QATAR UNIVERSITY

COLLEGE OF ENGINEERING

DESIGN OF PMU BASED REAL TIME FUZZY LOGIC SVC DAMPING CONTROLLER TO ENHANCE INTER- AREA OSCILLATION DAMPING

BY

SAMI SULIMAN ABUEIDA

A Thesis Submitted to the Faculty of

College of Engineering

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science in

Electrical Engineering

June 2016

©2016 Sami Suliman Abueida. All Rights Reserved.

COMMITTEE PAGE

The members of the Committee approve the thesis of **Sami Suliman Abueida** defended on17 May 2016.

_____

Dr. Uvais Ahmed Qidwai
Thesis/Dissertation Supervisor

_____

Dr. Khaled Ellithy
Thesis/Dissertation Co-Supervisor

_____

Dr. Khaled El-Metwally
Committee Member

_____

Dr. Nader Meskin
Committee Member

_____

Dr. Hasan Mehrjerdi
Committee Member

Approved:

_____

Dr. Khalifa Nasser Al-Khalifa, Dean, College of Engineering

Abstract

Inter-area oscillation has been identified as a significant problem in the utility systems due to the damages that it may cause as well as the limitation introduced to power transfer capability. A contemporary solution to this issue is by adding power system stabilizer (PSS) to the generator's automatic voltage regulator (AVR). Although nowadays most of the generators are equipped with conventional PSSs, their effects are only noticed on the damping of local oscillations and they do not contribute effectively on damping the inter-area oscillations. Adding auxiliary signals (stabilizing signals) to Flexible AC Transmission System (FACTS) device such as Static VAr Compensator (SVC)&Static Synchronous Compensator (STATCOM) would help in extending the power transfer capability and enhancing the voltage. The stabilizing signals can be derived from damping controller. In this thesis, a Phasor Measurement Unit (PMU) based real-time, Hardware in the Loop, fuzzy logic shunt FACTS controller is proposed to ensure a satisfactory damping of inter-area oscillations which will enhance system stability and increase power transfer capability.

The concerned power system has been modeled using Real-Time Digital Simulator (RTDS), where the designed Hardware-in-the-loop damping controller was tested for the sake of evaluating the effectiveness of the proposed controller in enhancing the damping of inter-area oscillations. Time-domain simulations results have shown that the designed Fuzzy damping controller enhance the damping of inter-area oscillations of interconnected power system. This study is aimed to analyze the potential applications of PMU in the interconnected power systems of GCC smart

power grid. These systems are expected to face a stability problem of the inter-area mode of oscillations due to the weak tie-lines that connect the systems.

Table of Contents

List of Tables

List of Figures

List of Abbreviation

AC	Alternative Current
ADC	Analog To Digital Converter
ANFIS	Adaptive Neuro-Fuzzy Inference System
AVR	Automatic Voltage Regulator
CT	Current Transformer
DAC	Digital To Analog Converter
DFT	Discrete Fourier Transform
FACTS	Flexible Ac Transmission System
FC	Fixed Capacitor
FCDHT	Fuzzy Controller Design Helper Tool
FLC	Fuzzy Logic Controller
GCC	Gulf Cooperation Council
GPS	Global Position System
HVDC	High Voltage Direct Current
LFO	Low Frequency Oscillation
MRFLC	Model Reference Fuzzy Learning Controller
MVAr	Mega Var
MW	Mega Watt
PCB	Printed Circuit Board
PI	Proportional Integral
PID	Proportional Integral Derivative

PMU	Phasor Measurement Unit
PSS	Power System Stabilizer
RTDS	Real Time Digital Simulator
STATCOM	Static Synchronous Compensator
SVC	Static Var Compensator
TCR	Thyristor Controlled Reactors
TCSC	Thyristor Controlled Series Capacitor
TSC	Thyristor Switch Capacitor
TSR	Thyristor Switch Reactor
TSSC	Thyristor Switched Series Capacitors
UPFC	Unified Power Flow Controllers
VT	Voltage Transformer
WAMS	Wide Area Monitoring System

Acknowledgements

First of all, I would like to express my sincere gratitude to almighty Allah and then to Dr. Khaled Ellithy and Dr. Uvais Qidwai for their constant support, expert guidance and constructive supervision.

I would also like to extend my appreciation to Eng. Abdulaziz Al Mahmoud from KAHRAMAA, and Eng. Ahmed Anad from Gdansk University of Technology for their help and support. I also thank all of those who helped me in accomplishing this work.

Finally, I must express my profound gratitude to my parents, my sisters, my wife, and my lovely son for their patience and continuous encouragement throughout the master study.

Chapter 1. Introduction

1.1. **Power System Stability**

The stability of a power system could be defined as the ability of a power system to return to the state of operational equilibrium after facing a disturbance so that the whole power system remains synchronized [1], [2]. In other words, the ability of the power system to generate opposing forces that are equal or more than the disturbance forces in order to keep the equilibrium state.

Nowadays, power systems are huge dynamic systems which are operated as interconnected systems and most of them are no longer islanded. Gulf Cooperation Council (GCC)power grid is an example of an interconnected power system as it consists of six different power systems as shown in Figure 1.1.

It is worth noting that, there are advantages and disadvantages of interconnected power systems. For instance, as advantages, it increases the system reliability as it has a better ability to support a sudden load change or generation lose. Also, an interconnected system would have higher inertia, which means that disturbances have a reduced effect. It also reduces the need of future power system expansion, as they share the spinning reserve, which lowers the spinning reserved requirement in each area / country.

On the other hand, the interconnected power systems have some disadvantages such as, the long tie-lines between adjacent power systems (ex. Qatar and Kingdom Saudi Arabia) are quite weak if it is compared with the connections within each member's power system, and this issue (weak tie-line) may lead to have low frequency oscillation (LFO).

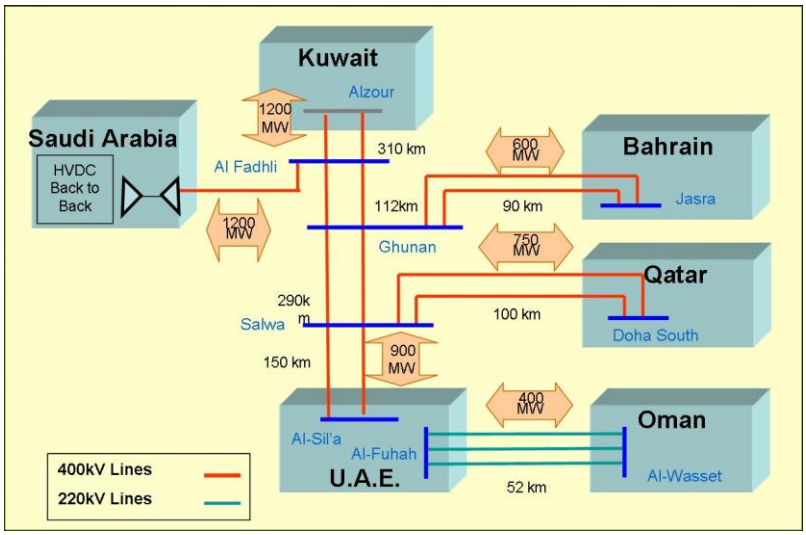


Figure 1.1GCC interconnection Distances & Capacities [3]

Power system stability can be classified into the following categories:

1. Voltage stability: which is the ability of the power system, after facing a disturbance, to keep steady voltage at all buses within the allowable limits [2].
2. Frequency stability: which is the ability of a power system to keep the steady frequency within the allowable limits following a cruel disturbance [2].
3. Rotor Angle Stability: which is the ability of synchronous generators in a power system, after facing a disturbance, to remain in synchronism [2].

The rotor angle stability has a major role in "power system stabilization via excitation control".

1.2. **Low Frequency Oscillations (LFO)**

One of the main causes of having Low Frequency Oscillation (LFO)is the high gain poorly tuned generation's excitation system [2]. This issue would add a negative damping torque to the generation units, which will cause the generators to have rotor angle oscillations. LFO has a range of frequencies from 0.1 Hz to 3.0 Hz which could be considered as a small-signal stability problem. Another cause of LFO is heavy power transfer across weak tie-lines.

As mentioned above, LFO belongs to small signal stability problem which is usually a result of small disturbances, such as load changing. These disturbances would cause gradual increases or decreases in the rotor angle, and this is due to either not having enough synchronizing torque, or insufficient damping torque [4]. The inadequate damping torque could cause:

1- Local mode &Interplant oscillations (0.7 Hz to 2Hz) which are associated with one generating unit or within the generation plants [4].

2- Inter-area oscillations (0.1Hzto0.8Hz) between two areas (Group of power plants) space [4].

1.3. **Inter-area Oscillations**

Normally, one of the main problems that could rise in heavily stressed widely spread system is the enhancement of the damping of LFO. That occurs because of inadequate damping torque in some generation units.

Traditional, Power System Stabilizer(PSS) has been used in damping the electromechanical oscillations and inter-area oscillations [5], through providing a supplementary signal to the generation excitation system.

Recently, Flexible AC Transmission System (FACTS) devices has turned into a common practice in utilities (ex. Static VAr Compensator (SVC) has been installed in KSA) for the sake of both limiting the modification in the current power systems and fully utilizing the existing transmission capacities. For instance, FACTS devices can be used instead of adding new long transmission lines (Over Head Lines) which, in some cases, could not be an economical solution, in addition to their other environmental & health impacts.

Unified Power Flow Controllers (UPFC), Thyristor Switched Series Capacitors (TSSC), and Static VAr Compensator (SVC) are examples of FACTS devices. In addition to their main advantage of having faster voltage and power flow control, adding an appropriate supplementary or auxiliary control signal to the FACTS devices can help in damping the inter-area oscillations.

1.4. **Flexible AC Transmission System Devices (FACTS)**

FACTS devices can be divided into series devices, shunt devices, and series shunt devices. The main feature of FACTS devices is the high speed action in controlling either active power, reactive power or both, which is achieved by using power electronics, therefore, they may used to improve the power grids transient stability[6]. Static VAr Compensator (SVC)and Static Synchronous Compensator (STATCOM) are shunt devices, and their primary application is to keep the bus bar voltage at the nominal value, through their voltage regulator

controller (main controller). This controller provides synchronizing torque, whereas their damping torque contribution is small [7]. Regarding the Series Devices, such as Thyristor Controlled Series Capacitor (TCSC) devices, they are mainly used to provide power flow control over the transmission lines. By tuning the main controller parameters, their damping contribution can be slightly improved [8].

Based on above discussion, having a secondary controller (damping controller),which is necessary for generating an additional damping signal, is required for the sake of increasing the FACTS Devices damping torque.

Conventionally, supplementary or auxiliary control signal is taken or generated from the conventional damping controller that is similar to PSS in term of design. That damping controller is, usually, designed in frequency domain which limits its own performance into certain operation value/s. Using Fuzzy logic controller instead may help in extending that range to be wider, as it does not needs an accurate mathematical model to establish a good control.

Usually, the damping controller input is taken from the generation units such as speed deviation, power deviation, etc. In this thesis, speed deviation will be taken as an input. Having such input from generation would limit our self to generation units' location, which may not be adequate input for many cases. Using Phasor Measurement Unit (PMU) instead would extend our capability to have the input from any bus or substation which would add a new feature which is the mobility.

1.5. **Static VAr Compensator**

The Static VAr Compensator (SVC) is considered as shunt FACTS device as it is connected in parallel with the grid substation. The main target of SVC is to regulate the voltage at its terminals by injecting or absorbing reactive power from the power system. Based on that SVC injects reactive power when the voltage is low, and absorbs reactive power when the voltage is high. Generally, SVC consists of Thyristor controlled capacitor & inductor banks, and controlling system. These banks are connected to the power grid through step-up transformer, as shown in Figure1.2 below.

