

2015

A Multi Agent System Design for Power Distribution Restoration Using Neural Networks

Mohamad Mashta

Follow this and additional works at: <https://researchrepository.wvu.edu/etd>

Recommended Citation

Mashta, Mohamad, "A Multi Agent System Design for Power Distribution Restoration Using Neural Networks" (2015). *Graduate Theses, Dissertations, and Problem Reports*. 6172.
<https://researchrepository.wvu.edu/etd/6172>

This Thesis is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Thesis in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This Thesis has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact researchrepository@mail.wvu.edu.

A Multi Agent System Design for Power Distribution Restoration Using Neural Networks

Mohamad Mashta

Thesis Submitted

To the Benjamin M. Statler College of Engineering and Mineral Resources at
West Virginia University

In partial fulfillment of the requirement for the degree of
Master of Science in the Electrical Engineering

Muhammad A. Choudhry, Ph.D., Chair

Ali Feliachi, Ph.D.

Natalia A.Schmid, Ph.D.

Department of Computer Science and Electrical Engineering

Morgantown, West Virginia

2015

Keywords: Power Distribution Systems, Multi-Agent Systems, Fault Detection, Restoration,
Neural Networks

Copyright 2015 Mohamad Mashta

Abstract

A Multi Agent System Design for Power Distribution Restoration Using Neural Networks

Mohamad Mashta

The state of the art of power distribution systems is to demand a more accurate response. It also provides more reliability for fault location and restoration respectively. A multi-agent system design for power distribution has been developed using the change of current methodology to detect and locate any type of faults. Employing the artificial intelligence for restoration process is the most important contribution to this study. Since feed-forward neural networks are weight training based back propagation concept, radial basis neural networks showed more efficiency by using the minimum error method to optimize the decision. A Probabilistic radial basis Neural Network (PNN) is designated at each feeder agent to implement the reconfiguration by analyzing the impedance and current values for each zone. The appropriate decision for the optimal reconfiguration case is a vector of activation signals associated with each switch to restore the power to the un-faulted zones of distribution feeder.

This study examines the role of Universal Asynchronous Receiver Transmitter (UART) buffer circuits in the laboratory experiment demonstration of the multi-agent system design. The main approach of a self-healing concept is the protection system. A recloser has been developed and improved for more sensitivity and faster response to detecting a fault where ever it occurs and lead the process of isolating and re-configuration. An electronic buffer circuit using digital microcontroller has been associated with the recloser and agents switches in order to offer a satisfying feedback for the proposed approach. Simulation studies, using MATLAB SimPowerSystems and, Neural Network toolboxes, for the proposed power distribution system showed improved results for fault location and restoration using Radbas neural networks. Hardware implementation with high accurate software data scoping of results has been employed to show the difference in time response using Universal Asynchronous Receiver Transmitter buffers at each switching relay in the design.

Contents

Abstract.....	II
1. Introduction.....	1
1.1 Thesis Outline	3
2. Multi-Agent Systems	4
2.1 Multi Agent Systems Approach	4
2.1.1 Multi Agent Systems definition	4
2.1.2 Agent definition	4
2.1.3 Intelligent agent	5
2.2 Multi agent System Structure	5
2.2.1 Centralized	6
2.2.2 Decentralized	6
2.2.3 Hybrid	7
2.3 Proposed Structure	8
2.3.1 Feeder agent.....	8
2.3.2 Zone agent.....	8
2.3.3 Substation agent.....	9
2.4 MAS Communications.....	10
2.4.1 Wired technique.....	10
2.4.2 Wireless technique.....	10
2.4.3 Agent Communication Language	10
3. Fault Location and Isolation	12
3.1 Fault Types.....	12
3.1.1 Shunt Fault.....	12
3.1.2 Series Fault.....	13
3.2 Fault Location Equations	13
3.3 Fault Location Algorithm.....	13
4. Buffer Design.....	15
4.1 Introduction	15
4.2 Protection Integrated Circuit Chip	15
4.3 Microcontroller Integrated Circuit Chip	19
4.3.1 Microcontroller	19

4.3.2 Oscillator	21
4.3.3 Power supply	22
4.4 ON/OFF Switching Circuit	23
5. Artificial Intelligence	26
5.1 Fundamental Concepts in Artificial Neural Networks	26
5.2 Common Activation Functions	27
5.2.1 Step Function	27
5.2.2 Linear Function	28
5.2.3 Sigmoid Function	28
5.2.4 Hyperbolic Tangent Function	29
5.2.5 Gaussian Function	29
5.3 Perceptron Neural Network	30
5.4 Multiple Adaptive Linear Neural Network	31
5.5 Multilayered Back Propagation Feed Forward Neural Network	32
5.5.1 Back propagation algorithm steps	33
5.5 Radial Basis Neural Network	36
5.5.1 Probabilistic Neural Network	37
5.5.2 Probabilistic Neural Network Step by Step:	39
5.5.3 Probabilistic Neural Network Numerical Problem:	39
5.5.4 Radial Basis Neural Networks in MATLAB:	42
5.5.5 Probabilistic Neural Network use in Restoration:	43
6. MATLAB Simulation Studies	44
6.1 West Virginia Super Circuit	44
6.2 Recloser	45
6.3 Results	50
6.3.1 Overcurrent Fault Detection Isolation and Restoration Case	51
6.3.2 Ground Fault Detection Isolation and Restoration Case	54
7. Implementation	57
7.1 Multi-Agent System Fault detection and Restoration	57
7.1.1 Multi-Agent System Structure	57
7.1.2 Fault Location Equations	58
7.1.3 Fault location and restoration Flow chart diagram	59

7.1.4 Sequence diagram	60
7.2 Components.....	61
7.2.1 SEL measurement tool.....	61
7.2.2 Control Units.....	62
7.2.3 Ethernet Communications.....	63
7.2.4 Software	64
7.3 Results	66
8. Conclusion	69
Future Work	69
References.....	70

1. Introduction

In many applications, Multi Agent Systems (MAS) have been developed in order to save time by leaving room for more than one unit to share solving a problem. Most recent research studies have emphasized on enhancing the performance and the quality of multi agent systems. The term multi agent system has been defined in many different ways. International Workshop on Intelligent Robotics and Systems has defined the MAS as “the multi-agents working together for doing something that creates a progressive result such increasing performance or saving time” [1]. *Changhong et al.*, have proposed two definitions: “one agent adopts the goal of another agent. Its hypothesis is that the two agent have been designed in advance and, there is no conflict goal between them, furthermore, one agent only adopts another agent’s aim passively” and “one autonomous agent adopts another autonomous agent’s goal. Its hypothesis is that cooperation only occurs between the agents, which have the ability of rejecting or accepting the cooperation” [2]. As long as there is no clear cut definition for multi agent system, many papers may define an agent in different words. *Janeth et al.*, introduced an agent as, “an agent is autonomous when it operates without the direct intervention of humans or other agents. It can decide by itself what it needs to do in order to achieve its objectives” [3].

Power Distribution Systems have encountered issues with interruption since research on multi-agent systems for power distribution began. More than 80% of the outages occur in distribution systems according to U.S department of Energy studies [4]. When a fault occurs in a power distribution feeder, the corresponding protection recloser will lock out after three trials to eliminate the fault effect. From this point, the idea of automating power distribution has been introduced. Self-healing power distribution system can at least move the outage scenario to a better case by intelligently making the best decision to analyzing the type, location. In other words, re-configuration will reduce the outage of power distribution systems by maximizing the operation capability under fault conditions.

In 2007, *Collazo et al.*, have simulated a Self-Reconfigurable Electric Power Distribution System using Multi-Agent Systems “in order to maximize the number of served loads with highest priority”. MAS implemented using Java programming language, and jade platform, FIPA agent technology standards are used for fault location and reconfiguration [3].

Belkacemi and Feliachi demonstrated another concept on a model of Southern California Edison’s Circuit of the Future in 2010. An artificial immune system approach that has the capability of self-learning and memorizing has been proposed for power distribution healing. Different agents with different functions are associated with the multi-agent system used for the application. “One of the used agent is a distributed generator agent that has the ability to communicate with other agent to decide whether or not to connect the generator to the network depending on the state of the circuit” [5].

“Operation of a Multi-Agent System for Load Management in Smart Power Distribution System design” was proposed in 2012 by *Biabani et al.* The dispatch algorithms in regularity base are used for controllable loads. MAS is programmed by ZEUS, tool-kit for building collaborative

agent applications, in order to make use of the forecasted load data to dispatch the load in such a way to decrease the peak demand. Neural Network and SimPowerSystems MATLAB toolboxes have been employed to obtain the load forecasting model. Load forecasting is designed to perform with pre-trained neural networks, the information is given as an input to forecast the load [6].

A decentralized multi-agent system, which works in real time with a power distribution system for fault detection applications, was simulated to meet the architecture of West Virginia Super Circuit (WVSC) by *Ghorbani et al.*, in 2012. The design has been set to deal with RMS values of voltage and current in the grid. The multi-agent model is simulated in MATLAB Simulink using level-1 S-functions and the power system is modeled using the Simulink SimPowerSystems toolbox. The design was successfully able to detect the faulted zone and recognize the type of fault [7].

Ghorbani et al., continued their work on fault detection isolation in power distribution systems using MAS by proposing the algorithm of current usage at each zone agent in 2013. WVSC is tested with Distributed Generation Sources using MATLAB Simulink environment. Results showed that faulted zones have been determined accurately with identifying the type of fault [8]. As an improvement, a Hybrid Multi Agent approach for Power Distribution System Restoration has been studied by the same authors in 2013. A combination of centralized and decentralized concepts is considered in the same circuit for quicker restoration. Three main optional states represent the algorithm: normal state, fault location state, and restoration state. Fault location and isolation algorithm is done by the communication among agents while the restoration algorithm is processed by the corresponding feeder agent. Two scenarios with different fault types have been explained and analyzed successfully using MATLAB [4].

As a related study, an investigation of communication media requirements for self-healing power distribution systems has been done by *Ghorbani et al* as well in 2013. The author illustrated the different structures of multi agent systems such as hierarchical and centralized, bandwidth and reliability terms have been discussed before a comparison between wired and wireless communication media performance is presented. The required time pauses for fault detection and isolation algorithm are evaluated for centralized, hierarchical and distributed structures. Results showed that wireless is more desired with the high quality of messaging feature, results also showed that centralized and distributed are better in latency than hierarchical [9].

One of the most recent studies about smart power distribution reconfiguration was presented in 2014 by *Saraiva et al*. The reconfiguration has been applied to various objectives: minimization of power losses, voltage regulation, load balancing, to name a few. The model of MAS has been applied to a power distribution system with two study cases of 11-bus and 16-bus [10]. *Nagata et al.*, have presented a decentralized model of power distribution system restoration using MAS concept in 2014, the autonomous operation of maximizing the de-energized area is realized in their proposal. The number of agents is restricted in order to reduce hanging messages during the restoration process [11].

1.1 Thesis Outline

Multi-agent systems are presented in the chapter (2). The definition of a multi-agent system has been introduced in addition to the definition of an agent. The characteristics of an intelligent agent are illustrated as well. Multi-agent system structure types are classified; possible MAS communications are also listed.

In chapter (3), most common fault types are defined according to IEC standards. Fault location and isolation equations and algorithm are illustrated. Flowchart has been presented to describe the algorithm step by step.

Chapter (4) is about designing the electronic buffer circuit that is used at each switch in the laboratory experiment demonstration. All components such as buffer chip and the microcontroller with their specifications and operation conditions are listed. All Softwares used in the design are illustrated too.

Fundamental concepts in artificial intelligence including common activation functions are presented and explained in the chapter (5). Popular neural networks such as perceptron, multiple adaptive linear and back propagation feed-forward neural networks are defined and illustrated. Radial basis neural network is studied and applied to restoration in more details.

Simulation studies of two restoration scenarios are discussed using MATLAB in the chapter (6). Two different types of fault have been applied at two different zones in the proposed circuit. Results from the simulation studies are shown in the end.

Chapter (7) demonstrates a laboratory experiment of fault location isolation and restoration algorithms. All components used for the hardware implementation have been listed. Data results are plotted for different cases.

2. Multi-Agent Systems

2.1 Multi Agent Systems Approach

In order to exhibit the advantages of using multi-agent systems (MAS) in automation and power distribution engineering, basic concepts and function of MAS must be understood in detail. Multi-agent systems are solutions to problematic or difficult functions. They support and send information within agents but operate independently.

2.1.1 Multi Agent Systems definition

Multi-agent systems are subsets of intelligent agent units that exchange data to achieve a shared interest. There are no specific applications for multi-agent systems; they are simply useful when a process of two or more functions are needed. Examples include traffic and transport organizing, power restoration, and robotics applications [12]. Moreover, the main idea is that designing a multi-agent system requires multiple agents to interact with each other in order to meet the purpose of the design. The foundation of multi-agent systems is to communicate amongst various agents directly and indirectly. In this work, the multi-agent system represents a power distribution network that has the ability to detect a variety of faults and determine its location. Additionally, it can reconfigure the power network by treating the unaffected zones with power backup restoration approach.

2.1.2 Agent definition

While multiple agents are widely used and promoted, single agents can be utilized depending on the designer. The term “agent” is routinely seen amongst computer networking systems. The computer science community has defined an agent as one of the thousands of functions working in parallel to produce a real artificial intelligence [13]. However, from an engineer’s perspective, Rohbogner et al., defines agents in smart grids as “a control entity can be labeled as an agent in smart grids when: first, it has the ability to communicate with other control units or agents. Second, it consists of different parts of software implemented systems. Third, it is responsible for an automatic power management of an underlying physical system” [14]. Power distribution agents are observable and can include lists of parts associated with the interaction between agents. Tools may be sensors, high-quality measurement tools, control units, amplifiers or protection modules. Agents deal with the associated parts in two ways: physically or centrally. They physically can

change the status of a switch from open to close. The central system can hold sample values or move from function to function.

2.1.3 Intelligent agent

An intelligent agent is simply an independent agent that can control its behavioral functions and internal status. It also can sense the surrounding environment and hinder any unwanted behavior that it may encounter [13]. The three following characteristics can summarize an intelligent agent:

1. Reactivity: an intelligent agent can react in timely manner to changes that occur in its environment in order to accomplish its goal.
2. Pro-activeness: all agents must have access to each other so that the goal can be achieved with no obstacle. For instance, if communication with an agent is lost, same needed data can be acquired from another agent.
3. Social ability: all agents deal and exchange data interdependently and have the ability to collaborate with the other intelligent agents. Furthermore, if intelligent agents are operating with different hardware or software, exchanging information should be accessible and compatible with the next agent.

In power distribution applications, the three characteristics explored are required to fulfill fault locations and restoration algorithms. Social ability is important because different types of control units can be used on the same network.

2.2 Multi agent System Structure

The multi-agent system method is applied in power distribution networks in order to obtain protection, fault detection, fault location and restoration to self-heal the network. In general, centralized and decentralized approaches are used for multi-agent system architectures to reach the desired protection. More details will be stated on both strategies in this chapter. In some applications, agents have different function ability and responsibility [8]. Agent Communication Language (ACL) standards, Knowledge Query and Manipulation Language (KQML), and Foundation for Intelligent Physical Agents (FIPA) are the common standard resources for MAS. In this work, all agents have the same functions installed with the same standard of messaging and communication protocols [15].

2.2.1 Centralized

In the centralized approach, all agents are connected to a master control unit and cannot function without a controlled dependent algorithm.

Centralized structure is shown in the Fig. 2.1 below:

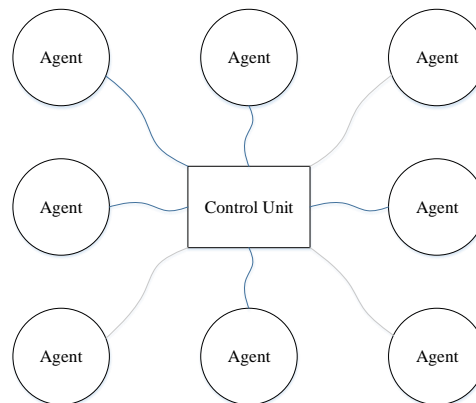


Figure 2.1: Centralized MAS structure

The Central control unit is responsible for all algorithms of fault detection, location, and isolation and reconfiguration restoration in MAS. All other agents are subject to receive and implement decisions taken by the master control unit. The disadvantage of this approach is if the control unit is tampered, it ultimately creates negative outcomes to the whole MAS design and stops the self-healing. Moreover, there will be no benefit amongst the neighboring zones when there is a loss of connection between an agent and the master control unit regarding the architecture.

2.2.2 Decentralized

In this architecture of MAS design, all agents are connected to its neighbor as presented in Fig. 2.2. Each intelligent agent is responsible for its algorithms and is supposed to make the decision independently. In case of connection loss, each agent can be programmed and equipped to re-establish a new connection with the nearest neighbor. This kind of MAS structure is preferred because of its capability to detect fault location and restoration. Each algorithm can run independently and provide a plethora of backup communications.

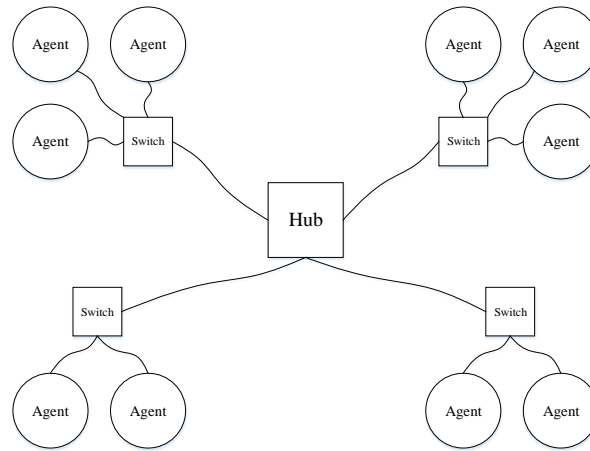


Figure 2.2: Decentralized MAS structure

2.2.3 Hybrid

Hybrid version is a combination of both centralized and decentralized structures. The idea of hybrid architecture is to obtain both advantages in centralized and decentralized structures in one design. The functionality of this system is designed to act as centralized and decentralized. If the control unit malfunctions, then the decentralized structure becomes the forefront. In addition, all agents can have access to its neighbors. If the centralized mode is not available then the system can switch to decentralized without any interruption of function. If the control unit were to fail, the disruption does not affect the entire MAS [4]. The following Fig. 2.3 illustrates the combination of centralized and decentralized structures into hybrid version.

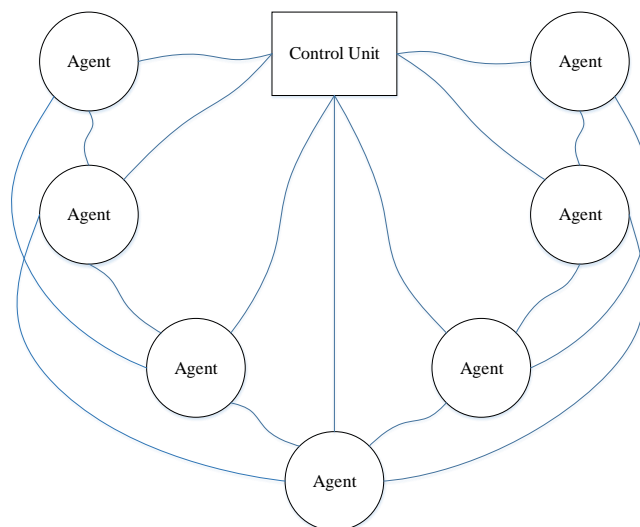


Figure 2.3: Hybrid MAS structure

Three types of multi-agent systems have been stated to compare and figure out the best choice for fault location, isolation and reconfiguration restoration. Decentralized has been chosen to implement the approach because of its reliability and flexibility. The decentralized mode is most convenient because it only requires hubs and switches.

2.3 Proposed Structure

West Virginia Super Circuit (WVSC) is presented [8] in figure (2.4). Proposed algorithms of self-healing will be tested and simulated for this WVSC structure.

Ghorbani has described WVSC in [8]:

Different zones represent West Virginia Super Circuit by sixteen switches which are located in between zones for the self-healing propose as shown in figure (2.4). The black color indicates that the initial status of the switch is normally closed while the white color indicates the normally open ones. Feeder agents are located at the recloser location. West Run Substation has five feeders. They are supplied with protection reclosers. Whenever a fault occurs, the recloser will start three trials and then will lock if the fault is not cleared.

Each agent is represented by an industrial grade computer supplied with uninterruptible power supplies (UPS). Main role of switches is to isolate fault and to do re-configuration for restoration.

2.3.1 Feeder agent

Feeder agent role is to communicate with the other feeders on the same power distribution system to examine if the problem can be solved by merging affected zones to other feeder when fault location is detected. Feeder agent is located by the recloser.

2.3.2 Zone agent

Each zone agent is composed of a control unit that is connected to sensors. A control unit is equipped with multiple communication ports in order to broadcast to its neighbors. Sensors are mainly provided to keep track of voltage and current waveforms and have the ability to communicate with the control unit. Each zone agent is represented by the analogous switch which means sixteen switches in WVSC are basically sixteen zone agents.

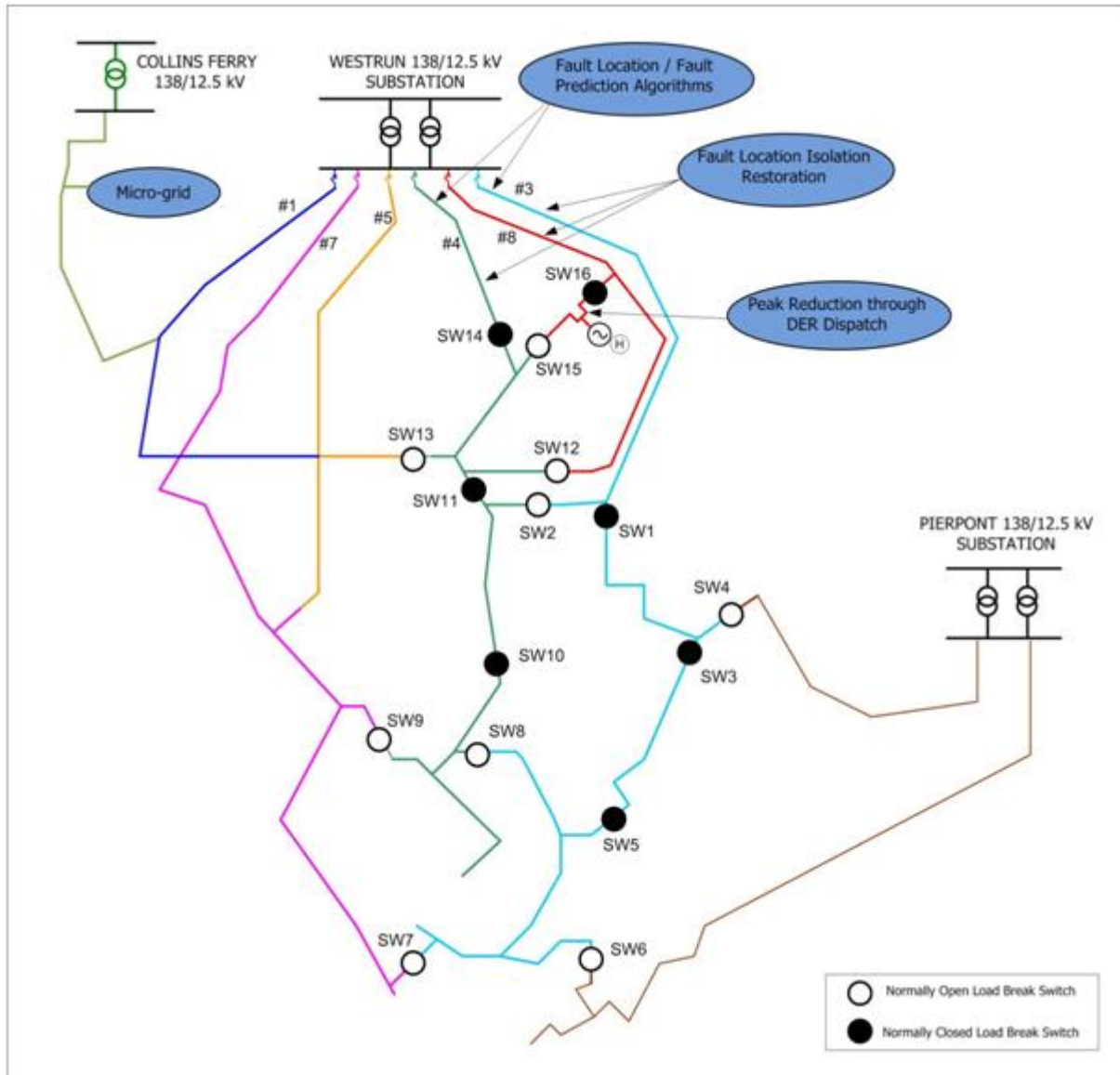


Figure 2.4: West Virginia Super Circuit [8]

2.3.3 Substation agent

Substation agent is significantly responsible for arbitration with other substation agents that are directly in charge of faulted zones. They acquire data from their neighbor substation about the corresponding fault.

