Intelligent Decision Making Based on Fuzzy Logic System in Remote Wireless Communication

Pouya Bolourchi

Submitted to the
Institute of Graduate Studies and Research
In partial fulfillment of the requirements for the Degree of

Master of Science
in
Electrical and Electronic Engineering

Eastern Mediterranean University
January 2012
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Elvan Yılmaz
Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Master of Science in Electrical and Electronic Engineering.

Assoc. Prof. Dr. Aykut Hocanın
Chair, Department of Electrical and Electronic

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Electrical and Electronic Engineering.

Prof. Dr. Şener Uysal
Supervisor

Examining Committee

1. Prof. Dr. Osman Kükrer
2. Prof. Dr. Şener Uysal
3. Assoc. Prof. Dr. Hasan Demirel

ABSTRACT

Wireless sensor networks (WSNs) consist of a large number of sensor nodes. The sensors are tiny devices, which are easy to manufacture, low cost and very power efficient. The major objective of this thesis is to use WSNs in intelligent decision making based on the collected data. Intelligent decision making has important application especially in autonomous systems used in homeland security, health care improvement, wildlife monitoring, environmental surveillance, climate research and natural disaster – crises management. The main advantage and growing significance of intelligent decision making is the elimination of human factor which makes it reliable, conformable, adoptable and a major player in energy management of remotely located autonomous systems.

The focus of this thesis is to design a system to make an intelligent decision based on the five levels of sensitivities introduced by Uysal et al [2]. In designing such a system, we will consider some parameters like power consumption, total cost and efficiency of the system in comparison to the PLCs developed by companies such as Siemens and Mitsubishi. For software implementation among the available techniques, we have chosen one that has a faster processing method in comparison to computational methods that are widely used in the processors. We introduce Fuzzy logic system, which is a very powerful method, commonly used in control systems, and can be easily simulated in MATLAB Toolbox.

Keywords: Intelligent decision making, wireless sensor network, fuzzy logic, security system, PLC component

ÖZ

Kablosuz Sensör Ağları çok sayıdaki sensör düğümlerinden oluşmaktadır. Sensörler, üretilmesi kolay, üretim maliyeti düşük ve yüksek güç performansına sahip cihazlardır. Bu tez çalışmasının temel amacı, Kablosuz Sensör Ağları'nın toplanmış olan verilere dayalı akıllı karar verme sistemlerinde kullanılmalarından ibarettir. Akıllı karar verme sistemleri, özellikle ülke güvenliği, sağlık sistemlerinin geliştirilmesi, vahşi hayatın izlenmesi, çevre gözetimi, iklim araştırmaları ve doğal felaket-krizlerin yönetiminde kullanılan özerk sistemlerde olmak üzere önemli uygulama alanlarına sahiptir. Akıllı karar verme sistemlerinin asıl avantajları ve büyümekte olan önemi, bu sistemleri güvenilir, uygun, uyarlanabilir ve uzakta yerleştirilen özerk sistemlerin enerji yönetiminde önemli bir rol üstlenen sistemler haline gelmesine neden olan insan faktörünün ortdan kaldırılmış olmasıdır.

Bu tez çalışması, Uysal ve diğerleri [2] tarafından tanımlanan beş duyarlılık düzeyine dayanan akıllı bir kararın alınmasını mümkün kılacak bir sistemin tasarlanması üzerinde yoğunlaşmakatdır. Böyle bir sistemin tasarlanması sırasında, Siemens veya Mitsubishi gibi firmalar tarafından geliştirilen PLC sistemleri ile karşılaştırmalı olarak söz konusu olan sistemin güç tüketimi, toplam maliyeti ve verimi gibi parametreler incelenmektedir. Yazılım uygulaması konusunda ise mevcut olan teknikler arasından, yaygın bir şekilde işlemcilerde kullanılmakta olan hesaplamalı yöntemlere nazaran daha yüksek işleme hızına sahip olan bir teknik seçilmiştir. Bu çalışmada oldukça güçlü bir yöntem olup genellike kontrol sistemlerinde kullanılan ve MATLAB Toolbox'dan yararlanılarak kolaylıkla simülasyonu yapılabilen Bulanık Mantık sistemi tanıtılmıştır.

Anahtar Kelimeler: Akıllı karar verme, kablosuz sensör ağları, bulanık mantık, güvenlik sistemi, PLC bileşeni

To My Dear Parents and My Lovely Sister

ACKNOWLEDGMENT

I would like to thank my supervisor, Prof. Dr. Şener UYSAL, for his encouragement and support during my master degree's period. I gratefully acknowledge the invaluable guidance and advise he has provided to me throughout this process. I really appreciate the opportunities he has given me and cannot say enough about my gratitude to him.

Special thanks also go to all my friends and especially my dear friend Afshin Jooshesh for sharing the literature and providing invaluable assistance.

I would like to express my deepest gratitude to my lovely family; they gave me a chance for completing my higher education in Cyprus. Without their support, both in financial and emotional matter, achievement of this level was impossible.

Last but not least I would like to express my deepest appreciation to my Love who has always been the source of my motivation. Without her patience I would not have been able to complete my master degree.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS/ABBREVIATIONS

ADC	Analog-to-Digital Converter
AI	Artificial Intelligence
AWR	Automatic Workload Repository
BC	Base Station
BIST	Built-In Self Test
BIT	Built In Test
CO	Carbon Monoxide
CPU	Central Processing Unit
CS	Computational System
DAC	Digital-to-Analog Converter
DM	Decision Maker
FLS	Fuzzy Logic System
GA	Genetic Algorithm
GSM	Global System for Mobile Communications
IDMS	Intelligent Decision Making System
IHSSS	Integrated Homeland Security Surveillance System
LCD	Liquid Crystal Display

MCU	Microcontroller
NN	Neural Network
PCB	Printed Circuit Board
PLC	Programmable Logic Controller
PPM	Parts Per Million
RAM	Random Access Memory
ROM	Read Only Memory
RTC	Real Time Clock
SAPS	Stand Alone Power System
TTL	Transistor-Transistor Logic
USB	Universal Serial Bus
WSN	Wireless Sensor Network

Chapter 1

INTRODUCTION

1.1 General Introduction

Intelligent decision making (IDM) is one of the most important issues in wireless communication systems during the past decade. By designing such a system, primarily we diminish human intervention factor and as a result, we overcome man-made errors thereby increasing the reliability of the system. Furthermore, significant power saving can be achieved which is crucial for remotely located surveillance systems where the available power can be very limited. This is especially true for systems located at rural cross-borders, systems deployed for transnational gas pipeline security and systems used for visual surveillance in mountainous terrain for natural hazard monitoring such as flooding and forest fires by deploying WSNs [1] on the other hand, it is possible to significantly increase the functional capabilities of the system.

Employing IDM architecture in remote wireless communication systems can widely increase the application areas. They can be used in industrial control and monitoring, home automation and consumer electronics, security and military sensing, asset tracking and supply chain management, intelligent agriculture and environmental sensing and health monitoring [21].

1.2 Definition of the Problem

This thesis is a part of the massive project called Integrated Homeland Security Surveillance System (IHSSS) being carried out by a research group under the supervision of Prof. Dr. Sener Uysal [2].

This project mainly consists of processors, camera(s) and batteries rechargeable by solar energy. The main idea in IHSSS is the study on a solar-powered system composed of modules in order to develop a multipurpose surveillance system for increased situation awareness and enhanced crisis management in cases of terrorism, natural hazards and human trafficking. In this thesis, we define five levels of sensitivities (very low, low, medium, high and very high) for each of terrorism threat, human trafficking, natural hazards and connectivity. Based on these levels the system will make the intelligent decision. Major investigation will be at the root of the potential threat but the very dynamic climatic/natural phenomena and the unpredictable human-terrorism actions also necessitate the need for enhanced crisis management. That is, fire, flooding and suspicious activities which may lead to malicious actions (e.g., close to flight paths, water supplies, around gas pipelines, close to railways, power plants, close to or around physical plants for telecommunication networks, cross-border – land and sea) will be linked to five levels of sensitivities and will be investigated based on possible threats. Figure 1.1 illustrates the natural hazards, terrorism and human trafficking and their levels we must take into consideration.

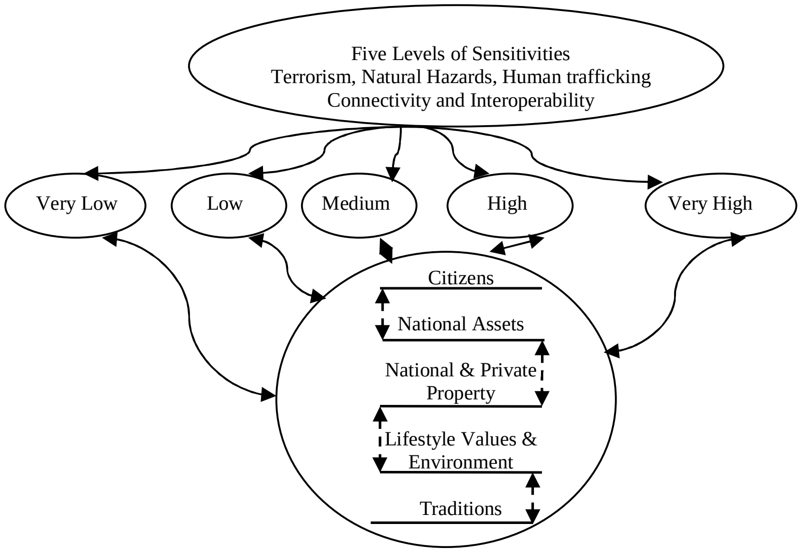

Figure 1.1 Illustration of the Targeted Sensitivity Levels [2]

It will then activate a series of pre-defined actions. The actions may be autonomous (desired by our proposed architecture) based on the threat, cause, targeted area, and location; multiple actions are inevitable in some cases.

1.3 Definition of Terms

1.3.1 Sensor Node Sensor node is a node in a WSN that consists of a number of different sensors that has a capacity for detecting, processing, and communicating with other nodes in the network. The sensor node generally consists of a microcontroller, transmitter, receiver, external memory, power source and a bunch of sensors. The sample structure of the sensor node is given in Figure 1.2.

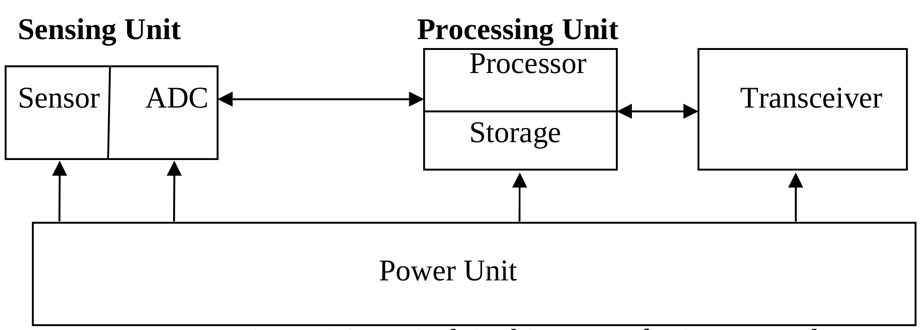

Figure 1.2 Typical Architecture of a Sensor Node

1.3.2 Wireless Sensor Network A wireless sensor network is constituted by a large amount of sensor nodes, which are densely deployed either inside the phenomenon or very close to it [3].

1.3.3 Data aggregation Meaningful summary of the given data, which forwards to sink node, is called data aggregation.

1.3.4 Data fusion This is a technique of combining data from multiple sources and collecting information in order to get results in more efficient and accurate way.

1.3.5 Sink Node A sink node or a base station is a node that collects and controls data gathered by cluster heads. In clustering algorithm, each cluster consists of different nodes and one of them is selected as a cluster head.

1.3.6 Data Routing This is a process of selecting the path in order to transfer the collected information to the base station.

1.4 Structure of the WSN

We deployed sensor nodes and microprocessors on each station. In order to save power consumption and depending on the application (details are given in Chapter 5), microprocessors operating modes can be changed during specific times. These time- frames contain the modes such as operating, sleeping, suspending, standby, and idle mode. During the operating mode, a microprocessor receives a beacon message from the base station and in turn, it sends respective beacon messages to all the sensor nodes in that station. Sensor nodes regularly report the information they capture to a microprocessor (cluster head) for data analysis and corresponding actions, if needed. The microprocessor compares the current information with the previous outcome. If there is no change or if the changes are not very critical, the microprocessor makes a very simple task decision. That is, if the level of sensitivity is low or very low, the microprocessor sends messages to actuators in order to perform specific action. On the other hand, if the changes are critical, that is, if the level of sensitivity is medium, high or very high, microprocessor will send the data to the servers for making an appropriate decision. The details of WSN structure are given in Chapter 3. The sample architecture of the WSN is given in Figure 1.3. In order for the end user to supervise the parameters on a remote module through the interface, it is crucial to introduce a protocol. The reason of utilizing RS485 port is to select the desired sensors in the network that require to be employed in a specific application and can easily be modified according to the application. For instance, consider the situation in Amazon region where the danger of a large-scale forest fire is high but according to other statistics on the other hand, there is no danger of earthquakes.

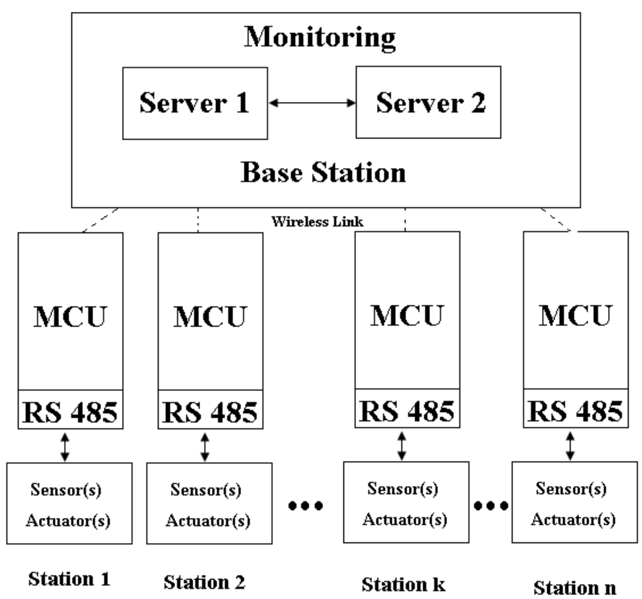

Figure 1.3 Structures of Sensors and Microprocessor in the WSN

Therefore, the regional application in the Amazon region is offered only to detect the fire danger; for this we deploy light sensors, temperature sensors, smoke, humidity, CO sensors and range sensors in order to detect and diagnose the level of sensitivity of fire; MCU or base station then makes a respective intelligent decision based on the received data. On the other hand, think about Honshu in Japan where there is a continuous danger of earthquakes off the coast . We are sure that in this area, there is no threat of fire or other natural hazards and we are interested to detect threats of earthquakes only. Gas sensors, sound-audio sensors, motion sensors and pressure sensors will be the type of sensors that we choose to deploy to detect the level of sensitivity. For accuracy - which defines the duplication of crucial elements of a system with the purpose of increasing reliability of the system, in the case of back up - we set out two servers. If one of the servers is out of order, the second one automatically takes over.

1.5 Intelligent Decision Making Process

In summary, as an overview of the steps of intelligent decision making, we define our step process for these basic areas: Sensing field, cyberspace and command control center. For sensing field, we deploy a large number of sensor nodes based on the application. The most important factor when deploying a large number of sensors is the tradeoff between the cost and coverage quality level. Then, data routing structure may be used to transfer the gathered data to a base station. For cyberspace, based on the computer networks, the large amount of sensor fusion is required for collecting individual sensor reading at some central sink nodes. A fault-tolerant data fusion algorithm combines the computed local outcome from the data gathered by individual sensors in order to achieve optimal solutions for getting final decision-making. The final step is the command control center, which makes intelligent decisions based on the results from sensor data processing and information fusion [5]. Figure 1.4 gives a summary of the steps we mentioned in this section.

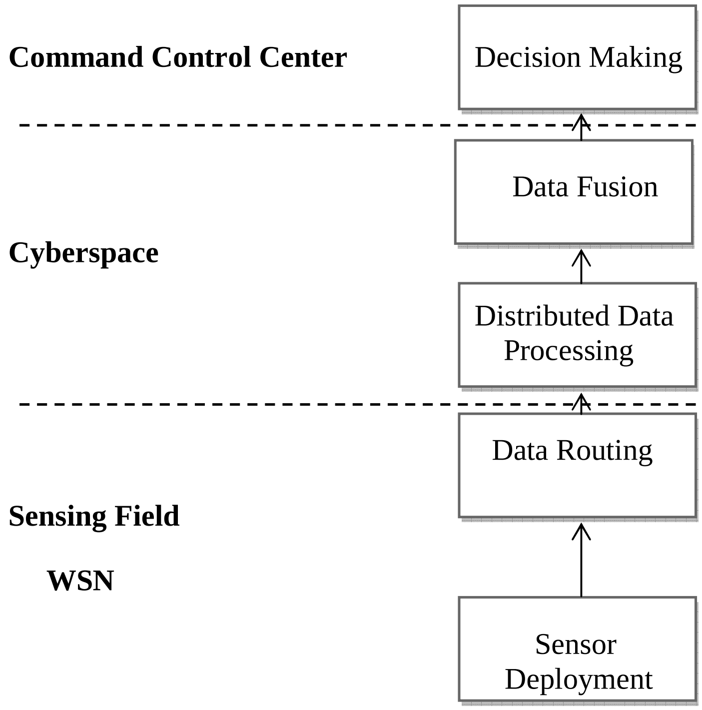

Figure 1.4 IDM Integrated Sensing Steps Components & Design Process[5]

1.6 Overview

In this chapter, we introduce the basic introduction to IDM. We had an overview to design an intelligent system and we defined the basic terms. In Chapter two, we have an overview of architectures of a WSN. We will see some factors that have tremendous effect in designing WSNs. Fault tolerance, scalability, production cost, power consumption, security and network topology are among some of them. Since our basic aim is to make a proper decision we want to ensure that sensors in the network work properly and therefore it is vital to find sensor failures. Third chapter mainly discusses the design of an intelligent system. The chapter introduces a prototype design of the problem, then a comparison between the given prototype and available PLC (programmable logic controller) in the market, is made. The main design and its advantages and disadvantages will be also studied. In Chapter four, we introduce a methodology that we will use in order to make an intelligent decision. We introduce neural network (NN), genetic algorithm (GA) and fuzzy logic system (FLS) and each method followed by an example explaining the major advantages and disadvantages. A case study is given in Chapter five. The implementation and outcome are also given in this chapter. We consider two different applications. Our aim is to increase the awareness of natural hazards in the presented applications. The final chapter consists of the conclusion and further work.

Chapter 2

ARCHITECTURE OF WIRELESS SENSOR NETWORK

2.1 Quick Overview

In this chapter, we will study architectures for a WSN. Many factors play crucial roles in designing the architecture of a WSN. The most significant factors can be named as follows: fault tolerance, scalability, production cost, power consumption, security, routing, reliability, mobility, real time, and network topology. Therefore, in this chapter we will have an overview of these factors that have tremendous effects on the design [6][22].

Generally, we deploy different sensors in each node. In case of fire, we deploy temperature, smoke, light, distance, humidity and Carbon Monoxide (CO) Sensors.

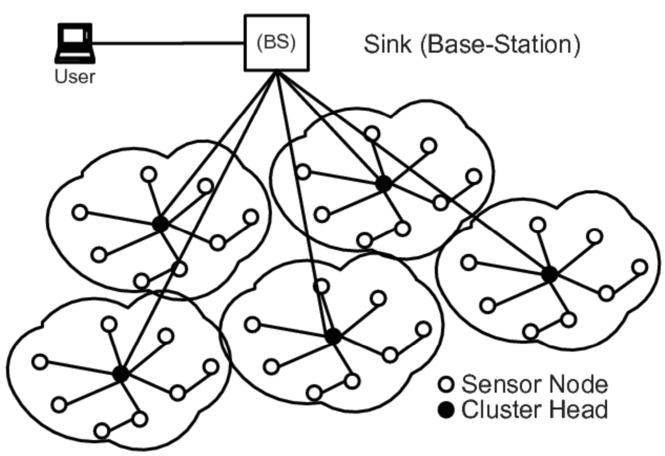

Figure 2.1 Sensor Network Architecture [31]

In this structure, the network is divided into different parts. Each part is called a cluster. Each cluster also consists of nodes and each node is composed of different sensors. In each cluster there is one node called the cluster head. Sensor node collects the environmental information and transmits it to the cluster head. Cluster head aggregates and forwards just the meaningful data to the sink such as base station (BS). The sink collects the data from and transmits the data to the user via antenna.