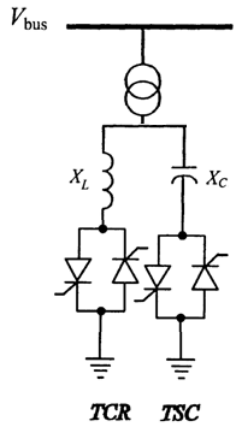


Figure1.2 The Basic Diagram Of The SVC

The injection or absorbing of reactive power is performed by switching capacitor banks & inductor banks on-off through Thyristor switches. The SVCs are designed in different ways/ combinations, such as Fixed Capacitors (FC) with Thyristor Controlled Reactors (TCR) and Thyristor Switched Capacitor (TSC) with TCR. TCR is controlled through courteous phase controller (firing angle) while for Thyristor Switch Reactor (TSR) is controlled through switching on-off the reactor banks (allocator is used for that) [9].

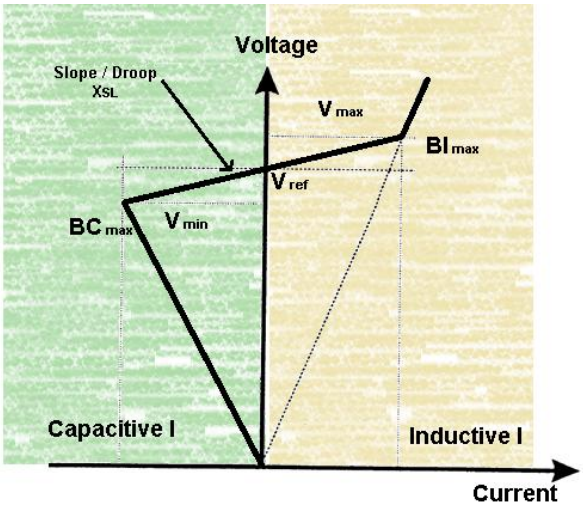


Figure1.3 V-I Characteristic Of SVC

The SVC has a main feature which is the voltage regulation characteristic that is accomplished by controlling the current susceptance and subsequently the reactive power in the grid. The regulated voltage is following a slope (droop)

characteristic. The value of the slope is decided based on the desired voltage regulation [10]. Figure1.3 shows dynamic voltage - current characteristics of the SVC.

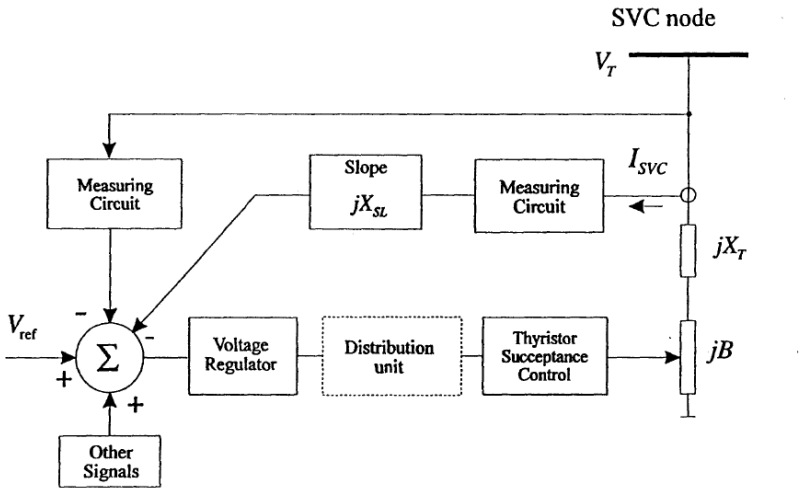


Figure1.4 SVC Control System [11]

The voltage control system shown in Figure1.4 consists mainly of the following modules:

- *Measuring Module*: The module, in Figure 1.5,measuresthevoltages and currents at the Bus where the SVC is connected as well as the current flow through the SVC [11].

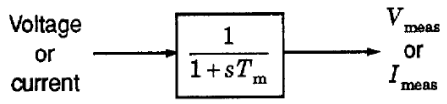


Figure1.5 Measurement module [11]

- *Voltage Regulator Module*: It uses the difference between the reference voltage and SVC terminal (error) as an input to the controller, PI regulator/ controller, in order to generate SVC susceptance reference value [11] as shown in Figure 1.6.

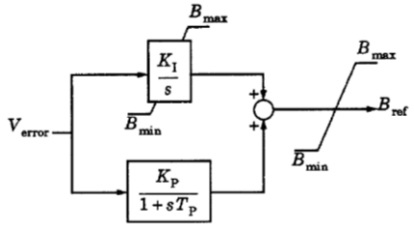


Figure1.6 Voltage regulator model [10]

- *Distribution Module or Allocator:* It converts the voltage regulator output (susceptance reference) into the number of reactor (TSR) / capacitor (TSC)

banks that should be switched on, in addition to compute the required firing angle for TCR [11].

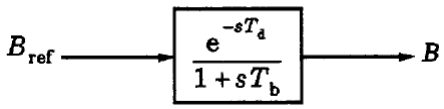


Figure1.7 Thyristor Susceptance Control [10]

- *Thyristor Susceptance Control Model*: This block, in Figure 1.7, represent the delay associated with the firing of the Thyristor, through T_d (gating transport delay) and T_b (Thyristor firing sequence) [10].

1.6. **Pharos Measurement Unit (PMU)**

Nowadays, PMUs are considered as one of the most important instrument in Wide Area Monitoring Systems (WAMS), as it is responsible for monitoring the power system. PMUs are mainly used to monitor precisely the voltage and the current phases of power systems based on GPS time-stamped reference, which is also called synchrophasors, because it helps in time aligning and synchronizing the different location measurements. Based on that power system conditions snapshot can be generated [12]. The power system snapshot plays a key role in

overcoming the voltage instability in the power systems which may lead to blackout situation. As the Instability is mainly caused by mismatching between the load dynamics, transmission lines, and generation that are needed to be monitored [13]. Figure 1.8 shows physical WAMS & PMU available in Qatar University, and Figure 1.9 shows the blocks diagram of the PMU.

Figure 1.8 WAMS & PMU in QU

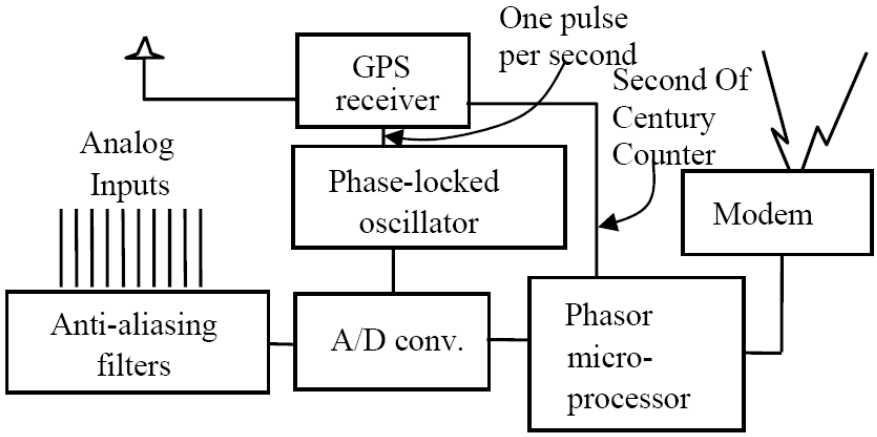


Figure1.9 PMU Blocks Diagram [14]

PMU starts with measuring the voltage and current waveforms through regular Current Transformer (CT) and Voltage Transformer (VT). After that, those waveforms signals are filtered and sampled using Analog Digital Converter (ADC). The fundamental frequency, voltage phasor and current phasor are then calculated using phasor microprocessor that use Discrete Fourier Transform (DFT). Finally, results are time-stamped using GPS and sent through communication media[14].

1.7. **Problem Statement**

In power systems, the existence of LFO opposes the power systems objectives of having maximum power transfer and stable system. Conventionally, the solution was the generator's damper. As a result of increasing the loads, the power systems reach close to the stability operation

limits, causing a weakness in the synchronizing torque between generation units in the power system. This issue was considered as a main source of system instability. The Automatic Voltage Regulators (AVR) has then been used to help in improving the steady-state stability power systems.

The introduction of interconnected power systems with massive power transfer on long transmission lines, cause low frequency oscillations that raise a need of additional supplementary controller added to the generators' AVRs. That supplementary controller is known as the power system stabilizers. Having conventional PSS may not be sufficient in damping the inter-area oscillation which leads to use the FACTS devices with damping controller. Conventional damping controllers, such as PI, PID& lead-lag, are designed to be operated around a certain operation point, with input signals from the generation. This issue could be overcome by using damping controller based on Fuzzy logic technique combined with the PMU which will be considered as our contribution in the thesis.

1.8. Objectives of the Work

The objectives of this thesis are

- To design a fuzzy logic based damping controller for SVC that will stabilize the system even without the presence of any conventional PSS at the testing system.

- Expand the Fuzzy Logic damping controller operation points by introducing online adaptive mechanism.

- Explore the possibility of using PMU as a source signals for the damping controller instead of the conventional signals source.

- Establish a Fuzzy Controller Design Helper Tool (FCDHT), for introducing a Fuzzy controller model in the Real Time Digital Simulator (RTDS)& generating microcontroller programming code as well as other important functions.

- Implement the designed Fuzzy damping controller through microcontroller and conducting Hardware In the Loop test through RTDS.

1.9. **Thesis Structure**

Chapter 1:

In chapter 1, the introduction of power system stability, low frequency oscillations, inter-area oscillation, FACTS Devices, SVC, Problem statement and the objective of the work are presented.

Chapter 2:

In chapter 2, the conducted literature review is presented.

Chapter 3:

In chapter 3, the system modeling where the models of generation unit, excitation system, power system stabilizer and SVC are covered, as well as a description of the testing system (two area and four machine).

Chapter 4:

In chapter 4, an introduction of fuzzy logic controller as well as the design of Fuzzy Logic based SVC damping controller is presented, in

addition to the introduced adaptation mechanism.

Chapter 5:

In chapter 5, the simulation results of MATLAB/SIMULINK as well as RTDS are presented in addition to the results of proposed adaption mechanism.

Chapter 6:

In chapter 6, Fuzzy Logic Controller platform preparation, Fuzzy Controller Design Helper Tool (FCDHT), and the implementation results are presented.

Chapter 7:

In chapter 7, the conclusion and future works are presented.

Chapter 2. Literature Review

Inter-area oscillation is a critical issue with interconnected power systems that may decrease the transfer capability of tie-lines between interconnected areas and deteriorate the power stability as well [15, 16]. As the electrical demand grows up, the power systems tend to operate near to their stability limits that made them more vulnerable to inter area oscillation [17]. To deal with this issue, PSS along with excitation system [18] is considered as a simple and low cost choice for enhancing the oscillation damping.

2.1. **FACTS Devices In Damping Inter-Area Oscillation**

A Variety of FACTS devices based inter-area damping controllers have been proposed and designed using Various methods such as in [19] where the Unified Power Flow Controller is based on Lyapunov-based adaptive neural network. While in [20], the Unified Power Flow Controller (UPFC)is based on adaptive input-output feedback linearization control, Static VAr compensators SVCs in [21] has been used along with WAMS. High Voltage DC (HVDC) link along with model predictive controller has been proposed in [22], whereas, in [23] active-power modulation of multi-terminal has been used. However, most of the methods mentioned above are also relying on linearizing system around an operating point, that may limiting their efficiency [24].

