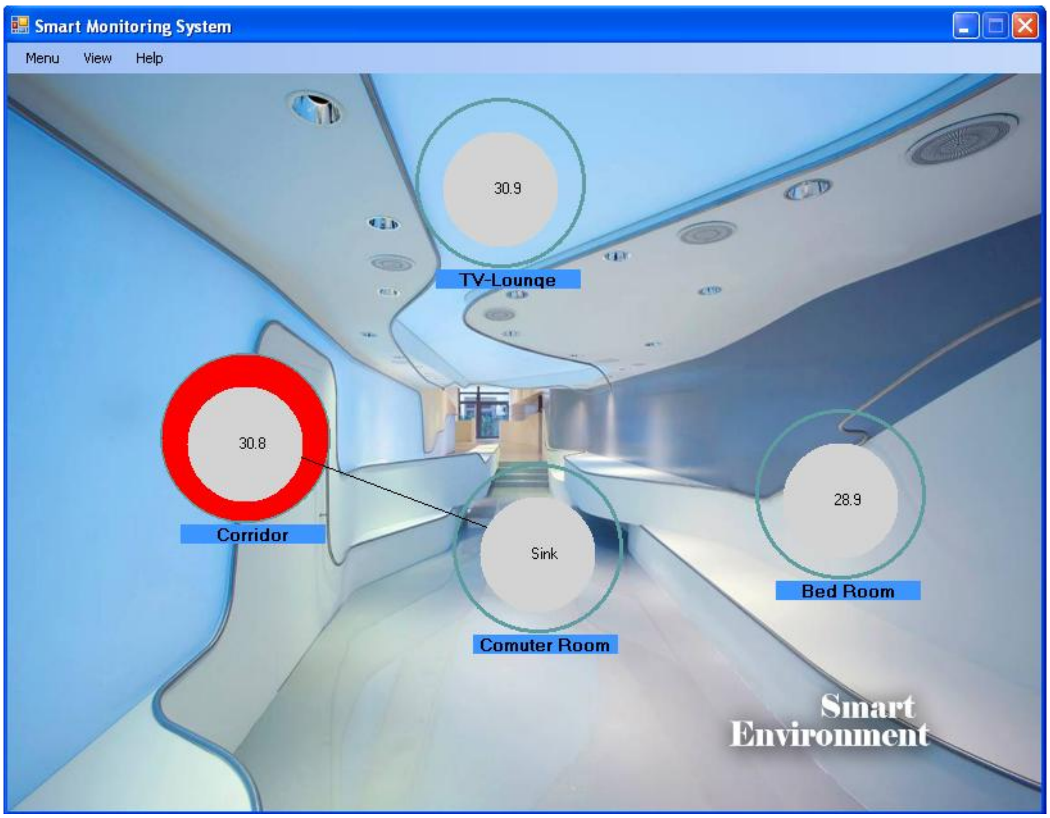

Figure 4 – 2 Intursion Detection

Voltage

Each device also sends the voltage of its batteries at the time of transmission. This can be viewed by keeping the mouse over the bubble (tooltip). In the picture below, it shows that the device in the bedroom has a voltage of 2.8 V.