2.2 Fault Tolerant Data Acquisition

As we mentioned in previous chapter our aim is to design an intelligent system, which makes the decision. For designing such a system we deployed wireless sensors and therefore study of sensors becomes one of the important parts of the thesis. If some sensors are faulty it may causes uncertain decision-makings, therefore detection of faulty sensors in the network is vital. We will design a fault tolerant method in WSN. There are some techniques like Kalman filtering, particle filtering, and wavelet transforms [7][8], however these methods are not a good choice since they have exhaustive computation while because of low computation of the sensor node; the chosen method should have a very low computation overhead.

We introduce Built-in Self Test (BIST) method, which commonly used to capture hard faults that happen rapidly [9].

2.2.1 Built-in Self Test

A built-in self-test is a range test method. In this method, we define two windows as usable threshold window and guaranteed windows. The guaranteed window is a specific range that we are sure in this range every sensors work properly. The range of this window is usually defined by manufacturer. On the other hand usable threshold window is a range in which we assume each sensor within a node will operate. A Built-in Test (BIT) is said to have passed if the sensor reading is within the threshold window. The minimum range of threshold windows is less than the minimum range of the guaranteed windows and the maximum range of threshold window is higher than the maximum of guaranteed windows. We should consider three different situations. First situation if the sensor is in the guaranteed windows that as we mentioned it will work correctly. The second situation is when a sensor is out of the range of threshold windows. In that case, the sensor said to be faulty since its reading exceeds the threshold limits (minimum and maximum values specified). The last situation is when the sensors are in the range of threshold window. In a case the sensor said to be working but with a much lesser accuracy. Defining the threshold window enable us to trade off the performance of each sensor against the performance of the sensor network. We should define the limitation of threshold window precisely. If the limitations get high range, we will achieve the high false alarm. On the other hand, if the limitations have a very low range we will obtain a very low fault. For better understanding, we can consider the normal distribution in which the weight will diminish rapidly as the reading sensor deviate from guaranteed window. For those reading sensors are located outside a guaranteed window but they still be within the usable threshold window- by considering the following function- can be weighted accordingly:

$$w_{ij} = e^{-(\frac{r_{ij}}{\varepsilon})^2} \tag{2.1}$$

$$r_{ij} = w_{ij} * r_{ij} \tag{2.2}$$

Where r in the above equations is sensor reading, w indicates weighting scale, i refers to sensor node number, j signifies the sensors on-board the i-th sensor node and $\mathcal{E}$ is the number that is selected in order to replace the weighting scale between zero and one. This equation assists us to get less attention for sensor whenever the reading deviates from guaranteed windows and it reaches to usable threshold window.

Another critical issue in this range test is a problem of detecting faulty sensor in the usable threshold window. We use usable threshold to find out the faulty sensor in the network. This procedure is not a simple task, because if the sensor is in usable threshold window it may have deal with two different situations. Either the sensor is faulty or it captures the environment’s data. For example, in the case of fire in the temperature sensor reads 150 ˚C and we define our guaranteed window for temperature between -10 ˚C to 110 ˚C. 150˚C indicates that the sensor reading is outside the guaranteed window and that means either this reading detects the fire in the environment or that temperature sensor is faulty. Therefore, in order to overcome this problem we deployed sensor redundant nodes to capture the same occurrence in the application. Suppose we have n sensors, each sensor has n-1 neighbor, which is spatially correlated. Each sensor should read the same value as other neighbor sensors have read. This helps us to find the faulty sensor(s) easily and cross them out.

Although this method is very helpful to detect the faulty sensor but we should consider the main drawback when the sensor reading is faulty while it is in the guaranteed window. Suppose the sensor reads 25˚C instead of 70˚C, although both ranges are still in the guaranteed windows but you can observe an error occurs in reading.

2.3 Scalability

The ability of the system to be varied with altering the load on the system will be defined as scalability. Many aspects will affect the capacity of the system to scale. It is vital for designing the scalability of the any system a consideration of merging the solutions to these aspects. "A system is said to be scalable if its availability and fault tolerance are within the acceptable threshold when the load or subsystem is increased in the system." [10]. On the other hand, if the system is scalable, its performance and its availability will increases. By availability, we mean that there is no failure in the structure. The scalability usually refers to the tasks that work together in order to reach the same goal. If some systems work in tandem their tasks have more efficient rather than if those system work separately. Therefore, one of the crucial aspects for designing the scalability factor is the ability of the structure to add more system without need to re-engineer it. Commonly there are two methods of adding more system. Scale horizontally or simply scale out refers to method by adding more nodes. The advantage of this method is by deploying more nodes in the system we can coverage a vast area of interest; however, this method leads to growing the faulty sensor. Generally increasing the sensor nodes will accompanied with decrease in the performance of the overall system. We define the scale vertically or scale up by adding more resources such as processor or memory to a single node in the system. For constructing scalable design, it should be noted that base station should have more power, memory and computation ability than sensor nodes. This feature alleviates the system to work faster which means more scalable and decreases the cost due to indicating the faulty sensor quickly. The structure of the scalable design is shown in the Figure 2.2.

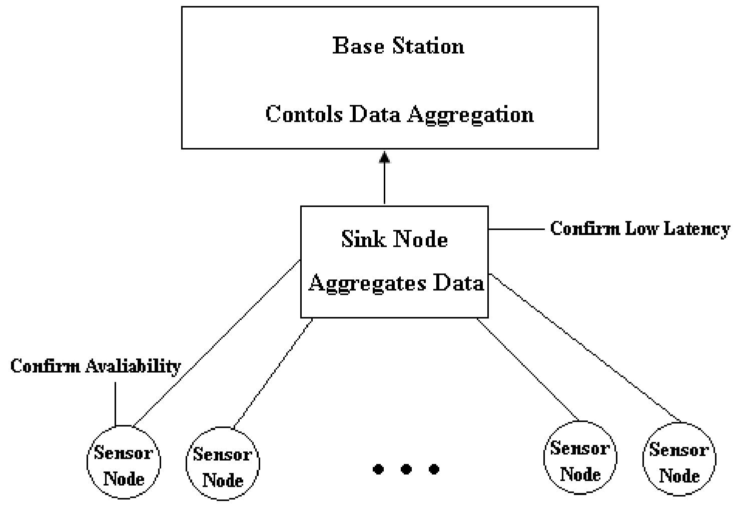

Figure 2.2 Scalable Design Schematic

2.4 Power Consumption

One of the most important factors for designing WSN is power consumption. Actually, this power factor discriminates designing between wireless sensor networks with other wireless networks. Our fundamental focus is to design a system such that it makes the power as minimum as possible. This brings us to following advantages. Firstly, the lifetime of the network will increase and secondary this system reduces the power and it means the costs will also decreases. The lifetime of sensor is important issue since the sensor deployed in large number and disregarded this make the process difficult to change or recharge the battery [11]. Some models widely used for designing of low power consumption. The Heinzelman [15] is usually used for designing the communication protocols. Measuring the average using ampere meter current for evaluating power consumption is a method introduced by Anastasi [30]. Another method of indicating power consumption is using the oscilloscope. It should be noted that each method is valid in the specific time interval and therefore we cannot generalize them [4].

2.5 Network Topology

There are different network topologies that can be employed to design the WSN. We introduce five of widely used topologies as fully connected network, bus network, star network, mesh network and hybrid star-mesh network. We will discuss each topology and its strength and weakness.

The Star network is the simplest communication topology. In this topology, there is only a single base station and it can send and receive data to different number of remote nodes. It means that these remote nodes can only send signals to the base station and therefore they are unable to send signal between each other. The advantage of this topology is that the nodes will consume a minimum power beside its simplicity. The disadvantage is that base station should be in the scope of all nodes [12].

The mesh network on the other hand permits other nodes in the network to communicate with each other. This topology is useful since if two nodes are out of range from each other, then it is possible that a first node sends a message via an intermediate node in order to forward the message to the desired node. The advantage of this topology is that if a node fails then another is available to forward any signal. It should also be considered that performance of the system is covering longer range. This is important fact since connections between nodes is usually unlimited. The disadvantage of this network is that it uses a lot of power and this follows by increasing in costs [12].

The hybrid star-mesh network is a more flexible topology to keep the power of WSN in a minimum state. Nodes with the lowest power are powerless to forward messages; therefore, minimal power is maintained. Nevertheless, nodes with more power in this network are able to forward these messages from those of low power [12].

The main drawback of fully connected networks is when additional nodes are added; the number of links expanded exponentially. Thus, for large networks, the routing problem is faulty even with the availability of large amounts of computing power [23].

In the bus topology, messages are broadcasted on the bus to all nodes. Each node investigates the station (destination address) in the message header, and processes the messages addressed to it. Bus topology is easy to implement and less expensive than other topologies. The main disadvantage of bus topology is the topology is not responsible for retransmitting any messages [23]. In addition, the rate of data transfer is slower than other topologies.

2.6 Security

This factor is another important fact that we should take an account for designing such a system since in most of the applications we deal with the security of the system. For detecting threats by environment sensors deployed very close to event or inside the event of desire application. This may increase the tendency of sensor to be corrupted, and it will leads to failure in decision-making.

2.7 Production Cost

As we mentioned earlier in chapter one sensors are cheap and therefore they widely used for manufacturing. Wireless sensor network on the other hand are developed since they are in the very small size, low power consumption and as a result very low cost. For designing the WSN if we consider all the factors mentioned in this chapter like power consumption, faulty and scalability, we will decrease the cost drastically.

2.8 Routing

This process involves selecting the data transmission path in network and could be played a role for data transmitting. Routing is performed for multiple networks like the telephone network, the Internet and transfer. Routing can be reason that packets sent from source to destination. Hardware devices are known as routers; bridges, port, firewall and switch. Computers that have network-cards can. This process is sending packages according to the routing tables and it can keep the routers in the destination. Therefore, constructing routing tables is very important for efficient routing. Most routing algorithms use only one network path at a time, but multipath routing techniques enable the use of multiple alternative paths.

2.9 Real-Time

In some applications, it is vital that messages arrive at a station by a specific time. Since the level of certainly in WSN is low, it is complicated to develop routing algorithms with any guarantees. Some protocols utilize an idea of velocity to prioritize packet transmissions; since it plays a role of combining the deadline and distance that a message must travel.

2.10 Mobility

Routing is become very hard task if either the message source or station (destination) or both are moving. The way for overcoming this problem involves consecutively updating neighborhood tables or identifying proxy nodes, which are responsible for keeping track of where nodes are. “Proxy nodes for a given node may also change as a node moves further and further away from its original location [22].”

2.11 Reliability

Because messages travel multiple hops, it is necessary to have a high reliability on each link; otherwise, the probability of a message transiting the entire network would be undesirability very low. In order to recognize the reliable links some studies like as received signal strength and packet delivery ratio has been done. Critical empirical evidence indicates that packet delivery ratio is the best metric, although it can be expensive for assembling. It also shows that many links in a WSN are asymmetric, that means if node X can successfully transmit a message to node Y, the node Y may fail to retransmit a message to node X [22].

Chapter 3

DESIGNING THE SYSTEM

3.1 Chapter Preview

In this chapter, we introduce intelligent decision making architecture in remote wireless communication systems by representing a prototype of the system. Our studies follow a comparison of existing products with the prototype introduced in this chapter.

3.2 Prototype

The first step in designing an intelligent decision making system is to introduce a prototype with as many parameters as possible that are considered to be the major players/actors in the system. In the preparation of this prototype, we must gather all the possible factors/elements which can be realized with a reasonable cost and good quality. We start with a component that we assume to get the maximum benefit by employing them in the design and we then measure their power consumption. We compare our design with the existing ones and if the comparison is not acceptable then we introduce modifications for these parts. The major parts of our design include sensors, convertors, microcontroller and transceiver. Figure 3.1 illustrates the prototype of the system. Each part is described in detail in the following pages.

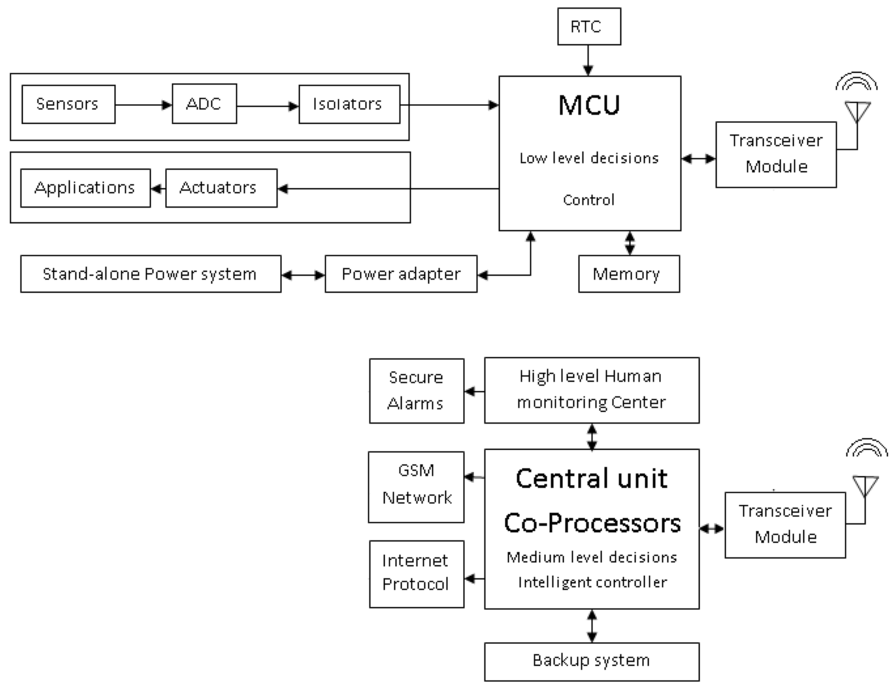

Figure 3.1 Intelligent Hierarchy Decision Making and Control System

3.2.1 Sensors: As we have discussed earlier in pervious chapters our aim is to design a system, which is capable of detecting terrorism threat, human trafficking and natural hazards. We may deploy one thousand sensors in each application for detecting and controlling a specific task. As you will see in Chapter 5, as a case study we are interested in controlling the threat of fire in a wide forest area. In this case, we utilize temperature sensors, light sensors, gas sensors and humidity sensors. All these sensors are analog and their voltages are in the range of zero and 5V.

3.2.2 Analog-to-Digital Converter (ADC): All the inputs we need before the data fed to the microcontroller for processing should be in digital format and since all the sensors are analog, we assign an ADC to convert analog signals to digital.

3.2.3 Isolators: Generally, it is called opto-isolator or optocoupler. This device is designed for transferring electrical signals to supply coupling with electrical isolation between its input and output. The function of an isolator in the system is to prevent high voltages or rapid voltage changes on one side of the circuit from damaging components or distorting transmissions on the other side.

3.2.4 Real Time Clock (RTC): This plays a crucial task in designing the system. In fact, as input is fed to the microcontroller, RTC determines the current time of data. Since one of the most important functions of our design is controlling, the time difference lets us to observe and measure changes that have occurred in the respective time reference. The time values represent the year, month, date, hours, minutes, and seconds. The device itself is a microchip powered by a battery. It also has a small memory that keeps current time values stored by the RTC.

3.2.5 Microcontroller (MCU): The heart of the design is MCU. A MCU typically consists of Random Access Memory (RAM), Read Only Memory (ROM), timers, input and output ports (I / O) and sequential port (Serial Port serial port) in the their own chips, and can impose controls on other devices. MCU processes input data. As discussed earlier for each threat we define five levels of sensitivity. While MCU controls data, it also can perform some of the very simple decision for low-level sensitivity, i.e., in the case of fire, if the threat is very low or low, the MCU performs a simple function, such as giving an alarm or activating the fire extinguisher. For higher levels of sensitivity, we utilize co-processors.

3.2.6 Memory: Memory saves all the information processed by the MCU. This feature helps us to access all the events, if required. In case of fire, if there is a drastic change in the input, we can find out the cause.

3.2.7 Transceiver: MCU controls the system and performs some simple tasks for low-level threats, but as the level of terrorism threat, human trafficking, natural hazards and connectivity are increased, we should transfer the data to the monitoring station for further processing. Since the system may be located at rural areas like a mountainous region, near a cross-border, etc. we do not have a direct visual contact with the system. The monitoring system may be located 100 of kilometers away from the surveillance system, which means that we need a wireless system near the system and one at the monitoring station. We introduce a transceiver, a device that consists of a transmitter and a receiver integrated together. A transceiver can be two completely separate circuits or can have some common parts including the antenna and modulator. A transceiver sends and receives the information from MCU to the station.

3.2.8 Co-Processors: Ambient data gathered from the sensors are fed to the co-processors. Based on the level of sensitivity, information may be monitored by a high-level human monitoring center, GSM Network, or an internet protocol. For example, in case of fire if the level of sensitivity is medium GSM Network sends an automatic message to the nearest firefighting station. In some cases, an internet protocol can be used for sending alarm signal to the pre-specified corresponding authorities. For high and very high level of sensitivities, we prefer human interception. In any case, all possible solutions are considered to opt for the best action to overcome the crises.

3.2.9 Digital-to-Analog Converter (DAC): Since the input of the applications deal with analog signals, we put a DAC to convert back all data to original format. Again, we place an isolator to be sure that the high voltage or short circuit in the system does not damage the circuit.

3.2.10 Actuator: This device is inserted into the system for final controlling and making it available for applications. In fact, this part is for moving or controlling a mechanism or a subsystem. For example, in case of fire this device enables us to release the fire extinguisher.

3.2.11 Stand Alone Power System (SAPS): The system is intended to be used in areas where maintenance cycle may be long; also, we do not want the system become vulnerable to planned or irresponsible attacks when choosing its location. Therefore introducing a very reliable device, which can provide self-sufficient electricity for a very long time, becomes a must. Solar power system is a good example of SAPS.

3.3 Benefits of the Proposed System

The proposed module is comparable with the programmable logic controllers (PLC) which are available in the industry. The most common PLCs are manufactured by Siemens, Omron, Mitsubishi and Moeller. The main difference between the proposed module and the existing products is the method we use in the system. We have used Fuzzy Logic System (FLS) whereas those products use computational systems (CS). CS processes higher size of data and therefore it would take more time for processing. FLS on the other hand has a higher speed of processing which makes it suitable for mass production in the applications that do not require very precise computation. It is an ideal device for IDM purposes. Data sheets of some PLC's have inspired our design. Table 3.1 shows us the difference between Mitsubishi's PLC and our proposed design.

Table 3.1 Comparison of the Proposed Design with Mitsubishi's PLC.

Type	Number of Inputs	Number of Outputs	Power Supply	User Memory	Digital outputs	Memory type	Further interfacing
FX1S	6–16	8–36	100–240 VAC/ 24 VDC	8 k EEPROM (internal)	Relay, Transistor	RAM, ROM, FLASH	RS232
FX1N	4–14	6–24	100–240 VAC/ 12-24 VDC	32 k EEPROM (internal), EEPROM/EPROM cassettes (optional)	Relay, Transistor	RAM, ROM, FLASH	RS232
Our Design	8	4Analog/ 4 Digital	12-24 VDC	64 K Program Memory/2K EEPROM	transistor FET Driver	RAM, FLASH EEPROM	USB RS 485

3.4 Main Design

The proposed IDM in WSN includes inputs, RTC, CPU, LCD, USB temperature, antenna, power, keypad and outputs. For simplicity, the left hand side refers to inputs, while the right hand side of the schematic is for the outputs. In the center of the illustration, the processing and monitoring hardware are located. These are shown in Figure 3.2; for a better readability of the circuit, we have divided the whole figure into 7 sub-circuits. Each part in Figure 3.2 follows by a number. Each number indicates the part(s) that will be explained in the following pages. In the generation of this schematic, we have used Altium Designer, which is very powerful PC-based electronics design software for engineers.