2.2. **Local and WAMS in Damping Controllers**

With local measured signals, the damping will be mostly effective with locally observable oscillations because its observation area is limited [19]. The conventional PSSs efficiency using local signal for damping the inter-area oscillation is somehow ambiguous [25,26]. To tackle this issue, in [27] the multi-band PSS has been used, while in [28]non linear PSS has been applied, and the coordinated PSS used in [29]. On the other hand, WAMS with PMUs is capable to monitor and measure the power system data such as voltage, current, angle, and frequency which give the ability to monitor wider area [30]. This method enables the remote signals to be fed to the controller to enhance the system dynamic performance more than the local measured signals [31].

Table 2.1Summary of 2.1 and 2.2

Ref.	***FACT Device***	***Controller***	***Input Signal***
5	*Unified Power Flow Controller (UPFC)*	*Lyapunov-based adaptive neural network*	*Local*
6	*UPFC*	*adaptive input-output feedback linearization*	*WAMS*
7	*Static VAr compensators (SVC)*	*lead-lag*	*WAMS*
8	*HVDC*	*model predictive*	*Local*
9	*HVDC*	Active-power modulation of multi-terminals	*WAMS*
11	*SVC*	*Conventional PSS*	*WAMS& LOCAL*
13	*N/A*	multi-band PSS	*WAMS*
14	*N/A*	non linear PSS	*LOCAL*
15	*N/A*	Coordinated PSS	*LOCAL*
16	*Static Synchronous Series Compensator (SSSC)*	FUZZY CONTROLLER	*WAMS*
17	*N/A*	PSS	*WAMS*

2.3. **WAMS & PMUs Based Damping Controllers**

Many classical and advanced controllers' design techniques utilizing WAMS have been proposed. For instance, as classical methods, modal analysis has been implemented in Norwegian transmission network in[32],and also modal analysis has been used along with time delay issue in [33],and the same thing has been applied in [34] where HVDC damping controller is designed considering 200 ms as a delay in China-southern power grid. Regarding advanced controller designing techniques, in [35] the authors has investigated multi-objective robust HVDC supplementary controller. Whereas, in [36] H_∞ controller has been used. Classification and regression-tree based adaptive damping control is proposed in [37] and Multivariable self-tuning feedback linearization controller is presented in [38].Probabilistic collocation method is applied in [39],while non-linear excitation controller using inverse filtering is proposed in [40]. Networked Predictive Control Approach considering the delay in communication in [41].

Table 2.2 Summary of 2.3

Ref .	***Technique***	***FACT / Exciter***	***Remark***
18	*Classical **(Modal Analysis)***	*SVC*	*In Norwegian transmission network*
19	*Classical **(Modal Analysis)***	*FACTS devices*	*Time delay considered*
20	*Classical **(Modal Analysis)***	*HVDC*	*200 ms as a delay*
21	*Advanced (**Multi-Objective**)*	*HVDC*	
22	*Advanced (**H_∞**)*	*Multiple (FACTS)*	*Time delay considered*
23	*Advanced (**Classification and regression-tree based**)*	*TCSC, SVC &ESD*	
25	*Advanced (**Probabilistic Collocation Method**)*	*VSC-HVDC*	*Time delay considered*
26	*Advanced (**inverse filtering technique**)*	*Exciter*	
27	*Advanced (**Networked Predictive Control Approach**)*	*Exciter*	

2.4. **Robust Controller**

The main objectives of designing methods are to realize a robust controllers covering wide range of operating conditions, as well as achieving some of the controller specifications.

In [37, 38,39] the controllers design is adaptive for various operating points. In [37] the robust controllers are designed off-line in line with the real-time operating point. While in [38] and [39]the controller parameters are automatically updated using real time model prediction and estimation.

2.5. **Classical Fuzzy Damping Controller**

In [42] PID controller has been used as a main damping controller while the fuzzy controller has been used for Thyristor susceptance control. In [43] Fuzzy controller has been combined with the PI controller. The fuzzy controller has been placed in series with the integral part of the PI controller for substances control. In [44] the proposed SVC controller consists of two parts which are the traditional PI–controller as well as the supervisory fuzzy logic controller. The output of the fuzzy controller is the supplementary substances signal that will be combined with PI output.

2.6. **Adaptive Fuzzy Damping Controller**

In[45] adaptive fuzzy controller has been used that consists of two linear damping controllers for the two extreme operating conditions in addition to a fuzzy logic adaptation mechanism . Both linear controllers' outputs are combined through weighted summation, and those weights are generated by the fuzzy

controller.. In [46] the Strategy of Oscillation Energy Descent method has been used to make the adaptive fuzzy controller. The proposed controller consists of two fuzzy controllers. One of them is the main damping controller while the other one is used for adjusting main controller's gain factors. The output of the controller is the additional amount of substances required to be either generated or absorbed by the SVC. In [47] adaptive neuro-fuzzy inference system has been used for adjusting the PI gains, main damping controller, according to the system loading conditions. In [48] Hybrid Damping controller has been introduced. It consists of PD Fuzzy controller as well as PI controller tuned by Genetic Algorithm.

Table 2.3 Summary of 2.6

Ref.	***Type of controller***	***Input***	***Input type***	***Fuzzy controller enrollment***
28	Classical Fuzzy +PID	Delta Speed	Local	PID has been used as damping controller while Fuzzy controller is used to identify the amount of substances needed to be generated / absorbed by SVC.
29	Classical Fuzzy + PI controller	Voltage	Local	Fuzzy Controller has been placed in series with the integration part of the PI controller.
30	Classical Fuzzy + PI controller	Voltage	Local	Fuzzy controller has been places in parallel with main SVC PI controller, both controllers' output are combined for having better voltage compensation.
31	Adaptive Fuzzy + linear damping controller	Power	Local	Fuzzy logic controller has been used to produce the summation weighs of both linear damping controllers.
32	Two fuzzy controller	Power	local	One of them has been used as a main controller and the other one is used as gain modifier for the first fuzzy logic controller.
33	Adaptive Neuro-Fuzzy Inference System + PI	Power & Speed deviation	Local	Adaptive Neuro Fuzzy is used to modify the gains of the main SVC PI controller.
34	PD Fuzzy controller + PI	equivalent machine angle difference	local	PD fuzzy controller is combined with PI controller in order to generate the supplementary damping signal, The gains of both PI & PD fuzzy controller are tuned using Genetic Algorithm.

2.7. **Our Contribution**

In this thesis, WAMS & PMUs with SVC's robust fuzzy damping controller along with adaptation mechanism is proposed. The Fuzzy controller will be used as it can cover wider range of operating points than the conventional one. The operation range of fuzzy controller will be extended using an adaptation mechanism. WAMS & PMUs will be used as source for the controller in order to widening monitoring range as well as extending the operation points range. The system will be implemented and tested through the RTDS systems as Hardware in the Loop.

Chapter 3. Power System Modeling

In this chapter, the models of the generators, AVR and PSS used in the testing system (two area, four machine system) are described in fairly details.

3.1. **Generator Model**

For stability studies, several models can be used in modeling synchronous generators, some of them include damper windings and some are not. In the sixth order model, the generator has four windings, two are in the q-axis and the rest are in the d-axis. In this model, the network and stator transients are neglected which will lead to conservative results, and this issue is preferred in stability studies [49].This model of synchronous machine is usually described by six equations as follows [50].

$$\frac{d}{dt}E_d^{'} = \frac{1}{T_{q0}^{'}}\left[-E_d^{'} + (X_q - X_q^{'})\left\{I_q - \frac{X_q^{'} - X_q^{''}}{\left(X_q^{'} - X_{lk,s}\right)^2}(\psi_{2q} + \left(X_q^{'} - X_{lk,s}\right)I_q + E_d^{'}\right\}\right] \quad 3.1$$

$$\frac{d}{dt}E_q^{'} = \frac{1}{T_{d0}^{'}}\left[-E_q^{'} + (X_d - X_d^{'})\left\{I_d - \frac{X_d^{'} - X_d^{''}}{\left(X_d^{'} - X_{lk,s}\right)^2}(\psi_{1d} + \left(X_d^{'} - X_l\right)I_d + E_q^{'}\right\}\right] \quad 3.2$$

$$\frac{d}{dt}\psi_{1d} = \frac{1}{T_{d0}^{''}}\left[-\psi_{1d} + E_q^{'} - (X_d^{'} - X_l)I_d\right] \quad 3.3$$

$$\frac{d}{dt}\psi_{2q} = \frac{1}{T_{q0}^{''}}\left[-\psi_{2q} + E_d^{'} - (X_q^{'} - X_{lk,s})I_q\right] \quad 3.4$$

$$\frac{d}{dt}\omega_r = \frac{1}{2H}[P_m - P_e - D\ \Delta\ \omega_r] \quad 3.5$$

$$\frac{d}{dt}\delta = \left(\omega_r - \omega_{syn}\right) = \Delta\ \omega_r \quad 3.6$$

Where:

X_d	:	Synchronous Reactances in d-axes
X_q	:	Synchronous Reactances q-axes
$X_d^{'}$	:	Transient Reactances of d- axes
$X_q^{'}$	:	Transient Reactances of q-axes
$X_d^{''}$	:	Sub-Transient Reactances of d-axes
$X_q^{''}$	:	Sub-Transient Reactances of q-axes
X_l	:	Leakage Inductance
$T_{d0}^{'}$	:	Transient Time Constant of d-axes
$T_{q0}^{'}$	:	Transient Time Constant of q-axes
$T_{d0}^{''}$	:	Sub-Transient Time Constant of d -axes
$T_{q0}^{''}$	:	Sub-Transient Time Constant of q-axes
H	:	Inertia Constant, stored energy at rated speed,
I_d	:	Armature Current of d-axis
I_q	:	Armature Current of q-axis
ψ_{1d}	:	Flux Linkage Damper Winding of d-axis
ψ_{2q}	:	Flux Linkage Damper Winding of q-axis
P_e	:	Electrical Power
P_m	:	Mechanical Torque
ω_r	:	Generator Rotor Speed
ω_{syn}	:	Rated Generator Rotor Speed (2πf)
$E_d^{'}$	:	Transient Voltage in d-axes
$E_q^{'}$		Transient Voltage in q-axes
δ	:	Rotor Angle

3.2. **Excitation System Models**

The excitation system's main objective is to regulate the generation terminal voltage through controlling the field current. As the time constant of field current is usually high, in term of seconds, fast control is required. Therefore,

exciter should have a very high voltage output. The used exciter, as shown in Figure3.1, is Fast Exciter (Static Exciter) [51].

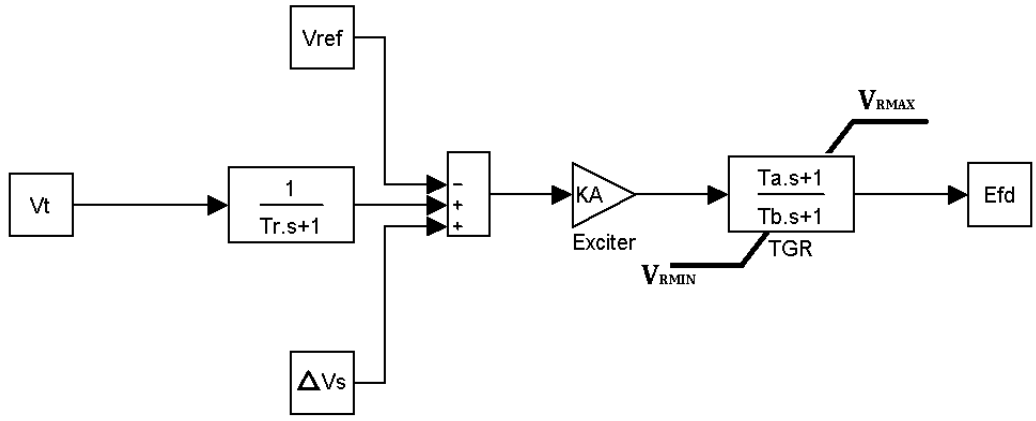


Figure3.1 Exciter Model Static Exciter [51]

Where:

K_A : Exciter Gain
T_r : Transducer Time constant
T_a& T_b : Transient Gain Reduction lead-lag time constant

3.3. **Power System Stabilizer**

The main aim of the PSS is to enhance the angular stability limits and to damp out the oscillation in synchronous machines' rotors through providing a supplemental signal to the generator's excitation system. The PSS supplementary control signal (Vs) is very useful during huge power transfers, as it increases the tie-line thermal limit by damping the oscillation [52, 53]. However, the PSS negative damping effects on the machine’s rotor may lead to instability, as the conventional PSS is usually tuned around certain operating point where their

efficiency is within small range around it [53]. During large disturbances, a Power System stabilizer may cause the generators to lose synchronism as it is trying to control the excitation field [53].Figure3.2 shows the PSS Model used.

