Sensor Network Embedded Intelligence: Human Comfort Ambient Intelligence

Mohd Izani Mohamed Rawi

A thesis submitted to
Auckland University of Technology
in fulfilment of the requirements for the degree of
Doctor of Philosophy (PhD)

School of Engineering
Auckland, New Zealand
2013

Primary Supervisor: Professor Adnan Al-Anbuky

Attestation of Authorship

"I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgements), nor material which to a substantial extent has been accepted for the award of any other degree or diploma of a university or other institution of higher learning."

November 2013

Mohd Izani Mohamed Rawi

Acknowledgements

First and foremost I would like to express my deepest gratitude to الله , the most gracious and most merciful God for the blessing, wisdom, health, strength and patience that he gave upon me throughout this long adventurous, exciting, inspiring and challenging PhD journey.

This journey will not be a dream comes true without my supervisor's undivided guidance, patience and support throughout this study, and especially for continuously encouraging me to keep on working hard to complete this thesis. Thank you Professor Adnan Al-Anbuky, Director of Sensor Network and Smart Environment Research Centre (SeNSe) for keeping me focused and on the right track throughout these years.

I would like to thank my co-supervisor, Dr. Paola Leardini, School of Architecture and Planning, Auckland University, for her generous and patient to reviews, comments, and give thoughtful suggestions throughout the research period. I would also like to acknowledge Associate Professor Uwe Rieger, School of Architecture and Planning, Auckland University and Kerstin Rosemeier for their support in the early stage of my work. I am forever grateful and thankful to have met and been given the opportunity to work with all of you.

I dedicate this thesis to my dear parents, Mohamed Rawi Che Muda and Nafisah Hamat, who have always been there to support me in my journey in every possible way. Without their blessing and encouragement, this thesis would have not been completed. My deepest thanks also to my parent-in-law, brothers and sisters, for their endless encouragement for me to walk the journey till the end.

Special thanks to my dear friends in Auckland – Kamarul & Zila, Badrul & Fariza, Azizi & Zaimah, Suhaimi & Zakiah, Wan & Zah, Farid & Dila, Haza & Ina and Anas & Fizah. Thank you for the supports, helps, joy and laughter we shared together throughout these years. You will be forever remembered.

I would also like to acknowledge my fellow SeNSe collegues and AUT friends – Hakilo, Siva, Mirjana, Zimran, Saud, Yang, and many more – who have always provided support, insight, thought, interest and advice to my thesis.

I would also like to acknowledge the postgraduate office of AUT School of Engineering for their support and assistance necessary related to the PhD funding and doctoral forms. Special thanks to the Administrators of the School of Engineering, Auckland University of Technology.

Finally, the last and the best – to my dearest wife, Zarini Ghazali. Without your patience, sacrifices, love and support this thesis would not have been completed. To my dear children, Harith, Hanis, Hazim and Hanif, thank you for being with me, day and night, rain and shine. I will always cherish the loves, supports, joy and hard time we faced together throughout this journey.

Fiat Sapientia Virtus

Abstract

This study explored the multidiscipline domain of the Wireless Sensor Network (WSN) and Ambient Intelligence (AmI) in addressing the problem of the comfort of a living space. This thesis addresses the potential for embedding an intelligent engine into WSN and the aggregation of multiple comfort factors in a living space. The four most important comfort factors for humans are taken into account. These are thermal comfort, visual comfort, indoor air comfort and acoustical comfort.

This thesis introduces a WSN based embedded intelligent system architecture and a system framework for a living space's comfort level. Human Comfort Ambient Intelligence System (HCAmI) architecture is presented. The HCAmI key component encompasses a flexible generic distributed fuzzy engine embedded within WSN nodes. The engine serves as a key knowledge component in solving specific human comfort requirements.

With the proliferation of pervasive computing, there is an increasing demand for the inclusion of WSN in wider areas such as buildings, living space, system automation and much more. Focusing on buildings and living space alone, multitudinous studies have been made of environmental comfort for occupants. A smart environment and low energy homes are amongst the driving forces behind this research. Also, WSN research has been progressing well and expanding into various aspects of life such as support of the elderly, environmental senor, security and much more. Unfortunately, separate studies have been conducted in their own discipline focusing on specific issues and challenges. Little attention has been paid to putting it together under one roof. Lack of interdisciplinary research inspired this effort to unite these unconnected research domains. This has acted as the key motivational catalyst. The motivation behind combining these effects brings us to the specific issue of the human comfort realm that prompted this study. Human comfort deals with providing a comfortable and healthy place for people to live. Hence, in a living space, other than good design and construction, it is essential to monitor and maintain the modifiable environment such as temperature, lighting, humidity, noise, air quality and psychological factors. Functional environmental comfort system adaptability and the WSN system determination to solve the problem is a fascinating issue that certainly warranted further investigation.

The HCAmI concept was designed and implemented based on a knowledge based architecture and framework. This approach addressed the component level first, catering for the four key human comfort factors. The system level design was then looked at. Each individual component was subjected to simulated and real sensor data and tested against a corresponding model built using appropriate tools such as the MATLAB Simulink and Sun SPOT Solarium WSN simulator. The HCAmI System was used to collect raw data from 20/04/2010 to 26/08/2010 (four months of data) in the SeNSe Laboratory, School of Engineering. A short snapshot of the collected data (from 08:00am 25/08/2010 to 11:40am 26/08/2010) is presented as a case study.

The main achievement / contribution of this thesis is a distributed fuzzy logic based wireless sensor node in the human comfort realm. The framework, architecture and development of an integrated human comfort concept could be embedded in a wireless sensor network environment. The modular architecture and framework presented here highlights the flexibility and integrated approach of the design. The knowledge component of each comfort area can be changed easily and adding or removing comfort components is catered for as well.

Overall, this thesis adds to the WSN body of knowledge in an embedded distributed generic fuzzy engine, thermal comfort engine, spatial sensing engine, human comfort index engine, application layer communication protocol and specific external sensor driver development and interface for Sun SPOT WSN.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AAL	Ambient Assisted Living
AC	Acoustical Comfort
AI	Artificial Intelligent
AmI	Ambient Intelligence
ANN	Artificial Neural Network
API	Application Programming Interface
Clo	Clothing Level
DBT	Dry Bulb Temperature
DI	Defuzzyfication Interface
DSSS	Direct Sequence Spread Spectrum
FI	Fuzzyfication Interface
FIE	Fuzzy Inference Engine
FRBS	Fuzzy Rule Based System
GPIO	General Purpose I/O
HCAmICP	HCAmI Communication Protocol
HCI	Human Comfort Index
HVAC	Heating, Ventilation, and Air-Conditioning
IAC	Indoor Air Comfort
IAQ	Indoor Air Quality
IDE	Integrated Development Environments
IDW	Inverse Distance Weight
JVM	Java Virtual Machine
LQRP	Link Quality Routing Protocol
Met	Metabolic Rate
MRT	Mean Radiant Temperature

OTA	Over The Air
OTA-Conf	Over The Air Configuration
PMV	Predicted Mean Vote
RF	Radio Frequency
RH	Relative Humidity
TC	Thermal Comfort
VC	Visual Comfort
Vel	Air Movement
WHANs	Wireless Home Automation Networks
WSN	Wireless Sensor Network

Chapter 1 Overview

1.1 Introduction

This chapter provides background information, providing a general description of the Wireless Sensor Network (WSN), followed by embedded intelligence and ambient intelligence. Subsequently, a brief explanation of a smart environment and human comfort are discussed. Finally, the research context and thesis outline are presented.

1.2 Background to the Research

WSN enables a close analysis of sensor data about the real world, and promises to revolutionise our understanding of and interaction with our environment. WSN devices are small, communicate wirelessly, have limited power, are low-cost, and most importantly are embedded in the physical world through their onboard sensors and actuators.

WSN is the key enabling technology for building systems that adapt autonomously to their environment, without direct human intervention. Most sensor networks operate in free air. Research being conducted in Ireland, by the School of Computer Science and Informatics at UCD Dublin and the Centre for Adaptive Wireless Systems at Cork IT, begun to explore the tools and techniques we need in order to build 'augmented materials' that combine sensing, actuation and processing into the fabric of built objects [1].

With the proliferation of pervasive computing in the digital environment there is an increasing demand for the inclusion of WSN in broad areas such as buildings, utilities, industry, home, on ship, transportation systems in automation and much more [2]. Sensory data comes from multiple sensors of different modalities in distributed locations. The smart environment needs information about its surroundings as well as about its internal workings. With a more powerful WSN being introduced, more intelligence can be embedded into WSN itself – such as fuzzy logic based smart routing, swarm behaviour, ant colony optimisation and so forth. Systems, in combination with pervasive (or ubiquitous) computing, cognitive intelligence and software-intensive systems, which in fact means 'embedded intelligence', or 'smart systems' in the broader context of 'smart environments', are the most important challenge for strategic, long term research, with a huge impact on society and the economy [3].

All embedded intelligence work in the WSN is in line with latest the Artificial Intelligence evolution – Ambient Intelligence (AmI). AmI refers to electronic environments that are sensitive and responsive to the presence of people. AmI is a vision of the future of consumer electronics, telecommunications and computing originally developed in the late 1990s for the time frame 2010 – 2020 [4].

In an ambient intelligence world, devices work in concert to support people in carrying out their everyday activities, tasks and rituals in an easy, natural way using information and intelligence that is hidden in the network connecting these devices. As these devices grow smaller, more connected and more integrated into our environment, the technology disappears into our surroundings until only the user interface remains perceivable by users [4].

In essence, AmI refers to a digital environment that proactively, but sensibly, supports people in their daily lives [5, 6].

In this work, WSN is evaluated as an enabling instrument to improve human comfort. This is perceived by monitoring the human comfort level and energy efficiency of a living space. Furthermore, WSN is examined here as an enabling tool and a technology enabler of AmI by means of embedding the intelligence within the WSN itself.

This work offers a unified solution / framework that provides distributed heterogeneous sensors that have the ability to execute AmI components. An architecture that defines solution’s implementation, execution and management is presented. The work also describes the performance of the system’s implementation in a real life environment.

1.3 Wireless Sensor Network

A wireless sensor network is a collection of nodes organised into a cooperative network. Each node consists of processing capability (one or more microcontrollers, CPUs or DSP chips), may contain multiple types of memory (program, data and flash memories), have a RF transceiver (usually with a single omni-directional antenna), have a power source (e.g., batteries and solar cells), and accommodate various sensors and actuators. The nodes communicate wirelessly and often self-organise after being deployed in an ad hoc fashion. Systems of 1000s or even 10,000 nodes are anticipated. Currently, wireless sensor networks are beginning to be deployed at an accelerated pace. It is not unreasonable to expect that in 10-15 years the world will be covered with wireless sensor networks. One possible access to these networks is via the Internet. This can be thought of as the Internet becoming part of a physical network. This emerging technology is exciting, with unlimited potential for numerous application areas including environmental, medical, military, transport, entertainment, crisis management, homeland defence, and smart spaces [7]. Today, such network arrangements are used in countless consumer and industrial applications, for instance patient monitoring systems in the health sector, industrial process monitoring in the manufacturing domain, environmental watches in wildlife monitoring, structural monitoring of a bridge or building, interior and exterior monitoring for a smart home, and so on.

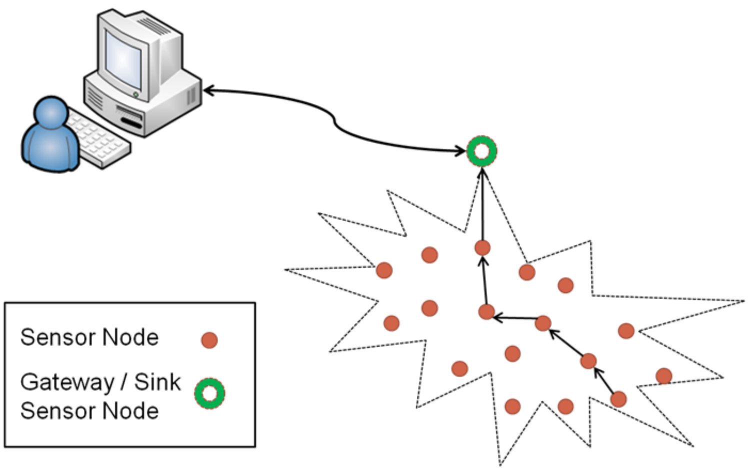


Figure 1-1. Typical Multihop Wireless Sensor Network Architecture

Figure 1-1 [7, 8] depicts a typical arrangement of WSNs with multiple sensor nodes and at least one gateway sensor node / sink node that connects other sensor nodes to the outside world. From a WSN topology point of view, WSN can be as simple as a star network. It could also be as complex as an advanced multi-hop wireless mesh network. The nodes communicate among themselves between hops by means of a routing or flooding propagation technique. With this kind of network arrangement, WSN can cope with acute power constraints, the ability to recover from node failures, nodes nobilities, communication failure dynamic, scalability issues, and the heterogeneity of nodes [8].

There are several standardisation groups in the field of WSN communication protocol. IEEE, the Internet Engineering Task Force, the International Society of Automation and several non-standard, proprietary mechanisms and specifications work on covering all communication protocol layers. Predominant standards used in WSN communications are WirelessHART (an extension of wired HART protocol), the IEEE 1451.5 Wireless Smart Tranducer Interface standard, ZigBee Alliance / 802.15.4 Low Rate Wireless Personal Area Network standard, ISA100, created by the International Society of Automation, and the IPv6 over Low Power Wireless Personal Area Network (6LoWPAN) [9].

1.4 Intelligence

1.4.1 Embedded Intelligence

Embedded intelligence is a modern frontier of research combining all sorts of computing elements and the development of a new class of intelligent algorithms such as an unsupervised learning algorithm, dynamic adaptation and efficient heuristics. Embedded intelligence can be said to be the ability to embed sensors in the environment (fixed and mobile, low and high bandwidth sensors) and to use them to extract information about the environment for use by a human working in the environment [10, 11].

Generally, components supporting embedded intelligence systems are small and usually hidden within much larger and more complex computing or electronic devices, so they often go unnoticed. Embedded intelligence systems are becoming smarter and more network friendly.

1.4.2 Ambient Intelligence

Ambient Intelligence is the latest in Artificial Intelligence evolution as shown in Figure 1-2 [6]. The ambient intelligence paradigm builds upon ubiquitous computing and human-centric computer interaction design and is characterised by systems and technologies that are:

- Embedded: many networked devices are integrated into the environment
- Context awareness: these devices can recognise people and their situational context
- Personalised: they can be tailored to your needs
- Adaptive: they can change in response to you

- Anticipatory: they can anticipate your desires without conscious mediation.

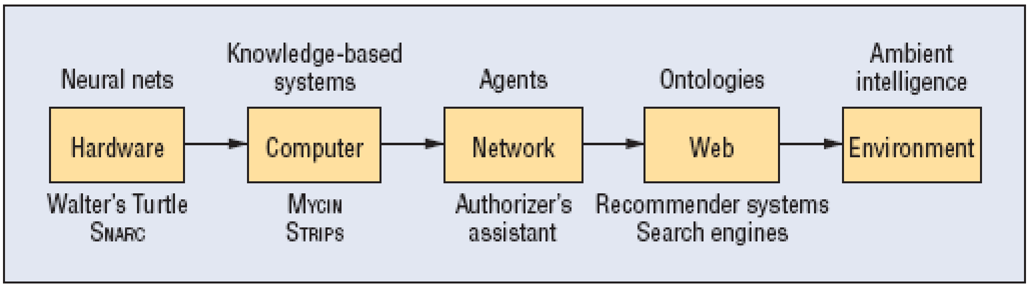


Figure 1-2. The Evolution of Artificial Intelligence

In essence, ambient intelligence can be defined as a digital environment that is sensitive, adaptive and responsive to the presence of people [12]. Ambient intelligence will cover the home, vehicle and clothing in private, work and public places. One could assume that ambient intelligence is something that will happen in the near future and will be everywhere in our modern life.

Soon, ambient intelligent will be the latest trend within an information society environment, with intelligence embedded anywhere and in an inconspicuous manner [13]. The trend is on greater user friendliness, an improved and efficient service support, smart user empowerment and well integrated human interaction.

1.5 Smart Environments and Low Energy Homes

What is a Smart Environment? According to Weiser et al. [14], it can be regarded as "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". It is envisioned as the derivative of pervasive computing and the availability of cheap and powerful computing power, thus making human interaction with a system a pleasurable experience. Ubiquitous computing can be regarded as an enabling technology and tools for smart environment that could have far reaching consequences for society [15]. The modern way of life could be surrounded with high tech gadgetry such as smart laundry machines, intelligent kitchen appliances, embedded location tracking and communication that might drastically change the social fabric of our lives.

Meanwhile, Cook and Das [2] define a Smart Environment as "a small world where different kinds of smart device are continuously working to make inhabitants' lives more comfortable". The aim is a satisfying experience in all environments with the replacement of hazardous work, physical labour intensive and repetitive tasks with automated agents. It ultimately provides supports and enhances the abilities of its occupants in executing tasks [16], whilst minimising safety risks such as bodily harm and injury. This spans tasks from unfamiliar navigation activities, activities reminders, to shifting or moving heavy objects for senior citizens or disabled persons.

The Smart Environment can be differentiated as i) smart environments for systems, services and devices, ii) virtual (or distributed) computing environments, physical environments and human environments or iii) a hybrid combination of these [17]. *Virtual computing environments* facilitate smart devices to access relevant services anywhere and anytime such as the Google Cloud Print Service. *Physical Environments* could be embedded with multitude smart devices of different kinds such as tags, RFID, sensors and controllers and come in various sizes, from nano to micro to macro. *Human Environments* represent humans, either individually or collectively, essentially forming a smart environment for devices. They may be coupled with smart devices such as smart phones, wearable computing, and embedded devices such as pacemakers and hearing implants.

Fundamentally, Smart Environments are regarded as having a number of distinct features such as remote control of devices, device communication, information acquisition / dissemination from sensor networks, enhanced services by intelligent devices and predictive and decision making capabilities.

One important area of implementation is Low Energy Homes. These homes can be considered as any type of dwelling that from design, technology, building materials and products uses significantly less energy than a traditional or average modern house. Generally, a Low Energy Home makes use of sustainable design, sustainable architecture, low-energy building, and energy efficient landscaping to reduce its energy expenditure [18]. A variety of methods can be used, such as active solar and passive solar building techniques and components. Coupled with a Smart Environment, it presents a paradigm shift in human experience, consequently changing forever how we live.

1.6 Human Comfort

As a warm-blooded species, humans maintain a constant body temperature of 37°C, in spite of the surrounding air temperature. However, to be truly comfortable, in a certain environment, it is desirable that surrounding environmental factors such as air temperature, light intensity, air quality and sound levels are within certain dynamic and interrelated comfortable ranges [19]. Human comfort is the result of the interaction between all the senses. Hence, when we plan to build and construct our living space, it is very important that we bear this in mind.

1.6.1 Thermal Comfort

Thermal comfort is defined as a condition of mind that expresses satisfaction with the thermal environment [20, 21]. It is regarded as a state of mind that expresses satisfaction with the surrounding thermal environment [21, 22]. Thermal comfort is affected by heat conduction, convection, radiation and evaporative heat loss. Thermal equilibrium is influenced by environmental parameters as well as physiological parameters [23]. Thermal comfort is maintained when thermal equilibrium with the surroundings is achieved. Human thermal sensation is mainly related to the thermal balance of the body as a whole. This balance is influenced by environmental parameters as well as physiological parameters.

Factors that determine thermal comfort include:

i) Environmental Parameters such as Dry Bulb Temperature (DBT), Mean Radiant Temperature (MRT), Relative Humidity (RH) and Air Movement (Vel)

ii) Physiological Parameters such as Metabolic Rate (Met) and Clothing Level (Clo)

When these factors have been estimated or measured, the thermal sensation for the body as a whole can be predicted by calculating the Fanger's Predicted Mean Vote (PMV) [23]. The PMV equation for thermal comfort is a steady-state model. It is an empirical equation for predicting the average vote of a large number of people on a 7 point scale (-3 to +3) of thermal comfort [23].

Maintaining the right thermal comfort for building / living space occupants is one of crucial goals of architect and engineers. The importance of thermal comfort is recognised as being closely related to human work performance and productivity rate. Thermally discomfort environment can lead to serious problem such as *Sick Building Syndrome* symptoms [20]. In the long term, thermal comfort related problems could be costly to deal with.

1.6.2 Visual Comfort

Visual comfort is another factor that contributes to total environmental comfort. For over four billions years our sun has been shining over the earth. Thus, throughout time, a plethora of living beings have relied on the natural light provided by the sun for health and survival. Humanity, as an intelligent species, moved from living in caves to huts and finally to modern buildings. As people build modern sealed living spaces that are fully air conditioned and artificially lit, they create a challenge to architects and engineers to control light fluctuation, levels, colours and direction with natural and artificial light.

The lighting function of an interior is to ensure the safety of people by ensuring any hazards are visible, to facilitate the performance of visual tasks by making relevant details of the task easy to see and to aid the creation of an appropriate visual environment. An inadequate lighting setup can cause sore eyes and fatigues (negative visual comfort). Different visual environments can be created by manipulating the relative emphasis given to various objects and surfaces. This art and science of daylighting is not just about light, but involves consideration of heat gain, glare, light levels and uniformity, and solar penetration. A good daylighting design can increase productivity and may lower overall energy consumption of a living space. With lower energy spent on lighting (about 20-25% of total energy consumption), consequent reduction of CO_2 and pollutants can be achieved.

Defining visual comfort from a scientific perspective has yet to be agreed upon. It is difficult to define what is meant by visual comfort. Visual comfort can be viewed as a state of mind that expresses satisfaction with the lighting scenes and visual environment. With this in mind, we can define the main characteristics that have been studied and that contribute to an individual's visual comfort as illuminance recommendation, glare and sparkle and visual comfort probability.

Proposed illuminance recommendations require luminance for type of task as follows:

Category of Visual Task	Required Luminance (fL)
Casual	3 – 6
Ordinary	6 – 30
Moderate	30 – 60
Difficult	60 – 120
Severe	Above 120

Table 1-1. Illuminance Recommendations

These numbers represent the amount of light in foot candles required to provide visual comfort doing a certain task.

Glare can be regarded as a negative sensation of perception that results from excessive brightness in the field of view. Sparkle is a positive sensation of perception that results from excessive brightness in the field of view.

Visual comfort probability is defined as the percentage of normally individuals who will be comfortable in a specific lighting scene and visual environment [24].

1.6.3 Indoor Air Comfort

Indoor Air Quality (IAQ) deals with the contents of interior air that could affect the health and comfort of building occupants, for example, microbes, chemicals, allergens, or particulates [25, 26]. For hundreds of years, the threat posed by air pollution to public health has been acknowledged. With the Industrial Revolution and rapid use of fossil fuel, such threats posed by air pollution were recognised. Throughout recent times, a series of air pollution related disasters have been documented, which prompted a wave of epidemiologic and other research that recognised the public health risk of air pollution [25, 26].

Current evidence suggests there are at least four major links between human health & productivity and the quality of indoor environment [25, 26]. The links are:

i) infectious disease transmission such as influenza and pneumococcal disease
ii) allergies and asthma triggered by a number of allergens
iii) acute building-related health symptoms commonly called *Sick Building Syndrome Symptoms*

iv) direct impact of indoor environmental conditions on human performance such as thermal conditions, lighting, acoustics, vibration, aesthetics and odours

Therefore, it is appropriate to recognise indoor air quality factor as a determinant in designing a healthy building. The preservation of satisfactory indoor air quality involves proper construction and monitoring of the building and its systems. Hence, finding measurement technologies and techniques that are suited to the monitoring task is an ongoing activity for indoor air quality engineers and designers.

Indoor air comfort depends on IAQ, which can be affected by gases, particulates and microbial contaminants. Quantifying IAQ involves the collection of air samples and monitoring human exposure to pollutants. Common gases that can be sampled for are radon, carbon monoxide, CO_2 and ozone.

CO_2 is the prime factor in indoor pollution, mainly emitted by humans and associated with human metabolic activity. High levels of indoor CO_2 may cause occupants to feel drowsy, get headaches, or have significant drops in activity levels. Thus, maintaining a healthy level of CO_2 is the primary goal when we deal with indoor air comfort. ASHRAE recommends that indoor CO_2 should not exceed 700ppm above outdoor ambient levels [22].

1.6.4 Acoustical Comfort

Acoustical comfort can be related to the reaction of occupants to the indoor acoustical environment. Acoustical environment can be described in term of sound pressure level and audibility [27].

The degree of acoustical comfort in a living space is highly related to the combined effects of unwanted noise and the desired level of speech privacy [28]. Low ambient noise level can have a soothing effect and excessive noise level is not. Excessively high noise levels can lead to major problems in a living space, where unwanted noise can cause stress, fatigue, inability to concentrate, affect human wellbeing and eventually impair productivity [22].

Several common noise problems influence occupants such as:

i) too much noise outside the building entering the space, for example road traffic and machinery sound,
ii) too much noise from adjacent spaces, for example human activity sounds from adjacent rooms and
iii) lack of sound control in the space itself, for example sound from within the occupied space (voices, human activities, entertainment devices and office machinery)

Usually these noises are not high enough to be harmful to human hearing, but might distract the occupant's concentration on work or enjoyment [22].

ISO 61813 emphasises that the general principles of building environment design should take into account a healthy indoor environment for the present occupants, and protect the environment for future generations [27]. It promotes the assessment of the proposed building's design criteria for indoor air quality, thermal comfort, acoustical comfort, visual comfort, energy efficiency and HVAC system controls at every stage of the design process [27]. The goals of excellent acoustic design are to enhance wanted sounds and attenuate unwanted sounds (noise). People prefer to work in a silent environment but one not completely void of sound. Human need sound for orientation, awareness and our own speech privacy [29, 30].

1.7 Research Context

Over the past few decades, many universities, research institutes and commercial bodies in many parts of the world have been undergoing a significant shift from traditional independent research to collaboration with various disciplines. Recent years have seen a wide and increasing interest in WSN and the Human Comfort domain. The reason WSN has become so attractive is that due to advancement in embedded electronics (processors, sensors, actuators, etc), availability and economics of scale make it an easily accessible tool to work with. This opens the possibility for real-time decision making based on information from numerous WSN subsystems and machinery working together in an efficient manner [31].

As human life progresses, the issue of high quality lifestyles becomes more important. Diverse areas of human comfort have been examined in great detail, such as the thermal, visual, air quality, acoustical and spatial comfort domains. Initially, these areas studied separately. Human comfort is complex and complicated business, it became clear that these areas cannot be studied in isolation. The information collected is extensive, so it becomes impossible for a human operator or single, simple machine to analyse such an overwhelming amount of information to make any decision. Therefore, this work focuses on combining these multidisciplinary research domains to help unite information types into one simpler, more convenient structure for human operator / owner.

1.7.1 Problem Statement

Many recent research topics have focused on WSN technology development. Gradually, with the introduction of more powerful and energy efficient WSN, embedding intelligence into the sensor itself has become possible. WSN no longer functions as a basic “sense and send” device. It has moved into pervasive computing environments, where a sensor node has close interaction with actuators in the environment and behaves according to the context information surrounding it. Also, through the demand for a better lifestyle and lower energy consumption, considerable effort has been devoted to intelligent building / living space research and development in recent years. The embodiment of three different research domains of WSN, embedded intelligence and a comfortable and energy efficient living space is essential in realising this goal. One promising living space that envisages a comfortable living space with low energy need is Smart and Low Energy Building. With the rapid growth of this kind of building throughout the world, there is a pressing need for a complete solution to manage, monitor and interact with occupant and living space activities. WSN offers a promising solution. The implementation of intelligence can be defined within the sensor nodes themselves rather than a letting higher entity perform the intelligence process. Detailed research could be done in this area to look at what kind of intelligence can be embedded, with all the constraints of WSN in mind, yet fulfil the need for a Smart and Low Energy Building. Close cross-domain collaboration would produce an optimised environment control system for a Smart and Low Energy Building.

1.7.2 Motivation

To provide a comfortable and healthy place to live is the sole reason such a building is designed and constructed. Therefore, the primary goal of WHAN is to monitor and maintain acceptable human comfort levels in dwellings, at the same time catering for the secondary objective of optimising energy use. For example, most immediate to human comfort is thermal comfort. Is 17.5 degrees Centigrade enough for comfort? Should the temperature be increased? What is the caveat? How can it be achieve? Does the action have any impact on other comfort factors as well?

In our enthusiasm for having ideal cosy homes, complex relationships between the aforementioned comfort factors have to be looked at in detail. The association between these comfort factors was explored and modelled as a hybrid fuzzy reasoning expert system that was marked as a human comfort index. Given a living space, a group of WSN nodes will sense the environment (temperature, humidity, clothing, metabolic rate, wind speed, luminosity, etc), make sense of raw information and by means of fuzzy logic, weighting and integration criteria, measure the Human Comfort Index (HCI).

Functional environmental comfort system adaptability and WSN system determination in solving the problem are fascinating issues that certainly warrant further investigation.

1.7.3 Research Questions

Specifically, this study seeks to answer the following research questions:

i) How would a WSN environment accommodate these individual comfort factors?

ii) How would integrated comfort factors be implemented within a WSN environment to provide the necessary AmI?

iii) How would the availability of individual comfort information, as well as advanced sensor network technologies, be able to accommodate human comfort?

iv) What kind of architecture can accommodate and integrate these knowledge components? What challenges does this pose?

1.7.4 Research Goal and Objectives

The goal of this research is to explore and develop a novel WSN based embedded intelligent system for a comfortable living space and specifically to manage integrated, rather than discrete comfort system. In order to achieve that goal, this study investigated, and developed a working WSN concept based on a distributed embedded Fuzzy Rule Based System (FRBS).

Specifically, the research included the following main objectives:

i) To investigate and analyse the problem as it relates to individual comfort factors and human comfort. Even though plenty of separate individual solutions / models have been researched and developed, there are few systems that try to integrate a multiple of comfort factors into a single solution. Herein, the requirement for human comfort and attempts at maximising the role of the sensor in accommodating for a relevant intelligent system for a human comfort solution is analysed.
ii) To design and develop a WSN based human comfort ambience intelligent system architecture and framework.
iii) To design and develop an intelligent FRBS engine WSN with low computational cost for individual comfort factors and to analyse its performance in different scenarios. A generic FRBS engine was built that can be tailored to individual comfort needs.
iv) To design and develop a WSN based thermal comfort engine.

Alongside the main objective stated, the secondary research objective was to:

i) Design and develop a spatial sensing WSN based engine.
ii) Develop a simple and light weight application layer communication protocol that serves as a communication backbone among WSN nodes.

The secondary objective serves to assist components in delivering the main objective within the HCAmI System.

In summary, the ultimate objective of this study was to explore and develop a new WSN based solution to be used for human comfort activity in a living space that improves living conditions and eventually leads to improvement in the overall quality of life.

1.7.5 Research Scope

This study is limited to simple prototype implementation and uses a laboratory as a test bed. Due to Sun SPOT sensor node physical limitations, certain types of sensor could not be connected, such as gas sensor for detecting CO_2 and a sound level meter. Hence we acquired the data separately, using specific data collection equipment, and fed the values into the sensor node manually. We also manually determined the value of certain parameters due to lack of a suitable sensor, such as for determining metabolic rate and clothing level of a person.

For development and testing, the focus was only on sensor cloud activity, as shown in Figure 1-3.

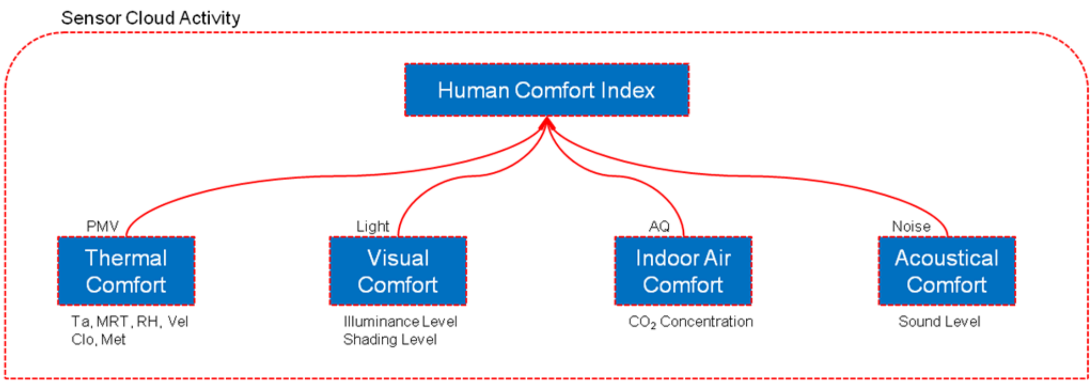


Figure 1-3. HCAmI Sensor Cloud Activity

In this study, we laid out a pioneering architecture and framework for human comfort activity within the WSN.

1.8 Thesis Outline and Contributions

The structure of the thesis is outlined as follows:

Chapter 1 Reviews current states in the WSN, Ambient Intelligence, Smart Home and embedded intelligent domains. Outlines brief history of said domains that led to the proliferation of multiple domain problems and solutions. Finally, an entire outline of this thesis is presented.

Chapter 2 A summaries of the literature and related work is reviewed, in order to establish the theoretical underpinning of the problem under consideration.

Chapter 3 Summarises the environment and tools used in modelling, prototyping and testing the ideology developed for this thesis.

Chapter 4 This chapter is where an ideology / solution to the identified problem is proposed. This chapter discusses in detail the theoretical architecture and framework of the HCAmI System. HCAmI components are presented and discussed. The ideology for the identified problem's solution is proposed.

Chapter 5 This chapter describes implementation of the proposed solution. The modelling, simulation and implementation process for each HCAmI component / module are discussed in detail.

Chapter 6 This chapter outlines the findings, conclusion and summary of the proposed solution. The HCAmI System was subjected to a series of simulated data and real life data from a testbed. The effectiveness of each component is presented. HCAmI System effectiveness in its entirety is established.

Chapter 7 Outlines the findings summary, conclusions, and recommendations for further work in the area of WSN and human comfort.

This study has yielded the three publications as listed in Appendix A Published Papers.

1.9 Chapter Summary

This chapter presented an introduction to WSN embedded and ambient intelligence. It also discussed the key human comfort factors like thermal comfort, visual comfort, indoor air comfort and acoustical comfort.

The chapter also described the background of the research, focusing on the research context, where problem statement, motivation as well as the research questions, goal and scope were highlighted.

Finally, the chapter provides an outline to the thesis and a list of contributions.

Chapter 2 WSN and Human Comfort Related Literature

2.1 Introduction

This chapter thoroughly explores the WSN, human comfort and AmI knowledge domains. Interrelationships among these areas and a knowledge gap are also presented.

A number of separate studies have been conducted in various disciplines that focused on specific issues and challenges associated with smart living space, ubiquitous computing and environmental comfort [32-40]. These studies were done in their own specific domains and focus on solving specific domain issues such as fuzzy based thermal comfort systems, climate comfort in passive office buildings, real-time thermal comfort systems and many more. Consequently, little attention has been paid to combining the various multidisciplines involved.

Within the smart living space and ubiquitous computing community, a number of studies have focused on using WSN as a tool to recognise human activity based on interaction with everyday objects [39, 41].

On the other hand, numerous works from the smart house and environmental comfort community concentrate on building a high efficiency low energy living space [33, 34, 36, 38]. A myriad of low energy building approaches have been explored. Yet, in all these high tech building approaches, they still use conventional separate / individual wired monitoring and management equipment such as conventional Heating, Ventilation, and Air-Conditioning (HVAC) systems, conventional power / utility loggers and manual adjustment of living space fixtures such as opening / closing a blind.

Environmental quality in a living space refers to the provision of visual comfort, acoustical comfort, thermal comfort and acceptable indoor air quality for its occupant [42]. Hence, the evaluation of human comfort in a living space may be broken down into relevant comfort factors, to achieve the desired comfort level.

In essence, this chapter explores what has been done in the area of human comfort from the perspective of knowledge that supports AmI.

2.2 WSN in Everyday Life

An Advance Wireless Sensor Network for Health Monitoring is an architecture for smart healthcare based on an advanced Wireless Sensor Network [43]. It specifically targets assisted-living residents and others who may benefit from continuous, remote health monitoring. The advantages, objectives, and status of the design were assessed. An experimental living space has been constructed at the Department of Computer Science at UVA for evaluation, with early results showing a strong potential for WSNs to open new research perspectives for low-cost, ad hoc deployment of multimodal sensors for an improved quality of medical care. The most prominent benefits of this work were: portability and unobtrusive ease of deployment and scalability, real-time and always-on, and self-reconfiguration and self-organisation. From the end user point of view, the system architecture provides seamless integration into their work environment.

Due to rapid advances in networking and sensing technology, we are witnessing a growing interest in sensor networks, in which a variety of sensors are connected to each other and to computational devices capable of multimodal signal processing and data analysis [44]. The network plays a significant role as key enabler in emerging pervasive computing technologies. Many initiatives have been put together in this domain, such as the European FP6-IPs Multi-modal Services for Meetings and Communications, Cognitive Robot Companion (COGNIRON), Context Aware Vision using Image-based Active Recognition (CAVIAR), Ambient Intelligence for the Networked Home Environment (AMIGO), Human-Computer Interfaces Similar to Human-Human Communication (SIMILAR), and Ambient Intelligence for Mobile Communications through Wireless Sensor Networks (e-SENSE), as well as MIT's Project Oxygen, MERL's Ambient Intelligence for Better Buildings and Georgia Tech Aware Home [44]. Every one of these projects focuses on various aspects of the AmI experience through extensive use of the WSN. Home automation with optimal living comfort and minimal energy cost, smart meeting rooms that accommodate occupant needs and smart beds that unobtrusively monitor vital signs of the sleeper are among the traits of AmI that being researched. A series of time based data streams collected from various sensors can be mined for some other hidden traits such as temporal patterns, behaviour and rules that might indicate a cross relation / correlation between recorded parameters.

In the field of object or person location sensing, an inexpensive and robust wireless localisation network that can track the location of patients in an indoor environment and monitor their physical status that is, walking, running, etc. were developed [45]. Multiple static nodes are placed at a known predetermined spot. These static nodes are used to calculate and determine the location of user's a mobile node. The mobile node at the same time monitors the motion state of the carrier, based on an onboard three-axis accelerometer sensor. The Delaunay Triangulation geometric method is used to determine the region the mobile node is located. Further enhancement of the work may be developed for three dimensional spaces and simultaneous multi mobile node tracking.

Over the years, security has become more important and is being talked about more frequently. The Swarming Sensor Network was introduced as a method for detecting the presence of an intruder and provides localisation and tracking of the intruder via monitoring the swarm behaviour of the sensor nodes on the network. The network is based on simple signals and cueing, with no data protocols to minimise cost and complexity [46]. It uses a special sensor that detects the blockage of illumination signals by the intruder. The works mimic a swarm of organism behaviour when they discover something interesting such as food, water or enemy, thus alerting neighbouring nodes, consequently waking up the neighbouring nodes to join the swarm.

Another usage of the WSN is in terms of providing context-aware services for mobile users by accessing surrounding WSNs. The CONSORTS-S platform provides the following facilities:

i) Communicating to wireless sensor networks via a mobile sensor router attached to a user's mobile phone,
ii) Analysing the sensed data derived from networks by cooperating with sensor middleware on a remote server to capture one's contexts, and
iii) Providing context aware services for mobile users of cellular telephones.

The platform is designed to process the sensed data effectively through cooperation between a mobile phone, a mobile sensor router, and a sensor middleware on a remote server [47].

In essence, WSN can be used extensively by putting together various subsystems to form a complete solution for a problem domain.

2.3 Embedded Intelligence

AmI pursues a grand vision that small, networked computers will jointly perform tasks that create the illusion of an intelligent environment. One of the most pressing challenges in this context is the question how one could easily develop software for such highly complex, but resource-scarce systems [48]. With the complexity and multi-platform hardware, embedding intelligence into these platforms proves to be challenging and complicated for the system developer. One has to code the intelligence using various non-compatible development platforms. On the other hand, Java pledges to leverage this predicament by offering write once runs anywhere in the development environment. AmbiComp is one of the works that make use of the Java offering. It is a platform for distributed execution of Java programs on embedded systems by offering a single system image. It consists of small, modular hardware, a flexible firmware including a Java Virtual Machine, and an Eclipse-based integrated development environment [48]. Its goal is to provide a distributed Java Virtual Machine (VM) that runs on bare sensor node hardware where Objects and threads can migrate freely between these nodes [49]. With ease of development, intelligence can be embedded into various platforms effortlessly.

With myriads of distributed sensors, it is not easy to modify or upgrade their software to suit the ever changing needs of the system. One possible way is by unsupervised learning and control that can adapt to new needs in the environment. Modern approaches to the architecture of living and working environments emphasise the dynamic reconfiguration of space and function to meet the needs, comfort, and preferences of inhabitants [50]. By employing autonomous intelligence, this can satisfy the needs of inhabitants without human intervention [50] through a multiagent approach, where multiple agents control subparts of the environment using fuzzy rules that link sensors and effectors. Decisions are taken based on sets of dynamic fuzzy rules that represent the knowledge of the system. Other form of Artificial Intelligent (AI), such as genetic algorithms and rules based approaches may also be incorporated as optimisation and decision making activities inside the system. COllective INtelligence

(COIN) is an example of a macro-learning paradigm to steer the system towards global optimisation with improved performance and minimal communication overheads [51].

In the smart home environment, the integration of everyday objects with information processing has been extensively researched. The object changes state based on the user's activities and situation. Activity recognition is predicted based on user interaction with the everyday object. Each object will track the state changes produced based on the user's interaction with them, and such information is used in recognising the user's activities and situation [40]. With the introduction of a more powerful energy efficient microcontroller, more computational intelligence can be put forward closer to the lower sensing level, hence reducing the needs of PC or server based intelligence. Everyday objects literally become smarter.

AmI can be summarised by the figure shown below:

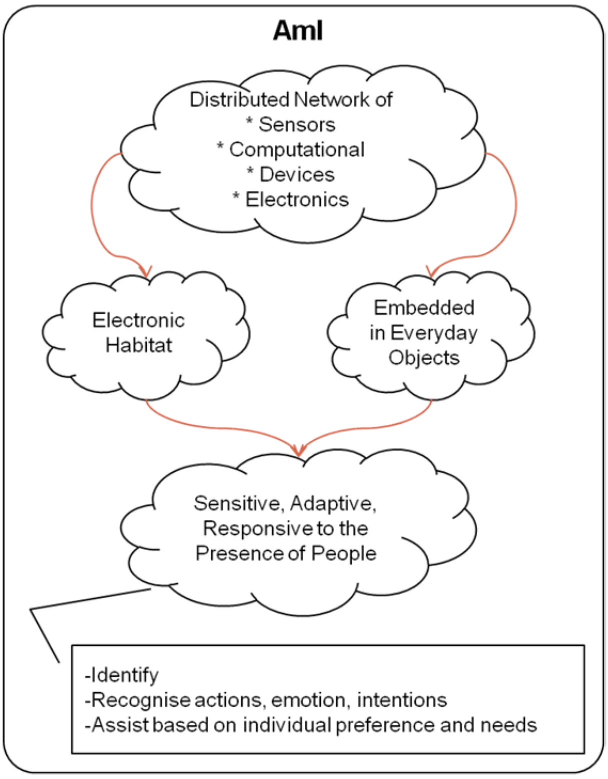


Figure 2-1. Ambient Intelligence

From Figure 2-1, AmI can be expressed as distributed networks of sensors, computational entities and devices within an electronic habitat and might be embedded in everyday objects that are sensitive, adaptive and responsive to the presence of people. It is a decentralised system where every object has its own task and responsibility for accommodating human needs and requirements. AmI can identify and recognise actions, emotions and intentions where it will provide assistance based on individual preference and needs.

2.4 Human Comfort

Human comfort is a collective sensory experience for a human where he / she feels pleasant and at ease with the surroundings. Human comfort can be described as a state of mind that expresses satisfaction with the surrounding environment / living space [52, 53]. It encompasses thermal comfort, visual comfort, indoor air comfort and acoustical comfort as shown in Figure 2-2.

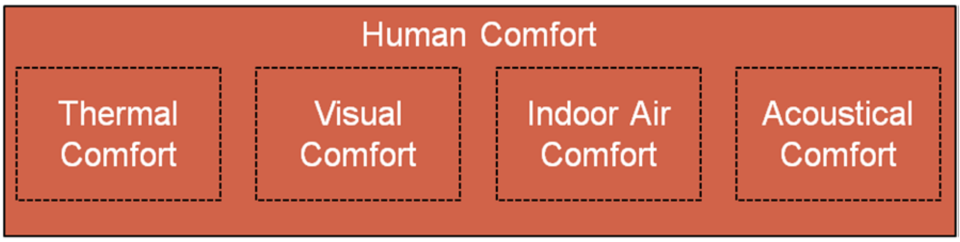


Figure 2-2. Human Comfort

To achieve the goal of human comfort, a significant amount of funding is allocated by various government establishments to research designing and constructing environmentally comfortable living space. A large portion of this fund is dedicated to research related to thermal comfort alone.