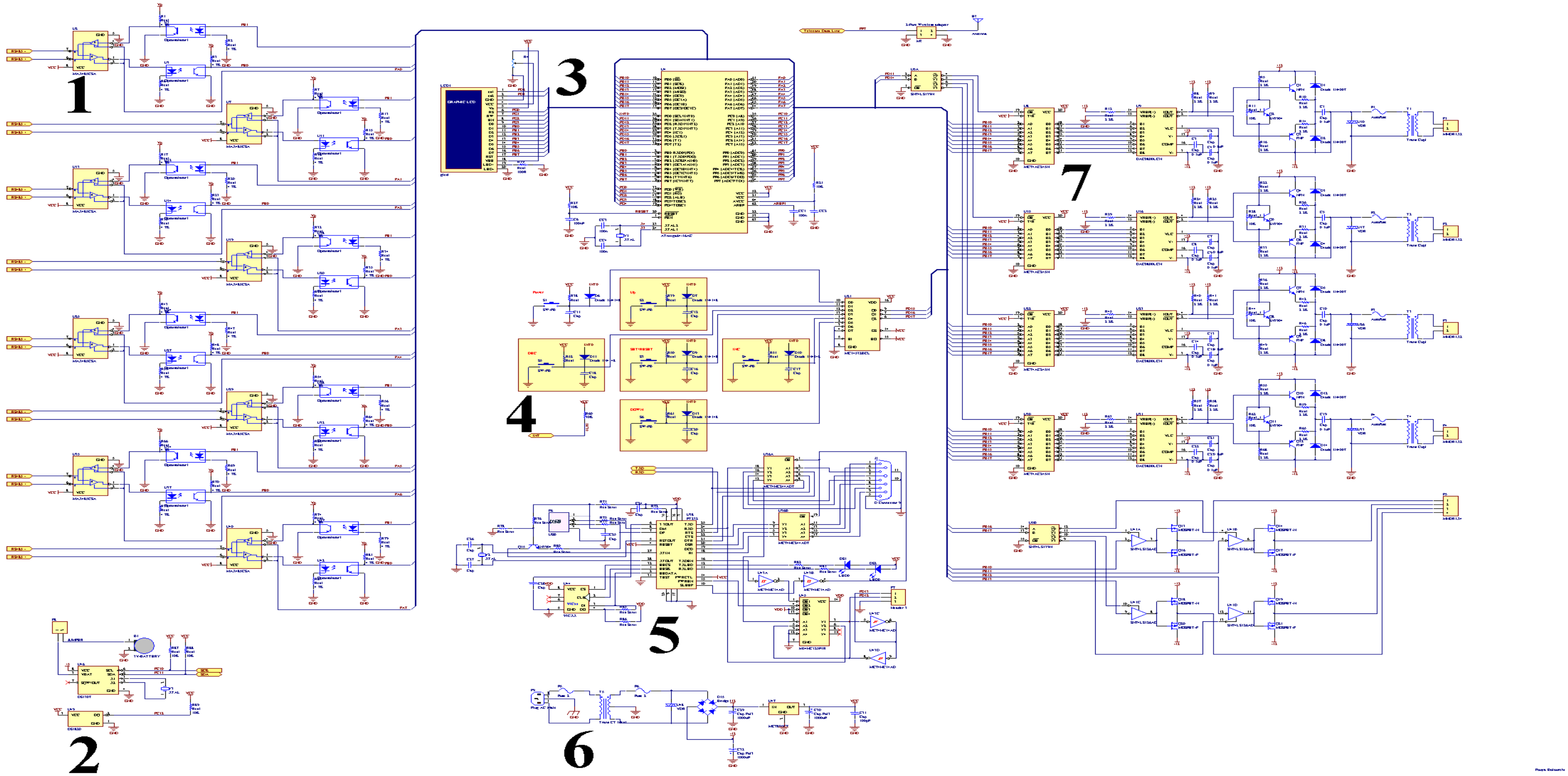

Figure 3.2 The Proposed Design of IDM in WSN

3.4.1 Input

The structure of the input is illustrated is Figure 3.3. RS485 is a port that supports 32 drivers or receivers at the same time. Voltage levels vary from -7 V to +12 V. The main reason of utilizing this port is that it can be used over long distances and in electrically noisy locations. The data is transmitted over a 2-wire twisted pairs. Twisted wire has the advantage of eliminating electromagnetic interference from external sources. As can be seen from Figure 3.3, in the transceiver there are extra pins (2&3) that control the signal either to be sent or to be received. This transceiver converts RS485 port to TTL. Transistor-transistor logic sets 5V voltage that is required by the circuit. The inputs follow by an isolator. This isolator limits the high voltage, which can damage the system. PE1, for example, shows that this should be connected to MCU port PE1. If PE1 is 1, the 5V will pass from Diode and the transistor behaves like an open circuit and it is grounded. Therefore transceiver receives a zero and vice versa.

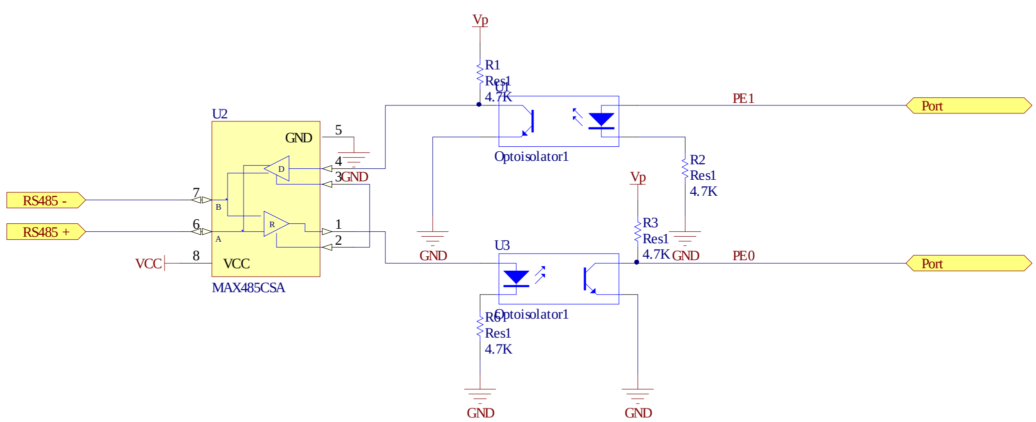

Figure 3.3 Input Part of Design

3.4.2 RTC and Temperature

The DS1307 has an integrated power-sense circuit that discovers power failures and spontaneously switches to the backup source. Timekeeping operation continues while the fragment operates from the backup supply [24]. The upper circuit in Figure 3.4 indicates the RTC structure and the lower one shows the temperature. The DS18S20 digital thermometer delivers 9-bit Celsius temperature measurements and has an alarm task with nonvolatile user-programmable upper and lower trigger points. It has an operating temperature range of -55°C to +125°C and is accurate to ±0.5°C, over the range of –10°C to +85°C. Moreover, the DS18S20 can derive power straight from the data line ("parasite power"), removing the need for an external power supply [26].

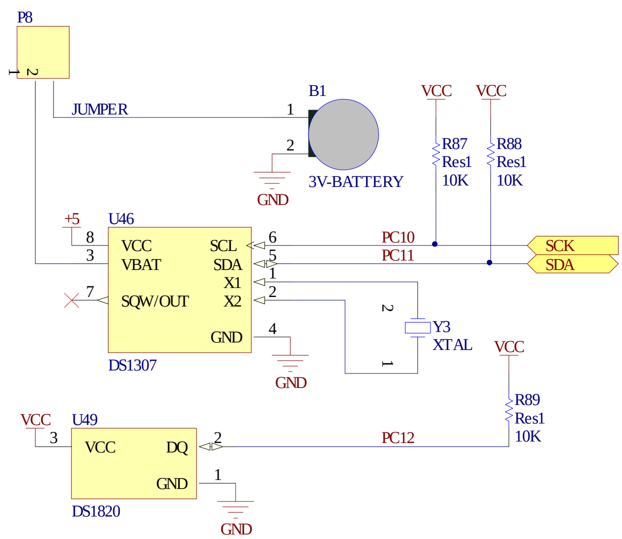

Figure 3.4 RTC and Temperature Part of Design

3.4.3 MCU, LCD and Antenna

MCU is the heart of our design. We utilize high-performance, low-power 8 bits AVR (ATmega64) microcontroller produced by Atmel. It has Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby. Some features are as follows: External and internal interrupt sources, power-on reset and programmable brown-out detection, internal calibrated RC oscillator, software selectable clock frequency and ATmega103 compatibility [25]. The fuzzy logic system becomes executable in MCU by writing a C++ code.

LCD is used to show the events for controlling. We also utilize antenna in order to send and receive the information from the MCU to the central unit. This part of the design is undertaken by another research student. Figure 3.5 shows MCU, LCD and antenna.

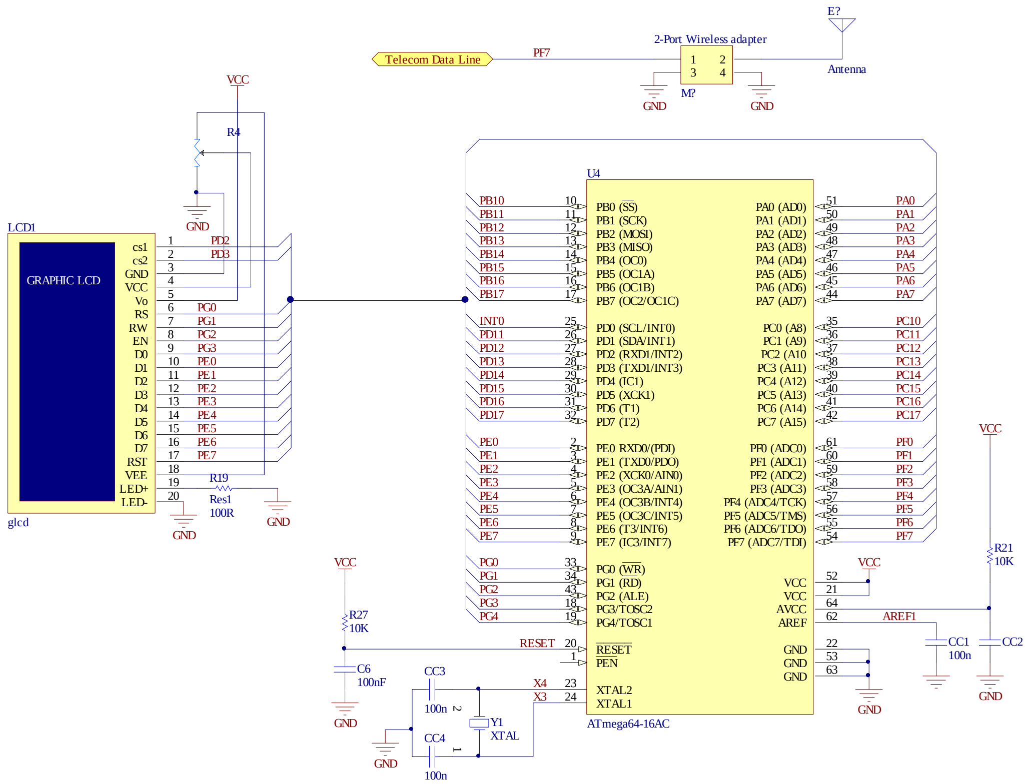

Figure 3.5 MCU, LCD and Antenna Part of Design

3.4.4 Keypad

It consists of six keys as can be observed from Figure 3.6. From top left side to bottom right: Power, Up, Dec, Set/Reset, Inc and down. The power key allows user to turn the circuit on or off. Dec and Inc refers to decrease and increase in controlling some of the features such as temperature or time of the system. The up and down keys are used to change the menu. By set and reset, we can define the desired setup for the system. In the right hand side of Figure 3.6 we use an encoder to code the outputs. Eight data inputs and an enable input are provided. Five outputs are available; three are address outputs one group select and one enable output. For encoder we introduce MC14532BCL. Some feature is listed as follows: Diode protection on all inputs ,supply voltage range = 3.0 Vdc to 18 Vdc, skilled of driving two low–power TTL loads or one low–power and Schottky TTL load over the rated temperature range [27].

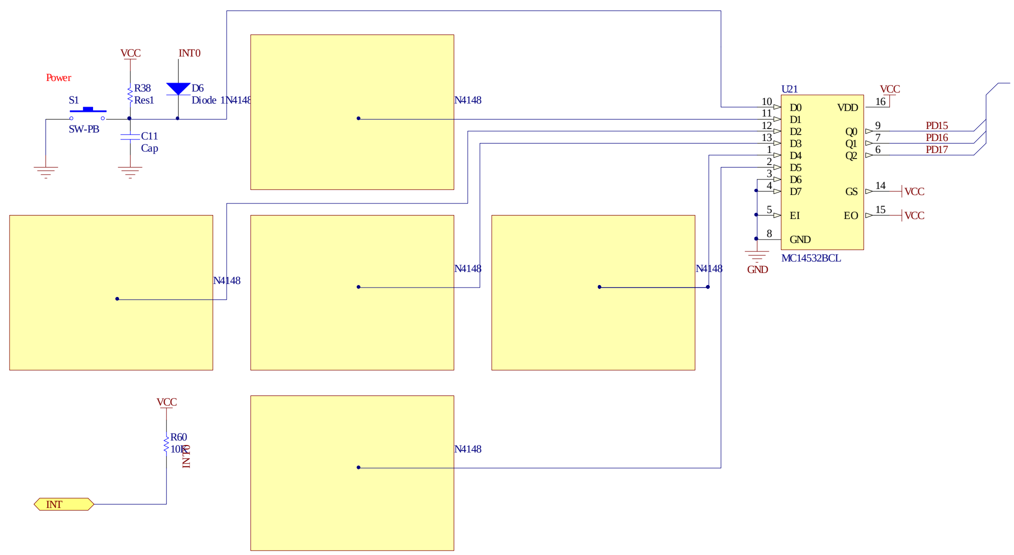

Figure 3.6 Keypad Part of Design

3.4.5 Universal Serial Bus (USB)

In this part of design, we consider the UB232R, which is the smallest USB serial module available in the market. This element allows us to decrease the size of module and develop new applications by adding a USB interface. It supports data transfer rate up to 3 Mbaud[1]. Some of its features are: reduced development time, rapid integration into existing systems USB powered which means no external power

1. baud synonymous to *symbols per second* or *pulses per second*

supply needed, 40°C to +85°C operating temperature range lower operating is (15mA) and USB suspend mode current (70µA) [28]. Figure 3.7 shows the structure of USB design.

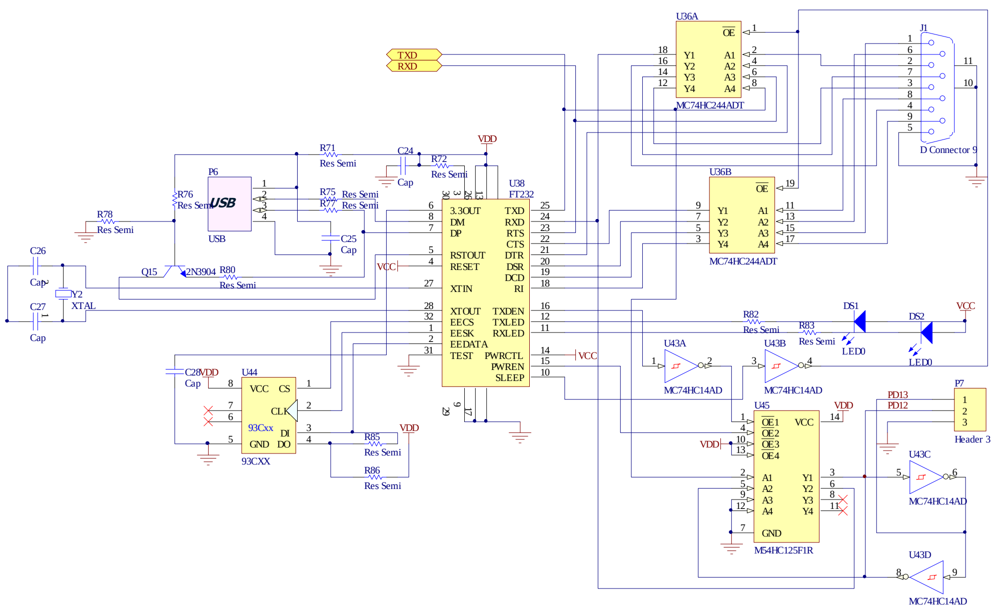

Figure 3.7 USB Part of Design

3.4.6 Power

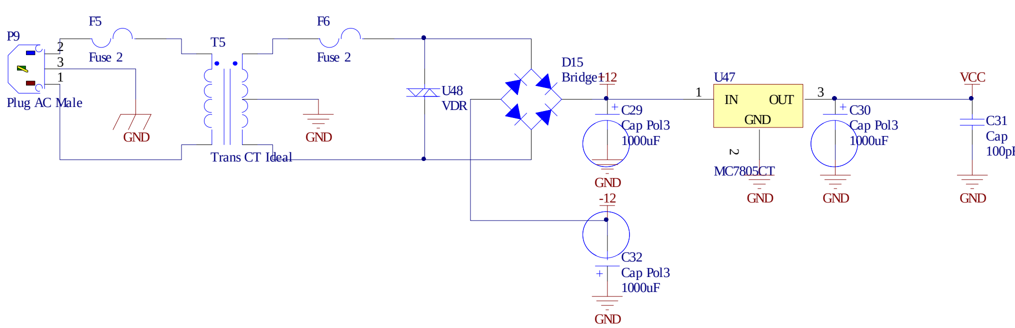

Figure 3.8 Power Part of Design

This power consists of the voltage regulator as fixed−voltage regulators for a wide variety of applications including local, on−card regulation. Some features are output current in excess of 1.0 A, no external components required, internal thermal overload protection and internal short circuit current limiting [30].

3.4.7 Output

We consider both analog and digital outputs in our design. This feature increases the capacity of choosing a desired output. In Figure 3.9, the top part shows the analog parts, while the bottom shows the digital part. Both parts consist of 2 to 4 decoder indicates which ICs should be active.

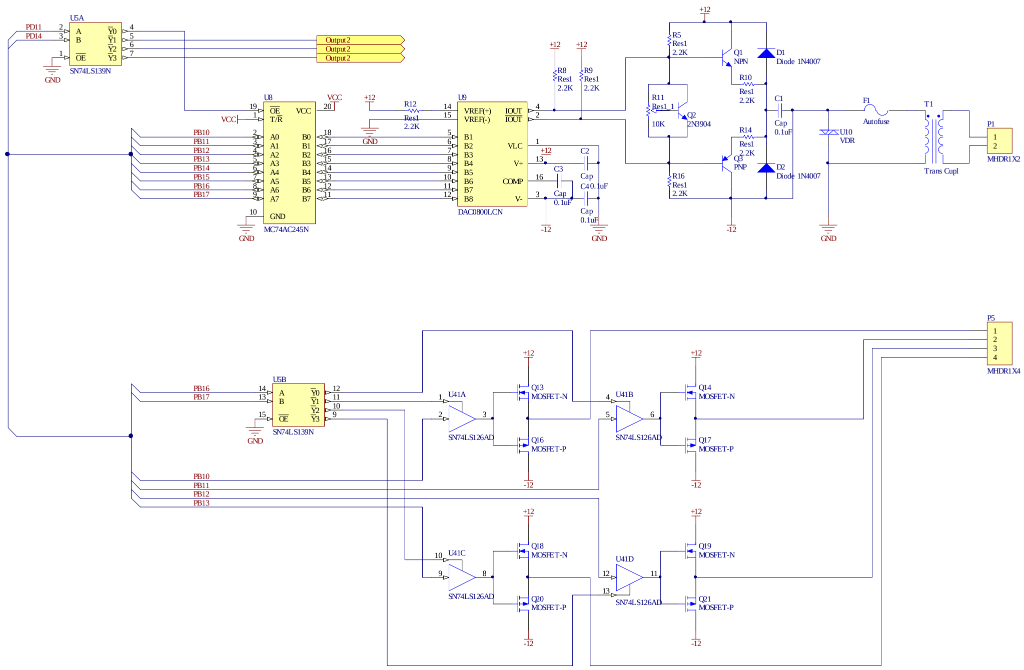

Figure 3.9 Output Part of Design

We use the bus buffers feature in digital part, when they are enabled, they have the low impedance quality of a TTL output with additional drive capability at high logic levels for admitting driving heavily loaded bus lines without external pull up resistors. When disabled, both output transistors are turned off, presenting a high-impedance state to the bus so the output will act neither as a significant load nor as a driver. For analog part, we use a DAC to convert digital to analog.