It is worth noting that washout block is used to eliminate the DC component from the signal &Lead –Lag blocks are used for phase compensation.

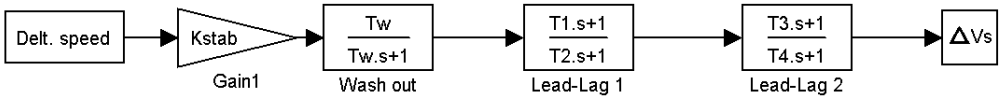


Figure3.2 PSS Model

where:

K_{stab} : Stabilizer Gain
T_w : Washout Time Constant
T_1 & T_2 : Lead Lag 1 Time Constants
T_3 & T_4 : Lead Lag Time Constants

3.4. **Static VAr Compensators**

The main aim of SVC is to regulate the voltage at its terminals by injecting or absorbing reactive power. Figure3.3 below shows the basic block diagram of SVC, which mainly consists of a PI controller that represents the voltage regulator part. The voltage regulator input is the difference between the reference voltage and the bus voltage(error) in addition to the damping controller signal, while the

controller output is the susceptance value required from the SVC to generate or absorb.

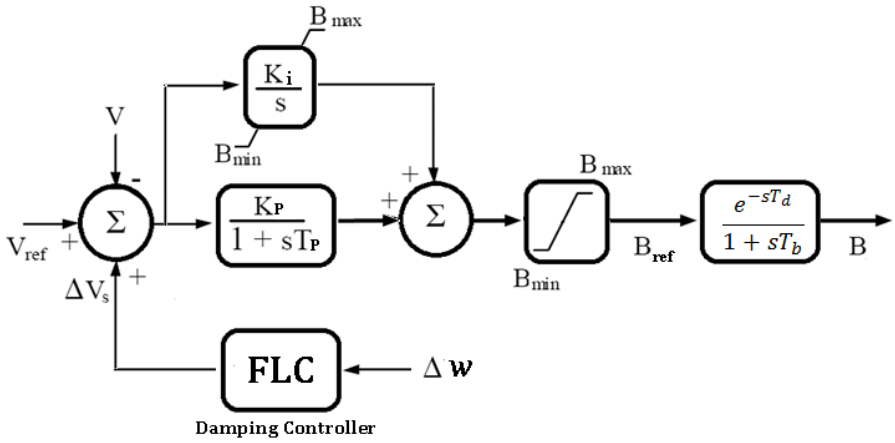


Figure3.3 SVC Basic Block Diagram of SVC Model [54]

Where:

$\frac{K_p}{1+sT_p}$	The proportional part of the voltage regulator
$\frac{K_i}{S}$	The Integrator part of the voltage regulator
$\frac{e^{-sT_d}}{1+sT_b}$	Thyristor succeptance control

The typical values for the controller are shown in the Table3.1below .

Table3.1 Typical Values of SVC controller [51]

Module	Parameter	Definition	Typical values
Measuring	T_m	Measuring Time constant	0.001s - 0.005s
Thyristor Control	T_d	Gating transport delay	0.001s
	T_b	Thyristor firing sequence delay	0.003s - 0.006s
Voltage regulator	K_i,K_p	Integrator& Proportional Gain	Vary based on how fast & well damping
Slope	X_{SL}	Slope, Droop	0.01-0.05 p.u.

3.5. **Testing System**

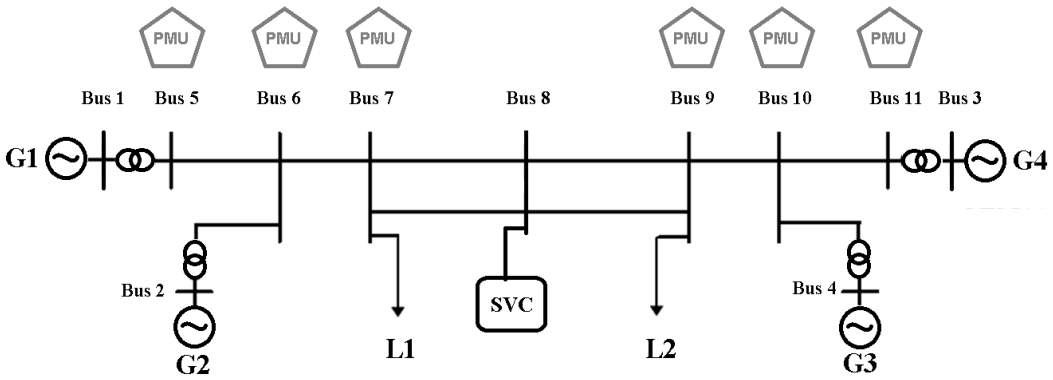


Figure3.4 Testing System[51]

Figure3.4 shows the single line diagram of the two-area, four-machine power system (as well as the PMUs location) that is used for testing the inter-area oscillation problem[49]. Based on that, this system has become a reference for inter-area oscillation problem [49]. In this system, there are four generators, GEN1and GEN2 are in area one, while GEN3 and GEN4 are in area two. All of

the generators are associated step up 20kV/230kV transformers. Two loads allocated at buses 7 and 9 with 967MW and 1767MW respectively. SVC is installed in the midpoint of the tie-line at Bus 8 with capacities of 100MVAr& 200MVAr. The testing system model parameters are shown in Tables3.2to 3.8 [49].

Table3.2 Transformer Parameters[49]

Transformers Parameters (Based on 900MVA)			
From Bus	**To Bus**	**R (P.U.)**	**X (P.U.)**
1	5	0.0	0.15
2	6	0.0	0.15
3	11	0.0	0.15
4	10	0.0	0.15

Table3.3 Branches Parameters [49]

Line Parameters (Based on 100MVA)						
From Bus	**ID**	**To Bus**	**Length(km)**	**R(PU)**	**X(PU)**	**B(PU)**
5	1	6	25	0.0025	0.025	0.04375
6	1	7	10	0.001	0.01	0.0175
7	1	8	110	0.011	0.11	0.1925
7	2	8	110	0.011	0.11	0.1925
8	1	9	110	0.011	0.11	0.1925
8	2	9	110	0.011	0.11	0.1925
9	1	10	10	0.001	0.01	0.0175
10	1	11	25	0.0025	0.025	0.04375

Table3.4 Load Flow Data [49]

Generator / Load	**Voltage (PU)**	**Angle**	**P (MW)**	**Q (MVAr)**
G1	1.03	18.56	700	185
G2	1.01	8.8	700	235
G3	1.03	-8.5	719	176
G4	1.01	-18.69	700	202

Table3.5Loads [49]

Bus	P_L	Q_L	Q_C
7	967	100	200
9	1767	100	350

Table3.6 Generator Parameters [49]

Generators Parameter (Based on 900MVA, 20kV)		
Parameter	**Generators 1, 2**	**Generators 3, 4**
x_d	1.8	1.8
x'_d	0.3	0.3
x''_d	0.25	0.25
x_q	1.7	1.7
x'_q	0.55	0.55
x''_q	0.25	0.25
T'_{do}	8.0	8.0
T''_{do}	0.03	0.03
T'_{qo}	0.4	0.4
T''_{qo}	0.05	0.05
A_{sat}	0.015	0.015
B_{sat}	9.6	9.6
H	6.5	6.175
R_a	0.0025	0.0025
Xl	0.2	0.2
Ψ_{T1}	0.9	0.9
K_D	0	0

Table3.7 Exciter & PSS Parameters [49]

Parameter	**Value**
K_A	200
T_R	0.01
T_a	0
T_b	0
K_{Stab}	20
T_W	10
T_1	0.05
T_2	0.02
T_3	3
T_4	5.4

Table3.8SVC Parameters

Parameter	**Value**	
Reactive power limit	$\pm$ 100MVAr	$\pm$ 200MVAr
C	1029.9 uf	2059.8 uf
L	5.0399 mH	2.51995 mH
T_d	0.001	
X_{SL}	0.03	
K_p	0	
K_i	300	

Chapter 4. Design of Fuzzy Logic Damping Controller

4.1. **Introduction**

Lofty A. Zadeh, in 1965, published "Fuzzy Sets" where the mathematical theory and logic have been represented [56]. His main idea was to let the digital computers to represent unclear ideas as what the human can do, in order to allow the computers to threat the data similar to human reasoning.

FLC effectiveness appears when the plant, needed to be controlled, is difficult to be modeled, whereas the experienced operator is available. FLC main idea is to emulate the operator qualitative way / rules of controlling the system.

FLC has several advantages over the conventional controllers such as:

- The simplicity, as it is based on the linguistic rules (human logic) based on IF-THEN structure.
- Does not need an accurate system model of the plant.
- Can deal with nonlinear systems.
- Does not need clear data and can work with approximate data.

4.2. **Fuzzy Logic Controller (FLC)**

Figure 4.1 shows the fuzzy logic control system's block diagram

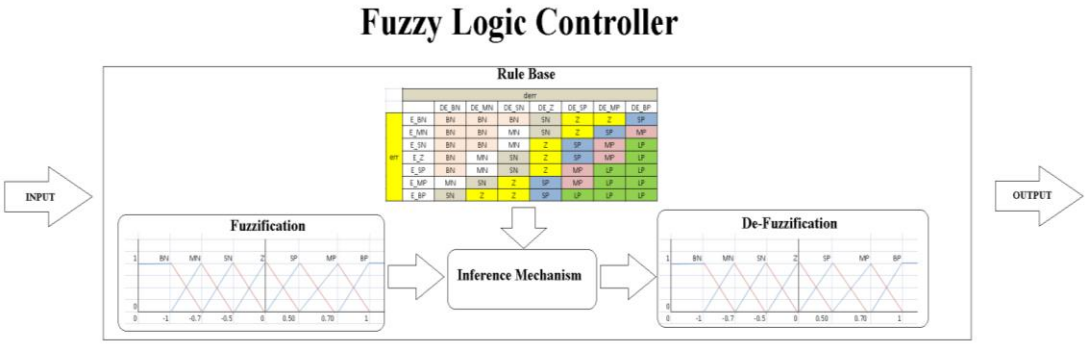


Figure4.1 FLC Stages

The FLC consists of the following four main stages:

1. Fuzzification stage, in which the regular / crisp inputs are converted into linguistic Variables in order to help the inference engine in executing the rule base.
2. Knowledge Base or Rule Base (set of multiple If-Then rules), that contains the linguistic knowledge of the expert of describing how to accomplish a good control of the plant.
3. Inference engine, which emulates the expert decision of how to apply the knowledge base in controlling the plant.