2.4.1 Thermal Comfort

Thermal comfort is a condition of mind, that expresses satisfaction with the thermal environment [22, 23, 27]. Human thermal sensation is mainly related to the thermal balance of the body as a whole. This balance is influenced by two groups of factors. These are environmental factors and physiological factors.

Environmental factors consist of the following:

i) Dry Bulb Temperature (DBT) – ambient air temperature measured by a thermometer freely exposed to the air but shielded from moisture and radiation.

ii) Mean Radiant Temperature (MRT) – the area weighted mean temperature of all objects surrounding the body.

iii) Relative Humidity (RH) – ratio between absolute humidity of the air compared with the maximum amount of water the air can hold at the measured temperature.

iv) Air Movement (Vel) – air movement in proximity to a person involved, measured using an anemometer.

Physiological factors consist of two factors:

i) Metabolic Rate (Met) – human body heat production based on various activity levels.

ii) Clothing Level (Clo) – insulation of clothes where 0 Clo corresponds to a naked person and 1 Clo corresponds to a person wearing a typical business suit.

When these factors have been estimated or measured, the thermal sensation for the body as a whole can be predicted by calculating the Predicted Mean Vote (PMV). The PMV equation for thermal comfort is a steady-state model. It is an empirical equation for predicting the average vote of a large number of people on a 7 point scale (-3 to +3) of thermal comfort. PMV values can be calculated as Equation 2–1 [23]:

$$PMV = (0.303e^{-0.036M} + 0.028)\left\{\begin{matrix} (M-W) \\ -3.05\times 10^{-3}[5733 - 6.99(M-W) - P_a] \\ -0.42[(M-W) - 58.15] \\ -1.7\times 10^{-5}M(5867 - P_a) \\ -0.0014M(34 - t_a) \\ -3.96\times 10^{-8} f_{cl}[(t_{cl}+273)^4 - (t_r+273)^4] \\ -f_{cl}h_c(t_{cl} - t_a) \end{matrix}\right\} \quad \text{Equation 2–1}$$

Where:

$$t_{cl} = 35.7 - 0.028(M-W) - I_{cl}\{3.96\times 10^{-8} f_{cl}[(t_{cl}+273)^4 - (t_r+273)^4] + f_{cl}h_c(t_{cl} - t_a)\}$$

$$h_c = \begin{cases} 2.38(t_{cl}-t_a)^{0.25}\ for\ 2.38(t_{cl}-t_a)^{0.25} > 12.1\sqrt{V_a} \\ 12.1\sqrt{V_a}\ for\ 2.38(t_{cl}-t_a)^{0.25} \leq 12.1\sqrt{V_a} \end{cases}$$

$$f_{cl} = \begin{cases} 1.00 + 1.29I_{cl}\ for\ I_{cl} \leq 0.078m^{2o}C/W \\ 1.05 + 0.645I_{cl}\ for\ I_{cl} > 0.078m^{2o}C/W \end{cases}$$

The parameters are defined as follows:

PMV Predicted Mean Vote

M metabolism / metabolic rate (W/m^2)

W external work / effective mechanical power, equal to zero for most activity (W/m^2)

I_{cl} thermal resistance of clothing / clothing insulation (m2 · K/W)

fcl ratio of body's surface area when fully clothed to body's surface area when nude

t_a air temperature (°C)

t_r mean radiant temperature (°C)

V_a relative air velocity (m/s)

P_a partial water vapour pressure (Pa)

h_c convectional heat transfer coefficient (W/(m2 · K))

t_{cl} surface temperature of clothing (°C)

NOTE

1 metabolic unit = 1 met = 58.2 W/m^2 (356 Btu/h);

1 clothing unit = 1 clo = 0.155 $m^2 \cdot$ °C/W

h_c and t_{cl} may be solved by iteration

Table 2-1 summarises common activity metabolic rate that can be used in PMV calculation.

Activity	Metabolic rate (met)
Reclining	0.8
Seated, relaxed	1.0
Sedentary activity (office, school, lecture)	1.2
Standing and light activity (shopping, light work)	1.6
Standing and medium activity (cashier, machine operation, domestic chores)	2.0

Walking on level ground 2km/h 3km/h 4km/h 5km/h	 1.9 2.4 2.8 3.4
Playing volleyball	4.0
Bicycling, Softball, Golf	5.0
Gymnastics	5.5
Basketball, Swimming, Aerobic Dancing	6.0
Backpacking, skiing on level good snow at 9km/h	7.0
Playing handball, hockey, soccer	8.0
Forestry – working with an axe (33 blows / min and weight 2kg)	8.5
Running at 15km/h	9.5

Table 2-1. Metabolic Rates

Table 2-2 summarises common clothing that can be used in PMV calculation.

Clothing	Clothing Level (Clo)
Nude	0
Underwear – pants Pantyhose Panties Briefs	 0.02 0.03 0.04
Underwear – shirts Bra Shirt sleeveless T-shirt	 0.01 0.06 0.09
Shirts Short sleeve Light blouse with long sleeves Normal long sleeves Turtle neck long sleeves	 0.09 0.15 0.25 0.34
Trousers Shorts Light trousers Normal trousers Overalls	 0.06 0.20 0.25 0.28

Sweaters	
Vest	0.12
Thin sweater	0.20
Thick sweater	0.35
Jacket	
Vest	0.13
Summer jacket	0.25
Normal jacket	0.35
Coats and overjacket	
Down jacket	0.55
Coat	0.60
Sleepwear	
Under shorts	0.10
Short gown with thin straps	0.15
Long gown with long sleeves	0.30
Long pyjamas with long sleeves	0.50

Table 2-2. Clothing Insulation Value

For multilayer clothing, Clo value can be calculated by simply adding up the Clo value for each individual garment worn by the person.

PMV will provide an index of comfort / discomfort levels taking all six above-mentioned parameters as shown by Figure 2-3. In order to ensure a comfortable thermal comfort, ISO7730 recommends maintaining the PMV at 0 with a tolerance of 0.5. There are seven levels of the PMV ranging from -3 to +3, which represent cold and hot respectively. When PMV is equal to zero, it implies a neutral and comfortable state.

Due to differences among individuals, it is not feasible to identify a thermal comfort that satisfies everybody. There will always be a number of unsatisfied dwellers. But it is feasible to specify an environment that is acceptable to a certain percentage of dwellers.

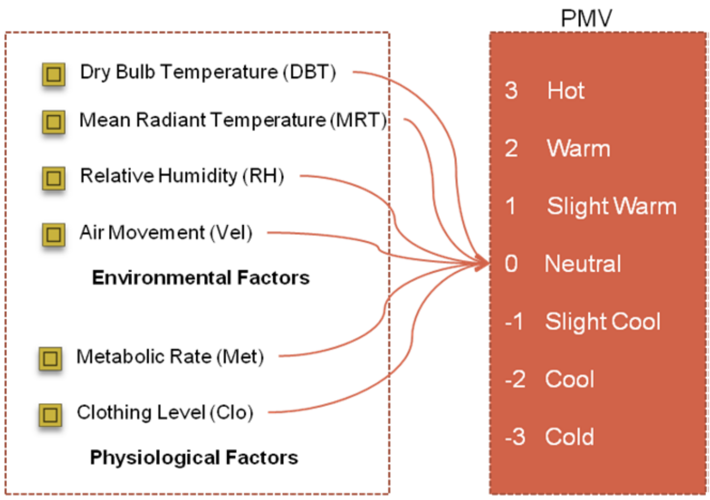


Figure 2-3. PMV and Thermal Sensation

Countless works have been published regarding maintaining a high degree of thermal comfort yet minimising the energy consumption of living spaces. Fuzzy logic is the most widely used method of evaluating control strategies for the adjustment and preservation of air quality, thermal and visual comfort for occupant while, energy consumption reduction is achieved [54-56]. The fuzzy engine is used as the brain behind a building's HVAC and air conditioning unit where it gives better control than conventional PID [32].

Other than fuzzy logic, a neural network also is used as the intelligence behind HVAC and air conditioning. The system was trained using data from a thermal comfort survey and the PMV index is determined via neural computing, where a neural network evaluation model of indoor thermal comfort is established based on a back propagation algorithm [57].

Kemajou et al. [58] recently proposed a method for predicting indoor air temperatures in modern buildings using an Artificial Neural Network (ANN). The work predicts indoor air temperature in modern buildings, seven hours in advance in humid regions, using only the outdoor air temperature and the last six hourly values of the indoor air temperature as input.

2.4.2 Visual Comfort

The human eye is a very complex organ that reacts to light from the surrounding environment. It detects light and converts it into electro-chemical impulses in neurons. The signals then are transmitted to the brain through complex neural pathways that connect the eye via the optic nerve to the visual cortex region of the brain. The brain will then interpret it into meaningful information. It allows us to see the real world. The sense of incoming light serves as the basis of our visual comfort.

Visual comfort literally signifies the absence of physiological pain, irritation or distraction to our vision. Visual comfort inside a particular space depends on the contrast levels and luminance variations across a space [59]. Glare is the prominent cause of visual discomfort and can influence an occupant to react to the lighting system. The reaction can be as simple as turning away from the glare source, to the modification of lightning control properties within the space. The activity can significantly impact the energy expenditure of the space.

Visual comfort and visibility are dependent upon the luminance (energy reflected from the surfaces) pattern within the visual field. Luminance variation (luminance ratios) within the visual field reflects the level of visual comfort. Visual comfort calculations are intrinsically complex to perform, as they depend upon the location and brightness of light sources and also the apparent size of the light sources, as seen from a particular viewpoint.

In this work, the problem domain was simplified to suit WSN sensing capability. One can articulate that visual efficiency corresponds to visual comfort, hence deduce the recommended lighting level for a range of common tasks and environments. Table 2-3 summarises minimum lighting levels required based on the Illuminating Engineers Society (IES) recommendations [60].

Lighting Level (Lux)	Activity	Tasks Area and Environments
50	Background light	Car Parks Store rooms Hotel bedrooms Garages
100	Casual seeing	Corridors and passageways Changing rooms Rest rooms Foyers Domestic living rooms

200	Continuously occupied	Lifts and lift lobbies Vaults and strong-rooms Shopping centre circulation areas Airport lounges School assembly halls Lecture theatres
400	Visual task moderately difficult	Enquiry desks and counters Food preparation areas General clerical offices Library reading tables Classroom white-boards Laboratories
600	Visual task difficult	Engine testing rooms Computer rooms Food sales counters Cashier counters Supermarkets School art rooms
900	Visual task very difficult	Electronics assembly areas Instrumentation workbenches Supermarket displays
1200+	Visual task extremely difficult	Sorting and grading areas Clothing inspection areas Hand engraving workbenches Jewellery workbenches

Table 2-3. Minimum Illuminance Levels

At its simplest, different lighting levels are required for different types of activities. The illuminance needed depends on how much detail needs to be seen, the age of the occupant, and the speed and accuracy of the task being done [61]. The more detailed the chore, the greater the light requirement.

2.4.3 Indoor Air Comfort

Indoor air quality is one of the most important aspects of human comfort. Occupants wish for healthy and comfortable living conditions. If the air quality is not up to expectations, satisfaction, productivity and health may be compromised. Good indoor air quality leads to good indoor air comfort. Indoor air quality issues tend to be complex and most indoor air quality problems are the result of:

i) Poor ventilation in a building, such as ventilation system systems not being designed for the right level of occupancy or human activities happening inside it.

ii) A poorly maintained ventilation system. The ducting, ventilation blades and filters not regularly serviced.

iii) Poorly located fresh air intakes that allow outside fumes or other pollutants to contaminate the inside of a building.

iv) Lack of ventilation causing indoor mould growth, hence causing serious health issues.

v) An inadequate local ventilator to move contaminants from indoor to outside, such as lack of a range hood in the stove area.

vi) Last but not least, it might be poorly designed building air tightness, causing not enough outside air to be introduced into the building

Effects of poor indoor air quality may include immediate health linked problems such as headaches, dizziness, nausea, fatigue, difficulty in concentrating, sinus problems, congestion, eye irritation, nose and throat problems, and coughing [25, 26]. Such immediate effects are usually short-term and treatable. Treatment can be as simple as letting in adequate fresh air from outside.

Other health effects may show up after prolonged and repeated bad air quality exposure, such as respiratory disease, heart disease and debilitating or fatal cancer. Therefore, it is sensible to try to improve the indoor air quality in our living space even though there are no noticeable symptoms present.

The most significant aspect in maintaining good indoor air quality is ensuring sufficient ventilation and minimising airborne contaminants in living space. Generally, for ventilation, carbon dioxide may be used as an indicator of air circulation, because its concentration is related to the number of persons / living beings in the space and the space's general ventilation rate [22]. As people / animals breathe, oxygen is inhaled and carbon dioxide is exhaled, thus causing CO_2 builds up. As the CO_2 concentration increases, the air gets stale and people will start to feel uncomfortable. People start to complain when CO_2 concentration reaches 800ppm. Some health issues arise when CO_2 exceeds 1000ppm as shown in Table 2-4 [22, 25, 26]. If the CO_2 level gets too high, it means that more fresh air from outside needs to be brought into the living space.

ASHRAE guidelines recommend that CO_2 concentration in occupied buildings should not exceed 1000ppm.

CO_2 Level (ppm)	Conditions / Sensations
350 – 450	Normal outdoor
< 600	Acceptable level
600 – 1000	Complaints of stiffness and odours
1000 – 2500	General drowsiness, headache, fatigue, and eye / throat irritation. Indicates inadequate ventilation.
2500 – 5000	Adverse health effects
> 30,000	Dangerous level, slight intoxicating, breathing and pulse increase, nausea
> 50,000	Slight intoxicating and slight impairment
100,000	Unconscious and death is a possibilities

Table 2-4. Typical CO_2 Level

Poor indoor air quality not only affects human life, but also effects living space and their furnishings. Excessive dust, mildew and mould are all symptoms of poor air quality that can eventually lead to health problems and costly building repairs.

To provide comfortable, clean and healthy air inside a living space, one must assure that the ventilation is up to expectation. Indoor air quality can be determined by sampling air properties at a particular moment, thus monitoring exposure to pollutants.

In this research we chose CO_2 as the prime factor in determining indoor air comfort [25, 26], since CO_2 is exhaled by occupants at predictable levels. It is well known that people emit CO_2 at a rate that depends on their size and their level of physical activity [62]. CO_2 in the air may be considered a significant indicator of air quality. Improvement in CO_2 indoor levels systematically leads to improvement of indoor air comfort. Table 2-5 shows a typical CO_2 recommendation by various organisations for particular living conditions. Current technology allows easy and relatively cheap CO_2 measurement, thus making it a workable solution as part of WSN based monitoring system.

Organisations	CO_2 Level (ppm)
National Institute for Occupational Safety and Health (NIOSH) for indoor CO_2 concentration	< 1000
American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE) for ambient levels	< 700
Occupational Safety and Health Administration (OSHA) in workplace	Limits 5000 ppm for prolonged periods 35,000 ppm for 15 minutes

Table 2-5. CO_2 Recommendations

Based on the EN 15251: Indoor Environmental Criteria standard [19], we can use increased CO_2 level as an indoor air quality indicator. Clearly, elevated indoor CO_2 levels suggest inadequate amounts of outside air being brought into a living space or building. Undesired CO_2 levels can be addressed by fairly simple and relatively inexpensive method such as using mechanical ventilation (bringing in fresh outside air) and better air circulation throughout the living space (minimising stale spots). Hence, these simple methods should make our home healthier and keep us more comfortable as well.

2.4.4 Acoustical Comfort

Acoustical comfort can be perceived as sound that we hear in relation to its level (intensity) and character (frequency). Is it too quiet or too loud? Disruptive sound can be a significant issue for a person engaged in normal, daily activities such as writing, reading, enjoying leisure time, watching a movie or sleeping. Unwanted noise can cause distraction, making it hard for the person to give full concentration to the task at hand, consequently causing stress, anxiety and reduced productivity. Excessive indoor noise can cause elevated stress and affect human wellbeing, impair productivity and lead to permanent health damage such as hearing loss. Therefore, a good acoustic environment is extremely important to maintain a high degree of satisfaction and a healthy acoustic environment.

The main elements that effect indoor acoustical comfort are indoor noise sources and outdoor noise sources. Indoor noise sources can be electronic instruments such as radios, televisions, kitchen appliances, whiteware equipment and electro mechanical housing fixtures such as a HVAC, air conditioning, heat pump or fan heater. Outdoor noise can come from nearby traffic, industry and natural noise such as rain, lightning and storms. Combined noise sources can have a great impact on indoor acoustical comfort. A good acoustic environment keeps noise at bay so that it does not interfere with activities.

Acoustical comfort depends on what kind of space we are looking at. Different spaces requires different requirements for acoustical comfort, whether it is intelligibility (a balance of absorption and attenuation so that the degree of speech can be understood), concentration (an undisturbed environment) or confidentiality (privacy) [63]. Choosing the right acoustic solutions will enhance the user's needs and eventually lead to higher overall comfort.

Another advantage of maintaining good acoustical comfort is from hearing wellbeing. Studies show that excessive exposure to high levels of noise / sound can cause severe hearing impairment. The effect might not be immediate, but can cause loss of hearing at later stages. Table 2-6 [63-66] shows typical noise levels and the damage caused.

Noise / Sound Level (dB)	Damage / negative effect
140	Adult hearing impairment
120	Child hearing impairment
85	Hearing impairment (daily for 1 hour)
70	Hearing impairment (daily for 24 hours)
55	Serious annoyance
35	Communication disturbance
30	Sleep disturbance
0	Comfort zone

Table 2-6. Noise Level and Damage

Noise disturbances can cause pathological effects such as temporary or permanent hearing loss. This might start with reduced auditory sensitivity, auditory pain and fatigue. Other effects might be physiological, causing distress, discomfort and annoyance.

When accessing acoustical conditions in a living space, the basic and most important aspect to check is the intelligibility of speech. It can be ruined by excessive background noise or long reverberation [63].

ISO 3382 [64] describes the measurement procedure for open plan offices and room acoustic parameters for objective evaluation. The evaluation consists of a variety of measures that give a general evaluation of the room. These parameters can be separated in two groups. The first group consists of parameters that can be measured easily (using omnidirectional microphone) such as Early Decay Time (EDT in seconds), Strength (G in dB), Clarity (C_{80} in dB), Definition (D_{50}) and Reverberation Time (T_{30} in second). The second group contains dual channel parameters such as Inter Aural Cross Correlation (IACC), Late Lateral Energy (LG), and Early Late Lateral Energy (LF and LFC). These parameters cannot be measured with a single omnidirectional microphone. The measurement parameters are too complex to be implemented in a WSN node equipped with a single sound level meter (measuring Strength). Hence, we had to simplify the acoustical comfort measurement as it serves as an indicator of acoustical comfort in the node area only. The parameter sound strength (G) is linked to the noise level as shown in Table 2-6. Here sound strength (Decibel level) was used as a measure of sound / noise level from a sound source. Decibels are most frequently used for measuring sound level (intensity). Humans perceive loudness as varied by frequency. Our ear can hear between 20 Hertz and 20,000 Hertz. We hear very well between 500 Hertz and 6,000 Hertz (normal human conversation).

The depth and complexity in acoustics can be seen in numerous research projects [65-71] where the most prominent measurement parameter is sound strength in Decibels. Various places and conditions were examined in determining acoustical comfort. Therefore, measuring acoustical comfort requires accurate measurement of all forms of sound. Sound level and frequency can be measured with a sound meter, as described in Chapter 3.

2.4.5 Human Comfort Summary

Most of these studies focused on embedding a thermal comfort engine into the environment controller itself, such as the HVAC system or air conditioning system. It lacks the perspective of a whole living space management solution.

In conclusion, Human Comfort performance measurement can be summarised as follows:

Comfort Factors	Measurement Parameters	Measurement Units, Range And Resolution
Thermal Comfort	Environmental Factors a) Dry Bulb Temperature (DBT), b) Mean Radiant Temperature (MRT), c) Relative Humidity (RH), and d) Air Movement (Vel). Physiological Factors a) Metabolic Rate (Met), and b) Clothing Level (Clo).	 a) °C, -10 to 50, 0.1°C b) °C, -10 to 50, 0.1°C c) %, 0 to 100, 1% d) m/s, 0 to 10, 1m/s a) Met, 0 to 10, 0.1 Met b) Clo, 0 to 2, 0.1 Clo
Visual Comfort	a) Ambient and task levels: artificial and daylight, b) Contrast, brightness ratio (glare), and c) Colour rendition	Lux, 0 to 1500, 1 Lux
Indoor Air Comfort	a) Ventilation rate: fresh air supply, circulation, and b) Mass pollution: gasses, vapours, microorganisms, fumes, smoke and dust	ppm, 350 to50,000, 350ppm
Acoustical Comfort	a) Sound pressure level and frequency b) Reverberation and absorption c) Speech privacy, articulation index, and d) Vibration	dB, 30 to 130, 1.5dB

Table 2-7. Human Comfort Sumary

Table 2-7 summarises measurement parameters that serves as a boundary in this study. Selected measurement parameters will be taken into account. Due to technical issues, only selected measurement sensors were chosen to be linked with WSN nodes as discussed in Chapter 3.

2.5 Human Comfort: Environmental Control systems

Research shows strong links between a comfortable living environment and human productivity. Uncomfortable people are less productive, and in the long run, it could be costly from an economic point of view. Measuring the environment is easy, but quantifying human comfort is not. Human comfort is notoriously tricky and difficult to measure; hence most systems simply use typical environment measurements and adjustments such as temperature, humidity, light luminosity and air quality. Human comfort is, to some extent, subjective, and there are many comfort factors that, influence individual perceptions of comfort such as thermal comfort, visual comfort, indoor air comfort and acoustical comfort.

To provide a comfortable and healthy place for people to live is one of the key factors in building design and construction. Hence, a human comfort environmental control system's primary goal is to monitor and maintain acceptable human comfort levels, and at the same time cater for the secondary objective of optimising energy use. For example, most important to human comfort is thermal comfort. Is 17.5°C good enough for a comfortable living? Should the temperature be increased? What is the caveat? How can it be achieve? Does the action have any impact on other comfort factors as well?

Given our enthusiasm for having ideal cosy homes, complex relationships between the aforementioned comfort factors have to be looked at in great detail. The delicate assessment balance between these comfort factors has been examined and modelled as a hybrid fuzzy reasoning expert system, marked as human comfort index. In a given living space, a group of WSNs will sense the environment (temperature, humidity, clothing, metabolic rate, wind speed, luminosity, etc.), make sense of raw information and, by means of fuzzy logic, weighting and integration criteria, measures the human comfort level.

Figure 2-4 illustrates the human comfort activity of a living space through the use of WSN and an embedded fuzzy logic approach. It shows each comfort component with connected sensors and the weighted average method for overall human comfort handled by the WSN node.

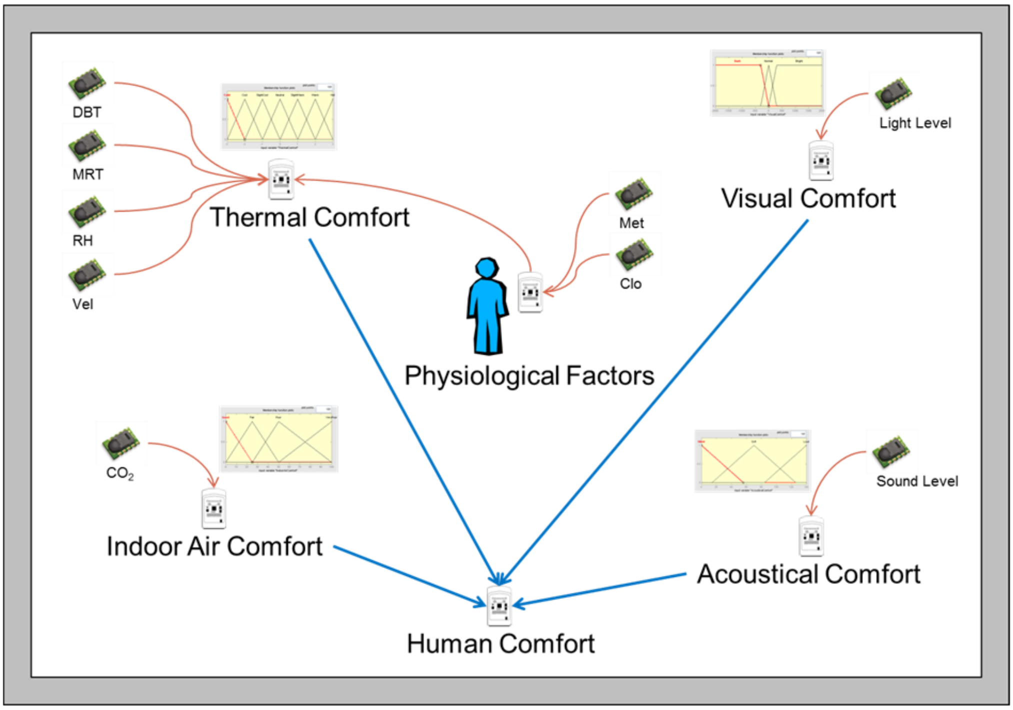


Figure 2-4. Human Comfort Environment

The following literature review indicates that there is currently no framework or architecture that can fully cater for human comfort activity. Specifically, none of the solutions is able to quantitatively combine multi comfort factors when predicting the human comfort level of a monitored living space.

For this study, a number of related areas of research were thoroughly reviewed, including the wireless sensor network, embedded intelligence, ambient intelligence and the smart environment and low energy home. Most recent research and development related to the work has been visualised and summarised in Figure 2-5.

The visualisation of related key literature is important in order to place this study in the right context. This section provides a review of the key literature involved, from a HCAmI System perspective.

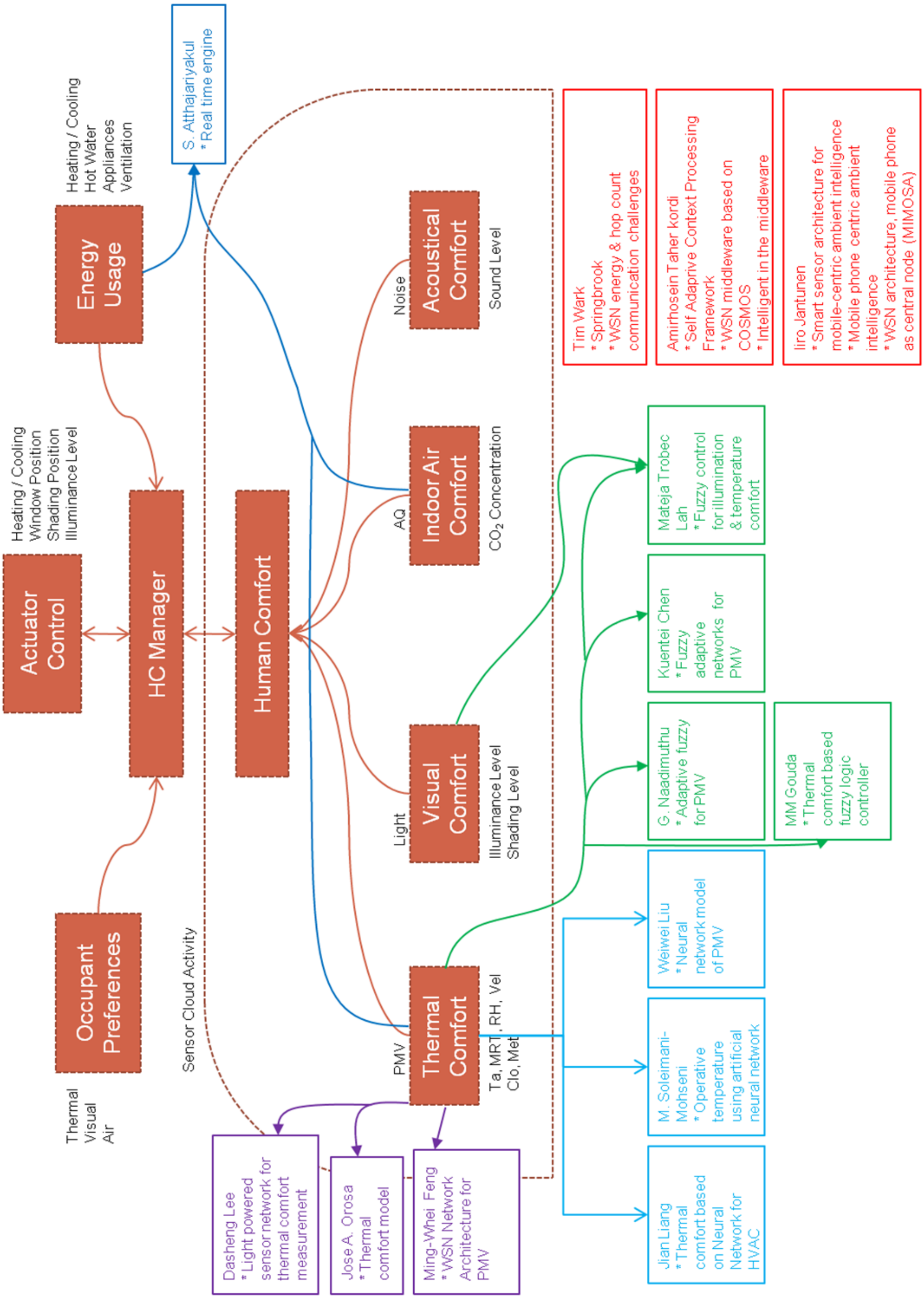


Figure 2-5. HCAmI and Existing Works

2.5.1 WSN and Thermal Comfort

Orosa et al. [72] scrutinises in great detail various thermal comfort models based on ISO 7730 standards and ASRAE Standards. Coupled with scientific research, they present both deterministic and empirical models for application to building design and environmental engineering. Feng et al. [39] proposed a network infrastructure by integrating the TCP/IP network with the ZigBee network in respect to thermal comfort activity for an ubiquitous smart living space application. They provide a WSN based algorithm to coordinate smart-skin equipment and air conditioners in order to improve thermal comfort. Network optimisation is achieved by means of clustering huge networks into smaller chunks. Also in a WSN hardware context, Lee et al. [73] proposed light powered sensor networks that were able to gather indoor thermal comfort information. The information then is used in air conditioning systems by implementing a comfort-optimal control strategy. The sensing node integrates an IC-based temperature sensor, a radiation thermometer, a relative humidity sensor, a micro machined flow sensor and a microprocessor for Predicted Mean Vote (PMV) calculation. The 935 MHz band RF module was employed for the wireless data communication, with a specific protocol based on a special energy beacon enabled mode, capable of achieving zero power consumption during the inactive periods of the nodes. A 5W spotlight, with a dual axis tilt platform, can power the distributed nodes over a distance of up to five metres. A special algorithm, the maximum entropy method, was developed to estimate the sensing quantity of climate parameters if the communication module did not receive any response from the distributed nodes within a certain time limit.

The works reviewed are highlighted in purple in Figure 2-5.

2.5.2 Real Time HVAC Engine

Atthajariyakul et al. [37] proposed an alternative methodology to deal with real time determination of optimal indoor air conditions for Heating, Ventilation and Air Conditioning (HVAC) systems in order to achieve the overall requirements of the system. PMV, CO_2 and the cooling / heating load are the input parameters that indicate thermal comfort, indoor air quality and energy consumption respectively. A real time gradient-based technique is used in order to yield optimal indoor air conditions for the

HVAC system. The performance index of the HVAC system is defined by the summation of square errors between each parameter indices and the desired ones.

Mirinejad et al. [74] introduce intelligent controllers (fuzzy logic based and neural network based) that are based on the sensation of thermal comfort. Direct comparisons were done against various HVAC controllers such as traditional controllers (on / off and PID) and, advanced controllers (auto tuning PID, modern and non-linear controllers, and optimal controller). From the varieties of HCAV controller reviewed, it can be concluded that intelligent controllers perform better, more reliable and are more energy efficient.

The works reviewed are highlighted in dark blue in Figure 2-5.

2.5.3 Thermal Comfort and Artificial Neural Network

Liang et al. [75] describe the design of a direct neural network (NN) thermal comfort controller, based on a back-propagation algorithm for indoor thermal regulation, with PMV as the control objective and six variables as input data. Coupled with an energy saving strategy (minimum-power strategy), a thermal space model for Variable-Air-Volume (VAV) application was developed and simulated. This model excludes two personal-dependent variables: Metabolic Rate (Met) and the Clothing Level (Clo), which is involved in the PMV calculation. Hence, they were set as a constant with respect to the current season. However, Soleimani-Mohseni et al. [76] experimented with two different models (linear and non-linear artificial neural network estimation models) for estimating the operative temperature in rooms and buildings. The operative temperature can be estimated fairly well by using variables that can be measured easily, such as the indoor and outdoor temperatures, the electrical power use in the room, the wall temperatures, ventilation flow rates and time of day. The downside of this approach is that the optimisation of neural networks often results in different networks, dependent on the initial values of the synaptic weights. Therefore, the result will, in general, not be the same in two different trials even if the same training examples have been used. Liu et al. [57] recently worked with an evaluation model for individual thermal comfort based at the back propagation neural network. The training data came from a thermal comfort survey where the evaluation results showed a respectable match with the subject's real thermal sensations. The result can be used to evaluate individual thermal comfort correctly. The work requires further enhancement to include various different combinations of environmental parameters.

The works reviewed are highlighted in blue in Figure 2-5.

2.5.4 Thermal Comfort and Fuzzy Logic

Gouda et al. [55] proposed a PMV-based fuzzy logic controller to evaluate the PMV level and used a linguistic description of the thermal comfort sensation for ease of use. The controller uses Mamdani's minimum operator method as an inference engine. The controller shows better performance and gives better control of tracking and robustness compared with traditional PID-based comfort controllers. Meanwhile, Naadimuthu et al. [32] and Chen et al. [77] dealt with a fuzzy adaptive network (FAN) to model at thermal comfort system based on a real world experiment. Finally, Lah et al. [78] worked with fuzzy control for thermal and visual comfort. The work combined two comfort parameters to strike the right balance in harmonising the thermal and optical behaviour of a building with regulated energy flows throughout the space. Different control strategies for different seasons were proposed and tested.

The works reviewed are highlighted in green in Figure 2-5.

2.5.5 WSN Communication, Architecture and Framework

Wark et al. [79] describe the design, development and findings from the first phase of a rainforest ecological sensor network. WSN energy and hop count is the main focus in providing reliable and long-term monitoring of the rainforest ecosystem. An extensive analysis around energy and communication challenges was discussed. In order to continuously observe the environmental context over a long period, the sensor node should be considered a context-aware device having particular contextual parameters, such as residual energy or sample rate [38]. Taherkordi et al. [38] proposed an approach for modelling sensor network context information, and a middleware framework based on COSMOS that maps the context model to software components. The sensor application is able to adapt itself to the current situation in the environment through the execution of a high-level context model. A high degree of intelligence is embedded into the middleware itself. Work proposed by Jantunen et al. [80] deals with an open architecture platform for implementing mobile-phone-centric ambient intelligence. A mobile phone acts as a central node, hosting applications and connecting a local sensor network to back-end servers. The architecture includes a context awareness layer that abstracts sensor measurements into context atoms through rule-based reasoning and notifies changes in atoms to local and remote applications. The technologies consist of a Simple Sensor Interface (SSI) protocol, nanoIP and low-power short-range radios.

From the aforementioned works, it can be seen that:

i) Comfort factor research has been done in isolation. Most work focuses on a specific comfort factor and does not take into account other comfort factors in providing better environmental control and there is no comprehensive comfort factor activity being researched.

ii) No framework / solution exists that combines all comfort factors into one integrated human comfort solution.

iii) No attempt has ever been made to solve the problem of conflicting comfort needs in providing human comfort solutions.

All of this work independently researched – individual comfort factors, and to the best of our knowledge, fusing different comfort domains as one indicator remains largely unexplored territory.

Based on these findings, a novel system that attempts to address the gap in current solutions was proposed. The collaborative approach based on the use of WSNs presented in this thesis could be considered the first initiative that uses FRBS and incorporates multiple comfort factors. This approach allows the user to define the alliance among sensors by means of specific knowledge based sets (fuzzy sets, rules and variable) and presents the following advantages:

i) The collaborative design deals with uncertainty and ambiguity.

ii) It is able to map HCI throughout the living space without the need for sensor nodes in all places.

iii) Due to modular nature of the design, it is possible to include or remove any comfort factor as needed.

The work will be beneficial in providing better living conditions in particular living spaces where complex association between comfort factors will be dealt with accordingly.

The works reviewed are highlighted in red in Figure 2-5.

2.6 Chapter Summary

In this chapter, WSN in everyday life and embedded intelligence were presented and thoroughly explored. Various related literatures were presented and discussed.

Then, the domain of human comfort was discussed. Each comfort factor (thermal comfort, visual comfort, indoor air comfort and acoustical comfort) were thoroughly reviewed.

Following that, the related literature in regard to WSN, human comfort and environmental control systems was reviewed and analysed, leading to the identified gap in this research as summarised in Figure 2-5.

Chapter 3 Test Environment, Tools and Resources

3.1 Introduction

This chapter introduces a specific test environment, tools and resources that were utilised in this work to model, simulate and test the notion of WSN and human comfort. A small scale prototype system was constructed as proof of concept of these ideas. The specific environmental setup, tools and resources closely represented real life scenarios and replicated various measures from it, to obtain a sense of actuality and draw conclusions from data gained. Analysis of the gained data from the simulation provided accurate descriptions and may guide a later, real system development. The main test environment, tools and resources used in this research follow:

3.2 Software

The ideology presented in this thesis was brought to fruition by a number of proprietary and open source software / modelling tools. The software utilised in this research was involved directly and indirectly in all phases of the research. It is summarised below:

3.2.1 Simulation Tools

MATLAB, MATLAB Fuzzy Toolbox and Simulink

In this study, we used MATLAB, MATLAB Fuzzy Toolbox and Simulink as our main simulation tools. The MATLAB primary version used is MATLAB R2008a bundled with Java Virtual Machine (JVM) 1.6.0. The written MATLAB codes and Simulink models should be compatible with future versions of MATLAB.

MATLAB facilitates specific HCAmI component engine modelling, development and testing. We mainly used MATLAB code / script to test specific equations or algorithms, MATLAB Fuzzy Toolbox for the HCAmI FRBS comfort engine and Simulink for modelling and operating a simulation of a built engine. For example:

i) Fanger's PMV thermal comfort engine as shown in Equation 2–1. Calculation of PMV was defined as MATLAB function `[PMVOUT] = pmvf(PMVPAR)` in `pmvf.m`. `PMVPAR` is an n-by-7 matrix which contains the input parameters for calculating the PMV as shown below.

```
PMVPAR(:,1): metabolism [W/m^2]
PMVPAR(:,2): external work [W/m^2]
PMVPAR(:,3): radiant temperature [degree Celsius]
PMVPAR(:,4): air temperature [degree Celsius]
PMVPAR(:,5): relative humidity    [0 < Rh < 1]
PMVPAR(:,6): clothing [clo]
PMVPAR(:,7): air velocity [m/s]
```

`PMVOUT` is a matrix containing the calculated PMV values.

Example of `pmvf` function usage:

```
PMVVAR=   [58.2 0 20 20 0.5 1 0.2;
           58.2 0 20 20 0.5 1 0.3];
[PMVOUT]=pmvf(PMVVAR)
PMVOUT =
     -1.1337
     -1.2965
```

The MATLAB function was then incorporated into Simulink which was subjected to simulated and real data (offline sensor data) for validation.

ii) HCAmI FRBS comfort engine was used for TC, VC, IAC and AC. A number of engine designs and implementations were developed, modelled and tested using the MATLAB Fuzzy Toolbox as shown in Figure 3-1. The engine was modelled using the Mamdani type fuzzy inference system and a triangular-shaped membership function (trimf).

Varieties of HCAmI FRBS engine were designed, modelled and finalised based on membership function complexity, number of rules involved and overall output accuracy. This is to suit the WSN node processing capability that has the simplest membership function and fewest rules, yet produces acceptable accuracy outcome.

The engine was then incorporated into Simulink and subjected to simulated and real data (offline sensor data) for evaluation and validation.

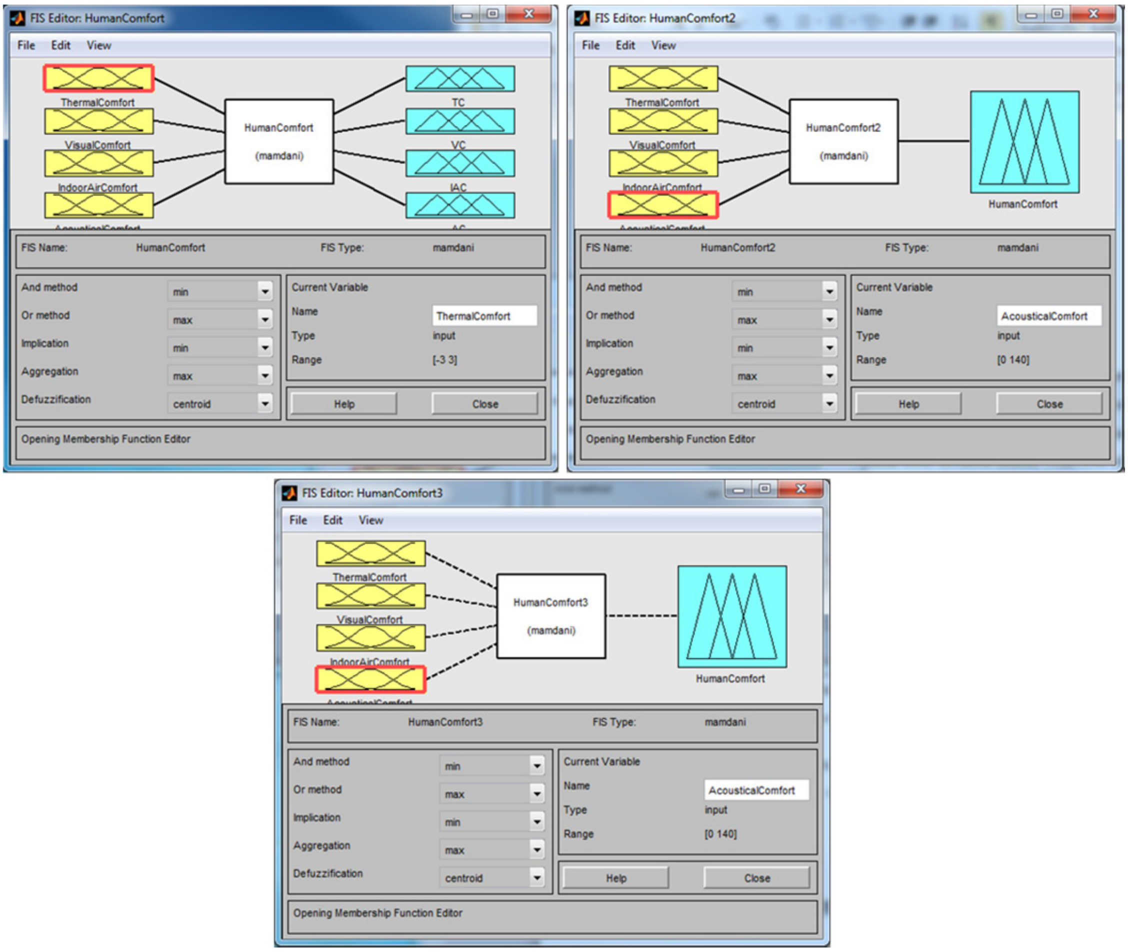


Figure 3-1. HCAmI FRBS Evolution

Solarium – Sun SPOT Emulator

An equally important simulation tool used in this study was the Solarium – Sun SPOT emulator. The Sun SPOT WSN platform comes with a complete java based software development kit called Sun SPOT Manager Tool [81]. The Sun SPOT Manager Tool is a Java WebStart Application that is installed from the web. The application package is used for installing and managing the Sun SPOT software development kit. The application was also used for managing, updating and configuring connected Sun SPOT nodes. The application includes an emulator called Solarium [82] that is capable of running Sun SPOT software in a similar manner to the physical Sun SPOT – either for software testing or emulating Sun SPOT when a real Sun SPOT is not available.

Solarium is a Java application that can be used to remotely manage the Sun SPOT network, discover any nearby Sun SPOT and manage a complete Sun SPOT application life cycle running on those nodes. Solarium is capable of:

i) Sun SPOT Discovery and Display – Solarium can discover and display any real Sun SPOT that is connected to the host PC via USB or any Sun SPOT that is reachable via radio communication.

ii) Sun SPOT Interaction – Solarium can be used to deploy or remove software on Sun SPOT, start / pause / resume / stop an application and query the current Sun SPOT state (e.g. memory usage and energy statistics). Solarium also provides a Radio View that visualises the Sun SPOT's radio connectivity.

iii) Managing a Network of Sun SPOTs – Solarium is also capable of managing a group of Sun SPOTs via the *Deployment View* feature. A single click feature enables application deployment of each Sun SPOT. The current status of each Sun SPOT also can be viewed.

iv) Emulating Sun SPOT – Solarium comes with a built in emulator that can be used to run applications in a similar manner to a real Sun SPOT.