Chapter 4

METHODOLOGY

4.1 Introduction to Methods of Artificial Intelligence (AI)

In this chapter, we will introduce neural network (NN), genetic algorithm (GA) and fuzzy logic system (FLS). We will study the advantages and weakness of each them and we will investigate the details of FLS method that we have chosen to apply in the proposed decision making system [13].

4.2 Neural Network (NN)

Neural network is a novel information-processing pattern that inspired by the human brain and it processes information in an analogous way that biological nervous systems perform. NNs are composed of interconnecting artificial neurons in order to solve specific problems. NNs will learn by example, therefore, it is vital to choose the most appropriate examples to increase the accuracy in making decisions. A trained NN can be considered as an expert. This expert then can be used in a new application of interest and can answer "what if" questions.

The structure of a NN usually involves three parts or layer of units: layer of input unit, hidden unit and output unit. The sample structure of a NN is given in Figure 4.1.

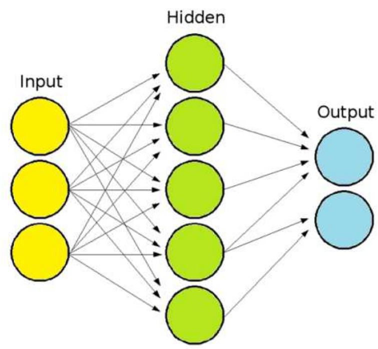

Figure 4.1 An Example of a Simple Neural Network [14]

The input unit indicates the raw information that is fed into the network and send data via synapses to the hidden unit. The hidden unit learns to recode the input and base on the input it can build its own representation. If we increase the number of hidden layers the structure becomes more powerful than a single hidden layer network. The output units are based on the activity of the hidden units [16].

4.2.1 Example of NN

In order to get more insight of a neural network, in this section we will give an example of a specific application. Let us consider that our aim is to detect the fire threat and recognize its level of sensitivity and the corresponding decision making would be based on the level of sensitivity to be decided by using NN. For fire detection, we deploy temperature, light, distance, and smoke sensors; therefore, NN inputs are temperature, light, distance, and smoke and our desire outputs in NN are level of sensitivity as follows: very low, low, medium, high and very high. For each sensor, we will consider the range that we want to train the input. For temperature, we define the range as -10˚C to 140˚C. Then we normalize this range from zero to one. It means that we divide the interval between 0 and 1 into 150 equal sections; for distance, we define the range from 0m to 100m and again we normalize it for the input of NN. Then for light, we consider the range between 0 to 1000 lux and finally for smoke we examine the level of grayness from 0 (white) to 5 (black) and normalize them. For each sensor, we should have different normalized inputs and their respective outputs. More data leads to better training of the system and as a result, decision making accuracy is significantly increased. In our example for each sensor, we assume about 400 inputs and their corresponding outputs. The system will then take these data and change the weight according to the new input it receives. This process continues until all the input-output training is completed. Then for any unknown input given to the system later on, the trained system calculates the weight accordingly and it gives the best output. This output indicates the level of the sensitivity. Therefore, at this stage by using NN we detect the sensitivity level. These levels are then used as new input for FLS that we will see in detail in section 4.4.

4.2.2 Advantages of NN

Advantages of neural networks can be named as follows:

1. Adaptive learning: Ability to learn how it can functionalize itself based on the information provided on early experiences.

2. Organizing themselves: A NN can perform an automatic organization and representation of the data received during the training time. Neurons adapted with the learning rules and response to changing input fields.

3. Simultaneous operation: Computation of NN can be done in parallel and by designing special hardware used for manufacturing them.

4. Fault tolerance: Failure in the network leads to decrease in the performance of the system but some features are still preserved.

5. Classification: NN are able to classify inputs in order to get the desire outputs.

6. Generalization: This property enables the network to deal with only a limited number of samples, a general rule earned based on the learned results is extended to other cases.

7. Stability - Flexibility: A NN is also stable enough to maintain their acquired information and has the flexibility to adapt to new cases without losing previous data.

4.2.3 Disadvantages of NN

NN also have some disadvantages that researchers in this field have tried to minimize which can be listed as follows:

1- Specific rules or instructions to design a network do not exist for the desired application.

2- Modeling issues cannot be merely based on the physical structure of the network. In other words, connection to the network structural parameters or process parameters is often impossible.

3- Accuracy depends on the results of training set; therefore, it requires more training sets to achieve better results.

4- Some networks may be difficult or even impossible to train.

5- Predicting future network performance (by popularity) is not possible.

4.3 Genetic Algorithm (GA)

Genetic algorithms are suggested by Darwin's concept about evolution. Algorithm is initialized with a set of candidate solutions, which characterized by chromosomes called population. Solutions from one population are taken and involved to form a novel population. These solutions are selected according to their fitness. Obviously those solutions that are more suitable have better chances to reproduce. This is repeated until some conditions, like number of populations or improvement of the best solution, are satisfied. The population develops by selection, crossover and mutation [17].

4.3.1 Outline of the Basic Genetic Algorithm

The basic GA can have the following procedure in which we can formulate five basic steps for the algorithm [19]:

“Step 1: Start with an accidental population of n strings.

Step 2: Calculate the fitness f(x) of all the strings in the population.

Step 3: Repeat step 1 and step 2 until n new strings have been produced and now elect a pair of parent strings according to the current population, then crossover will take place with crossover likelihood, form two new strings. Mutate the two new strings with the mutation likelihood, and site the outcome strings in the new population.

Step 4: Substitute the current population with the new population.

Step 5: Go back to second step [19]".

The above procedure is summarized in Figure 4.2.

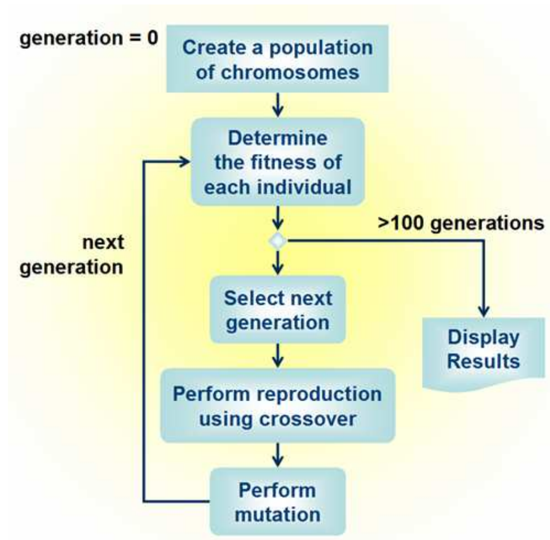

Figure 4.2 A Path Through the Components of the GA

4.3.2 Advantages and Disadvantages of GA

The first positive feature of this algorithm is to achieve optimal overall point (Global Optimum) rather than achieving a local optimum point. The algorithm, in the same form, can solve and apply to problems of the same kind and there is no need for verification. In fact, what we need to do about every issue is to represent different solutions in the form of chromosomes.

Although the genetic algorithm for solving discrete optimization problems is used in the applications, similar methods such as evolutionary strategy algorithm to water or steel (Simulated Annealing) could have applied. Definition and implementation of this algorithm is so that they make appropriate execution using a multiprocessor.

On the other hand, the main drawback in spite of GA's ease in implementation is the added cost in the system realization. Often solving a problem requires the production of several thousand generations of chromosomes and this has required much time (especially if the initial population is high and the objective function is a complex function). Another weakness of GA is that sometimes in solving a problem the duration of execution is too long, for example, a Pentium processor is required to work more than a week to run the program. This time is obviously too much to solve a problem and as a result the algorithm encounters difficulties.

4.4 Fuzzy Logic System (FLS)

Fuzzy logic system is a system that imitates the human thinking. Fuzzy logic was proposed by Lotfi Zadeh in 1965. In contrast with binary logic system in which we deal only with zeros and ones that are exact, FLS can have a continuous value between zero and one which are approximations. The basic structure of FLS consists of Fuzzification, Inference and Defuzzification. Fuzzification is a process that crisp inputs are converted to fuzzy inputs. A membership function (MF) demonstrates the degree of truth of each input and output. MF has value between 0 and 1 over an interval of crisp variable. The mostly used shapes of membership functions are triangular, trapezoidal and bell curves. Each MF has different variables according to number of variables are pre-defined for each individual input/output. MF in applications introduced in chapter five has at most five inputs and output variable as follows: Very Low (VL), Low (L), Medium (M), High (H) and Very High (VH). These fuzzy inputs then fed to inference in which fuzzy rule manage the inference for yielding a fuzzy output. Although there are lots of fuzzy rule basis, Mamdani and Sugeno [20] [29] are mostly used in the applications due to their easy concepts. In this thesis, we will focus on Mamdani method. In 1975, Ebrahim Mamdani built the first fuzzy system to control a steam engine and boiler combination that mimics human operators. In Mamdani type, the focus is to find the centroid of a two-dimensional function. We will use the weighted average of a few data points to find the centroid by integrating across a continuously varying function. The final step is Defuzzification process in which the fuzzy output is converted to a single number. There are diverse built-in methods applied for Defuzzification: centroid, bisector, middle of maximum (the average of the maximum value of the output set), the largest of maxima, and the smallest of maxima. Centroid method is commonly used in applications. The schematic structure of FLS is given in Figure 4.3.

The fuzzy rule is written as the following statement:

IF x_1 is E_1 and x_2 is E_2 and …and x_n is E_n THEN y is y_k (4-1)

Where x_1, x_2 ,…and x_n indicate the input variables of FLS. E_1, E_2,…and E_N are fuzzy

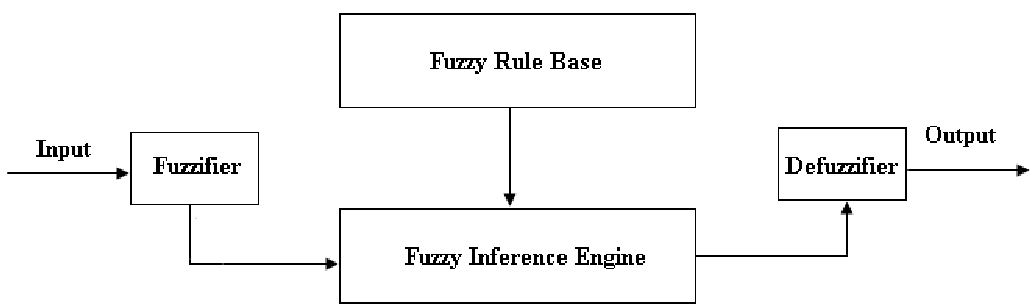

Figure 4.3 An Example of Fuzzy Logic System

memberships. y is the output in a specific parameter and y_k is the output variable of that parameter.

In fire detection application five inputs are temperature, humidity, light, distance and CO. MF for temperature and output which is the possibility of fire has five variables as follows: very low (VL), low (L), medium (M), high (H) and very high (VH). MF of humidity, light and CO has three variables as low, medium and high and MF of distance has only two variables: close and far. These fuzzy inputs then fed to inference in which fuzzy rule base mange the inference for yielding a fuzzy output. Figure 4.4 shows a summary of this section.

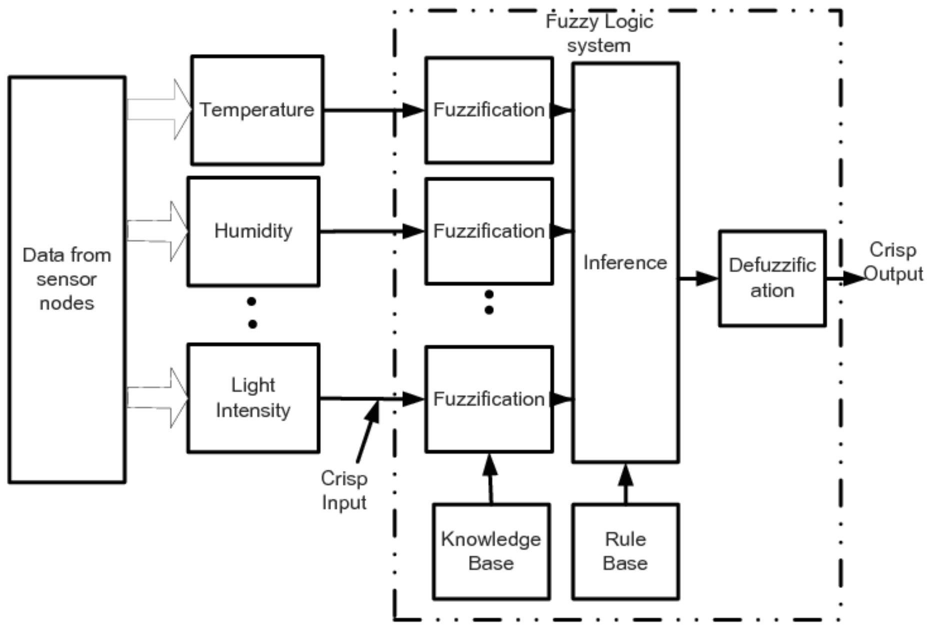

Figure 4.4 Structure of a Fuzzy Logic System with Multi Sensors [31]

Chapter 5

Case Study

5.1 Summary of Tasks

We have studied the behavior of sensor networks in Chapter 2, and then the system architecture was proposed in Chapter 3. In Chapter 4, three different methods of intelligent decision making were introduced with their own advantages and disadvantages. In this chapter, we use the fuzzy logic system for two different situations. First, is a case study for a simple task that involves threat of fire and the next one will be a fuzzy control in a feedback system in controlling the climate in greenhouses.

5.2 Case 1: Threat of Fire

In this section, we give a case study for a real life application. Then by using MATLAB toolbox (Fuzzy Logic Toolbox) we implement the given example and finally we discuss the outcomes. The fuzzy logic rule programming in this example is given in Appendix A. As we mentioned earlier in section 4.2.1 our main goal is to detect the threat of fire by deploying five different inputs and making a proper decision based on the outputs. For inputs, we consider X_1 as a temperature variable and define five levels in the range from -20˚C to 150˚C as very low, low, medium, high and very high. X_2 is humidity and can be set out in three levels as high, medium and low. It has a unit of ppm and the range is between zero and 1000. X_3 indicates light and is described as three levels: low, medium and high. The unit of light is lux and the range varies from 0 to 1000. X_4 indicates CO that has a range of zero to 1000 ppm and follows by three levels as low, medium and high. X_5 is the distance, which can be specified in two levels as far and near indicating the range between zero and 120 m. The following figures (5.1 to 5.6) show the membership function through X_1 to X_5 and output, respectively. Horizontal axis represents the range of input crisp that is from -20 to 150˚C. Vertical axis is normalized and indicates degree of membership.

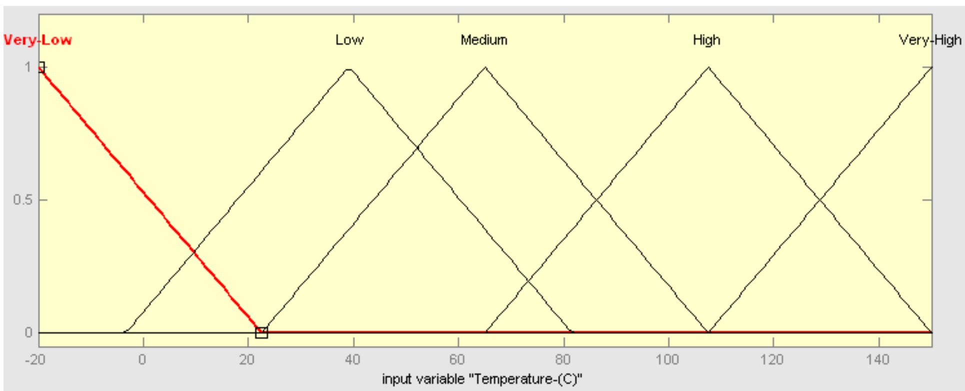

Figure 5.1 Membership Function of X_1

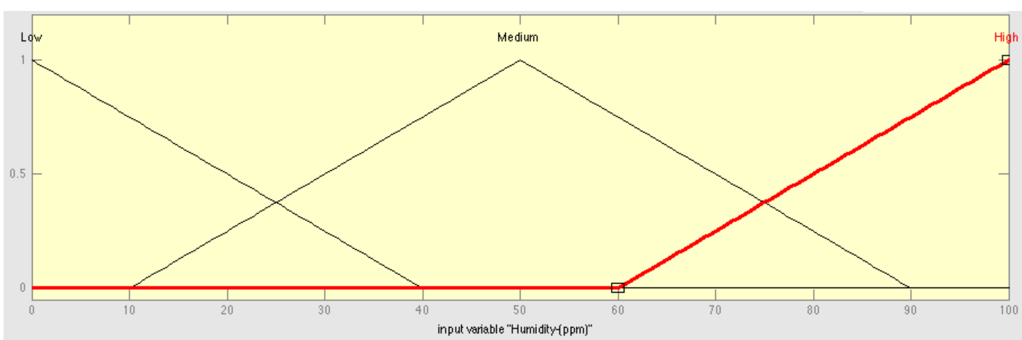

Figure 5.2 Membership Function of X_2

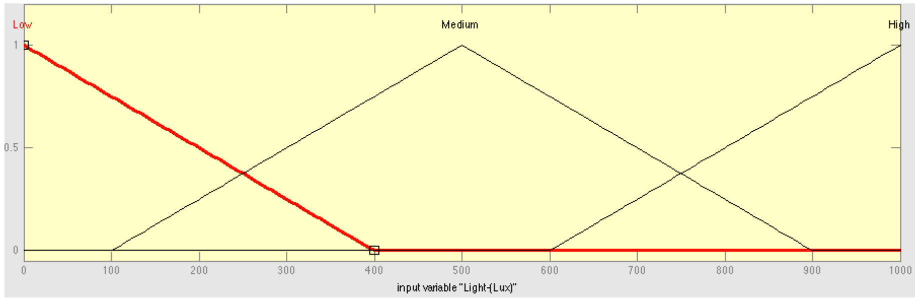

Figure 5.3 Membership Function of X_3

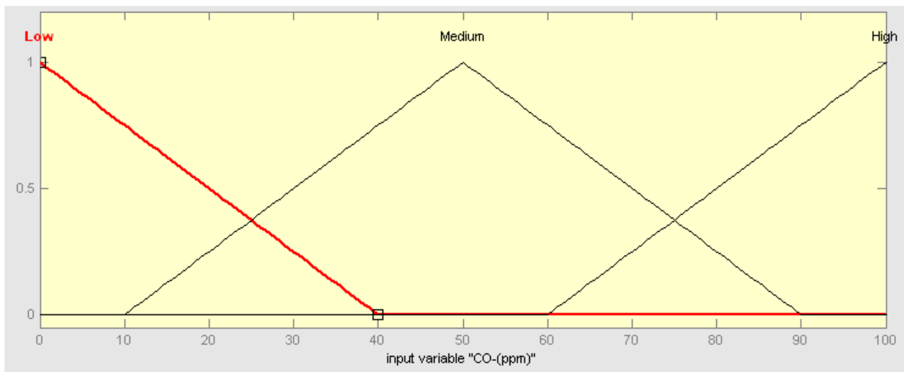

Figure 5.4 Membership Function of X_4

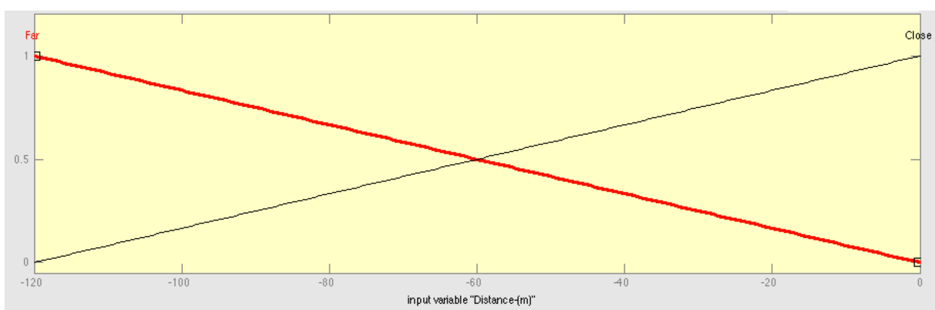

Figure 5.5 Membership Function of X_5

We can also define the range of sensitivity. As can be seen the distribution of range of sensitivity in temperature is not equal while for other membership we define the same range. In addition, membership function of temperature has five values while for humidity, we have three values and for distance, we define only two values.