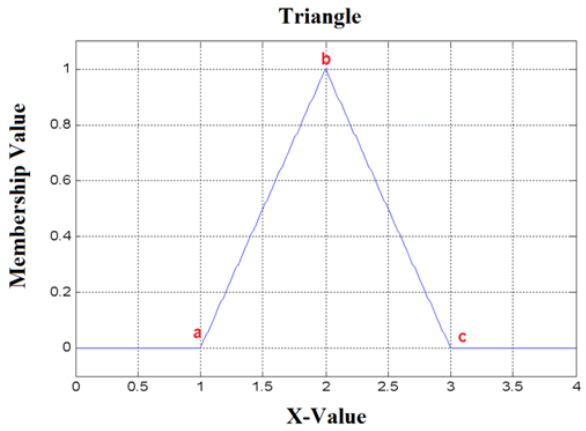
4. De-Fuzzification stage, where the inference engine's linguistic outputs is converted into crisp outputs.

4.2.1 Fuzzification Stage:

In this stage, the controller's inputs will be transformed from its original form, crisp, into linguistic form that is closer to the human way of thinking using membership functions.

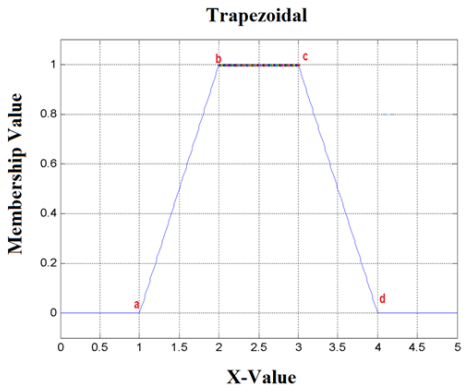
The membership functions are functions that linked the crisp Variables, in a certain region, with the percentage of belonging (0 %--> 100%) to a membership. These membership functions will have names used in our daily life (linguistic values/ labels) such as big, medium, or small.

There are Various types of membership functions some of them are described below in Figures 4.2 to 4.6.[57]



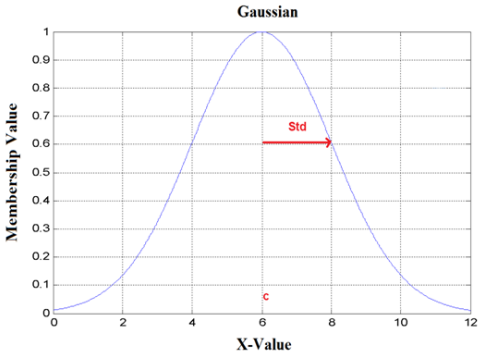
$$Triangle\ (x,a,b,c) = \begin{cases} 0, & x \leq a \\ \dfrac{x-a}{b-a}, & a \leq x \leq b \\ \dfrac{c-x}{c-b}, & b \leq x \leq c \\ 0, & c \leq x \end{cases}$$

Figure4.2Triangle Membership Function



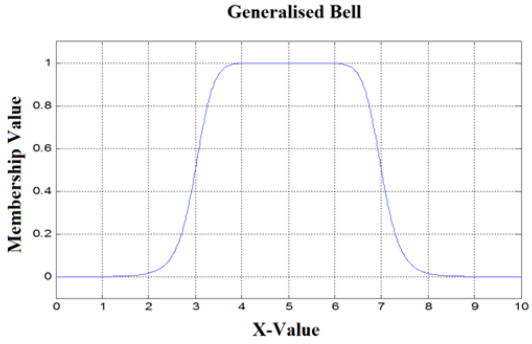
$$Trapezoidal\ (x, a, b, c, d) = \begin{cases} 0, & x \le a \\ \frac{x-a}{b-a}, & a \le x \le b \\ 1, & b \le x \le c \\ \frac{d-x}{d-c}, & c \le x \le d \\ 0, & d \le x \end{cases}$$

Figure4.3 Trapezoidal Membership Function



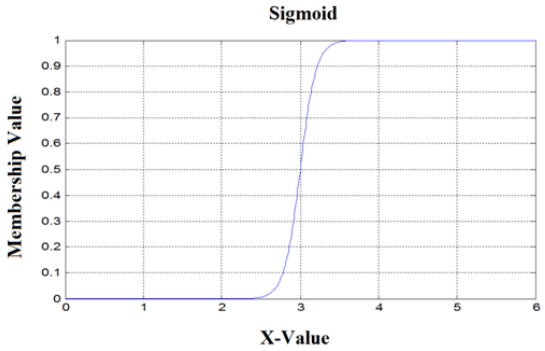
$$Gaussian(x, c, std) = e^{\frac{-1}{2}(\frac{x-c}{std})^2}$$

Figure 4.4 Gaussian Membership Function



$$Bill(x, a, b, c) = \frac{1}{1 + \left|\frac{x-c}{a}\right|^{2b}}$$

Figure 4.5 Generalized Bill Membership Function



$$Sig(x, a, c) = \frac{1}{1 + e^{[-a(x-c)]}}$$

Figure4.6 Sigmoid Membership Function

4.2.2 The Rule / Knowledge Base

As mentioned above the rule base contains the linguistic rules description of the expert knowledge of how to control the plant, in other words it is a Table that shows the relationship between the inputs and the outputs, in form of IF-THEN relationship. For instance,

IF **Error** is Big Positive AND **Change in Error** is Big Negative THEN

Output is Small Negative.

4.2.3 Inference Engine / Mechanism

It is the engine that determines the conclusions after applying the rule base or knowledge base. It is conducted by evaluating each rule independently, and then combines all of the rules output to determine controller linguistic output. There are two main inference systems Mamdani, and Sugeno.

A. Mamdani Inference Systems

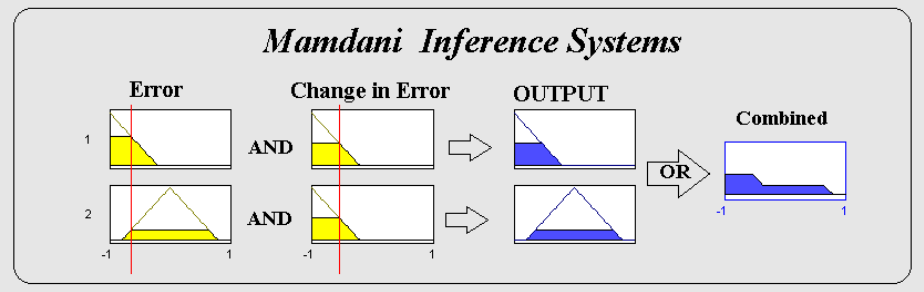


Figure4.7 Mamdani Inference System

In Figure4.7, there are two FLC rules in the form of

R_1: if *Error* is $X_1$and *Change in Error* is Y_1 then *Output* is C_1
R_2: if *Error* is X_2 *and Change in Error is* Y_2 then *Output* is C_2

Basically, it starts the AND operation which is the min of (Error & Change in Error) for each rule, and then the overall system output is calculated by combining those individual rule outputs, by union operation / OR operation.

B. Sugeno Inference Systems

The Sugeno Inference system or T-S inference system introduced in 1985[58]. It has the same form of Mamdani inference system but with output as a function of the inputs. In others words, it has no output membership function, and the output is computed evaluating the function.

The following is the form:

R1: if Error is X_1 and Change in Error is Y_1 then Output is z= f(Error ,Change in Error)

Where z= f(Error ,Change in Error)is conclusion crisp function.

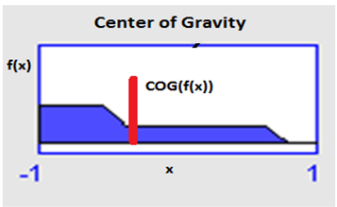
In case, z = f (Error, Change in Error) is a polynomial of first order, and then the Sugeno fuzzy model is named as first order. Whereas, if z is a constant, then the Sugeno fuzzy model is called a zero order.

4.2.4 De-Fuzzification

In de-fuzzification, the linguistic output of the inference system is converted in to crisp output. There are many types for De-fuzzification methods such as Center of Gravity (COG), Mean of Maximum (MOM), Min-Max, Weighted Average Formula (Min – Max WAF), Center of Sum (COS), …etc [59], and we will limit our self to three of the most common methods which are Center of Gravity (COG),Mean of Maximum (MOM),and Weighted Average(WAF) [59,60]

A. Center of Gravity (COG)

In the center of gravity method, the center location of the total area of the inference engine's output is computed by using the following formulas [59]:



For continuous signal version:

$$COG(f(x)) = \frac{\int_{-\infty}^{\infty} f(x).x \ dx}{\int_{-\infty}^{\infty} f(x)dx}$$

Discrete Version:

$$COG(f(x)) = \frac{\sum_{i=0}^{n} f(x_i).x_i}{\sum_{i=0}^{n} f(x_i)}$$

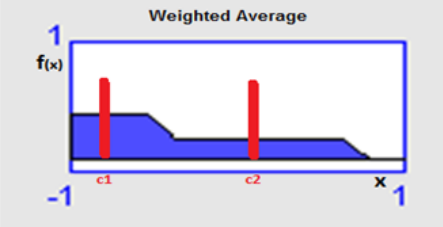
Figure 4.8 Center of Gravity

Where:

f(x) is the degree of membership function

B. Weighted Average

This is another method of de-Fuzzification which can be implemented by using the following equation [59].



$$WA = \frac{\sum_{i=1}^{n} f(c_i).c_i}{\sum_{i=1}^{n} f(c_i)}$$

Figure 4.9 Weighted Average

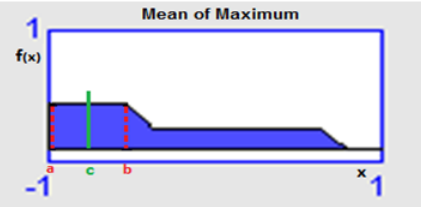
Where:

Ci is the center of output membership function

f(Ci) is the degree of membership function

C. Mean of Maximum

MOM is the average of the maximizing membership as shown in Figure 4.10. The average of the maximum range can be calculated as the following [59].



$$MOM = \frac{\int_a^b x\,dx}{\int_a^b dx} = \frac{a+b}{2}$$

Figure 4.10 Mean Of Maximum

4.3. **Design of Fuzzy Logic Damping Controller for SVC**

Figure4.11 shows the basic structure of the controller

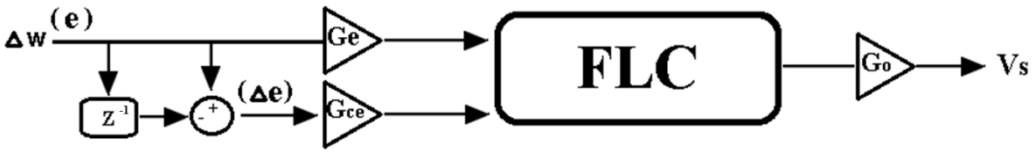


Figure4.11 Damping Controller Structure

4.3.1 Choosing The Controller Inputs

As a starting point, the controller inputs are needed to be chosen such a way that they can help in figuring out the power oscillation in the systems, based on that both the change in power and generation speed deviation can be used. Changing in power may not be a proper choice because the power is not constant as it changes all over the day/ session / year. On the other hand, Speed deviation can be used as it is inversely proportional to the change in the power. Because there is relationship between the speed and the frequency which is the 2π, and also

there is another relationship between the power and the frequency which is that the system frequency will goes down if the power goes up and vice versa.

Therefore the change in speed could be a better choice as the existing systems has either 314 RPM (50Hz) or 377PRM (60Hz). Hence, both Speed deviation and its derivative have been chosen as inputs to the FLC.

4.3.2 Choosing The Membership Functions Limits:

Many possibilities have been investigated in order to find out what could be considered as semi optimal and they are shown below Figure4.12. Appendix-C show some of the membership function's tested limits.