A created Virtual Sun SPOT appears on Solarium as a graphical Sun SPOT representation depicting their sensor panel, controls and outer casing as shown in Figure 3-2. Software running on virtual Sun SPOT can do almost anything a physical Sun SPOT is capable of, including:

- Change LED colour and illumination
- Set digital pin output levels
- Receive and send messages by network emulated radio
- Sense emulated light level
- Sense emulated temperature
- Sense emulated acceleration
- Sense emulated physical switch buttons and reset button
- Sense emulated digital pin input levels and analogue input voltages

However, there are other aspects of the real Sun SPOT not emulated, such as the virtual Sun SPOT cannot control the radio channel, pan id or power level, or the ability to turn the radio on and off. Various sensor board functions such as the UART, tone generation, servo control, pulse generation, timing a pulse width and a doing logical operation on the Atmega registers is not emulated in the virtual Sun SPOT. There is also no emulation of low level processor hardware function.

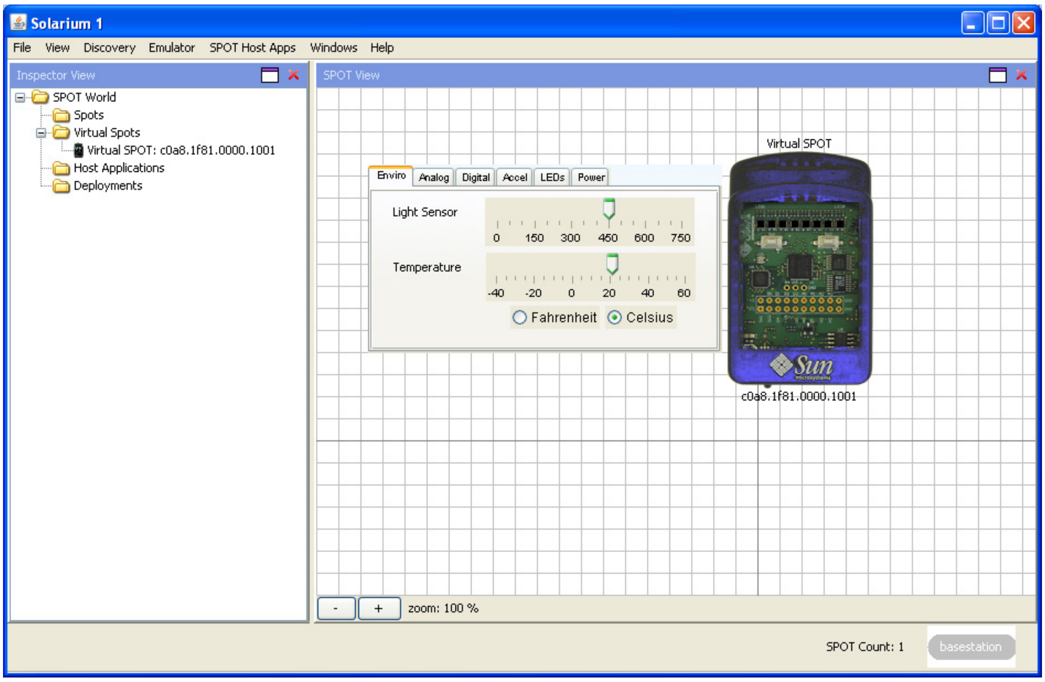


Figure 3-2. Solarium – Sun Spot Emulator

By right clicking on the virtual Sun SPOT, one is presented with a menu of possible commands as shown below in Figure 3-3.

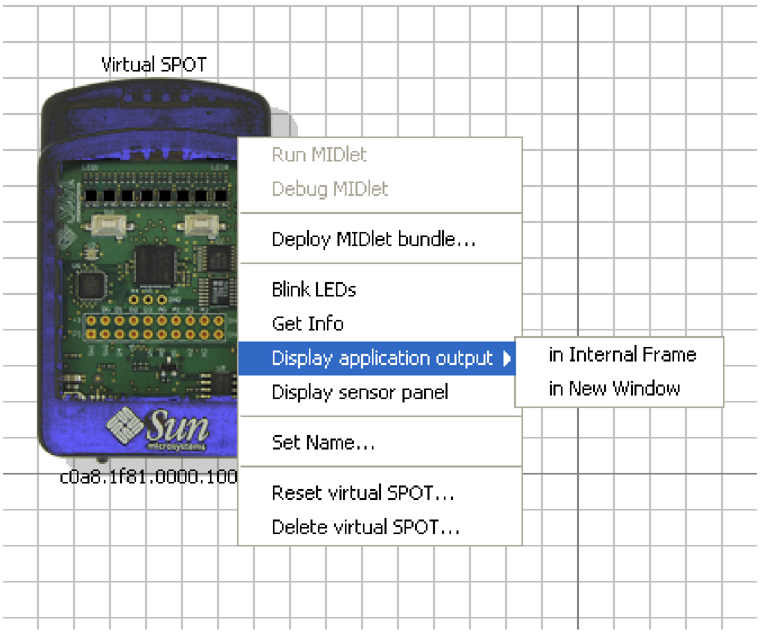


Figure 3-3.Virtual Sun SPOT

To run an application on the virtual Sun SPOT, choose *Deploy MIDlet bundle* command. This command lets you deploy a Sun SPOT application to the virtual Sun SPOT. The command will bring up a file chooser dialogue box that we can use to select a previously created or existing application's jar file. Alternatively, we can choose a project's `build.xml` file, where a process will begin to compile the project's source code, build the jar file and finally load it into the virtual Sun SPOT. Once loaded, use *Run MIDlet* command to display a submenu that lists all applications deployed, and start the application. Any running application will be displayed beside the virtual Sun SPOT.

Any running application on a virtual Sun SPOT can be debugged via the *Debug MIDlet* command. This command allows us to connect to an external Java Debugger for debugging purposes.

If needed, we can use the *Reset virtual SPOT* command to fully reset the virtual Sun SPOT. The command will cause any running application to be killed and Sun SPOT Squawk VM to be restarted. Any deployed jar file / application will be there, ready to be run again.

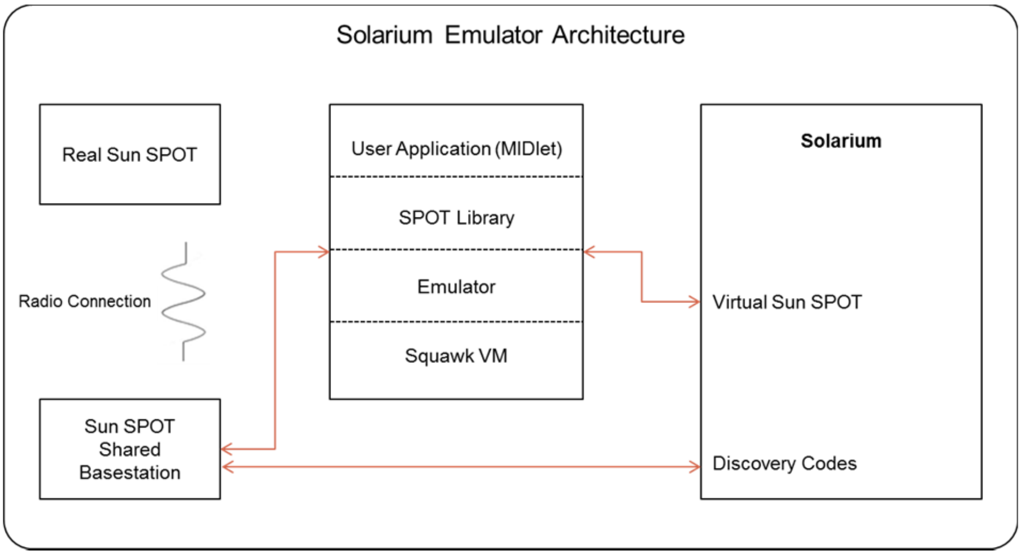


Figure 3-4. Solarium Architecture Block diagram

Last but not least, Solarium provides virtual Sun SPOT communication emulation. Virtual Sun SPOTs can communicate with each other by opening virtual radio connections, both broadcast and point to point. Instead of using real radio, these connections take place over regular and multicast socket connections. If a Sun SPOT basestation is connected to the host PC and a shared basestation is running, virtual Sun SPOT can use it to communicate with real Sun SPOTs using the basestation's radio. Figure 3-4 shows the block diagram of Solarium emulator architecture.

Each virtual Sun SPOT has its own Squawk VM running in a separate process on the host PC. Each Squawk VM includes a complete host side radio stack (part of the Sun SPOT library) that allows a deployed applications to communicate with other Sun SPOT application running on the host PC.

3.2.2 Development Tools

NetBeans IDE

For the development of Sun SPOT applications, we can use an ordinary command line or use a number of Integrated Development Environments (IDE) such as:

- NetBeans – well integrated with the Sun SPOT SDK and Sun SPOT Manager Tool
- IntelliJ – does not provide seamless integration with Sun SPOT SDK
- Eclipse – does not provide seamless integration with Sun SPOT SDK

For this study, we used a Java SE Development Kit 6 Update 25 and NetBeans 7.0 as standard Java IDE to write Sun SPOT applications.

Systematically, the written Sun SPOT application code was compiled and packaged into a "jar" file by the ANT script specified in `build.xml` and `build.properties` project files. The jar file then could be deployed to Sun SPOT from a host PC either directly via USB connection, or over the air (OTA) using the 802.15.4 radio. OTA deployment requires a Sun SPOT basestation that is connected to the host PC via a USB cable. Figure 3-5 shows screen capture of HCAmI projects under development using NetBeans.

Technically, management and deployment of application are handled by ANT scripts, which are called from NetBeans. Alternatively, an over the air tool can be used to push an application into Sun SPOT via the Solarium application provided with the Sun SPOT.

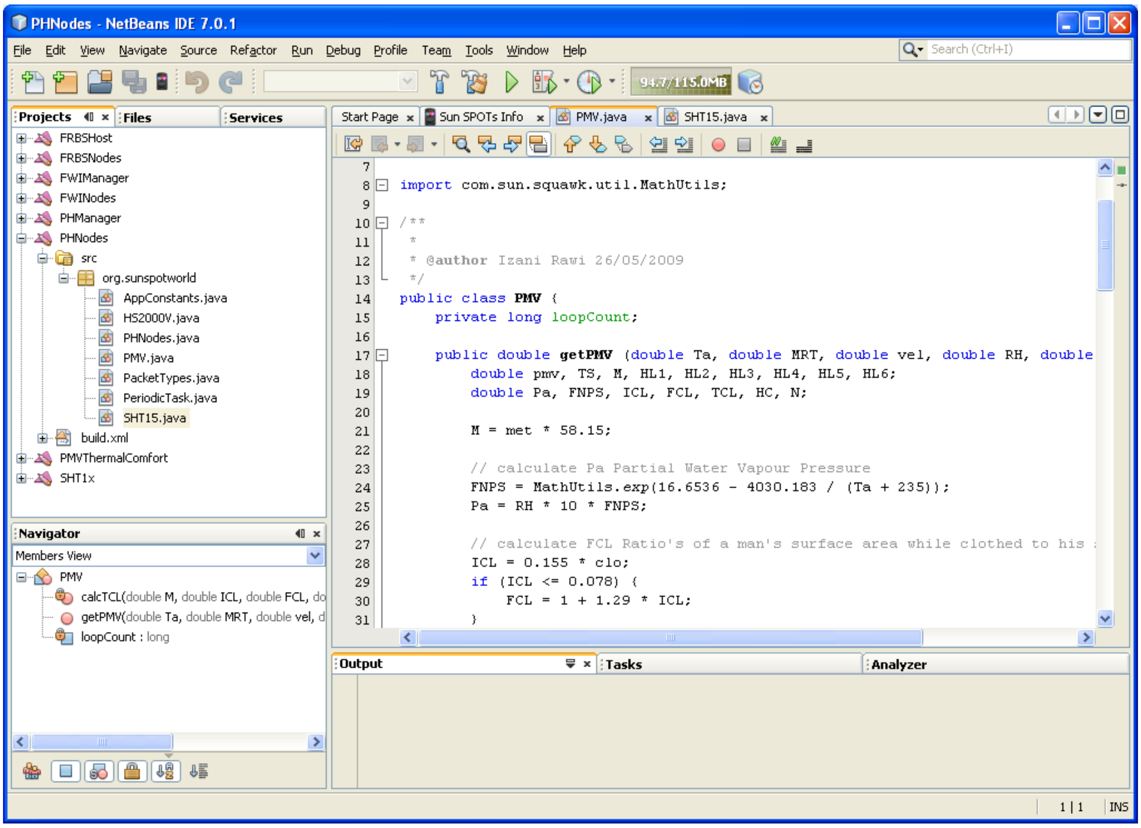


Figure 3-5. NetBeans Sun SPOT Projects

In general, there are three methods that can be used to develop, build, deploy and manage an application and the corresponding Sun SPOT node as summarised in Table 3-1.

Method / Tools	Tasks			
	Develop	Build	Deploy	Manage
Command Line	✓	✓	✓	
IDE (NetBeans)	✓	✓	✓	
Sun SPOT Manager Tool			✓	✓

Table 3-1. Sun SPOT Tools

The Sun SPOT application build and deploy process life cycle can be summarised as in Figure 3-6.

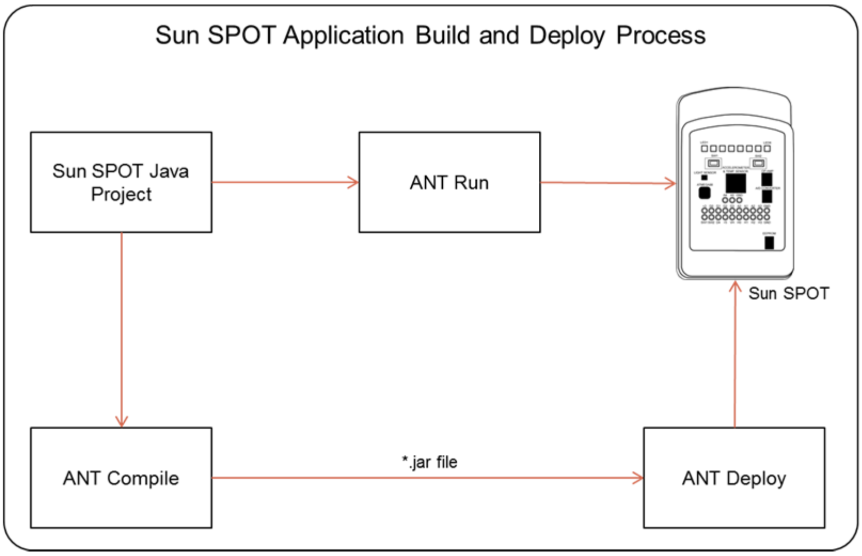


Figure 3-6. Sun SPOT Application Build and Deploy Process

Sun SPOT SDK

Sun SPOT SDK v5.01 Red [83] dated 4th January 2010 was used for this study. The SDK can be downloaded online using the Sun SPOT Manager Tool. Available SDKs can be acquired from *General Release*, *Beta Release* or *Dev Preview* update centre. Figure 3-7 shows a screenshot of the Sun SPOT Manager Tool displaying installed SDKs and available SDKs from chosen update centre. An acquired SDK was installed at *C:\Program Files\Sun\SunSPOT*.

Sun SPOT SDK provides hundreds of Application Programming Interface (API) specifications for the Sun SPOT Libraries. Sun SPOT applications are programmed in Java using APIs. APIs can be divided into four major packages as follows:

i) *SPOT and Sensorboard Libraries* – provide support for Sun SPOT hardware, peripheral and resources.
ii) *SPOT Generic Connection Framework* – provides support for IO and network connections resources.
iii) *Squawk Java ME Library* – provides access to Squawk VM resources.
iv) *Other Packages* – miscellaneous helper packages.

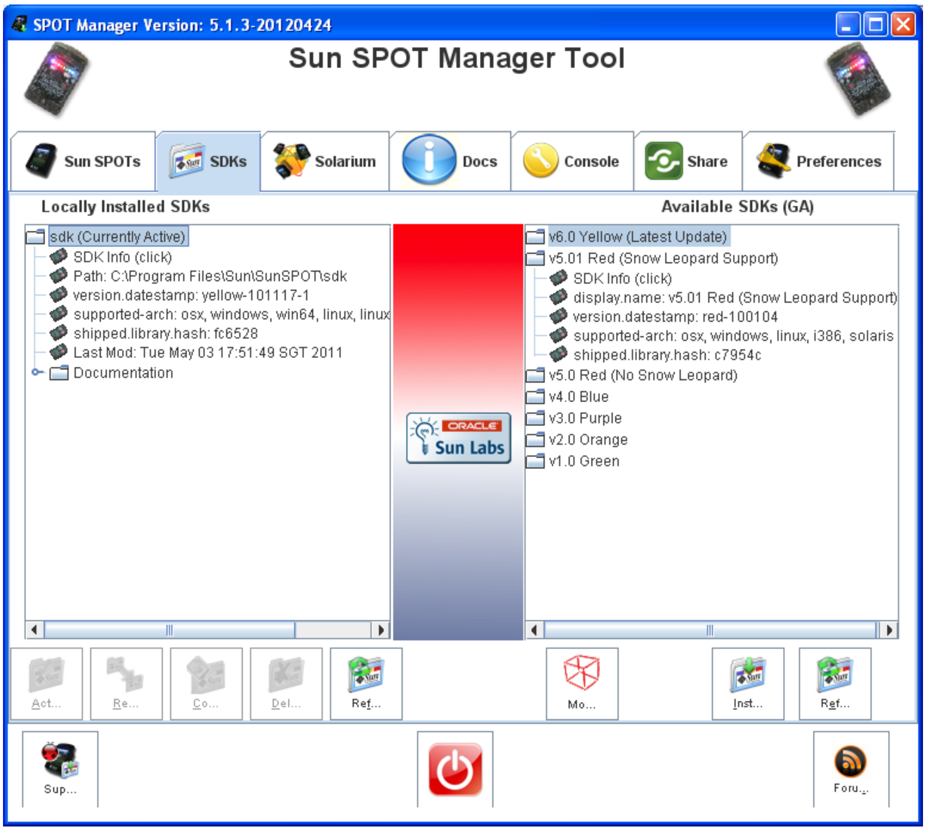


Figure 3-7. Sun SPOT SDKs

Sun SPOT Manager Tool

The Sun SPOT Manager Tool [81] is an application package that comes with the Sun SPOT Java Development Kit set (one basestation and two free range Sun SPOTs). Sun SPOT Manager Tool will facilitate development by providing a means to automatically install everything needed. The Sun SPOT Manager Tool will also allow us to update our Sun SPOT in future with the latest firmware and documentations.

Figure 3-8 shows the initial screen when Sun SPOT Manager was run, and checks for the presence of other required components. If any of those components is not present, it will assist the user to acquire and install it.

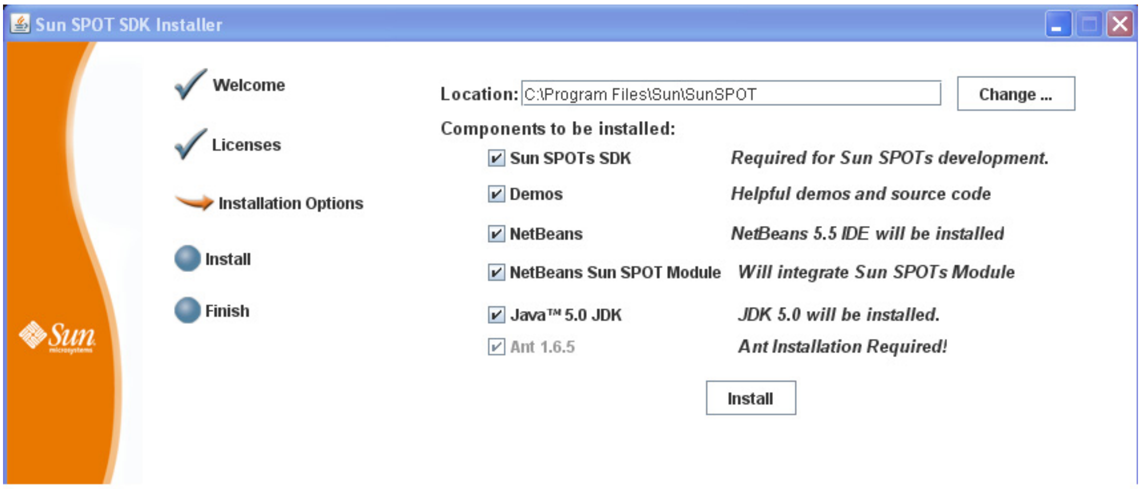


Figure 3-8. Sun SPOT Manager Tool Installation Validator

3.3 Hardware

The research utilised four of the shelf tools to model and simulate the concept of said ideas.

3.3.1 Sun SPOT

The hardware platform for all nodes is based on the Sun SPOT (Small Programmable Object Technology) wireless sensor platform [81-85]. Sun SPOT is a WSN mote originally developed by Sun Microsystems. Dimension of the Sun SPOT are 41mm x 23mm x 70mm and it weighs approximately 54grams.

Sun SPOT is capable of running a plethora of application [84] such as:

- Computer interaction device for 3D applications such as a virtual glove project
- Water quality management (pH, redox, turbidity)
- Monitoring a small rocket's flight from a laptop with ground telemetry software in a Sun Labs Space Program
- Video game controller
- Motor control in robotics such as in the SPOTCopter: Remote Control Helicopters project
- Simple apps (e.g. games running only in the Sun SPOT)

Sun SPOT WSN was chosen for this case study implementation platform due to it being purely Java based and easy to develop. Sun SPOT SDK offers stable implementation libraries (hardware interface, on board sensors, communication and radio, maths, network, desktop and testing, etc) that significantly simplified development tasks. Java is a popular language with convenient APIs, and a potential support for code mobility. Java is designed as a strong, secure platform. Java applications are run by a Java Virtual Machine (JVM), which isolates the programmer from dealing with low-level features directly. Java also comes with a run time garbage collection that restores used memory to the memory pool safely, thus avoiding nasty memory leaks (common issues with the C programming language).

Sun SPOT also supports handling multiple tasks (multitasking) and this is achieved by Java multithreading operations. Furthermore, the Sun SPOT platform allows for over the air deployment of Sun SPOT applications (a.k.a midlet). One more powerful trait of Sun SPOT is its hardware capabilities: it is computationally capable of handling complex mathematical operations (32-bit operations) in a small amount of time. It is far more powerful than TinyOS based sensor networks hardware. Sun SPOT's WSN platform therefore is considered to be the most suitable for rapid prototyping of sensor network applications.

The Sun SPOT uses a 180 MHz 32-bit ARM920T core processor with a 512K RAM and 4M Flash, 3.7V 720 mAh Li-ION battery (three mode: awake – between 70mA and 120mA, shallow sleep – 24mA, deep sleep – 32µA), onboard sensors (3-axis accelerometer, temperature, light), and several digital and analogue I/Os, and it is programmed almost entirely in Java. It was built on the Squawk Java Virtual Machine [86], which runs directly on the processor without an OS. It is a bare metal Java OS implementation (Squawk VM). The use of Java allows programmers to create projects without having to learn specific specialised embedded system development skills.

Sun SPOT can be connected to a wide range of application-specific sensors and actuators such as accelerometers, lights, temperature, humidity, sound, GPS units, and servos.

Figure 3-9. Sun SPOT Kit

Sun SPOT comes with 2.4 GHz radio with an integrated antenna on the board. The radio is a TI CC2420 (formerly ChipCon) and is IEEE 802.15.4 compliant. The nodes communicate using IEEE 802.15.4 standard with a basestation approach to sensor networking. It supports a IEEE 802.15.4 MAC layer, on top of which Zigbee can be built.

Sun SPOT routing and meshing uses the LowPAN protocol, while multi-hop connectivity is accomplished by the link quality routing protocol, a sophisticated ad hoc network routing protocol. This routing protocol takes into account not just connectivity, but also the quality of the connections when choosing a route [84].

A complete Sun SPOT kit contains a basestation (shown with USB cable) and two free range Sun SPOTs as shown in Figure 3-9. A free range Sun SPOT contains a Main Board (processor board) with a rechargeable Li-ION battery and an eDemo Board (sensor board) as shown in Figure 3-10.

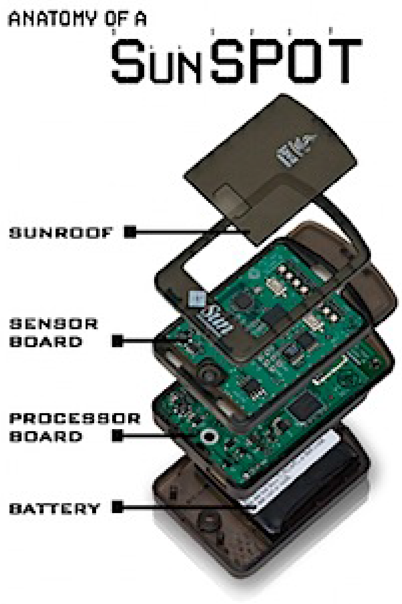


Figure 3-10. Sun SPOT Anatomy

A Sun SPOT Main Board (Figure 3-11) contains a:

i) ARM based main processor (Atmel AT91RM9200 system on chip)

ii) 4Mbyte flash memory and 512Kbyte RAM (Single Spansion S71PL032J40 multichip package)

iii) Power management circuit (Linear Technology LTC3455 U2 with integrated Li-ION battery charger, USB power manager and dual switching regulator)

iv) Power controller (8-bit microcontroller Atmel ATmega88)

v) 802.15.4 radio transceiver and antenna (TI CC2440 / formerly ChipCon) with a 2.4GHz RF transmitter / receiver and a digital Direct Sequence Spread Spectrum (DSSS) baseband modem with MAC support.

vi) Battery connector

vii) Daughterboard connector

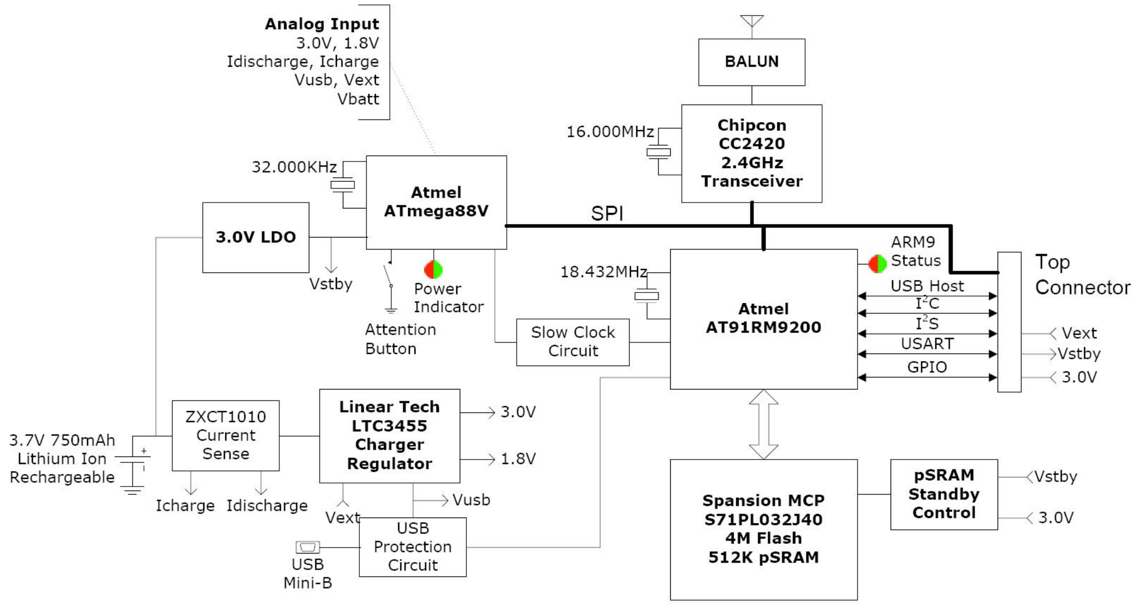


Figure 3-11. Sun SPOT Main Board Schematics

The Sun SPOT eDemo Board (Figure 3-12) has the following features:

i) 8 tri-colour LEDs (LED1 to LED8)

ii) 2 momentary switches (SW1 and SW2)

iii) 3-axis accelerometer

iv) Temperature sensor

v) Light sensor (Lux ≈ 2 x ADC)

vi) Analogue to digital input converter pins (A0, A1, A2 and A3)

vii) General Purpose I/O (GPIO) pins (D0, D1, D2 and D3)

viii) High current output pins (H0, H1, H2 and H3)

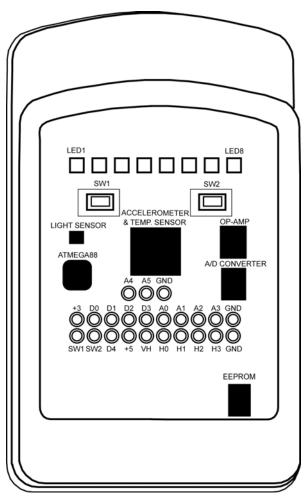


Figure 3-12. Sun SPOT eDemo Board Diagram

Figure 3-13 shows a typical topology used with Sun SPOT.

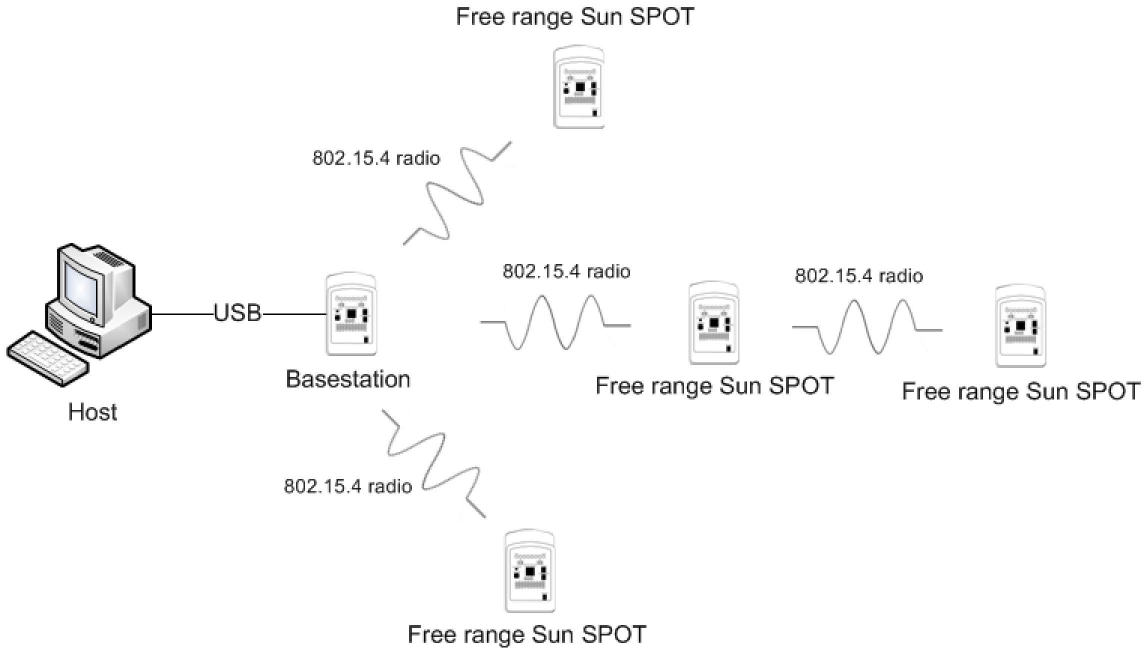


Figure 3-13. Sun SPOT Topology

The free range Sun SPOT communicates wirelessly (802.15.4 radio, mesh routing) with other sun SPOTs and the basestation. In the event of more than one hop communication between Sun SPOTs, a Link Quality Routing Protocol (LQRP) algorithm is used to determine the best route to choose. A LQRP request for a route to a particular targeted Sun SPOT is broadcast by a requester, and then re-broadcast by each Sun SPOT in the network that receives them. Each Sun SPOT that happens to know the route to the intended target then sends a reply back to the requester. The route that is going to be used is the one with the best link quality.

The basestation (a Sun SPOT without eDemo Board and battery) is connected to a host via USB. The basestation is powered solely by a USB connection. Java based host applications (J2SE) interact with all Sun SPOT through the basestation. The objective of the Sun SPOT basestation software is to let application running on the host to interact with applications running on any free range Sun SPOT. The host can be on any platform – Windows PC, MacOS or Linux box.

3.3.2 Sensirion SHT1x Humidity and Temperature Sensor

SHT15 is a digital fully calibrated humidity and temperature sensor that comes with a tiny footprint (4.93mm x 7.47mm x 2.5mm) as shown in Figure 3-14. SHT15 is Sensirion's family of surface mountable relative humidity and temperature sensors [87].

A capacitive sensor element is used for measuring relative humidity while temperature is measured by a band-gap sensor. Both sensors are seamlessly coupled to a 14-bit analogue to digital converter (ADC) and a serial interface circuit. It is individually calibrated in a precision humidity and temperature chamber with its own calibration coefficients saved on the sensor's own OTP memory on the chip. The two-wire serial interface and internal voltage regulation allows for fast, easy system integration. SHT15 is supplied in a surface mountable LCC (Leadless Chip Carrier) for an easy soldering process even for the most demanding applications.

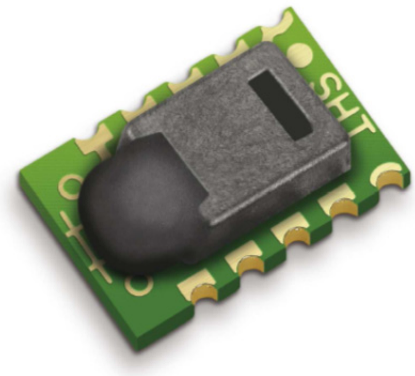


Figure 3-14. SHT15 Digital Humidity and Temperature Sensor

SHT15 technical specification as follows:

Specification	Notes
Energy consumption	800µW at 12-bit, 3V, 1 measurement/s
RH operating range	0 – 100% RH
Temp operating range	-40°C – +125°C (-40°F – +257°F)
RH response time	8 sec (tau 63%)
Output	Digital with 2 wires interface

Table 3-2. SHT15 Technical Specification

SHT15 can be easily integrated into any digital system with a 2-wire serial interface as shown in SHT15 schematics (Figure 3-15).

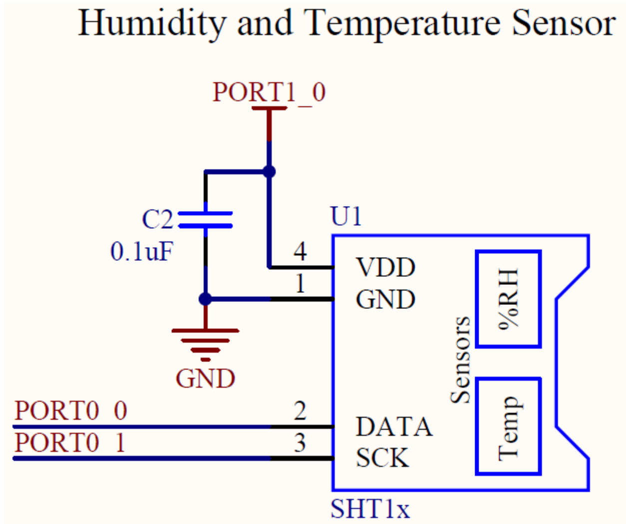


Figure 3-15. SHT15 Schematics

SHT15 pin assignment as follows:

Pin	Name	Comment
1	GND	Ground
2	DATA	Serial Data, bidirectional
3	SCK	Serial Clock, input only
4	VDD	Source Voltage (2.4V – 5.5V)

Table 3-3. SHT15 Pin Assignment

The supply voltage must be between 2.4V to 5.5V with a recommended supply voltage of 3.3V. Power supply pins Source Voltage (VDD) and Ground (GND) must be decoupled with a 0.1µF capacitor as shown in Figure 3-15.

The Serial Clock (SCK) pin is used to synchronise the communication between the microcontroller and SHT15, which consists of fully static logic with no minimum SCL frequency needed.

Finally, the Serial Data (DATA) pin is a tri-state pin used for transferring data in and out of the sensor. Data is valid on the rising edge of the SCK and must remain stable while SCK is high.

3.3.3 Precon HS-2000V Humidity and Temperature Sensor

Precon HS-2000V is an analogue factory calibrated humidity and temperature sensor. HS-2000V, shown in Figure 3-16, combines capacitive-polymer sensing technology with a novel measurement method, eliminating the need for temperature correction and calibration by the user. HS-2000V (11.9mm x 22.6mm x 9.7mm) is pre-packaged, with good stability, is field replaceable, and has excellent chemical resistance.

The sensor includes a thermistor and circuitry to correct for temperature and calculate the true relative humidity. The sensor provides both humidity and temperature outputs and is accurate to ±2% RH and ±0.40°C respectively [88].

Figure 3-16. HS-2000V Humidity and Temperature Sensor

HS-2000V technical specification as follows:

Specification	Notes
Operating Temperature	-30°C – +85°C (-22°F – +185°F)
Output Range	-30°C – +100°C (-22°F – +212°F)
Storage Temperature	-40°C – +125°C (-22°F – +257°F)
Operating Humidity Range	0% – 100%
Supply Voltage	+5.5 volts
Soldering Temperature	10 sec at 250°C (520°F)

Table 3-4. HS-2000V Technical Specification

HS-2000V can be easily integrated into any analogue system with a four wire interface as shown in HS-2000V schematics Figure 3-17.

Figure 3-17. HS-2000V Schematics

HS-2000V pin assignment as follows:

Pin	Comment
1	Temperature Out (0V to V supply)
2	Power (2V to 5.5V)
3	Relative Humidity Out (0V to V supply)
4	Ground

Table 3-5. HS-2000V Pin Assignment

HS-2000V output is ratiometric, the output voltage varies from zero to the supply voltage, as the measured parameter varies from zero to full scale. For instance, if supply voltage is 4.0V, 60% relative humidity produces a 2.4V output signal on RH output pin. It is similar for the temperature pin.

3.3.4 UNI-T UT351 Sound Level Meter

UNI0T UT351 (Figure 3-18) is a portable sound level meter with DC and AC analogue signal output. The meter is used in capturing ambient noise and is suitable for use in noise control, quality control, health care and several kinds of environmental noise testing. UT351 measurement range is from 30dB to 130dB with ±1.5dB accuracy. The meter is equipped with a ½” electrets condenser and is sensitive between the 31.5Hz to 8000Hz frequency range, and performs 125mS or 1s time samplings. Captured output DC analogue voltage is converted to corresponding dBs (10mV/dB, 100Ω output impedance). UT351 comes in a sturdy casing of 273mm x 69mm x 39mm dimensions and weighs around 386 grams.

Figure 3-18. UT351 Sound Level Meter

The availability of DC analogue voltage makes it suitable for real-time data capturing. The output DC voltage can be converted into the appropriate dB level and eventually be stored permanently.

3.4 Chapter Summary

This chapter outlines both the hardware and software utilised in this research. A number of proprietary and open source software items were used as simulation tools and development tools.

For hardware, the Sun SPOT WSN platform was presented as case the study platform. The SHT15 and Precon HS-2000V provided stable and high resolution measurements. The UNI-T UT351 sound level meter provides sound level readings for the project.

Chapter 4 Human Comfort Ambient Intelligence (HCAmI)

4.1 Introduction

This chapter discusses the HCAmI concept design and architecture from a number of perspectives.

One of the primary application domains of WSN as emerging technology is in building and surrounding space monitoring and automation. WSN could be part of the bigger picture, providing solutions to Ambient Assisted Living (AAL) and Wireless Home Automation Network (WHAN) systems. AAL can be considered an ambient system for occupants, with WHAN for the living space [89].

AAL is an emerging area within AmI with a focus on enabling elderly people to live more independently and for longer in their homes [90]. AAL systems can be considered the brain that allows the monitoring of their activities, such as managing medication routines [91], physical movement, body positions and vital life signs. AAL provides the solution which helps to minimise the onset of chronic conditions which can be costly to deal with [92]. Innovative use of new technologies may provide the care needed where WSN could be part of the solution in AAL systems.

Meanwhile, WHAN systems facilitate monitoring and control applications for home user comfort and efficient home management. WHANs normally consist of several types of embedded device that sense the environment, are equipped with low-power Radio Frequency (RF) transceivers and may be battery powered. The use of RF as means of communication allows flexible installation (addition or removal of devices at any location) and cuts down installation costs by eliminating conduits and cable trays from the system. On the other hand, RF itself poses a significant challenge itself. The dynamics of radio propagation such as refraction, deflection, signal attenuation and signal loss pose a significant challenge to the design, deployment and use of WSN in WHANs [1]

The increasing level of device availability in modern dwellings provides several opportunities for owners, building managers, equipment manufacturers and solution providers. Interconnected devices open limitless opportunities to intelligently monitor and control smart homes in a future Internet of Things. Energy saving applications, for instance, controls indoor climate and energy usage according to circumstances at any particular moment. Appliances, room temperature, window openings and air circulation can be controlled accordingly. In essence, the home environment emphatically should be able to respond and modify itself continuously according to its diverse inhabitants and their changeable requirements. Therefore, intelligent smart homes are regarded as pioneering the procession of creating sensitive, adaptive and responsive home surroundings.

Smart home technology products and services can be classified in six categories: comfort, energy management, multimedia and entertainment, healthcare, safety and security and communication. The boundary between categories is not strict, different categories do overlap. The different categories strengthen and complement each other in providing the overall functionality and traits of a smart home. Our research falls within the comfort, communication and energy management categories, with the main emphasis on comfort.

4.2 Smart Home Environment Requirements

Smart home environment requirements can be viewed from various perspectives. One can see it from a security point of view, environmental monitoring and control, entertainment, communication, occupant monitoring and much more.

Here, we look at the requirements for a smart home environment from an activity viewpoint. The requirements can be explored from both an energy centric and human centric aspect.

4.2.1 Energy Centric Perspective

From the energy centric activity viewpoint, the smart home environment focused on energy activity, as depicted in Figure 4-1 [93].

Figure 4-1. Energy Centric Activity: Energy Optimisation

The power management subsystem works in tandem with the climate control subsystem, sensors, actuators, and energy supply sub system whilst maintaining optimum energy usage based on various comfortable living standards and specifications. Minimising energy consumption is the primary goal, while at the same time retaining a high degree of environmental comfort [93].

4.2.2 Human Centric Perspective

From a user perspective, it is important to appreciate that occupants consider their home should be a safe and comfortable place. Technology is seen as a tool for residents to control their environment. It provides tools and services that allow and enable people to address their own social, rational and emotional needs.

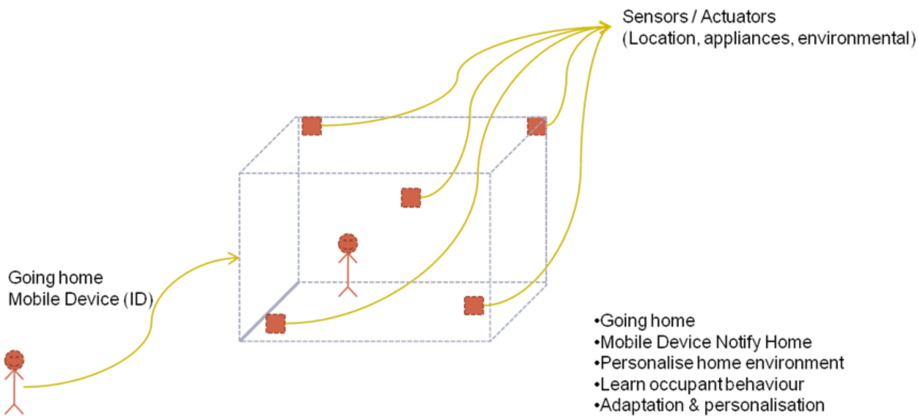


Figure 4-2. Human Centric Activity: Automation, Personalisation, and Adaptation

Figure 4-2 [93] illustrates the human centric activity perspective. Automation, personalisation, and adaptation are the primary goal. As the occupant comes into the living space, the sensors and actuators should spring into action to provide optimum environmental comfort based on occupant preferences and profile, whilst maintaining the minimum energy expenditure. The system will maintain optimal comfort by manipulation of various comfort factors as necessary [93].

The notion of unsupervised learning and control should be the tenet of AmI into a smart home environment control strategy. However, it creates an enormous challenge, as the decisions must be made in near real time. It must interact with the user to obtain feedback and at the same time must not interfere with user activity. The system should adapt, not the user. The system should use unsupervised learning and control mechanisms such as fuzzy logic for classification, a rule based system for decision making and a genetic algorithm for optimisation.