2.4 MAS Communications

There are two main communication methodologies that are associated with MAS: wired and wireless. Each approach has their own advantages and disadvantages. The convenience of wired approach is that it can be directly plugged in without any standing utility software regardless of setting the addresses. On the other hand, any loss of connection will not be cost effective and will require manual labor to replace and fix hardware. Wireless technique is expensive to implement and has multiple protocols regarding the desired distance of MAS design [9].

2.4.1 Wired technique

Multiple useful configurations are associated with the wired approach for MAS design. Power Line Communication (PLC) is the most common type used with MAS besides fiber optics and Copper-wire line for Wide Area Network (WANs) [16]. While the wired technique can be affected by surrounding noise, it would be advisable to attach a noise filter.

2.4.2 Wireless technique

Wireless configuration guarantees a long lasting connection despite any power failure in the power distribution. This allows for solutions to be processed faster. Wireless term indicates many choices such as ZigBee, WiFi, WiMax and cellular (3G/4G) for MAS design. The advantage of choosing wireless technology is the high rate of data transfer in metering purposes. The range of both WiFi and WiMax is large while ZigBee is small [16].

2.4.3 Agent Communication Language

As MAS technology has linearly advanced, Agent Communication Language has followed closely. Knowledge Query Management Language (KQML) was first presented in 1980s by US Defense Advanced Research Projects Agency's Knowledge-Sharing Effort. At the beginning of 1990s, Arcol was proposed by France Télécom, which uses fewer messages than KQML. A set of thirteen primitives has recently been scheduled by Foundation for Intelligent Physical Agents (FIPA). FIPA has surpassed the previous standards of Arcol [15]. FIPA messaging parameters [17] sequence architecture is described in the following table:

Message Field	Description
performative	Type of communicative acts
sender	Participant in communication
receiver	Participant in communication
reply-to	Participant in communication
content	Content of message
language	Description of Content
encoding	Description of Content
ontology	Description of Content
protocol	Control of conversation
conversation-id	Control of conversation
reply-with	Control of conversation
in-reply-to	Control of conversation
reply-by	Control of conversation

Table 2.1: FIPA messaging parameters sequence

3. Fault Location and Isolation

The determining of fault location is the first process in power distribution when an intelligent method is applied. The trip of fault supervision can be summarized in the following four levels:

- 1- Immediate sensing when a fault occurs.
- 2- Fault location identification.
- 3- Faulty zone isolation.
- 4- Restore the power to zones those are affected by isolation

Non-faulty zones will start communicating with the corresponding feeder to start restoration process for power supply re-configuration. Once the fault is isolated, restoration is recommended to decrease the outage time. Multi-agent system will communicate among agents for a quick fault location and isolation.

In order to understand fault location and isolation method, possible fault types in power distribution need to be classified.

3.1 Fault Types

Fault term in power distribution refers to a wrong action which can cause a direct danger to humans and damage of properties. Type of fault is significantly related to the magnitude of current, the location of fault, the system earthing, the source impedance and the impedance of fault [18].

Fault types can be summarized to shunt or series faults. Shunt faults are faults when one or more of the phases are short-circuit (can be to earth) [18]. Series faults are caused by an interruption to one or more phases.

3.1.1 Shunt Fault

Shunt fault is more severe than series fault and directly associated with the system impedance and fault impedance. An IEC-standard has defined the shunt fault [19] as following

“Shunt fault is a short circuit fault that is characterized by the flow of current between two or more phases or between phase(s) and earth at the frequency of the associated power system”

3.1.2 Series Fault

Series fault can be caused by a circuit breaker malfunction, broken conductor or fuse operation in more than on phase. An IEC-standard [19] has the following definition of a series fault

“A fault for which the impedances of each of the three phases are not equal, usually caused by the interruption of one or two phases.”

3.2 Fault Location Equations

The deducted current between the sum of currents entering the zone and currents leaving the zone is considered as the zone current usage [8]. Current consumed in zone J can be calculated by Kirchhoff's law while each agent can communicate with its neighbor agents:

$$I_{Zj} = \sum_{i=1}^n I_{Enter} - \sum_{i=1}^m I_{Leave} \quad 3.1$$

Auditing the change of current usage in each zone would be a substantial factor to determine the degree basis of the usual event [8]. Change of current can be calculated by the following formula:

$$I_{Zc} = \left(\frac{|I_{Znew}| - |I_{Zold}|}{|I_{Zold}|} \right) * 100 \quad 3.2$$

3.3 Fault Location Algorithm

The following flow chart demonstrates step-by-step fault location and isolation process in the multi agent system design for power distribution:

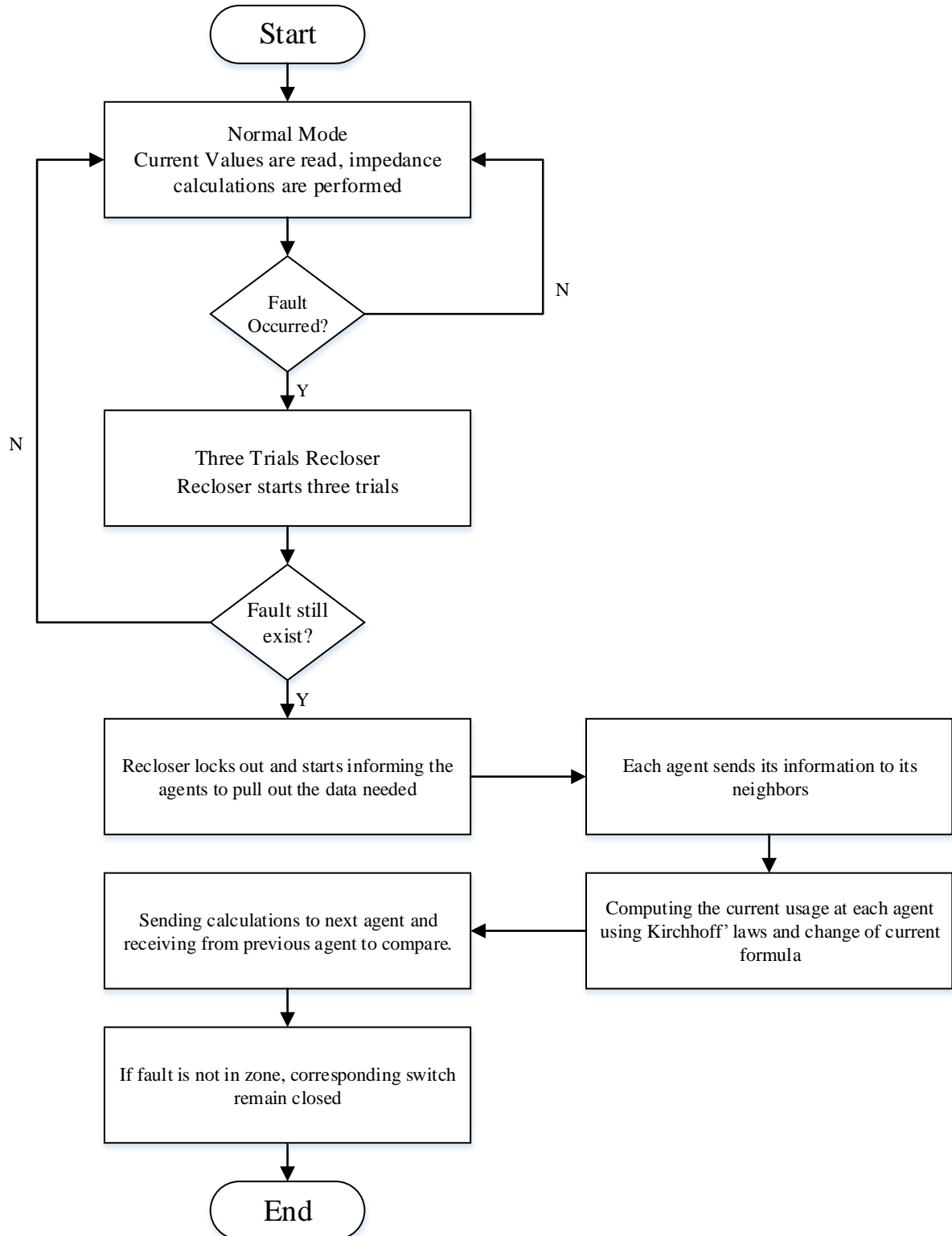


Figure 3.1: Fault location and isolation flow chart

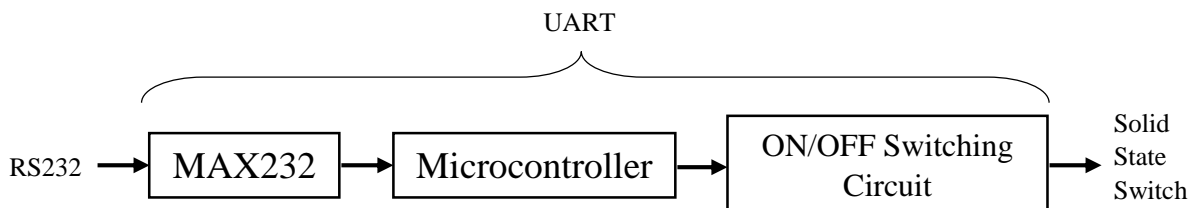
4. Buffer Design

4.1 Introduction

The main idea of designing a buffer is to protect the control unit (which can be a personal computer, Industrial computer or measurement tool) when a control signal is needed from the control unit. The control signal is specifically to activate a desired operation such as switching a transistor, solid state relay or led indicator. In fault detection operation, Universal Asynchronous Receiver Transmitter (UART) buffer has been designed to switch the recloser on-off and to control the isolation relays. In the restoration process, buffers play the same role for switches to restore the power after fault isolation. The response of buffers was efficient and accurate by setting the on-time and off-time. RS232 port is a serial communication port that is used to acquire the control signal of 5V from the control unit where the buffer output is adjustable from [5 – 12] V since switching relays input is [5-25] V.

RS232 was first introduced by the Electronic Industries Association (EIA) in 1960s. It is a control port provided in control units and computers. The first version of RS232 was with 25 pins; then it has been modified to be 9 pins because most pins in the old version are not used in PC applications. Most computers and other devices use RS232 interface to send and receive data with UART circuits which are built with microcontroller chips [20].

In order to obtain protection for RS232 (RS232 are usually designed as a part of an integrated circuit in computers), MAX232 has been added to UART circuit. MAX232 is an integrated circuit that has been created by Maximum Integrated Products in 1987 [21]. It interfaces the signal from RS232 port to Transistor-Transistor Logic (TTL) so that transmitted signal can be used in microcontroller chip.



4.2 Protection Integrated Circuit Chip

As mentioned before, protection for UART is needed when designing a buffer. MAX232 is normally an IC chip that converts the received signal to five volts TTL/CMOS level. Common applications of MAX232 are in modems, computers, battery powered systems and terminals. A simplified schematic of MAX232 chip [21] is given in figure (4.1) below:

A simplified schematic of MAX232 chip is given in figure (4.1) below:

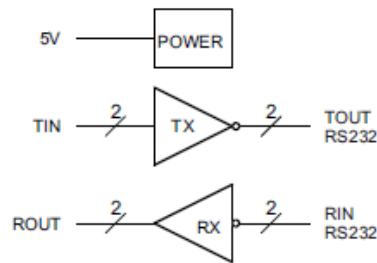


Figure 4.1: MAX232 general schematic

Features given by Texas Instrument for MAX232 in data sheet (MAX232, MAX232I SLLS047M –FEBRUARY 1989–REVISED NOVEMBER 2014) are the following:

- 1- Meets or Exceeds TIA/EIA-232-F and ITU Recommendation V.28.
- 2- Operates from a Single 5-V Power Supply With 1.0- μ F Charge-Pump Capacitors.
- 3- Operates up to 120 Kbit/s.
- 4- Two Drivers and Two Receivers.
- 5- \pm 30-V Input Levels.
- 6- Low Supply Current: 8 mA Typical.
- 7- ESD Protection Exceeds JESD 22 – 2000-V Human-Body Model (A114-A).
- 8- Upgrade with Improved ESD (15-kV HBM) and 0.1- μ F Charge-Pump Capacitors is Available with the MAX202 Device.

MAX232 functions and pin configuration are given in MAX232 datasheet (MAX232, MAX232I SLLS047M –FEBRUARY 1989–REVISED NOVEMBER 2014) below:

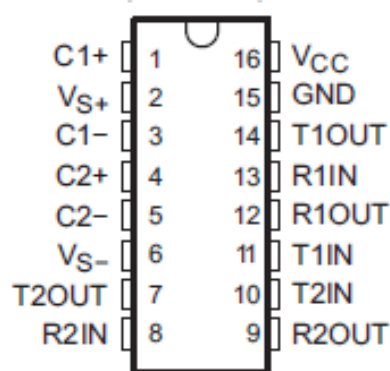


Figure 4.2: MAX232 top view

Table (4.1) shows pins functions in MAX232:

PIN		Type	Function
Name	PIN#		
C1+	1	N/A	Capacitor positive basis C1
VS+	2	Output	To charge in capacitor only (positive)
C1-	3	N/A	Capacitor negative basis C1
C2+	4	N/A	Capacitor positive basis C2
C2-	5	N/A	Capacitor negative basis C2
VS-	6	Output	To charge in capacitor only (negative)
T _{2out} , T _{1out}	7, 14	Output	Data output (write in RS232)
R _{2in} , R _{1in}	8, 13	Input	Data input (read from RS232)
R _{2out} , R _{1out}	9, 12	Output	Data output (write in UART)
T _{2in} , T _{1in}	10, 11	Input	Data input (read from UART)
GND	15	N/A	Ground
V _{cc}	16	N/A	Chip power supply (5V)

Table 4.1: MAX232 pins functions

^[9]Preferred power supply for MAX232 is a maximum of 5.5V and not below 4.5V. Input voltages for T1in, T2out must be in the range [-0.3 0.3] V. Data to be read from RS232 can be up to 30 V with an unlimited duration of Short-circuit. Operating virtual temperature is high at 150 C°. Recommended operating conditions for Max232 is 5V supply voltage and minimum voltage of 2V for high-level input voltage (T1in, T2in). Receiver input voltage is in the range of [-30 30] V for (R1in, R2in) with operating free-air temperature of 70 C° maximum and 0 C° minimum. Supply current is not more than 10mA at maximum power supply of 5.5 V with free air temperature 25 C°. Switching characteristics [21] are bulleted below:

- Slew rate is 30 V/μs at load resistance of 3 to 7 kΩ
- Transition region slew rate 3 V/μs.
- Data rate of 120 Kbit/s for one Tout switching.
- Receiver propagation delay time is 500 ns when low to high level output.
- Receiver propagation delay time is 500 ns when high to low level output.

Note: C1-C4 is 1 μF with power voltage of 5 V for switching characteristics.

Typical operating circuit for MAX232 in the buffer design is shown in the figure (4.3). Four capacitors of (0.1 μ F) have been used in MAX232 circuit. (T1out, R1in) pins are connected to the Serial port.

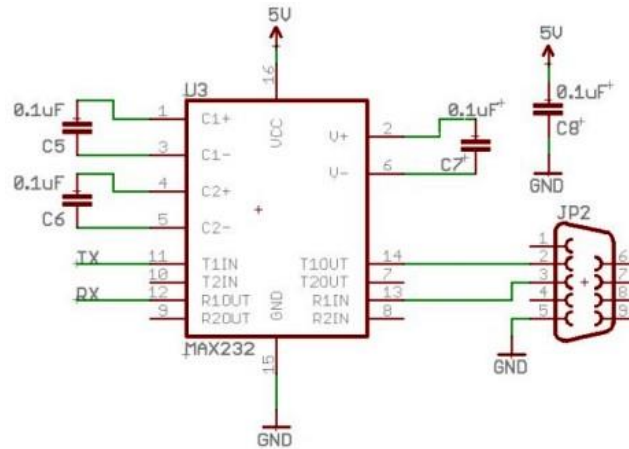
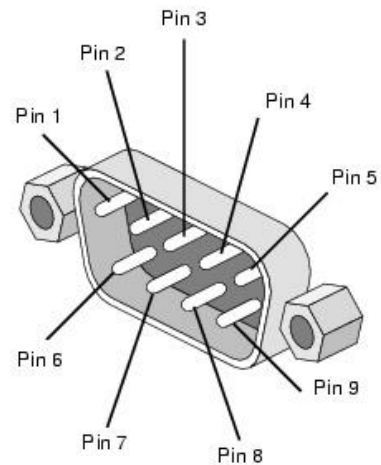


Figure 4.3: MAX232 Typical Operating Circuit

In order to understand the connections for MAX232, serial port pins functions are listed in the table (4.2) below:

PIN		Function
Name	PIN#	
DCD	1	Data carrier detector pin
RD	2	To receive data from UART
TD	3	To transmit data from UART
DTR	4	Data terminal ready notification
GND	5	Ground
DSR	6	Data set ready notification
RTS	7	Request to send
CTS	8	Clear to send
RI	9	Ringing indicator

Table 4.2: RS232 pins functions



Switching relays do not require a lot of functions when using UART protocol. To transmit and receive data from UART, only read and transmit pins (RD, TD) from the serial port of each control unit are needed. Figure (4.4) illustrates the connections needed from RS232 to MAX232 in a simple diagram.

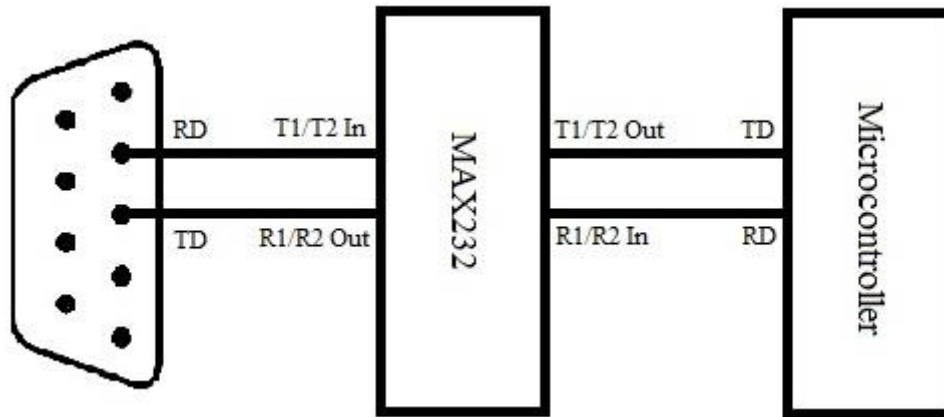


Figure 4.4: MAX232 connections box diagram

4.3 Microcontroller Integrated Circuit Chip

4.3.1 Microcontroller

In many applications, the need of using digital controllers became a priority; thus, microcontrollers were implemented. Microcontrollers vary regarding the function needed to drive a system. However, the hardest part in designing with microcontrollers, is the actual programming of the chip, especially if many functions are required in one system. The idea of using a microcontroller in buffer design is to facilitate the switching process and deal with TTL/CMOS levels to get 5V.

A comparison between AVR family microcontrollers and PIC family microcontroller is listed below:

	PIC	AVR	8051
Max CPU frequency	20 MHz	16 MHz	24 MHz
Pulses per Command	4 Cycle	1 Cycle	12 Cycle
Commands per Second	$20/4 = 5$ MIPS	$16/1 = 16$ MIPS	$24/12 = 2$ MIPS
Commands available	35	132	215
Code memory size	64 KBytes	256 KBytes	32 KBytes
Memory Structure	banked	Liner	Liner
CPU structure	RISC/Harvard	RISC/Harvard	CISC/Von Neumann
Band width	12 bit	16 bit	8 bit

PIC microcontroller has been chosen because it is easy to learn and only 35 commands of coding. It can be programmed in many languages, such as, MicroC and Bascom. Special features (given by Microchip in PIC12F683 data sheet) includes [22]:

- Provided with six I/O pins which is enough for buffer design.
- Programmable with many languages.
- Programmable with code protection.
- Internal oscillator 8 MHz to 125 MHz
- Power-saving sleep mode.

PIN		Function
Name	#	
VDD	1	Power Supply
GP5/T1CKI/OSC1/CLKIN	2	GPIO : I/O pin T1CKI : Timer OSC1 : Crystal Oscillator CLKIN : External Clocker pin
GP4/AN3/T1G/OSC2/CLKOUT	3	GPIO : I/O pin AN3 : A/D Channel T1G : Timer1 gate OSC2 : Crystal Oscillator CLKOUT : Clocker output
GP3/MCLR/VPP	4	GPIO : I/O pin MCLR : Master Clear Vpp : programming voltage
GP2/AN2/T0CKI/INT/COU1/CCP1	5	GPIO : I/O pin AN2 : A/D Channel 2 T0CKI : Timer Clock In 0 INT : Interrupt COU1 : Comparator1 CCP1 : PWM Output
GP1/AN1/CIN-/VREF/ICSPCLK	6	GPIO : I/O pin AN1 : A/D Channel 1 CIN- : Comparator 1 input Vref : Voltage reference value ICSPCLK : Serial Programming
GP0/AN0/CIN+/ICSPDAT/ULPWU	7	GPIO : I/O pin AN0 : A/D Channel 0 CIN+ : Comparator 1 input ICSPDAT : Serial Programming Ultra low power wake up In
Vss	8	Ground

Table 4.3: PIC12F683 pins functions

Figure (4.5) below shows a top view of microcontroller PIC12F683 that is used in designing the buffer.

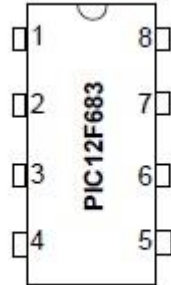


Figure 4.5: PIC12F683 microcontroller top view

PIC12F683 is flexible to be programmed and function for UART circuit. It only needs a few commands to receive the signal from the serial port and to send output port. This is suffice for buffer function. Uploading the code is easy by compiling it to hex file and using USP programmer connected directly to a personal computer.