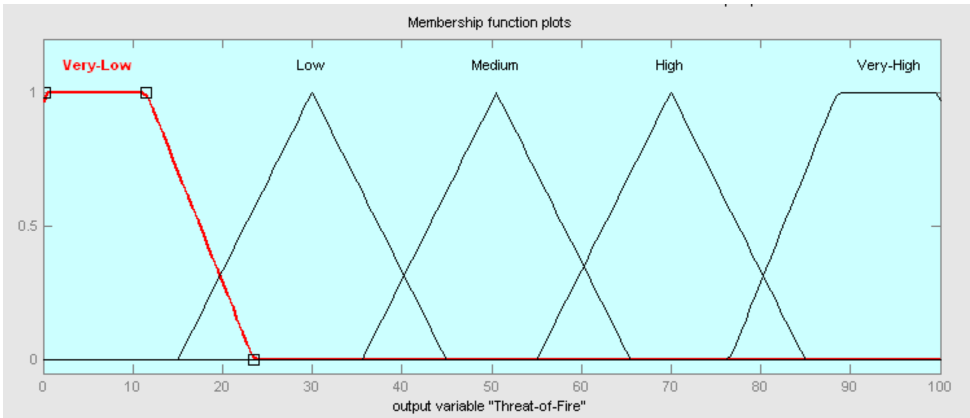

Figure 5.6 Membership Function of Output

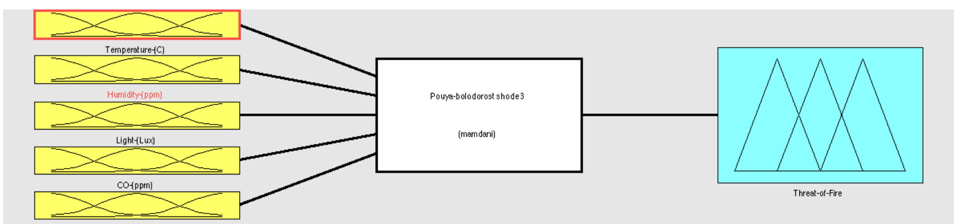

Figure 5.7 Fuzzy System Structure

Figure 5.7 shows the whole structure of the fuzzy logic system including five inputs, reasoning rules and outputs. We define the rules for each situation. Since we have five inputs for X_1, 3 inputs for X_2, X_3 and X_4 and 2 inputs for X_5, the maximum number of rules is 270 (5*3*3*3*2). Although it is not necessary to complete all the rules, but for getting the result, we precisely define all possible 270 rules. The following page gives the first 10 rules:

1- If (Temperature is Very Low) and (Humidity is High) and (Light is Low) and (CO is Low) and (distance is Far) **then** (Threat of Fire is Very Low)

2- If (Temperature is Very Low) and (Humidity is High) and (Light is Low) and (CO is Low) and (distance is Close) **then** (Threat of Fire is Very Low)

3- If (Temperature is Very Low) and (Humidity is High) and (Light is Low) and (CO is Medium) and (distance is Far) **then** (Threat of Fire is Very Low)

4- If (Temperature is Very Low) and (Humidity is High) and (Light is Low) and (CO is Medium) and (distance is Close) **then** (Threat of Fire is Very Low)

5- If (Temperature is Very Low) and (Humidity is High) and (Light is Low) and (CO is High) and (distance Far) **then** (Threat of Fire is Very Low)

6- If (Temperature is Very Low) and (Humidity is High) and (Light is Low) and (CO is High) and (distance is Close) **then** (Threat of Fire is Low)

7- If (Temperature is Very Low) and (Humidity is High) and (Light is Medium) and (CO is Low) and (distance is Far) **then** (Threat of Fire is Very Low)

8- If (Temperature is Very Low) and (Humidity is High) and (Light is Medium) and (CO is Low) and (distance is Close) **then** (Threat of Fire is Very Low)

9- If (Temperature is Very Low) and (Humidity is High) and (Light is Medium) and (CO is Medium) and (distance is Far) **then** (Threat of Fire is Very Low)

10- If (Temperature is Very Low) and (Humidity is High) and (Light is Medium) and (CO is Medium) and (distance is Close) **then** (Threat of Fire is Low)

We also can arrange the rules by using a table. For simplicity instead of Very Low, Low, Medium, High and Very High, we use VL, L, M, H and VH respectively. This model avoids to use (if), (and) and (then) and it reduces the complexity and is in a more presentable style. The 270 rules can be observed as a table in Appendix A. Appendix B also represents the MATLAB code for the given problem.

Table 5.1 First 10 Rules of Fuzzy Logic in Case of Fire

Rule	Temperature	Humidity	Light	CO	Distance	Output
1	VL	H	L	L	Far	VL
2	VL	H	L	L	Close	VL
3	VL	H	L	M	Far	VL
4	VL	H	L	M	Close	VL
5	VL	H	L	H	Far	VL
6	VL	H	L	H	Close	L
7	VL	H	M	L	Far	VL
8	VL	H	M	L	Close	VL
9	VL	H	M	M	Far	VL
10	VL	H	M	M	Close	L

As an example for the fire threat, consider the temperature is 30˚C, the distance is 100m, the light is 300 lux, the CO is 30 ppm and humidity is 80 ppm. For each sensor, we define a level of sensitivity. For temperature, 30˚C is the range between low and medium with a different weighting scale and low sensitivity has a higher weight scale rather than medium. If we follow the membership functions in Figure 5.1, we will see that for medium level, the weight is 0.3 but for low level, it has a weight around 0.7. For distance, 100m has a higher weight scale in the far area rather than close area and it can be estimated from Figure 5.5 that the scale is 0.2 for close and 0.8 for far area. For humidity, light and CO the weight can be estimated in the same way according to membership functions. Obviously low sensitivities in CO and light have a greater weight rather than medium sensitivities and for humidity, high sensitivity has a greater weight in comparison to its medium level. By the use of fuzzy logic, we can easily calculate the threat of fire. Therefore, in the given example, the probability of fire is calculated and the result is 36%. Figure 5.8 shows the first 30 rules and by giving the specific number for each sensor, we can see how inputs-output varies. As an example, consider the first rule:

If (Temperature is Very Low) and (Humidity is High) and (Light is Low) and (CO is Low) and (distance is Far) **then** (Threat of Fire is Very Low)

MF of temperature has five variables and rule 1 for temperature illustrates to what extend one variable (very low) effects on the output of the first rule. For temperature, 30˚C indicates the range between low and medium. Therefore, for very low there is no weight. For humidity, focus is on high sensitivity that has a scale of 80 ppm, and has a weight of 0.8, which is indicated by yellow part in Figure 5.8. Degree of low level of light (300 lux) and CO (30 ppm) and far level of distance (100m) can be seen by yellow parts. Since for all rules we apply the AND fuzzy operation the intersection or minimum between the two sets, expressed as:

$$\mu_A \cap \mu_B \cap \mu_C \cap \mu_D \cap \mu_E(x) = \ min\ [\mu_A(x), \mu_B(x), \mu_C(x), \mu_D(x), \mu_E(x)] \qquad (5\text{-}1)$$

Where: μ is the degree of truth for each crisp input. A, B, C, D and E are Temperature, Humidity, light, CO and distance respectively.

For all 30 first rules and 30 last rules shown in figures 5-8 and 5-9 since all temperature values have no weights and according to equation (5-1) the minimum value of all inputs evaluates the responsible outputs, there is no weight for any output in these 60 rules.

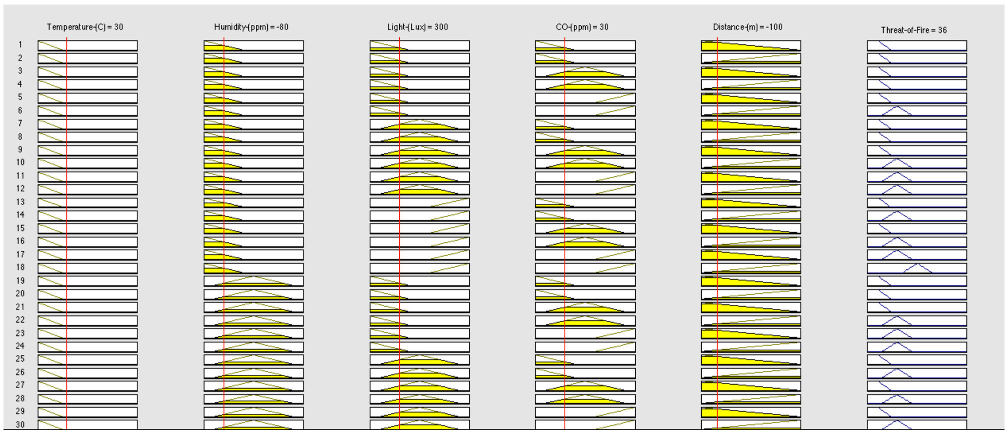

Figure 5.8 Mamdani Method to Calculate the Output

The output is calculated by the centroid method as we have discussed in the previous chapter. For other 269 rules, the output is calculated in the same manner. At the end, we sum up the outputs and their average gives the result of the output. The result in the above example is given in Figure 5.9. Output gives a specific number – the red line indicates the probability of fire is 36%.

With this system, we are able to calculate the threat of fire just by reading the crisp value of input sensor. We examine the method by six other reading sensors and we put the final result of each sensor in Table 5.2. Each time we fix four values only by modifying one value.

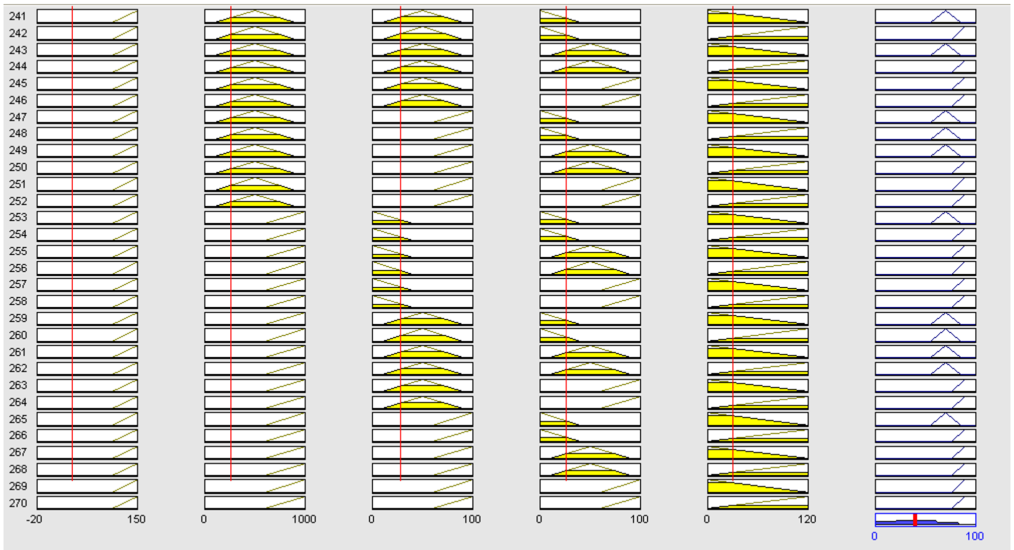

Figure 5.9 Result of Calculating Fire Threat

As we can see, by increasing the value of temperature, light and CO the treat of fire is also increasing, while by increasing distance and humidity the threat of fire is decreasing. For calculating sensitivity of these sensors, we define the range from a negative value to zero.

Table 5.2 Some Outputs Due to Given Inputs

	Temperature (Celsius)	Humidity (ppm)	Light (Lux)	CO (ppm)	Distance (m)	Threat of Fire (percentage)
1	30	80	300	30	100	36
2	80	30	300	25	40	61.6
3	20	30	300	25	40	46
4	20	60	300	25	40	38.5
5	20	60	700	25	40	40.3
6	20	60	700	90	40	61.9
7	20	60	700	90	120	50.5

By using MATLAB, we can find out the relationship between inputs and output. First, we analyze each input output alone in Figures 5.10 to 5.13. These figures show the relationship between temperature, humidity, CO and distance with output, respectively. Figure 5.10 shows that from -20˚C to 0˚C threat of fire is about 20%.

For 60˚C the threat of fire is 50%. The probability of fire increases gradually as temperature increases and it reaches to its maximum, which is 80% as temperature

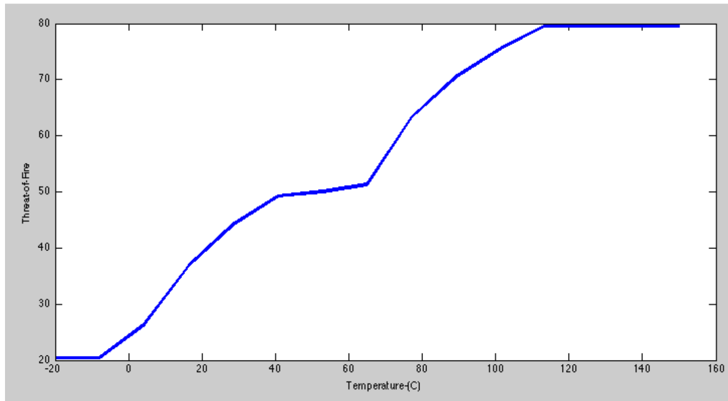

Figure 5.10 Relation between Temperature and Output

reaches to 120˚C. In Figure 5.12 threat of fire is about 50% if amount of CO is less than 50 ppm. As CO increases, the output increases drastically and reaches to 75%. Figure 5.13 also shows that as distance decreases, the threat of fire increase. For showing that distance and humidity is inversely proportional to output, we use negative sides of axis.

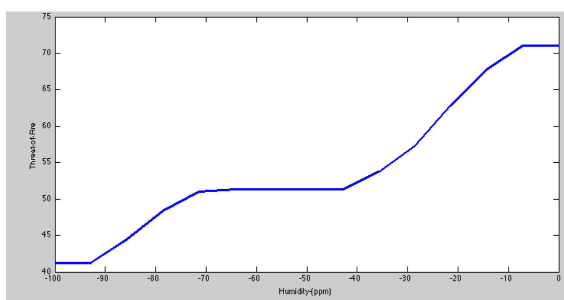

Figure 5.11 Relationships between Humidity and Output

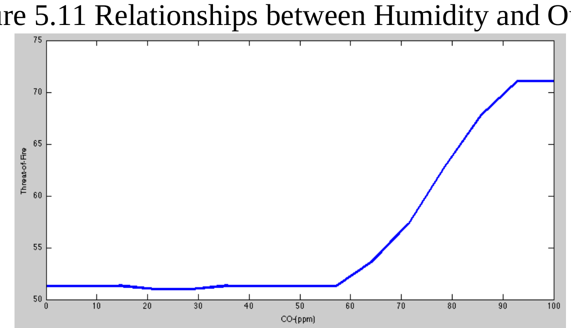

Figure 5.12 Relationships between CO and Output

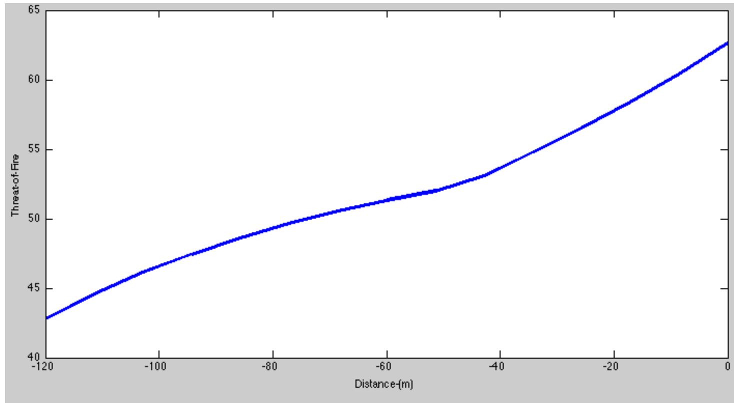

Figure 5.13 Relationships between Distance and Output

Now we analyze the control surfaces of two inputs and output. Since there are 5 inputs the total available graphs are 10(4+3+2+1). We only show a few of them; in Figure 5.14 the relationship between light and CO with output is given.

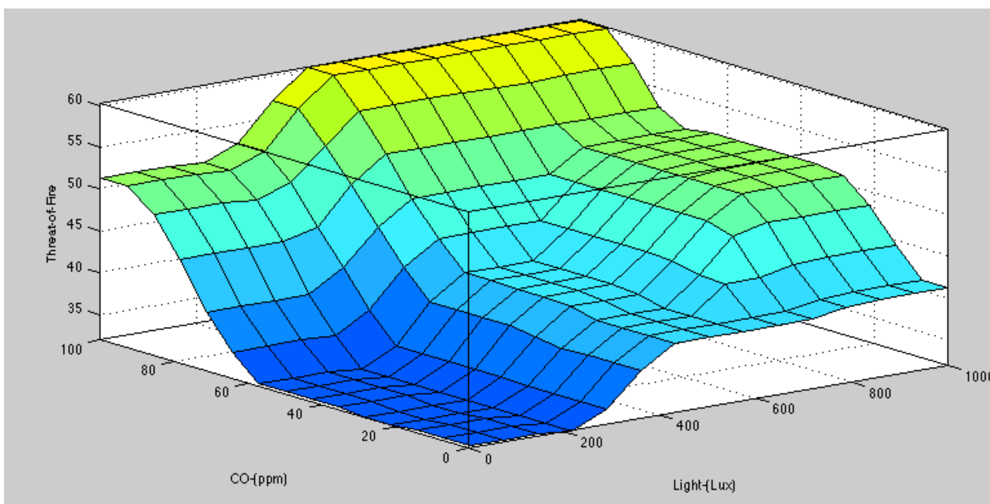

Figure 5.14 Light and CO vs. Output

As we can observe from the graph when both light and CO have a low sensitivity (near to zero), the output is near to zero that indicates very low level of sensitivity. As deviation of light is departed from zero and reaches to 200 lux and CO reaches to about 60 ppm, still the threat of fire is around zero. In the range of 0 and 50 for CO and range of 200 to 400 for light the threat is increasing gradually and probability of fire is around 40%. While with the same range of light (200 to 400 ppm) if we consider the range of CO between 50 and 100, then the threat of fire grows rapidly and it reaches to about 45% to 60%. This graph also indicates that if both factors reach to their maximum value, the output tends to its maximum value which is about 60. We also conclude from the figure that, if we consider only these two inputs, the maximum probability of fire is around 60.