4.3 HCAmI Framework and Architecture

The concept proposed uses the HCAmI framework and architecture, with a focus on the functionality available to the user, in line with Aldrich's five hierarchical classes of smart homes [94]:

i) Homes that contain intelligent objects – homes contain single, standalone applications and objects that function in an intelligent manner
ii) Homes that contain intelligent, communicating objects – homes contain appliances and objects that function intelligently in their own right and which also exchange information between one another to increase functionality
iii) Connected homes – homes have internal and external networks, allowing interactive and remote control of systems, as well as access to services and information, both within and beyond the home
iv) Learning homes – patterns of activity in the homes are recorded and the accumulated data are used to anticipate users' needs and to control the technology accordingly
v) Attentive homes – the activity and location of people and objects within the homes are constantly registered, and this information is used to control technology in anticipation of the occupants' needs.

In our work, HCAmI focuses on providing meaningful insight into a typical living space comfort state from a human comfort perspective. The main objective of our work is to design and implement a distributed WSN based system that allows the execution of a set of complex applications (sense, measure, FRBS, communication, etc), updating / modifying the sensor intelligent matrix, dealing with raw sensory data, and communicating among sensor nodes. The research attempts to deal with the AmI frontier at the sensor node level rather than the traditional / conventional server or central computational system level.

4.3.1 Overview of HCAmI System

Figure 4-5 illustrates the building blocks of HCAmI architecture from a software functionality perspective. The HCI System provides a living space comfort index based on aggregation of individual Thermal Comfort (TC), Visual Comfort (VC), Indoor Air Comfort (AC) and Acoustical Comfort (AC) values.

Each comfort subsystem works independently and serves as human comfort knowledge components. Each component will work out the respective comfort values from sensed parameters such as air temperature, mean radiant temperature, relative humidity, air velocity, clothing level, metabolic rate, luminance level, shading level, CO_2 concentration and ambient sound level. Due to modular organisation, each comfort factor can be calculated within the sensor group itself even though the sensed value might come from a different node as shown in Figure 4-3 [95].

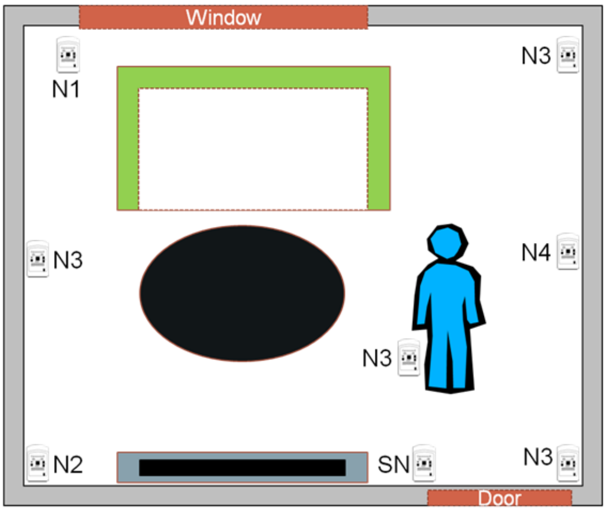


Figure 4-3 Typical Living Space With Node Placement

Figure 4-3 depicts an example of a typical living space where minimal sensor nodes placement points are shown. Every node may consist of one or more sensors and may share their sensor reading with other sensor that requires them.

For example, node N1 in the vicinity of the window, may sense shading level and luminance level, and therefore be responsible for visual comfort calculation. Visual comfort depends upon indoor illuminance levels [96] and therefore the sensor is placed as shown.

Node N2 is responsible for sensing CO_2 concentration and is responsible for indoor air comfort computation. N2 was placed as recommended by CIBSE [97] where indoor air quality depends on the CO_2 concentration inside a living space.

N3 nodes collaborate among themselves to determine the thermal comfort PMV value [93]. Thermal comfort depends upon indoor air temperature, relative humidity, air movement and indoor mean radiant temperature (indoor surface temperature). Additionally, thermal comfort depends upon subjective parameters such a person's clothing, metabolism and activity level. The first four parameters are measured by fixed N3 nodes while subjective parameters are measured by a mobile N3 node (depicted in proximity to the person figurine) [93].

Node N4, equipped with a microphone, listens to the living space's ambient sound. It is responsible for figuring the acoustical comfort value. Increase in awareness and growing interest in monitoring noise pollution parameters in urban areas shows that noise plays a significant contribution to determining human comfort [98, 99]. Recent studies [98, 99] have demonstrated that exposure to unwanted noise increases the risk of hearing loss, sleep disorder and hypertension, and can negatively influence productivity and social activity.

Finally, node SN is the sink node that finalises the HCI calculation and acts as gateway to the outside world. SN was designed as the final stop point, where it will communicate with various comfort group sensors, acquiring individual comfort values, and calculate the HCI. The modular design of the system gives the flexibility of adding or removing individual comfort factors as required. It also gives the flexibility of adding or removing additional sensor nodes as needed. The whole system also can be replicated in other living space such as the bathroom, living hall, study room, bedroom and kitchen with minor modifications, as seen in Figure 4-4. It can be reconfigured simply by 'pushing' new knowledge into respective nodes, as required. Each room / space requires different comfort traits, hence the alterations are warranted.

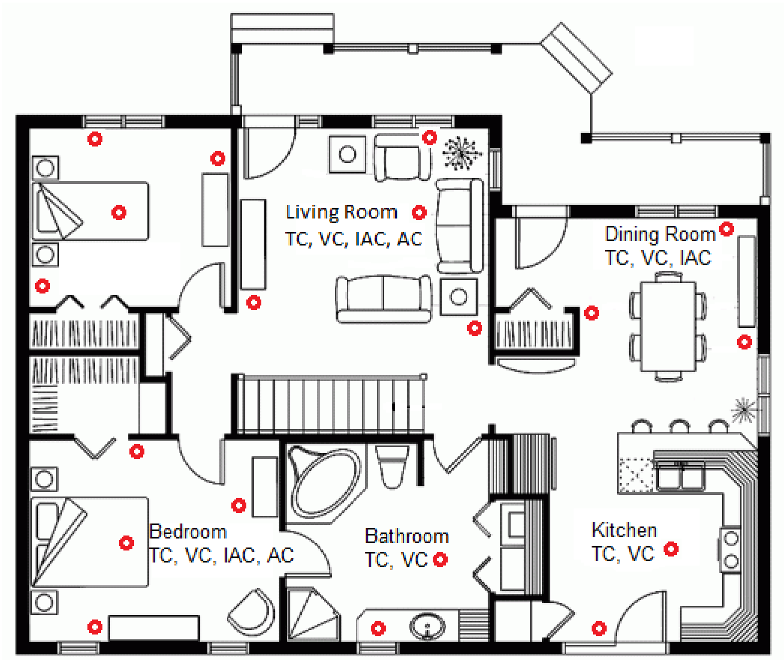


Figure 4-4. HCAmI in a House

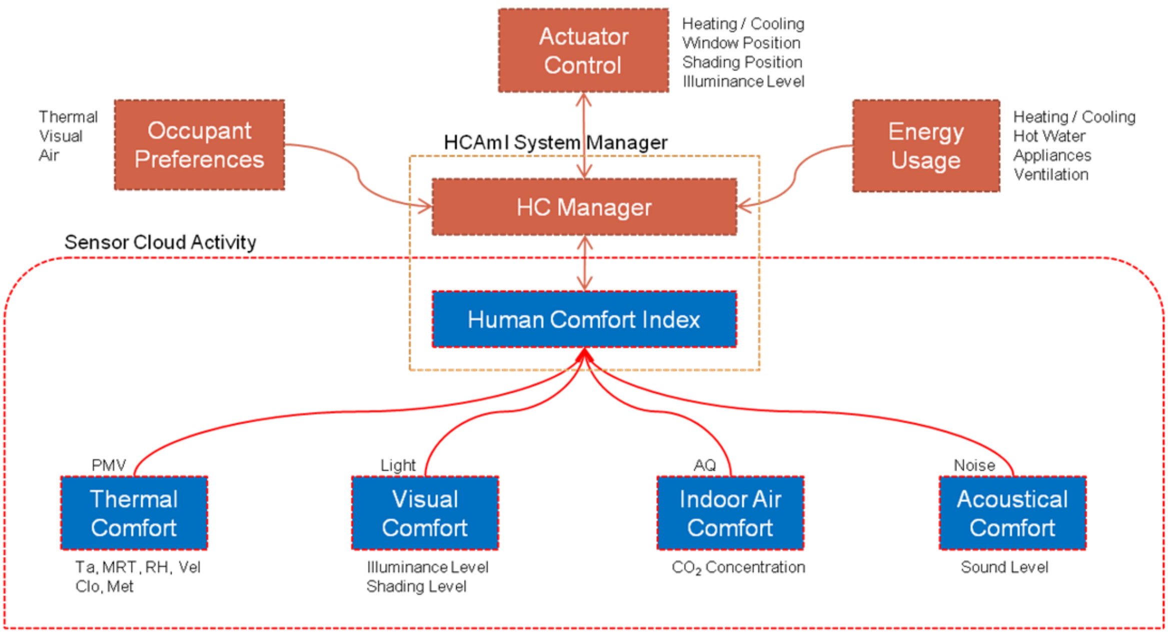


Figure 4-5. HCAmI Architecture

The key functionality of each component / subsystem, shown in Figure 4-5 [95] is as follows:

i) *HC Manager*: Middleware that acts as the bridge between the sensor cloud and the outside world, such as input interface from user, input / output to actuator control and input regarding energy usage / consumption. HCI Manager also acts as overall system indicator, for instance showing current living space HCI, individual comfort state / value, energy expenditure and actuator position / status.
ii) *Occupant Preferences*: Input interface that captures user feedback into the system (e.g. thermal preference, visual preference).
iii) *Actuator Control*: Interfaces with the living space environmental control system, such as heating / cooling system, luminance level, etc.
iv) *Energy Usage*: Block that monitors energy expenditure at the system level.
v) *Human Comfort Index*: Act as summative node that finalises HCI calculation and HCI weight. HCI weight adjustment / fine tuning will be based on HCI Manager recommendation.
vi) *Thermal Comfort*: Computational block that manages thermal comfort activity by means of sharing and fusing data among thermal sensors.
vii) *Visual Comfort*: Computational block that manages visual comfort activity by means of sharing and fusing data among light intensity sensors.
viii) *Indoor Air Comfort*: Computational block that manages indoor air comfort activity by means of sharing and fusing data among CO_2 sensors.
ix) *Acoustical Comfort*: Computational block that manages acoustical comfort activity by means of sharing and fusing data among acoustic sensors.

This work focuses on integrating multiple comfort requirements into a novel solution for managing the overall human comfort in a living space. Various previous studies only dealt with separate individual comfort functions, without looking into an integrated solution. Combined with the WSN and the FRBS domain, this study investigates how collaboration among sensor nodes and knowledge components is dealt with, within the WSN scope and limitations.

4.3.2 HCAmI Knowledge Component Architecture

In this section, various knowledge components for the HCAmI system are presented. The main goal of the system is to correctly interpret environment data and make sound environmental decisions. The decision process is complex because of the multiple comfort factors involved. The most prominent and complex is thermal comfort. This work relied on academic / scientific knowledge accessible from the literature to extract the relevant information and represent it in fuzzy rule based forms (fuzzy rule based knowledge representation systems).

4.3.2.1 Spatial Sensing

An interpolation based mechanism was developed where any group of WSN nodes can determine any raw environment reading, or calculate comfort value where no sensor is present. PMV mapping of a living space is demonstrated in Chapter 5

In a living space, it is impossible or impractical to have sensors fitted in every corner. It is also not viable in certain place, such as a sensor dangling in the middle of a living room. However, to calculate HCI at a particular point in a living space, one has to know all the relevant environmental values. To overcome this, the Inverse Distance Weighted (IDW) interpolation technique is used to determine the environmental parameters that are used to compute comfort factors (e.g. PMV value) of a person in a living space at a particular location.

IDW is a deterministic estimation method whereby virtual measurements of a location that has no sensing points are determined. This is achieved by a linear combination of values at known sample points. IDW assumes that each point has a local influence that diminishes with distance. The predicted values at unknown points are calculated with a weighted average of the values available at the known points. IDW combines the notion of proximity with that of gradual change of the trend surface. IDW relies on the assumption that the value at an unsampled location is a distance-weighted average of the values from surrounding data points, within a specified space. The points closest to the prediction location are assumed to have greater influence on the predicted value than those further away, such that the weight attached to each point is an inverse function of its distance from the target location [100].

The simplest form of IDW is sometimes called "Shepard's Method" [100]. The general formula is given in Equation 4–1 as follows:

$$\hat{Z}(X_0, Y_0) = \sum_{i=1}^{n} \lambda_i * Z(X_i, Y_i) \qquad \text{Equation 4–1}$$

Where:

$\hat{Z}(X_0, Y_0)$ is the value being predicted for the target location (X_0, Y_0)

n is the number of measured / known data points in the set

$Z(X_i, Y_i)$ are the observed / known values at the location (X_i, Y_i)

λ_i are the weight functions assigned to each scatter point. The sum of the weights is equal to 1. The classical form of the weight function is:

$$\lambda_i = \frac{\frac{1}{d_i}}{\left(\sum_{i=1}^{n} \frac{1}{d_i}\right)} \qquad \text{Equation 4–2}$$

d_i is the distances to all known scatter points.

Shephard's Method was chosen due to its simplicity and thus minimises computing load on the WSN node.

Usage example of IDW can be visualised in the following example. Assume that we have a living space fitted with four temperature sensors (N1 = 24^{o}C, N2 = 23^{o}C, N3 = 21^{o}C and N4 = 22^{o}C) as depicted in Figure 4-6. We wish to know the temperature at N0.

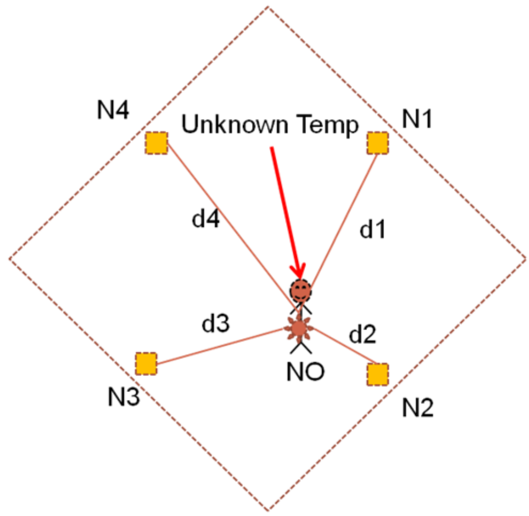


Figure 4-6. Imaginary Living Space

Steps to calculate temperature at N0:

i) Get all the distances N1, N2, N3 and N4 to N0, say that d1 = 3.04m, d2 = 1.58m, d3 = 2.24m and d4 = 3.64m

ii) Calculate the weight of each point using the Equation 4-2 (λ_1 = 0.195, λ_2, = 0.376, λ_3 = 0.265 and λ_4 = 0.163)

iii) Calculate weighted average for N0 temperature using the Equation 4-1. Here, N0 is 22.5^{o}C.

We can use a similar technique for other environmental parameters, such as MRT, RH, Vel, dB, Lux, etc. The method is rather straight forward and can be easily incorporated into WSN itself. The generic IDW engine was coded based on the IDW flowchart in the Appendix B IDW Flowchart.

4.3.2.2 PMV Engine and Thermal Comfort Knowledge Component

The PMV engine was developed for the WSN and used for establishing the thermal comfort of a living space. The interpolation-based mechanism (Spatial Sensing) is used for calculating the PMV at any point within the space. PMV mapping of a living space is demonstrated in Chapter 5.

Based on Equation 2–1, the PMV algorithm was devised in such way that it suits WSN's limited computational resources.

PMV Algorithm:

i) Start

ii) Get Ta, MRT, Vel, RH, Clo and Met from sensor(s)

iii) Calculate M = met * 58.15

iv) Calculate Saturated Vapour Pressure FNPS = exp(16.6536 – 4030.183 / (Ta + 235))

v) Calculate Partial Water Vapour Pressure Pa = RH * 10 * FNPS

vi) Calculate Clothing Thermal Resistance ICL = 0.155 * clo

vii) Check if ICL <= 0.078, if yes go to viii else go to x

viii) Calculate Clothing Surface Area Factor FCL = 1 + 1.29 * ICL

ix) Go to xii

x) Calculate Clothing Surface Area Factor FCL = 1.05 + 0.645 * ICL

xi) Go to xii

xii) Calculate Clothing Surface Temperature TCL

xiii) Calculate Convection Heat Transfer Coefficient HC

xiv) Calculate Thermal Sensation TS = 0.303 * exp(-0.036 * M) + 0.028

xv) Calculate Skin Diffusion Loss HL1 = $3.05 \times 10^{-3}[5733 - 6.99(M - W) - P_a]$

xvi) Calculate Sweat Loss HL2 = $0.42[(M - W) - 58.15]$

xvii) Calculate Latent Respiration Loss HL3 = $1.7 \times 10^{-5} M(5867 - P_a)$

xviii) Calculate Dry Respiration Loss HL4 = $0.0014M(34 - t_a)$

xix) Calculate Radiation Loss HL5 = $3.96 \times 10^{-8} f_{cl}[(t_{cl} + 273)^4 - (t_r + 273)^4]$

xx) Calculate Convection Loss HL6 = $f_{cl} h_c (t_{cl} - t_a)$

xxi) Calculate PMV = TS * (M – HL1 – HL2 – HL3 – HL4 – HL5 – HL6)

xxii) Stop

Note:

HC and TCL are calculated by iteration.

HL1, HL2, HL3, HL4, HL5 and HL6 are the heat loss components.

Ta, MRT, Vel, RH, Clo and Met value from sensor source as discussed in Chapter 5.

PMV calculation flowchart can be seen in the Appendix C PMV Flowchart. Calculated PMV value is then used as raw data for thermal comfort determination.

A series of production rules were formulated to cater for human comfort activity. Rules are formulated as if / then clauses. If the condition is true or has reached a certain degree of validity, then the conclusion or action is activated. The generic form of the rule is as follows:

```
IF <condition> THEN <conclusion / action>
```

Thermal comfort FRBS was adapted from [93] where PMV was calculated. The PMV calculation (PMV Engine) takes into account six raw input values – Dry Bulb Temperature (DBT), Mean Radiant Temperature (MRT), Relative Humidity (RH), Air Movement (AM), Metabolic Rate (Met) and Clothing Level (Clo). PMV value is then used as the input for thermal comfort FRBS as shown in Figure 4-7.

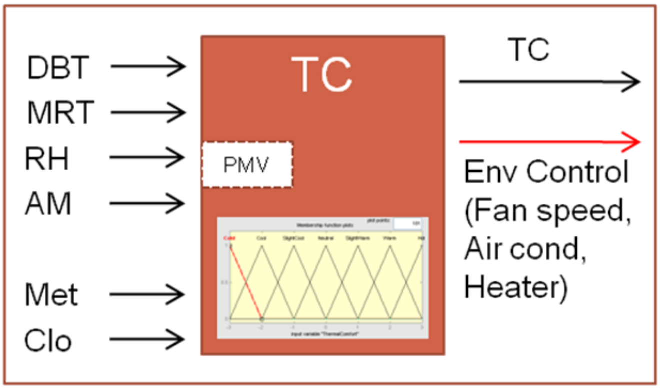


Figure 4-7. Thermal Comfort FRBS Node

The input membership function of thermal comfort is illustrated in Figure 4-8. The value of membership function was fine-tuned according to [20]. It was divided into 7 membership functions as follows:

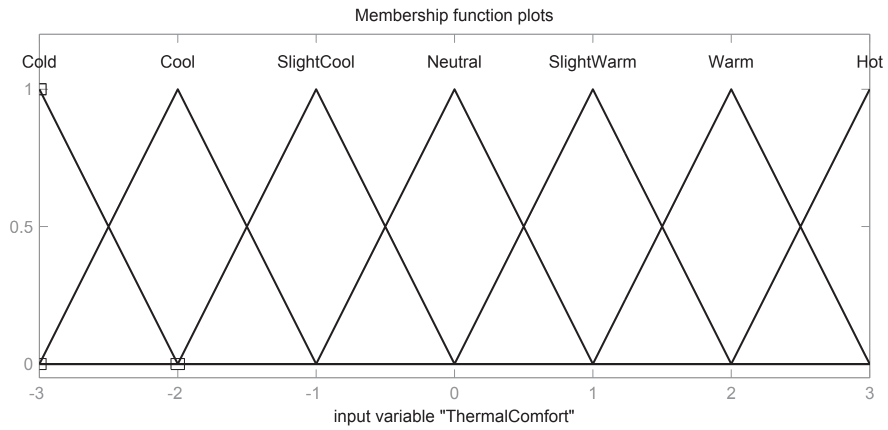


Figure 4-8. Thermal Comfort Input Membership Function

i) *Cold* – represents PMV value ranging from -3 to -2

ii) *Cool* – represents PMV value ranging from -3 to -1

iii) *SlighCool* – represents PMV value ranging from -2 to 0

iv) *Neutral* – represents PMV value ranging from -1 to 1

v) *SlightWarm* – represents PMV value ranging from 0 to 2

vi) *Warm* – represents PMV value ranging from 1 to 3

vii) *Hot* – represents PMV value ranging from 2 to 3

The output of Thermal Comfort FRBS is the Thermal Comfort value (poor–good–excellent) and environment control requests such as fan speed, air condition and heating will be relayed to the actuator control via HC Manager. The output membership function of Thermal Comfort is shown in Figure 4-9. The output membership function was divided as follows:

i) *Poor* – represents a TC value ranging from 0 to 40

ii) *Good* – represents a TC value ranging from 10 to 90

iii) *Excellent* – represents a TC value ranging from 60 to 100

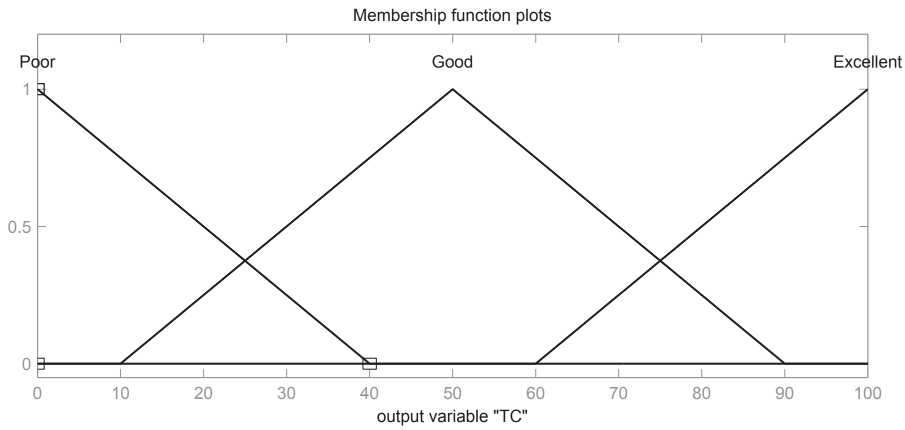


Figure 4-9. Thermal Comfort Output Membership Function

Based on work from several published studies [20, 35, 53, 101, 102], the thermal comfort rules were formulated. The rules devised reflect the PMV value. The Thermal Comfort FRBS node was loaded with seven rules via OTA-Conf as follows:

```
if pmv is cold then tco is poor

if pmv is cool then tco is poor

if pmv is slightcool then tco is good

if pmv is neutral then tco is excellent

if pmv is slightwarm then tco is good

if pmv is warm then tco is poor

if pmv is hot then tco is poor
```

The output is then used as the feeder value for the Human Comfort Index engine.

4.3.2.3 Visual Comfort Knowledge Component

Light is the visible part (visible spectrum) of the electromagnet spectrum. Light rays can be reflected, transmitted or absorbed when they strike an object. Light originates from natural sources such as the sun and from artificial light such as candles, tungsten filaments and gas discharge lamps [96]. These light have a compound effect on the human eye. This work, utilised Sun SPOT's built in light sensor as an ambient light sensor to detect light or brightness in a manner similar to the human eye.

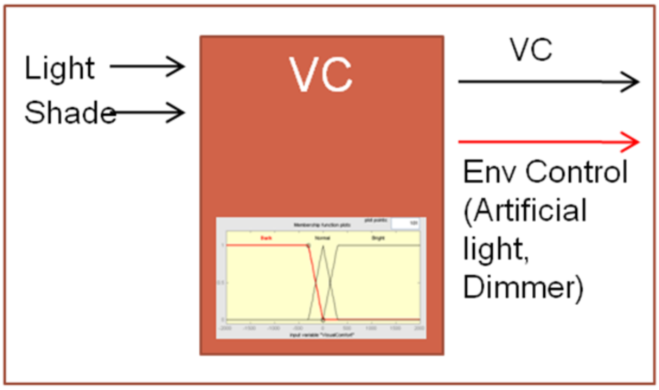


Figure 4-10. Visual Comfort FRBS Node

The input membership function of Visual Comfort FRBS is illustrated in Figure 4-11. Visual Comfort is determined by the luminance level from the Sun SPOT light sensor, ranging from 0 ADC (total darkness) to 700 ADC (1 Lux ≈ 2 x ADC) as shown in Figure 4-11. Based on [96], the membership function of light was divided into three functions as follows:

i) *Dark* – represents light value ranging from 0 to 350

ii) *Normal* – represents light value ranging from 250 to 450

iii) *Bright* – represents light value ranging from 350 to 700

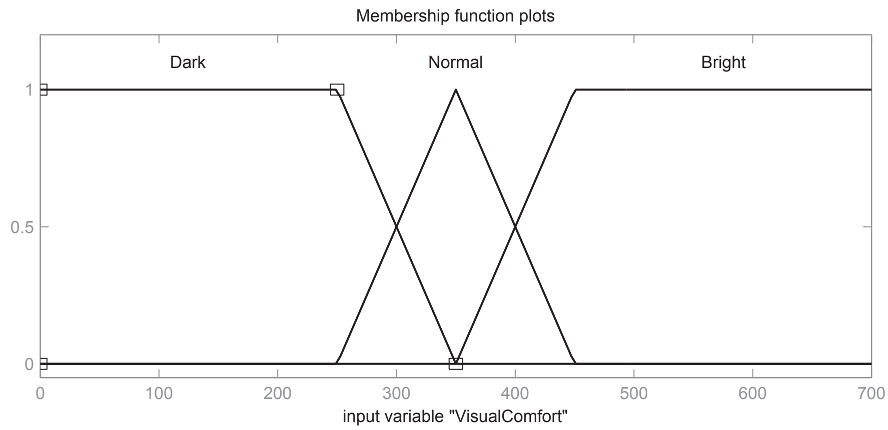


Figure 4-11. Visual Comfort Input Membership Function

The output is Visual Comfort value (poor–good–excellent) and environment control requests such as artificial light and dimmer value are relayed to the actuator control via HC Manager. The output membership function of Visual Comfort is shown in Figure 4-12.

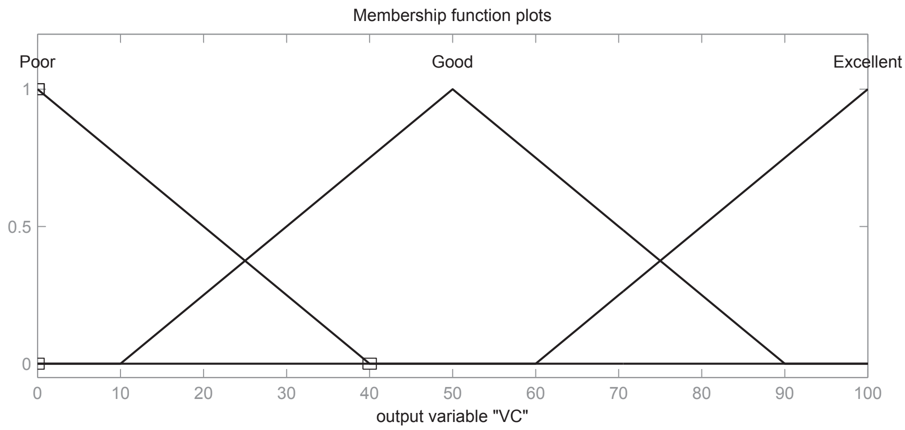


Figure 4-12. Visual Comfort Output Membership Function

Based on work from several published studies [19, 59-61, 96, 103], visual comfort rules were formulated. The devised rules reflect the light level sensed by the Sun SPOT light sensor. The visual comfort FRBS node was loaded with three rules via OTA-Conf as follows:

```
if light is dark then vco is poor

if light is normal then vco is excellent

if light is bright then vco is good
```

The output serves as feeder value to the Human Comfort Index engine.

4.3.2.4 Indoor Air Comfort Knowledge Component

Figure 4-13 represents the Indoor Air Comfort FRBS node. CO_2 was chosen as the gas to be monitored because it is regarded as the most prominent gas and greatly influence indoor air quality.

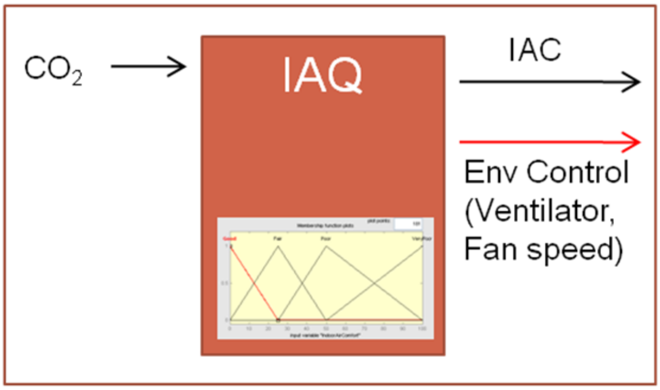


Figure 4-13. Indoor Air Comfort FRBS Node

The Indoor Air Comfort FRBS input membership function is illustrated in Figure 4-14. Indoor air quality is mainly influenced by the concentration of pollutants in the living space where CO_2 concentration (measured in aggregate scale from 0 to 100) was selected as it represents the presence of users as well as various sources of pollutants in the living space [22, 97]. Based on analysis given by references [22, 97], the membership function of CO_2 was divided into four membership sets as follows:

i) *Good* – represents CO_2 value ranging from 0 to 25
ii) *Fair* – represents CO_2 value ranging from 0 to 50
iii) *Poor* – represents CO_2 value ranging from 25 to 100
iv) *VeryPoor* – represents CO_2 value ranging from 50 to 100

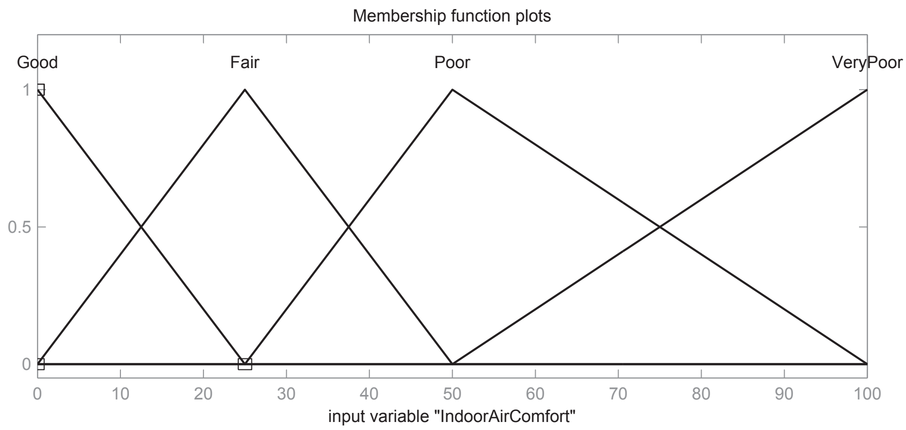


Figure 4-14. Indoor Air Comfort Input Membership Function

The output is Indoor Air Comfort value (poor–good–excellent) and environment control requests such as ventilator and fan speed are relayed to the actuator control via HC manager. The output membership function of Indoor Air Comfort is shown in Figure 4-15.

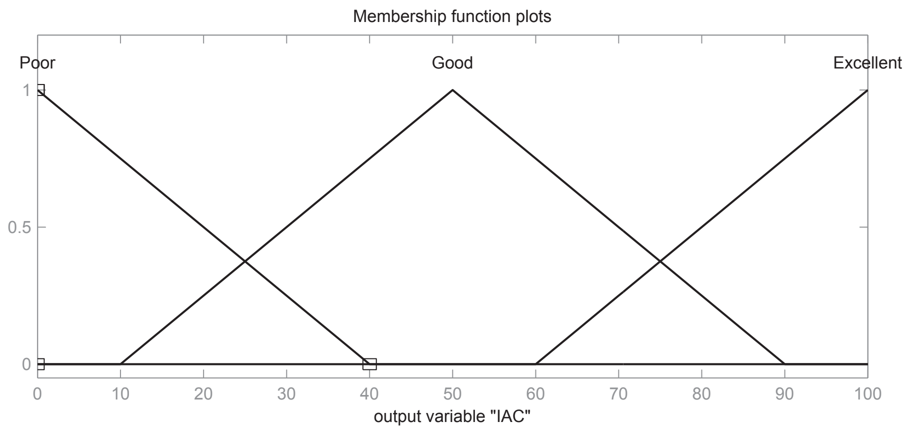


Figure 4-15. Indoor Air Comfort Output Membership Function

The Indoor Air Comfort FRBS node was loaded with four rules via OTA-Conf as follows:

```
if co2 is good then iaco is excellent

if co2 is fair then iaco is good

if co2 is poor then iaco is poor

if co2 is verypoor then iaco is poor
```

The Indoor Air Comfort output is used as feeder value to the Human Comfort Index engine.

4.3.2.5 Acoustical Comfort Knowledge Component

Studies have shown that exposure to environmental noise may increase the risk of health problems such as hypertension, hearing loss and sleep disorders [98]. Ambient noise plays a significant role in human comfort provision.

Figure 4-16 represents the Acoustical Comfort FRBS node. Ambient sound was measured using a UNI-T UT351 Sound Level Meter. UT351 was connected to the Sun SPOT node via DC analogue output to analogue to digital input converter pins (A0) on Sun SPOT.

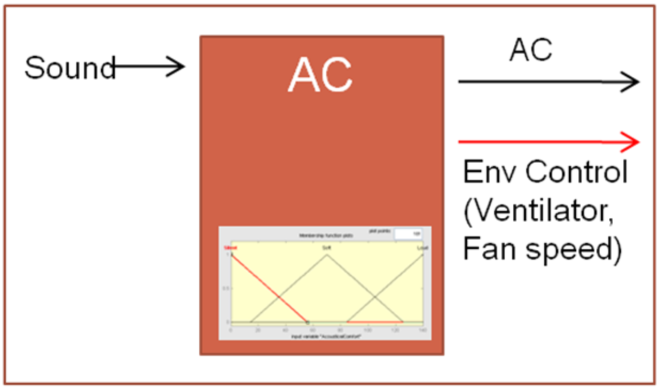


Figure 4-16. Acoustical Comfort FRBS Node

The Acoustical Comfort FRBS input membership function is illustrated in Figure 4-17. Sound level (dBA) from the sound level meter was used, and the membership function of sound was divided into three functions as follows:

i) *Silent* – represents sound value ranging from 0dB to 56dB
ii) *Soft* - represents sound value ranging from 14dB to 126dB
iii) *Loud* – represents sound value ranging from 84dB to 140dB

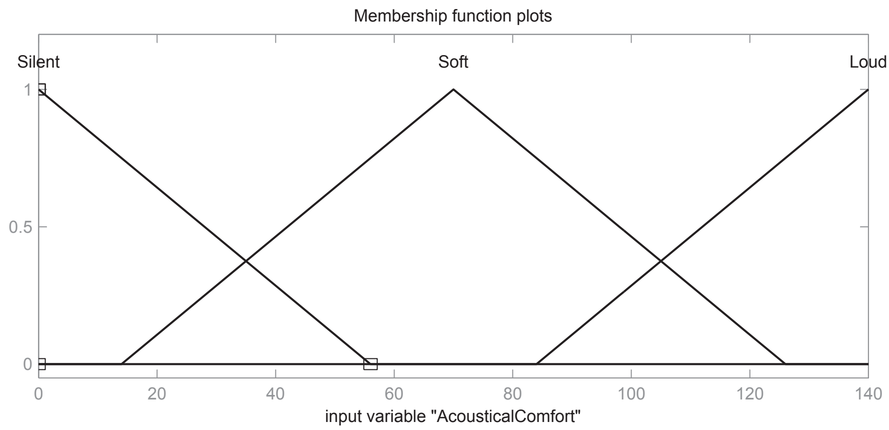


Figure 4-17. Acoustical Comfort Input Membership Function

The output is Acoustical Comfort value (poor–good–excellent) and environmental control requests such as ventilator and fan speed are relayed to the actuator control via HC manager. The output membership function of Acoustical Comfort is shown in Figure 4-18.

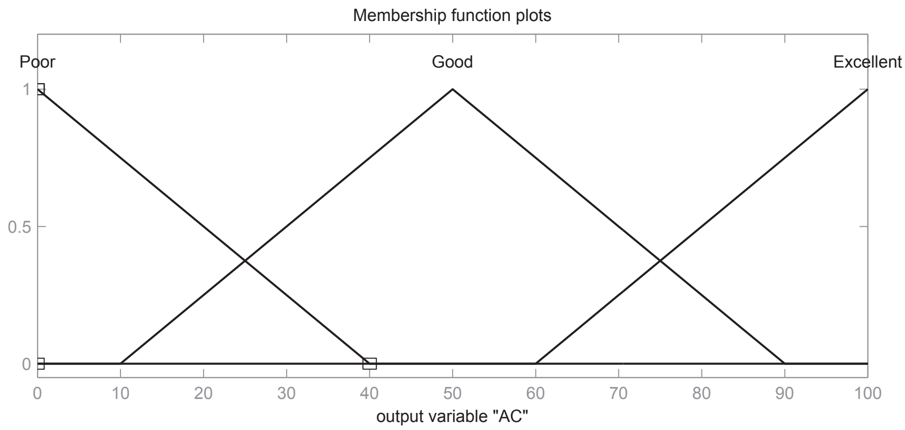


Figure 4-18. Acoustical Comfort Output Membership Function

Based on work from several published works [28, 63, 66, 67, 69-71, 99], acoustical comfort rules were formulated. The devised rules reflect the sound level value. The Acoustical Comfort FRBS node was loaded with three rules via OTA-Conf as follows:

```
if sound is silent then aco is excellent

if sound is soft then aco is good

if sound is loud then aco is poor
```

The Acoustical Comfort output contributes to the Human Comfort Index calculation. The value is used by the Human Comfort Index engine.

As a summary, the output membership function for all TV, VC, IAS and AC also uses similar three membership functions per comfort output (poor–good–excellent). These can have a numerical value of 0 to 100. The three comfort level outputs chosen reflect human sensations of the individual comfort level of a living space. The chosen levels were deemed to be simple enough for a dweller to differentiate between, yet provide the required accuracy.

A total of 17 rules were formulates to compute the individual comfort value, with seven rules for thermal comfort, three rules for visual comfort, four rules for indoor air comfort and three rules for acoustical comfort. These rules form the fewest necessary to cater for each comfort factor whilst providing adequate accuracy as required. The rules were distributed among FRBS comfort nodes.

4.3.2.6 HCAmI Fuzzy Engine

Generic FRBS architecture is presented that caters for the main comfort components including Thermal Comfort, Visual Comfort, Indoor Air Comfort, and Acoustical Comfort. The system is composed of an internet capable sink node and a number of heterogeneous sensor nodes. The sink node acts as an information gateway / interface to the outside world. The whole system consists of multiple sensor nodes where FRBS is executed. The communication protocols that govern the data and information sharing among the nodes are also part of the solution.

A number of techniques were developed to implement intelligence at the sensor level. A variety of artificial intelligence system mechanisms can be used, such as artificial neural networks, fuzzy logic and the hybrid fuzzy neural network system [32, 35, 37, 55, 72, 73, 78]. In WSNs, several embedded artificial intelligence models have been used and researched. Although some studies have presented sensors that utilise fuzzy logic such as Benoit et al. [104], little attention has been given to knowledge based sensors. Moreover, limited attention has been given to embedding a fuzzy rule based system (FRBS) into WSNs.

In this research, a complete FRBS, adapted to a 32bit sensor, has been designed. Each sensor node on the network is capable of a complete localised FRBS execution. Each comfort node is FRBS capable. Thirteen core java classes were coded for FRBS on Sun SPOT WSN implementation. Each comfort node comprises the following components as shown in Figure 4-19 [95]:

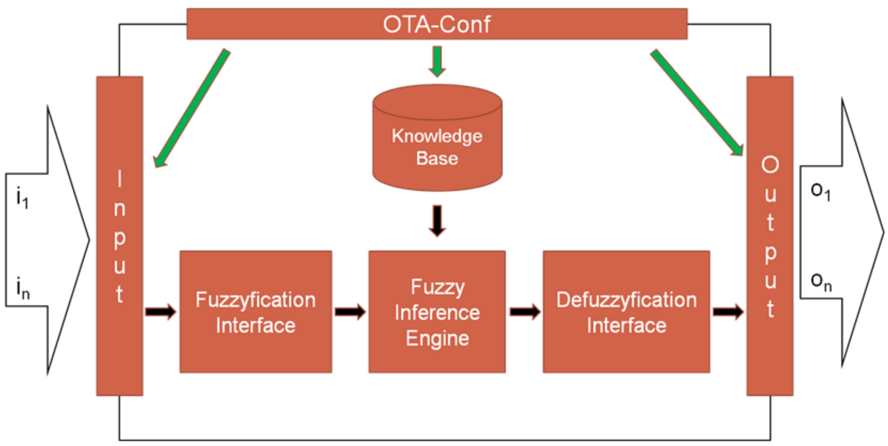


Figure 4-19. FRBS Structure

i) A scalable input ($i_1 . . . i_n$) and output ($o_1 . . . o_n$) variable(s) that can be configured over the air via the Over The Air Configuration (OTA-Conf) function. An unlimited number of input / output variables can be created by instantiating `LinguisticVariable` class. Each `LinguisticVariable` object is associated with an unlimited number of membership functions.

ii) The Fuzzyfication Interface (FI) transfers the crisp value of input variables into fuzzy value, assigning grades of membership to each fuzzy set involved for the variable. The `LinguisticVariable` object that was created will be supplied with crisp input value via the `setInputValue(inVal)` method.

iii) Fuzzy Inference Engine (FIE), which infers fuzzy action through fuzzy implications and the rules involved. Upon configuring the input and output variables, a fuzzy engine object will be instantiated from `FuzzyEngine` class. All variables and rules will be associated with the engine via the `register(lVarObj)` method.

iv) The Knowledge Base (KB) in the form of IF-THEN fuzzy rules, contains a linguistic terms set that represents expert knowledge in the following form:

```
IF X IS A THEN Y IS B
```

where X – input variables, A – fuzzy sets are related to the input variables, the Y – output variable and B – fuzzy set are related to the output variable.

The KB consist of data, rules based, variable definitions for each comfort factor and defined fuzzy sets for each variable and rules. The KB was described in pre-formatted plain text, simplifying FRBS definition, modification and management activity. The prepared KB description was transferred to the selected node over the air via the Over The Air Configuration (OTA-Conf) function as needed.

v) Defuzzyfication Interface (DI) that converts fuzzy value to a single crisp value / number. DI was implemented in `LinguisticVariable` class and called via `defuzzyfy()` method.

vi) The Over The Air Configuration (OTA-Conf) function is responsible for modifying the KB (variables, fuzzy sets and rules) over the air. Therefore, with this facility, each FRBS capable node starts with a ‘blank’ FRBS that can accommodate any form of fuzzy implementation. This is where the novelty of the HCAmI system shines, extremely flexible yet very capable of delivering complex computational tasks within the node itself.

For the FIE method, Mamdani's style inference method [105] was chosen because it is the most commonly used in applications, widely accepted for capturing expert knowledge and its simple structure of 'min-max' operations. It allows us to describe the expertise in a more intuitive, human like manner and is well suited to human input [106].

The Mamdani style fuzzy inference process is performed in four steps:

i) Fuzzification of the input variables
ii) Rule evaluation
iii) Aggregation of the rule outputs
iv) Defuzzification

The working set for the Mamdani style for the HCAmI system is shown and discussed in Chapter 5

4.3.3 HCAmI System Manager

HCAmi System Manager consists of two subsystems: *HC Manager* resides in the host machine and the *Human Comfort Index* resides in the basestation. The HC Manager and Human Comfort Index communicate via a connected USB cable. Figure 4-20 illustrates HCAmI System Manager components.