4.3.2 Oscillator

An external oscillator can be used with PIC12F683 microcontroller for more accurate and time manner of clock pulses. 1 MHz quartz crystal resonator has been used as a clock source to operate the microcontroller. The following figure shows the configuration and connection of the oscillator:

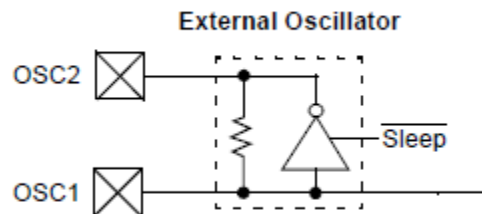


Figure 4.6: external oscillator associated with mc PIC12F683

4.3.3 Power supply

A linear power supply with high quality and performance is used. Four basic levels of rectifying the voltage from alternative to direct voltage are shown in this diagram:

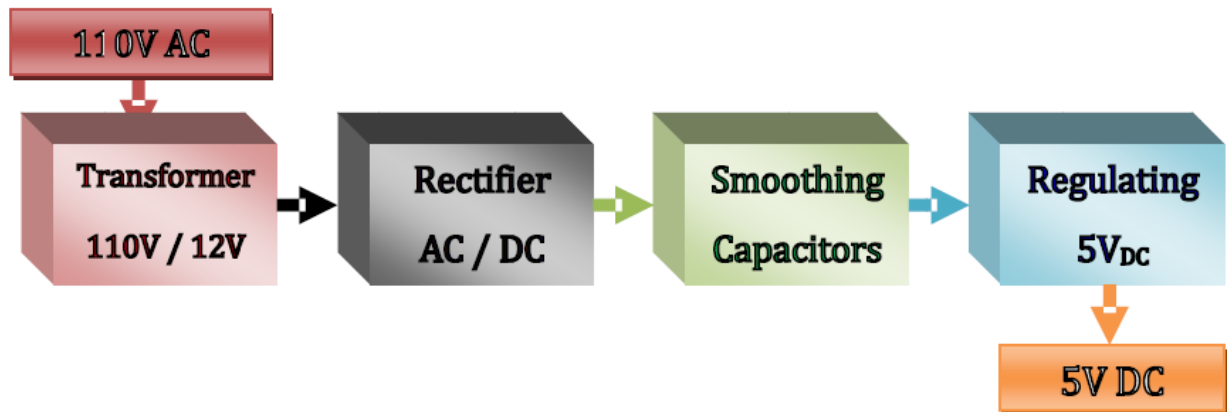


Figure 4.7: diagram of voltage rectifying levels

Power supply circuit has been simulated and tested using Proteus. Some applications, such as our design, need a power supply of (5-12) volts in order to operate more than one system. Both outputs 5 and 12 volts were accurate. The following figure illustrates the components used in the simulation:

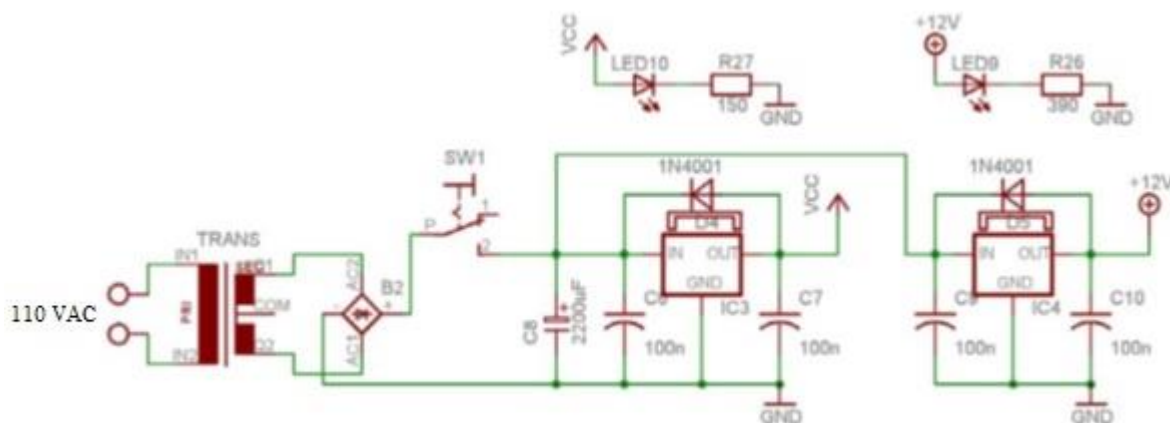


Figure 4.8: Schematic of power supply using Proteus electronic circuit simulator

4.4 ON/OFF Switching Circuit

In order to control high current loads such as (Induction Motors and heaters), microcontroller output current ($20 \text{ mA} \pm 10\%$) is not enough to connect directly to these kinds of loads. ON/OFF switching transistor circuits are required to handle this level of current.

As long as the power supply - for any control circuit –meets standard protocol, the functionality of the electronics elements used would be stable and ideal. The maximum current of each element used in the design should be considered.

In general, there are two transistor types used for ON/OFF switching:

- BJT Bipolar Transistors.
- FET Unipolar Transistors.

Practically, each type is used in regard to the controlled load. Some loads require power electronics in designing the control approach whereas other kinds can be handled using semiconductors. Table (4.4) inserted below shows the differences in using each one:

	BJT	FET (MOSFet)
Control Methodology	Controllable by base pin current with $V_{be}=0.6\text{v}$ and $I_b= 1 - 10 \text{ mA}$	Controllable by gate pin voltage
ON/OFF switching speed	200 MHz	2000 MHz
Operation	Affected by temperature	Not affected by temperature
Forward resistance	Small	big
Static effect	Not affected by static charges	affected by static charges
Input resistance	Medium	Huge ($10^{12} \Omega$)
Operation voltage	Not greater than 100 V	Can be more
Load current	Work for small currents	Can drive high current load
Operation noise	High noise	Low noise

Table 4.4: BJT Vs FET Transistors

In digital control systems, transistors are mainly used as controllable electronic switches (ON/OFF). In addition, three important factors must be considered:

- Load current through transistor.

- Power dissipated in transistor.
- On/off switching speed.

To control low-load current, the use of BJT transistor is enough while FET transistors are used to control high-current load. To use a transistor as an electronic switch, transistor must operate in both off and saturation regions.

- Off state : base current is zero
- Saturation state: load current is being passed through transistor.
- Active state: Collector current is the product of base current (I_B) and amplifier factor (h_{fe}) which is considered as an amplifying operation.

Figure (4.9) below illustrates all BJT transistor operations:

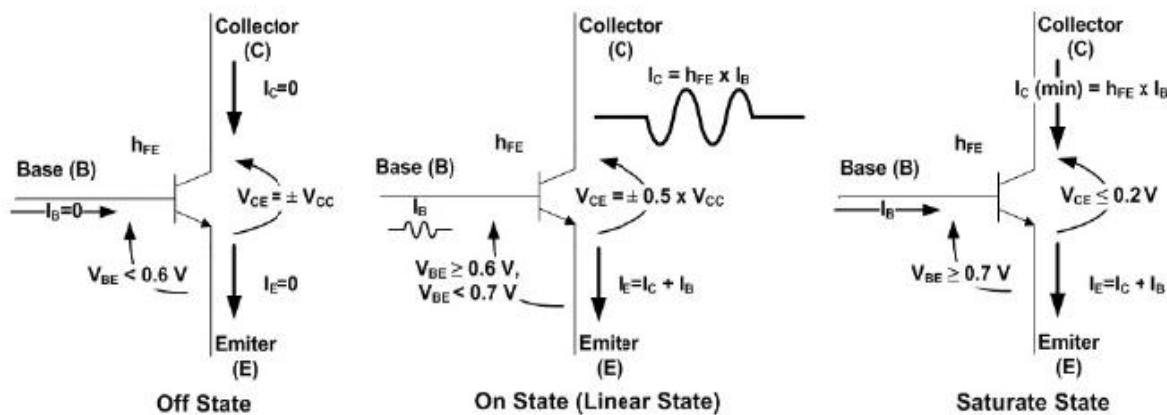


Figure 4.9: BJT three operation states

BC337 transistor has been added to the design. It has the following specifications:

$$I_{C_max} = 800 \text{ mA} \quad V_{BE_saturate} = 0.65 \text{ V} \quad V_{CE_saturate} = 0.2 \text{ V} \quad h_{FE} = 100 \quad V_{CE_max} = 50 \text{ V}$$

The minimum required current to control a transistor by gate can be calculated by:

$$I_C = h_{FE} \times I_B \quad \rightarrow \quad I_B = \frac{I_C}{h_{FE}} = \frac{100}{100} = 1 \text{ mA}$$

The maximum collector power is given by:

$$P_{C_max} = U_{CE} \times I_C = 0.2 \times 100 = 20 \text{ mW}$$

Then the base resistor can be evaluated by:

$$R_B = \frac{V_P - V_{BE}}{I_B} = \frac{5 - 0.7}{1} = 4.3 \text{ K}\Omega \quad P_{RC} = (V_P - V_{BE}) \times I_B = (5 - 0.7) \times 1 = 4.3 \text{ mW}$$

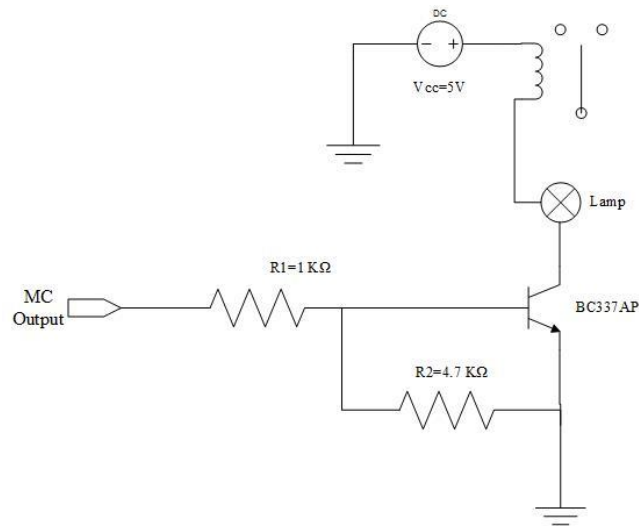


Figure 4.10: transistor electronic switch circuit

The final design of UART circuit as presented below has been simulated using Proteus:

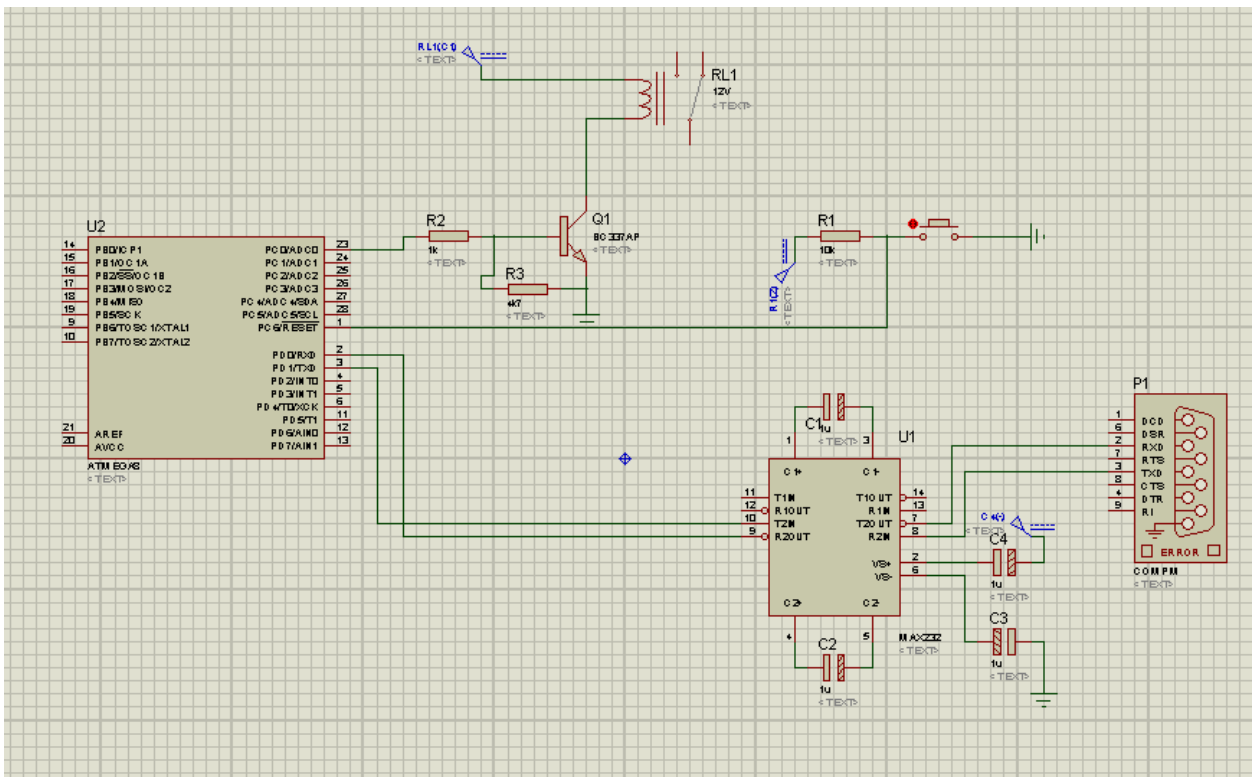


Figure 4.11: UART buffer design in Proteus environment

5. Artificial Intelligence

The concept of Artificial Intelligence (AI) was first presented in the 1950's [23]. The idea of artificial intelligence is to have machines operating in a similar fashion as humans. Producers try to create a system that mimics human analysis in how to make proper decisions. Furthermore, as a part of artificial intelligence, AI Neural Networks have been expressed to imitate the human brain neural networks. The idea of signals flowing through networks shares a similar concept of how the brain functions.

Artificial Neural Networks (ANN) use math techniques designed to simulate the way the human brain performs specific processes. It comprises process units for calculating the target output called (Neurons or Nodes). Those neurons have the ability to retain data and make it available when it is needed by adjusting neural network weights.

Artificial Neural Networks types have the same purpose in network training to obtain the proper weights and outputting the desired result for a certain input. For instance, the main goal of a neural network is to identify a person in relation to his or her biometrics, iris detection or face recognition (which can be represented in a vector of numbers). At the same time, it is important to exhibit the most common types of ANNs.

5.1 Fundamental Concepts in Artificial Neural Networks

An Artificial Neural Network consists of a certain number of neurons. Each neuron is connected to the other neurons in the next layer by a weighted and directed connector as shown in the Fig. 5.1.

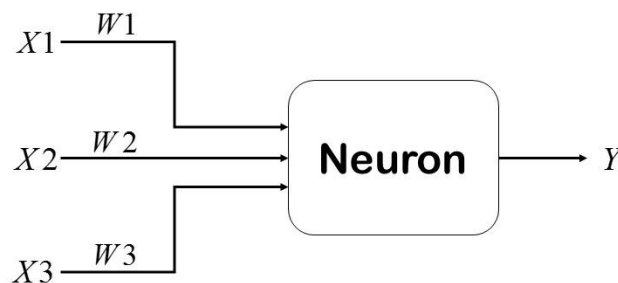


Figure 5.1: Basic Neuron

Fig. 5.1 illustrates a single neuron with an input vector of three elements and a single output. At the input of the neuron, each single input can be set by an amount of weight. Additionally, the type

of input can be varied and can be real positive, real negative, binary or bipolar. Weights in a neural network are mainly to magnify or shrink a neuron input signal. Weights must be positive to magnify and negative to shrink. A neuron can be represented by activation function where the input of it is the accumulated input of the total input.

$$y = f(\text{net}) \quad 5.1$$

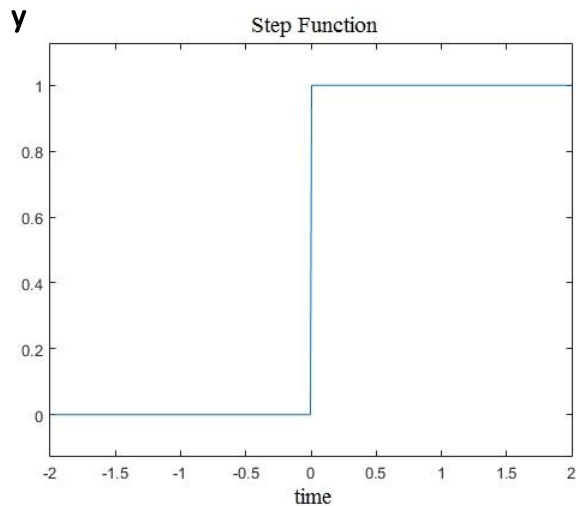
$$\text{net} = \sum_{i=1}^n x_i w_i + x_2 w_2 + \dots + x_n w_n \quad 5.2$$

5.2 Common Activation Functions

Activation functions are useful to transfer a signal from a certain level to a desired level in order to be applicable. For example, in the restoration process of a multi-agent design, step function can be used to generate a control signal regarding the neural network to activate the breakers. Activation function performances vary regarding the desired output. Moreover, the output of the step function is either (0) or (1), whereas bipolar sigmoid function generates an output in the range of [-1, +1].

5.2.1 Step Function

As mentioned, Step function is useful when an output of either (0) or (1) is needed. Step function formula can be given by:

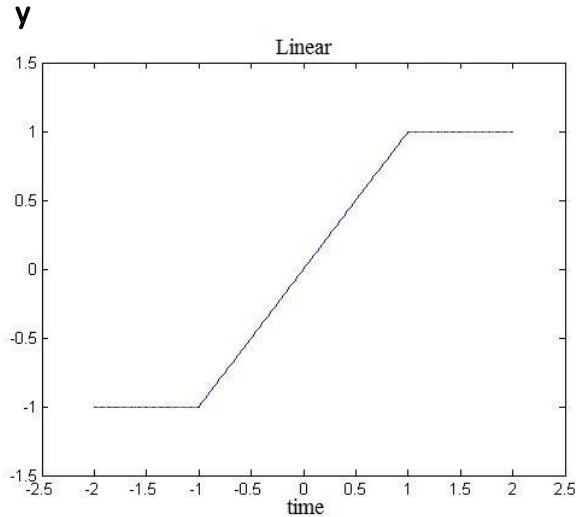


$$y = \begin{cases} 0 & \Leftrightarrow t < 0 \\ 1 & \Leftrightarrow t \geq 0 \end{cases} \quad 5.3$$

Figure 5.2: Step Function

5.2.2 Linear Function

The Linear function is mainly used to output a signal equal to the input. Also, it can be structured to function linearly in a certain range of the input, rather than constantly act for the remaining values of the input. The linear function can be represented as following:

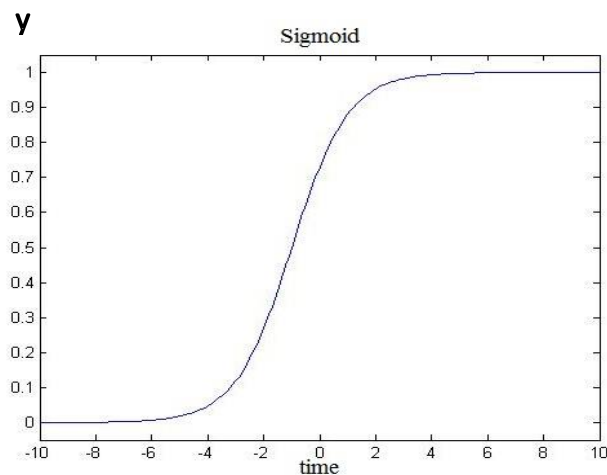


$$y = \begin{cases} -1 & \Leftrightarrow t < -1 \\ t & \Leftrightarrow t \in [-1, 1] \\ 1 & \Leftrightarrow t > 1 \end{cases} \quad 5.4$$

Figure 5.3: Linear Function

5.2.3 Sigmoid Function

Bekir Karlik et al., have stated in [24] that: “This function is especially advantageous to use in neural networks trained by back-propagation algorithms. Because it is easy to distinguish, and this can interestingly minimize the computation capacity for training. The term sigmoid means ‘S-shaped’, and logistic form of the sigmoid maps the interval $(-\infty, \infty)$ onto $(0, 1)$ ” as seen in the figure below:

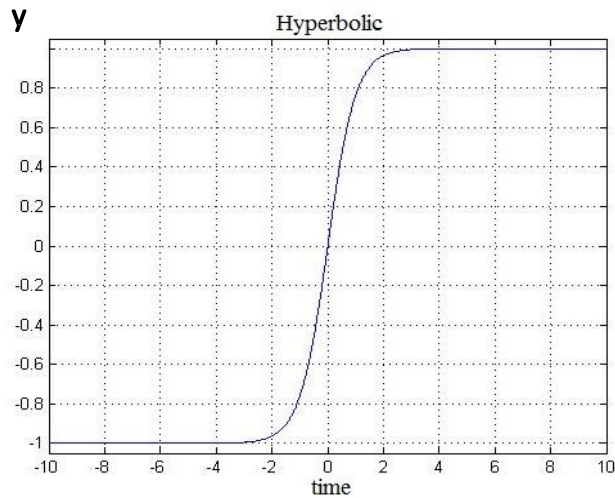


$$y = \frac{1}{1 + e^{-t}} \quad 5.5$$

Figure 5.4: Sigmoid Function

5.2.4 Hyperbolic Tangent Function

The hyperbolic function is another version of the sigmoid function. The main difference between the two is that the output range of hyperbolic is extended down to (-1) instead of (0). This function can be used for the bipolar application. The hyperbolic tangent function is the ratio of hyperbolic sinusoid to hyperbolic cosine as it is shown in Fig. 5.5

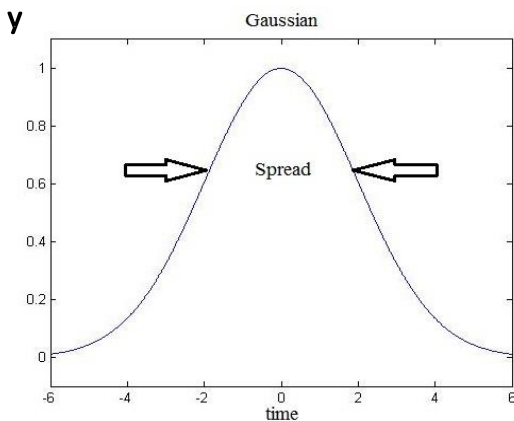


$$y = \frac{\sinh(t)}{\cosh(t)} \quad 5.6$$

Figure 5.5: Hyperbolic Tangent Function

5.2.5 Gaussian Function

For radial basis neural networks, the Gaussian function is flexible and precise for measuring the distance between the input and the weight matrix. Additionally, spread factor gives it more efficiency and accuracy when the input values are fractional by decreasing the amount of it.



$$y = e^{-\frac{(t-c)^2}{2\sigma^2}} \quad 5.7$$

Figure 5.6: Gaussian membership Function

5.3 Perceptron Neural Network

Perceptron Neural Network consists of one feed-forwarded layer. The basic concept of training this type of neural networks is to update the weights when there is an error between the calculated output and the desired output (target output).

The methodology of perceptron is that the approximation is computed by training perceptron neural network **iteratively** where the desired output is computed for random inputs. By grouping results, the batch algorithm can be used to find values for weights and bias. This process may cause a lack of memory and not be time efficient with training or updating weights.

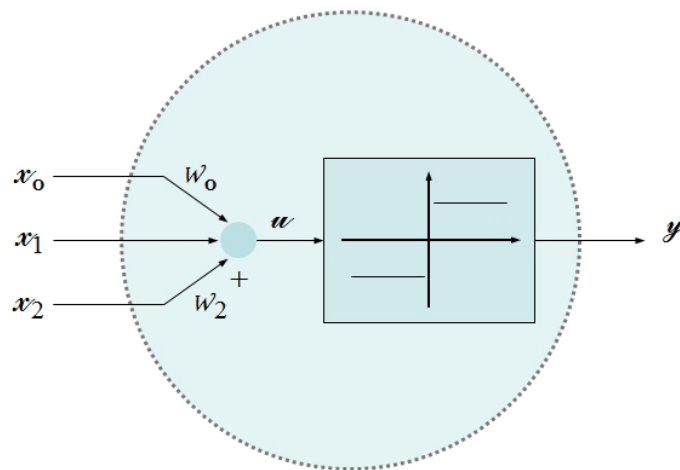


Figure 5.7: Perceptron Neural Network

The formula of updating weights in Perceptron Neural Network can be given by:

$$W_i^{new} = W_i^{old} + \alpha X_i \quad 5.8$$

Where (α) is the training factor and (X) is the input.

The disadvantage of using perceptron in the restoration process of a multi-agent system is the delay caused by training the network to find the weights. Also, in order for each update to occur within the power distribution system, the new weights must be found for the whole network. This process is similar in the multilayer perceptron.