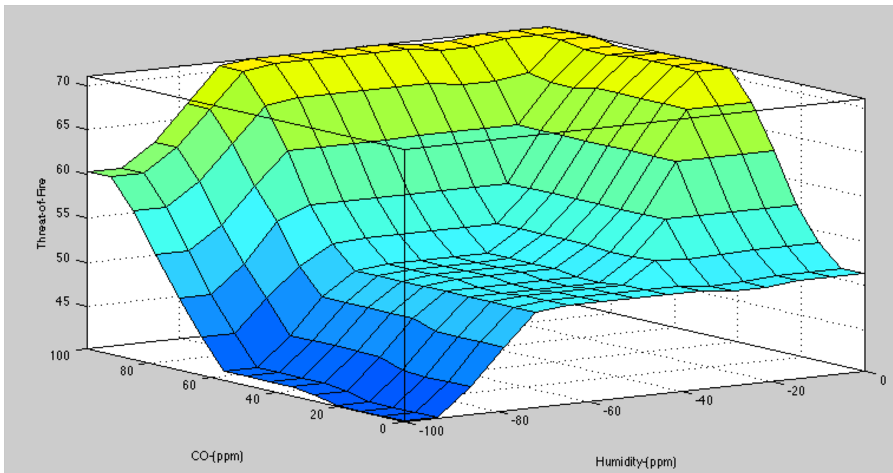

Figure 5.15 Humidity and CO vs. output

With the same concept, we can interpret the relation of any two-dimensional inputs with the output.

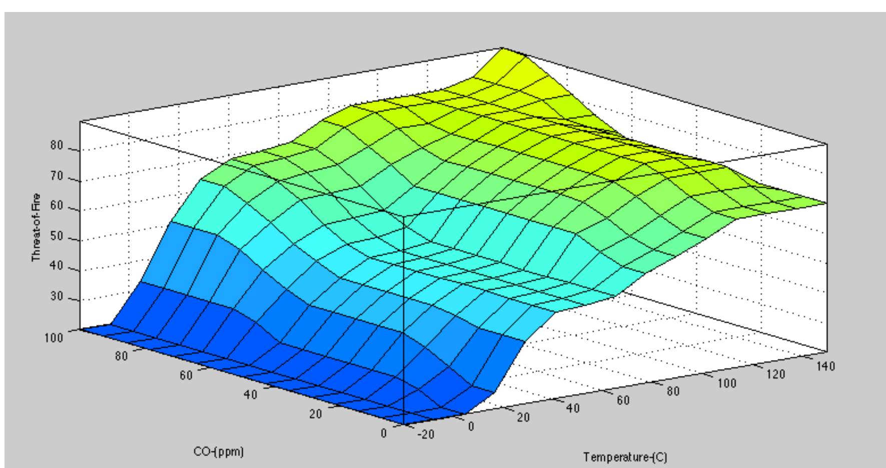

Figure 5.16 CO and Temperature vs. Output

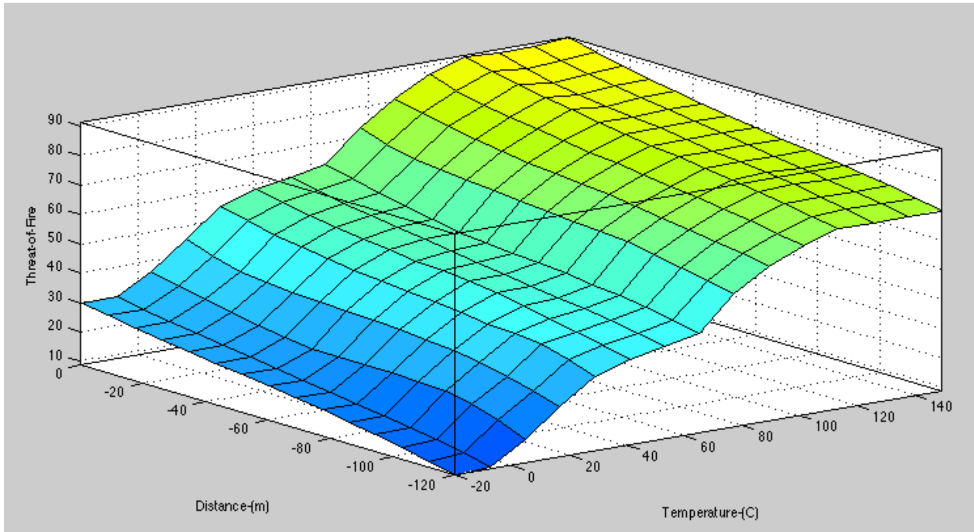

Figure 5.17 Distance and Temperature vs. Output

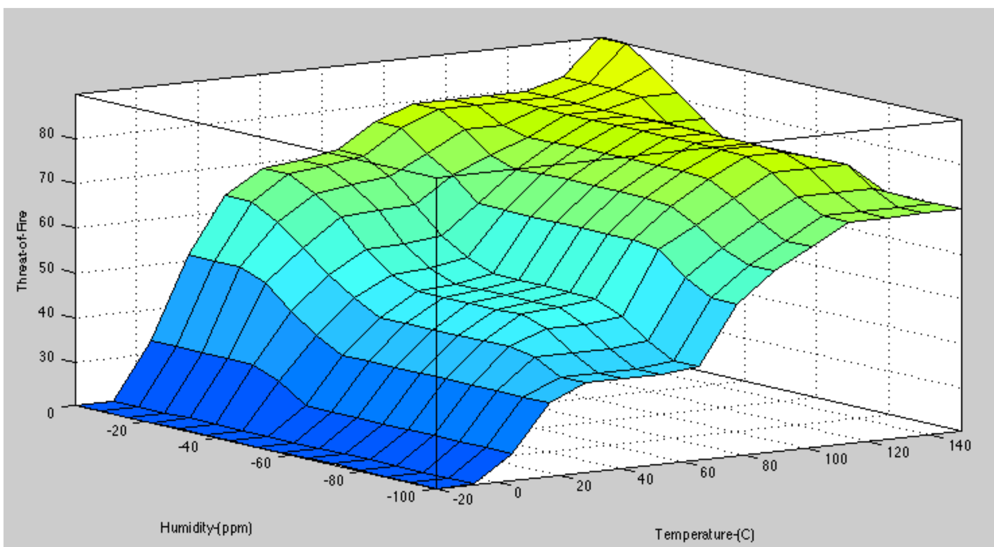

Figure 5.18 Humidity and Temperature vs. Output

Figure 5.15 to 5-18 show the humidity and CO, temperature and CO, distance and temperature and humidity and temperature respectively.

5.3 Case 2: Fuzzy Control in Feedback System

In the last section, we have studied the threat of fire. In fact, our system intelligently makes its own decision according to the level of output, i.e., if the level of sensitivity is about medium or even higher it sends the information to the base station for further analysis. Otherwise, the system performs some simple task such as giving an alarm. Now we want to develop our system. With the use of FLS, we want to satisfactorily control an event. That is, if some inputs give an undesirable outcome, then by changing the output function we can overcome some unwanted environmental threats. For better understanding, we study a special case in a greenhouse [32-34] where monitoring for crops is essential. Many items can affect the whole system. Medium and high-tech greenhouses make use of a range of sensors combined in an automated control system. These systems can monitor temperature, humidity, light intensity, electrical conductivity, pH, carbon dioxide, wind speed and direction and even whether or not it is raining. The gathered data are used to control heating, venting, fans, screens, nutrient dosing, irrigation, carbon dioxide supplementation and fogging or misting systems [18]. In the following example, we consider the most important items as inputs that should be controlled in a greenhouse. These inputs are temperature, humidity and light. We define four outputs such as fan, heater, humidifier and light-controller as we can see in Figure 5.19. Since controlling of light is independent from other outputs such as fan and heater, we can define the following rule for controlling the light.

1- If light is high then light-controlling is Off
2- If light is Medium then light-controlling is Medium
3- If light is Low then light-controlling is On

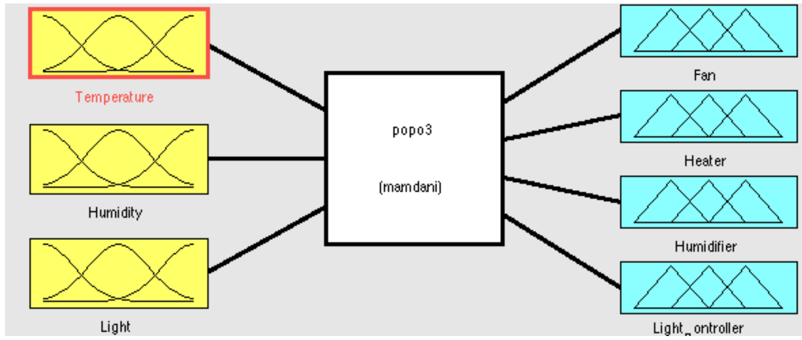

Figure 5.19 Input Output Relation for Greenhouse

As we see, each input is followed by three membership functions: low, medium and high. The output follows three memberships as off, medium and on. The system intelligently makes a decision for each input accordingly. If the light is low for compensating darkness, the system automatically turns florescent lights on.

We are also interested to control the temperature and humidity of the greenhouse. We deploy temperature and humidity sensors as inputs. Like previous section, we define the member function for each input. The membership functions for both sensors are very low, low, medium, high and very high. Outputs are fan, heater and humidifier with each having five-membership functions as off, minimum, medium, high, and maximum. The main difference between this case and fire case is, in the case of fire we are interested in detecting the probability of fire, while here with the utilization of outputs we want to reach to a desired outcome. That is, if for example the temperature of environment decreases by using heater, the system compensates the low temperature that may harm plants. Figure 5.20 shows some rules of the given example and Table 5.3 indicates all the rules given for controlling the temperature and humidity.

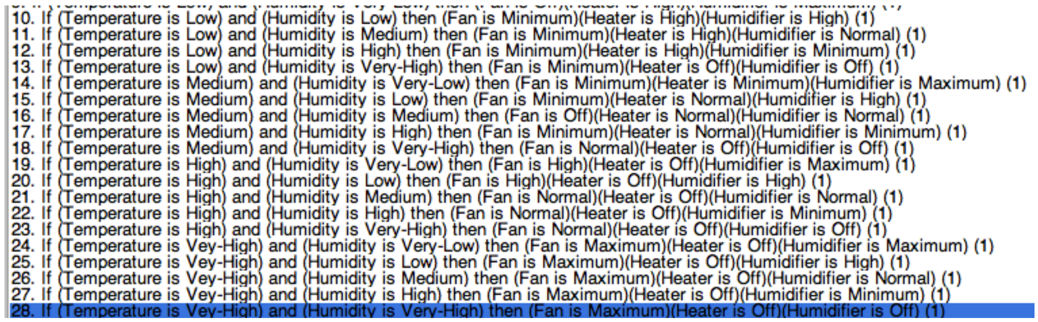

Figure 5.20 Input-Output Rules in the FLS

Table 5.3 All Input-Output Rules in FLS

Rules	Temperature	Humidity	Fan	Heater	Humidifier
1	Very Low	Very Low	Off	Maximum	Maximum
2	Very Low	Low	Off	Maximum	High
3	Very Low	Medium	Off	Maximum	Medium
4	Very Low	High	Off	Maximum	Low
5	Very Low	Very High	Off	Maximum	Off
6	Low	Very Low	Minimum	High	Maximum
7	Low	Low	Minimum	High	High
8	Low	Medium	Minimum	High	Medium
9	Low	High	Minimum	High	Low
10	Low	Very High	Minimum	Maximum	Off
11	Medium	Very Low	Minimum	Normal	Maximum
12	Medium	Low	Minimum	Normal	High
13	Medium	Medium	Off	Normal	Medium
14	Medium	High	Minimum	Normal	Low
15	Medium	Very High	Normal	Normal	Off
16	High	Very Low	High	Minimum	Maximum
17	High	Low	High	Minimum	High
18	High	Medium	Normal	Minimum	Medium
19	High	High	Normal	Minimum	Low
20	High	Very High	Normal	Minimum	Off
21	Very High	Very Low	Maximum	Off	Maximum
22	Very High	Low	Maximum	Off	High
23	Very High	Medium	Maximum	Off	Medium
24	Very High	High	Maximum	Off	Low
25	Very High	Very High	Maximum	Off	Off

Now we study behavior of the temperature and humidity for fan, heater and humidifier, respectively. Figure 5.21 shows how fan controls the temperature and humidity of the system. With a very low temperature (-20˚C) and very low humidity (0 ppm) the fan is off. If temperature stays at this level and level humidity varies to very high still the fan is kept in the sleep mode. As temperature increases, the fan starts to work with its minimum speed.

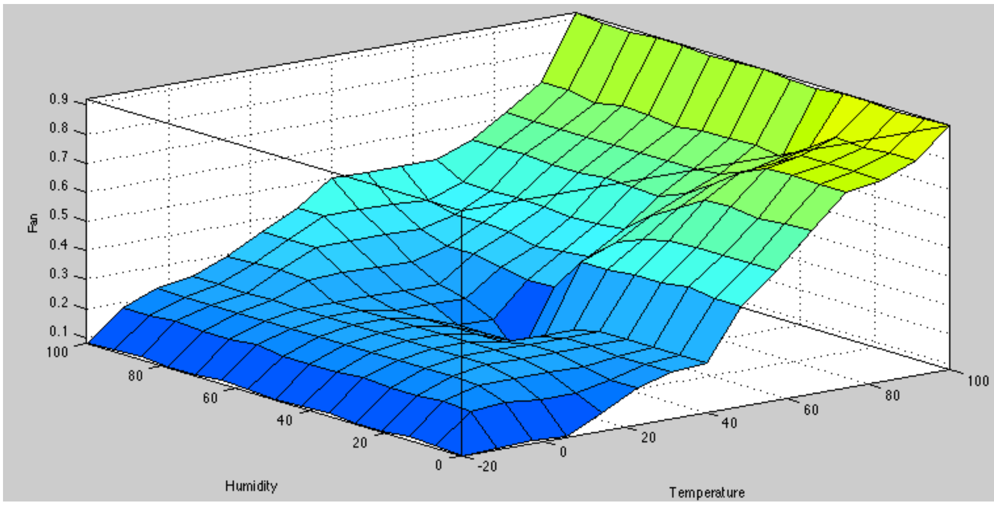

Figure 5.21 Temperature and Humidity vs. Fan

If the humidity reaches to near 50 ppm that is in the range of medium level and the temperature is, around 40 degrees then fan stops working again. After 60 degrees the fan works faster and faster. If the temperature reaches to 100 degrees then fan works at the maximum speed. Figures5.22 and 5.23 show the relationship of inputs and heater and humidifier respectively, in the same manner as we have explained.

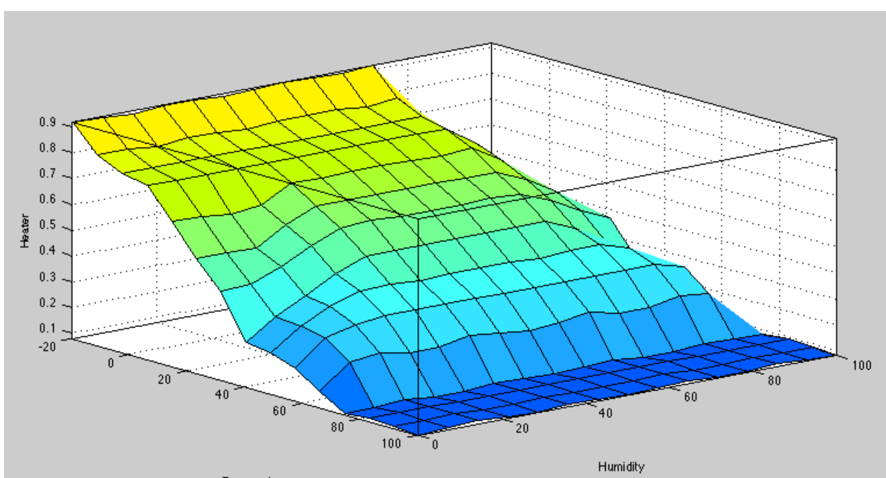

Figure 5.22 Temperature and Humidity vs. Heater

Now, we can see some examples to see how the system responds to control the environment according to data gathered by sensors. Consider temperature and humidity have a value of 20˚C and 20 ppm and light has a value of 500 lux.

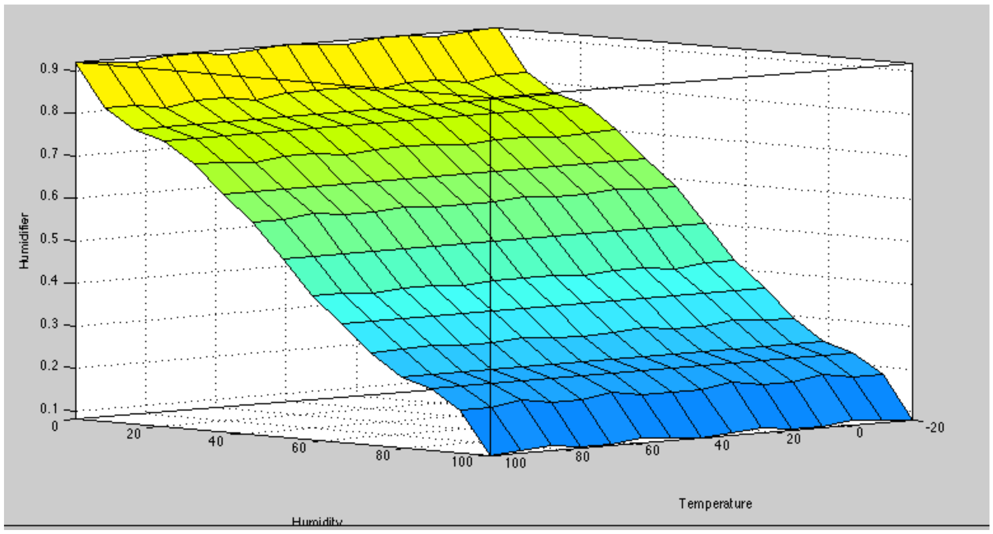

Figure 5.23 Temperature and Humidity vs. Humidifier

In this case, we define 28 rules. The first three rules show the relationship between light and light-control. Other 25 show the relationship between the temperature and humidity with fan, heater and humidifier. For light, we consider the input is 500 lux and the output is 0.5, that is, the light control is varying from off to on. We have calculated the output of fan, heater, humidifier and light-controller and we got the following number as we can see in Figure 5.24. Fan has a value of 0.191 and has a weighting factor only for four rules (rule 9, 10, 14 and 15); rule 9 and 10 indicate the extend the fan is off in this area. Rule 10 shows the fan has more tendencies to be off. On the other hand rule 14 and 15 show how to extend the fan's tendency to work with minimum speed. All these four rules cauterized and they give a unique number that illustrates whenever the temperature and humidity are 20˚C and 20 ppm, respectively. The tendency of fan to work is 0.191, which means that the fan varies from off to work with minimum speed. The heater is 0.587 and again four rules will be employed to show the tendency of heater to work.

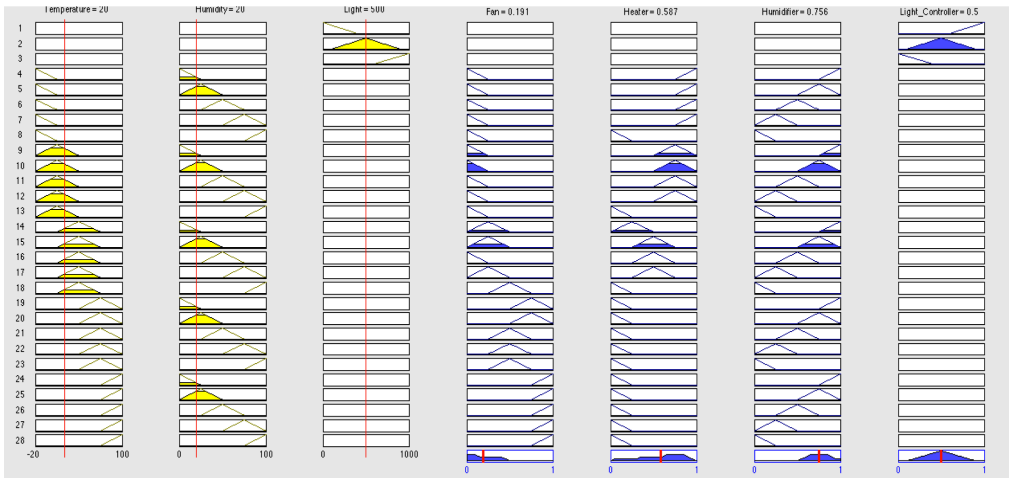

Figure 5.24 Final Result for a Given Example

For rules 9 and 10, it is in the area of high speed, rule 14 with minimum speed and rule 15 with normal speed. Each color inside them shows the extension in that area. For example, rule 10 indicates the heater wants to work in high speed. The centroid number shows that the heater has a tendency to work between minimum and high speed.