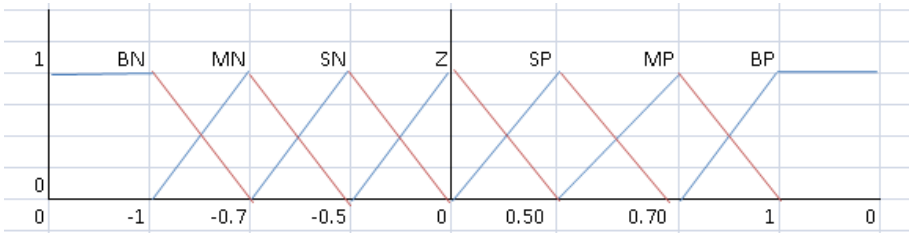


Figure4.12 Triangle Membership function

4.3.3 Fuzzification Stage

As one of thesis targets is to implement the controller practically through the microcontroller and RTDS, therefore, a Fuzzification method that needs less computational power has to be chosen.In other words, in order to let the controller to be a real-timecontroller, it should be able to response to the input in immediately. Based onabove, triangle membership function, shown in Figure4.12,

has been chosen. Furthermore the triangle membership function could be considered as an approximation to the Gaussian membership function.

In Fuzzification stage, seven linguistic Variables per input & output have been chosen forming a total of 21 linguistic Variables, and they are presented in Table4.1.

Table4.1 Linguistic Variables

#	Variable	Speed Dev.	Change in Speed Dev.	Output
1	Big Negative	E_BN	DE_BN	BN
2	Medium Negative	E_MN	DE_MN	MN
3	Small Negative	E_SN	DE_SN	SN
4	Zero	E_Z	DE_Z	Z
5	Small Positive,	E_SP	DE_SP	SP
6	Medium Positive	E_MP	DE_MP	MP
7	Big Positive	E_BP	DE_BP	BP

4.3.4 Inference Engine & Rule Base

Mamdani method has been chosen for the hardware implementation due to its simplicity. Based on the lowest settling time criteria, fifteen rule bases (7x7) have been tested prior to choosing the one shown in Table 4.2 for the damping controller. Actually, two rule bases have been chosen No.3 & No.9 (in Appendix-B) but with further investigation (combined source signal input) it is found out that Table 4.2 could be consider as the best rule base among them. Appendix-B

contains the tested 15 rule bases and the related system performance (based on single source input).

Table 4.2 Main Controller Rule Base

		derr						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Err	E_BN	BN	BN	BN	SN	Z	Z	SP
	E_MN	BN	BN	MN	SN	Z	SP	MP
	E_SN	BN	BN	MN	Z	SP	MP	LP
	E_Z	BN	MN	SN	Z	SP	MP	LP
	E_SP	BN	MN	SN	Z	MP	LP	LP
	E_MP	MN	SN	Z	SP	MP	LP	LP
	E_BP	SN	Z	Z	SP	LP	LP	LP

4.3.5 De-Fuzzification

The main purpose of the de-Fuzzification stage is to convert the linguistic Variables from the inference engine into crisp output value. Weighted average method described earlier in this chapter has been used, as it does not need huge computational power compared to the other techniques / methods. Furthermore, it does not need output triangle membership functions.

4.3.6 Controller Tuning (Choosing Gain's Values Ge, Gce & Go)

The controller has been designed such a way that it can be tuned by changing some gains. Those gains are Ge, Gce and Go which are Error gain (speed deviation), Change in error gain (change in speed deviation), and output gain respectively.

Based on above the tuning process has been as follows:

1. Finding the limits of each gain
2. Conducting a deep searching within the limits

Stage 1:

In order to find the limits or the initial gains the steps below has been followed:

1. The Error or the speed deviation has been found by conducting many simulation scenarios plus adding 10 % on the top of the max reading as a safety margin to keep the read values away from the saturation.
2. The range of change in Speed deviation has been identified using trial and error to find the initial values.
3. By trial and error the output gain has been identified.

Stage 2:

In order to find more tuned value, a deep searching within the limits has been conducted, it is mainly for the change in speed deviation gain and the output gain.

The deep gains searching have been automated using MATLAB m-code for SIMULINK, and the C scripting for the RTDS.

4.3.7 Online Adaptation Mechanism For The Fuzzy Logic controller

After conducting the simulation in RTDS, it is found that the performance of the controller drops as the load in the tie-line goes up, and it is needed to be re-tuned in order to have a better performance (the problem is described in details in

chapter No. 5). Based on that the controller structure has been modified to give the controller the ability of retuning itself.

Table4.3shows the tuned values needed to extend the range of better performance:

Table4.3 Loading Cases

#	Case	100MVAr			200MVAr		
		Error Gain	Change in Error Gain	Output Gain	Error Gain	Change in Error Gain	Output Gain
1	+20%	5	0.5	1	5	1	0.42
2	+15%	5	0.5	1	5	1.15	0.42
3	+10%	5	0.75	1	5	1.3	0.42
4	+5%	5	1	1	5	1.4	0.42
5	+0%	5	1.5	1	5	1.5	0.42
6	-5%	5	1.75	1	5	1.6	0.42
7	-10%	5	2	1	5	1.7	0.42
8	-15%	5	2.25	1	5	1.8	0.42
9	-20%	5	2.5	1	5	1.9	0.42

As can be seen from the Table above the needed tuning is only in the gain of change in Error (change in speed deviation).The gain of change in the speed deviation along with the tie-line loading has been correlated as shown in Figures4.13 and 4.14 as well as correlating equations 4.1& 4.2below.

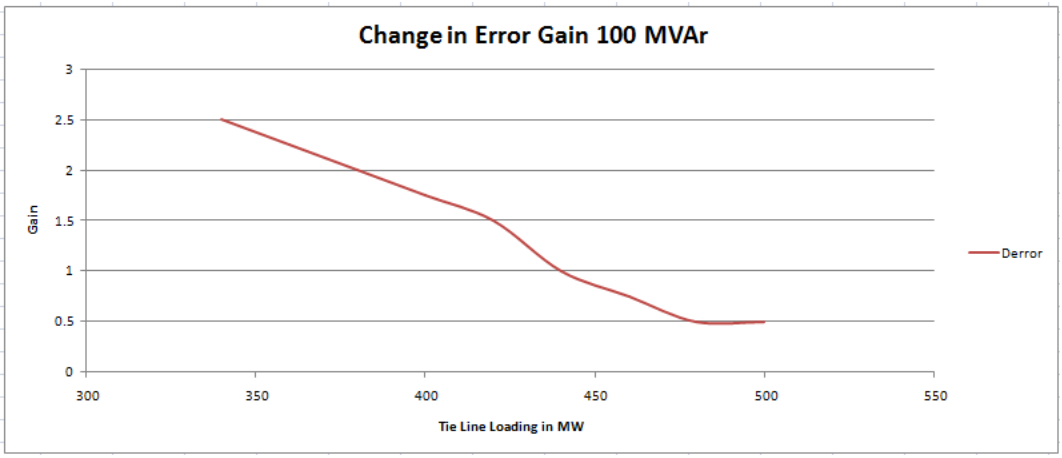


Figure4.13 Change in Error Gain Curve (100MVAr SVC)

$$\text{Gain} = -0.01375 \text{ x (Tie Line Loading)} + 7.19167 \quad \text{.....} \quad 5.1$$

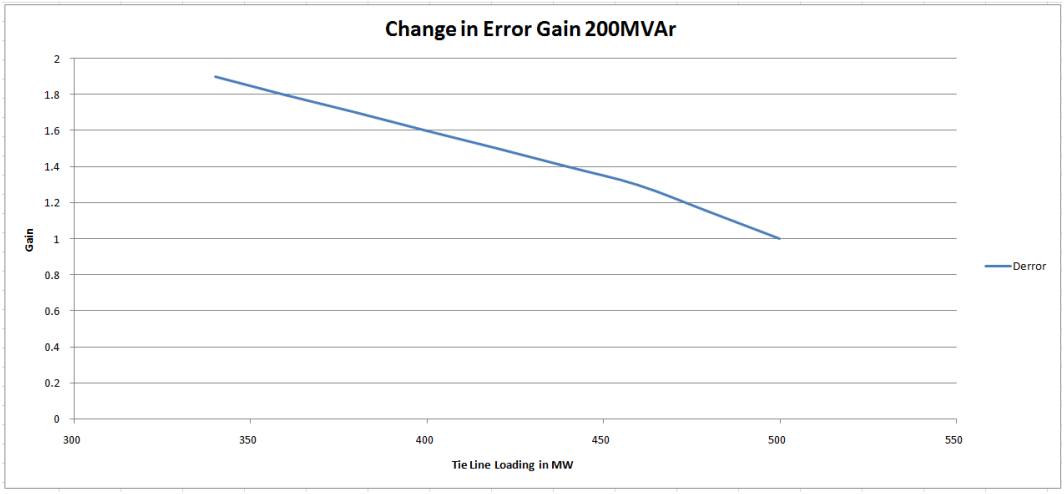


Figure4.14 Change in Error Gain Curve (200MVAr SVC)

$$\text{Gain} = -0.00546 \text{ x (Tie Line Loading)} + 3.77583 \quad \ldots\ldots\ldots \quad 5.2$$

As can be seen the relationships are fairly linear, and the R^2 measures of equations 4.1 & 4.2 are 0.98108 and 0.99311 respectively. The major thing with these two equations is that they need the Steady State Power value in the tie-line, but in our case, we are dealing with in the transient period. To incorporate those equations in our controller, we have to find the proper time to capture the power value. In order to do that we have proposed to capture the power values at the time where:

1. The rate of change of power is almost Zero, and
2. Its rate of change is almost Zero

The above methodology will have a main drawback that it will capture the peaks and bottoms the power values in transient period as well as the needed value. Therefore a limiter has been added to pass the proper power values. Based on above, Fuzzy logic can be used to define when the power values should be captured and when "change in error" gain should be modified. Figure4.15 shows the new structure of the fuzzy controller.

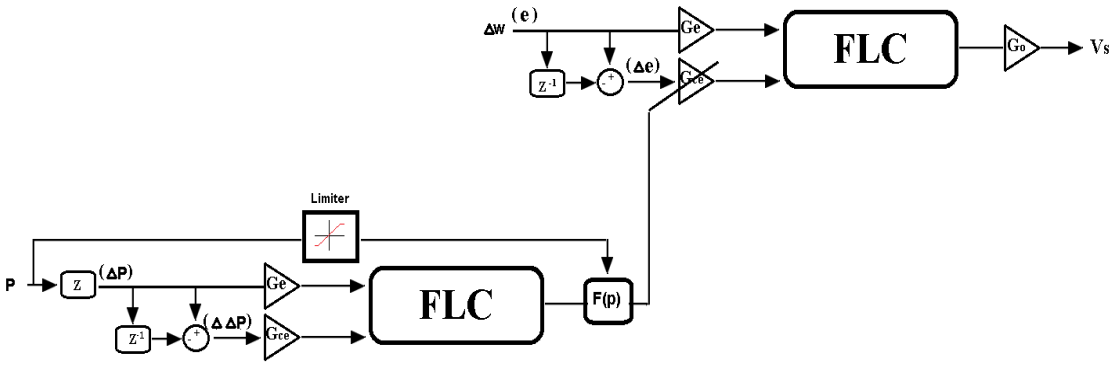


Figure4.15 FLC Controller with the Adaptation Mechanism

The rule base and the input/output membership functions used are shown Table4.4 and Figure4.16.