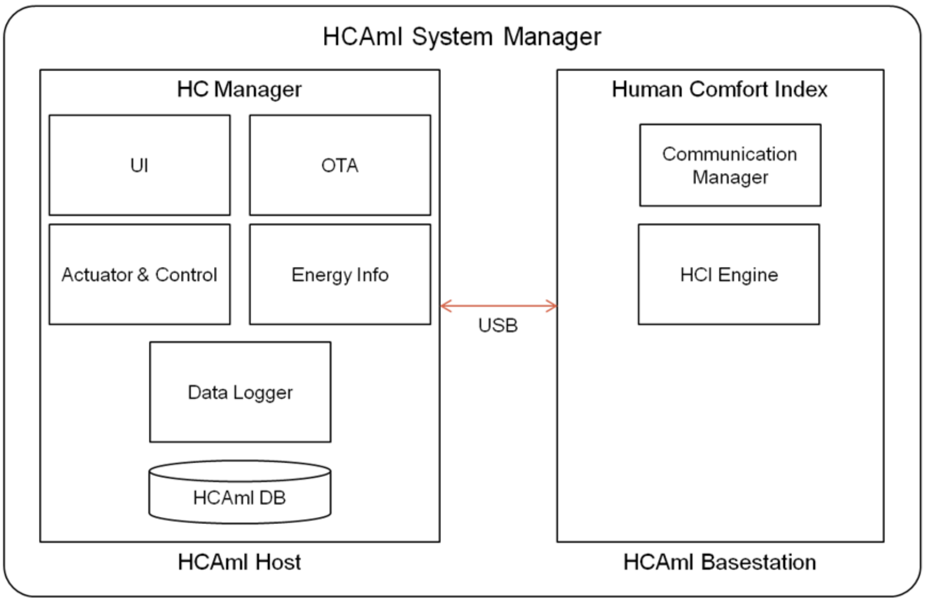


Figure 4-20. HCAmI System Manager

The HC Manager and Human Comfort Index subsystems are inseparable even though they reside in different physical entities. They work as one unit in the HCAmI System.

4.3.3.1 HC Manager Subsystem

The HC Manager subsystem was designed as an entry point for HCAmI System sensor cloud activity. It serves as a gateway from inside HCAmI sensors to the outside world. Ideally, it resides on a controller host / PC or any other high powered device such as the Raspberry Pi [107] (a credit card sized single board computer). The controller host should be capable of communicating with other non WSN systemss such as actuator control, energy expenditure reading and feedback, and occupant preference user interface.

HC Manager features and capabilities:

i) Gateway for HCAmI System sensor cloud to outside world, where all incoming and outgoing data and information traverses through HC Manager.

ii) Occupant Preference, Actuator Control, Energy Usage, Real-time Display control ready. Sorts out the human computer / machine interface task. Interprets real world input into machine friendly forms such as occupant feedback towards environmental conditions at one moment.

iii) Data and information repository that serves as long term storage of acquired data for later use. Over a period of time, one can use certain data mining techniques to acquire hidden information from the heaps of data available.

iv) Configuration master for sensor cloud via OTA-Conf. This feature enables over the air configuration and modification of any sensor node knowledge component, such as the modification / fine tuning of fuzzy rule based and membership function of a specific node.

v) Sensor deployment master. Deploys application into sensor node and is in charge of physical node monitoring such as battery level and application execution states.

vi) Offline data analysis. Shows data trends and history for living space human comfort and individual comfort. Might be used for higher intelligence system training and fine tuning.

vii) Scalabilities, uncomplicated to configure and redeploy. The modular system design eases the deployment and configuration task.

The key functionality of each component as shown by Figure 4-20:

i) *UI*: Serves as a human friendly interface for the system. Shows recent human comfort index, individual indexes, actuator and control status, and energy expenditure information to the user.

ii) *OTA*: Plays a crucial role for disseminating new / updated knowledge component(s) into designated node(s). The component is also responsible for installing, removing and updating applications that run in the nodes. Any OTA command / task will be transferred to HCAmI Communication Manager in the HCAmI Basestation for distribution via the 802.15.4 radio component.

iii) *Actuator and Control*: Functions as I/O for controlling and monitoring the mechanisms of environmental manipulation apparatus such as air ventilation, heating / cooling, lighting level and shades and window controls.

iv) *Energy Info*: Monitors energy expenditure of monitored dwelling.

v) *Data Logger*: Manages data and information gathered from sensor nodes. Keeps unprocessed data and processed information in permanent storage.

vi) *HCAmI DB*: Permanent storage for HCAmI system data and information such as the human comfort index, individual comfort values, and basic environmental data. Either flat file or relational database schema can be used for storage.

Note: Actuator and Control, and Energy Info components are beyond the scope of this thesis.

4.3.3.2 Human Comfort Index Subsystem

The Human Comfort Index subsystem consists of two important components: *HCAmI Communication Manager* and *HCI Engine*.

HCAmI Communication Manager:

Two major roles are entrusted to HCAmI Communication Manager. One is to serve as a system wide communication backbone for application deployment, application upgrades, and application monitoring of sensor nodes. The role is platform specific and solely depends on the readymade mechanism offered by the chosen WSN platform.

The other is to perform application level communication functions such as data sink / endpoint for all acquired data and information from sensor nodes. A sensor value sharing and distribution mechanism was implemented with this protocol. System specific custom design protocols are used to perform the undertaking.

HCI Engine:

Research on HCI has been done over a long time. It started with a simple index based on an effective temperature scale, using dry and wet bulb temperatures along with wind speed. Houghton and Yaglou coined the term “effective temperature” when this index was established, as a method for determining the relative effects of air temperature and humidity on comfort [108]. In [109-113], similar indexes were mainly based on earlier studies, resulting in different indexes to describe human comfort. Now, HCI can be looked as a corporation of multitudes of factors such as bio-meteorological parameters and physical environment parameters. Complex bio-meteorological parameters consist of ambient vapour pressure, the human dimension, the effective radiation area of skin, clothing, internal body temperature, surface temperature and vapour pressure of skin and clothing, activities, effective wind speed, clothing resistance to heat and moisture transfer, skin surface radiation, skin surface convection and sweating rate. As complicated as bio-meteorological parameters, physical environment parameters deal with illumination, ambient sound, spatial and air quality.

All comfort indices are mathematical models designed according to physiology and the theory of body surface heat balance, along with the human sensation of comfort. The comfort level of an individual varies directly with the specific location and people involved. There is no general model for a comfort index. The HCAmI system was designed to accommodate multiple comfort factors as HCI. Each individual comfort factor was considered to have some degree of influence towards HCI by means of a specific weighting. With multiple comfort factors involved, there is an imperative for a quantitative measurement of overall comfort; hence, HCI has been introduced here to serve as a human comfort pointer or indicator of the sensed environment.

HCI is based on the weighted average of comfort factor values (*C*) for the sensed environment during a given interval of time, as shown in Equation 4–2. Based on work by Reffat et al. and Graça et al. [30, 103], the initial weight (*W*) given to TC, VC, IAC and AC is 0.5, 0.3, 0.1 and 0.1, respectively. TC was given the largest weight because the human body is greatly influenced by thermal comfort factors [93]. One can tolerate other discomfort well, compared with discomfort due to extreme thermal comfort. These weight values can be reconfigured / changed later on, based on user preferences and the system learning process via the system personalisation function (which is beyond the scope of this thesis).

$$
\begin{aligned}
HCI &= \frac{\sum_{i=1}^{n} C_i W_i}{\sum_{i=1}^{n} W_i} \\
&= \frac{(C_1 W_1 + \cdots + C_n W_n)}{(W_1 + \cdots + W_n)} \\
&= \frac{C_{TC} W_{TC} + C_{VC} W_{VC} + C_{IAC} W_{IAC} + C_{AC} W_{AC}}{W_{TC} + W_{VC} + W_{IAC} + W_{AC}}
\end{aligned}
$$

Equation 4–2

The parameters are defined as follows:

C_{TC} Thermal Comfort value

C_{VC} Visual Comfort value

C_{IAC} Indoor Air Comfort value

C_{AC} Acoustical Comfort value

W_{TC} Thermal Comfort weight

W_{VC} Visual Comfort weight

W_{IAC} Indoor Air comfort weight

W_{AC} Acoustical Comfort weight

For simplicity, the HCI has been classified into four categories in Table 4-1 [89]. The maximum HCI value is 100, and minimum HCI value is 0. HCI between 50 and 100 is defined as the comfort zone where HCI 75 – 100 is very comfortable and HCI 50 – 74 fairly comfortable.

On the other hand, an HCI less than 50 indicates discomfort, where HCI 25 – 49 is fairly uncomfortable and HCI 0 – 24 is very uncomfortable.

HCI	Human Sensation
75 – 100	Very comfortable
50 – 74	Fairly comfortable
25 – 49	Fairly uncomfortable
0 – 24	Very uncomfortable

Table 4-1. Human Comfort Index Classification

Based on Equation 4–2, an HCI algorithm was formulated.

HCI Algorithm:

i) Start
ii) Reset variables A = 0, B = 0, n = 0 and i = 0
iii) Get the number of comfort factor involved n
iv) Get comfort value i C_i
v) Get weight for i W_i
vi) $A = A + (C_i * W_i)$

vii) $B = B + W_i$

viii) $i = i + 1$

ix) If i = n, go to x else go to iv

x) Calculate HCI = A / B

xi) Stop

The HCI engine was coded based on a corresponding flowchart, shown in the Appendix D HCI Flowchart.

4.3.4 HCAmI Communication

HCAmI Communication component is a piece of system component that is crucial in integrating all nodes. Even though it is not involved directly in the computational aspect of HCAmI, it provides a seamless application level communication protocol. It is a necessary component in the HCAmI system; therefore, a simple HCAmI Communication Protocol (HCAmICP) was developed, as shown in Figure 4-21.

To simplify development and testing, this application level communication protocol was implemented as broadcast messages. HCAmICP was used as a multi-function protocol. HCAmICP serves as a transport for the HCAmI sensor cloud. It carries both data and information such as raw environmental data, comfort information and node technical status.

The first byte of the datagram contains information regarding the datagram's data type. The byte has a value of:

i) 11 for thermal comfort operation / communication

ii) 12 for visual comfort operation / communication

iii) 13 for indoor air comfort operation / communication

iv) 14 for acoustical comfort operation / communication

v) 40 for Sun SPOT status / operation

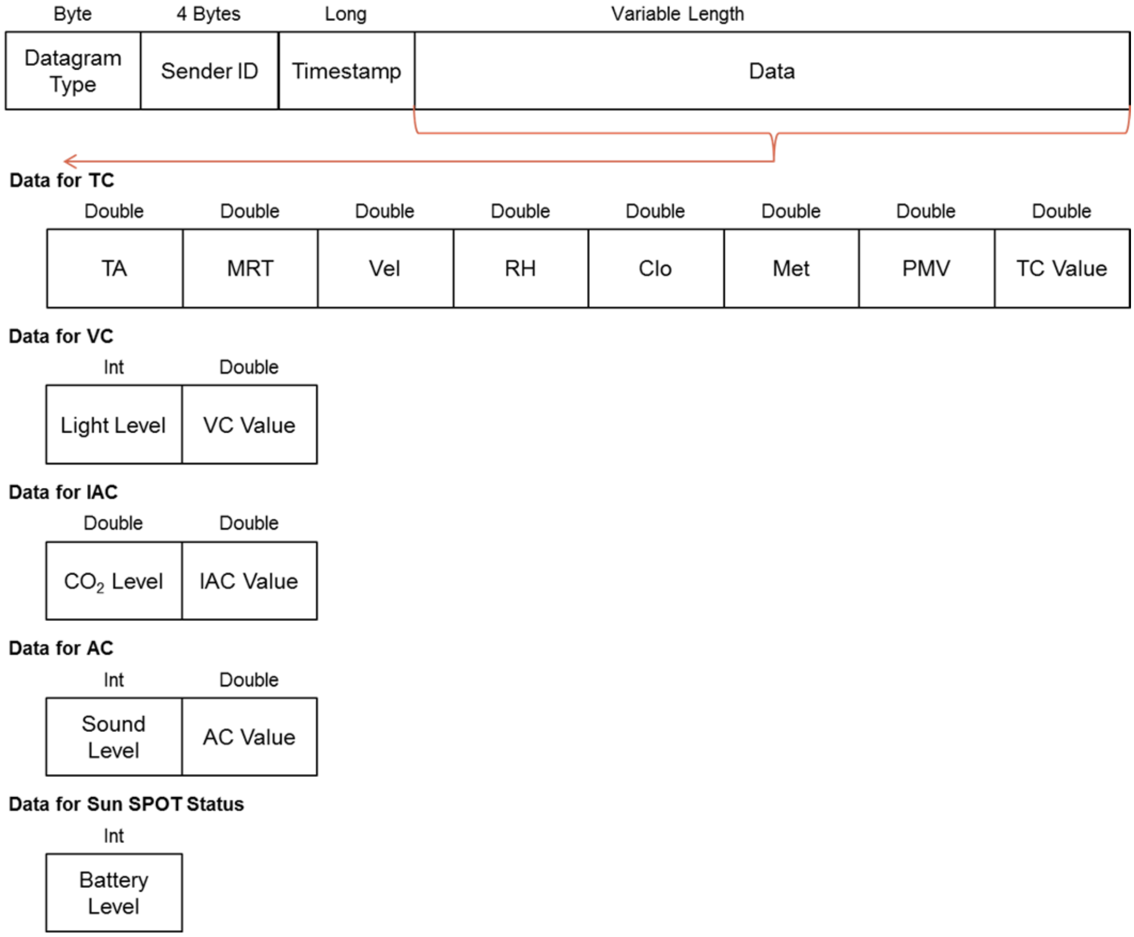


Figure 4-21. HCAmI Communication Protocol Format

Next is the 4 bytes sender ID in the form of a MAC address for the sender node. This unique ID will serve as an identification number for any datagram being broadcast in the HCAmI sensor cloud.

A long number follows that serves as the datagram timestamp. This value is used to identify the temporal value of the data. The exact date and time in the UNIX time format is recorded.

Finally, the data portion of the datagram is a variable length data payload. The length and definition depends on the first byte. It can be data for thermal comfort, which carries a raw sensor reading, PMV and TC value as shown in Figure 4-21. Other data types are also shown and carry their own detail significance.

In a sending operation, after a node completes its sensing operation, it will then compose a message to be broadcast. To begin with, a message type will be selected (e.g. a byte value of 11 for thermal comfort data). A current time stamp will be inserted. Finally, all available thermal comfort data are added. The complete message will then be passed to a lower communication library that will physically broadcast the message throughout the HCAmI cloud nodes.

Meanwhile, from a receiving operation point of view, when a node receives a message, the node communication library will read the first byte. Based on the packet data type definition, the process will pass the remaining message to the appropriate handler or ignored altogether. Each node, with its own specific task, will digest the rest of the message accordingly. For example, with a message for thermal comfort received by a thermal comfort node, the communication stack will decompose the message and extract all available data / information. The data will then be populated within a thermal comfort object for further processing, such as calculating the PMV value, or for permanent storage in the database.

The versatility of the protocol makes it expandable and configurable based on any additional requirements. New comfort factors can be added and new node technical information can also be added.

4.4 Chapter Summary

This chapter provides the theoretical aspect of HCAmI, starting with smart home environment requirements and perspectives. HCAmI as smart home enabler was viewed from energy centric activity perspective and human centric perspective. A few different but related AmI system such as AAL and WHAN were reviewed.

Then detailed descriptions of each HCAmI framework and architecture component are discussed in term of Aldrich's five class hierarchy of smart home. HCAmI design and implementations as distributed WSN system that allows the execution of a set of application for sensing, FRBS and communication were presented.

Key functionality of each component / subsystem involved is described in detail such as:

i) Interpolation technique using Shepard's method in determining a quantity at the occupant's location.
ii) The algorithm for computing PMV (PMV Engine) and thereby establishing the thermal comfort via an FRBS.

iii) Thermal comfort, visual comfort, indoor air comfort and acoustical comfort knowledge components (membership functions and fuzzy rule base).

iv) Description of HCAmI Fuzzy Engine.

v) HCAmI System Manager that consist of two subsystems – HC Manager and Human Comfort Index (HCI Engine) that serves as the final point in concluding HCAmI System sensing activity.

vi) HCAmI Communication as a piece of system component that is crucial in integrating all nodes communication activity.

In general, this chapter presents original material on HCAmI concept design and architecture.

Chapter 5 HCAmI Implementation

5.1 Introduction

In this chapter, we look in detail at each HCAmI component / module modelling, simulation and implementation process. HCAmI components are designed to be embedded and run on limited resource (computational power, available memory and power source) nodes of WSNs. It is built on the basis of task distribution over multiple WSN nodes. HCAmI is implemented as an automated sensor cloud that feeds on available raw data from a monitored environment. Its primary goal is to acquire raw environmental data and process it into meaningful information to be passed to space occupants and space control mechanisms. Strategically, the data is processed within the WSN cloud and sends only necessary information out, minimising data traversing on the network towards the outside world and consequently extending the network's lifetime (especially for cluster head and sink).

MATLAB, NetBeans, Sun SPOT Manager Tool and Solarium have been used as the primary tools for HCAmI system development, modelling, simulation and implementation. The PMV, FRBS, HCI, Communication protocol and library, and Sensor Library (SHT15, Precon, and UNI-T) were modelled and individually tested before whole system wide implementation was done. The individual model implementation and testing results are presented.

5.2 Modelling, Simulation and Implementation

HCAmI is separated into individual components / module based on the HCAmI architecture presented in Chapter 4. First, each component was modelled with clear abstraction from reality, resulting in formal specifications of a conceptualisation and underlying assumptions and constraints. At this stage, everything was still on paper, nothing tangible was produced. The process of refining the model was repeated until satisfactorily sound. The sanity of the model was checked against possible shortcomings.

Next, based on the firm model produced, individual components were assembled using a suitable simulation platform, for example the gas sensor reading was simulated using Sun SPOT Solarium, the PMV engine was simulated using MATLAB and each individual FRBS were simulated using MATLAB Simulink. Each component was simulated rigorously, with real and simulated data sets. This stage represents a thorough understanding of the model formed. Any flaw or defect could be rectified and resolved.

Finally, with sensible simulation results, conceptualisation and implementation of HCAmI were explored. HCAmI components were implemented into the corresponding Sun SPOT platform. At this stage, a small scale complete HCAmI was constructed to make sure that the developed system was aligned with the system design and architecture. During this phase, the component was repeatedly tested against simulations and fine-tuned accordingly. Real world implementation, such as code incompatibility, lack of API / library, and hardware issues (circuit and external sensor), was dealt with at this phase.

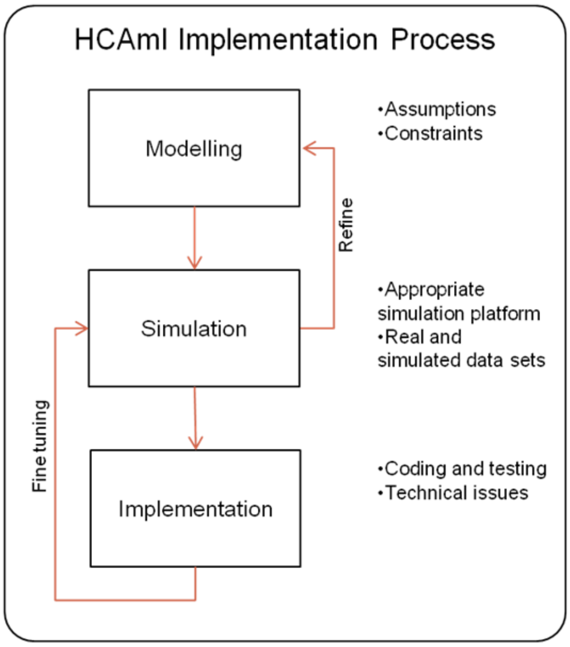


Figure 5-1. HCAmI Implementation Process

Figure 5-1 shows an overview of the HCAmI implementation process. Each HCAmI component / module was put through the process repeatedly until it performed as intended.

5.3 Spatial Sensing Engine Operation

Since node location and position determination is a complex matter and has its own research domain, it was assumed that every node involved knew its location respectively. PMV for Thermal Comfort determination was selected as a test case, in building the spatial sensing engine. PMV was selected because it is the most complicated of all the human comfort factors. A simplistic living space was modelled, as shown in Figure 5-2.

In this case, four static Environmental Parameters Nodes (sensing DBT, MRT, RH and Vel) were placed approximately 1.1 to 1.6m above the floor in the vicinity of four living space walls. Since interest is in the conditions experienced by the occupants of a living space, the placement height of the sensor chosen should be representative of the occupant's height, reflecting his / her sensory experience.

Then a mobile Physiological Parameters (sensing Met and Clo) node was positioned in the middle of the living space. In this case, the distance between the nodes was determined manually and in a real world implementation, we assume the node position would be handled automatically.

In this scenario, we would like to determine the PMV at node N0 (the living space occupant). At node N0, we know Met and Clo, on the other hand, the DBT, MRT, RH and Vel values are unknown. Here, a spatial sensing engine will be used to determine the missing values.

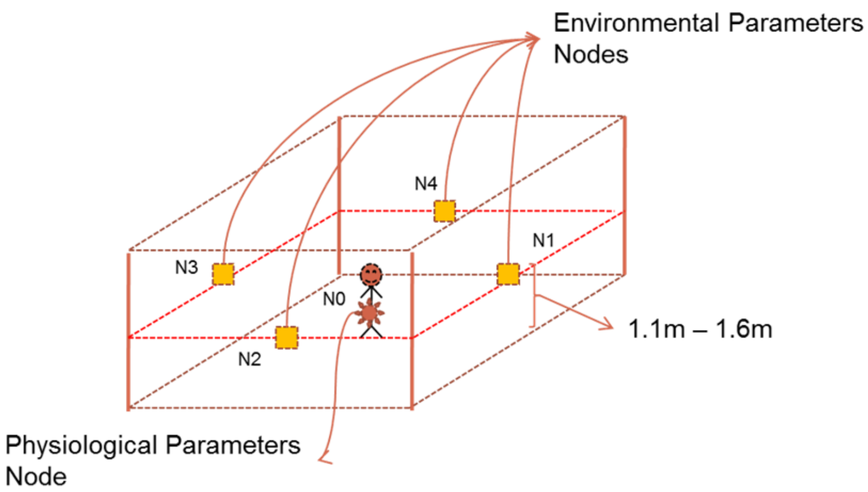


Figure 5-2. Simplified Living Space

Initially, the engine algorithm and operation were constructed and tested with a plain Java based application where raw TC values for N1, N2, N3, N4 and N0 were keyed in manually. Distances between N0 – N1, N0 – N2, N0 – N3 and N0 – N4 were supplied as well. A number of data sets were entered into the application. The calculated output was verified manually using Microsoft Excel.

Thermal Comfort Parameters	Nodes				
	N1	N2	N3	N4	N0
DBT (°C)	24	23	21	22	22.50
MRT (°C)	27	25	20	21	23.41
RH (%)	50	57	60	54	55.94
Vel (m/s)	0.1	0.1	0.2	0.1	0.14
Met (met)	-	-	-	-	1
Clo (clo)	-	-	-	-	1
				PMV at N0 =	-0.16

Table 5-1. PMV Value At Location N0

Table 5-1 [93] summarises the calculation results of a typical living space simulation with four Environmental Parameters nodes and one Physiological Node. The distances between N0 and N1, N2, N3 and N4 are 3.04m, 1.58m, 2.24m and 3.64m respectively. The calculated PMV value at N0 is equal to -0.16 (Neutral).

The engine was further subjected to various random N0 locations to determine the PMV values in the living space. This exercise simulates various locations the occupant could be and reflects his / her thermal comfort sensory experiences.

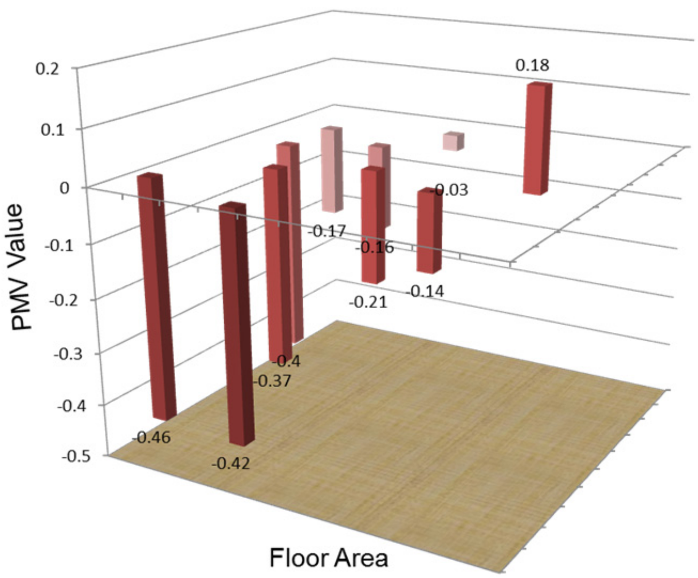


Figure 5-3. PMV Value at Various Locations

Figure 5-3 [93] shows PMV value at various locations in a living space. PMV value varies from -0.46 to 0.18. The most comfortable spot is where the PMV value is 0.03 and the least comfortable spot is where the PMV value is -0.46. Overall, it shows that the simulated living space is leaning towards the "Slight Cold" region due to the negative value of PMV.

From this simulation, one can seek the most comfortable spot in a living space and the sensor nodes can recommend to the occupant the best place for thermal comfort. The area with poorest thermal comfort can also be identified and the sensor nodes can recommend the living space environment sub system to make the necessary adjustment to rectify it.

Upon a satisfactory simulation result, the spatial sensing engine is coded into a universal form. The spatial sensing class library (`SpatialSensing` class) was coded and implemented inside the Sun SPOT node.

Spatial Sensing Algorithm (for sensor x):

i) Get neighbouring node list with sensor x, N_1, N_2 ... N_n

ii) Get sensor x values, X_1, X_2, … X_n

iii) Get node distances, d_1, D_2, … d_n

iv) Compute sensor x value at this location X_0

The spatial sensing class can then be used to calculate any sensor value, provided there is a neighbouring node that can share their sensed value.

In conclusion, the engine provides a mechanism for a sensor node to determine the sensed value where no sensor node is available.

5.4 PMV Engine Function

The PMV engine was modelled based on Equation 2–1 and detailed in the Appendix C PMV Flowchart. Six factors served as raw data input: DBT, MRT, RH and Vel indicate Environmental parameters and Met and Clo as Physiological parameters of a PMV calculation. It was assumed that all values are available to the engine. The engine was constructed using MATLAB and tested with Simulink for interactive and pre-

prepared data sets to test the engine validity. Figure 5-4 shows the Simulink model of PMV engine.

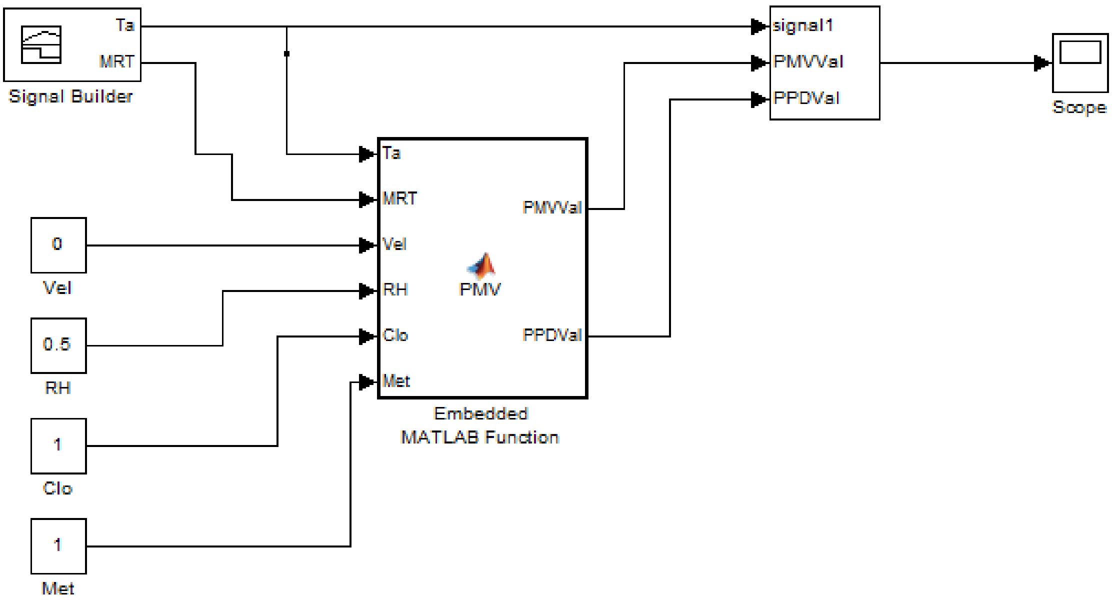


Figure 5-4. PMV Simulink Model

The PMV engine was simulated with a combination of pre-set data and a signal builder. One factor is tested while others remain constant to see the effect on PMV value. Due to the complexity of Equation 2–1, it is not possible to plot a single 3D graph to show the multiple parameters at work. Figure 5-5 shows the complex relationship between parameters.

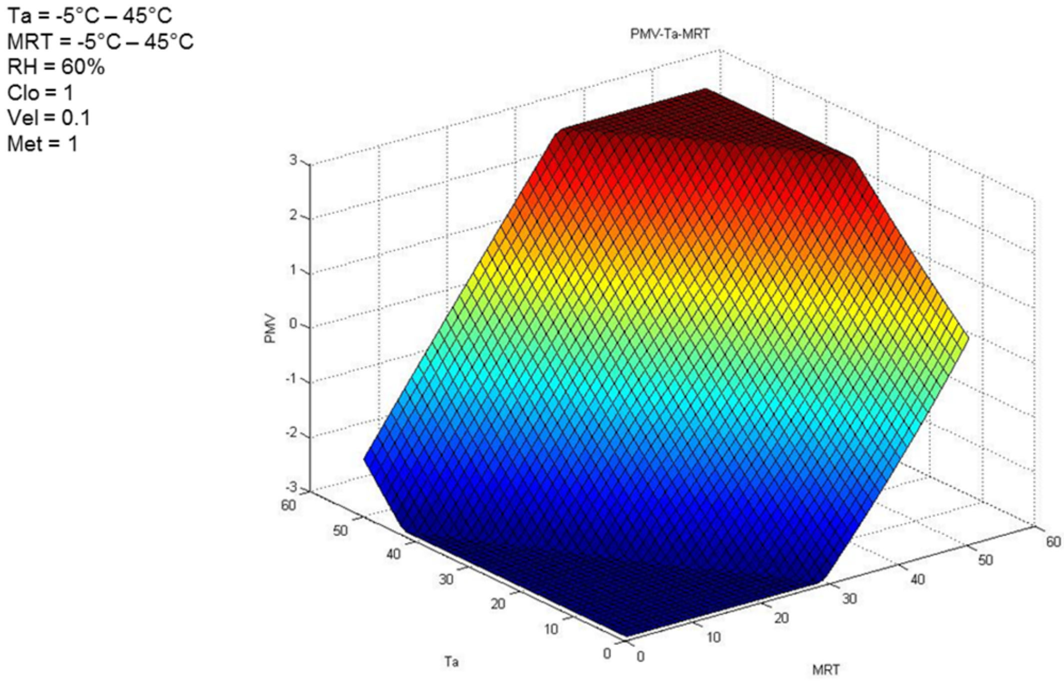


Figure 5-5. PMV DBT - MRT Relationship

Figure 5-5 shows the surface area of a range of DBT / Ta (-5°C to 45°C) and MRT (-5°C to 45°C) with a static value of 60% RH, 0.1 Clo, 0.1m/s Vel and 1Met.

Figure 5-6 illustrates a range of calculated PMV values based on a series of fixed parameters against one parameter. Figure 5-6 (RH Varies) shows PMV value changed from 0 to 1 with RH from 0% to 100% for fixed Ta = 25°C; MRT = 25°C; Clo = 1; Vel = 0.1; Met = 1. RH can be seen to not have any significant impact on PMV.

On the other hand, Figure 5-6 (Clo Varies) demonstrates the effect of clothing (from naked to thick cloth) on PMV with fixed Ta = 25°C; MRT = 25°C; RH = 60%; Vel = 0.1; Met = 1. From this graph, it is safe to deduce that a person's clothing has a substantial impact on his / her thermal sensation.

Figure 5-6 (Vel Varies) indicates PMV value for fixed Ta = 25°C; MRT = 25°C; RH = 60%; Clo = 1; Met = 1. Air velocity from 0m/s to 2m/s has some effect on thermal sensation, but the effect diminishes with higher air velocity. In other words, it is a waste of energy to increase air velocity for the sake of 'cooling'.

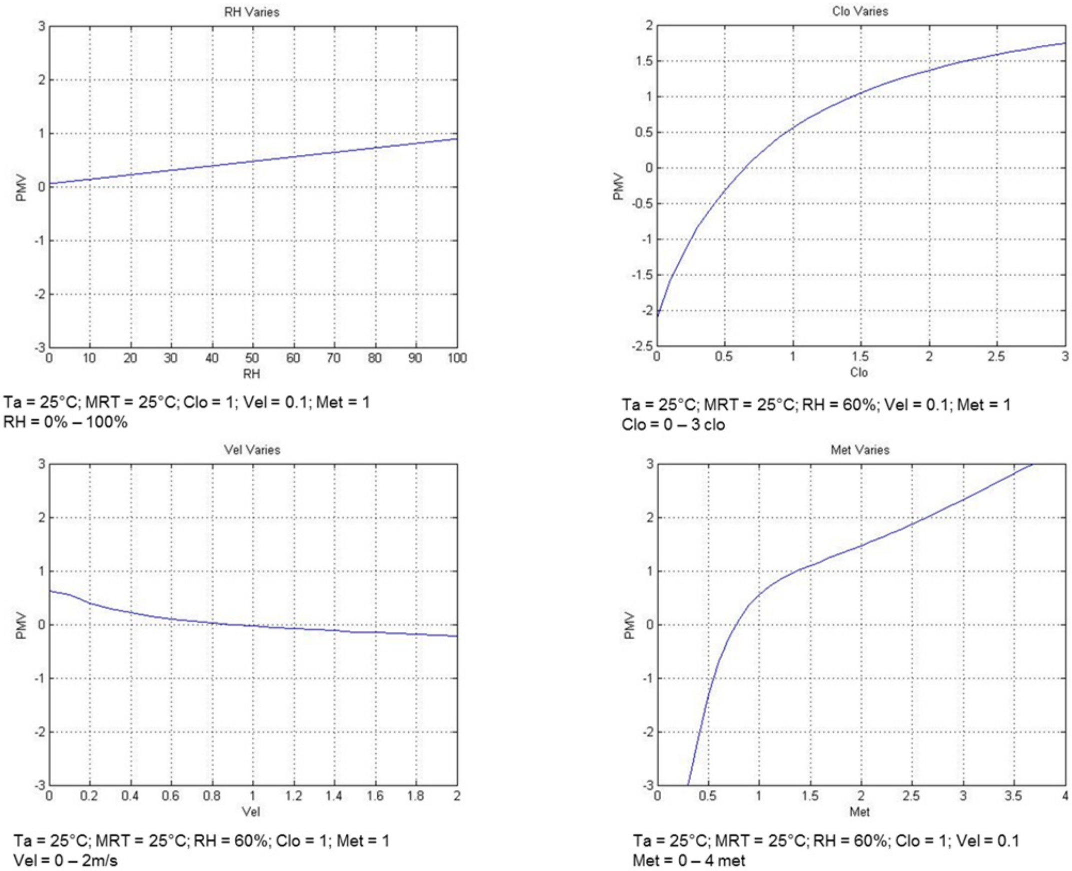


Figure 5-6. PMV with Different Parameters Values

Finally, Figure 5-6 (Met Varies) shows the effect of human activity (metabolic rate) for fixed Ta = 25°C; MRT = 25°C; RH = 60%; Clo = 1; Vel = 0.1. One's activity does play a big role in thermal sensation. There is a wide difference in thermal sensation for one sedately sitting from one doing rigorous exercise, even though other environment parameters stay the same. As a consequence, when there is more than one occupant in a living place, one can feel cold while the other can feel warm, partly due to their current metabolic rate.

As shown in Figure 5-5 and Figure 5-6, it can be deduced that temperatures (DBT and MRT) and a person's metabolic rate (Met) and clothing (Clo) play a significant role in his / her thermal sensation. On the other hand, air velocity (Vel) and humidity (RH) do effect thermal sensation but have a lesser impact.

The PMV engine was then coded in Java as `PMV` class. A simple Java application for Sun SPOT (*PMVThermalComfort Midlet*) was built to test the engine. The engine was subjected to the same datasets as used in MATLAB and Simulink previously. Upon a satisfactory PMV engine outcome, the engine was incorporated into the Sun SPOT node.

PMVThermalComfort Midlet (Sun SPOT binary executable) embedded in Sun SPOT comprises of three classes: the main class *PMVThermalComfort.class*, the *PMV.class* and connected sensor driver class (e.g. *SHT.class* for the SHT15 Humidity and Temperature Sensor). *PMV.class* is coded based on the Appendix C PMV Flowchart.

From a Sun SPOT operational point of view, every ten minutes, *PMVThermalComfort.class* will:

i) Acquire the time stamp (current date and time)
ii) Read attached sensor values and acquire sensor value from neighbouring node if necessary.
iii) Calculate PMV value (performed by *PMV.class*)
iv) Indicate PMV value on Sun SPOT on board Tri Colour LED. LED Colour varies from Blue for PMV = -3 to Red for PMV = 3.
v) Prepare a datagram (<date time><DBT|MRT|RH|Vel|Met|Clo><PMV>)
vi) Transmit the datagram to the Sun SPOT basestation
vii) Put Sun SPOT nodes into deep sleep mode for ten minutes to conserve energy.

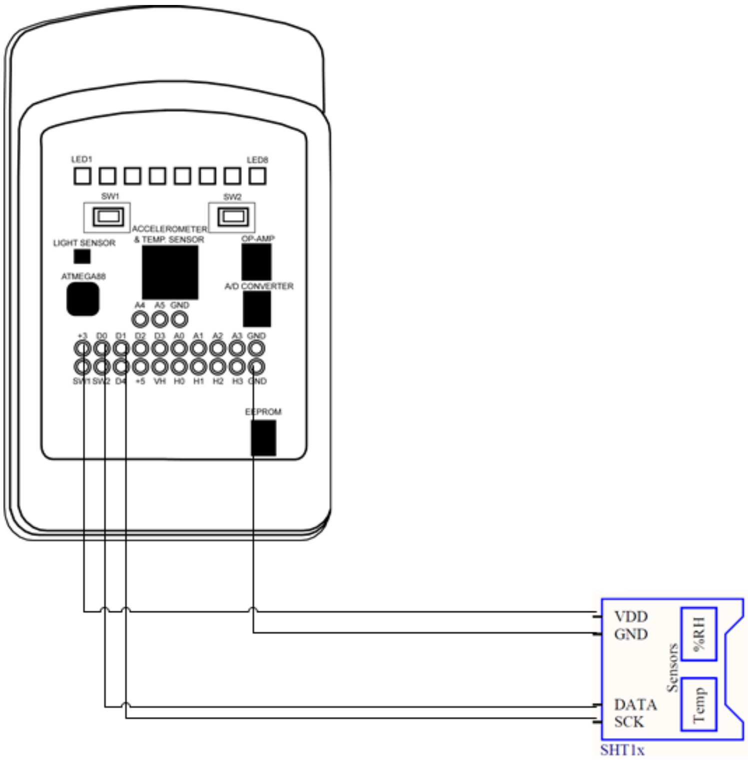


Figure 5-7. PMV Single Node

A Sun SPOT node with a PMV engine is subjected to real data testing. Two run-through scenarios were created as a test case.

i) Single node – One Sun SPOT node connected to multiple sensors. In this scenario, all thermal comfort factor values come from sensors attached to it. The node acquires a value either from its built in sensors or through external sensors connected via its GPIO / analogue pins. Figure 5-7 shows Sun SPOT connected to an external SHT15 Humidity and Temperature Sensor. As an alternative, Sun SPOT can also be connected to the Precon HS-2000V Humidity and Temperature Sensor. Appendix F Source Codes Extracts show both drivers constructed for use with Sun SPOT.

ii) Multiple nodes – A group of Sun SPOTs with certain thermal comfort factor sensors attached. A single node is incapable of calculating PMV due to incomplete data / sensor. One node with the highest / strongest power reserve will do the calculation, where all required values come from other nodes using the Spatial Sensing Engine mentioned above. Figure 5-8 shows Sun SPOT connected to various thermal comfort sensors scattered throughout the living space.

Note: In our testing, due to the unavailability of certain sensors (Met, Clo and Vel), their corresponding thermal comfort parameter readings were supplied directly into the Sun SPOT node.

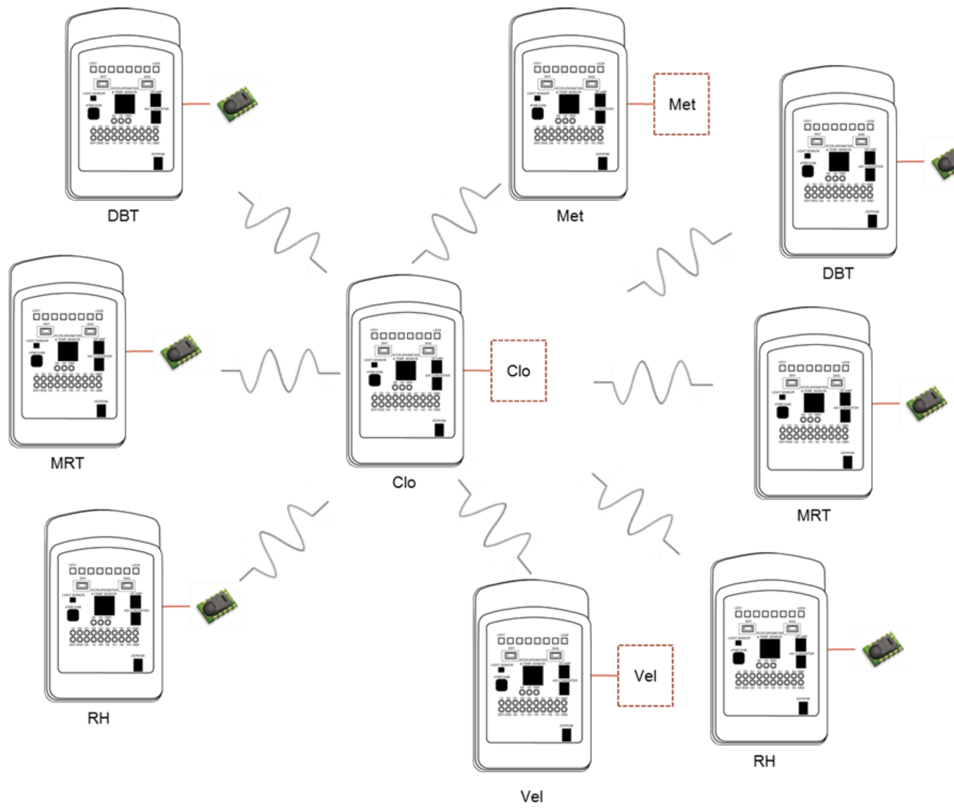


Figure 5-8. PMV Multiple Nodes

The working result for this exercise is discussed further in Chapter 6 as HCAmI Experimental Evaluation.

5.5 HCAmI Fuzzy Engine Implementation

To begin with, a HCAmI Fuzzy Engine was developed, implemented and tested in MATLAB to understand and explore its characteristics. MATLAB provides a Fuzzy toolbox that assists in designing fuzzy engine specifications. Coupled with Simulink, the visual output of the fuzzy engine can be seen and validated.

The system implementation is based on the basic structure of the Mamdani fuzzy rule base engine, with minor modification to suit Sun SPOT's embedded Java J2ME API and libraries. Mamdani's fuzzy inference method was chosen because it is the most commonly used, widely accepted for capturing expert knowledge, and its simple structure for 'min-max' operations.

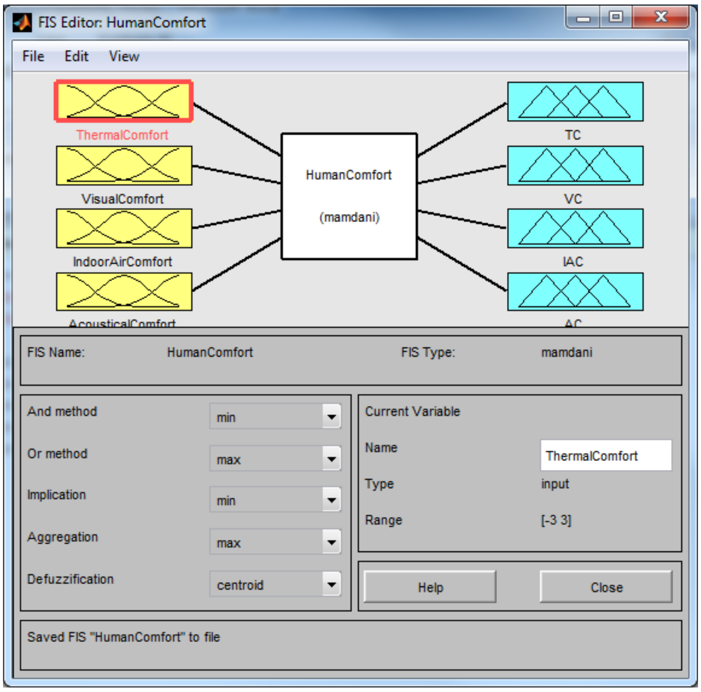


Figure 5-9. HCAmI Fuzzy FIS Editor

Figure Figure 5-9 shows the human comfort fuzzy engine component built using MATLAB Fuzzy Toolbox. Four sets of fuzzy input and output variable were created, one each for TC, VC, IAC and AC.