5.4 Multiple Adaptive Linear Neural Network

MADLine Neural Network is combined of a number of single adaptive linear neural network. ADLine neural network was first found by Idrow & Hoff in 1960s to use it for bipolar output applications (-1, +1). ADLine weights are updatable and it has a bias input (activation input) with an updatable weight too.

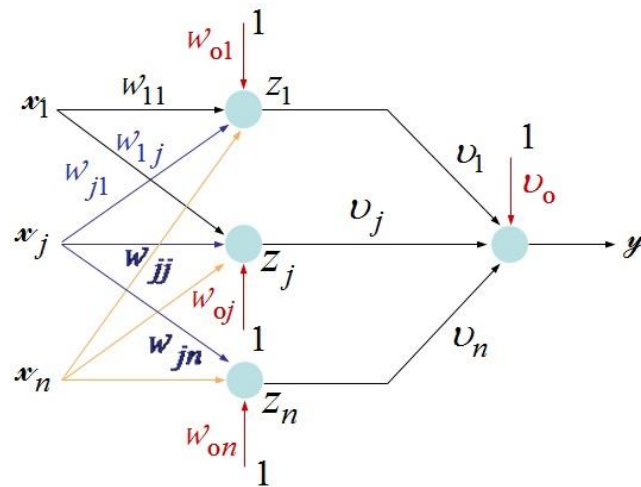


Figure 5.8: MADLine Neural Network

The training rule for each unit of the MADLine neural network is given by:

$$W_i^{new} = W_i^{old} + \alpha(t - y)X_i \quad 5.9$$

Where (α) is the training factor, (t) is the target output, (y) is the actual output and (X) is the input.

Initial Conditions:

- 1- Initial values for the weights (small values preferred).
- 2- Initial value for the training factor.

Note: when $t=y$ the old weights become the new weights after the update.

5.5 Multilayered Back Propagation Feed Forward Neural Network

It is more complicated to compute the values of weights with decreasing the number of layers and neural network constants. The multilayered neural networks have the capability to process the calculation operations for a wide range of Boolean functions rather than one-layered neural networks. Back propagation algorithm is substantially based on extracting the minimum error using gradient descent methodology. The optimal weights in this algorithm are those related to the minimum error function value which is considered the final training step. The most common activation function used with back-propagation neural network is the sigmoid function (Fig. 4.9). The calculation of the gradient of the error function is needed, which ultimately requires discontinuous function. The sigmoid function is given by the following formula:

$$y = \frac{1}{1 + e^{-Cx}} \quad 5.10$$

Where (C) is the sigmoid shape factor ($1/C$ is called the temperature factor in stochastic neural network).

Fig. 5.9 illustrates the change of sigmoid shapes for different values of (C).

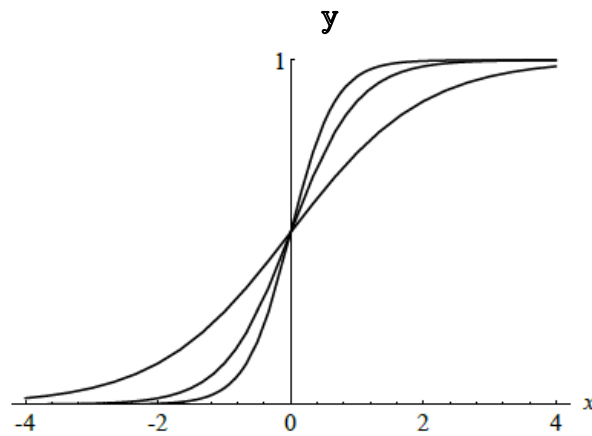


Figure 5.9: Three shapes of sigmoid (for C=1, C=2 and C=3)

The main purpose in back propagation method is to reduce the following error function:

$$E = \frac{1}{2} \sum \| o_i - t_i \|^2 \quad 5.11$$

5.5.1 Back propagation algorithm steps

In order to understand the algorithm, an extended multilayer network with a single pair of input in the hidden layer and target is presented to evaluate the error function. *R. Rojas* showed a figure of an extended multilayer network in the paper of Neural Networks 1996 as follows [25]:

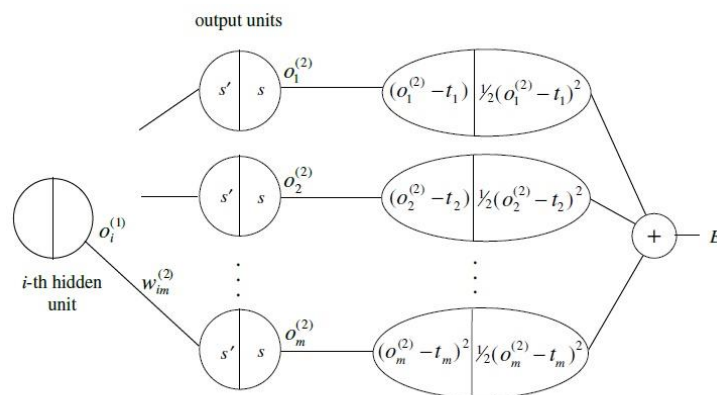


Figure 5.10: Extended Multilayer Network (Error function evaluation)

1- Feed Forward calculation step:

By applying an input vector to the network, the vectors for both hidden layer and output unit layer are calculated and retained. Activation function derivatives are stored for each unit too.

2- Backpropagation calculation step:

The first group of error function derivatives should be evaluated. *R. Rojas* stated the backpropagation path in the Fig. 4.10:

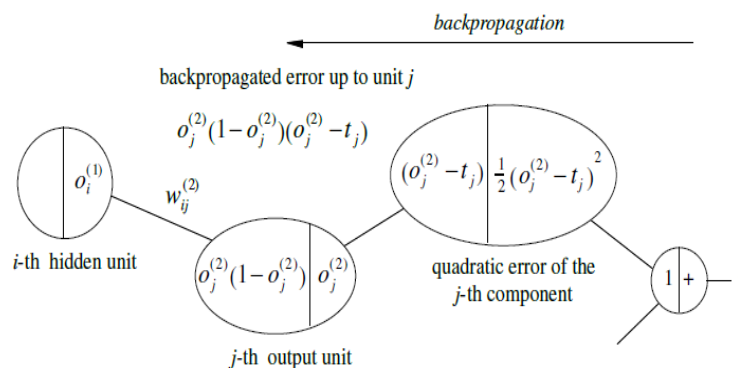


Figure 5.11: Backpropagation calculation path

To simplify the math operation of error function, backpropagation error can be defined as following:

$$\delta_j^{(2)} = o_j^{(2)}(1 - o_j^{(2)})(o_j^{(2)} - t_j) \quad 5.12$$

Then the partial derivative of error function can be extracted as following:

$$\frac{dE}{dW_{ij}^{(2)}} = [o_j^{(2)}(1 - o_j^{(2)})(o_j^{(2)} - t_j)]o_i^{(1)} = \delta_j^{(2)}o_i^{(1)} \quad 5.13$$

As a summary of backpropagation error computation, hidden unit input is multiplied by the partial derivative of error function as given below in the Fig. 5.12:

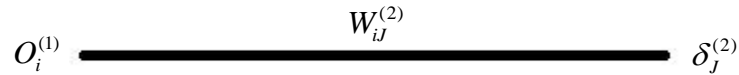


Figure 5.12: Input and Backpropagation error

3- Backpropagation to hidden layer calculation step

R. Rojas illustrated this step by stating “to compute the partial derivatives $dE/dW_{ij}^{(1)}$ each unit j in the hidden layer is connected to each unit q in the output layer with an edge of weight $W_{jq}^{(2)}$, for $q = 1 \dots m$. The back propagated error up to unit j in the hidden layer must be computed taking into account all possible backward paths”. The backpropagated error can be expressed as follows:

$$\delta_j^{(1)} = o_j^{(1)}(1 - o_j^{(1)}) \sum_{q=1}^m W_{jq}^{(2)} \delta_q^{(2)} \quad 5.14$$

Then the error function derivatives for this step can be given by:

$$\frac{dE}{dW_{ij}^{(1)}} = \delta_j^{(1)}o_i \quad 5.15$$

Backpropagated error evaluation is similar for any value in the hidden unit.

R. Rojas showed the backpropagation process in the following figure:

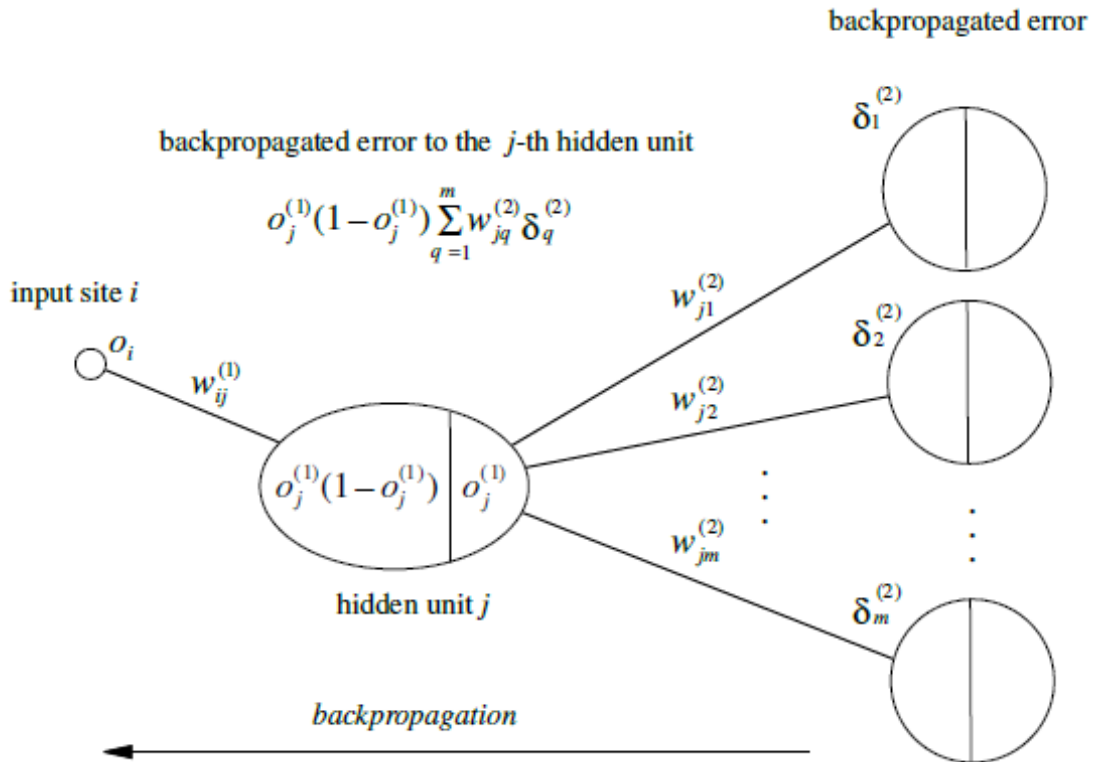


Figure 5.13: Backpropagation diagram to evaluate the backpropagated error

4- Updating weights

With considering the training factor, the weights need to be evaluated in the negative direction as the path of propagation. The change of weights can be calculated as following:

$$\Delta W_{ij}^{(2)} = -\gamma O_i^{(1)} \delta_j^{(2)} \quad \text{For } i=1, \dots, k+1; J=1, \dots, m \quad 5.16$$

$$\Delta W_{ij}^{(1)} = -\gamma O_i \delta_j^{(1)} \quad \text{For } i=1, \dots, n+1; J=1, \dots, k \quad 5.17$$

Updating weights must be after the propagated error has been found. New weights can be calculated by adding the evaluated changes to the old weights.

5.5 Radial Basis Neural Network

Radial basis neural networks require more neurons than feed-forward backpropagation networks. Radial basis neural networks [26] give more efficiency and reliability. Furthermore, there is no time wasted on training or weights updated since it is based on the distance function and finding the minimum error. The productivity of radial basis neural network is that it keeps performing accurate with more possibilities of weight matrix.

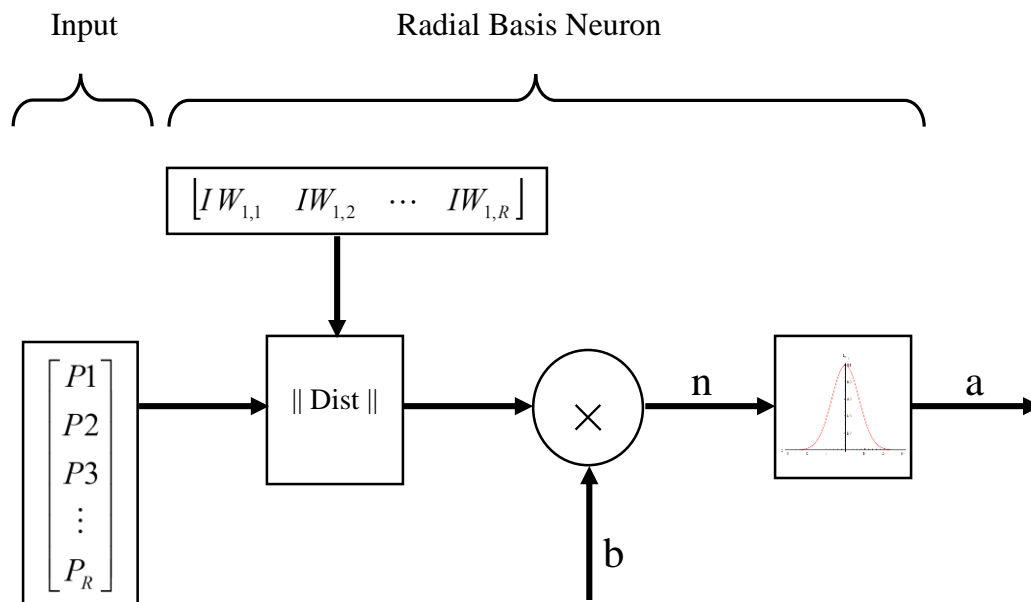


Figure 5.14: Radial Basis Neuron Unit

Fig. 5.14 shows a radial basis neuron unit with an input vector of R elements. The error vector is computed by measuring the distance between the input vector and each vector of the weight matrix. Basis vector is sufficient to adjust the sensitivity of the network. The radial basis function is used as an activation function with a maximum output of (1).

- Distance function inputs are both of the input vector and each vector of the weight matrix.
- Radial basis function input is the element-by-element product of distance function output vector and basis vector.
- Maximum output of radial basis function is (1) when the error is zero.

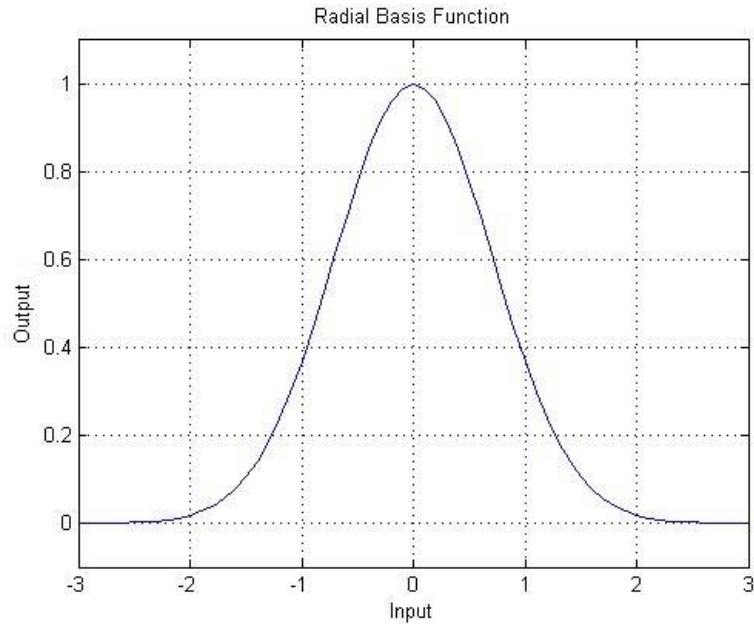


Figure 5.15: Radial Basis Function

Fig. 5.15 shows that the maximum output of radial basis function is one when the distance is zero. Moreover, decreasing the value of distance between the input vector and weight matrix-vector increases the value of output. As a result, Radbas function performs as a matching detector. The output of one is present when both input vector and one of the weight matrix vectors are identical.

5.5.1 Probabilistic Neural Network

A probabilistic neural network [27] consists of two main layers. Input layer calculates the distance vector, and then produces the error vector. Competitive layer is normally responsible for presenting the final decision. It is represented by a target matrix, radial basis matrix whose elements indicate the error values. The results are usually multiplied by target matrix.

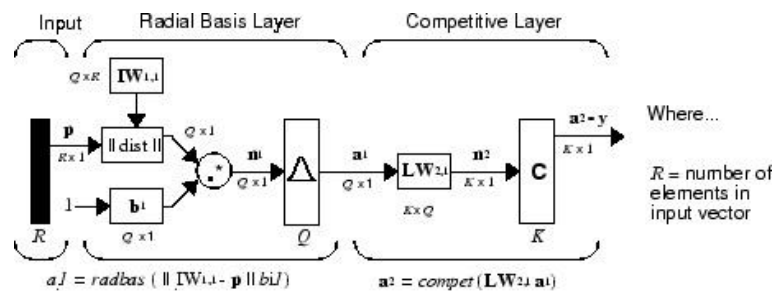


Figure 5.16: Probabilistic Neural Network structure

Wasserman showed in his paper *Advanced Methods in Neural Computing* the structure of a probabilistic neural network as in the Fig. 4.15 above.

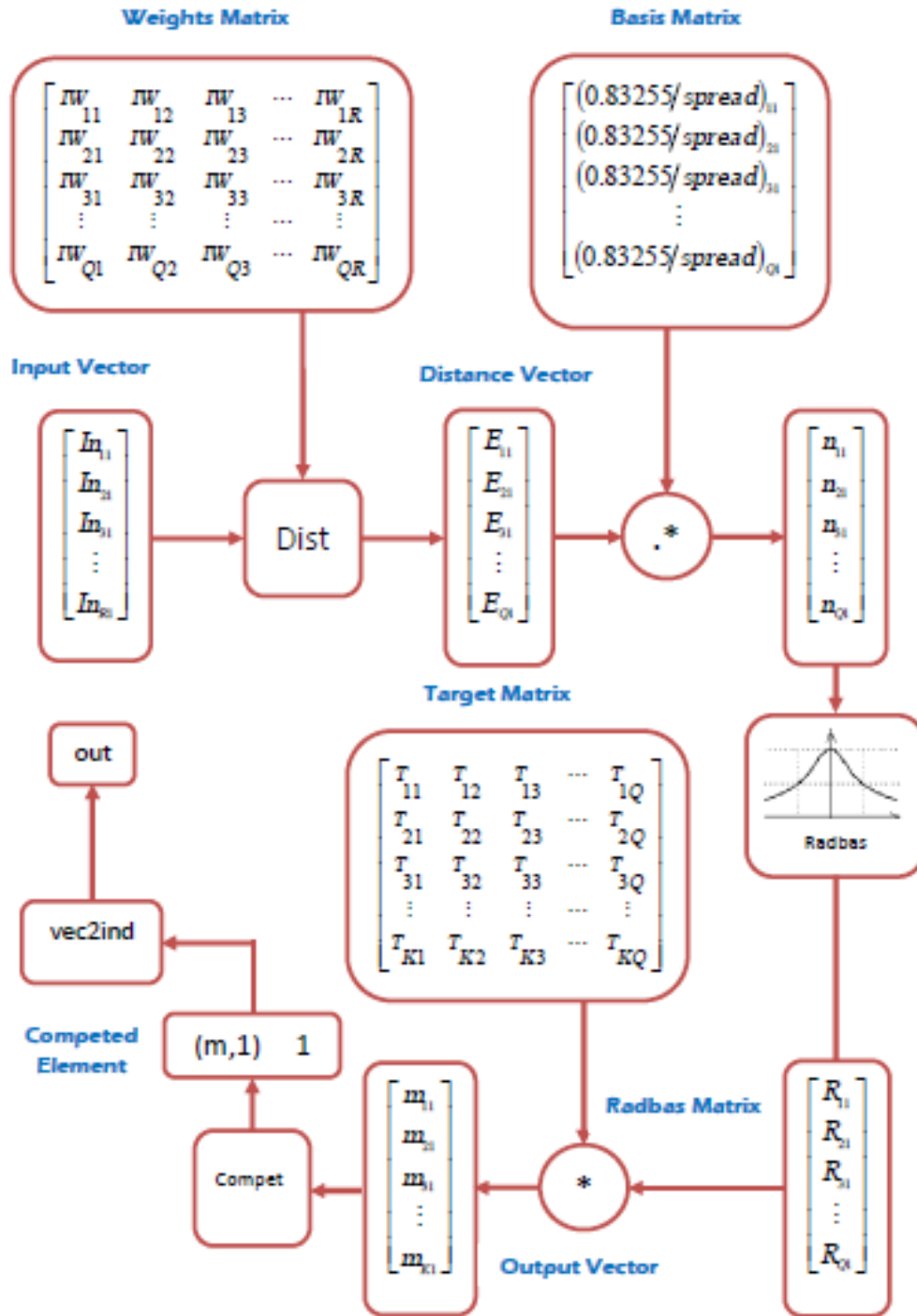


Figure 5.17: Probabilistic Neural Network step by step mathematical diagram

5.5.2 Probabilistic Neural Network Step by Step:

Fig. 5.17 illustrates step by step PNN technique and the effect of Radbas function and basis vector to obtain the desired result.

- Distance between input vector and input weight matrix is measured (input vector rows size must be same as weight matrix column size).
- Distance vector is produced (distance vector size is same as weight matrix rows size).
- Basis vector is multiplied with distance vector to adjust the sensitivity of the error vector (basis vector size must be same as distance vector).
- Basis vector is a vector whose elements are all the same value of $(0.83255/\text{spread})$ where the default of spread factor is 0.1.
- Error vector is radbased to produce a vector of numbers those indicate the identity of input vector to each vector of weight matrix (greatest number indicates the nearest vector to input vector).
- Radbas vector is multiplied by the target matrix to obtain the desired result.
- Vectorizing a matrix allows indices to be presented as vectors containing a 1 in the row of the index they represent.
- Output vector is normalized by compete function which results a vector whose number one refers to the greatest number and set the other values to zero.
- Last step is to convert the output vector from vector form to decimal form.