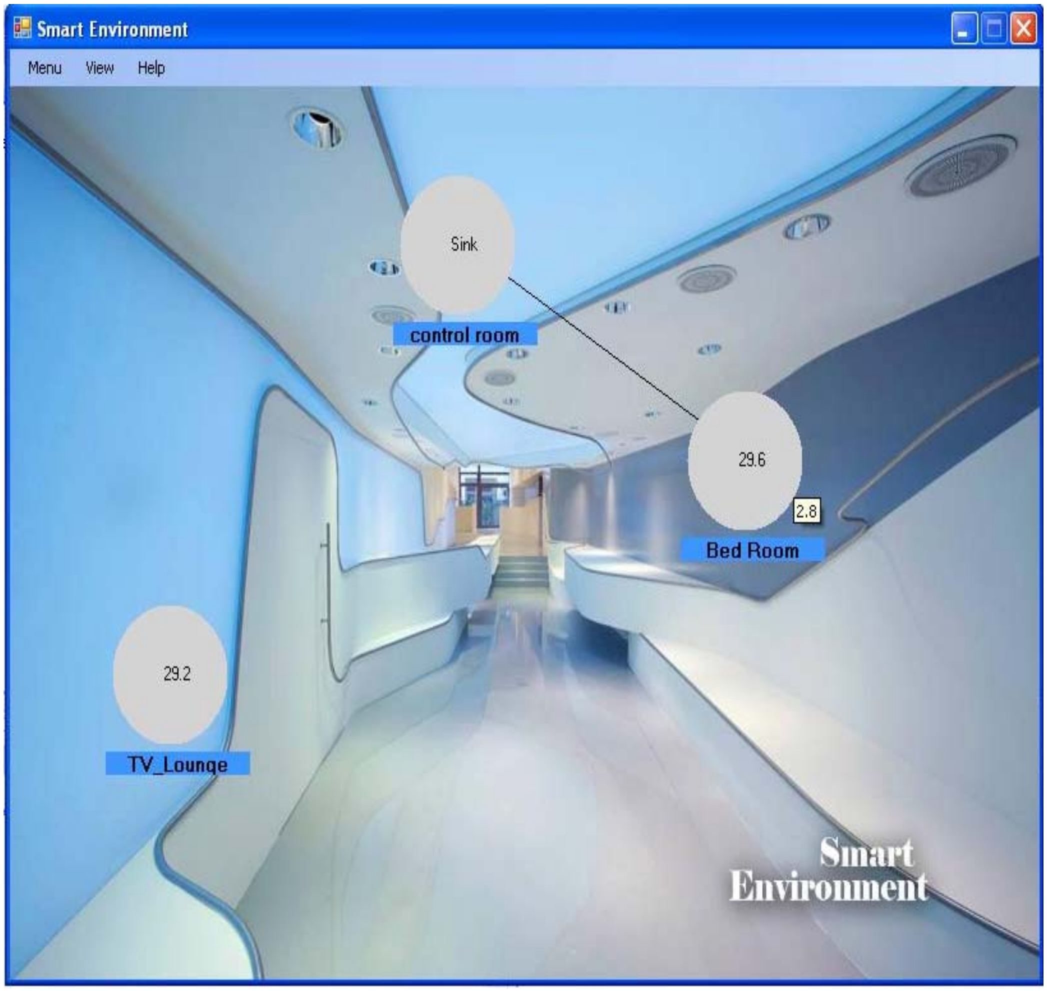

Figure 4– 3 Voltage

Node Name

Each device can be named as it joins the network. This name is displayed in with each device. This will assist the user to identify the location of the node. The user has the facility to edit the name by just double clicking the name.

Routing Path

When a packet is received, graphical lines are displayed from the device to the sink indicating the route taken.