Each input and output set was filled with the appropriate input and output membership function and parameters as shown in Figure 5-10. The details were discussed in Chapter 4

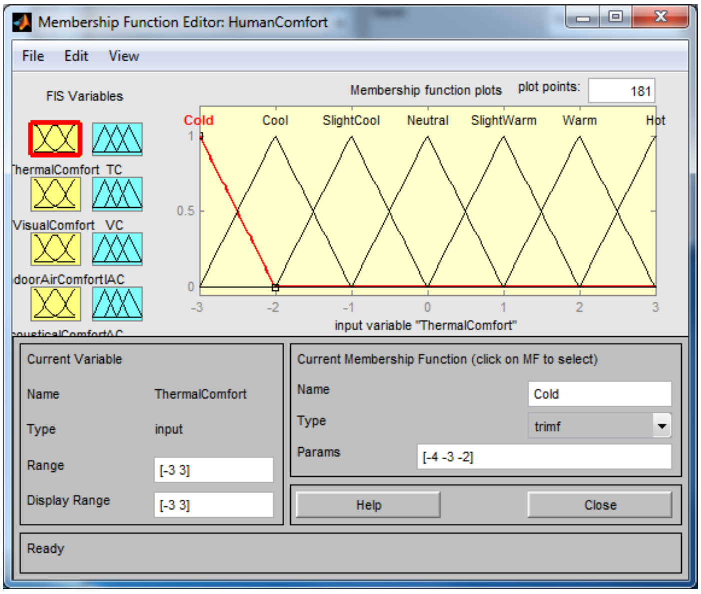


Figure 5-10. HCAmI Fuzzy FIS Membership Function Editor

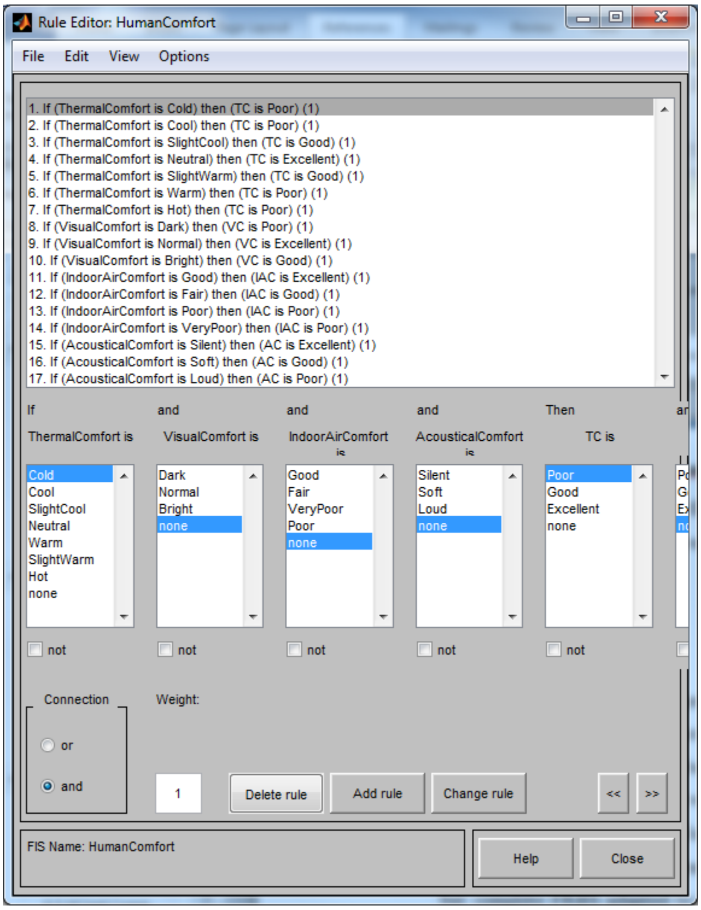


Figure 5-11. HCAmI Fuzzy FIS Rule Editor

Finally, 17 comfort rules were added to the engine, as shown in Figure 5-11. The rules were devised and reviewed in detail in Chapter 4.

The HCAmI Fuzzy Engine then was plugged into Simulink for simulated and real world data testing. Figure 5-12 shows the HCAmI Fuzzy Engine at work in Simulink. Real world human comfort data were fed into it and the output behaviour was observed, as shown in Figure 5-14.

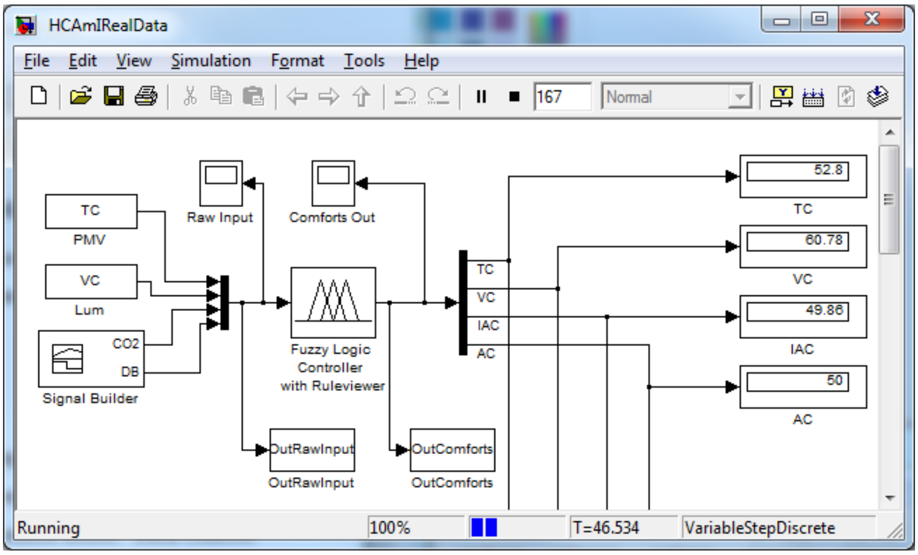


Figure 5-12. HCAmI Fuzzy Engine in Simulink

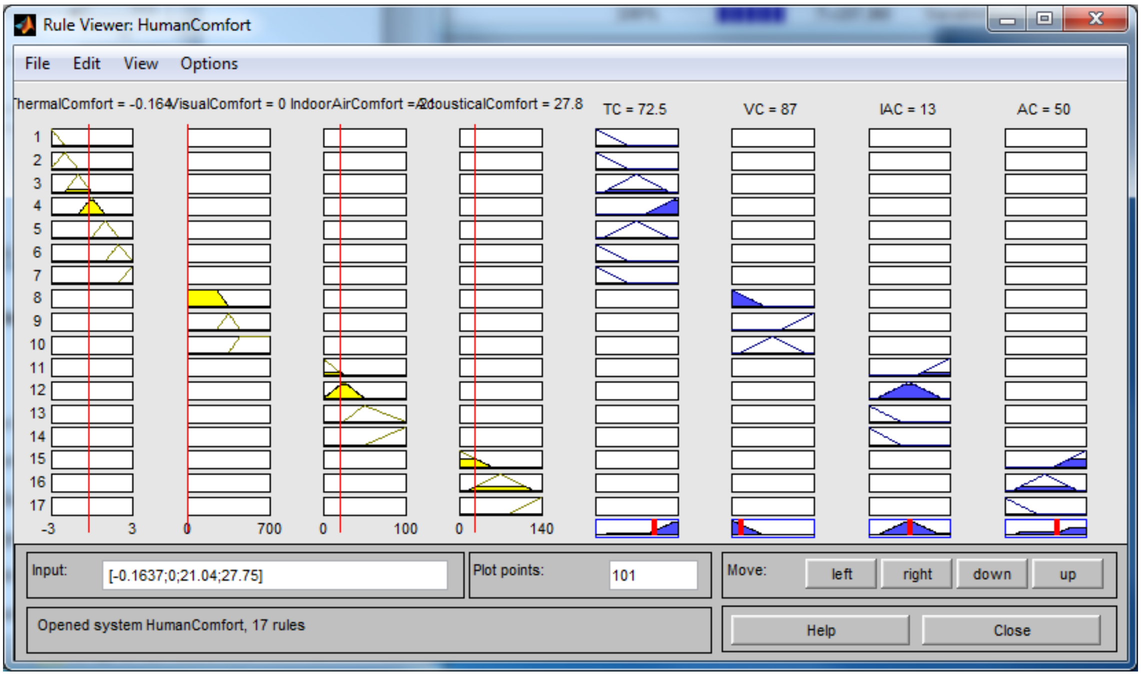


Figure 5-13. HCAmI Fuzzy Engine Rule Viewer

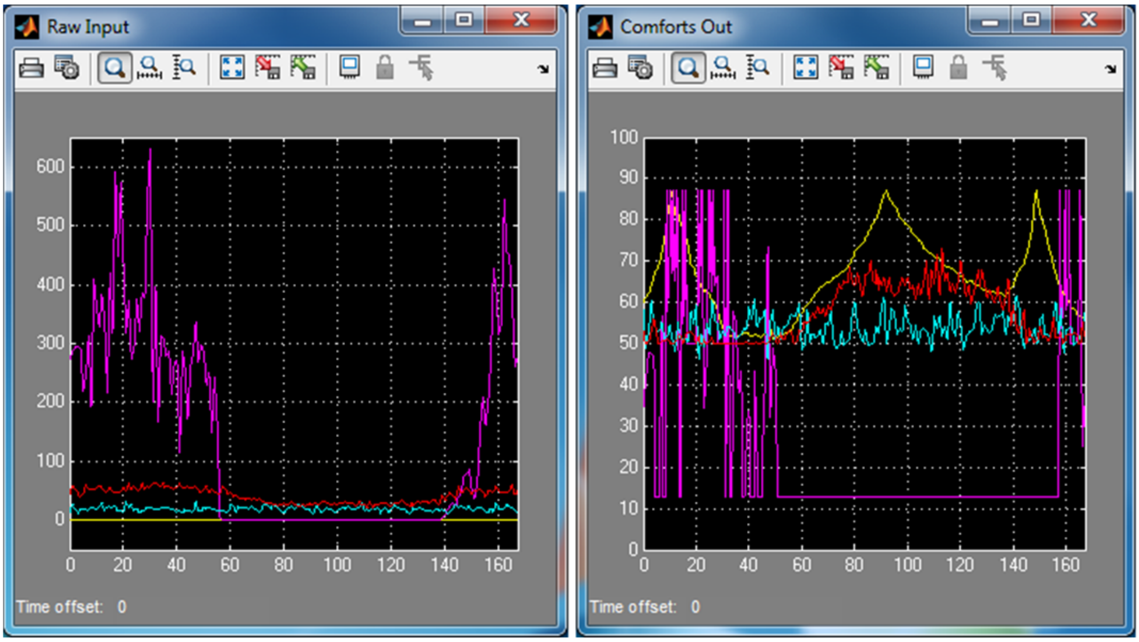


Figure 5-14. HCAmI Fuzzy Engine Input Output

Since the traditional centralised approach of FRBS cannot be implemented within one node only, due to limited node resources such as processing power, memory and battery power, a distributed FRBS adapted to the sensor nodes was developed and implemented. Each comfort node (TC, VC, IAC and AC) was loaded with stripped down FRBS with specific adaptations for the appropriate comfort factor, as discussed in Chapter 4. This was done to reduce the computational burden of each node, where each node executes only the small but complete FRBS adapted to it. Further customisation was done to reduce battery consumption, hence prolonging battery life by putting all nodes in deep sleep. Sensor cloud operation, as shown in Figure 5-19 is discussed further below.

For this work, due to sensor node (Sun SPOT) constraints, a custom fuzzy inference engine was adopted from Sazonov work [114]. A number of modifications were implemented, based on the original code that is compatible with the Sun SPOT J2ME-based API and library. The computational cost and battery consumption were taken into consideration when producing the code. The modifications are:

i) Rewrite the code from Java SE to Sun SPOT J2ME based API and library.

ii) Enhance the fuzzy membership function definition by adding OTA-Conf sub module.

The fuzzy engine allows evaluation of the fuzzy rule sets expressed as text strings and thus simplifies the configuration and setting of the node FRBS via OTA-Conf as shown in Figure 4-19. Initial set-up / configuration (linguistic variable, fuzzy rules and membership function definition) was prepared beforehand by system administrator and pushed into a specific node. The text-based configuration format was transmitted to the node as follows:

```
[Var]
var1 //variable name
mf1 a b c d //membership function for var1
.
.
[Var] //example variable "light"
light
dark 0 0 250 350
normal 250 350 350 450
bright 350 450 700 700
.
.
[Var]
varN
mfN a b c d

[Rules]
if varX is mfX then varY is mfY  //fuzzy rule
if light is dark then vco is poor
if light is normal then vco is excellent
if light is bright then vco is good
```

Complete HCAmI System FRBS text-based configuration as shown in Appendix E HCAmI FRBS Configuration.

The system was designed to separate function (inference engine) and knowledge (linguistic variables, rules), thus allowing the node to execute a fully-fledged FRBS system. A node starts with a ‘‘blank’’ FRBS, and a definition of knowledge components needs to be uploaded. Upon receiving a text-based configuration definition via OTA-Conf, the respective node prepares the necessary knowledge components class as follows:

1. Linguistic variables (`var1`) – represent an input / output variable involved in the fuzzy evaluation performed by the fuzzy engine. The given string argument will be used to address this particular linguistic variable in related rules. For example:

   ```
   LinguisticVariable light = new
   LinguisticVariable("light");
   ```

2. Membership functions (`mf1 a b c d`) – represent the definition of the membership function of the said variable. `mf1` is the name used to address this membership function (`if light is dark` …) and followed by four trapezoid points `a b c d` – left bottom, left top, right top and right bottom, respectively. For example:

   ```
   light.add("dark", 0, 0, 250, 350);
   ```

3. Fuzzy rules `if varX is mfX then varY is mfY` – a text string that represents the fuzzy rule inferred by a node's fuzzy engine. Linguistic variables, membership functions, hedges and fuzzy operations are referred to by respective symbolic names. For example:

   ```
   ar_rules[0] = "if light is dark then vco is poor";
   ```

4. Instantiate fuzzy engine – a fuzzy engine will be instantiated and associated with created linguistic variables and fuzzy rules. The engine is ready to do the work. For example:

   ```
   FuzzyEngine fzyEng = new FuzzyEngine();

   fzyEng.register(light);
   ```

```
fzyEng.register(vco);

fzyEng.register(fuzzyBlockOfRules);
```

The number of fuzzy sets defined in each variable was kept to a minimum but gave completely accurate output, since excessive numbers of fuzzy sets would involve a large number of rules and might increase the inference time, and the additional unnecessary computational burden onto the node.

Overall, FRBSNode Midlet embedded in Sun SPOT comprises 15 classes. All fuzzy operation is controlled by the main class *FRBSNode.class*, *FRBSHelper.class* and the fuzzy engine itself consist of 13 classes coded in their own *fuzzy package*.

Each comfort node fuzzy engine was prepared accordingly, as discussed in Chapter 4,and the operational output is discussed in Chapter 6.

5.6 HCI Engine and Sample Calculation

Based on Equation 4–2, the HCI Algorithm and HCI Flowchart in Appendix D, the HCI Engine was modelled and tested in MATLAB Simulink. To simplify the model development, it was constructed under the assumption that a fixed number of comfort factors were known, namely TC, VC, IAC and AC. The output of the HCAmI Fuzzy Engine as shown in Figure 5-12 was used as the feeder value to the HCI Engine, as shown in Figure 5-15.

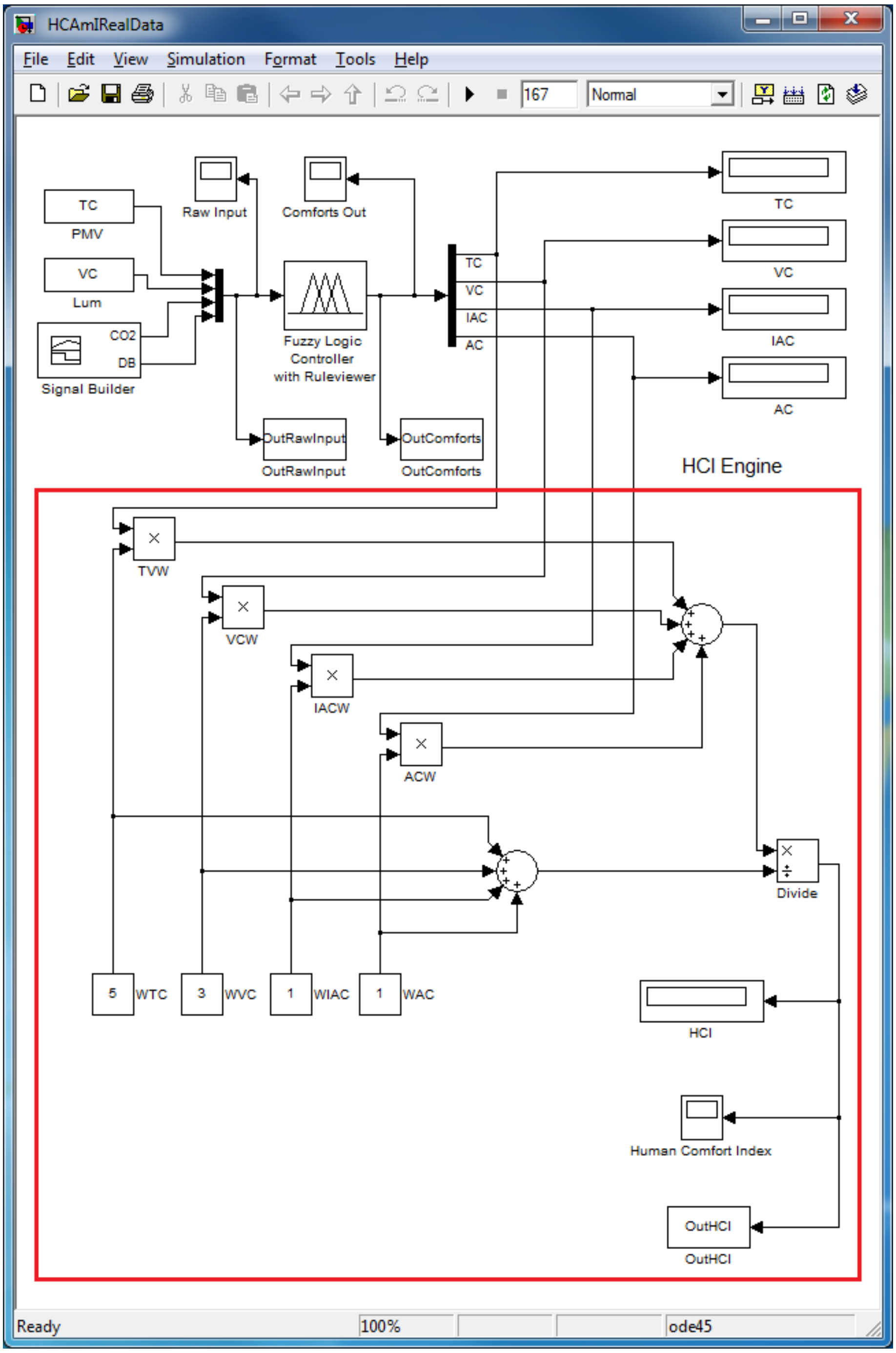


Figure 5-15. HCI Engine in Simulink

In relation to the HCAmI Fuzzy Engine simulation, the simulated value of TC, VC, IAC and AC were fed into corresponding HCI input nodes. Appropriate human comfort factor weights were supplied as well. The weights reflect their influence upon HCI calculations. HCI Engine output was observed and confirmed, as shown in Figure 5-16.

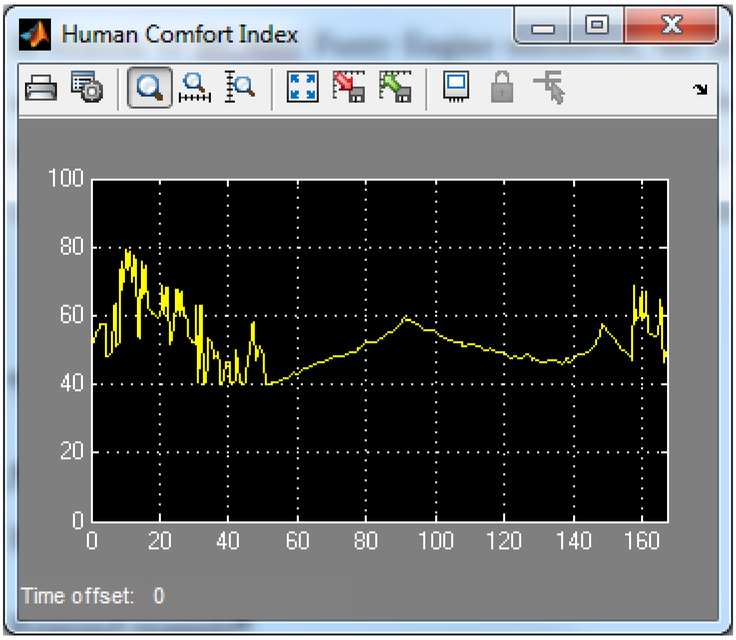


Figure 5-16. HCI Engine Output

Upon HCI Engine modelling validation, the engine was constructed for the Sun SPOT nodes. A generic *HCIEngine.class* was coded based on the Appendix D HCI Flowchart. The class will gather available comfort factors (*Comfort.class*) into an array. Each of these Comfort objects comes with its own comfort weight value. Iteratively, the engine will calculate the HCI based on Equation 4–2. The calculated HCI value is then passed to a higher process that transmits and stores it in the central PC.

5.7 HCAmI Manager

HCManager is a sub component in HCAmI System Manager, as shown in Figure 4-5. It serves as central controller of the HCAmI system. HCManager was programmed as a simple Java application running on the HCAmI host PC. It serves as master controller to time synchronise all node operation cycles as shown in Figure 5-19. Every ten minutes, the manager wakes up and starts listening to incoming datagrams from all Sun SPOT nodes via the attached Sun SPOT basestation. The ten minutes chosen is a typical / arbitrary value that suits Sun SPOT platform for proving the functionality of the system. Ideally, the sampling of the system should comply with the human interaction process that could be in a matter of seconds. The sampling rate also relates to the measured environment rate of change. Different living space warrants different sampling rate.

Upon receiving the communication datagram, it will spawn a process thread that will process each datagram independently, as shown in the code snapshot below.

```
// Main data collection loop
while (true) {
  try {
    // Read datagram received over the radio
    rCon.receive(dg);
    switch (dg.readByte()) {
    case THERMAL_COMFORT :
      processThermalComfort(); break;
    case VISUAL_COMFORT :
      processVisualComfort(); break;
    case INDOOR_AIR_COMFORT :
      processIndoorAirComfort(); break;
    case ACOUSTICAL_COMFORT :
      processAcousticalComfort(); break;
    case SUN_SPOT_STATUS :
      prosesSunSPOTStatus(); break;
    }
} catch (Exception e) {
    System.out.println("Error: " + e.getMessage());
    e.printStackTrace();
  }
}
```

In each spawn process (Java multithread), the datagram will be disseminated into its own entity object. The data dissemination is done according to the HCAmI Communication Protocol Format (Figure 4-21). This entity data is pushed / stored into permanent storage for record keeping and further offline analysis. Code snapshot below shows an initial process example for processing a Sun SPOT status datagram.

```
private void prosesSunSPOTStatus() throws IOException {
  String addr = dg.getAddress();// read sender's Id
```

```
long time = dg.readLong();     // read time of the reading
int battLevel = dg.readInt(); // read battery level value
…
```

This datagram carries Sun SPOT's unique MAC address, a time stamp for the data and the node's battery level.

5.8 HCAmI Communications

HCAmI communications are handled by a specific custom made Sun SPOT communication application library module. The details of this application layer protocol were discussed in Chapter 4. It serves as the message transport mechanism in delivering data and information throughout HCAmI cloud nodes.

A simplistic broadcast communication approach was used for the test bed. This is partly to simplify the development undertaking on the assumption all message node hopping (multi-hop communication) will be handled by the built in Sun SPOT multi-hop lower level communication library, as mentioned in Chapter 3.

The HCAmI communication library was composed of two main classes: *HCAmIComm.class* and *PacketType.class*.

HCAmIComm.class serves as entry point for message sending and receiving activity. It provides a two process thread – receiver thread and sender thread. The receiver thread will listen for any incoming broadcast and if there is an incoming message, it will process it accordingly based on message type. The receiver thread code extract is shown below.

```
try {
  System.out.println("Reset receiver datagram");
  dg.reset();
  System.out.println("Waiting receiver datagram until timeout");
  dgConnection.receive(dg);

  switch (dg.readByte()) {
    case THERMAL_COMFORT :
      processThermalComfort(dg.getAddress());
      break;
```

```
    case VISUAL_COMFORT :
      processVisualComfort(dg.getAddress());
      break;
    case INDOOR_AIR_COMFORT :
      processIndoorAirComfort(dg.getAddress());
      break;
    case ACOUSTICAL_COMFORT :
      processAcousticalComfort(dg.getAddress());
      break;
    case SUN_SPOT_STATUS :
      prosesSunSPOTStatus(dg.getAddress());
      break;
  }
} catch (Exception e) {
  System.out.println("*** Opps! Error: " + e.getMessage());
  e.printStackTrace();
}
```

On the other hand, the sender thread will wait for node data transmission. Upon receiving all the data, it will compose the message and pass the message to the lower level Sun SPOT communication library for delivery. The sender code extract for thermal comfort is shown below.

```
// Sending job
// Create a RadioConnection
RadiogramConnection dgConnection = null;
Datagram dg = null;

//wait for random second before transmit to minimise broadcast
collision
waiting(new Random().nextInt(1000));

try {
  // Flash blue LED to indicate a sending event
  leds[0].setRGB(0, 0, 255);

  // The Connection is a broadcast so we specify it in the creation
string
  dgConnection = (RadiogramConnection)
```

```
        Connector.open("radiogram://broadcast:" + BROADCAST_PORT);
  // Then, we ask for a datagram with the maximum size allowed
  dg = dgConnection.newDatagram(dgConnection.getMaximumLength());

  // Package the time and sensors reading into a radio datagram and
send it.
  // Package as Thermal Comfort data
  dg.reset();
  dg.writeByte(THERMAL_COMFORT);
  dg.writeLong(now);      // timestamp
  dg.writeDouble(ta);     // Ta
  dg.writeDouble(mrt);    // MRT
  dg.writeDouble(vel);    // Vel
  dg.writeDouble(rh);     // RH
  dg.writeDouble(clo);    // Clo
  dg.writeDouble(met);    // Met
  dg.writeDouble(pmvVal); // PMV
  dg.writeDouble(TCVal); // TC Value
  dgConnection.send(dg);  // send the data

  System.out.println("Broadcast is going through");
} catch (Exception ex) {
  System.out.println("*** Opps! Error: " + ex.getMessage());
   ex.printStackTrace();
}
```

PacketType.class is a second class HCAmI communication library / package that serves as communication library interface. It contains several HCAmI communication protocol constants, as shown in code snapshot below.

```
public interface PacketTypes {
  /** Port to use to locate the host application. */
  public static final String BROADCAST_PORT = "67";
  /** Port to use for sending commands and replies between the SPOT
and the host application. */
  public static final String CONNECTED_PORT = "43";

  /** Thermal Comfort Data group */
  public static final byte THERMAL_COMFORT            = 11;
  /** Visual Comfort Data group */
```

```
  public static final byte VISUAL_COMFORT           = 12;
  /** Indoor Air Quality Comfort Data group */
  public static final byte INDOOR_AIR_COMFORT       = 13;
  /** Acoustical Comfort Data group */
  public static final byte ACOUSTICAL_COMFORT       = 14;
  /** Spatial Comfort Data group */
  public static final byte SPATIAL_COMFORT          = 15;

  /** Free Range Sun SPOT status */
  public static final byte SUN_SPOT_STATUS          = 40;
}
```

The constants declared are used in differentiating message type being received / sent by a node.

In HCAmI system implementation, an HCAmI Communication component serves as the application-level communication function and protocol for the HCAmI System. The HCAmI Communication component is responsible for sending and receiving data within the nodes. In the Data Sharing stage (Figure 5-19), a window of five seconds was given for communication to happen. Two simultaneous threads were invoked: `startSenderThread()` and `startReceiverThread(LISTEN_DURATION)`.

On `startSenderThread()`, within each comfort node, upon acquiring all sensor data, an appropriate datagram is composed. The thread waits for a random duration of between 0 to 1 second. Random wait time is introduced to minimise datagram collision due to broadcast based communication used by the HCAmI System. Finally, the datagram is transmitted.

On the other hand, `startReceiverThread(LISTEN_DURATION)` was executed on both comfort nodes and basestation. The receiver thread opens a receiver process routine and waits for an incoming datagram. Upon receiving a datagram, it will decompose the datagram, and based on the datagram type, it will pass it to a specific processing thread. The thread will keep on listening for up to five seconds (`LISTEN_DURATION`). On comfort nodes, the incoming datagram will be processed if the datagram only belongs to its own comfort factor, which is thermal comfort nodes will only process incoming thermal comfort datagrams and discard all other non-related datagrams. On the HCAmI basestation, it will process all incoming datagram.

For each operation cycle, eight datagrams were broadcast. Four comfort datagrams from TC, VC, IAC and AC node and four Sun SPOT status datagrams from each node were broadcast within the testbed.

Over the course of the test run (20/04/2010 to 26/08/2010), it can be observed that approximately 18,000 datagrams were transmitted, with fewer than 100 datagrams lost. The loss was mainly due to hardware glitches, rather than transmission collision (the broadcast nature of HCAmI communication). The datagram size distribution varies based on datagram type, that is 77 bytes for a thermal comfort datagram, 25 bytes for a visual comfort datagram, 29 bytes for an indoor air comfort datagram, 25 bytes for an acoustical comfort datagram and 17 bytes for a Sun SPOT status datagram.

Detailed evaluation of a high-level application protocol such as an HCAmI protocol turns out to be surprisingly difficult due to a variety of external non-protocol related factors involved. The most obvious one is the underlying network and network stack performance employed by the Sun SPOT WSN platform. This platform processor and communication link and their network stack dominate the overall communication performance. Consequently, proper protocol measurement such as throughput cannot be effectively measured.

5.9 HCAmI Setup and Operation

In order to evaluate the HCAmI design and implementation, a test bed was developed. It was HCAmI in its simplest form. The hardware setup for our study includes a couple of free range Sun SPOT connected to sensor(s) and a PC with an attached basestation. A simple database installed on the PC that acts as the data sink / repository. Figure 5-17 shows the HCAmI test bed setup.

Each sensor node (free range Sun SPOT) was loaded with generic / bare FRBSNode Midlet. The FRBSNode Midlet fuzzy definition will be set up over the air based on its comfort factor responsibility. Each node is connected to its appropriate sensor(s).

On the Sun SPOT basestation, HCI Midlet was installed. The midlet will collect all available comfort factor values from sensing nodes and calculate HCI value. Then it will store the data / information on a simple flat file database.

For this exercise, two physical Sun SPOTs were used and two virtual Sun SPOTs that ran on Sun SPOT Emulator in Solarium (an application and emulator for discovering and managing Sun SPOT), a basestation and a personal computer as shown in Figure 5-18.

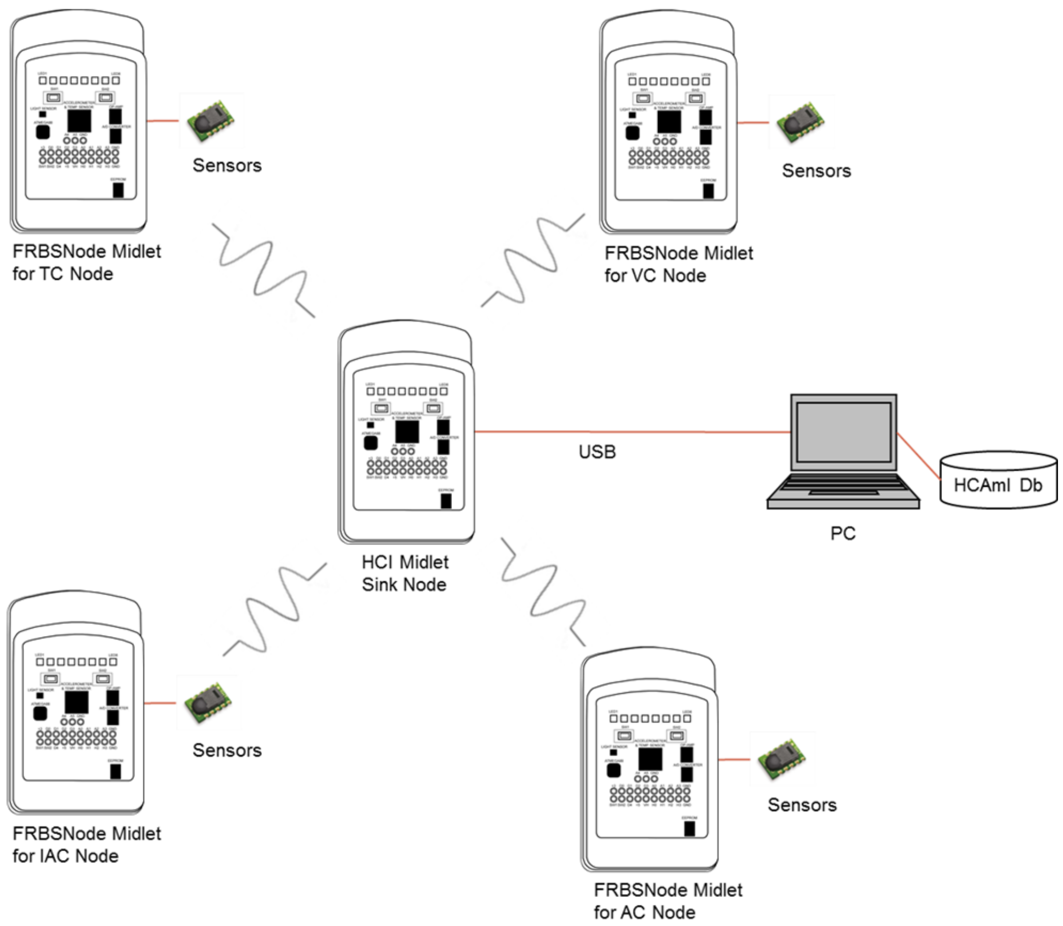


Figure 5-17. HCAmI Setup

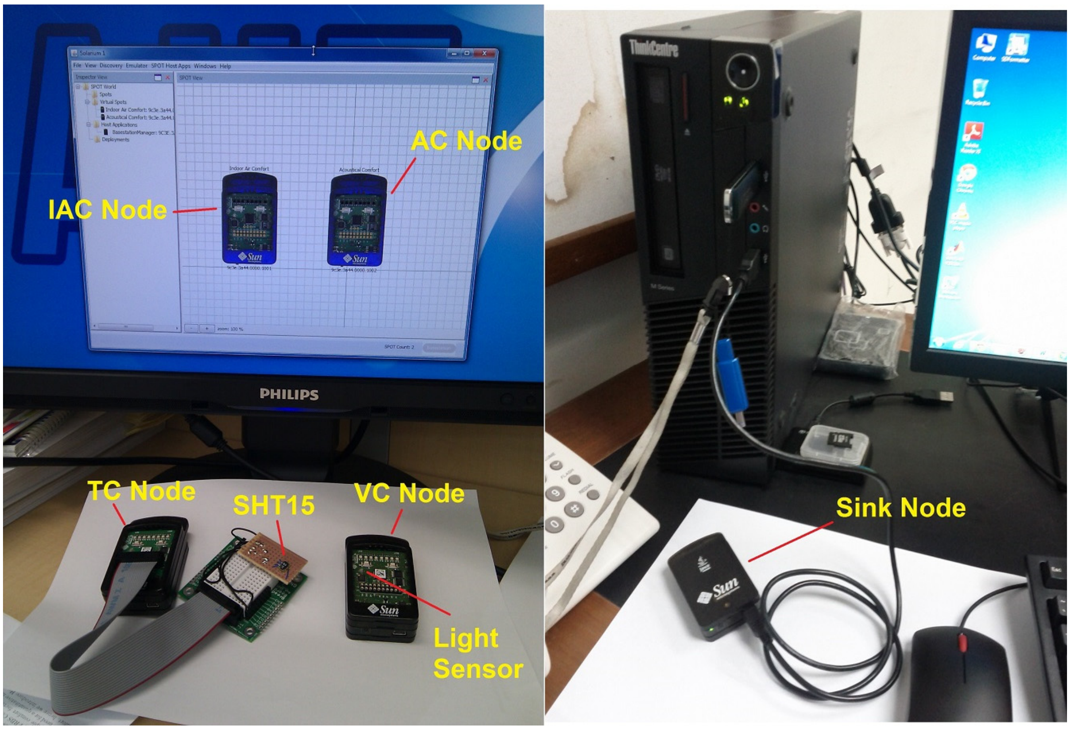


Figure 5-18. HCAmI Hardwares

One physical Sun SPOT with a SHT15 temperature and humidity sensor from Sensirion AG manages thermal comfort activity, and one physical Sun SPOT manages visual comfort activity via its built-in light sensor. Physically, a Sun SPOT free range node for TC was placed 1.2m above the floor and a Sun SPOT free range node for VC was placed at the table facing upward.

Two virtual Sun SPOTs managed indoor air comfort activity and acoustical comfort activity that was fed with a generated value of CO_2 and captured sound level data (using a UNI-T UT351 Sound Level Meter), respectively.

Each of these nodes acquired environment data from its sensors, computed its respective comfort factor and transmitted the result to the sink node which calculated the HCI value. The sink node was also responsible for passing all data and information received from free range Sun SPOTs to the attached PC for storage.

In our integrated all nodes test, the HCAmI sensor cloud was run in an endless looping mode. It woke up every ten minutes (time sync) for:

i) Sensing the environment. Each node sensed the appropriate environment based on the attached sensor(s). It read the sensor's raw reading and converted it into the appropriate value.

ii) Data sharing. Each node then prepared any acquired data and broadcast it throughout the cloud. On receiving ends, any nodes that needed the data captured and used it.

iii) Calculating individual comfort value. With an embedded FRBS engine, the node calculated its appropriate comfort value and sent it to the sink node.

iv) Calculating HCI. The sink node received all individual comfort values then finalised the process by calculating the HCI.

v) After a minute of active mode, all nodes went into deep sleep mode. The process repeated.

Figure 5-19 shows the HCAmI sensor cloud activity process cycle.

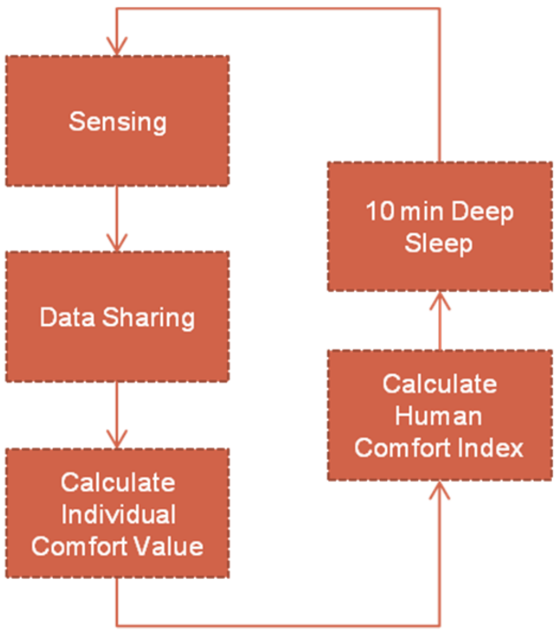


Figure 5-19. HCAmI Sensor Cloud Activity Cycle

5.10 Chapter Summary

This chapter detailed HCAmI implementation. HCAmI modelling, simulation and implementation and their technicalities were discussed. Each component of HCAmI was modelled as an abstraction, resulting in formal specification with assumptions and constraints. The model was refined repeatedly until it was deemed to be sound.

Each HCAmI component implementation was discussed and explained. The initialisation and detail implementation of each component was presented. The component was validated using MATLAB Simulink where applicable.

Lastly, HCAmI setup and operation was highlighted. The setup for a case study / test bed on a complete HCAmI system was described. The case study / test bed encompass two physical Sun SPOT linked to live sensors that measures temperature, humidity and light level. In addition, two virtual Sun SPOTs were fed with generated CO_2 values and captured sound level data were used to estimate indoor air comfort and acoustical comfort respectively.

Chapter 6 HCAmI Experimental Evaluation and Application

6.1 Introduction

This chapter presents a case study of HCAmI experimental evaluation and application as described in previous chapters. Full scale HCAmI implementation was carried out to demonstrate the effectiveness of the system architecture and framework. The various components of HCAmI were theoretically described and analysed in Chapter 4 and Chapter 5.

Most environmental comfort systems only deal with specific comfort factors individually, which makes it difficult to compare results across studies. This difficulty is exacerbated by the fact that different systems use varying levels of evaluation, criteria and data sets. In most cases, the size and attributes of data used for testing also vary; hence we cannot reliably compare the results. Due to this, the HCAmI system was subjected to self-evaluation where each component was evaluated and tested comparatively against similar implementation in suitable simulation tools such as MATLAB, Simulink and specific java test applications.

Subsequent sections describe the results of individual comfort factors and the human comfort of said living space. The detailed descriptions of these components are individually expressed in terms of real world operations and simulation modelling to characterise and analyse their various features. Real world operation characteristics are used to explain overall system design and performance. The results of the overall system performance and integration activities can be used as guidelines for full scale WSN system design and implementation.

6.2 Case Study

For this case study, we ran an HCAmI system in a full environment sensing scenario using simulated raw (for indoor air comfort) data and real data derived from thermal comfort, visual comfort and acoustical comfort real-time data collected from 20/04/2010 to 26/08/2010 (four months of data) in the SeNSe Laboratory, School of Engineering, AUT University. This analysis used a short snapshot of the collected data, from 08:00am 25/08/2010 to 11:40am 26/08/2010.

At that time, it was a typically gloomy late winter's day in Auckland, with indoor air temperature varying from 21°C to 27°C and indoor relative humidity varying from 37.2% to 46.7%. The laboratory was fitted with fixed non-adjustable hot water radiators as the heating elements, manual artificial lighting and had no active or passive ventilation system installed. The people in the laboratory wore shirts with light sweaters and normal trousers (clothing value Clo = 1) and mainly sat in front of PCs (metabolic rate value Met = 1). The Clo and Met value represents a typical value at that moment. The value can be changed accordingly.

The sound level fluctuates between 20dB to 70dB, due to the location of the laboratory – on level 3 of the WD building at 19 St. Paul Street, Auckland, which contributed significantly to the noise level.

Parameters	Datasets Range / Value
Dry Bulb Temperature (DBT)	21°C to 27°C
Mean Radiant Temperature (MRT)	21°C to 27°C
Relative Humidity (RH)	37.2% to 46.7%
Air movement (Vel)	0.1 m/s
Metabolic Rate (Met)	1 Met
Clothing Level (Clo)	1 Clo
Lighting Level	0 Lux to 325 Lux / 0 to 650 ADC
CO_2 Level	10 point to 30 point
Sound Level (dB)	20 dB to 70 dB

Table 6-1. HCAmI Case Study Datasets Range and Values

Table 6-1 summarises HCAmI System case study dataset ranges and values.

6.2.1 Thermal Comfort Subsystem

The thermal comfort subsystem consisted of two main engines: a *PMV Engine* and a *TC FRBS Engine*. The PMV Engine was responsible for sensing the environment and calculating PMV value. The calculated PMV value was forwarded to the TC FRBS Engine where TC was determined as discussed in Chapter 4 and Chapter 5. It is the most complicated comfort sub system available. It was presumed that if the node could manage the most complicated tasks, it would be more than sufficient to handle other less complicated tasks.

PMV Engine

In this case study, a single node was used to compute PMV value. The PMV Engine represents the most complex pre-processing engine that a node needs before passing the value into a FRBS Engine to determine comfort (TC, VC, IAC and AC). It is considered a benchmark of node computational capabilities. In this case, the calculated PMV value was then supplied to the TC FRBS Engine for further processing to determine the TC.

A Sun SPOT sensor node was used to measure the SeNSe lab's air temperature, calculate its PMV and transmit the recorded data to the Sun SPOT basestation for storage. Real time measurements were taken at ten minute interval started from 7th July, 2009 at 6:39pm until 9th July, 2009 at 4:49pm. In this exercise, it is assumed that [93]:

i) Mean Radiant Temperature (MRT) is equivalent to Dry Bulb Temperature (DBT) measured using Sensirion SHT15 or Precon HS-2000V sensor.
ii) Air Movement (Vel) = 0.1
iii) Relative Humidity (RH) = 60%
iv) Metabolic Rate (Met) = 1
v) Clothing Level (Clo) = 1

Figure 6-1 and Figure 6-2 shows Sun SPOT's DBT and calculated PMV values respectively.