5.5.3 Probabilistic Neural Network Numerical Problem:

Let's consider the following possibilities of input vectors (input weight matrix):

$$P = \begin{bmatrix} 0 & 1 & 5 & 7 & 6 & 3 & 9 & 2 \\ 2 & 3 & 0 & 8 & 7 & 1 & 0 & 5 \\ 3 & 4 & 2 & 9 & 7 & 5 & 9 & 8 \end{bmatrix}$$

With corresponding target output vector:

$$T = [2 \quad 2 \quad 1 \quad 1 \quad 5 \quad 5 \quad 3 \quad 3]$$

Considering spread factor is default, basis vector would be given by:

$$b = \begin{bmatrix} 0.83255 / 0.1 \\ 0.83255 / 0.1 \\ 0.83255 / 0.1 \\ 0.83255 / 0.1 \\ 0.83255 / 0.1 \\ 0.83255 / 0.1 \\ 0.83255 / 0.1 \\ 0.83255 / 0.1 \end{bmatrix} = \begin{bmatrix} 8.3255 \\ 8.3255 \\ 8.3255 \\ 8.3255 \\ 8.3255 \\ 8.3255 \\ 8.3255 \\ 8.3255 \end{bmatrix}$$

Assuming the following input is applied:

$$In = \begin{bmatrix} 7 \\ 8 \\ 9 \end{bmatrix}$$

The distance will be computed regarding the given algorithm:

$$E = dist(p, In) = sum((x - y).^2).^0.5$$

Which gives:

$$E = \| dist \| = \begin{bmatrix} sum((0 \ 2 \ 3) - [7 \ 8 \ 9]).^2).^0.5 \\ sum((1 \ 3 \ 4) - [7 \ 8 \ 9]).^2).^0.5 \\ sum((5 \ 0 \ 2) - [7 \ 8 \ 9]).^2).^0.5 \\ sum((7 \ 8 \ 9) - [7 \ 8 \ 9]).^2).^0.5 \\ sum((6 \ 7 \ 7) - [7 \ 8 \ 9]).^2).^0.5 \\ sum((3 \ 1 \ 5) - [7 \ 8 \ 9]).^2).^0.5 \\ sum((9 \ 0 \ 9) - [7 \ 8 \ 9]).^2).^0.5 \\ sum((2 \ 5 \ 8) - [7 \ 8 \ 9]).^2).^0.5 \end{bmatrix} = \begin{bmatrix} 11 \\ 9.27 \\ 10.81 \\ 0 \\ 2.45 \\ 9 \\ 8.24 \\ 5.91 \end{bmatrix}$$

Element-by-element multiplying by basis vector:

$$Error = E.*b = \begin{bmatrix} 11 \\ 9.27 \\ 10.81 \\ 0 \\ 2.45 \\ 9 \\ 8.24 \\ 5.91 \end{bmatrix} .* \begin{bmatrix} 8.3255 \\ 8.3255 \\ 8.3255 \\ 8.3255 \\ 8.3255 \\ 8.3255 \\ 8.3255 \\ 8.3255 \end{bmatrix} = \begin{bmatrix} 91.5805 \\ 77.177 \\ 89.99 \\ 0 \\ 20.39 \\ 74.9295 \\ 68.60212 \\ 49.203 \end{bmatrix}$$

Applying Radbas function:

$$radbas_vector = \begin{bmatrix} e^{-(91.5805)^2} \\ e^{-(77.177)^2} \\ e^{-(89.99)^2} \\ e^{-(0)^2} \\ e^{-(20.39)^2} \\ e^{-(74.9295)^2} \\ e^{-(68.60212)^2} \\ e^{-(49.203)^2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Radbas vector is multiplied by vectored target matrix where vectored target matrix:

$$T_{vectored} = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix} \quad (\text{Decimals to vectors})$$

$$Outcome_{vctored} = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix} * \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

By decimalizing the outcome:

$Outcome_{decimal} = 1$ Which is corresponding to the desired output of the input vector.

5.5.4 Radial Basis Neural Networks in MATLAB:

PNN object can be created either in M.file or Simulink file in MATLAB environment. MATLAB is provided with neural network toolbox for Simulink environment which is helpful to design a Radial Basis Neural Network. Likewise, distance function, compete function and Radbas function are available blocks to use in Simulink. Decimal to vector can be designed using sequence decoder methodology. The following table shows the commonly used functions for radial basis neural networks [26]:

Function	Purpose
newpnn	Design a probabilistic neural network
net.IW{1,1}	Returns the input weight matrix
net.b{1,1}	Returns the basis vector
dist(w,p)	Measure the distance between input and weight matrix
Radbas(n)	Radbas function
ind2vec(T)	Transfer decimals to vectors
compet(n2)	Set the greatest value to 1 and zeros for other elements
vec2ind(u)	Transfer vectors to decimals

Table 5.1: Radbas NNs Common used functions

5.5.5 Probabilistic Neural Network use in Restoration:

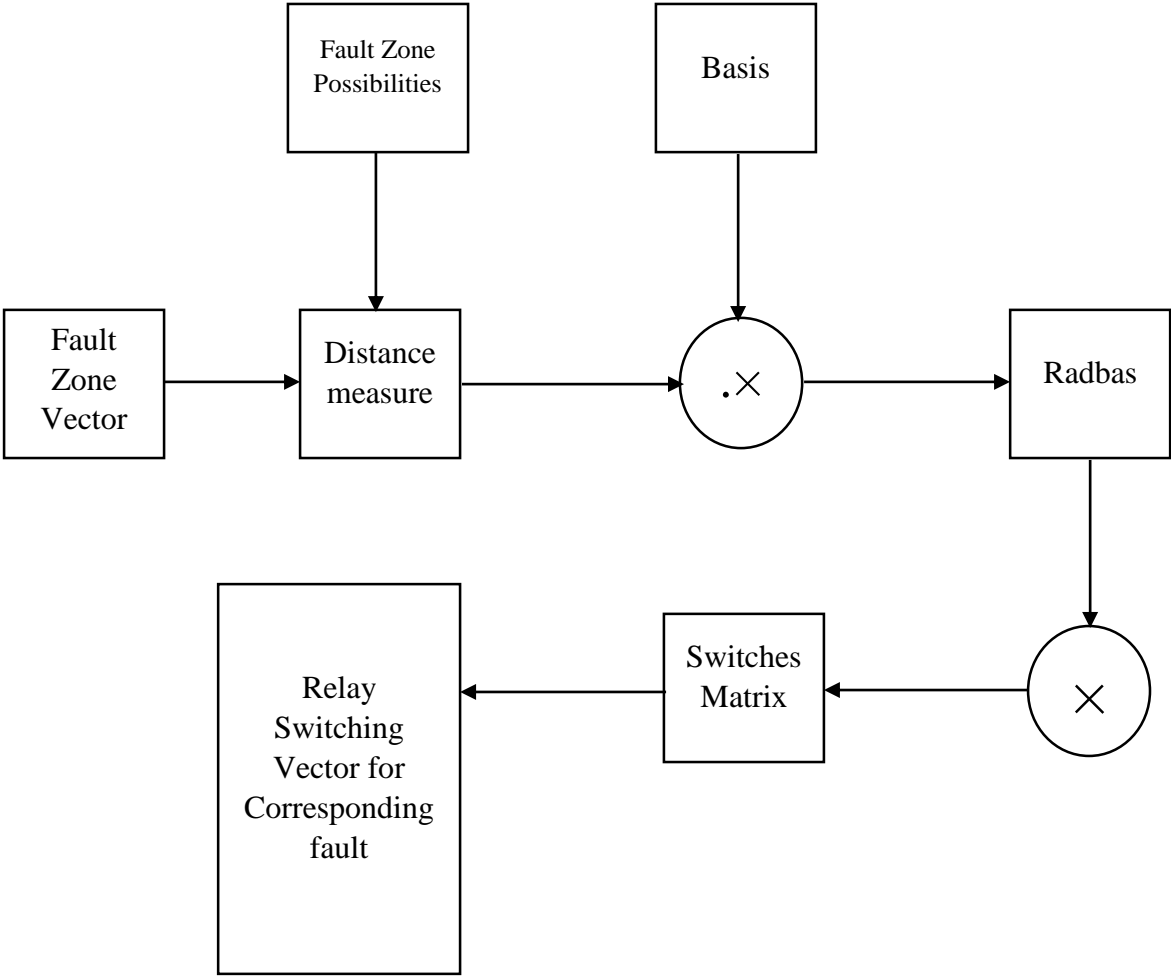


Figure 5.18: Probabilistic Neural Network in restoration diagram

6. MATLAB Simulation Studies

West Virginia Super Circuit (WVSC) has been studied and simulated to show both fault detection and restoration processes results. SimPowerSystems toolbox and Neural Network toolbox has been used to achieve this research. SimPowerSystems toolbox is provided with electrical power components and analysis tools that was helpful to design the fault detection model. In addition, Neural Network toolbox has useful functions to build up the restoration Radbas neural network model. This is shown by distance function block, compet function block and Radbas function block. S-function level (1) is used in simulation because it is flexible and has a powerful mechanism to extend the performance of a process in the simulation environment [28]. It can be written by C, C++ and FORTRAN programming languages in addition to MATLAB script and has an ability to read the input vector in a simulation file and apply a MATLAB function to it. Results and curves for fault detection and restoration are shown at the end of this chapter.

6.1 West Virginia Super Circuit

Ghorbani et al., have described WVSC in chapter (2). Layout is as follows:

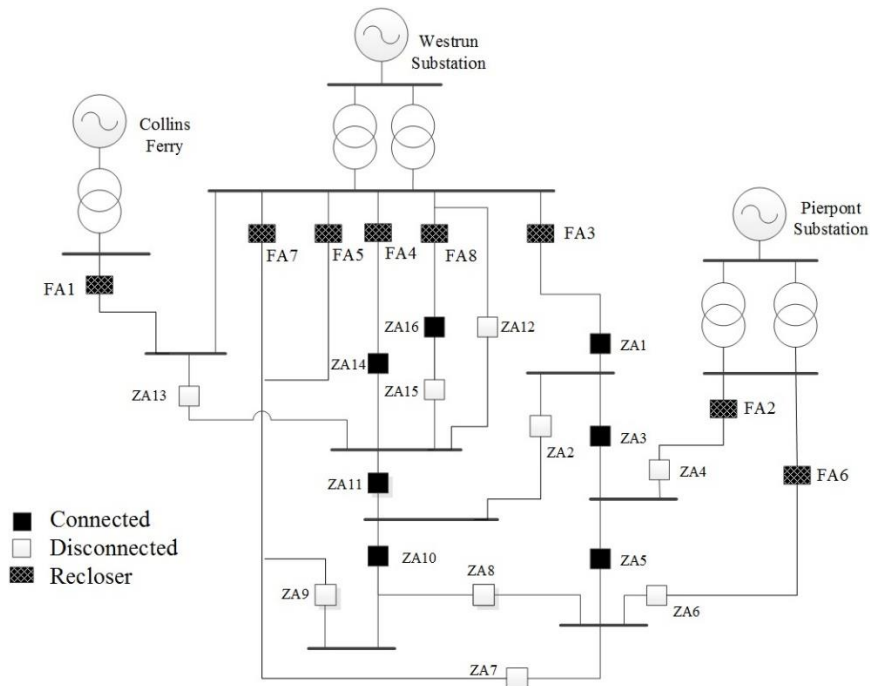


Figure 6.1: West Virginia Super Circuit

Ghorbani also stated WVSC feeders peak load data in Table 6.1 as following [4]:

Feeder#		3	4	5	6	7	8
KW KVAR	A	1916,746	1381,198	1395,-132	1036,-14	844,-25	677,133
	B	1467,455	1667,355	1664,26	773,-162	1191,167	549,69
	C	1864,697	1667,363	1429,-111	1237,85	725,-87	442,16

Table 6.1: WVSC Feeders Peak Load Data

Two reclosers have been set at two Westrun feeders to sense the fault. Nine normally open and seven normally closed Cooper DAS-15 three phase vacuum type switches are distributed in the network to process the fault isolation and network restoration functions. Transformers for each feeder are 138/12.5 KV and 33.6 MVA. WVSC is shown in Fig. 6.1.

6.2 Recloser

Recloser has been timely arranged as shown in Fig. 6.2 to close the contacts for three different durations for efficient performance in the fault sensing process. Contacts are closed instantaneously for the first duration (Duration 1) so that fault can be removed before any damage occurs. In order to make sure the fault is clear, the first time interval (TI-1) has been set to open the contacts for enough time before the duration of the second and third. Second and third durations are delayed in time so that the nearest device to fault zone has enough time to interrupt the over current.

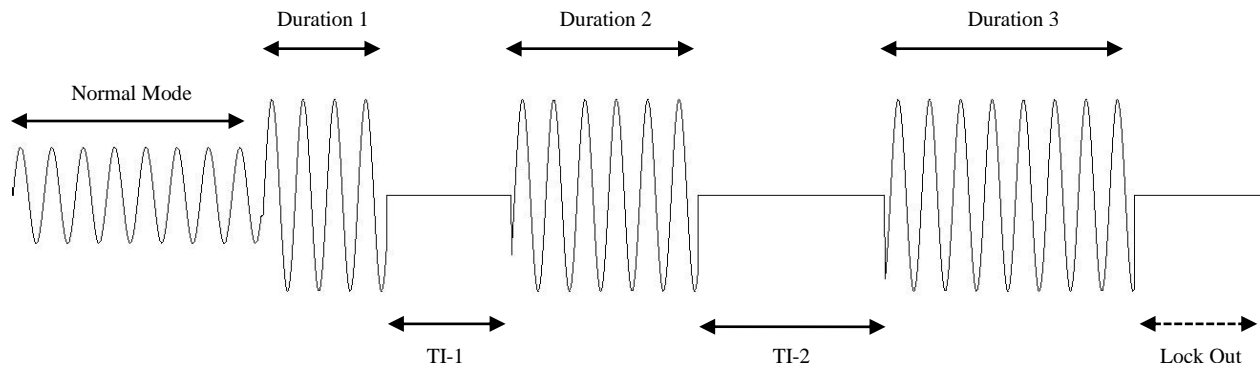


Figure 6.2: Recloser Operation Arrangement

As shown in Fig. 6.3 below, recloser has been simulated by three pulse generators shifted in time to result the desired recloser control signal for breaker. All functions are used in simulating the recloser. It has been compacted in a subsystem block with only one output pin connected to the main breaker.

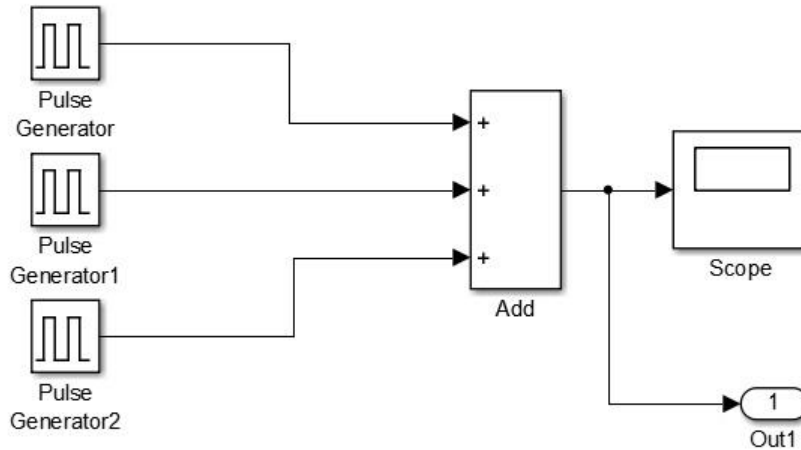


Figure 6.3: MATLAB Simulink Recloser

Simulink time has been set to 0.5s for fault detection process as shown in Fig. 6.4. First duration of closing contacts is 0.025s which is the shortest to clear the fault if it is temporary. Second and third durations have been set to 0.05s due to time constraints in MATLAB simulation.

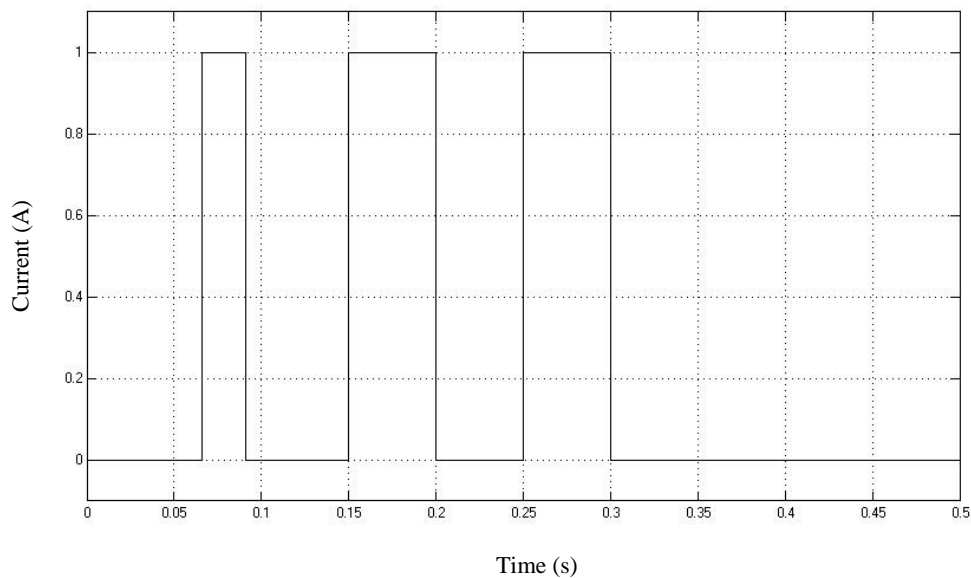


Figure 6.4: Recloser Control Signal Curve

The simulation model has been designed for West Virginia Super Circuit using MATLAB R2013a Simulink environment to monitor the total process of fault detection and restoration. Figure (6.2) presents WVSC in Simulink Environment. Four three phase power sources have been added to represent Pierpont, Westrun and Collins Ferry substations. Four transformers are also connected to the sources. Sixteen switches are used to isolate the fault and restore the other zones affected. White switches are normally opened whereas black switches are normally closed. Sixteen loads are located at each zone and have been set to meet the maximum load data given in table (6.1). Fault detection subsystem is designed with S-functions. They are programmed to calculate the impedance and evaluate currents values at each zone from the measurement tools. Five display sinks are connected to fault detection model to exhibit fault location during the process. Radbas neural network has been configured and built up in the restoration subsystem.

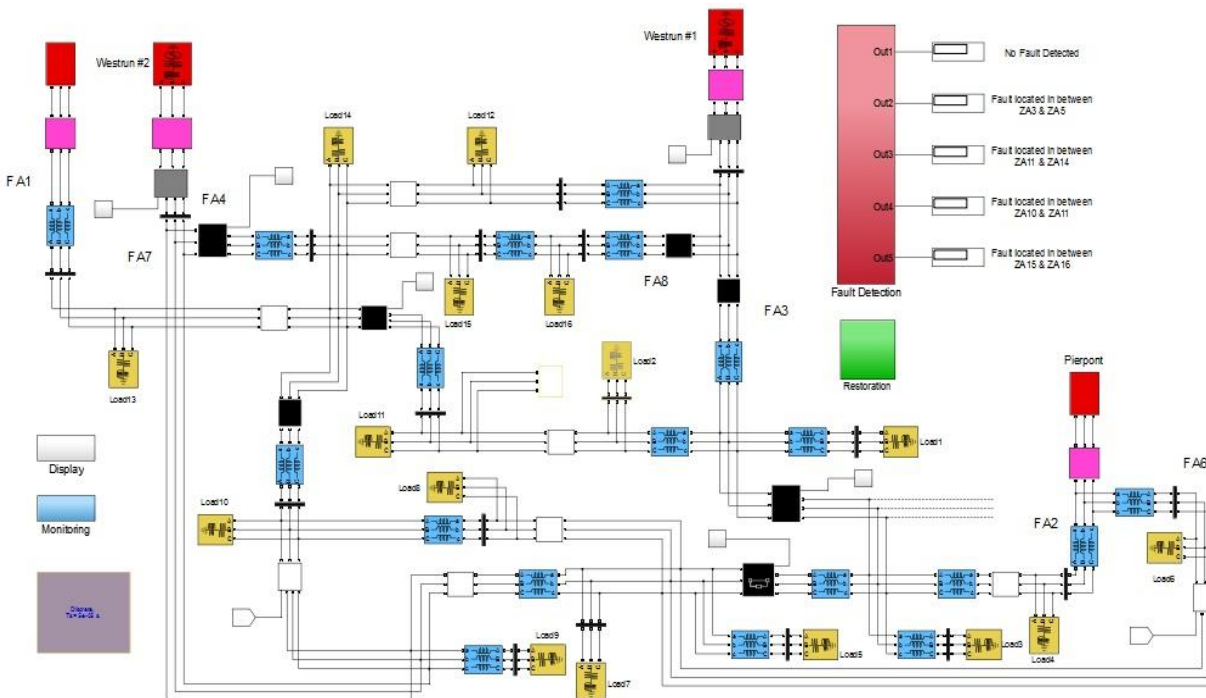


Figure 6.5: WVSC Simulink Environment

Fault detection model has been arranged as shown in figure (6.6). The design architecture is modeled to detect all kinds of faults such as ground, high current, low impedance and high impedance faults accurately. Read of all switch currents is the first step of the model; the main process is to compute the RMS values for currents and voltages. The second process is mainly to calculate the impedance and compare the RMS current values to the threshold current value at

each zone. Sampler and holder are used to hold and save the current values at the third trial of the recloser operation in case of the fault is permanent fault.

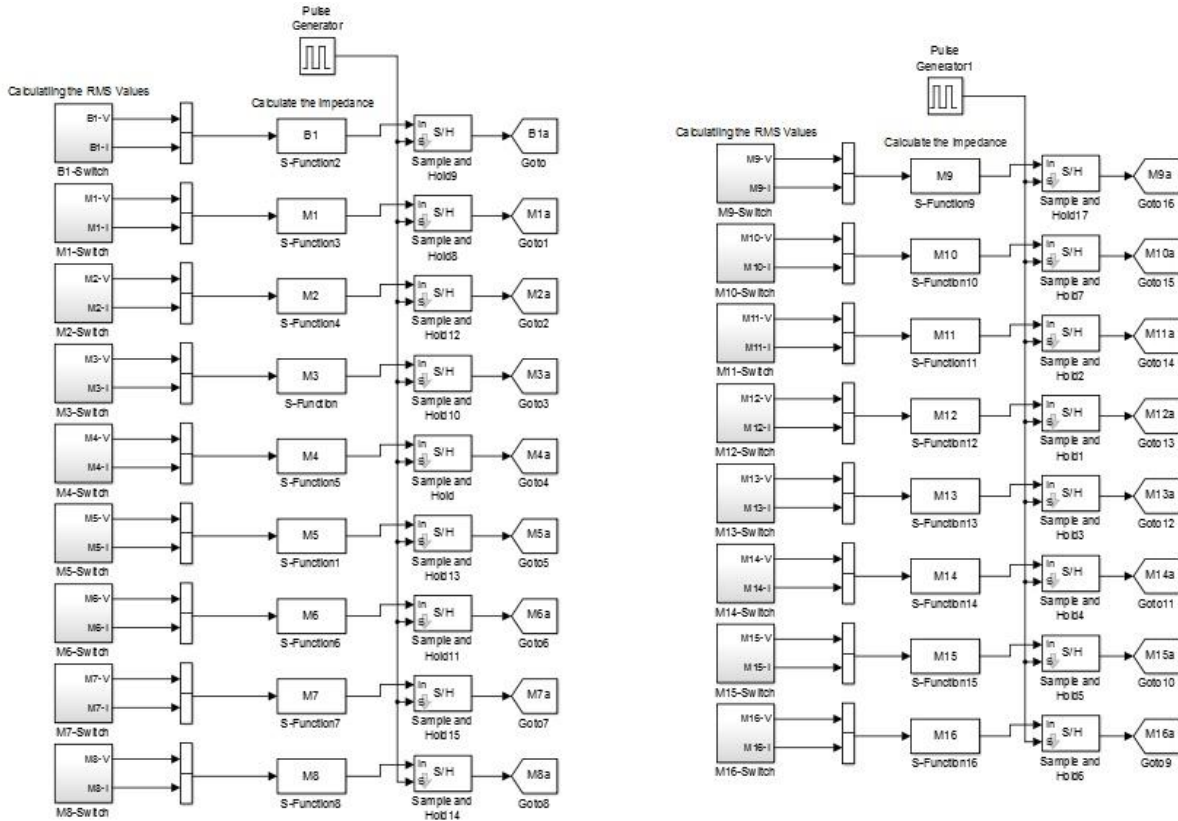


Figure 6.6: Fault detection model

Radbas is more flexible because it has the ability to design more cases and can be added to the network as much as the power distribution is extended to meet all possibilities of fault locations.