Table4.4 Rule Base of the Second Fuzzy Controller

		derr						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
err	E_BN	Z	Z	Z	Z	Z	Z	Z
	E_MN	Z	Z	SP	SP	SP	Z	Z
	E_SN	Z	Z	SP	LP	SP	Z	Z
	E_Z	Z	Z	LP	LP	LP	Z	Z
	E_SP	Z	Z	SP	LP	SP	Z	Z
	E_MP	Z	Z	SP	SP	SP	Z	Z
	E_BP	Z	Z	Z	Z	Z	Z	Z

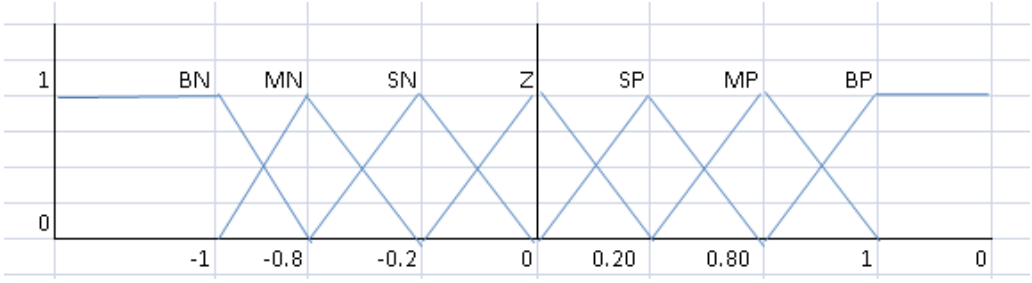


Figure4.16 Inputs & Output Membership Function

The limiters value used are 413MW±20%.

Chapter 5. Simulations and Results

In this chapter, the tested power system model as well as the fuzzy damping controller, described in chapters3 & 4,have been incorporated / modeled in both MATLAB/ SIMULINK and RTDS in order to evaluate the damping controller performance against a severe disturbance, such as three phase short-circuit. It is worth noting that MATLAB/SIMULINK simulation program is used to prove that SVC damping controller design can damp the inter-area oscillation without the presence of any PSS. On the other hand, RTDS is used for conducting further studies, analysis, and tuning as well as extending the range of operation points by making an online adaption mechanism. This chapter will be divided into two main sections related to simulation programs which are the MATLAB/SIMULINK and RTDS sections.

5.1. **MATLAB/ SIMULINK**

SIMULINK program under MATLAB environment has been used to simulate the damping controller within the two areas four machines testing model, as shown in Figure5.1. Recalling our studying case, Figure3.4.

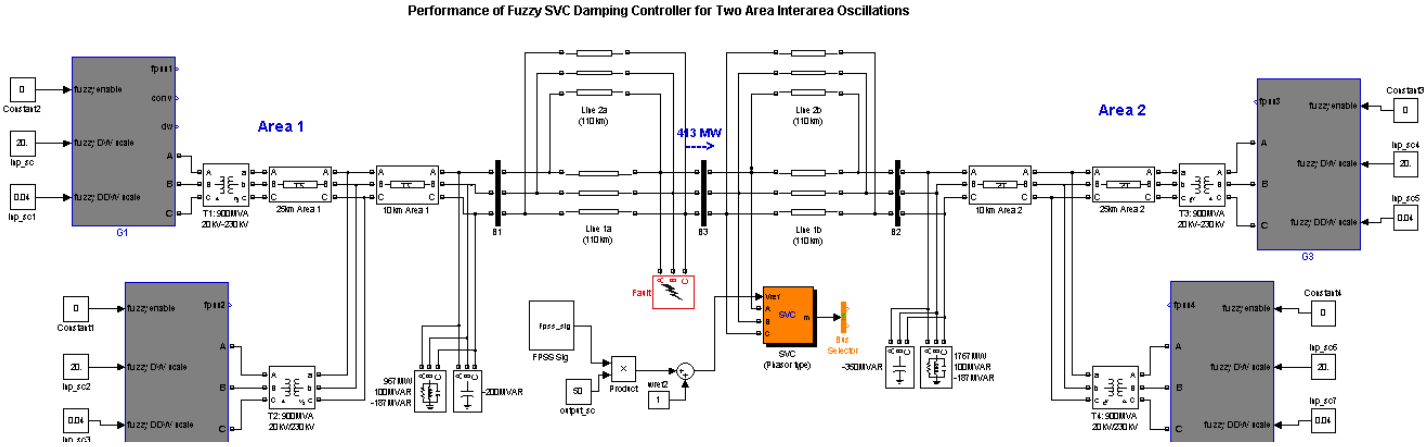


Figure5.1 MATLAB Model of Two Areas and Four Machines

In the first case, the power system stabilizers have been removed formal of the four machines. Three phase fault in the midpoint of the tie-line has been applied for 200 ms (self cleared).As shown in Figures5.2& 5.3 below, the system is unstable as there are oscillation in the both the power flowing through the tie-line and the voltages at both ends of the tie-line. The causes of this oscillation are both the exciters' high gain that reduces in the generation damping nature, and also the heavy loaded tie-line between the two areas.

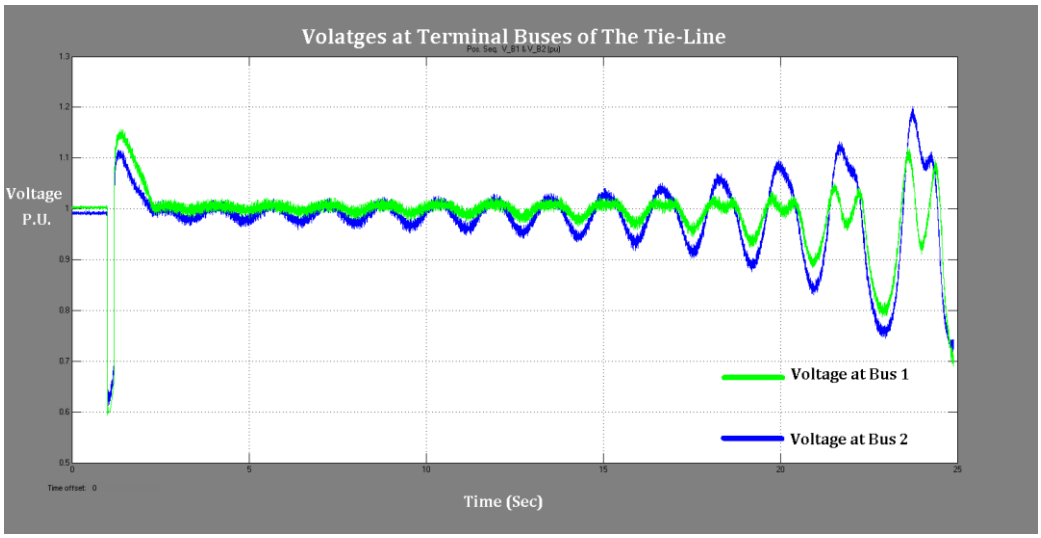


Figure5.2 Tie-Line Terminal Voltages, Time Response Under200 ms Fault@ Bus 3 (No PSS)

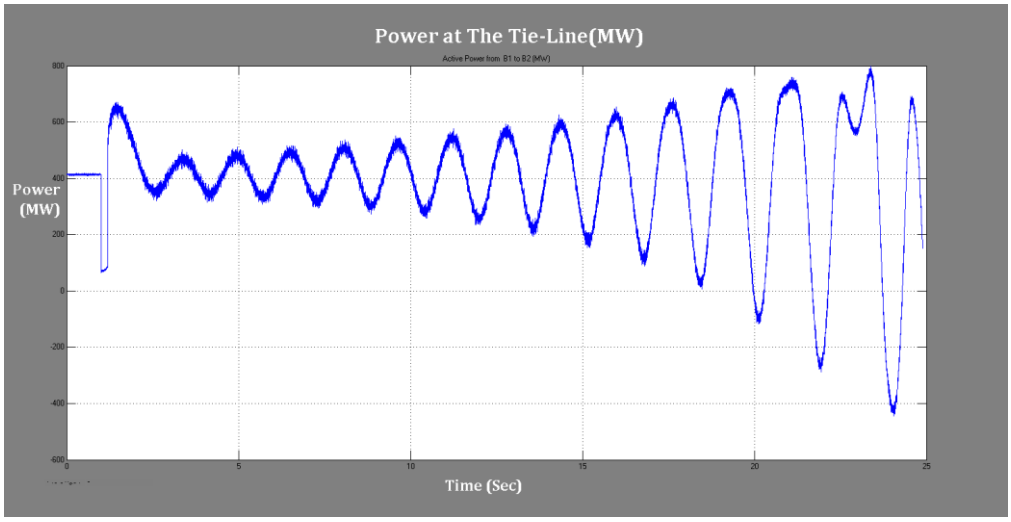


Figure5.3 Tie-Line Power, Time Response Under 200 ms Fault@ Bus 3 (No PSS)

In the second case, SVC has been installed in bus no. 3,where it is located in the midpoint of the tie-line. That location has been chosen to

1. Improve midpoint voltage,
2. Help in damping the oscillation.

But as shown in Figures5.4 & 5.5 the system tends to be unstable faster than the first case. It last about 20 sec compared to 25secin the first case.

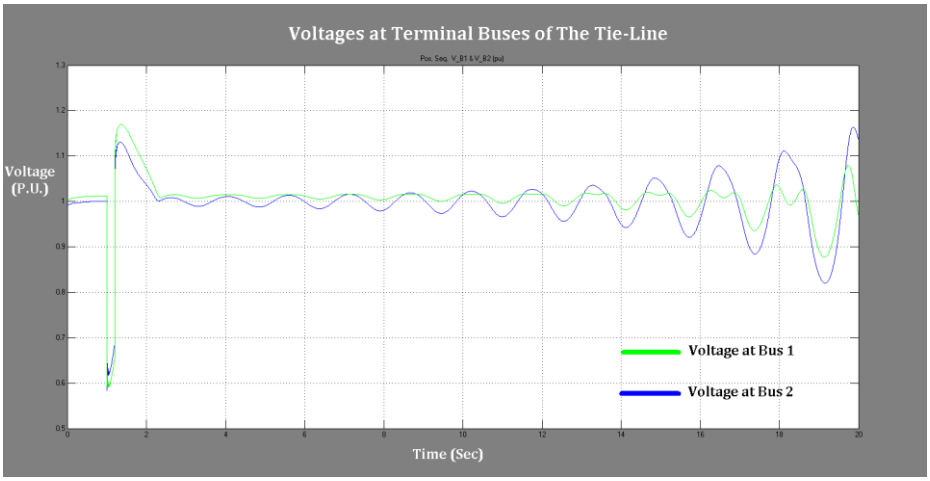


Figure5.4Tie-Line Terminal Voltages, Time Response Under 200 ms Fault @ Bus 3 (No PSS, with SVC)

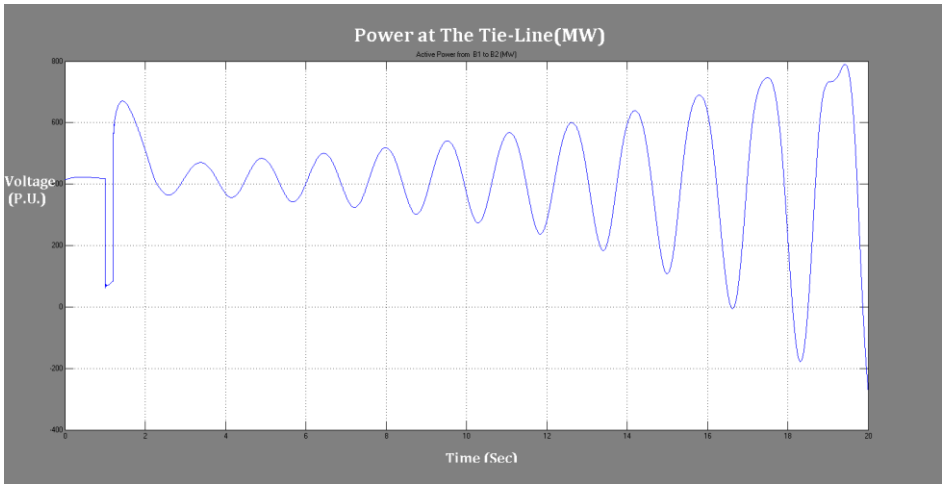


Figure5.5Tie-Line Power, Time Response Under 200 ms Fault @ Bus 3 (No PSS, with SVC)

In the third case, the auxiliary signal, from the damping controller, has been Incorporated along with the SVC reference voltage. This issue helps in

making the system tends to be stable as shown in Figures5.6& 5.7. It will take a long time to be stable more than 30 sec but this issue gives an indication that there is a possibility to have better results by using tuning technique.