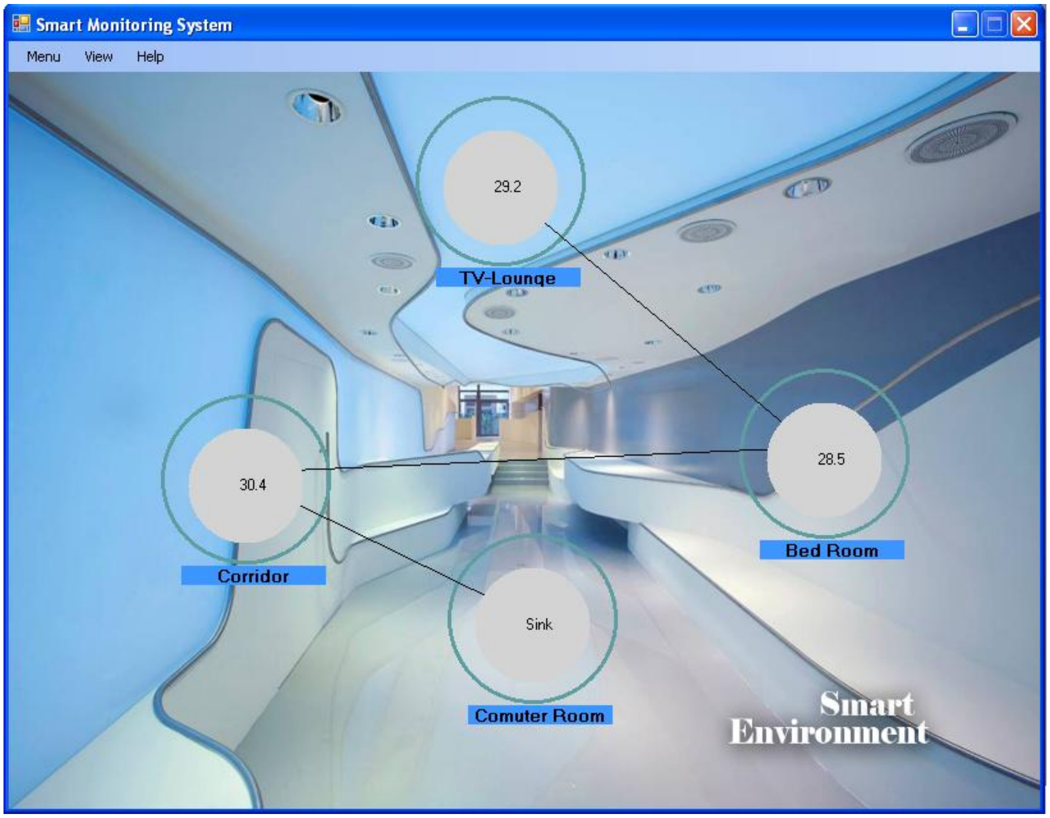

Figure 4– 4 Routing Path

In figure 4-4 the node in TV Lounge transmits to the node in the Bed Room which relays it to the corridor which in turn transmits the packet to the node in the corridor. Finally the packet is transmitted to the sink.

REFERENCES

[1]. P. Thubert, T. Watteyne, Z. Shelby, and D. Barthel, “LLN Routing Fundamentals,” IETF ROLL, IETF Internet-Draft, 9 April 2009, draftthubert-roll-fundamentals-01

[2]. R. D. Team, “RPL: Routing Protocol for Low Power and Lossy Networks,” IETF ROLL WG, IETF Internet-Draft, 4 February 2009, draft-ietf-roll-rpl-00

[3]. Robert D. Poor, “Gradient Routing in Ad Hoc Networks”, Available: www.media.mit.edu/pia/Research/ESP/texts/poorieeepaper.pdf

[4]. J. Faruque, K. Psounis, and A. Helmy, “Analysis of Gradient-based Routing Protocols in Sensor Networks,” in International Conference on Distributed Computing in Sensor Systems (DCOSS). MarinadelRey, CA, USA: IEEE/ACM, June 2005, pp. 258–275.

[5]. K. Liu and N. Abu-Ghazaleh, “Aligned Virtual Coordinates for Greedy Routing in WSNs,” in International Conference on Mobile Adhoc and Sensor Systems (MASS).Vancouver, Canada: IEEE, 2006, pp.377– 386.

[6]. Thomas Watteyne, Kris Pister, Dominique Barthel, Mischa Dohler and Isabelle Auge-Blum, “Gradient Routing in Wireless sensor networks” in Global telecommunication conference IEEE, 2009.

[7]. Charles E. Perkins, Elizabeth M. Belding-Royer, and Samir Das. "Ad Hoc On Demand Distance Vector (AODV) Routing." IETF RFC 3561

[8]. David B. Johnson, David A. Maltz, and Josh Broch. DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks. in Ad Hoc Networking, edited by Charles E. Perkins, Chapter 5, pp. 139-172, Addison-Wesley, 2001

[9]. Perkins, Charles E. and Bhagwat, Pravin (1994) (pdf). Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers

[10].Texas Instruments. *SimpliciTI.* Available: http://www.ti.com/corp/docs/landing/simpliciTI/index.htm?DCMP=hpa_rf general&HQS=NotApplicable+OT+simplicity. Last accessed Dec 2010

[11]. eZ430-RF2500 Development Tool, User's Guide, p-11

[12]. MSP430x2xx Family User's Guide, Texas Instruments, p.603

[13]. Futurlec*. PIR Sensor Module (SB0081) Data Sheet.* Available: http://www.futurlec.com/Datasheet/Sensor/PIR_Module_B.pdf. Last accessed Feb 2011

[14]. Cook, Diane; Das, Sajal (2005). *Smart Environments: Technology, Protocols and Applications*. Wiley-Interscience.

[15]. Wikipedia. *Burglar Alarm.* Available: http://en.wikipedia.org/wiki/Burglar_alarm. Last accessed Jan 2011.

[16]. Wikipedia. *Fire Alarm System.* Available: http://en.wikipedia.org/wiki/Fire_alarm_system. Last accessed Jan 2011.

[17]. Mark Greyson . (May 30, 2010). *Wireless Motion Detector Types – Ultrasonic, IR, PIR, Continuous Wave Radar and Vibration.* Available: http://ezinearticles.com/?Wireless-Motion-Detector-Types---Ultrasonic,-IR,-PIR,-Continuous-Wave-Radar-and-Vibration&id=5429129.