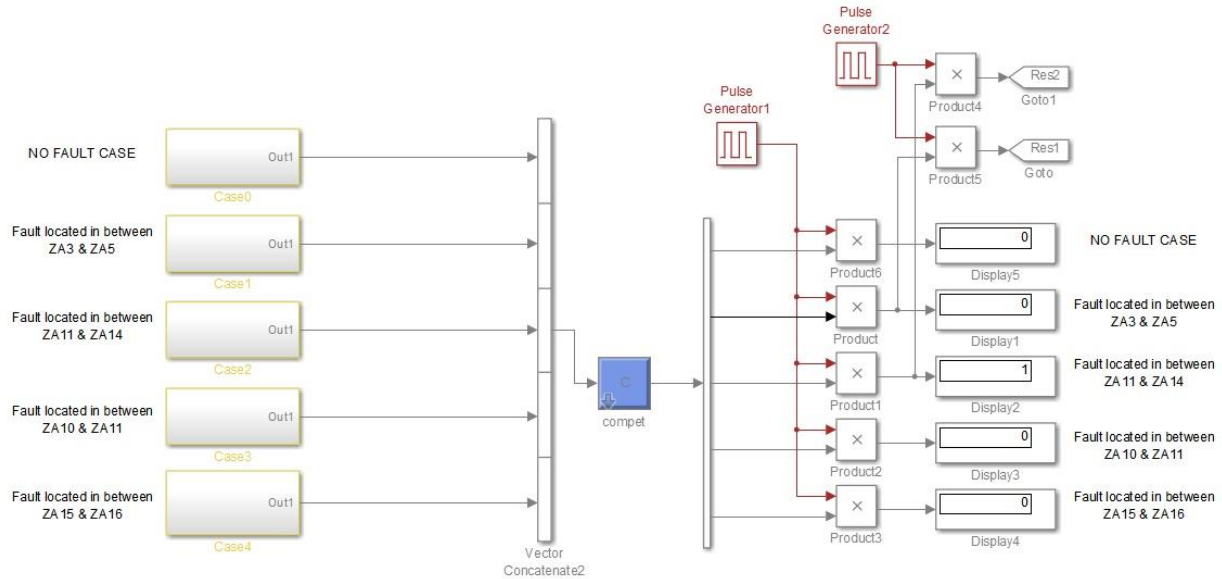


Figure 6.7: Rad bas model

In the artificial intelligence model illustrated in figure (6.7), four cases in addition to the case of no fault are simulated. Each Radbas neuron represents a case of fault location. Neurons results are concatenated for the compet block in order to generate the fault location vector.

Figure (6.8) shows one case in the network. All measurements are distanced with a fault location possibility vector. Radbas is connected to distance function to generate the value of engagement in between the two vectors.

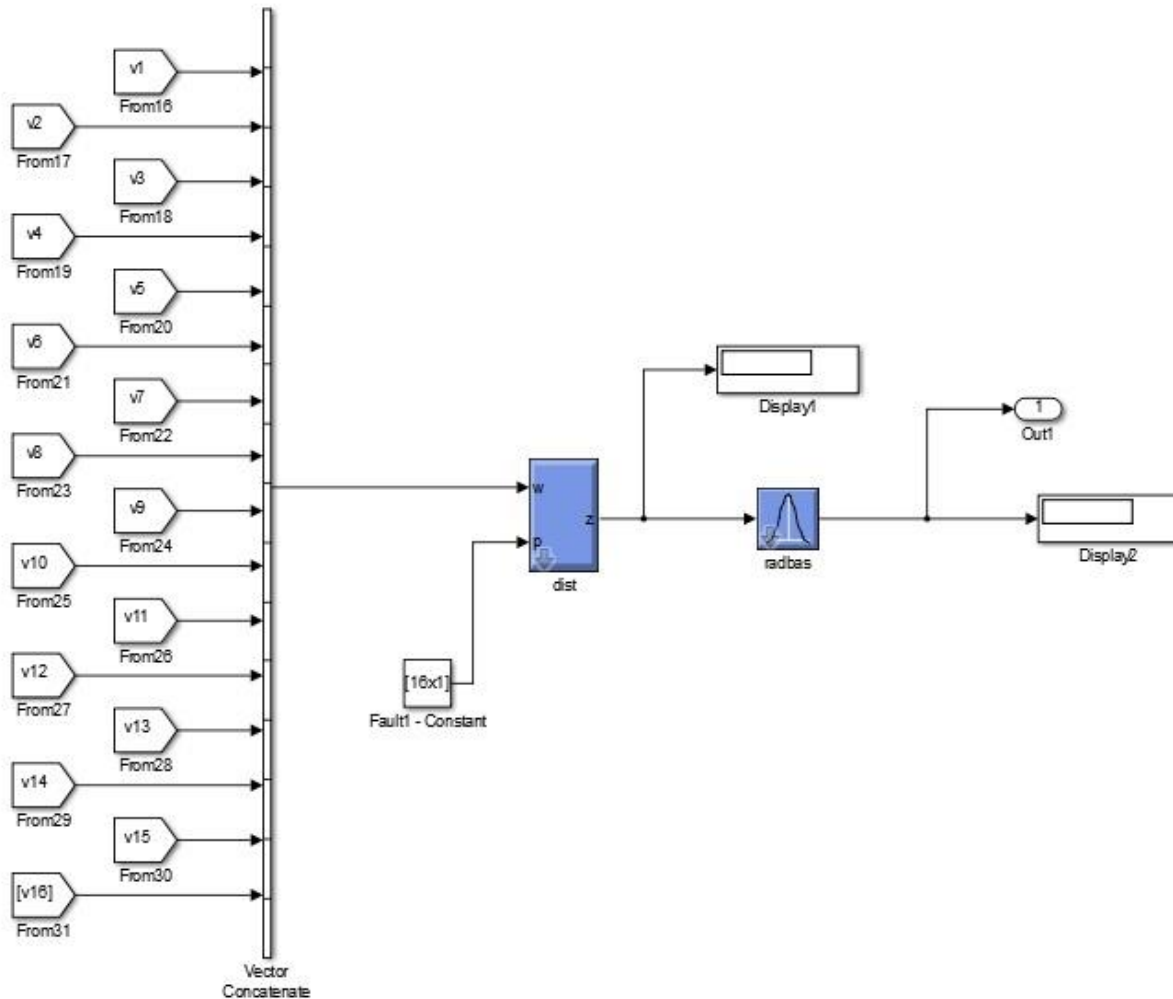


Figure 6.8: Radbas neuron for one case

Note: Vector concatenate is used to collect the read values in a vector. Output has been added to create a direct connection to network. Two display sinks are associated to monitor the results exported to the network.

6.3 Results

In order to assess the fault detection and power distribution reconfiguration results, two cases of different fault types are simulated in two locations of WVSC. Over current fault has been applied in zone 11 of feeder 4 and ground fault has also been applied in zone 3 of feeder 3. Two restoration processes for faults applied are presented.

6.3.1 Overcurrent Fault Detection Isolation and Restoration Case

A three phase over current fault affects zone (1) are shown below in Fig. 6.9 [4]. Recloser contacts will respond by opening the contacts to start the protection process. The first trial with the shortest duration is to sense the fault and clear it in case it is temporary. The second trial is longer in duration to confirm the existence of the fault and call the fault detection function to start down streaming data. Final recloser trial is more protracted for holding required data to detect the fault. The restoration process will start after acquiring fault vector from fault detection model. Power will be restored to zone (3) and zone (5) from feeder (4) through switch zone (2).

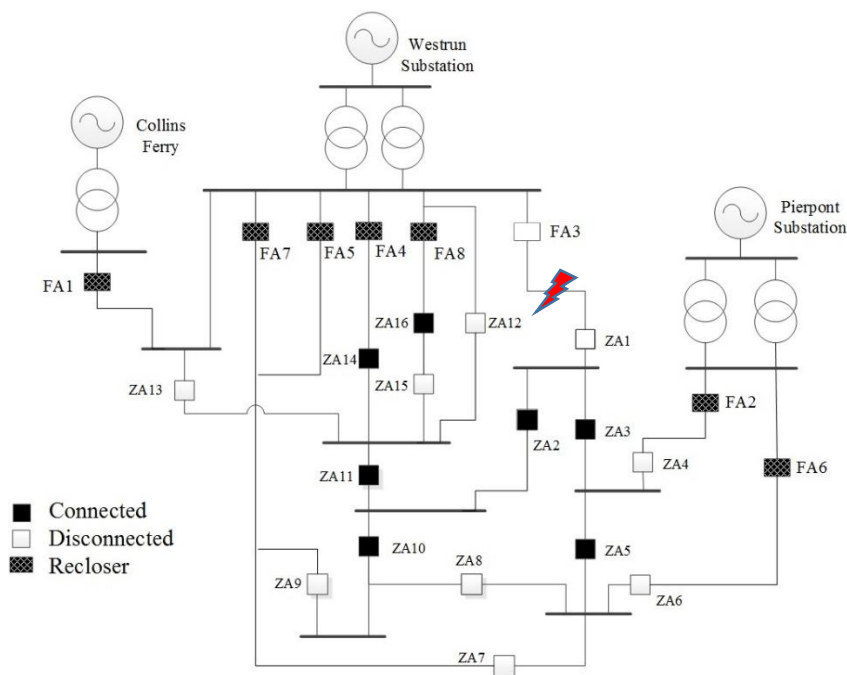


Figure 6.9: Over current fault case in zone (11)

Recloser current is represented in both Fig. 6.10 and Fig. 6.11. The fault affects feeder (3) and zone (1). Restoration has been exhibited by plotting the current through zone (3) and zone (2) current in Fig. 6.12 and Fig. 6.13 respectively.

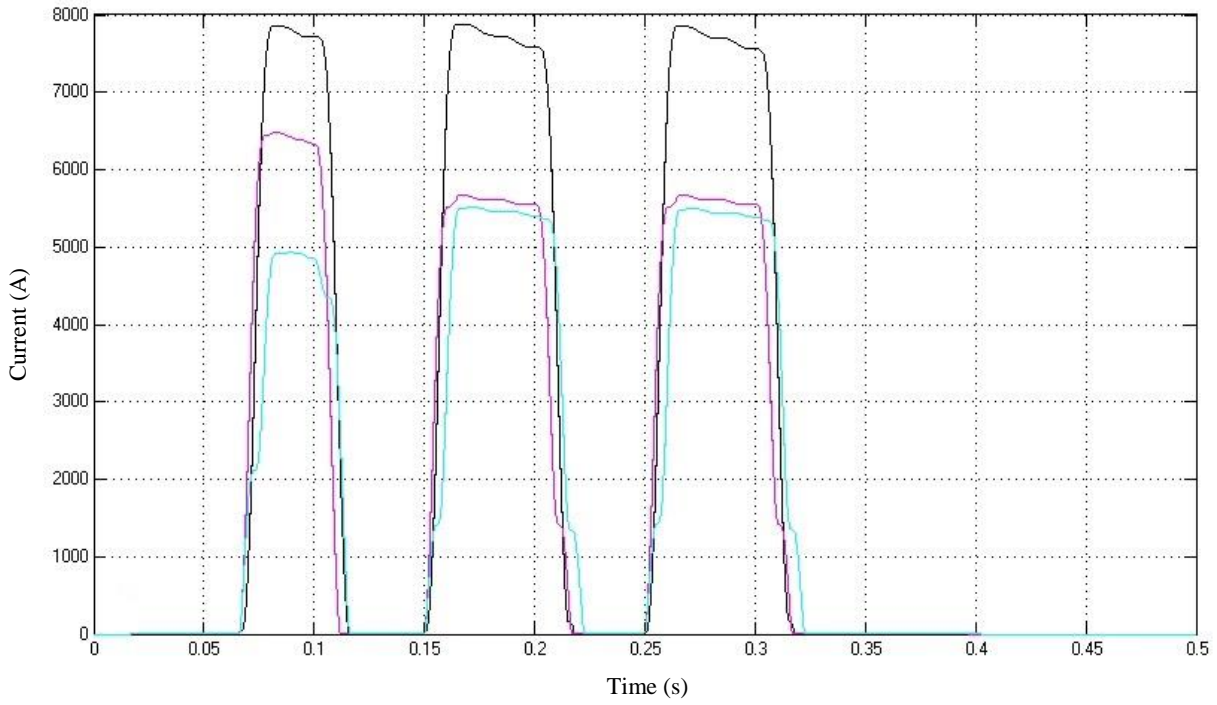


Figure 6.10: three recloser trials current fault case at feeder (3)

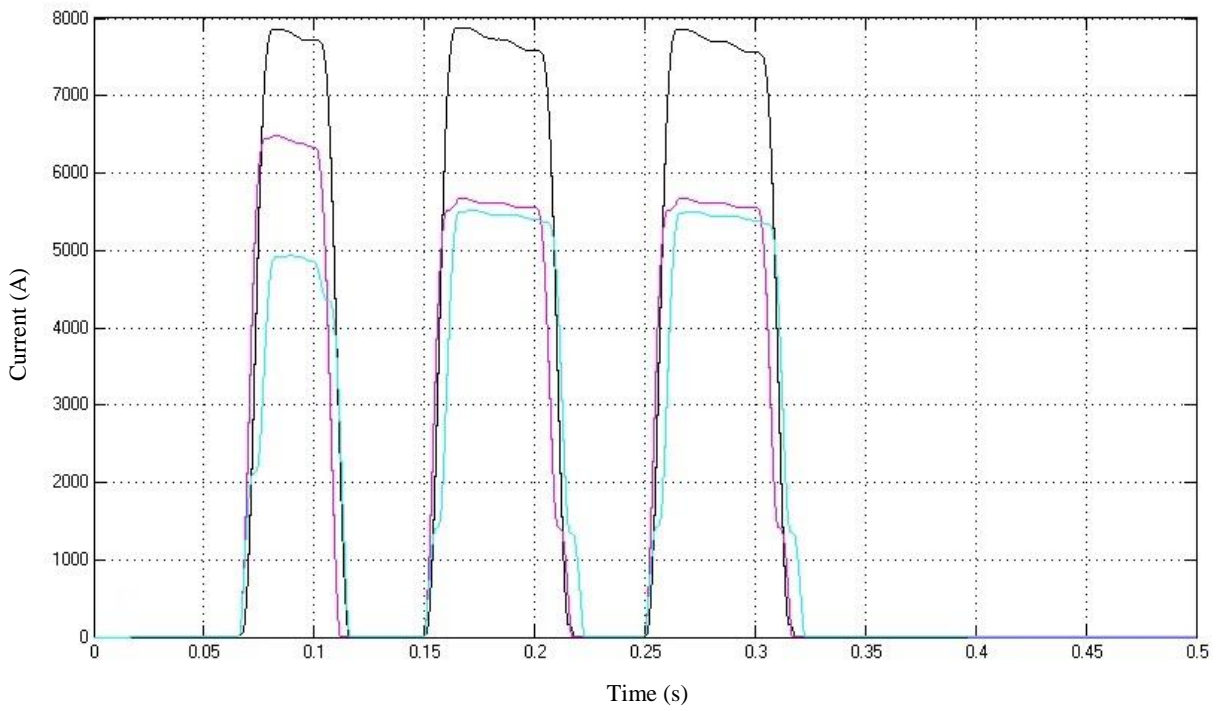


Figure 6.11: three recloser trials current fault case in zone (1)

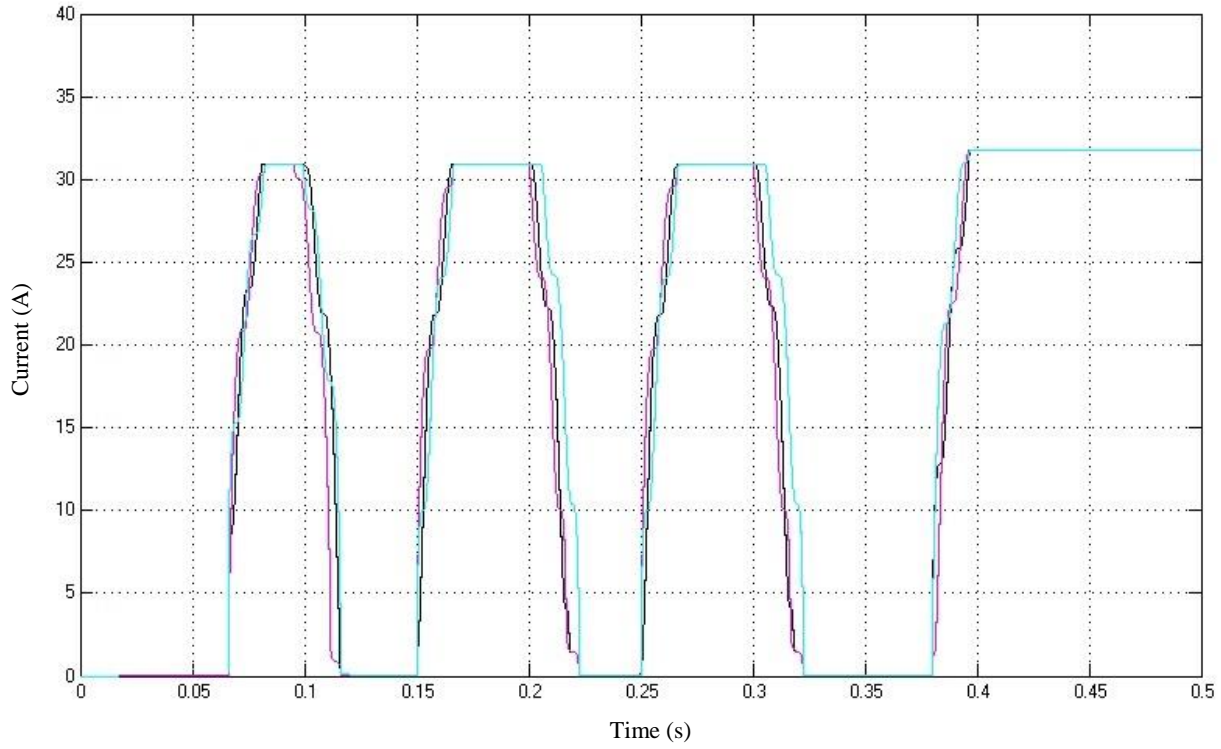


Figure 6.12: Zone (3) current after restoration

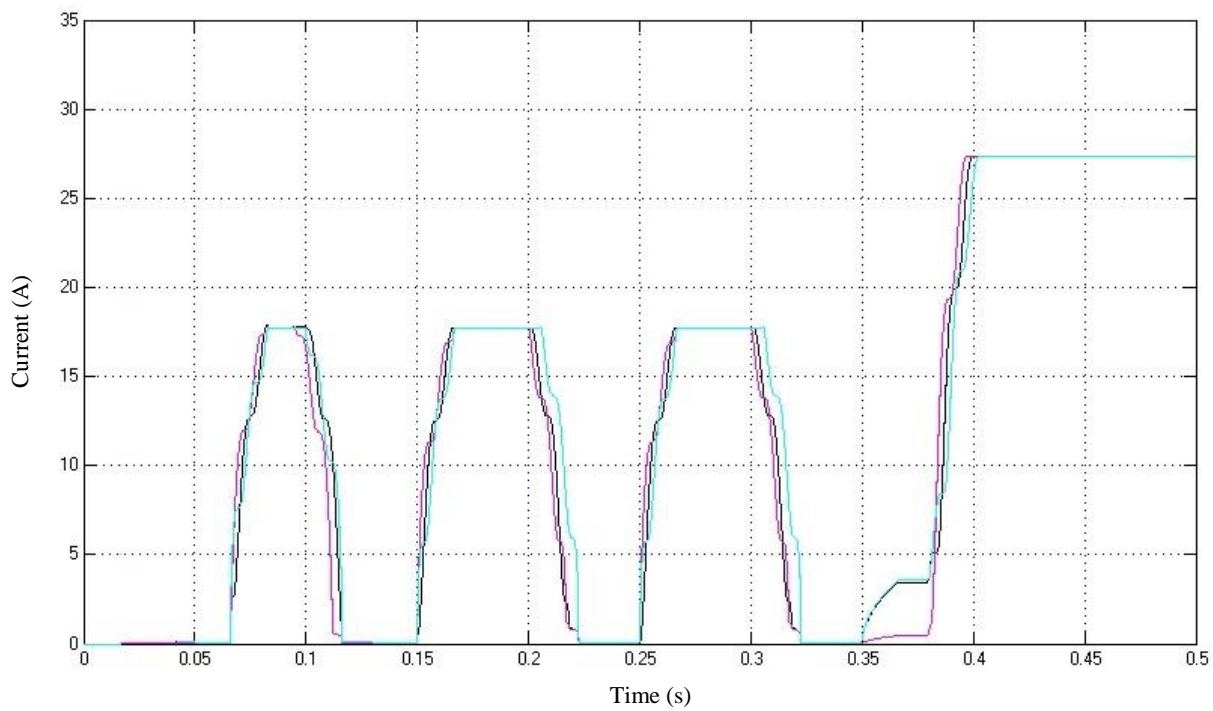


Figure 6.13: power restored through Zone (2)

6.3.2 Ground Fault Detection Isolation and Restoration Case

A three phase ground fault affects zone (3) are shown below in Fig. 6.14 [4]. Recloser contacts will respond by opening the contacts to start the protection process. The first trial with the shortest duration is to sense the fault and clear it in case it is temporary. The second trial is longer in duration to confirm the existence of the fault and call the fault detection function to start down streaming data. The final recloser trial is more protracted for holding required data to detect the fault. Restoration process will start after acquiring fault vector from fault detection model. Power will be restored to zone (5) from feeder (2) through switch zone (4).