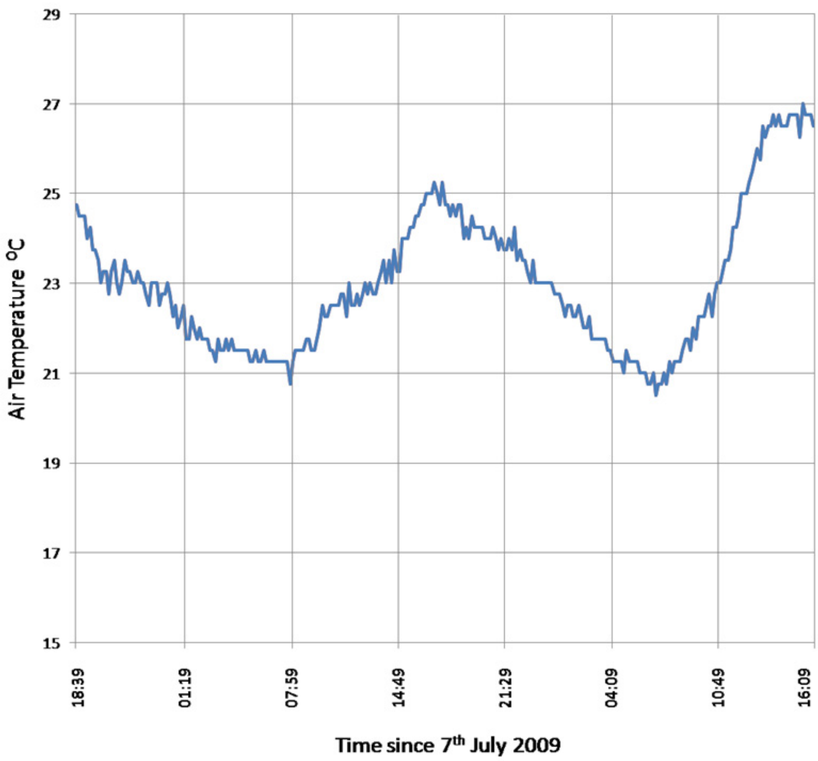


Figure 6-1. Air Temperature in SeNSe Lab

Figure 6-1 [93] shows air temperature changes throughout the day. Air temperature fluctuation can be seen here due to central heating operationing in the lab. During the day time, air temperature climbs to 27°C. During the night time, the central heating system was switched off, hence the drop in air temperature. The air temperature did not drop very much even though it was mid-winter in Auckland, partly due to the SeNSe lab building design.

It can be seen here that the temperature rises and falls steadily due to the lack of a smart control mechanism with the SeNSe laboratory heating radiator. It was controlled by a simple timer which will turn on in the morning, and off in the evening.

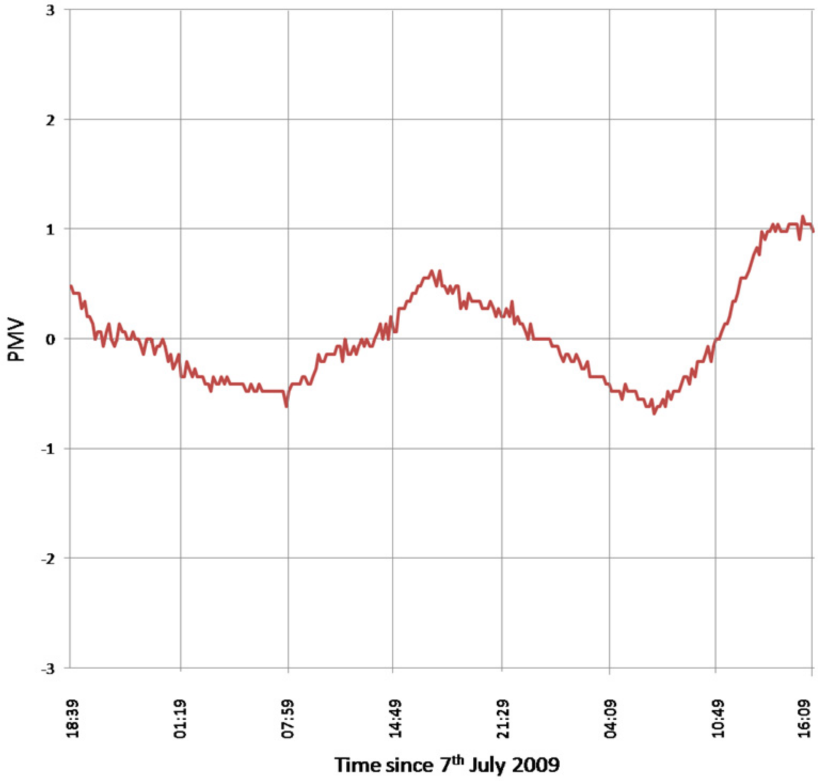


Figure 6-2. PMV in SeNSe Lab

Figure 6-2 [93] shows calculated PMV values that range from -0.61 (slight cool) to 1.11 (slight warm). From these observations, it can be presumed that the thermal comfort in the laboratory is somewhat well-off, since the PVM value never goes beyond -1 and 1.2. But, at certain periods, it can be fairly high, with PMV shooting up beyond 1. This is mainly due to a fixed control mechanism in the SeNSe laboratory radiator. Due to this, it can be observed that laboratory's occupants in the vicinity of the radiator resort to opening the window slightly to bring in some cool winter air from outside to reduce the heat they feel.

The sensing engine and PMV engine (*PMVThermalComfort Midlet*) embedded in Sun SPOT occupies only 4677 bytes of the 4Mb Sun SPOT flash memory. During the test run, it was observed that there are 312,208 bytes of free memory out of the 459,264 bytes available on Sun SPOT RAM. Available free RAM also varies a little due to the internal Java Virtual Machine's (JVM) garbage collection process. On average, Sun SPOT spends about 200 milliseconds (ms) in active mode for sensing, calculating PMV value, composing a datagram and transmitting the datagram to Sun SPOT basestation. Out of a nearly 46 hour (approximately 165,600,000ms) test run, Sun SPOT only spent about 55,400ms (less than a minute) in active mode, whereas the rest of the time it was in deep sleep mode. The Sun SPOT node only spent about 0.033% in active mode for a ten minute interval sampling rate.

In respect to this exercise, it can be observed that Sun SPOT WSN is computationally capable of acquiring raw measurements and at the same time calculating PMV value. Although PMV calculation is computationally complex, the Sun SPOT ARM based processor is capable doing the calculation.

Even though the sensor node's power consumption is not in the scope of this study, it can be observed that Sun SPOT battery life is not as good as other simple WSN platforms. After running it for more than 46 hours to sense, calculate and transmit data for every ten minutes, the battery level dropped from 100% to 85%. That is a huge drop in battery level over a short time span. Most probably, this is due to unnecessary power consumption from onboard apparatuses such as the accelerometer, light sensor, LEDs, general purpose I/O pins and switches. By default, these apparatuses have been turned on and available to any application running on Sun SPOT.

In conclusion, the PMV Engine embedded into Sun SPOT is a piece of computationally complex code that needs a strong processor (32bits) and platform (plenty of RAM and processing stack) to execute it. Therefore, it can be deduced that any platform that can carry out the task should be able to handle the whole HCAmI System without any significant difficulty.

TC FRBS Engine

The TC FRBS Engine node was implemented with seven IF-THEN fuzzy rules, six environmental sensor inputs that produce one input fuzzy variable (PMV) with seven membership functions, and one output fuzzy variable with three membership functions. On average, the TC node spends about 250ms in active mode for sensing environment, calculating PMV value, executing a TC fuzzy task, composing a datagram and transmitting the datagram to the sink node. The TC FRBS Engine itself only needs less than 50ms to execute FRBSNode Midlet. Meanwhile, FRBSNode Midlet for VC, IAC and AC shows slightly faster average execution time (less than 220ms). This is mainly due to the simpler tasks performed.

The output of the operation is presented and compared against simulation done with MATLAB and Simulink for validation.

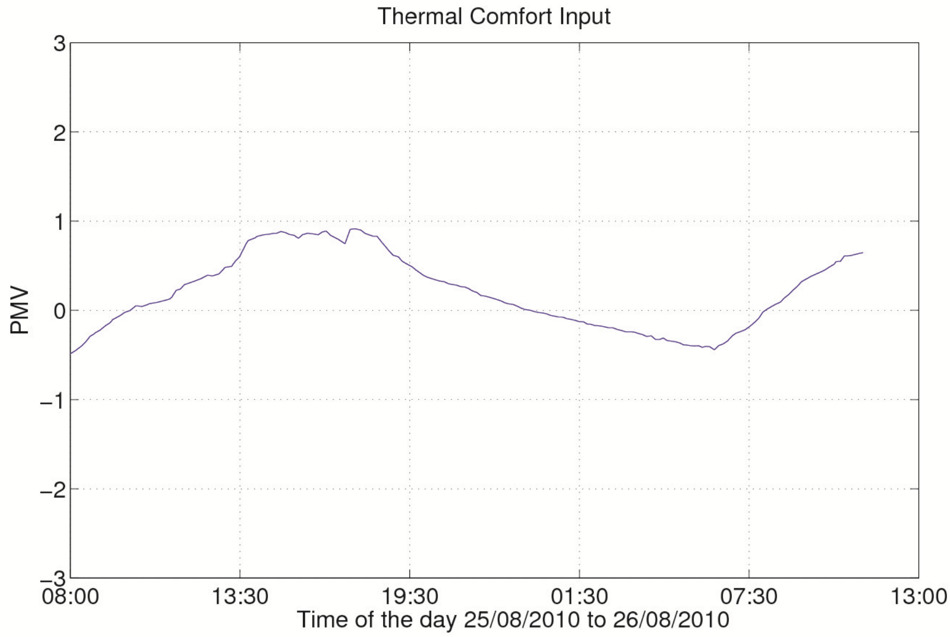


Figure 6-3. Thermal Comfort FRBS Input

Figure 6-3 [89, 95] shows the PVM value varying between -1 (Slight Cool) to 1 (Slight Warm). The PMV value peaked at around 2pm to 4pm, giving the thermal sensation of being slightly warm to the occupant. It then, dipped during the night because the laboratory heater was switched off, and climbed again after 6am because the heater was switched on again. The rise and fall of PMV value were mainly due to DBT and MRT changes that directly linked to the unregulated heating system used inside the SeNSe lab.

Figure 6-4 [89, 95] shows that thermal comfort of the SeNSe laboratory fluctuates between 50 (good) and 87 (excellent) throughout the sensed duration. The SeNSe laboratory can be considered thermally comfortable to live in.

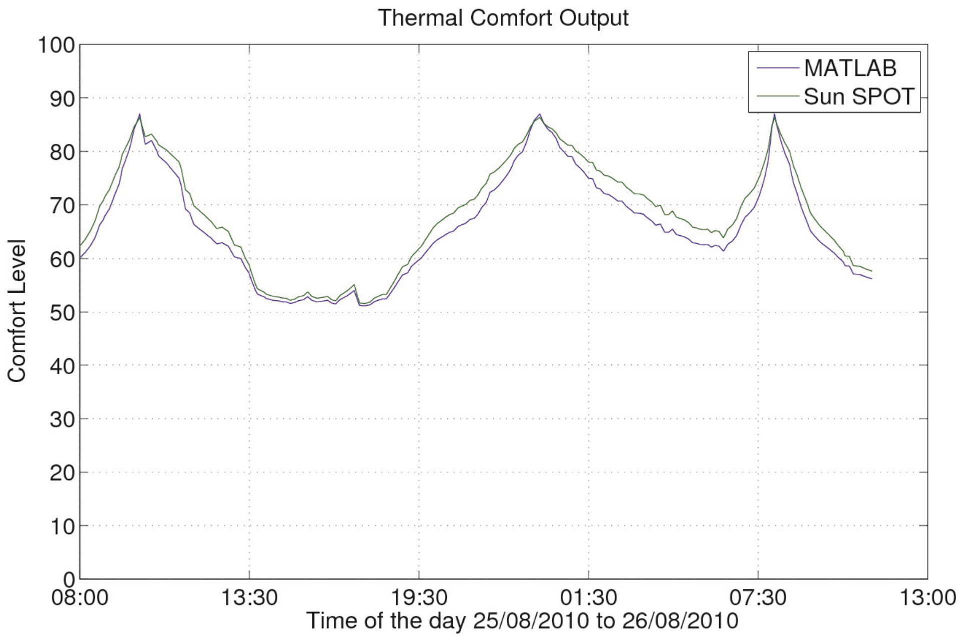


Figure 6-4. Thermal Comfort FRBS Output

In order to evaluate the accuracy and correctness of the TC FRBS Engine implementation of Sun SPOT, similar architecture was developed using MATLAB and Simulink, and it was subjected to identical raw data. Figure 6-4 also shows the Thermal Comfort FRBS calculated output performed with MATLAB and Simulink. Calculated values by Sun SPOT vs. MATLAB show a small discrepancy. They vary from 0% to 3.6%, a small percentage value that is not perceivable by human. The variance in calculations is mainly due to the internal Sun SPOT J2ME Math library, Vector library implementation and rounding mechanisms are handled by the built in libraries. These libraries are used extensively in building the Sun SPOT FRBS Engine (*FRBSNode Midlet*). Other contributing factor is a different way of building the Mamdani fuzzy inference engine itself. Generally, the output produced by Sun SPOT and MATLAB shows a similar pattern, therefore confirming the correctness of implementation.

6.2.2 Visual Comfort Subsystem

The visual comfort subsystem consists of a *VC FRBS Engine* and a light sensor driver. A VC FRBS Node was implemented with an appropriate fuzzy rules description, as discussed in Chapter 4. A Light sensor driver is a small piece of Sun SPOT code, as shown in Appendix F Source Codes Extracts that reads light levels from a Sun SPOT on board light sensor. The value returned in the Sun SPOT ADC Light Level unit where 1 Lux ≈ 2 x ADC. The acquired value was then forwarded to the VC FRBS Engine.

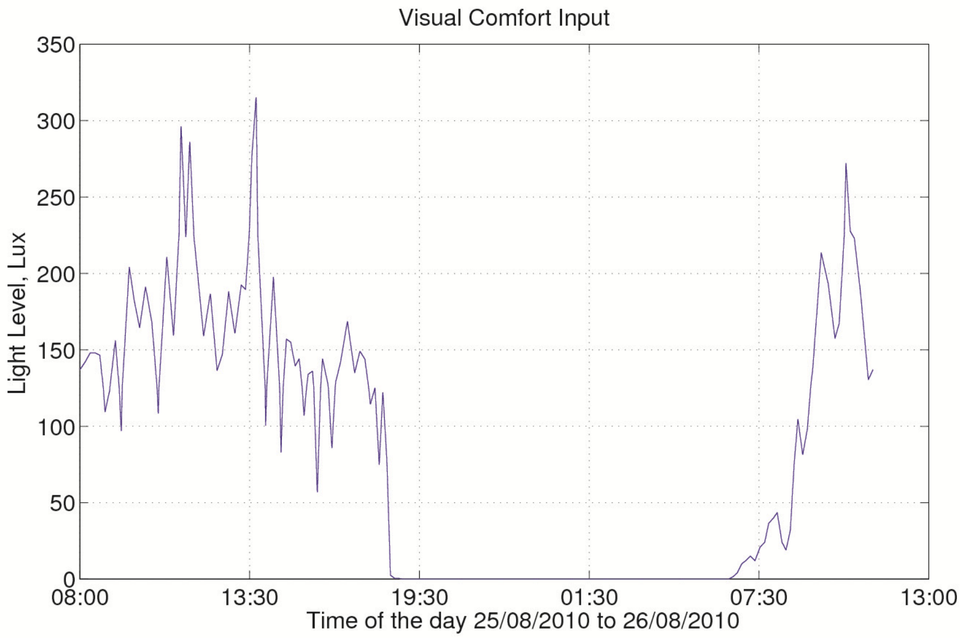


Figure 6-5. Visual Comfort FRBS Input

Figure 6-5 [89, 95] illustrates the light level in the laboratory. It fluctuates from 0 Lux to 325 Lux, representing typically gloomy late winter sunlight. During the daytime, it hovers between 100 Lux and 200 Lux with occasional spiking due to sun rays striking the laboratory. The laboratory was completely dark from 7 pm to 7 am the next day. This was due to artificial light being switch off. Similarly, at approximately 8 am (next day), the light value spiked due to switching on the artificial lighting in the laboratory.

The output of the VC FRBS Engine was plotted in Figure 6-6 [89, 95]. It can be observed that visual comfort fluctuated wildly from poor (Comfort Level = 13) to excellent (Comfort Level = 87) throughout the day, due to lack of natural light coming into the laboratory, sun rays striking momentarily and inadequate artificial light in that particular sensed spot.

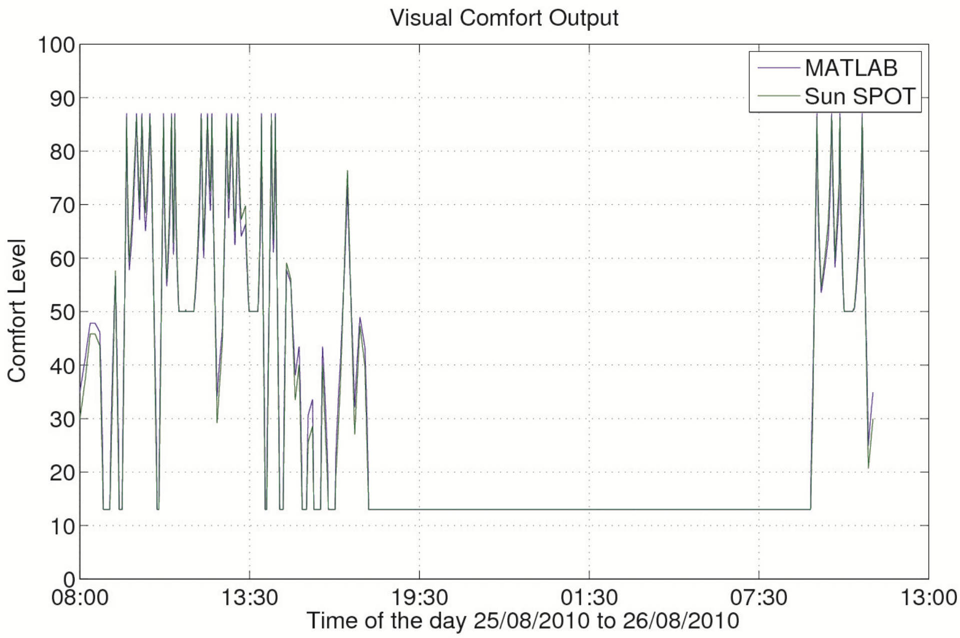


Figure 6-6. Visual Comfort FRBS Output

The output of the Sun SPOT VC FRBS Engine was compared with MATLAB and Simulink, as shown Figure 6-6. From the plot, it can be seen that it behaved similarly to the VC FRBS Engine with 0% to 3.6% difference. The output pattern is similar, confirming ability of Sun SPOT to implement the visual comfort algorithm written in MATLAB.

6.2.3 Indoor Air Comfort Subsystem

The indoor air comfort subsystem was comprised of the *IAC FRBS Engine* and a simulated CO_2 reading sub routine. The subsystem was implemented as a virtual Sun SPOT due to the inability of Sun SPOT WSN to connect with a CO_2 sensor. Lack of a consistently high power source was the main reason that it was not possible to connect a CO_2 sensor to a Sun SPOT node.

The IAC FRBS Node was implemented with its own fuzzy sets, as explained in Chapter 4. A series of simulated CO_2 readings was fed into the IAC FRBG Engine and indoor air comfort was determined.

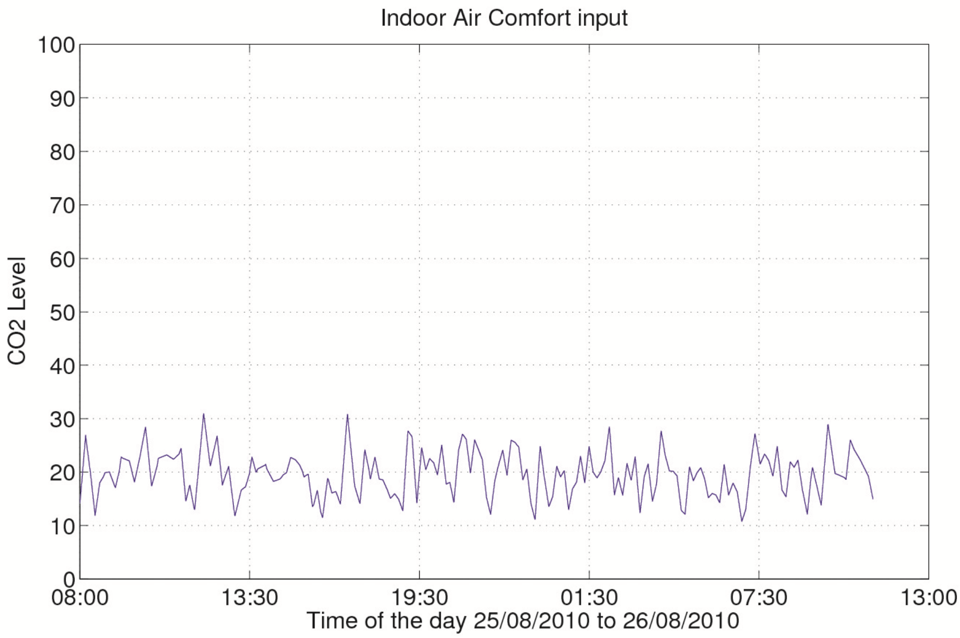


Figure 6-7. Indoor Air Comfort FRBS Input

Figure 6-7 [89, 95] shows simulated indoor air comfort where CO_2 levels remained constant throughout the day.

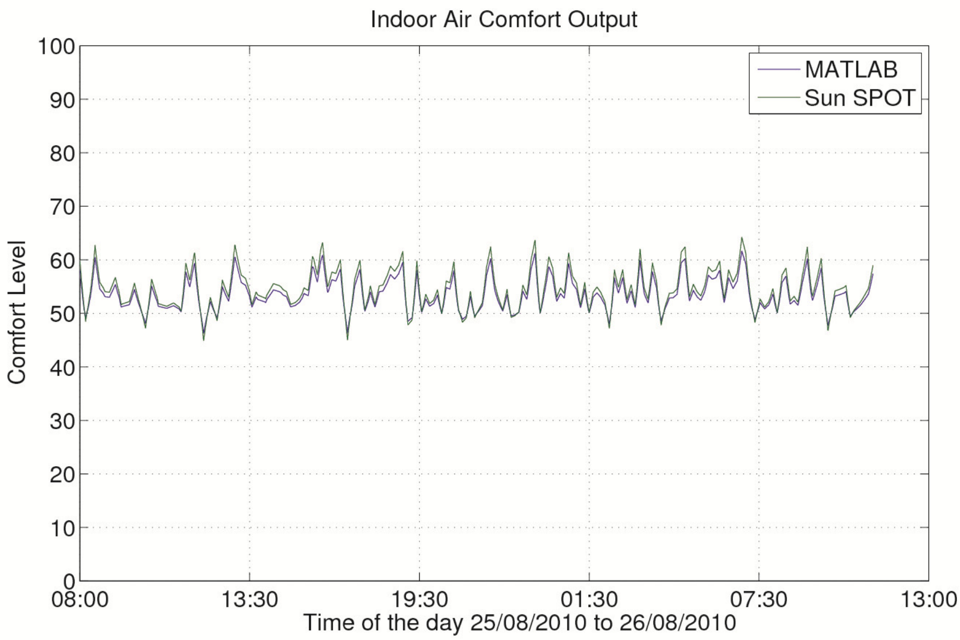


Figure 6-8. Indoor Air Comfort FRBS Output

Figure 6-8 [89, 95] illustrates stable indoor air comfort values (good – 46 to 65) throughout the day. The output then was compared with MATLAB and Simulink, as shown in Figure 6-8. It can be observed that the IAC FRBS Engine behaves similarly to

MATLAB and Simulink implementation, with 0% to 3.6% disparity. The output pattern is similar, thus confirming the correctness of the implementation on Sun SPOT.

6.2.4 Acoustical Comfort Subsystem

The acoustical comfort subsystem contains a *AC FRBS Engine* and an offline sound level reading sub routine. The subsystem was implemented as a virtual Sun SPOT because of the UNI-T UT351 Sound Level Meter auto power off feature, and high battery consumption issues when connected to the Sun SPOT node. Sound level data were recorded manually and fed into the virtual Sun SPOT node as offline mode operation.

The AC FRBS Engine was put into action based on its own fuzzy sets, as discussed in Chapter 4. Offline data from the UNI-T UT351 Sound Level Meter reading were fed into the AC FRBS Engine and acoustical comfort was determined.

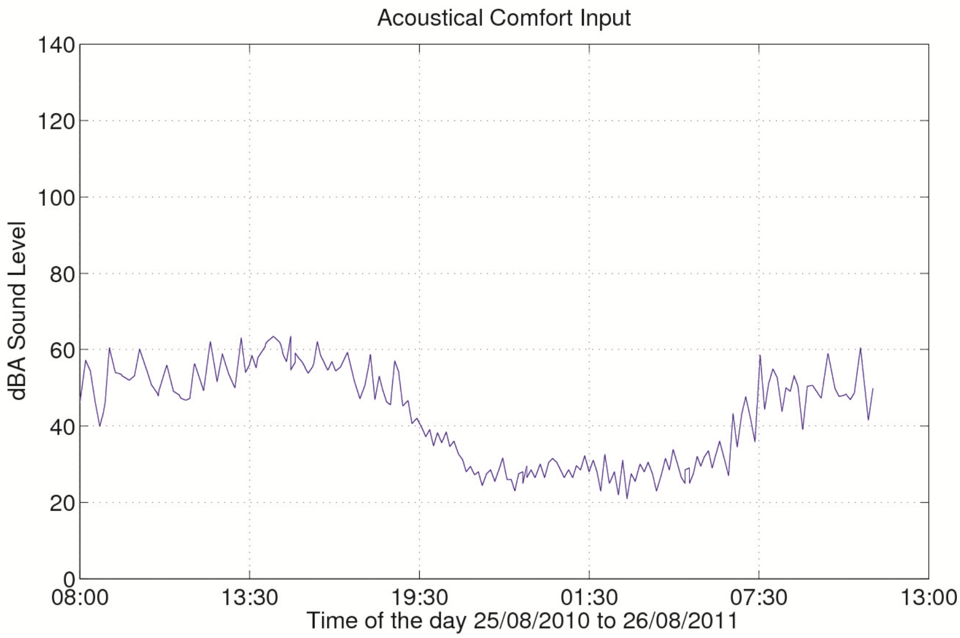


Figure 6-9. Acoustical Comfort FRBS Input

Figure 6-9 [89, 95] shows sound levels hovering between 40dBA to 65dBA during the day in the SeNSe laboratory. The background sound / noise mostly originated from outside of the building (road traffic noise, etc.) and from occupants' activities. After approximately 7:30pm, the sound level drops (20dBA to 35dBA) due to less activity in the laboratory and reduced outside sound / noise. At approximately 7:30am the next day, the background sound / noise rises again due to similar activities as those measured the previous day.

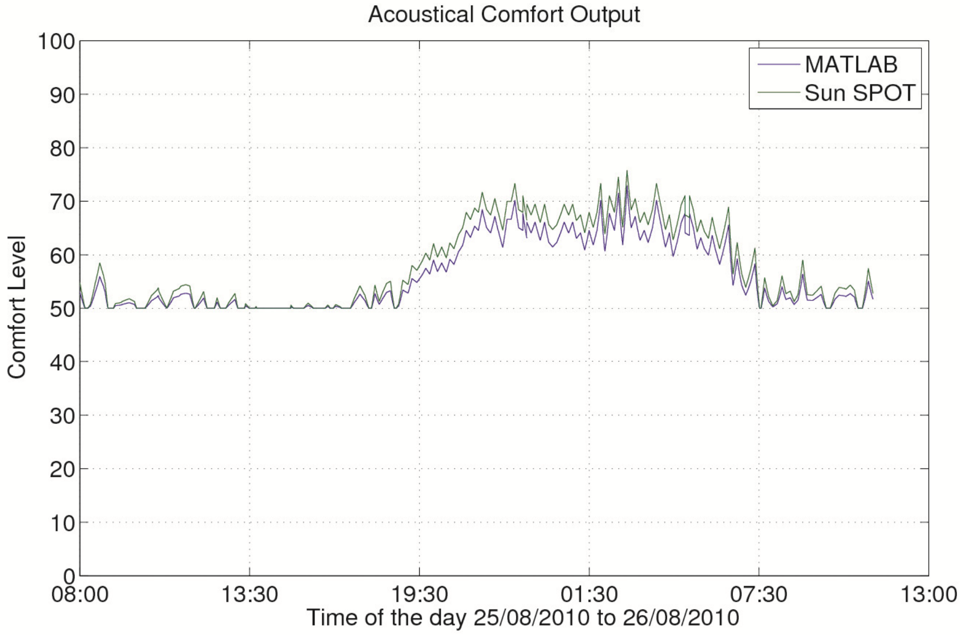


Figure 6-10. Acoustical Comfort FRBS Output

From Figure 6-10 [89, 95], the SeNSe laboratory can be deemed to have good acoustical comfort during the day and to be excellent at night. To confirm the accuracy of implementation on the Sun SPOT node, identical data were tested with MATLAB and Simulink. It can be seen in Figure 6-10 that the AC FRBS Engine behaves as intended, with 0% to 3.6% difference compared with MATLAB and Simulink, a small difference that is not perceivable by human ears.

6.2.5 HCAmI System Manager

The HCAmI System Manager is composed of two sub components – *HCI Engine* and *HC Manager*, as shown in Figure 4-5. This is a unique component that bridges the WSN realm and outside world realm (central PC).

HCI Engine

In making sense of HCI, the environment is scanned every ten minutes, giving a balance between the need for near real-time operation and WSN operational viability. The HCI Engine was embedded within a Sun SPOT basestation. It was tasked with the final undertaking of the HCAmI operation cycle.

The comfort values gathered from each comfort subsystem were fed into the HCI Engine and HCI was calculated. HCI comfort factor weights were used, as discussed in Chapter 4.

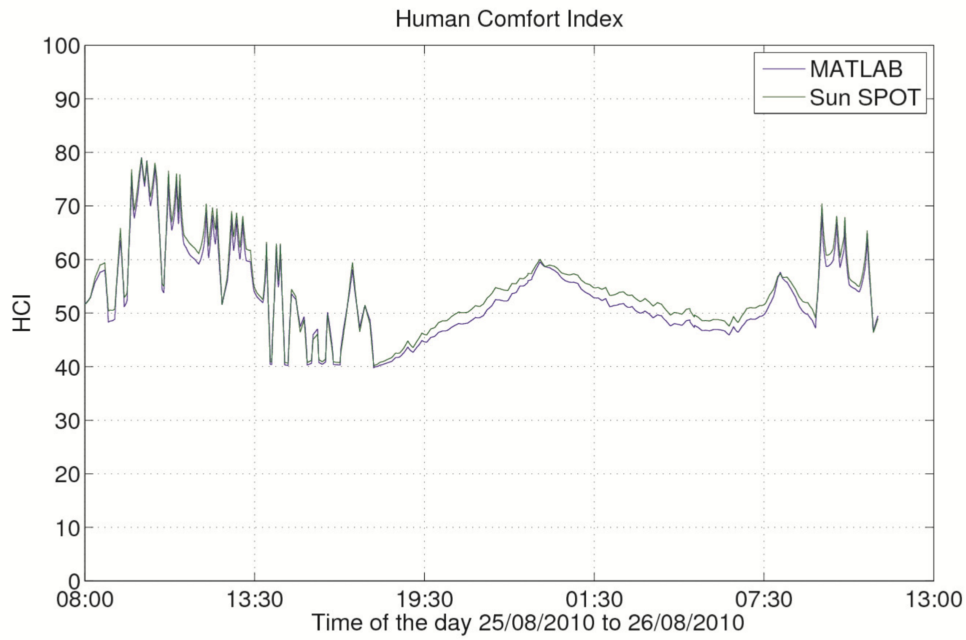


Figure 6-11. Human Comfort Index

Figure 6-11 [89, 95] reveals the HCI of the SeNSe laboratory. It hovers between 40 (Good) and 80 (Excellent). During the day, HCI values remain between good and excellent. The results indicate that the laboratory can be considered to be a good living space at that particular sensed time.

Figure 6-11 also shows the result as simulated by MATLAB and Simulink. Sun SPOT implementation demonstrates a similar HCI pattern as on MATLAB and Simulink, thus confirming the correctness of the engine. A small compound error can be observed. A small percentile inaccuracy of between 0% to 5% demonstrates the high degree of precision achieved.

On average, the `HCIEngine.class` needs less than 20ms to perform HCI calculations. There is a slight execution duration fluctuation that can be observed, caused by communication interruption in receiving comfort values from other comfort nodes. It can be very quick when all comfort values have already been received by the Sun SPOT basestation.

6.3 Chapter Summary

It can be observed that from the full scale HCAmI test bed presented, on average the HCAmI System operation consumed approximately 300ms to 320ms to accomplish one operation cycle. This is a small amount of time needed, relative to the ten minute operation cycle, thus a small amount of power is required during a Sun SPOT active period.

It can be seen that the Sun SPOT platform performed as expected, without any significant performance issues. The HCAmI System behaves as designed, even though the output is not 100% accurate compared with the MATLAB and Simulink simulation.

From a Sun SPOT power perspective, we can only observe the battery level. The battery level does not provide clear information on Sun SPOT node detail power utilisation. This is, unfortunately due to the Sun SPOT API library not providing accurate battery statistics and measurements to precisely quantify actual battery usage and performance.

Even though communication protocol is not directly related to the computational aspect of human comfort, it is vital for the system to have a reliable form of communication backbone. The work presents a broadcast based simple application layer protocol that distributes and shares sensor data within HCAmI System nodes. The protocol presented is light enough to be implemented, yet capable of delivering all sensor data and information effectively (minimal data lost for an extended sensing duration). The HCAmI Communication Protocol was found to be adequate to meet the essential requirements of HCAmI System communication and information sharing.

Chapter 7 Conclusion and Future Research

7.1 Introduction

This chapter concludes the research work that has been done for the research objectives as discussed in Chapter 1. This chapter also highlights the concluding remarks about the solution to the research problems and research questions as expressed in Chapter 1.

In this thesis, multi domain and multi comfort factors were initially discussed. A WSN based human comfort system framework was proposed to formulate a WSN architecture that utilises a fuzzy logic based intelligent engine as the core driving force in determining human comfort in a living space. Distributed fuzzy rule based system architecture was brought forward as the specific intelligent implementation architecture to be employed in the design of an HCAmI System.

A small scale Sun SPOT WSN testbed was developed to test the ideology. Acquired quantitative data from the testbed were used to validate the design and implementation against simulation on MATLAB and Simulink. The testbed case study presented shows that the HCAmI System produces a promising result in assisting better monitoring of living space.

A generic embedded human comfort framework and architecture was presented to bring forth new opportunities for the system to be adapted into different domain use such as human comfort in a bigger living space (hall, conference room, etc.) and open up possibilities for it to be used in different domains altogether (plant comfort – green house, livestock comfort – farming, etc.). Flexibility and adaptability gives this framework favourable prospect for future expansion and endeavour.

Finally, recommendations for feasible advancement of the work, possible future research and the research shortcomings are discussed.

7.2 Conclusions

This project has successfully demonstrated the use of WSN as a driving tool in understanding human comfort in a living space. The overall contribution of the thesis is the framework and architecture of an intelligent WSN based human comfort monitoring system in a living space that successfully combined multiple comfort factors, resulting in better understanding from the perspective of human comfort.

The main aim of this study was to investigate the use of limited recourse WSNs with embedded intelligent mechanism for monitoring and possibly managing human comfort in a living space. Six key objectives were identified to address this aim.

In this section, the conclusions of the research as stated in the main objectives formulated in Chapter 1 are presented.

i) *To investigate and analyse the problem related to individual comfort factors and human comfort.*

After developing separate individual comfort model (computationally with MATLAB and Simulink) for thermal comfort, visual comfort, indoor air comfort and acoustical comfort, the resulting models were tested experimentally with various WSN platforms. The Sun SPOT WSN platform was selected. It provided the right balance between computational capability, ease of development and implementation and sensor expansion capability.

A variety of aggregation methods that combined various comfort factors into a single human comfort index were investigated and simulated. Methods from a complex neural network base to a simple weighted averaging explored. Due to the lack of quantitative data in the literature and the lack of profound understanding within the human comfort knowledge domain, the study eventually settled on a simple weighted average form of aggregation. The weighted average method gives quick adjustment possibilities (with the addition of new comfort factors and new comfort weights) for leveraging multiple comfort factors without need for prior training data, as required by the neural network method.

The requirement for human comfort from various comfort factor perspectives and maximising the role of sensors to provide a human comfort solution was achieved in the form of a generic HCAmI System framework and architecture.

ii) *To design and develop a WSN based human comfort ambient intelligence system architecture and framework.*

An extensive literature review was presented, revealing that despite extensive research into human comfort and various individual comfort factors, embedded intelligence and WSN, many aspects of multi domain comfort systems are not yet fully understood or have never been fully integrated. A novel architecture and framework were proposed here to unite this separate multi domain research into a unified solution. The HCAmI System architecture and framework advances the unification of the research. The HCAmI System model led to the development of a small scale testbed to access and alters the human comfort levels of a living space effectively. The HCAmI System framework was designed to provide a generic solution to the human comfort predicament.

Experiments / testbeds were constructed and conducted to validate the framework and architecture against a simulated model on Sun SPOT Solarium, MATLAB and Simulink.

iii) *To design and develop intelligent FRBS engine WSN with low computational cost for individual comfort factor and to analyse its performance under different scenarios.*

A generic HCAmI System FRBS engine that can be tailored to individual comfort needs was built. Initially, the engine was modelled with a MATLAB Fuzzy Toolbox in several different configurations – a different inference engine (Mamdani and Sugeno), a membership function variable (type – trimf, trapmf, etc), a number of input and output functions and inference rules were involved. The most simplistic characteristic was chosen and presented in this thesis. This characteristic represents the leanest computational load for the WSN node. The FRBS engine operation accuracy was compared with its corresponding model in MATLAB and Simulink as discussed in Chapter 6.

The innovation of this generic FRBS engine is that it was designed in such way that it starts as a 'blank' fuzzy engine. The FRBS engine can be adapted to solve any fuzzy problem such as thermal comfort, visual comfort, air indoor comfort and acoustical comfort factor in each sensor node over the air. This opens a vast opportunity for adapting the engine for any problem domain whatsoever.

iv) *To design and develop WSN based thermal comfort engine.*

Thermal comfort is the most complex, influential comfort factor in predicting human comfort. Thermal comfort warrants extra effort, with a myriad of existing result from studies. Various strategies and methods have been proposed, but this thesis went back to basics with Fanger's PMV equation, on which all other method and techniques were based. The PMV equation is a piece of computationally complex code and requires huge computational capability.

The PMV equation was constructed for various WSN platforms and from experiments conducted, it appeared the Sun SPOT WSN platform was a suitable candidate to be embedded with the engine. From the PMV's computational complexity code one can deduce the Sun SPOT WSN platform is a capable platform for delivering complex embedded tasks.

Simulations and experiments were conducted with real sensor data to evaluate the accuracy of the engine. The output was presented and compared with similar engines modelled in MATLAB for its accuracy confirmation.

The main aim of this work, to understand the interconnected WSN domain, the embedded intelligence domain and the multi factor human comfort level of a living space has been achieved, to a large extent, in this work. In general, the testbed and modelling work has shown the soundness of the ideology proposed, with the delivery of promising results from these particular goals and this framework.

7.3 Recommendations and Future Research

This thesis has presented an initial framework and architecture system that deals with human comfort. This section suggests some promising directions for the WSN driven environmental comfort system. Concerning future enhancement of the HCAmI System, further improvements can be explored to improve overall system operation:

i) *HCI Weight values.* These weight values could be configured / changed based on user preferences and the system's self-learning process via the system personalisation function. This added functionality alone deserves its own study domain.

ii) *HC Manager.* Further enhancement work could be done on HC Manager on managing sensor nodes with multi sensors and multi nodes of various kinds. The scale up system limit could be investigated further, such as a zoning system for larger living spaces, quadrant based evaluation and so on. Optimisation of these dynamics presents extensive challenges to the design.

iii) *FRBS Engine.* This thesis employed the simplest fuzzy inference engine. With a better WSN platform, a more versatile and powerful engine could be incorporated. This provides an opportunity for more complicated intelligent implementation within WSN.

iv) *Spatial Sensing Engine.* This study considers stationary WSN nodes could perform the sensor value where a real sensor is not available. In real life implementation, living space occupants are useually mobile; we therefore recommend extending this capability into mobile nodes.

v) *Communication Protocol.* This study presents a simplified version of an application level protocol. Even though it shows adequate capability in delivering messages within sensor node, its limits, performance, reliability and accuracy should be analysed further. Better communication schemes could be explored – broadcast, multicast and unicast schemes.

vi) *Platform.* This research concentrated on a high power Sun SPOT WSN platform. This high power platform works well with an extended power source that is easily available within living space. For further expansion, it is presumably important to investigate energy factors as they relate to the sensing node / network platform.

In conclusion, the HCAmI System can be seen as pioneering work in combining multidisciplinary research on human comfort. The HCAmI System flexible framework and architecture can be easily adapted to different uses / applications. This study unlocks future research potential for a plethora of specific research focuses.

References

[1] S. Dobson and K. Delaney, "Embedded Intelligence," *ERCIM News,* pp. 37 - 38, 2006.

[2] D. J. Cook and S. K. Das, *Smart Environments: Technologies, Protocols, and Applications*. New York: John Wiley, 2004.

[3] E. Schoitsch and A. Skavhaug, "Embedded Intelligence," *ERCIM News,* pp. 14 - 15, 2006.

[4] L. W. W. Choong. (2010, 15 January). *Ambient Intelligence* [Online]. Available: http://ami-lab.org/

[5] J. C. Augusto, "Ambient Intelligence: The Confluence of Ubiquitous / Pervasive Computing and Artificial Intelligence," in *Intelligent Computing Everywhere*, A. Schuster, Ed., ed: Springer London, 2007, pp. 213 - 234.

[6] C. Ramos, J. C. Augusto, and D. Shapiro, "Ambient Intelligence - The Next Step for Artificial Intelligence," *IEEE Intelligent Systems,* vol. 23, pp. 15 - 18, 2008.

[7] J. A. Stankovic, "Wireless Sensor Network," in *Handbook of Real-Time and Embedded Systems*, I. Lee, J. Y.-T. Leung, and S. H. Son, Eds., ed: CRC Press, 2007.

[8] E. H. A. Callaway, *Wireless Sensor Networks: Architectures and Protocols*: Auerbach Publications, 2004.

[9] J. Kabara and M. Calle, "MAC Protocols Used by Wireless Sensor Networks and a General Method of Performance Evaluation," *International Journal of Distributed Sensor Networks,* vol. 2012, 2012.

[10] E. Charniak and D. McDermott, *Introduction To Artificial Intelligence*. Reading MA: Addison-Wesley, 1985.

[11] W. P. Henry, *Artificial Intelligence, Third Edition*. Reading MA: Addison-Wesley, 1992.

[12] S. Marzano and E. Aarts, *The New Everyday View on Ambient Intelligence*. Rotterdam: OIO Publishers, 2003.

[13] A. P. Wierzbicki, "The Concept of Ambient Intelligence and Decision Support for Telecommunications," *Journal of Telecommunications and Information Technology,* vol. 3, pp. 3 - 6, 2002.

[14] M. Weiser, R. Gold, and J. S. Brown, "The Origins of Ubiquitous Computing Research at PARC In The Late 1980s," *IBM Systems Journal,* vol. 38, 1999.

[15] M. Langheinrich, V. Coroamă, J. Bohn, and F. Mattern, "Living in a Smart Environment – Implications for the Coming Ubiquitous Information Society," *Telecommunications Review,* vol. 15, pp. 132 - 143, 2005.

[16] A. K. Dey, G. D. Abowd, and D. Salber, "A Context-Based Infrastructure for Smart Environments," in *Proceedings of the 1st International Workshop on Managing Interactions in Smart Environments (MANSE'99)*, Dublin, Ireland, 1999, pp. 114 - 128.

[17] S. Poslad, *Ubiquitous Computing: Smart Devices, Environments and Interactions*: John Wiley & Sons, 2009.