Humidifier shows 0.756 indicating that the motor has a high tendecy to work with high speed. Again 4 rules are accomplished to form the final answer. With the same strategy we can calculate other examples. Table 5.4 shows some result that we have achieved by the expriment. For second example we keep the temperature as 20˚C and change the humidity from 20 ppm to 80 ppm. The table shows that there is no change for heater but humidifier changes from 0.756 to 0.244. This means that since humidity is increased for compensating this amount of humidity the motor is working with minimum speed. There is also a little change for fan from 0.191 to

0.274. This shows that the fan has a tendency to work between minimim speed and normal speed instead of working with the minimum speed. The MATLAB Code for given problem can be seen in Appendix B.

Table 5.4 Results Based on the Temp & Humidity for Contorlling Outputs

	Temperature	**Humidity**	**Fan**	**Heater**	**Humidifier**
1	20	20	0.191	0.587	0.756
2	20	80	0.274	0.587	0.244
3	80	20	0.765	0.0876	0.756
4	80	80	0.596	0.0876	0.244
5	40	30	0.244	0.5	0.69
6	30	40	0.203	0.595	0.605

Chapter 6

CONCLUSION

6.1 Conclusion

In this thesis the design of an intelligent decision making architecture has been presented for remotely stationed solar-powered wireless communication systems. The subject was centered around designing an intelligent decision making system based on the data collected by a wireless sensor network. Deploying a wireless sensor network allows us to have more freedom in the overall architecture thereby allowing a real-time response for emergency actions. The WSN architecture results in a more manageable system size with a decrease in the total cost and ease in installation. In contrast to wired sensors accessed through a centralized monitor, the presented intelligent decision making system is self-organizing and calibrating automatically which can also be reconfigured easily for new applications. Our system consumes less power and can be installed with less cost when with the existing ones. Considering all these factors make a WSN ideal for most applications. The disadvantage of a WSN is that it has a lower speed compared to a wired network. Each of the intelligent decision methods introduced has its own benefits, drawbacks and difficulties, but the method that we used in our real application was not very complicated and has fewer disadvantages in comparison to other methods. In addition, FLS has a faster processing speed in contrast to a computational system.

The only drawback of this method is that it may have increased faults in decision-making when the system is not fully functional.

6.2 Further Work

This project has a great capacity for operation, modification and updating both in hardware and software implementation. The AVR microcontroller language program is C++, therefore, we need to write a C program that is able to convert FLS Command to a format that is readable by AVR. After all, we should test the whole system, and modify and re-test those components that need an attention. The last step is to calculate power dissipation and evaluate if the cost and efficiency can be optimized further.

In Chapter five we have observed that for deploying multiple sensors in multiple tasks the number of rules increase drastically. This means that for some applications we may need a million of rules, which make in hard for realization. We define a new powerful software Automatic Workload Repository (AWR) for visual system simulation. AWR Corporation is the software solution for the given structure of the wireless sensor networks. Using this software in the decision making process will have a huge potential for increased reliability, faster production cycle and significant reduction in cost.

REFERENCES

[1] C. Townsend & S. Arms, "*Wireless Sensor Networks: Principles and Applications*," Sensor Technology Handbook, 2005.

[2] S. Uysal, M. Konca, H. Demirel, & A. Hocanin, "An Intelligent Homeland Security Surveillance System (IHSSS)-Tx/Rx Architecture," *Proceedings of MMS2010*, METU-NCC, pp. 284-287, August 2010.

[3] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam & E. Cayirci, "Wireless Sensor Networks: a Survey," *Computer Networks (Elsevier)*, Vol. 38, pp. 393-422, December 2001.

[4] V. Shnayder, M. Hempstead, B. Chen, G.W. Allen & M. Welsh, "Simulating the Power Consumption of Large Scale Sensor Network Applications," *SenSys'04*, Baltimore, Maryland, USA, 2004.

[5] Q. Wu, M. Zhu & N.S.V. Rao, "Integration of Sensing and Computing in an Intelligent Decision Support," *Pervasive and Mobile Computing (Elsevier),* Vol. 5, pp. 182–200, 2009.

[6] V. Katiyar, P. Kumar & N. Chand, "An Intelligent Transportation System Architecture using Wireless Sensor Networks," *International Journal of Computer Applications*, Vol. 14, No. 2, pp. 22-26, January 2011.

[7] T. Wei, Y. Huang & P. Chen, "Particle Filtering For Adaptive Sensor Fault Detection and Identification," *International Conference on Robotics and Automation (IEEE)*, pp. 3807 – 3812, May 2006.

[8] J.Q. Zhang & Y. Yan, "Online Validation of the Measurement Uncertainty of a Sensor Using Wavelet Transforms," *Journal of Science, Measurement and Technology*, Vol. 148, No. 5, pp. 210 – 214, September 2011.

[9] A.M. Madni, P. Sridhar & M. Jamshidi, "Fault-Tolerant Data Acquisition in Sensor Networks," *International Conference of system of systems engineering (IEEE*), 2007.

[10] P. Sridhar & A.M. Madni, "Scalability and Performance Issues in Deeply Embedded Sensor," *International Journal on Smart Sensing and Embedded Sensor*, Vol. 2, No. 1, pp. 1-14, March 2009.

[11] S. Sendra, J. Lloret, M. García & J.F. Toledo, "Power saving and energy optimization technique for Wireless Sensor Networks," *Journal of Communications*, Vol. 6, No. 6, pp. 439-459, September 2011.

[12] S. Jafarizadeh & A. Jamalipour, "Fastest Distributed Consensus on Star-Mesh Hybrid Sensor Networks," *Sensors Journal* (*IEEE)*, Vol. 11, No. 10, pp. 2494-2506, October 2011.

[13] S. Yun, J. Lee, W. Chung, E. Kim, & S. Kim, "A Soft Computing Approach to Localization in Wireless Sensor Networks," *Expert Systems with Applications* (*Elsevier*), Vol. 36, pp. 7552–7561, 2006.

[14] H. Abdi, "A Neural Network Primer," Journal of Biological Systems, Vol. 2, No. 3, pp. 247-283, 1994.

[15] W.R. Heinzelman, J. Kulik & H. Balakrishnan, "Adaptive Protocols for Information Dissemination in Wireless Sensor Networks," *Proc. ACM MobiCom 99*, Seattle, WA, pp. 174–85, 1999.

[16] M. Frean, "The Upstart Algorithm a Method for Constructing and Training Feed Forward Neural Networks," *Neural Computation*, Vol. 2, pp. 198-209, 1990.

[17] S. Alizadeh & M. Shamsadini,"Genetic Algorithms," *IJCSNS International Journal of Computer Science and Network Security*, Vol. 11, No. 4, pp. 74-76, April 2011.

[18] "*Greenhouse Horticulture*," Computer Control Systems, Sensors and Monitoring Equipment in Greenhause, State of New South Wales, Retrieved from http://www.dpi.nsw.gov.au/

[19] "*Genetic Algorithm Basic Description*," 1998, retrieved from http://www.obitko.com/tutorials/genetic-algorithms/

[20] Fuzzy Logic Toolbox "*Sugeno-Type Fuzzy Inference*", The MathWorks, Inc. 2011, retrieved from http://www.mathworks.com/help/toolbox/fuzzy/

[21] M.D. Prokic, "*Low-Rate Wireless Personal Area Network Coordinator Design and Implementation*," Master Thesis, Department of Electrical and Computer Engineering McGill University, August 2005.

[22] Stankovic, J. "*Wireless Sensor Networks*" Department of Computer Science, University of Virginia Charlottesville, Virginia, June 2006.

[23] F.L. Lewis, "*Wireless Sensor Networks*" In Smart Environments: Technologies, Protocols, Applications, ed Cook, D.J., Das, S.K., & Wiley, J. New York, 2004.

[24] DS1307 64 x 8, Serial, I2C Real-Time Clock. Maxim Integrated Products. Dallas Semiconductor. 2004.

[25] ATMEL, 8-bit AVR with 8Kbytes In-System Programmable Flash. ATmega8.

[26] DS18S20, High-Precision 1-Wire Digital Thermometer. Maxim Integrated Products.2010.

[27] MC14562B 128-Bit Static Shift Register. Semiconductor Components Industries, LLC. August 2000.

[28] Future Technology Devices International Ltd. UB232R. USB Mini-B FT232R Evaluation Module Datasheet. Document Reference No. FT_000055 Version 1.12. 2010.

[29] T.J. Meitzler & E. Sohn, "A Comparison of Mamdani and Sugeno Methods for Modeling Visual Perception Lab Data," *IEEE Annual Meeting of the North American Fuzzy Information Processing Society*, pp. 116-120, 2005.

[30] G. Anastasi, M. Conti, A. Falchi, E. Gregori & A. Passarella, "*Performance Measurements of Mote Sensor Networks*," Department of Information Engineering, University of Pisa, Venezia, Italy.

[31] P. Manjunatha, A.K. Verna & A. Srividya, "Multi-Sensor Data Fusion in Cluster based Wireless Sensor Networks Using Fuzzy Logic Method," *IEEE Kharagpur Section & IEEE Sri Lanka Section, International Conference on Industrial and Information Systems*, December 2008.

[32] F. Lafont & J. Balmat, "Fuzzy Sets and Systems: Optimized Fuzzy Control of A Greenhouse," *Fuzzy Sets and Systems (Elsevier)*, Vol. 128, pp. 47–59, 2002.

[33] L.B. Tik, C.T. Khuan & S. Palaniappan, "Monitoring of an Aeroponic Greenhouse with a Sensor Network," *International Journal of Computer Science and Network Security*, Vol. 9, No. 3, pp. 240-246, March 2009.

[34] N. Sabri, S.A. Alijunid, R.B. Ahmad, A. Yahya, R. Kamaruddin & M.S. Salim, "Wireless Sensor Actor Network Based on Fuzzy Inference System for Greenhouse Climate Control," *Journal of Applied Sciences*, Vol. 11, pp. 3104-3116, 2011.

APPENDICES

Appendix A: Rules of Fuzzy Logic In the Case of Fire

Table A.1: Rules of Fuzzy Logic In the Case of Fire

Rule	Temperature	Humidity	Light	CO	Distance	Output
1	VL	H	L	L	Far	VL
2	VL	H	L	L	Close	VL
3	VL	H	L	M	Far	VL
4	VL	H	L	M	Close	L
5	VL	H	L	H	Far	L
6	VL	H	L	H	Close	VL
7	VL	H	M	L	Far	VL
8	VL	H	M	L	Close	L
9	VL	H	M	M	Far	L
10	VL	H	M	M	Close	M
11	VL	H	M	H	Far	L
12	VL	H	M	H	Close	M
13	VL	H	H	L	Far	L
14	VL	H	H	L	Close	M
15	VL	H	H	M	Far	M
16	VL	H	H	M	Close	M
17	VL	H	H	H	Far	M
18	VL	H	H	H	Close	H
19	VL	M	L	L	Far	VL
20	VL	M	L	L	Close	L
21	VL	M	L	M	Far	L
22	VL	M	L	M	Close	L
23	VL	M	L	H	Far	L
24	VL	M	L	H	Close	M
25	VL	M	M	L	Far	L
26	VL	M	M	L	Close	L
27	VL	M	M	M	Far	M
28	VL	M	M	M	Close	M
29	VL	M	M	H	Far	M
30	VL	M	M	H	Close	H
31	VL	M	H	L	Far	L
32	VL	M	H	L	Close	L
33	VL	M	H	M	Far	L
34	VL	M	H	M	Close	M
35	VL	M	H	H	Far	M
36	VL	M	H	H	Close	H
37	VL	L	L	L	Far	VL
38	VL	L	L	L	Close	VL
39	VL	L	L	M	Far	L
40	VL	L	L	M	Close	L
41	VL	L	L	H	Far	M
42	VL	L	L	H	Close	M
43	VL	L	M	L	Far	L

44	VL	L	M	L	Close	M
45	VL	L	M	M	Far	M
46	VL	L	M	M	Close	M
47	VL	L	M	H	Far	M
48	VL	L	M	H	Close	H
49	VL	L	H	L	Far	M
50	VL	L	H	L	Close	H
51	VL	L	H	M	Far	M
52	VL	L	H	M	Close	H
53	VL	L	H	H	Far	M
54	VL	L	H	H	Close	H
55	L	H	L	L	Far	VL
56	L	H	L	L	Close	L
57	L	H	L	M	Far	VL
58	L	H	L	M	Close	L
59	L	H	L	H	Far	L
60	L	H	L	H	Close	L
61	L	H	M	L	Far	L
62	L	H	M	L	Close	M
63	L	H	M	M	Far	L
64	L	H	M	M	Close	M
65	L	H	M	H	Far	M
66	L	H	M	H	Close	M
67	L	H	H	L	Far	L
68	L	H	H	L	Close	M
69	L	H	H	M	Far	L
70	L	H	H	M	Close	M
71	L	H	H	H	Far	M
72	L	H	H	H	Close	H
73	L	M	L	L	Far	L
74	L	M	L	L	Close	L
75	L	M	L	M	Far	L
76	L	M	L	M	Close	M
77	L	M	L	H	Far	M
78	L	M	L	H	Close	M
79	L	M	M	L	Far	L
80	L	M	M	L	Close	L
81	L	M	M	M	Far	L
82	L	M	M	M	Close	M
83	L	M	M	H	Far	M
84	L	M	M	H	Close	H
85	L	M	H	L	Far	L
86	L	M	H	L	Close	L
87	L	M	H	M	Far	L
88	L	M	H	M	Close	M
89	L	M	H	H	Far	M
90	L	M	H	H	Close	H

91	L	L	L	L	Far	L
92	L	L	L	L	Close	L
93	L	L	L	M	Far	L
94	L	L	L	M	Close	M
95	L	L	L	H	Far	M
96	L	L	L	H	Close	M
97	L	L	M	L	Far	L
98	L	L	M	L	Close	M
99	L	L	M	M	Far	M
100	L	L	M	M	Close	H
101	L	L	M	H	Far	M
102	L	L	M	H	Close	H
103	L	L	H	L	Far	L
104	L	L	H	L	Close	L
105	L	L	H	M	Far	L
106	L	L	H	M	Close	M
107	L	L	H	H	Far	M
108	L	L	H	H	Close	H
109	M	F	L	L	Far	L
110	M	F	L	L	Close	M
111	M	F	L	M	Far	L
112	M	F	L	M	Close	M
113	M	F	L	H	Far	M
114	M	F	L	H	Close	H
115	M	F	M	L	Far	L
116	M	F	M	L	Close	M
117	M	F	M	M	Far	M
118	M	F	M	M	Close	M
119	M	F	M	H	Far	M
120	M	F	M	H	Close	H
121	M	F	H	L	Far	M
122	M	F	H	L	Close	M
123	M	F	H	M	Far	M
124	M	F	H	M	Close	H
125	M	F	H	H	Far	M
126	M	F	H	H	Close	H
127	M	M	L	L	Far	L
128	M	M	L	L	Close	M
129	M	M	L	M	Far	M
130	M	M	L	M	Close	H
131	M	M	L	H	Far	M
132	M	M	L	H	Close	H
133	M	M	M	L	Far	M
134	M	M	M	L	Close	H
135	M	M	M	M	Far	M
136	M	M	M	M	Close	H
137	M	M	M	H	Far	H

138	M	M	M	H	Close	VH
139	M	M	H	L	Far	M
140	M	M	H	L	Close	H
141	M	M	H	M	Far	H
142	M	M	H	M	Close	H
143	M	M	H	H	Far	H
144	M	M	H	H	Close	VH
145	M	L	L	L	Far	L
146	M	L	L	L	Close	L
147	M	L	L	M	Far	M
148	M	L	L	M	Close	M
149	M	L	L	H	Far	M
150	M	L	L	H	Close	H
151	M	L	M	L	Far	M
152	M	L	M	L	Close	H
153	M	L	M	M	Far	H
154	M	L	M	M	Close	VH
155	M	L	M	H	Far	H
156	M	L	M	H	Close	VH
157	M	L	H	L	Far	M
158	M	L	H	L	Close	H
159	M	L	H	M	Far	H
160	M	L	H	M	Close	H
161	M	L	H	H	Far	H
162	M	L	H	H	Close	VH
163	H	H	L	L	Far	M
164	H	H	L	L	Close	H
165	H	H	L	M	Far	M
166	H	H	L	M	Close	H
167	H	H	L	H	Far	H
168	H	H	L	H	Close	H
169	H	H	M	L	Far	M
170	H	H	M	L	Close	H
171	H	H	M	M	Far	H
172	H	H	M	M	Close	H
173	H	H	M	H	Far	H
174	H	H	M	H	Close	VH
175	H	H	H	L	Far	M
176	H	H	H	L	Close	H
177	H	H	H	M	Far	H
178	H	H	H	M	Close	H
179	H	H	H	H	Far	H
180	H	H	H	H	Close	VH
181	H	M	L	L	Far	M
182	H	M	L	L	Close	H
183	H	M	L	M	Far	H
184	H	M	L	M	Close	H

185	H	M	L	H	Far	H
186	H	M	L	H	Close	VH
187	H	M	M	L	Far	H
188	H	M	M	L	Close	H
189	H	M	M	M	Far	H
190	H	M	M	M	Close	VH
191	H	M	M	H	Far	H
192	H	M	M	H	Close	VH
193	H	M	H	L	Far	H
194	H	M	H	L	Close	H
195	H	M	H	M	Far	H
196	H	M	H	M	Close	VH
197	H	M	H	H	Far	VH
198	H	M	H	H	Close	VH
199	H	L	L	L	Far	M
200	H	L	L	L	Close	M
201	H	L	L	M	Far	M
202	H	L	L	M	Close	H
203	H	L	L	H	Far	H
204	H	L	L	H	Close	VH
205	H	L	M	L	Far	H
206	H	L	M	L	Close	H
207	H	L	M	M	Far	H
208	H	L	M	M	Close	VH
209	H	L	M	H	Far	H
210	H	L	M	H	Close	VH
211	H	L	H	L	Far	H
212	H	L	H	L	Close	VH
213	H	L	H	M	Far	H
214	H	L	H	M	Close	VH
215	H	L	H	H	Far	VH
216	H	L	H	H	Close	VH
217	VH	H	L	L	Far	H
218	VH	H	L	L	Close	H
219	VH	H	L	M	Far	H
220	VH	H	L	M	Close	VH
221	VH	H	L	H	Far	VH
222	VH	H	L	H	Close	VH
223	VH	H	M	L	Far	H
224	VH	H	M	L	Close	VH
225	VH	H	M	M	Far	H
226	VH	H	M	M	Close	H
227	VH	H	M	H	Far	VH
228	VH	H	M	H	Close	VH
229	VH	H	H	L	Far	H
230	VH	H	H	L	Close	H
231	VH	H	H	M	Far	H

232	VH	H	H	M	Close	VH
233	VH	H	H	H	Far	VH
234	VH	H	H	H	Close	VH
235	VH	M	L	L	Far	H
236	VH	M	L	L	Close	H
237	VH	M	L	M	Far	H
238	VH	M	L	M	Close	VH
239	VH	M	L	H	Far	VH
240	VH	M	L	H	Close	VH
241	VH	M	M	L	Far	H
242	VH	M	M	L	Close	VH
243	VH	M	M	M	Far	H
244	VH	M	M	M	Close	VH
245	VH	M	M	H	Far	VH
246	VH	M	M	H	Close	VH
247	VH	M	H	L	Far	H
248	VH	M	H	L	Close	H
249	VH	M	H	M	Far	H
250	VH	M	H	M	Close	VH
251	VH	M	H	H	Far	VH
252	VH	M	H	H	Close	VH
253	VH	L	L	L	Far	H
254	VH	L	L	L	Close	VH
255	VH	L	L	M	Far	VH
256	VH	L	L	M	Close	VH
257	VH	L	L	H	Far	VH
258	VH	L	L	H	Close	VH
259	VH	L	M	L	Far	H
260	VH	L	M	L	Close	H
261	VH	L	M	M	Far	H
262	VH	L	M	M	Close	H
263	VH	L	M	H	Far	VH
264	VH	L	M	H	Close	VH
265	VH	L	H	L	Far	H
266	VH	L	H	L	Close	VH
267	VH	L	H	M	Far	VH
268	VH	L	H	M	Close	VH
269	VH	L	H	H	Far	VH
270	VH	L	H	H	Close	VH