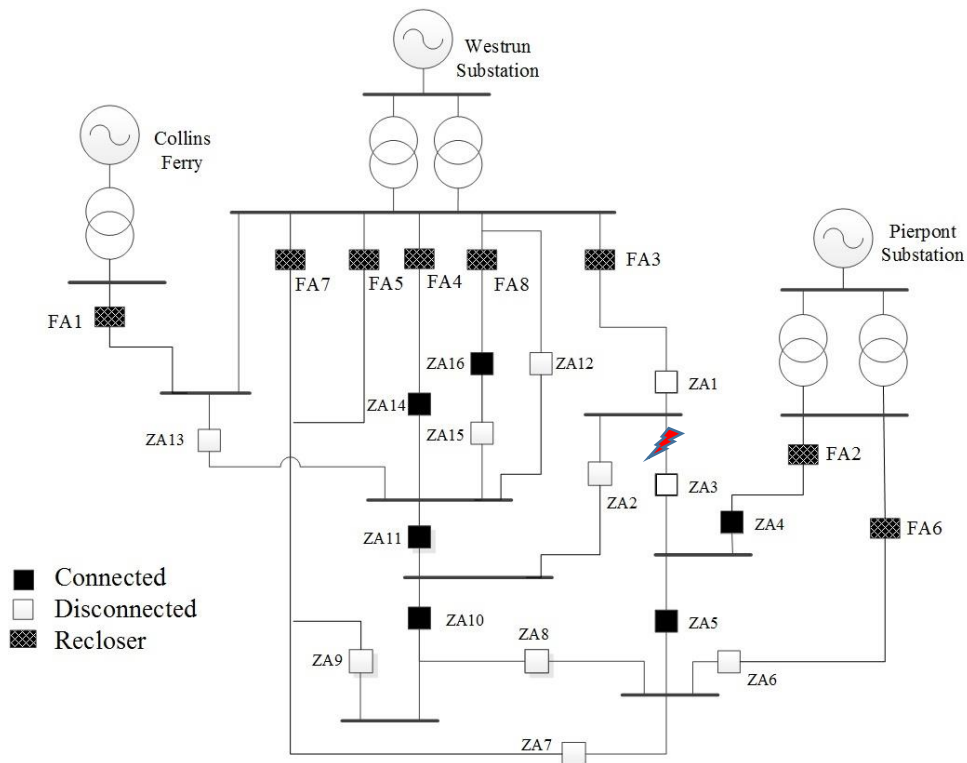


Figure 6.14: Ground fault case in zone (3)

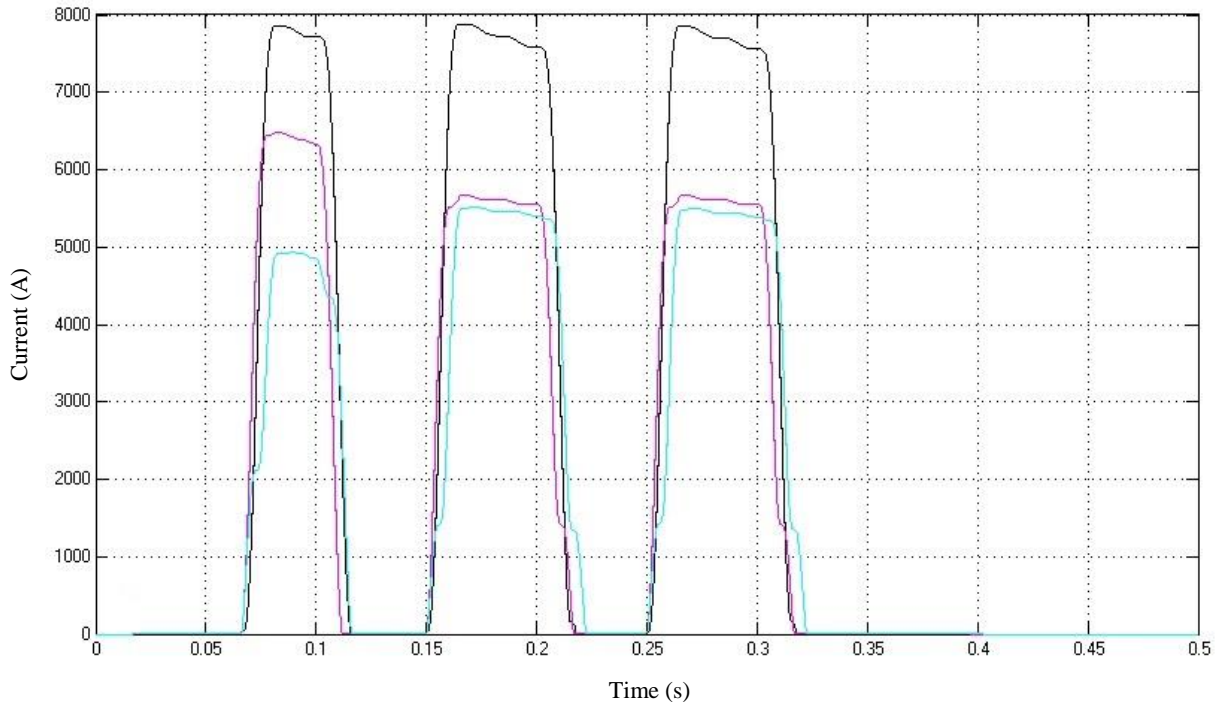


Figure 6.15: Zone (3) Ground Fault three trials trip

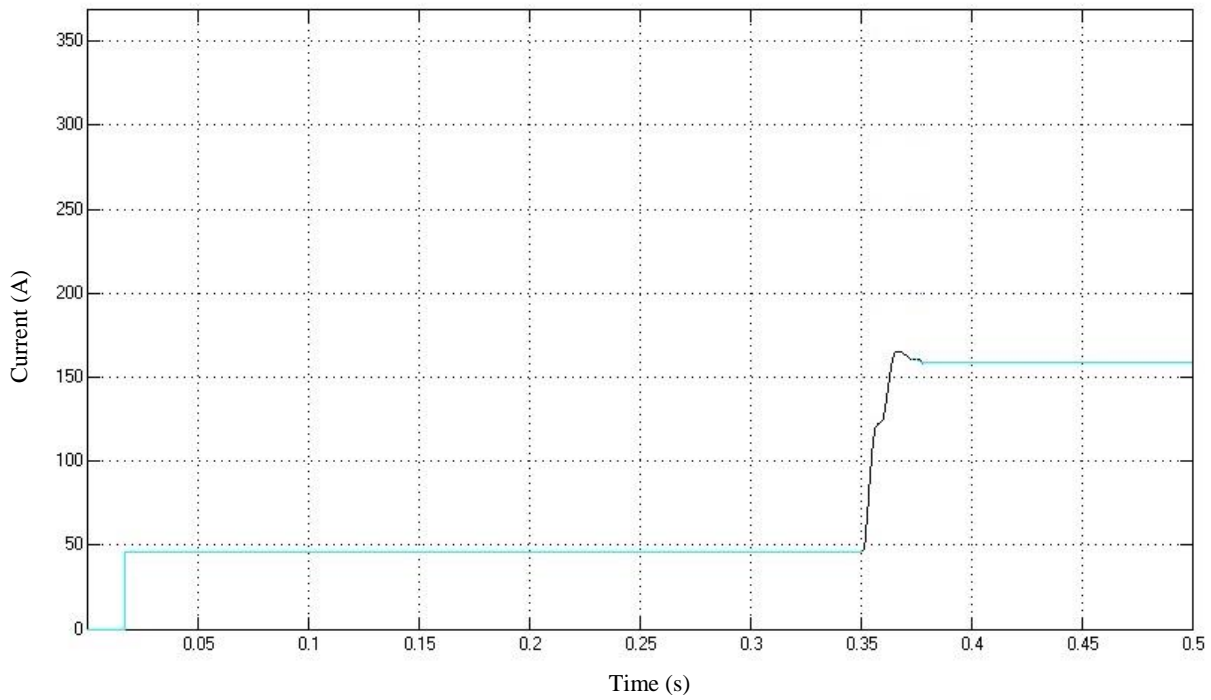


Figure 6.16: Zone (4) Normal Mode and restoration

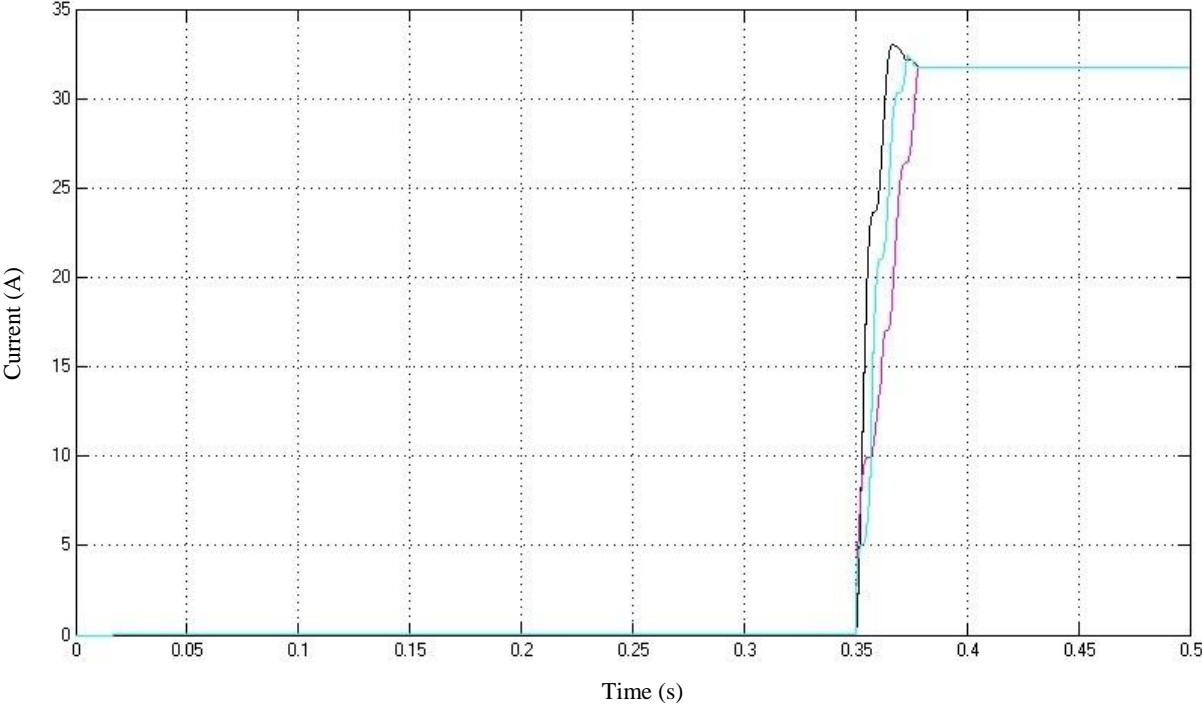


Figure 6.17: Restoration of zone (5)

7. Implementation

A multi-agent system design for fault detection and restoration in a power distribution network has been implemented for three zones and two feeders. Three computers are used as control units to read data and make the right decision to detect and locate the fault before reconfiguration. Three measurement tools are connected to the computers to send measurements read. They are also set to store samples at high accurate rating to plot the whole process. Solid state relays have been located as three at each zone. Three relays also have been designated to represent a protection recloser at the first feeder.

7.1 Multi-Agent System Fault detection and Restoration

The proposed Multi-Agent System design for fault detection, location and restoration is briefly described in this section. Algorithm and flow chart for the implemented approach are also presented.

7.1.1 Multi-Agent System Structure

Figure (7.1) shows the proposed MAS design for implementation. Each zone is provided with processing unit that is equipped with LAN Ethernet ports and math software to apply Fault detection and restoration algorithms.

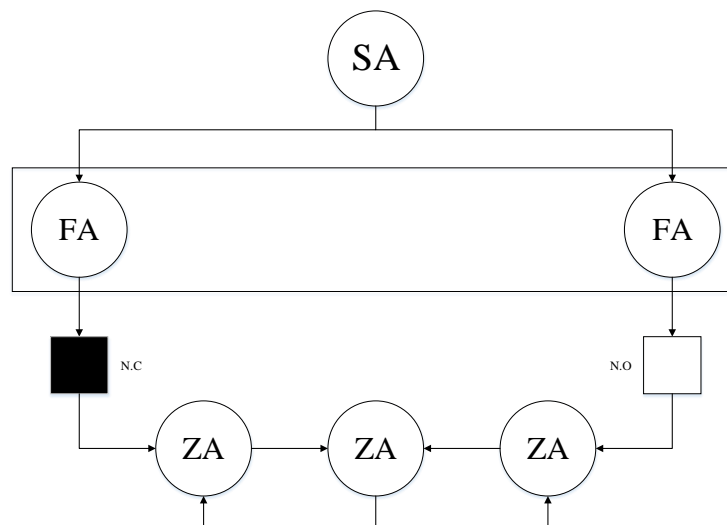


Figure 7.1: MAS structure implemented

Two feeders are supplying the agents from two locations. Recloser has been installed at the left feeder. Each zone is protected by three phase relay switch in case of fault occurs. Zone agent consists of a control unit that has the capability to process and evaluate the fault and run both fault detection and restoration algorithms. Figure (7.2) represents a diagram for each zone agent.

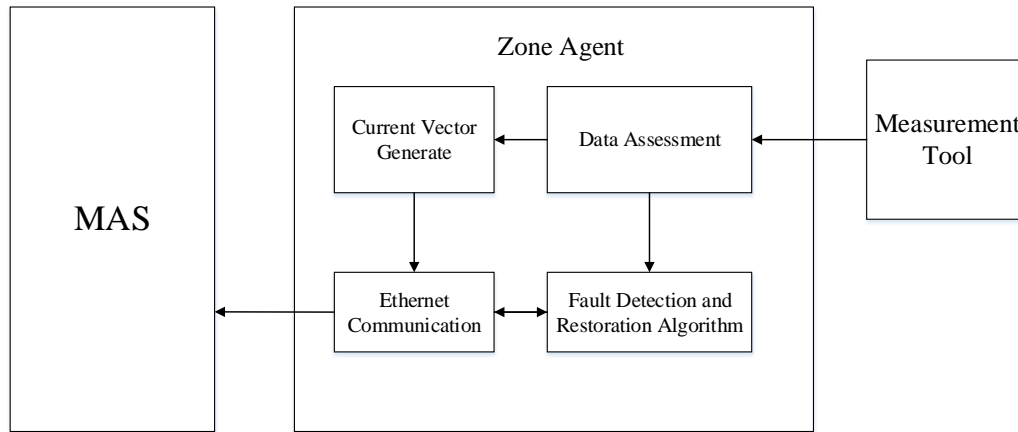


Figure 7.2: Zone agent diagram

7.1.2 Fault Location Equations

Current vector for all agents must be generated so that desired calculations can be done at all zones. Each zone is designated to collect its current values and send it to the next neighbor. Current supplied to each zone is calculated by using Kirchoff's law:

$$I_{Zk} = \sum_{i=1}^n I_{Enter} - \sum_{i=1}^m I_{Leave} \quad 7.2$$

Each zone agent is responsible for computing the change of current value from the generated current vector. Ethernet communication is established in between each two neighbors in order to exchange the current values to evaluate the change of current. This is shown in the following equation (7.1):

$$I_{Zc} = \left(\frac{|I_{Znew}| - |I_{Zold}|}{|I_{Zold}|} \right) * 100 \quad 7.1$$

7.1.3 Fault location and restoration Flow chart diagram

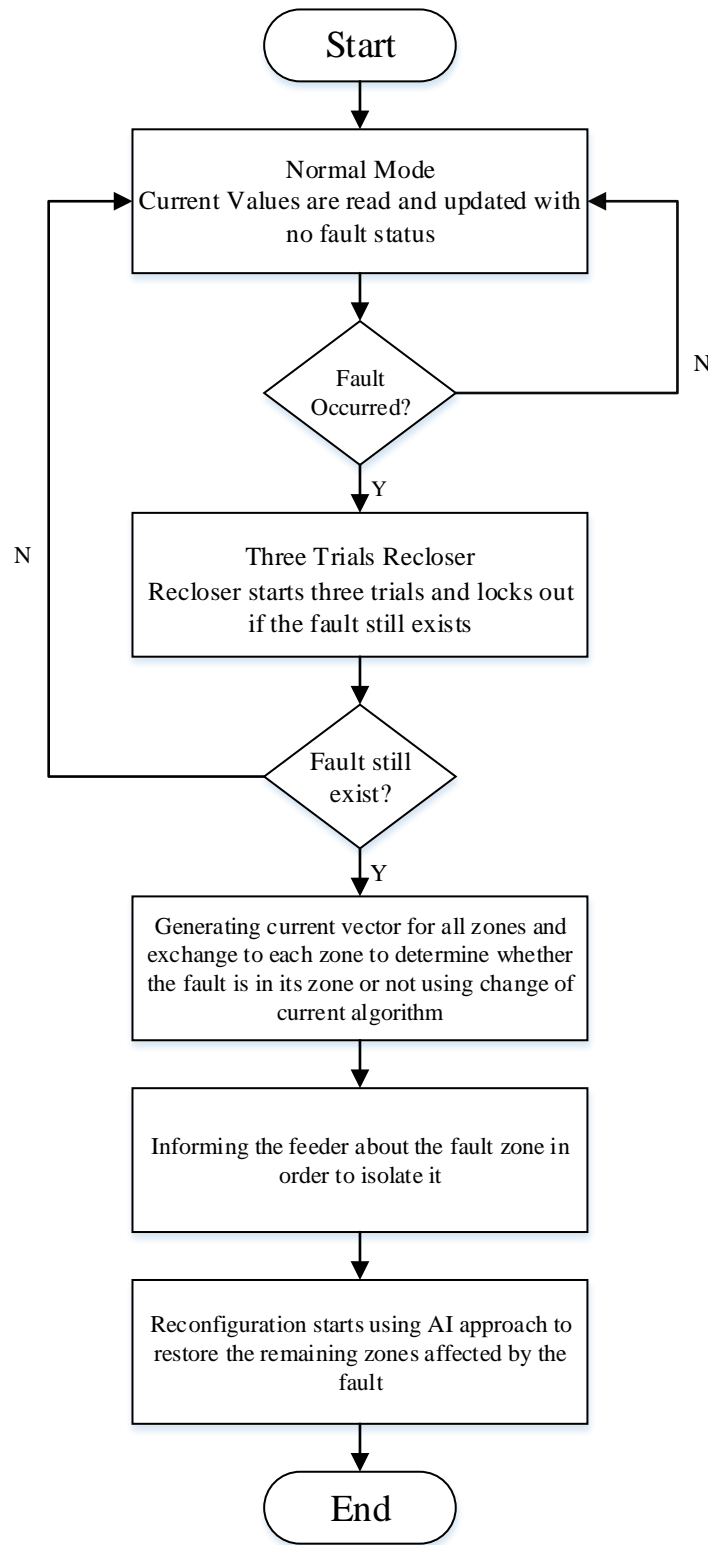


Figure 7.3: Fault location, isolation and restoration flowchart

7.1.4 Sequence diagram

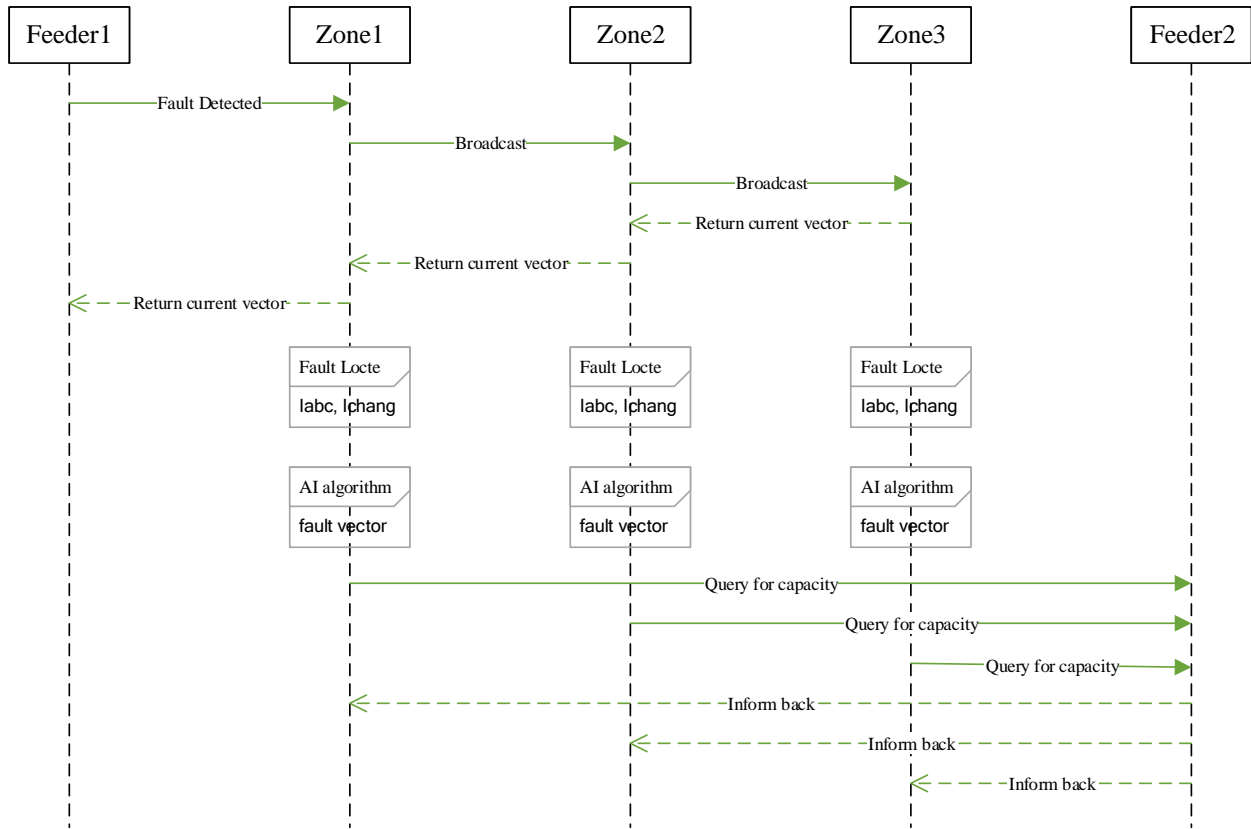


Figure 7.4: Sequence diagram

7.2 Components

High CPU performance computer with five Ethernet ports and four serial ports is added at each zone to process the proposed algorithm. An accurate power quality meter that has an ability to monitor and record values at high sampling rate is connected to the computer. MATLAB R2013a supported with NNET and SimPowerSystems toolboxes is installed on each computer. Ethernet connection has been established between each two zones.

7.2.1 SEL measurement tool

SEL 735 and SEL 734 are reliable highly accurate power meters which can be arranged to monitor the current, voltage, power and energy values. They are also provided with a QUICKset software to manage storing data values at high precision chart.



Figure 7.5: SEL 735 & SEL 734 used to implement a Multi Agent system design

SEL measurement tools are provided with two main access levels. First access level (ACC) is to enable the setting of SEL 735 for desired operation while the second access level is to enable the using of control ports and get RMS values. SEL is also a controller with the back control ports to activate a certain function regarding the reading of parameters [29]. Access level 1 is avoided since a buffer has been designed to control the switches used in the Lab experiment. SEL relays can communicate through Ethernet and serial port RS232.

Table (7.1) below shows the common commands used in SEL 735

Access Level	Command	Description
0	ACC	Move to Access Level 1
0	BNA	Binary Names
0	CAS	Compressed ASCII data Configuration
0	DNA	Compressed names
0	ID	Compressed ASCII Fast Meter ID
0	QUI	Quit to Access Level 0
0	SNS	Compressed SER settings
1	2AC	Move to Access Level 2
1	EVE	Latest event report, 16 samples per cycle
1	COM	Display MIRRORRED BITS channel statistics
1	COM C	Reset MIRRORRED BITS channel statistics
1	CHI	Compressed history
1	DAT	Show date
1	MET RMS	Display rms metering data
1	MET M	Display maximum and minimum values
1	MET	Display instantaneous metering data
1	TIM	Show time
E	-	Meter Parameter Setting
C	-	Troubleshooting and diagnosis

Table (7.1): SEL Common Commands

7.2.2 Control Units

Nuvo- 1000 Industrial grade computer with fanless design gives it exceptional long-term durability and tolerance to vibration. “Its -25/70 temperature range provides a more reliable operation in various environments”. With core i7 processor, this control unit gives a satisfying response of protection while used in multi-agent design [30]. Fig. 7.6 shows the Nuvo 1003s model that is used in the laboratory experiment:



Figure 7.6: Nuvo 1003s Industrial Grade Computer

The specifications of Nuvo-1003s are given in table (7.2) below:

Processor	Intel® Core™ i5-520M - 2.66 GHz Freq – 4MB Cache
Memory	2GB DDR3 extendable to 4GB DDR3
Power Consumption	Maximal: 3.1A @ 19V (58.9W)
Operation Temp	-25°C ~ 70°C
Ethernet	5x Intel ® 82574L GbE ports
Serial Port	1x P RS-232 & 3x RS-232 (COM1-COM4)
USB	6x USB 2.0 ports

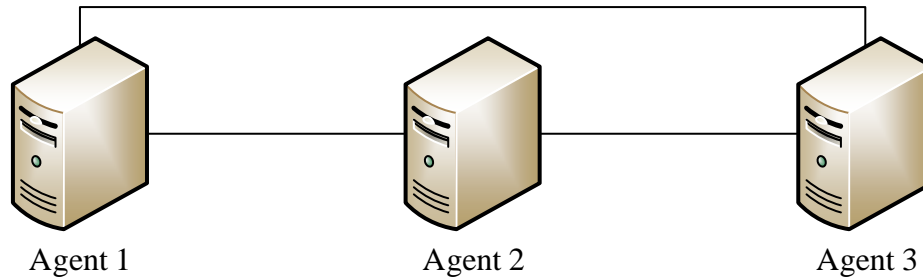
Table (7.2): Nuvo-1003s Specifications

7.2.3 Ethernet Communications

Communication protocol is established via Transmission Control Protocol/Internet Protocol (TCP/IP). Server/Client methodology is applied when feeder sends a message to next agent and same methodology is used when an agent starts exchanging with next agent as well. Feeder is a client when receiving message from any agent.

Agents were connected with LAN Ethernet Protocol as peer-to-peer connections using crossover cables, where every node is connected to the previous node and to the following nodes.

Messages between nodes flows starting from the feeder and go through the network to all the nodes.