[18] W. Feist. (2009, 15 March). *What is a Passive House* [Online]. Available: http://www.passivehouse.com/english/PassiveH.htm

[19] EN15251, "EN 15251 Indoor Environmental Input Parameters for Design and Assessment of Energy Performance of Buildings Addressing Indoor Air Quality, Thermal Environment, Lighting and Acoustics," 2007-09-01 ed: CEN, Brussels, 2007, p. 51.

[20] ISO7730, "ISO 7730 Ergonomics of The Thermal Environment — Analytical Determination and Interpretation of Thermal Comfort Using Calculation of The PMV and PPD Indices and Local Thermal Comfort Criteria," 3rd ed. Switzerland: International Standards Organization, Geneva, Switzerland, 2005, p. 60.
[21] K. Parsons, F. Nicol, M. Humphreys, O. Sykes, and S. Roaf, *Standards for Thermal Comfort: Indoor Air Temperature Standards for the 21st Century*. London, UK: Chapman & Hall, 1995.
[22] ASHRAE, *ASHRAE Fundamental Handbook*: American Society of Heating, Refrigerating and Air-Conditioning Engineers, 2005.
[23] P. O. Fanger, *Thermal Comfort*: McGraw-Hill, New York, 1972.
[24] W. T. Grondzik, A. G. Kwok, B. Stein, and J. S. Reynold, *Mechanical and Electrical Equipment for Buildings*, 11th Edition ed.: John Wiley & Sons, 2009.
[25] T. Incorporated, *Indoor Air Quality Handbook*. Shoreview, USA: TSI Incorporated, 2003.
[26] J. D. Spengler, J. M. Samet, and J. F. McCarthy, *Indoor Air Quality Handbook*: McGraw-Hill, 2001.
[27] ISO16813, "ISO 16813 Building environment design — Indoor Environment — General principles," 2006-05-15 ed: International Standards Organization, Geneva, Switzerland, 2006.
[28] J. S. Bradley and B. N. Gover, "Criteria for Acoustic Comfort in Open-Plan Offices," in *The 33rd International Congress and Exposition on Noise Control Engineering*, Prague, Czech Republic, 2004.
[29] R. M. Reffat and E. L. Harkness, "Expert System for Environmental Quality Evaluation," *Journal of Performance on Constructed Facilities,* vol. 15, pp. 109 - 114, 2001.
[30] R. M. Reffat and E. L. Harkness, "Environmental Comfort Criteria: Weighting and Integration," *Journal of Performance of Constructed Facilities,* vol. 15, pp. 104 - 108, 2001.
[31] C. Fischione, K. H. Johansson, F. Graziosi, and F. Santucci, "Distributed Cooperative Processing and Control over Wireless Sensor Networks," presented at the The 2006 International Wireless Communications and Mobile Computing Conference (IWCMC 2006), Sheraton Wall Centre,Vancouver, British Columbia, Canada, 2006.
[32] G. Naadimuthu, D. M. Liu, and E. S. Lee, "Application of an Adaptive Neural Fuzzy Inference System to Thermal Comfort and Group Technology Problems," *Computers and Mathematics with Applications,* vol. 54, pp. 1395 - 1402, 2007.
[33] U. Eicker, "Climate Control and Summer Comfort in Passive Office Buildings," in *13th International Passive House Conference*, Frankfurt, Germany, 2009, pp. 421 - 428.
[34] G. Alessia, T. Alexandra, S. Wolfram, and B. Paolo, "Comfort Analysis of A Passive House at Different Locations in Italy," in *13th International Passive House Conference*, Frankfurt, Germany, 2009, p. 93.
[35] M. Hamdi, G. Lachiver, and F. Michaud, "A New Predictive Thermal Sensation Index of Human Response," *Energy and Buildings,* vol. 29, pp. 167 - 178, 1999.
[36] L. Krajcsovics, "Passive House in The Climate Conditions of Slovakia," in *13th International Passive House Conference*, Frankfurt, Germany, 2009, pp. 99 - 104.

[37] S. Atthajariyakul and T. Leephakpreeda, "Real-time Determination of Optimal Indoor-air Condition for Thermal Comfort, Air Quality and Efficient Energy Usage," *Energy and Buildings,* vol. 36, pp. 720 - 733, 2004.

[38] A. Taherkordi, R. Rouvoy, Q. Le-Trung, and F. Eliassen, "A Self-Adaptive Context Processing Framework for Wireless Sensor Networks," in *MidSens '08*, Leuven, Belgium, 2008.

[39] M.-W. Feng, S.-L. Wen, K.-C. Tsai, Y.-C. Liu, and H.-R. Lai, "Wireless Sensor Network and Sensor Fusion Technology for Ubiquitous Smart Living Space Applications," in *2008 Second International Symposium on Universal Communication*, Osaka, Japan, 2008, pp. 295 - 302.

[40] D. Surie, O. Laguionie, and T. Pederson, "Wireless Sensor Networking of Everyday Objects in a Smart Home Environment," in *International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP 2008)*, Sydney, Australia, 2008.

[41] D. Surie, F. Lagriffoul, T. Pederson, and D. Sjölie, "Activity Recognition Based on Intra and Extra Manipulation of Everyday Objects," in *IFIP UCS Conference on Ubiquitous Computing Systems*, 2007, pp. 196 - 210.

[42] G. Davis, *Building Performance: Function, Preservation, and Rehabilitation*. Baltimore: American Society for Testing & Materials, 1986.

[43] G. Virone, A. Wood, L. Selavo, Q. Cao, L. Fang, T. Doan*, et al.*, "An Advanced Wireless Sensor Network for Health Monitoring," in *Transdisciplinary Conference on Distributed Diagnosis and Home Healthcare (D2H2)*, 2005, pp. 2 - 4.

[44] E. J. Pauwels, A. A. Salah, and R. Tavenard, "Sensor Networks for Ambient Intelligence," in *Workshop on Multimedia Signal Processing (MMSP)*, Chania, Crete, Greece, 2007.

[45] M. D'Souza, T. Wark, and M. Ros, "Wireless Localisation Network for Patient Tracking," in *International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP 2008)*, Sydney, Australia, 2008.

[46] J. A. Krill, K. W. O'Haver, M. J. O'Driscoll, I.-J. Wang, and D. G. Lucarelli, "Prototype Design and Experimental Results for an Intruder Detection Swarming Network," in *International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP 2008)*, Sydney, Australia, 2008.

[47] A. Sashima, Y. Inoue, T. Ikeda, T. Yamashita, and K. Kurumatani, "CONSORTS-S: A Mobile Sensing Platform for Context-Aware Services," in *International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP 2008)*, Sydney, Australia, 2008.

[48] J. Eickhold, T. Fuhrmann, B. Saballus, S. Schlender, and T. Suchy, "AmbiComp: A Platform for Distributed Execution of Java Programs on Embedded Systems by Offering a Single System Image," presented at the European Conference on Ambient Intelligence 2008, Nuremberg, Germany, 2008.

[49] B. Saballus, J. Eickhold, and T. Fuhrmann, "Global Accessible Objects (GAOs) in the Ambicomp Distributed Java Virtual Machine," presented at the International Conference on Sensor Technologies and Applications (SENSORCOMM 2008), Cap Esterel, France, 2008.

[50] U. Rutishauser, J. Joller, and R. Douglas, "Control and Learning of Ambience by an Intelligent Building," *IEEE Transactions On Systems, Man, and Cybernetics,* vol. 35, pp. 121 - 132, 2005.

[51] K. Shah and M. Kumar, "Resource Management in Wireless Sensor Networks using Collective Intelligence," in *International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP 2008)*, Sydney, Australia, 2008.
[52] A. M. Omer, "Renewable Building Energy Systems and Passive Human Comfort Solutions," *Renewable & Sustainable Energy Reviews,* vol. 12, pp. 1562 - 1587, 2008.
[53] A. F. Emery. (2010, 15 April). *Human Comfort and Health Requirements* [Online]. Available: http://courses.washington.edu/me333afe/Comfort_Health.pdf
[54] A. I. Dounis, M. J. Santamouris, C. C. Lefas, and A. Argiriou, "Design of A Fuzzy Set Environment Comfort System," *Energy and Buildings,* vol. 22, pp. 81 - 87, 1995.
[55] M. Gouda, S. Danaher, and C. Underwood, "Thermal Comfort Based Fuzzy Logic Controller," *The Chartered Institution of Building Services Engineers,* vol. 22, pp. 237 - 253, 2001.
[56] D. Kolokotsa, D. Tsiavos, G. S. Stavrakakis, K. Kalaitzakis, and E. Antonidakis, "Advanced Fuzzy Logic Controllers Design and Evaluation for Buildings' Occupants Thermal-Visual Comfort and Indoor Air Quality Satisfaction," *Energy and Buildings,* vol. 33, pp. 531 - 543, 2001.
[57] W. Liu, Z. Lian, and B. Zhao, "A Neural Network Evaluation Model for Individual Thermal Comfort," *Energy and Buildings,* vol. 39, pp. 1115 - 1122, 2007.
[58] A. Kemajou, L. Mba, and P. Meukam, "Application of Artificial Neural Network for Predicting the Indoor Air Temperature in Modern Building in Humid Region," *British Journal of Applied Science & Technology,* vol. 2, pp. 23 - 34, 2012.
[59] G. Steffy, *Architectural Lighting Design*, 3 ed.: John Wiley & Sons, Inc, 2008.
[60] D. L. DiLaura, K. W. Houser, R. G. Mistrick, and G. R. Steffy, *The IES Lighting Handbook, Tenth Edition*: Illuminating Engineering Society, 2011.
[61] HSE, *Lighting at Work*: HSE Books, 1998.
[62] A. K. Persily, "Evaluating Building IAQ and Ventilation With Indoor Carbon Dioxide," *Transactions-American Society of Heating Refrigerating and Air Conditioning Engineers,* vol. 103, pp. 193 - 204, 1997.
[63] M. Pavčeková, M. Rychtáriková, and P. Tomašovič, "Acoustical Comfort Predictions in Classrooms With Two Simultaneous Sound Sources," *Slovak Journal of Civil Engineering,* vol. 2009, pp. 17 - 25, 2009.
[64] ISO3382, "ISO 3382:1997 Acoustics - Measurement of the Reverberation Time of A Room With Reference to Other Acoustical Parameters," ed: International Standards Organization, Geneva, Switzerland, 1997.
[65] M. Katayama, "Field Measurement on the Comfort Level of Acoustic Environment in Coastal Zone," in *OCEANS '02 MTS/IEEE*, 2002, pp. 1051 - 1054.
[66] M. V. Damme, M. Blasco, C. Crispin, P. Huart, B. Ingelaere, and D. Wuyts, "Acoustic Comfort In Timber Frame Dwellings," *Building Acoustics,* vol. 14, pp. 15 - 24, 2007.
[67] B. Goujard, A. Sakout, and V. Valeau, "Acoustic Comfort On Board Ships: An Evaluation Based On a Questionnaire," *Applied Acoustics,* vol. 66, pp. 1063 - 1073, 2005.

[68] P. H. T. Zannin and C. R. Marcon, "Objective And Subjective Evaluation of The Acoustic Comfort In Classrooms," *Applied Ergonomics,* vol. 38, pp. 675 - 680, 2007.
[69] C. Haka, M. v. Haaren, R. Wenmaekersc, and R. v. Luxemburg, "Measuring Room Acoustic Parameters Using a Head And Torso Simulator Instead of An Omnidirectional Microphone," in *Inter-Noise 2009*, Ottawa, Canada, 2009.
[70] E. Nilsson, "Room Acoustic Measures For Classroms," in *Inter-Noise 2010*, Lisbon, Portugal, 2010.
[71] B. Chen and J. Kang, "Acoustic Comfort in Shopping Mall Atrium Spaces - A Case Study in Sheffield Meadowhall," *Architectural Science Review,* vol. 47, pp. 107 - 114, 2004.
[72] J. A. Orosa, "Research on General Thermal Comfort Models," *European Journal of Scientific Research,* vol. 27, pp. 217 - 227, 2009.
[73] D. Lee, "Development of Light Powered Sensor Networks for Thermal Comfort Measurement," *Sensors,* vol. 8, pp. 6417 - 6432, 2008.
[74] H. Mirinejad, S. H. Sadati, M. Ghasemian, and H. Torab, "Control Techniques in Heating, Ventilating and Air Conditioning (HVAC) Systems," *Journal of Computer Science,* vol. 4, pp. 777 - 783, 2008.
[75] J. Liang and R. Du, "Thermal Comfort Control Based on Neural Network for HVAC Application," in *IEEE Conference on Control Applications*, Toronto, Canada, 2005, pp. 819 - 824.
[76] M. Soleimani-Mohseni, B. Thomas, and P. Fahle´n, "Estimation of Operative Temperature in Buildings Using Artificial Neural Networks," *Energy and Buildings,* vol. 38, pp. 635 - 640, 2005.
[77] K. Chen, Y. Jiaob, and E. S. Lee, "Fuzzy Adaptive Networks in Thermal Comfort," *Applied Mathematics Letters,* vol. 19, pp. 420 – 426, 2005.
[78] M. T. Lah, B. Zupancic, and A. Krainer, "Fuzzy Control for The Illumination and Temperature Comfort in a Test Chamber," *Building and Environment,* vol. 40, pp. 1626 – 1637, 2005.
[79] T. Wark, W. Hu, P. Corke, J. Hodge, A. Keto, B. Mackey*, et al.*, "Springbrook: Challenges in Developing a Long-Term, Rainforest Wireless Sensor Network," in *International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP 2008)*, Sydney, Australia, 2008.
[80] I. Jantunen, H. Laine, P. Huuskonen, D. Trossen, and V. Ermolov, "Smart Sensor Architecture for Mobile-Terminal-Centric Ambient Intelligence," *Sensors and Actuators A,* vol. 142, pp. 352 – 360, 2007.
[81] S. Microsystems. (2009, 19 February). *Sun SPOT Manager Tool* [Online]. Available: http://www.sunspotworld.com/SPOTManager/index.html
[82] S. Microsystems. (2009, 18 November). *Solarium User's Guide* [Online]. Available: http://www.sunspotworld.com/docs/index.html
[83] S. Microsystems. (2009, 18 November). *Sun™ Small Programmable Object Technology (Sun SPOT) Developer's Guide* [Online]. Available: http://www.sunspotworld.com/docs/index.html
[84] S. Microsystems. (2009, 15 January). *Home of Sun SPOT World* [Online]. Available: http://www.sunspotworld.com/
[85] S. Microsystems. (2009, 5 August). *Sun™ SPOT Owner's Manual* [Online]. Available: http://www.sunspotworld.com/docs/index.html
[86] S. Microsystems. (2009, 1 February). *Squawk Virtual Machine* [Online]. Available: http://java.net/projects/squawk/pages/SquawkDevelopment

[87] SENSIRION. (2009, 5 March). *SHT1x (SHT10, SHT11, SHT15) Humidity and Temperature Sensor* [Online]. Available: http://www.sensirion.com/humidity
[88] Precon. (2010, 11 May). *Precon HS-2000V Precision Humidity Sensors* [Online]. Available: http://www.preconusa.com/humidity_moisture_dew_sensors.htm
[89] M. I. M. Rawi and A. Al-Anbuky, "Wireless Sensor Networks and Human Comfort Index," *Personal and Ubiquitous Computing,* pp. 1 - 13, 2012.
[90] A. Muñoz, J. C. Augusto, A. Villa, and J. A. Botía, "Design and evaluation of an ambient assisted living system based on an argumentative multi-agent system," *Personal and Ubiquitous Computing,* vol. 15, pp. 377 - 387, 2011.
[91] J. P. García-Vázquez, M. D. Rodríguez, Á. G. Andrade, and J. Bravo, "Supporting the Strategies to Improve Elders' Medication Compliance by Providing Ambient Aids," *Personal and Ubiquitous Computing,* vol. 15, pp. 389 - 397, 2011.
[92] J. C. Augusto, M. Huch, and A. Kameas. (2012). *Handbook of Ambient Assisted Living : Technology for Healthcare, Rehabilitation and Well-being* [Online]. Available: http://AUT.eblib.com.au/patron/FullRecord.aspx?p=836207
[93] M. I. M. Rawi and A. Al-Anbuky, "Passive House Sensor Networks: Human Centric Thermal Comfort Concept," in *Fifth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP 2009)*, Melbourne, Australia, 2009, pp. 255 – 260.
[94] F. K. Aldrich, "Smart Homes: Past, Present and Future," in *Inside the Smart Home*, R. Harper, Ed., ed London: Springer, 2003, pp. 17 - 39.
[95] M. I. M. Rawi and A. Al-Anbuky, "Development of Intelligent Wireless Sensor Networks for Human Comfort Index Measurement," in *The 2nd International Conference on Ambient Systems, Networks and Technologies (ANT)*, Niagara Falls, Canada, 2010, pp. 232 - 239.
[96] CIBSE, *CIBSE Code for Interior Lighting*. London, United Kingdom: Chartered Institution of Building Services Engineers, 1994.
[97] CIBSE, *CIBSE Guide: Air Infiltration and Natural Ventilation*. London, United Kingdom: Chartered Institution of Building Services Engineers, 1994.
[98] E. Commission, *Green Paper on Future Noise Policy (COM(96) 540)*. Brussels, Belgium: European Commission, 1996.
[99] ISO1999, "ISO 1999:1990 Acoustics - Determination of Occupational Noise Exposure and Estimation of Noise Induced Hearing Impairement," ed: International Standards Organization, Geneva, Switzerland, 1999.
[100] D. Shepard, "A Two-Dimensional Interpolation Function For Irregularly-Spaced Data," in *1968 23rd ACM National Conference*, New York, USA, 1968, pp. 517 - 524.
[101] K. V. D. Linden, A. C. Boerstra, A. K. Raue, and K. Stanley R, "Thermal Indoor Climate Building Performance Characterized By human Comfort Response," *Energy and Buildings,* vol. 34, pp. 737 - 744, 2002.
[102] Y. Epstein and D. S. Moran, "Thermal Comfort and the Heat Stress Indices," *Industrial Health,* vol. 44, pp. 388 - 398, 2006.
[103] V. A. C. d. Graça, D. C. C. K. Kowaltowski, and J. R. D. Petreche, "An Evaluation Method For School Building Design at The Preliminary Phase With Optimisation of Aspects of Environmental Comfort For The School System of The State São Paulo in Brazil," *Building and Environment,* vol. 42, 2007.

[104] E. Benoit, R. Dapoigny, and L. Foulloy, "Fuzzy-Based Intelligent Sensors: Modeling, Design, Application," in *8th IEEE International Conference on Emerging Technologies and Factory Automation*, Juan les Pins, France, 2001, pp. 401 - 407.
[105] E. H. Mamdani and S. Assilian, "An Experiment in Linguistic Synthesis With a Fuzzy Logic Controller," *International Journal of Man-Machine Studies,* vol. 7, pp. 1 - 13, 1975.
[106] S. N. Sivanandam, S. Sumathi, and S. N. Deepa, *Introduction to Fuzzy Logic Using MATLAB*. Berlin Heidenberg New York: Springer, 2006.
[107] R. Pi. (2012, 5 March). *Raspberry Pi* [Online]. Available: http://www.raspberrypi.org/
[108] F. C. Houghton and C. P. Yaglou, "Determining Lines of Equal Comfort," *ASHVE Transactions,* vol. 29, pp. 163 - 176, 1923.
[109] A. P. Gagge, A. J. Stolwijk, and Y. Nish, "An Effective Temperature Scale Based On a Simple Model of Human Physiological Regulatory Response," *ASHRAE Transactions,* vol. 77, pp. 247 - 257, 1971.
[110] R. R. Gonzalez and A. P. Gagge, "Magnitude Estimates of Thermal Discomfort During Transients of Humidity and Operative Temperature and Their Relation to The New ASHRAE Effective Temperature," *ASHRAE Transactions,* vol. 79, pp. 88 - 96, 1973.
[111] C. P. Yaglou, "Control of Heat Casualties at Military Training Centers," *A.M.A. Archives of Industrial Health,* vol. 16, pp. 302 - 316, 1957.
[112] E. C. Thom, "The Discomfort Index," *Weatherwise,* vol. 12, pp. 57 - 60, 1959.
[113] J. Pepi, "The Summer Simmer Index," *Weatherwise,* vol. 40, pp. 143 - 145, 1987.
[114] E. Sazonov. (2011, 12th February). *Open Source Fuzzy Inference Engine for Java* [Online]. Available: http://people.clarkson.edu/~esazonov/FuzzyEngine.htm

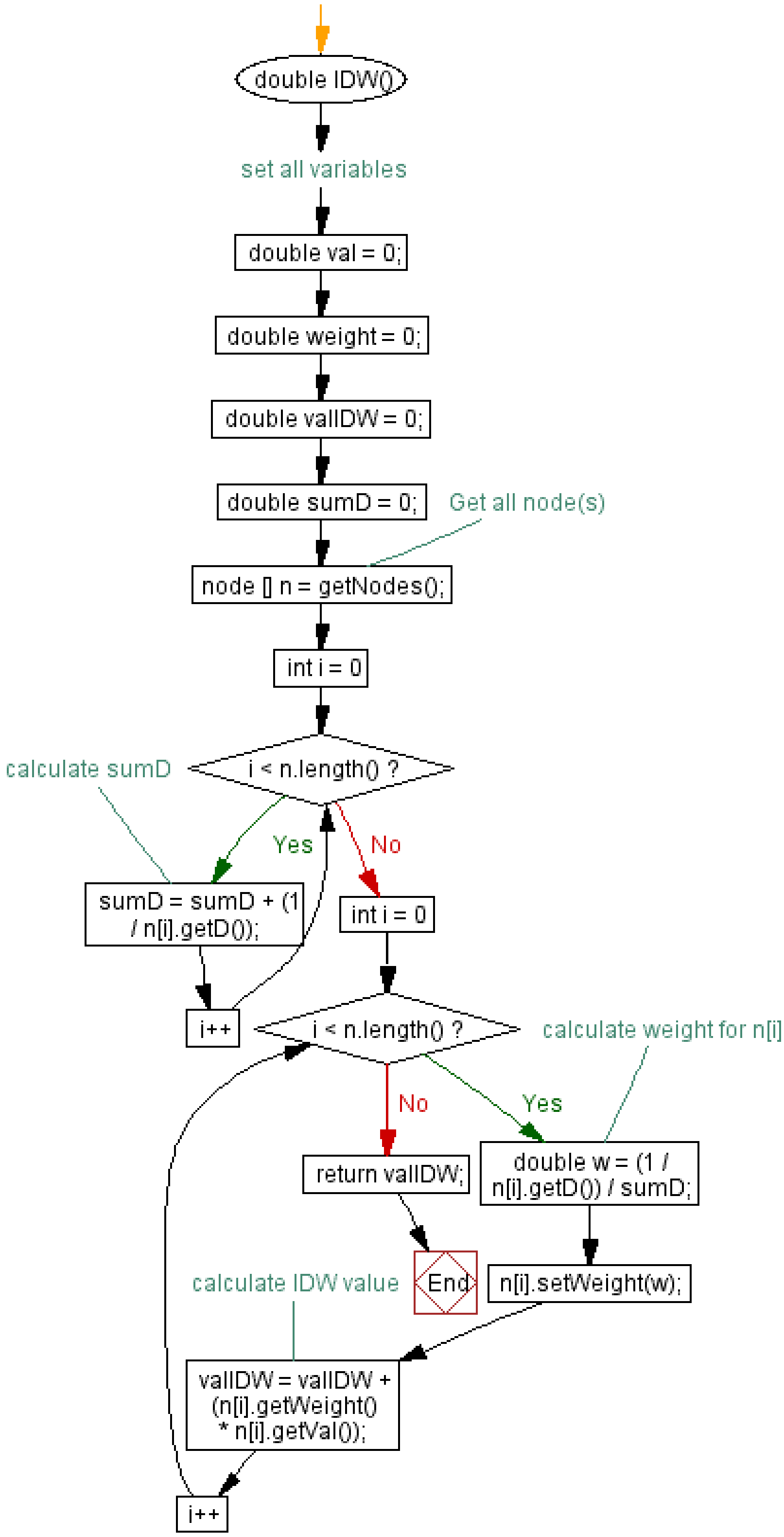
Appendix A Published Papers

[1] M. I. M. Rawi and A. Al-Anbuky, "Passive House Sensor Networks: Human Centric Thermal Comfort Concept," in *Fifth International Conference on Intelligent Sensors, Sensor Networks and Information Processing ISSNIP 2009*, Melbourne, Australia, 2009, pp. 255 – 260.

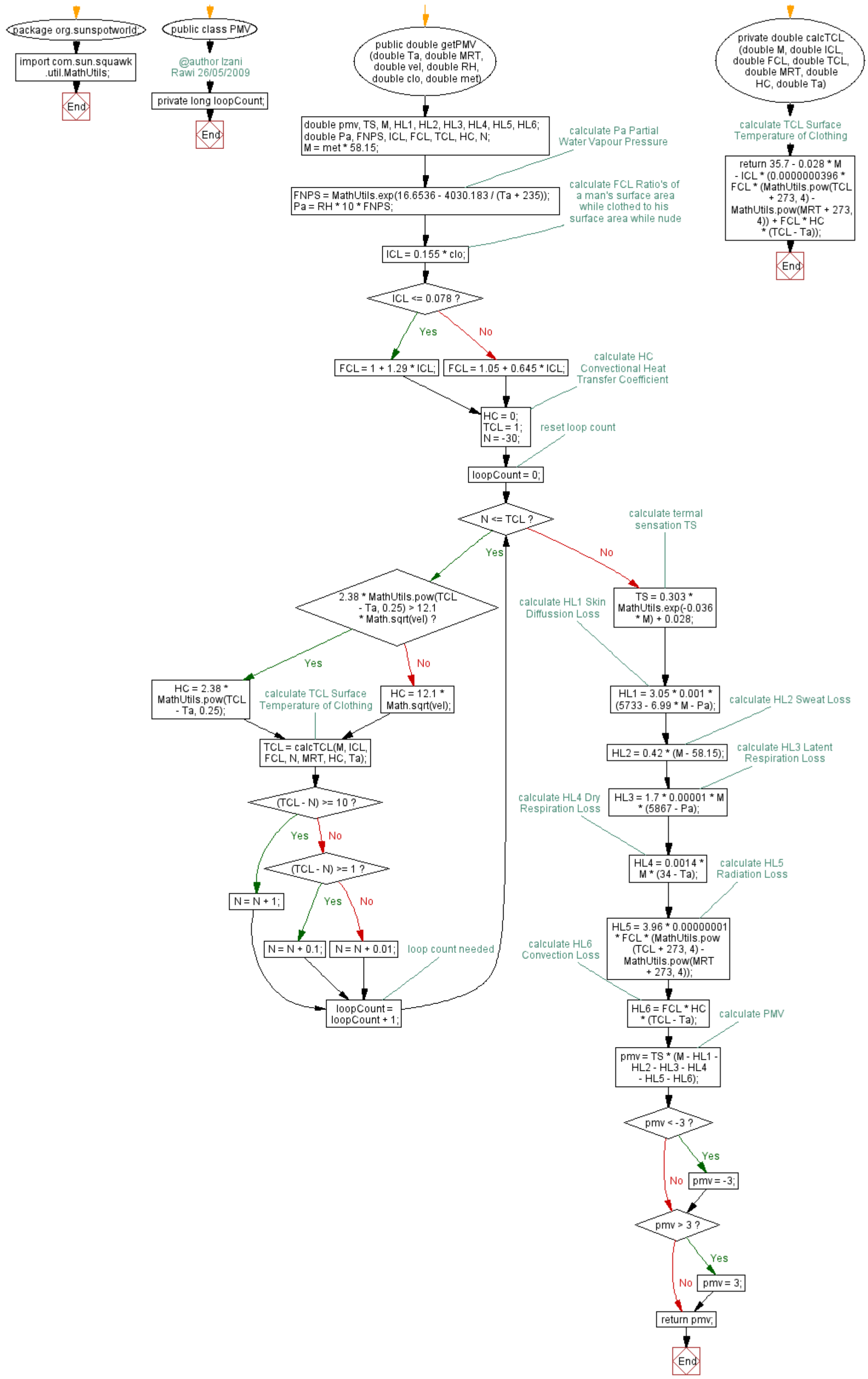
[2] M. I. M. Rawi and A. Al-Anbuky, "Development of Intelligent Wireless Sensor Networks for Human Comfort Index Measurement," in *The 2nd International Conference on Ambient Systems, Networks and Technologies (ANT)*, Niagara Falls, Canada, 2010, pp. 232-239.

[3] M. I. M. Rawi and A. Al-Anbuky, "Wireless Sensor Networks and Human Comfort Index," *Personal and Ubiquitous Computing,* pp. 1-13, 01/05/2012 2012. DOI 10.1007/s00779-012-0547-09

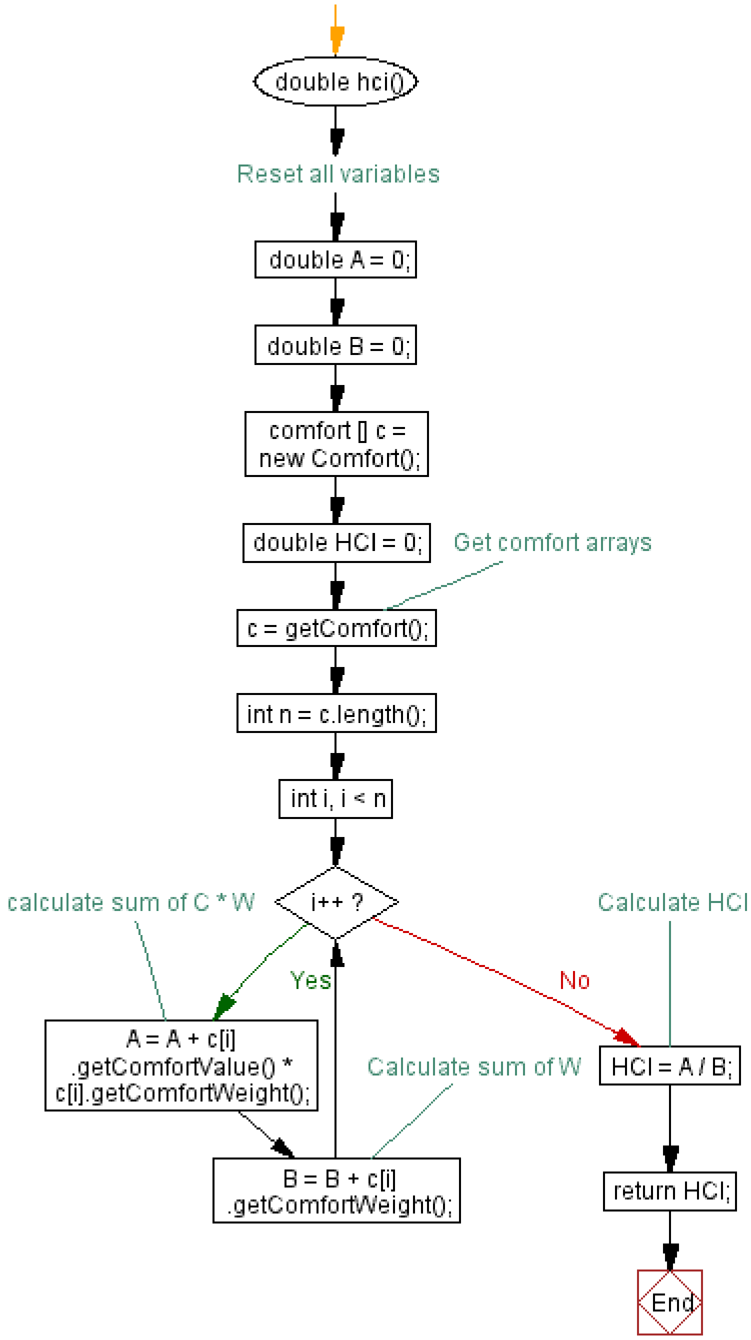
Appendix B IDW Flowchart



Appendix C PMV Flowchart



Appendix D HCI Flowchart



Appendix E HCAmI FRBS Configuration

```
// For Thermal Comfort FRBS
[Var]
pmv
cold -3 -3 -3 -2
cool -3 -2 -2 -1
slightcool -2 -1 -1 0
neutral -1 0 0 1
slightwarm 0 1 1 2
warm 1 2 2 3
hot 2 3 3 3

[Var]
tco
poor 0 0 0 40
good 10 50 50 90
excellent 60 100 100 100

[Rules]
if pmv is cold then tco is poor
if pmv is cool then tco is poor
if pmv is slightcool then tco is good
if pmv is neutral then tco is excellent
if pmv is slightwarm then tco is good
if pmv is warm then tco is poor
if pmv is hot then tco is poor


// For Visual Comfort FRBS
[Var]
light
dark 0 0 250 350
normal 250 350 350 450
```

```
bright 350 450 700 700

[Var]
vco
poor 0 0 0 40
good 10 50 50 90
excellent 60 100 100 100

[Rules]
if light is dark then vco is poor
if light is normal then vco is excellent
if light is bright then vco is good


// For Indoor Air Comfort FRBS
[Var]
co2
good 0 0 0 25
fair 0 25 25 50
poor 25 50 50 100
verypoor 50 100 100 100

[Var]
iaco
poor 0 0 0 40
good 10 50 50 90
excellent 60 100 100 100

[Rules]
if co2 is good then iaco is excellent
if co2 is fair then iaco is good
if co2 is poor then iaco is poor
if co2 is verypoor then iaco is poor
```

```
// For Acoustical Comfort FRBS
[Var]
silent 0 0 0 56
soft 14 70 70 126
loud 84 140 140 140

[Var]
aco
poor 0 0 0 40
good 10 50 50 90
excellent 60 100 100 100

[Rules]
if sound is silent then aco is excellent
if sound is soft then aco is good
if sound is loud then aco is poor
```

Appendix F Source Codes Extracts

Sun SPOT Light Sensor Driver

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package org.sunspotworld;

import com.sun.spot.sensorboard.EDemoBoard;
import com.sun.spot.sensorboard.peripheral.ILightSensor;
import java.io.IOException;

/**
 *
 * @author izani rawi
 */
public class LightSensor {
    public int getLightReading() {
        int lightSensorReading = 0;

        try {
            // get light sensor instance
            ILightSensor ourLightSensor =
                        EDemoBoard.getInstance().getLightSensor();

            // read light value from light sensor
            lightSensorReading =  ourLightSensor.getValue();
        } catch (IOException ex) {
            ex.printStackTrace();
        }

        return lightSensorReading;
    }
}
```

```
package org.sunspotworld;

import com.sun.spot.sensorboard.io.IIOPin;
import com.sun.spot.util.Utils;
import com.sun.squawk.util.MathUtils;
import javax.microedition.io.ConnectionNotFoundException;

/**
 * The SHT15 sensor can sense humidity and the temperature to 12
 * and 14 bits, respectively.  This class is allows the SunSpot to
 * interface with the SHT15.
 *
 * @author greg, modified by izani rawi
 */
public class SHT15 {

    private IIOPin dataPin;
    private IIOPin sckPin;
    private final byte TEMP_CONST = 3;
    private final byte HUMIDITY_CONST = 5;
//  private final byte SOFT_RESET = 20;
//  private final long SLEEP_CONST = 1;
    public static final boolean CELSIUS = true;
    public static final boolean FAHRENHEIT = false;

    private int timeOut = 500;

    /**
     * The SHT15 sensor requires two digital pins, one for data and
     * the other for a clock (marked SCK for the SHT15).
     * @param dataPin the digital pin to use for data.
     * @param sckPin  the digital pin to use for the clock (SCK).
     * @param timeOut Time until command will timeout, in
milliseconds.
     */
    public SHT15(IIOPin dataPin, IIOPin sckPin, int timeOut) {
        this.dataPin = dataPin;
        this.sckPin = sckPin;
        this.timeOut = timeOut;
    }

    /**
     * The SHT15 sensor requires two digital pins, one for data and
     * the other for a clock (marked SCK for the SHT15).
     * @param dataPin the digital pin to use for data.
     * @param sckPin  the digital pin to use for the clock (SCK).
     */
    public SHT15(IIOPin dataPin, IIOPin sckPin) {
        this.dataPin = dataPin;
        this.sckPin = sckPin;
    }

    /**
     * Initialize the sensor, using the wave as specified in the
     * SHT15 datasheet.  Automatically called when getTemperature()
     * or getHumidity() are called.
     */
    private void initialize() {
```

```
        dataPin.setAsOutput(true);
        sckPin.setAsOutput(true);

        /* This isn't written using setDataPin()
         * because the sckpin doesn't always toggle
         * with each change of the data pin in this
         * initialization.
         */

        /**
         * Create the wave for initialization by changing
         * the SCK and DATA pins.
         */
        dataPin.setHigh();
        sckPin.setHigh();

        dataPin.setLow();
        sckPin.setHigh();

        dataPin.setLow();
        sckPin.setLow();

        dataPin.setLow();
        sckPin.setHigh();

        dataPin.setHigh();
        sckPin.setHigh();

        dataPin.setHigh();
        sckPin.setLow();

        dataPin.setLow();
    }

    /**
     * Reset the SHT15 by toggling the clock "9 or more times" with
     * the data pin high (per the datasheet), and then re-
initializing.
     *
     * Shouldn't have to be called unless the SHT15 is unresponsive.
     */
    public void reset() {
        for (int x = 0; x < 10; x++) {
            setDataPin(true);
        }
        sckPin.setLow();

        //Re-initialize the sensor
        initialize();
    }

    /**
     * Send a byte to the SHT via the data pin.
     * @param b The byte to be sent.
     */
    private void sendByte(byte b) {
        for (int x = 0; x < 8; x++) {
            setDataPin((b & 0x80) != 0);
            b <<= 1;
        }
```

```
    }

    /**
     * Set the data pin and then toggle the clock.
     * @param high "true" to send a 1, "false" to send a 0.
     */
    private void setDataPin(boolean high) {
        dataPin.setAsOutput(true);
        sckPin.setAsOutput(true);

        sckPin.setLow();

        dataPin.setHigh(high);

        sckPin.setHigh();
    }

    /**
     * Toggle the clock pin, then read from the data pin.
     * @return If the data pin is high or not after the clock is
toggled.
     */
    private boolean tick() {
        dataPin.setAsOutput(false);
        sckPin.setAsOutput(true);

        sckPin.setLow();

        sckPin.setHigh();

        return dataPin.isHigh();
    }

    /**
     * Send a command to the SHT15, wait for sensor to ACK, then read
the value from the sensor.
     *
     * @param command Command to send to the sensor.
     * @return Value read from the sensor.
     * @throws javax.microedition.io.ConnectionNotFoundException
     */
    private int sendCommandAndRead( byte command ) throws
ConnectionNotFoundException{
        reset();

        sendByte(command);

        getACK();

        int timeElapsed = 0;
        while (dataPin.isHigh() && timeElapsed<timeOut) {
            timeElapsed++;
            Utils.sleep(1);

        }
        if(timeElapsed == timeOut){
            throw new ConnectionNotFoundException();
        }

        int result = readByte();
```

```
        result <<= 8;

        ack();

        result += readByte();

        return result;
    }

    /**
     * Use the SHT15 to find the humidity.  The 12 bit integer will
automatically be
     * converted into a double that represents the relative humidity
(a percentage).
     *
     * @return The relative humidity in a percentage.
     * @throws javax.microedition.io.ConnectionNotFoundException
     *
     */
    public double getHumidity() throws ConnectionNotFoundException{
        int result = sendCommandAndRead(HUMIDITY_CONST);

        double converted = -4.0 + 0.0405 * result + -0.0000028 *
result * result;

        return converted;
    }

    /**
     * Read the temperature from the SHT15.  Actual measurement will
take about 200ms,
     * the method will take a bit longer still, as it still must reset
the SHT15, send the
     * command, and then read the command.
     *
     *
     * @param unit Measure the temperature in Farenheight or Celsius.
     * @return  The temperature, in Farenheight or Celsius, as
specified by the user.
     * @throws javax.microedition.io.ConnectionNotFoundException
     */
    public double getTemperature(boolean unit) throws
ConnectionNotFoundException {
        int result = sendCommandAndRead(TEMP_CONST);

        double converted;
        if (unit == FAHRENHEIT) {
            converted = (result * .018 - 39.5);
        } else {
            converted = (result * .01 - 39.6);
        }
        return converted;
    }

    /**
     * Typically called after a byte is read in to acknowledge that
     * the byte has been read.
     */
    private void ack() {
        dataPin.setAsOutput(true);
```

```
        sckPin.setLow();

        dataPin.setLow();

        sckPin.setHigh();

        sckPin.setLow();

        //Set the dataPin high so that it can be treated as an input
pin due to a bug in the SDK.

        dataPin.setHigh();

        dataPin.setAsOutput(false);
    }

    /**
     * Calculate the dewpoint given temperature and humidity.
     * @param unit Temperature in Farenheight or Celsius
     * @param The temperature Temperature to use for the dewpoint
calculation
     * @param The humidity Humidity to use for the dewpoint
calculation
     * @return The dewpoint in the units given.
     */
    public double getDewpoint(boolean unit, double temperature, double
humidity) {
        if (unit == FAHRENHEIT) {
            temperature -= 32;
            temperature /= 1.8;
        }

        double m = 0;
        double tn = 0;

        if (temperature >= 0) {
            m = 17.62;
            tn = 243.12;
        } else {
            m = 22.46;
            tn = 272.62;
        }

        double dewPoint = tn * (MathUtils.log(humidity / 100) + (m *
temperature) / (tn + temperature)) / (m - MathUtils.log(humidity /
100) - (m * temperature) / (tn + temperature));

        if (unit == FAHRENHEIT) {
            dewPoint *= 1.8;
            dewPoint += 32;
        }
        return dewPoint;
    }

    /**
     * Read a byte from the data pin (8 bits).
     * @return The byte read from the data pin.
     */

    private int readByte() {
```

```
        dataPin.setAsOutput(false);

        int toReturn = 0;

        for (byte x = 8; x > 0; x--) {

            int result = (tick() ? 1 : 0);

            toReturn *= 2;
            toReturn += result;
        }
        return toReturn;
    }

    /**
     * Read in bits from the data pin.
     * @param bits The number of bits to read in.
     * @return The value of the bits that have been read in.
     */
    private int readBits(int bits) {
        dataPin.setAsOutput(false);

        int toReturn = 0;

        for (int x = bits; x >
                0; x--) {

            int result = (tick() ? 1 : 0);

            toReturn *= 2;
            toReturn += result;
        }
        return toReturn;
    }



    /**
     * Get an ACK from the SHT15 by waiting for specified
milliseconds.
     */
    private void getACK() throws ConnectionNotFoundException {
        dataPin.setAsOutput(false);

        sckPin.setLow();

        sckPin.setHigh();

        //Pin should be pulled low, then high.
        int timeElapsed = 0;
        while (dataPin.isHigh() && timeElapsed < timeOut) {
            Utils.sleep(1);
            timeElapsed++;
        }

        if(timeElapsed==timeOut){
            throw new ConnectionNotFoundException();
        }

        sckPin.setLow();
```

```
        timeElapsed = 0;

        while (dataPin.isLow() && timeElapsed < timeOut) {
            Utils.sleep(1);
            timeElapsed++;
        }

        if(timeElapsed==timeOut){
            throw new ConnectionNotFoundException();
        }
    }
}
```

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

package org.sunspotworld;

import com.sun.spot.sensorboard.EDemoBoard;
import com.sun.spot.sensorboard.io.IScalarInput;
import java.io.IOException;

/**
 *
 * @author izani rawi
 */
public class HS2000V {
    private IScalarInput temp;
    private IScalarInput rh;
    private IScalarInput vref;

        /**
    *
    * @param pin Pin number for analog reading of Temperature
    * @param vref IScalarInput pin for voltage reference
    */
    public HS2000V (int tempPin, int rhPin, int vrefPin) {
        this.temp =
EDemoBoard.getInstance().getScalarInputs()[tempPin];
        this.rh = EDemoBoard.getInstance().getScalarInputs()[rhPin];
        this.vref =
EDemoBoard.getInstance().getScalarInputs()[vrefPin];
    }

    /**
    *
    * @return Double value of degrees C
    * @throws java.io.IOException
    */
    public double getTemp() throws IOException {
        //return (((double) temp.getValue() - 115) / (double)
vref.getValue()) * 100;

        double vTemp = (double) temp.getValue();
        double vvref = (double) vref.getValue();
        double temperature = (vTemp / vvref * 130 ) - 30;
        System.out.println("vTemp = " + vTemp + " vvref = " + vvref);
        return temperature;
    }

    /**
    *
    * @return double-precision relative humidity as calculated by:
    * rh pin value / vref pin value * 100
    * @throws java.io.IOException
    */
    public double getRH() throws IOException{
        //return ((double)this.rh.getValue() /
(double)vref.getValue()) * 100;
```

```
        double vrh = (double) rh.getValue();
        double vvref = (double) vref.getValue();
        double relativehumidity = vrh / vvref * 100;
        System.out.println("vrh = " + vrh + " vvref = " + vvref);
        return relativehumidity;
    }
}
```