Appendix B: MATLAB Code for Probability of Fire

```
Name='Pouya-bolodorost shode8'

Type='mamdani'

Version=2.0

NumInputs=5

NumOutputs=1

NumRules=270

AndMethod='min'

OrMethod='max'

ImpMethod='min'

AggMethod='max'

DefuzzMethod='centroid'


[Input1]

Name='Temperature-(C)'

Range=[-20 150]

NumMFs=5

MF1='Very-Low':'trimf',[-62.5 -20 22.5]

MF2='Low':'trimf',[-3.35988359788359 39.1021164021164 81.6021164021164]

MF3='Medium':'trimf',[22.5 65 107.5]

MF4='High':'trimf',[65 107.5 150]

MF5='Very-High':'trimf',[107.5 150 192.5]


[Input2]

Name='Humidity-(ppm)'

Range=[-100 0]

NumMFs=3

MF1='High':'trimf',[-140 -100 -60]

MF2='Medium':'trimf',[-90 -50 -10]
```

MF3='Low':'trimf',[-40 0 40]

[Input3]

Name='Light-(Lux)'

Range=[0 1000]

NumMFs=3

MF1='Low':'trimf',[-400 0 400]

MF2='Medium':'trimf',[100 500 900]

MF3='High':'trimf',[600 1000 1400]

[Input4]

Name='CO-(ppm)'

Range=[0 100]

NumMFs=3

MF1='Low':'trimf',[-40 0 40]

MF2='Medium':'trimf',[10 50 90]

MF3='High':'trimf',[60 100 140]

[Input5]

Name='Distance-(m)'

Range=[-120 0]

NumMFs=2

MF1='Far':'trimf',[-240 -120 0]

MF2='Close':'trimf',[-120 0 120]

[Output1]

Name='Threat-of-Fire'

Range=[0 100]

NumMFs=5

MF1='Very-Low':'trapmf',[-3.5 0.132275132275133 11.5 23.5]

MF2='Low':'trimf',[15 30 45]

MF3='Medium':'trimf',[35.5291005291005 50.5291005291005 65.5291005291005]

MF4='High':'trimf',[55 70 85]

MF5='Very-High':'trapmf',[76.5 88.5 99.6031746031746 113]

[Rules]

1 1 1 1 1, 1 (1) : 1

1 1 1 1 2, 1 (1) : 1

1 1 1 2 1, 1 (1) : 1

1 1 1 2 2, 1 (1) : 1

1 1 1 3 1, 1 (1) : 1

1 1 1 3 2, 2 (1) : 1

1 1 2 1 1, 1 (1) : 1

1 1 2 1 2, 1 (1) : 1

1 1 2 2 1, 1 (1) : 1

1 1 2 2 2, 2 (1) : 1

1 1 2 3 1, 2 (1) : 1

1 1 2 3 2, 2 (1) : 1

1 1 3 1 1, 1 (1) : 1

1 1 3 1 2, 2 (1) : 1

1 1 3 2 1, 1 (1) : 1

1 1 3 2 2, 2 (1) : 1

1 1 3 3 1, 2 (1) : 1

1 1 3 3 2, 3 (1) : 1

1 2 1 1 1, 1 (1) : 1

1 2 1 1 2, 1 (1) : 1

1 2 1 2 1, 1 (1) : 1

1 2 1 2 2, 2 (1) : 1

1 2 1 3 1, 2 (1) : 1

1 2 1 3 2, 2 (1) : 1

1 2 2 1 1, 1 (1) : 1

1 2 2 1 2, 2 (1) : 1

1 2 2 2 1, 1 (1) : 1

1 2 2 2 2, 2 (1) : 1

1 2 2 3 1, 1 (1) : 1

1 2 2 3 2, 2 (1) : 1

1 2 3 1 1, 2 (1) : 1

1 2 3 1 2, 2 (1) : 1

1 2 3 2 1, 2 (1) : 1

1 2 3 2 2, 2 (1) : 1

1 2 3 3 1, 2 (1) : 1

1 2 3 3 2, 2 (1) : 1

1 3 1 1 1, 1 (1) : 1

1 3 1 1 2, 1 (1) : 1

1 3 1 2 1, 2 (1) : 1

1 3 1 2 2, 2 (1) : 1

1 3 1 3 1, 1 (1) : 1

1 3 1 3 2, 2 (1) : 1

1 3 2 1 1, 2 (1) : 1

1 3 2 1 2, 2 (1) : 1

1 3 2 2 1, 1 (1) : 1

1 3 2 2 2, 2 (1) : 1

1 3 2 3 1, 2 (1) : 1

1 3 2 3 2, 2 (1) : 1

1 3 3 1 1, 2 (1) : 1

1 3 3 1 2, 2 (1) : 1

1 3 3 2 1, 2 (1) : 1

1 3 3 2 2, 2 (1) : 1

1 3 3 3 1, 2 (1) : 1

1 3 3 3 2, 2 (1) : 1

2 1 1 1 1, 1 (1) : 1

2 1 1 1 2, 2 (1) : 1

2 1 1 2 1, 1 (1) : 1

2 1 1 2 2, 2 (1) : 1

2 1 1 3 1, 2 (1) : 1

2 1 1 3 2, 2 (1) : 1

2 1 2 1 1, 2 (1) : 1

2 1 2 1 2, 3 (1) : 1

2 1 2 2 1, 2 (1) : 1

2 1 2 2 2, 3 (1) : 1

2 1 2 3 1, 3 (1) : 1

2 1 2 3 2, 3 (1) : 1

2 1 3 1 1, 2 (1) : 1

2 1 3 1 2, 3 (1) : 1

2 1 3 2 1, 2 (1) : 1

2 1 3 2 2, 3 (1) : 1

2 1 3 3 1, 3 (1) : 1

2 1 3 3 2, 3 (1) : 1

2 2 1 1 1, 2 (1) : 1

2 2 1 1 2, 2 (1) : 1

2 2 1 2 1, 2 (1) : 1

2 2 1 2 2, 3 (1) : 1

2 2 1 3 1, 3 (1) : 1

2 2 1 3 2, 3 (1) : 1

2 2 2 1 1, 2 (1) : 1

2 2 2 1 2, 2 (1) : 1

2 2 2 2 1, 2 (1) : 1

2 2 2 2 2, 3 (1) : 1

2 2 2 3 1, 3 (1) : 1

2 2 2 3 2, 4 (1) : 1

2 2 3 1 1, 2 (1) : 1

2 2 3 1 2, 2 (1) : 1

2 2 3 2 1, 2 (1) : 1

2 2 3 2 2, 3 (1) : 1

2 2 3 3 1, 3 (1) : 1

2 2 3 3 2, 4 (1) : 1

2 3 1 1 1, 2 (1) : 1

2 3 1 1 2, 2 (1) : 1

2 3 1 2 1, 2 (1) : 1

2 3 1 2 2, 3 (1) : 1

2 3 1 3 1, 3 (1) : 1

2 3 1 3 2, 3 (1) : 1

2 3 2 1 1, 2 (1) : 1

2 3 2 1 2, 3 (1) : 1

2 3 2 2 1, 3 (1) : 1

2 3 2 2 2, 4 (1) : 1

2 3 2 3 1, 3 (1) : 1

2 3 2 3 2, 4 (1) : 1

2 3 3 1 1, 2 (1) : 1

2 3 3 1 2, 2 (1) : 1

2 3 3 2 1, 2 (1) : 1

2 3 3 2 2, 3 (1) : 1

2 3 3 3 1, 3 (1) : 1

2 3 3 3 2, 4 (1) : 1

3 1 1 1 1, 2 (1) : 1

3 1 1 1 2, 3 (1) : 1

3 1 1 2 1, 2 (1) : 1

3 1 1 2 2, 3 (1) : 1

3 1 1 3 1, 3 (1) : 1

3 1 1 3 2, 4 (1) : 1

3 1 2 1 1, 2 (1) : 1

3 1 2 1 2, 3 (1) : 1

3 1 2 2 1, 3 (1) : 1

3 1 2 2 2, 3 (1) : 1

3 1 2 3 1, 3 (1) : 1

3 1 2 3 2, 4 (1) : 1

3 1 3 1 1, 3 (1) : 1

3 1 3 1 2, 3 (1) : 1

3 1 3 2 1, 3 (1) : 1

3 1 3 2 2, 4 (1) : 1

3 1 3 3 1, 3 (1) : 1

3 1 3 3 2, 4 (1) : 1

3 2 1 1 1, 2 (1) : 1

3 2 1 1 2, 3 (1) : 1

3 2 1 2 1, 3 (1) : 1

3 2 1 2 2, 4 (1) : 1

3 2 1 3 1, 3 (1) : 1

3 2 1 3 2, 4 (1) : 1

3 2 2 1 1, 3 (1) : 1

3 2 2 1 2, 4 (1) : 1

3 2 2 2 1, 3 (1) : 1

3 2 2 2 2, 4 (1) : 1

3 2 2 3 1, 4 (1) : 1

3 2 2 3 2, 5 (1) : 1

3 2 3 1 1, 3 (1) : 1

3 2 3 1 2, 4 (1) : 1

3 2 3 2 1, 4 (1) : 1

3 2 3 2 2, 4 (1) : 1

3 2 3 3 1, 4 (1) : 1

3 2 3 3 2, 5 (1) : 1

3 3 1 1 1, 2 (1) : 1

3 3 1 1 2, 2 (1) : 1

3 3 1 2 1, 3 (1) : 1

3 3 1 2 2, 3 (1) : 1

3 3 1 3 1, 3 (1) : 1

3 3 1 3 2, 4 (1) : 1

3 3 2 1 1, 3 (1) : 1

3 3 2 1 2, 4 (1) : 1

3 3 2 2 1, 4 (1) : 1

3 3 2 2 2, 5 (1) : 1

3 3 2 3 1, 4 (1) : 1

3 3 2 3 2, 5 (1) : 1

3 3 3 1 1, 3 (1) : 1

3 3 3 1 2, 4 (1) : 1

3 3 3 2 1, 4 (1) : 1

3 3 3 2 2, 4 (1) : 1

3 3 3 3 1, 4 (1) : 1

3 3 3 3 2, 5 (1) : 1

4 1 1 1 1, 3 (1) : 1

4 1 1 1 2, 4 (1) : 1

4 1 1 2 1, 3 (1) : 1

4 1 1 2 2, 4 (1) : 1

4 1 1 3 1, 4 (1) : 1

4 1 1 3 2, 4 (1) : 1

4 1 2 1 1, 3 (1) : 1

4 1 2 1 2, 4 (1) : 1

4 1 2 2 1, 4 (1) : 1

4 1 2 2 2, 4 (1) : 1

4 1 2 3 1, 4 (1) : 1

4 1 2 3 2, 4 (1) : 1

4 1 3 1 1, 4 (1) : 1

4 1 3 1 2, 4 (1) : 1

4 1 3 2 1, 4 (1) : 1

4 1 3 2 2, 4 (1) : 1

4 1 3 3 1, 4 (1) : 1

4 1 3 3 2, 5 (1) : 1

4 2 1 1 1, 3 (1) : 1

4 2 1 1 2, 4 (1) : 1

4 2 1 2 1, 4 (1) : 1

4 2 1 2 2, 4 (1) : 1

4 2 1 3 1, 4 (1) : 1

4 2 1 3 2, 5 (1) : 1

4 2 2 1 1, 4 (1) : 1

4 2 2 1 2, 4 (1) : 1

4 2 2 2 1, 4 (1) : 1

4 2 2 2 2, 5 (1) : 1

4 2 2 3 1, 4 (1) : 1

4 2 2 3 2, 5 (1) : 1

4 2 3 1 1, 4 (1) : 1

4 2 3 1 2, 4 (1) : 1

4 2 3 2 1, 4 (1) : 1

4 2 3 2 2, 5 (1) : 1

4 2 3 3 1, 5 (1) : 1

4 2 3 3 2, 5 (1) : 1

4 3 1 1 1, 3 (1) : 1

4 3 1 1 2, 3 (1) : 1

4 3 1 2 1, 3 (1) : 1

4 3 1 2 2, 4 (1) : 1

4 3 1 3 1, 4 (1) : 1

4 3 1 3 2, 5 (1) : 1

4 3 2 1 1, 4 (1) : 1

4 3 2 1 2, 4 (1) : 1

4 3 2 2 1, 4 (1) : 1

4 3 2 2 2, 5 (1) : 1

4 3 2 3 1, 4 (1) : 1

4 3 2 3 2, 5 (1) : 1

4 3 3 1 1, 4 (1) : 1

4 3 3 1 2, 5 (1) : 1

4 3 3 2 1, 4 (1) : 1

4 3 3 2 2, 5 (1) : 1

4 3 3 3 1, 5 (1) : 1

4 3 3 3 2, 5 (1) : 1

5 1 1 1 1, 4 (1) : 1

5 1 1 1 2, 4 (1) : 1

5 1 1 2 1, 4 (1) : 1

5 1 1 2 2, 5 (1) : 1

5 1 1 3 1, 5 (1) : 1

5 1 1 3 2, 5 (1) : 1

5 1 2 1 1, 4 (1) : 1

5 1 2 1 2, 5 (1) : 1

5 1 2 2 1, 4 (1) : 1

5 1 2 2 2, 4 (1) : 1

5 1 2 3 1, 5 (1) : 1

5 1 2 3 2, 5 (1) : 1

5 1 3 1 1, 4 (1) : 1

5 1 3 1 2, 4 (1) : 1

5 1 3 2 1, 4 (1) : 1

5 1 3 2 2, 5 (1) : 1

5 1 3 3 1, 5 (1) : 1

5 1 3 3 2, 5 (1) : 1

5 2 1 1 1, 4 (1) : 1

5 2 1 1 2, 4 (1) : 1

5 2 1 2 1, 4 (1) : 1

5 2 1 2 2, 5 (1) : 1

5 2 1 3 1, 5 (1) : 1

5 2 1 3 2, 5 (1) : 1

5 2 2 1 1, 4 (1) : 1

5 2 2 1 2, 5 (1) : 1

5 2 2 2 1, 4 (1) : 1

5 2 2 2 2, 5 (1) : 1

5 2 2 3 1, 5 (1) : 1

5 2 2 3 2, 5 (1) : 1

5 2 3 1 1, 4 (1) : 1

5 2 3 1 2, 4 (1) : 1

5 2 3 2 1, 4 (1) : 1

5 2 3 2 2, 5 (1) : 1

5 2 3 3 1, 5 (1) : 1

5 2 3 3 2, 5 (1) : 1

5 3 1 1 1, 4 (1) : 1

5 3 1 1 2, 5 (1) : 1

5 3 1 2 1, 5 (1) : 1

5 3 1 2 2, 5 (1) : 1

5 3 1 3 1, 5 (1) : 1

5 3 1 3 2, 5 (1) : 1

5 3 2 1 1, 4 (1) : 1

5 3 2 1 2, 4 (1) : 1

5 3 2 2 1, 4 (1) : 1

5 3 2 2 2, 4 (1) : 1

5 3 2 3 1, 5 (1) : 1

5 3 2 3 2, 5 (1) : 1

5 3 3 1 1, 4 (1) : 1

5 3 3 1 2, 5 (1) : 1

5 3 3 2 1, 5 (1) : 1

5 3 3 2 2, 5 (1) : 1

5 3 3 3 1, 5 (1) : 1

5 3 3 3 2, 5 (1) : 1

Appendix C: MATLAB Code for Controlling Greenhouse

```
Name='popo'
Type='mamdani'
Version=2.0
NumInputs=3
NumOutputs=4
NumRules=28
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='centroid'

[Input1]
Name='Temperature'
Range=[-20 100]
NumMFs=5
MF1='Very-Low':'trimf',[-50 -20 10]
MF2='Low':'trimf',[-20 10 40]
MF3='Medium':'trimf',[10 40 70]
MF4='High':'trimf',[40 70 100]
MF5='Vey-High':'trimf',[70 100 130]

[Input2]
Name='Humidity'
Range=[0 100]
NumMFs=5
MF1='Very-Low':'trimf',[-25 0 25]
MF2='Low':'trimf',[0 25 50]
```

MF3='Medium':'trimf',[25 50 75]

MF4='High':'trimf',[50 75 100]

MF5='Very-High':'trimf',[75 100 125]

[Input3]

Name='Light'

Range=[0 1000]

NumMFs=3

MF1='Low':'trimf',[-400 0 400]

MF2='Medium':'trimf',[100 500 900]

MF3='High':'trimf',[600 1000 1400]

[Output1]

Name='Fan'

Range=[0 1]

NumMFs=5

MF1='Off':'trimf',[-0.25 0 0.25]

MF2='Minimum':'trimf',[0 0.25 0.5]

MF3='Normal':'trimf',[0.25 0.5 0.75]

MF4='High':'trimf',[0.5 0.75 1]

MF5='Maximum':'trimf',[0.75 1 1.25]

[Output2]

Name='Heater'

Range=[0 1]

NumMFs=5

MF1='Off':'trimf',[-0.25 0 0.25]

MF2='Minimum':'trimf',[0 0.25 0.5]

MF3='Normal':'trimf',[0.25 0.5 0.75]

MF4='High':'trimf',[0.5 0.75 1]

MF5='Maxiumum':'trimf',[0.75 1 1.25]

[Output3]

Name='Humidifier'

Range=[0 1]

NumMFs=5

MF1='Off':'trimf',[-0.25 0 0.25]

MF2='Minimum':'trimf',[0 0.25 0.5]

MF3='Normal':'trimf',[0.25 0.5 0.75]

MF4='High':'trimf',[0.5 0.75 1]

MF5='Maximum':'trimf',[0.75 1 1.25]

[Output4]

Name='Light_Controller'

Range=[0 1]

NumMFs=3

MF1='Off':'trimf',[-0.4 0 0.4]

MF2='Normal':'trimf',[0.1 0.5 0.9]

MF3='On':'trimf',[0.6 1 1.4]

[Rules]

0 0 1, 0 0 0 3 (1) : 1

0 0 2, 0 0 0 2 (1) : 1

0 0 3, 0 0 0 1 (1) : 1

1 1 0, 1 5 5 0 (1) : 1`1

1 2 0, 1 5 4 0 (1) : 1

1 3 0, 1 5 3 0 (1) : 1

1 4 0, 1 5 2 0 (1) : 1

1 5 0, 1 1 1 0 (1) : 1

2 1 0, 1 4 5 0 (1) : 1

2 2 0, 1 4 4 0 (1) : 1

2 3 0, 1 4 3 0 (1) : 1

2 4 0, 1 4 2 0 (1) : 1

2 5 0, 1 1 1 0 (1) : 1

3 1 0, 2 2 5 0 (1) : 1

3 2 0, 2 3 4 0 (1) : 1

3 3 0, 1 3 3 0 (1) : 1

3 4 0, 2 3 2 0 (1) : 1

3 5 0, 3 1 1 0 (1) : 1

4 1 0, 4 1 5 0 (1) : 1

4 2 0, 4 1 4 0 (1) : 1

4 3 0, 3 1 3 0 (1) : 1

4 4 0, 3 1 2 0 (1) : 1

4 5 0, 3 1 1 0 (1) : 1

5 1 0, 5 1 5 0 (1) : 1

5 2 0, 5 1 4 0 (1) : 1

5 3 0, 5 1 3 0 (1) : 1

5 4 0, 5 1 2 0 (1) : 1

5 5 0, 5 1 1 0 (1) : 1