The following table (7.3) shows IP Addresses used as static address for all the nodes. “Agent at row X can send a message to agent on column (Y) using the address at cell (X, Y) in the table”

	Agent 1	Agent 2	Agent 3
Agent 1		192.168.1.187	192.168.1.185
Agent 2	192.168.1.87		192.168.1.184
Agent 3	192.168.1.85	192.168.1.84	

Table (7.2): IP Addresses

7.2.4 Software

MATLAB R2013a is available on each Industrial grade computer Nuvo 1003s to program the multi-agent design and has a Neural Network toolbox to code the Radbas neural network function for restoration. Code has been divided to multi-task function so that trouble-shooting can be quickly done by user and MAS design can be extended in the program. Main function has the all responsible functions and can show step-by-step the process of the proposed approach. Functions are ordered as following:

- Main function: this function includes and runs all functions.
- Initialize: reset and clear ports in case it is already open.
- Connect to SEL: establishing Serial connection for SEL measurement tool.
- Connect to TCP: establishing Ethernet connection for next agent.
- Read Current: import current values from SEL to control unit.
- Detect fault: this function senses fault at the feeder wherever it occurs.

- Three trials: timing arrangement of the three trials recloser can be set in this function.
- Recloser lock out: locks recloser if the fault still exists.
- Process message: current values are sent to next agent to build current vector.
- Locate_fault and restore: this function mainly recalls both fault detection and apply PNN algorithms functions.

The following diagram illustrates the lab experiment demonstration for MAS design:

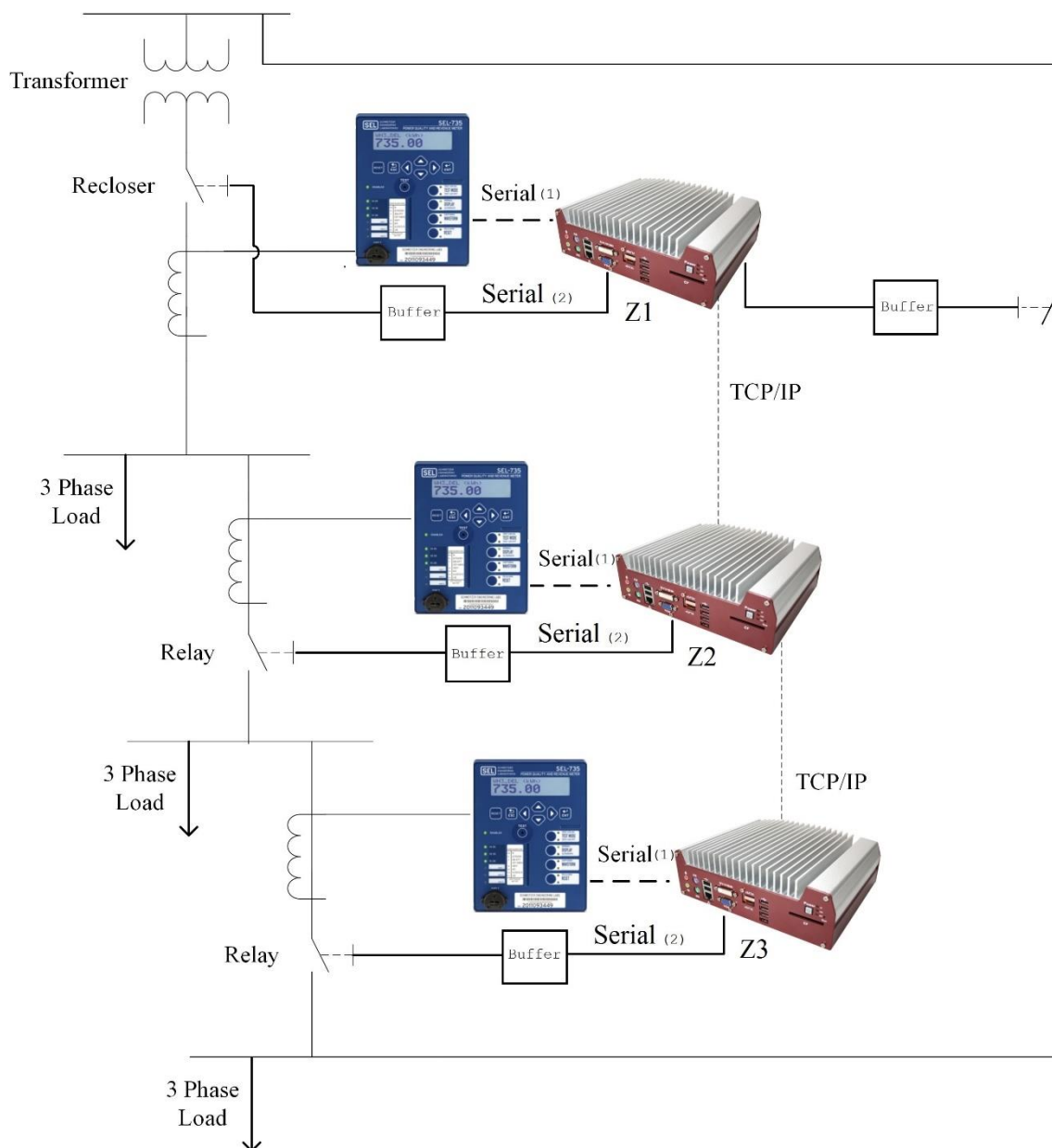
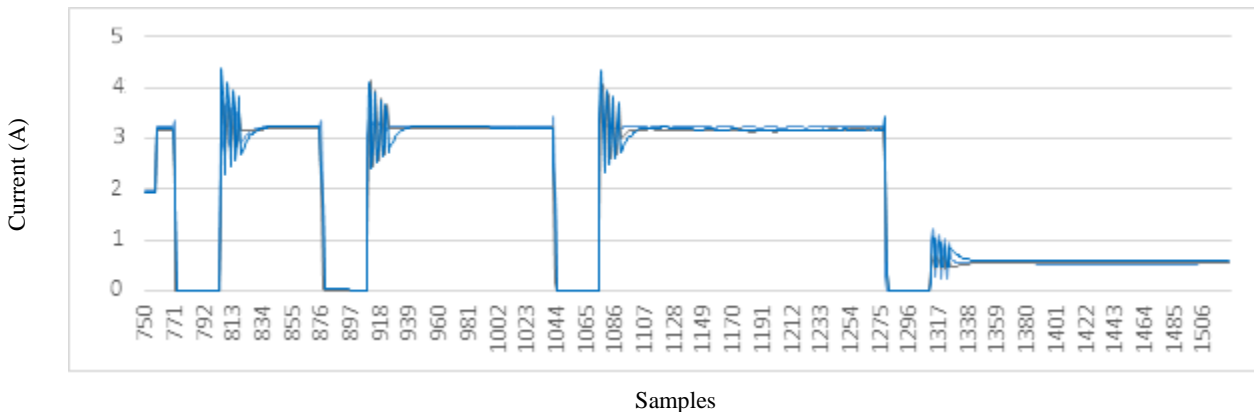
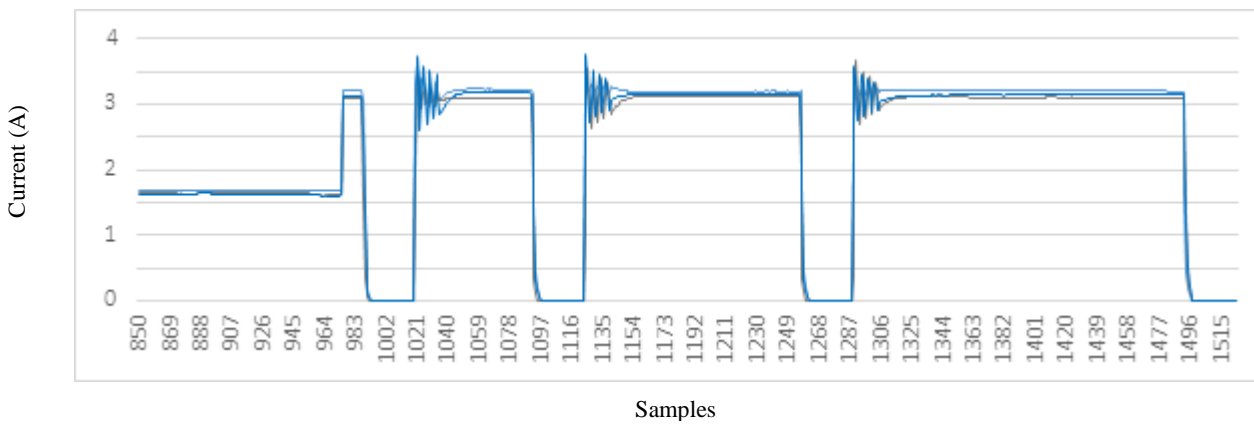


Figure 7.7: MAS implementation using buffers.

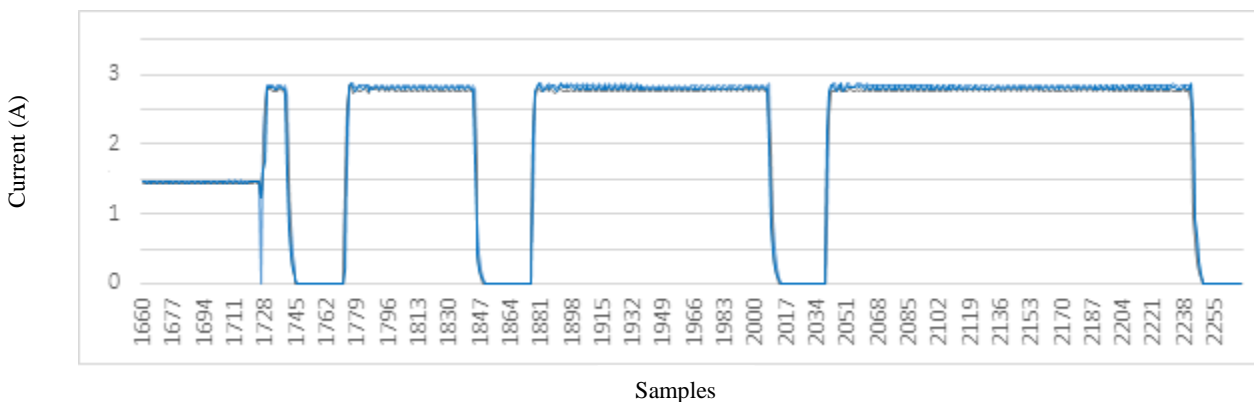
7.3 Results



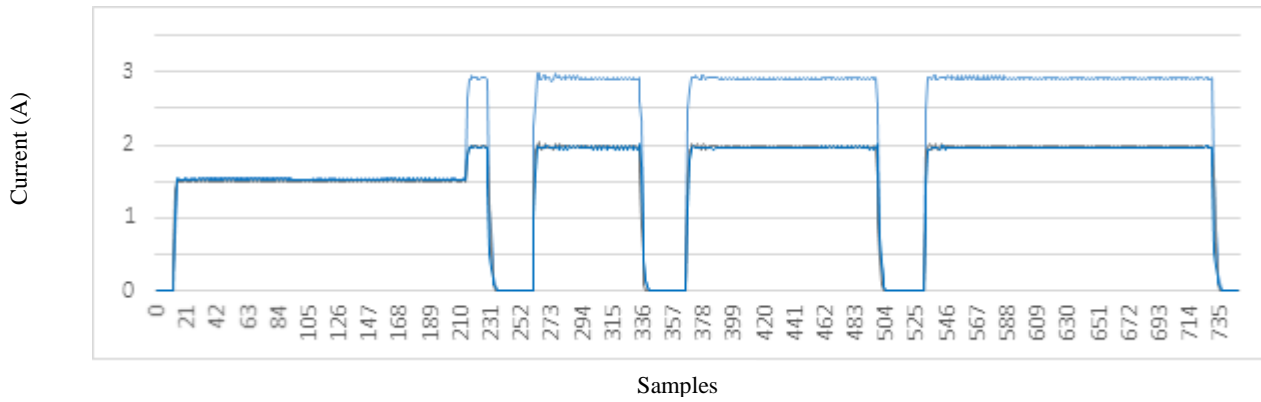
Zone 1 Fault three trials recloser process



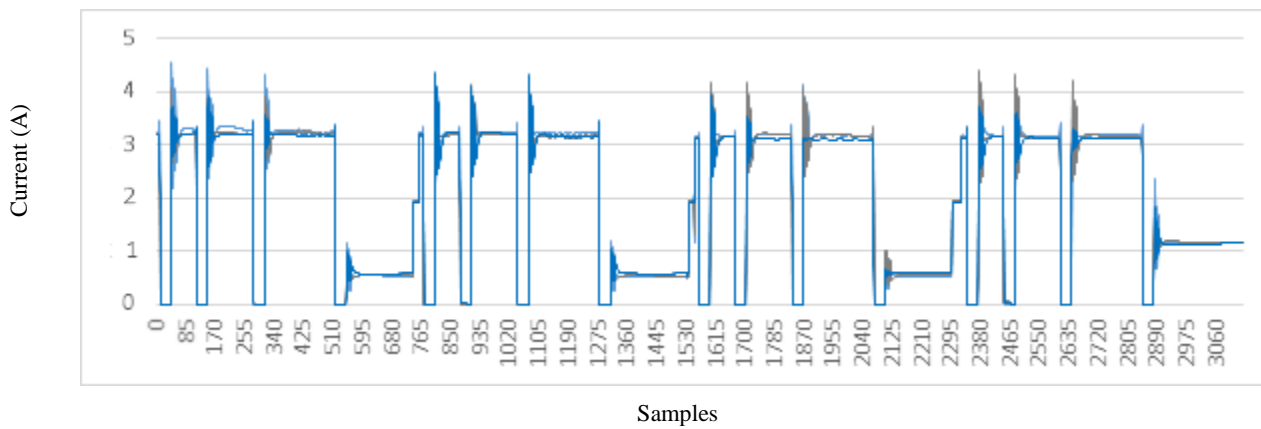
Zone 2 Fault three trials recloser process



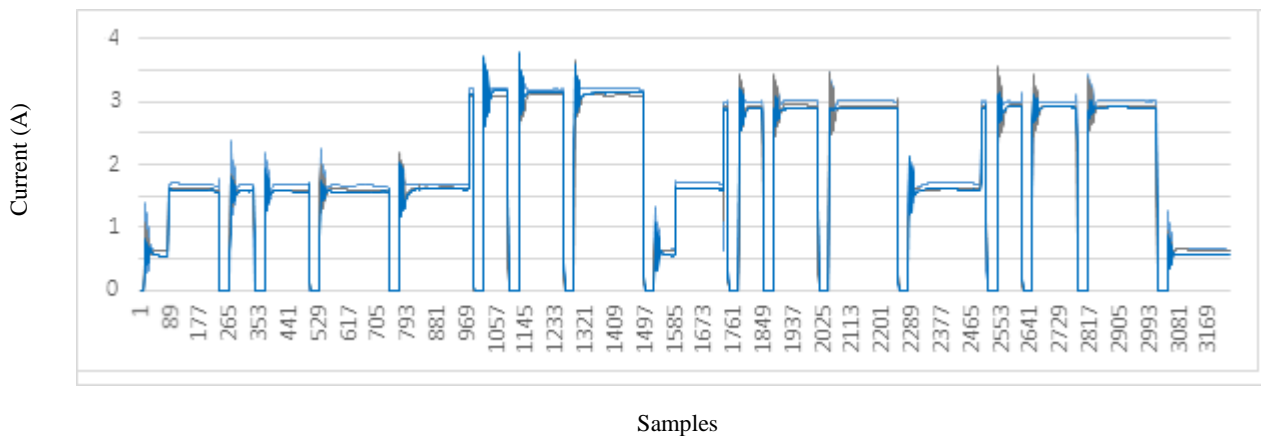
Zone 3 Fault three trials recloser process



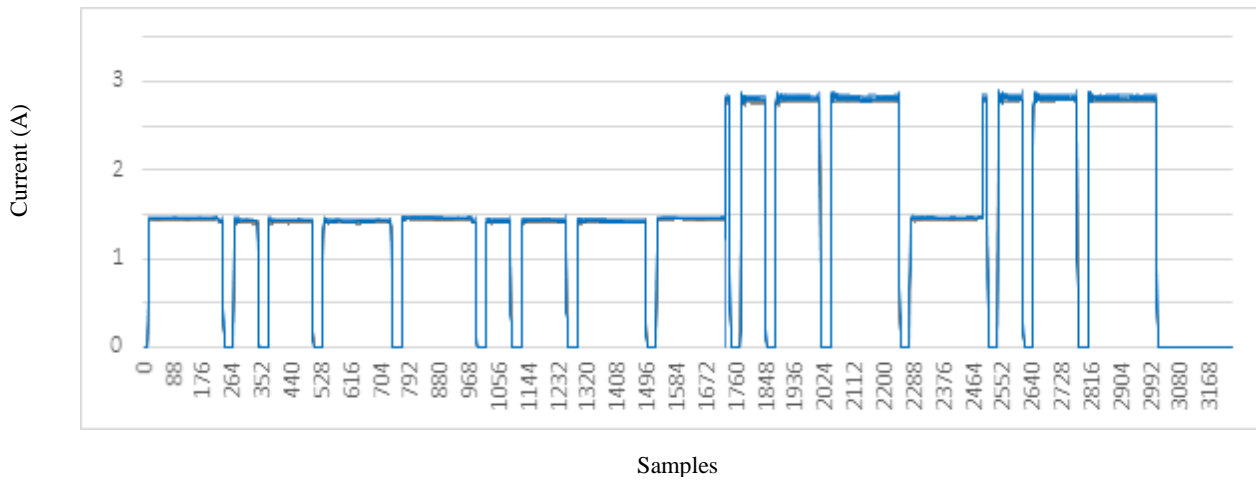
Zone 3 Line to Line Fault three trials recloser process



All zones Fault three trials recloser process – zone 1 full data



All zones Fault three trials recloser process – zone 2 full data



All zones Fault three trials recloser process – zone 3 full data

8. Conclusion

In this thesis, a multi-agent system design for power distribution restoration using artificial intelligence is presented. Neural Networks are used for the restoration process. Change of current algorithm is used for fault location and isolation as well. The following outline represents the main contributions of this work:

- Radial basis neural Network for restoration.
- Simulation studies for the super circuit in West Virginia.
- Hardware implementation of fault location and restoration in a laboratory platform.
- Decentralized configuration of MAS for restoration in the Lab experiment.
- Simulation of recloser in the power system laboratory using UART circuit.

Future Work

- 1- Determine the type of fault detected using Short Time Fourier Transform method.
- 2- Using of the Wavelet-based method to detect and locate the fault by analyzing the fault waveform.
- 3- Developing the buffer by adding Bluetooth module for each circuit in order to make the connection with the control unit wireless.
- 4- Apply Wi-Fi technology for designing the multi-agent system.

References

- [1] S. Premvuti and Y. Shin'ichi, "Consideration on cooperation of multiple autonomous mobile robots," in *IEEE International Workshop on Intelligent Robots and Systems*, Ibaraki, 1990.
- [2] L. Changhong, L. Minqiang and K. Jisong, "Cooperation Structure of Multi-agent and Algorithms," in *IEEE International Conference on Artificial Intelligence Systems*, Tianjin, 2002.
- [3] J. G. Gómez-Gualdrón, M. Vélez-Reyes and L. J. Collazo, "Self-Reconfigurable Electric Power Distribution," in *IEEE Electric Ship Technologies Symposium*, Arlington, 2007.
- [4] J. Ghorbani, S. Chouhan, M. Choudhry and A. Feliachi, "Hybrid Multi Agent approach for Power Distribution System Restoration," IEEE, Morgantown, 2013.
- [5] R. Belkacemi and A. Feliachi, "Multi-Agent Design for Power Distribution System Reconfiguration Based on the Artificial Immune System Algorithm," IEEE, Morgantown, 2010.
- [6] M. Biabani, M. A. Golkar and A. Sajadi, "Operation of a Multi-Agent System for Load Management in Smart Power Distribution System," IEEE, Warsaw, 2012.
- [7] J. Ghorbani, M. A. Choudhry and A. Feliachi, "Real-Time Multi Agent System Modeling for Fault Detection in Power Distribution Systems," IEEE, Morgantown, 2012.
- [8] J. Ghorbani, M. A. Choudhry and A. Feliachi, "Fault Location and Isolation Using Multi Agent Systems in Power Distribution Systems with Distributed Generation Sources," IEEE, Morgantown, 2013.
- [9] J. Ghorbani, Y. P. Fallah, M. A. Choudhry and A. Feliachi, "Investigation of Communication Media Requirements for Self Healing Power Distribution Systems," IEEE, Morgantown, 2013.
- [10] F. Saraiva and E. N. Asada, "Multi-agent Systems Applied to Topological Reconfiguration of Smart Power Distribution Systems," in *International Joint Conference on Neural Networks (IJCNN)*, Beijing, 2013.
- [11] T. Nagata and K. Okamoto, "A Decentralized Distribution Power System Restoration by using Multi-agent Approach," in *Proceedings of the International Electrical Engineering Congress*, Hiroshima, 2014.

-
- [12] B. Burmeister, A. Haddadi and G. Matylis, "Application of Multi-Agent Systems in Traffic and Transportation," IET, Berlin, 1997.
- [13] B. Chen, C. Wang and X. Gao, "Research on the Intelligent Agent of Distributed Fault Diagnose System," in *1st International Symposium on Systems and Control in Aerospace and Astronautics, 2006. ISSCAA 2006.*, Harbin, 2006.
- [14] G. Rohbogner, S. Fey, U. J. J. Hahnel, P. Benoit and B. Wille-Hausmann, "What the term Agent stands for in the Smart Grid Definition of Agents and Multi-Agent Systems from an Engineer's Perspective," in *Proceedings of the Federated Conference on Computer Science and Information Systems pp. 1301–1305*, Wroclaw, 2012.
- [15] M. P. Singh, "Agent communication languages: rethinking the principles," *Computer*, vol.31, no. Issue: 12, pp. 40 - 47, 1998.
- [16] H. Farooq and L. T. Jung, "Choices Available for Implementing Smart Grid Communication Network," in *International Conference on Computer and Information Sciences (ICCOINS)*, Kuala Lumpur, Malaysia, 2014.
- [17] Foundation for Intelligent Physical Agents, "FIPA ACL Message Structure Specification," 2002. [Online]. Available: <http://www.fipa.org/specs/fipa00061/>.
- [18] M. Ohrstrom, "Fast fault detection for power distribution systems," Royal Institute of Technology, Stockholm, 2003.
- [19] "Power system protection / Power system faults," IEC, 1995.
- [20] D. Summerville, *Embedded Systems Interfacing for Engineers using the Freescale HCS08 Microcontroller II: Digital and Analog Hardware Interfacing*, New York: Morgan & Claypool, 2009.
- [21] Texas Instrument, "MAX232x Dual EIA-232 Drivers/Receivers" SSSL047M datasheet, Feb. 1989 [Revised Nov. 2014]
- [22] Microchip, "PIC12F683 Data Sheet" DS41211D datasheet, 2007
- [23] Rasskin-Gutman, D. and Klosky, D. *Artificial Intelligence: Silicon Metaphors*. Cambridge, MA: MIT Press, 2009.
- [24] Karlik. B and Olgac. V. "Performance Analysis of Various Activation Functions in Generalized MLP Architectures of Neural Networks". *International Journal of Artificial Intelligence and Expert Systems (IJAE)*. Vol.1, Issue No.4, pp 111-113, 2004
- [25] Rojas. R. *Neural Networks : A Systematic Introduction*. Berlin, Germany: Springer, 1996
- [26] P.D. Wasserman, *Advanced Methods in Neural Computing*, New York: Van Nostrand Reinhold, 1993 on pp. 155-61, and pp.35-55

-
- [27] Chen, S., C.F.N. Cowan, and P. M. Grant, "Orthogonal Least Squares Learning Algorithm For Radial Basis Function Networks," *IEEE Transactions on Neural Networks*, vol. 2, no.2, March 1991, pp. 302-309
- [28] MATLAB Simulink. *Version 7.14.0 (R2012a)*. The MathWorks Inc., Natick, Massachusetts, 2012
- [29] *SEL 735 Power Quality and Revenue Meter, Instruction Manual*. Schweitzer Engineering Laboratories, Inc., Pullman, WA, 2011-2013
- [30] *Nuvo 1000 Series, User's Manual*. Neousys Technology Inc., 2011