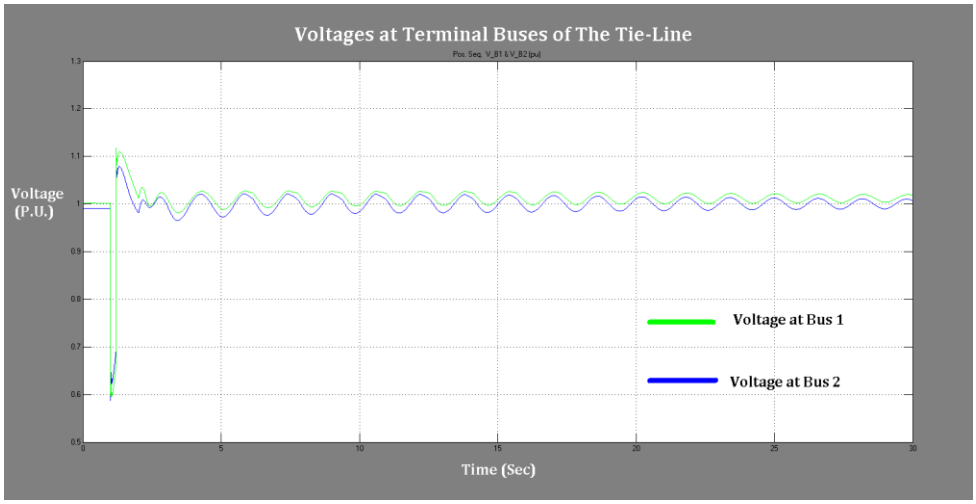


Figure5.6Tie-Line Terminal Voltages, Time Response Under 200 ms Fault @ Bus 3 (No PSS, with SVC &FL Damping Controller)

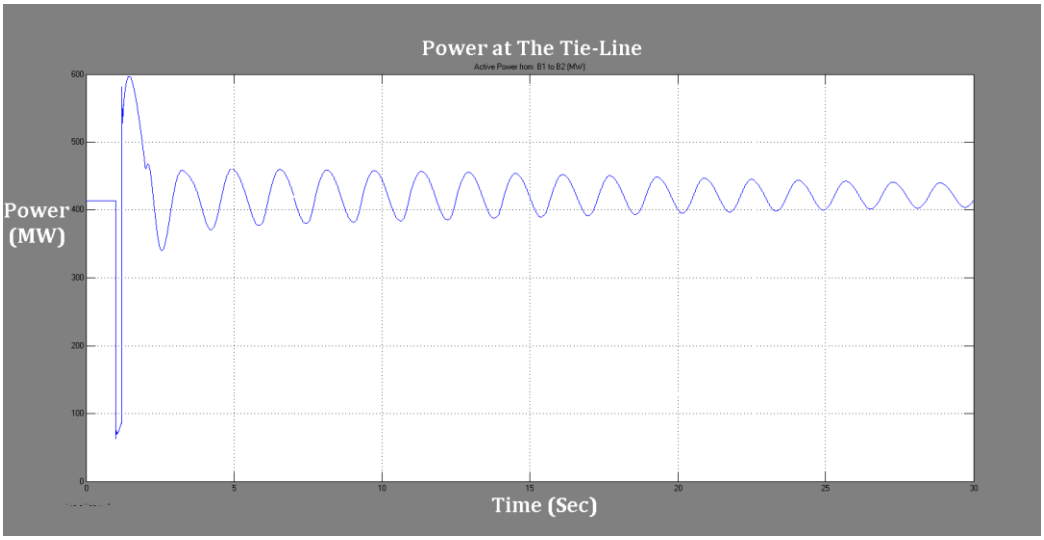


Figure5.7Tie-Line Power, Time Response Under 200 ms Fault @ Bus 3 (No PSS, with SVC &FL Damping Controller)

5.2. **RTDS Model**

The Fuzzy logic controller will be tuned further in RTDS environment as it is faster and has more detailed power systems models. Furthermore, RTDS has the ability of changing the controller's parameters online during the simulation rather than changing them after conducting the entire simulation. Moreover, RTDS has one main feature over the MATLAB that RTDS can be used for conducting Real Time Hardware In the Loop test, that will be used for FLC implementation.

Figure5.8 shows the draft file of RTDS model of two areas four machines testing system.(More clear Model is in Appendix E).

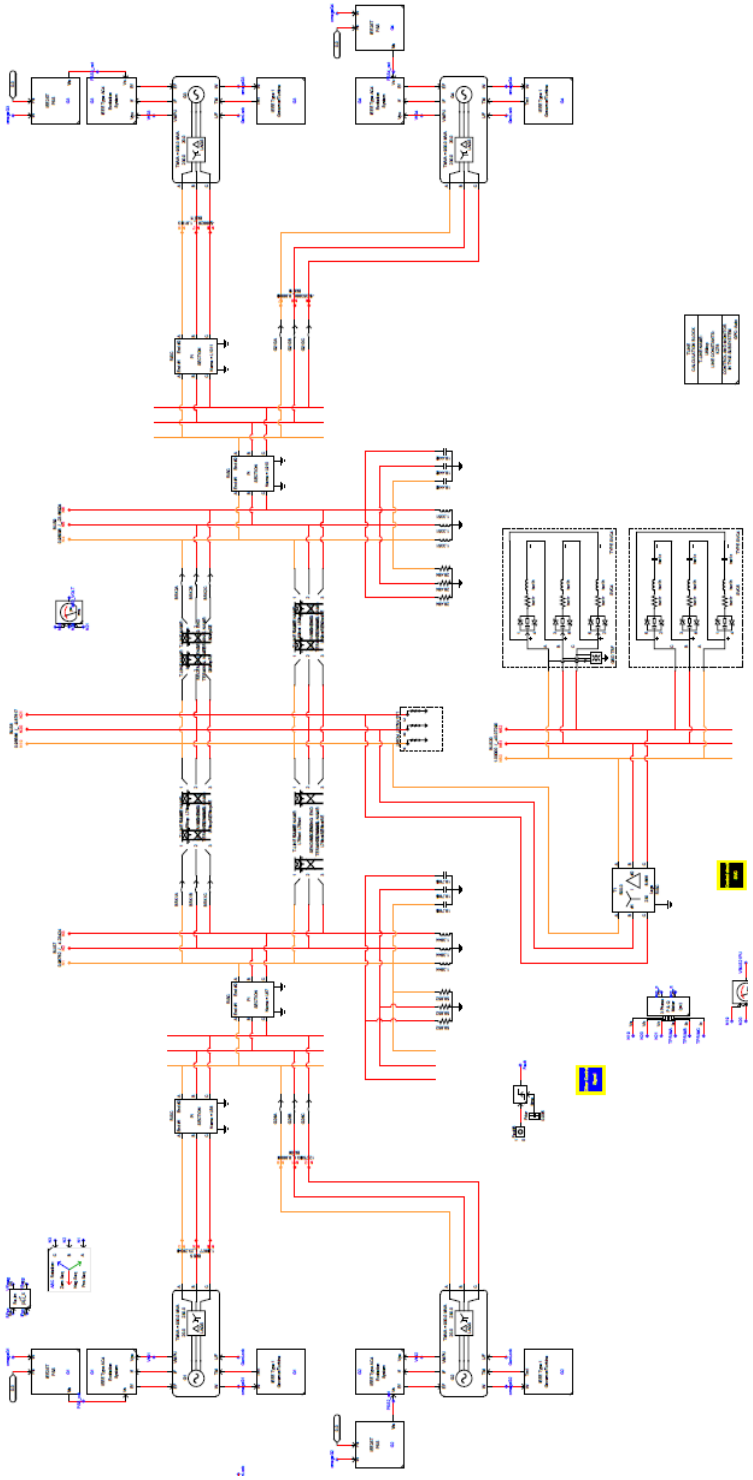


Figure5.8 RTDS Model

RTDS system has no Fuzzy Logic controller model, and since one of our objectives is to design Fuzzy logic damping controller, there is a need to build model, based on that we have built an RTDS component, shown in Figure 5.9, using component builder feature as well as our developed tool for auto-code generator for that model. More details regarding the tool are available in Chapter 6.

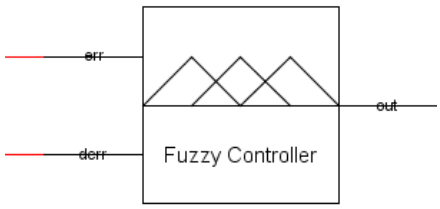


Figure5.9RTDS Fuzzy Model

RTDS model is considered as more accurate program than MATLAB as their models have much more details. For instance, the synchronous machine in RTDS has a saturation curve while in MATLAB it does not. In power system simulation, RTDS can capture more details, such as in our case RTDS can show the transient effect of starting machine which will contribute in showing the inter-area oscillation phenomena, without Power System Stabilizers, even without having a big disturbance (three phase short circuit) as shown in Figures 5.10 and 5.11. This oscillation, as stated before, is due to both high gain exciter, that cause

reduction in generation damping torque, and the heavy loaded tie-line between the two areas.

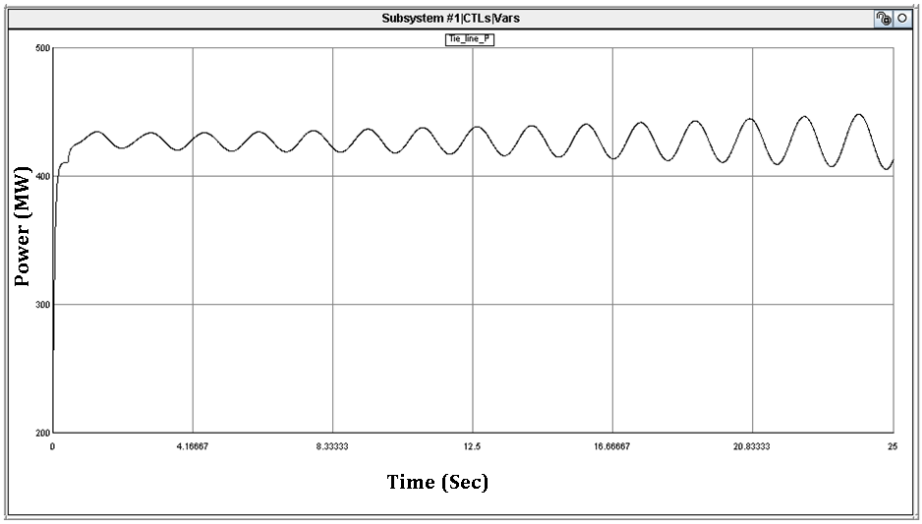


Figure 5.10Tie-Line Power, Time Response without Fault & No PSS (Inter-Area Oscillation phenomena)

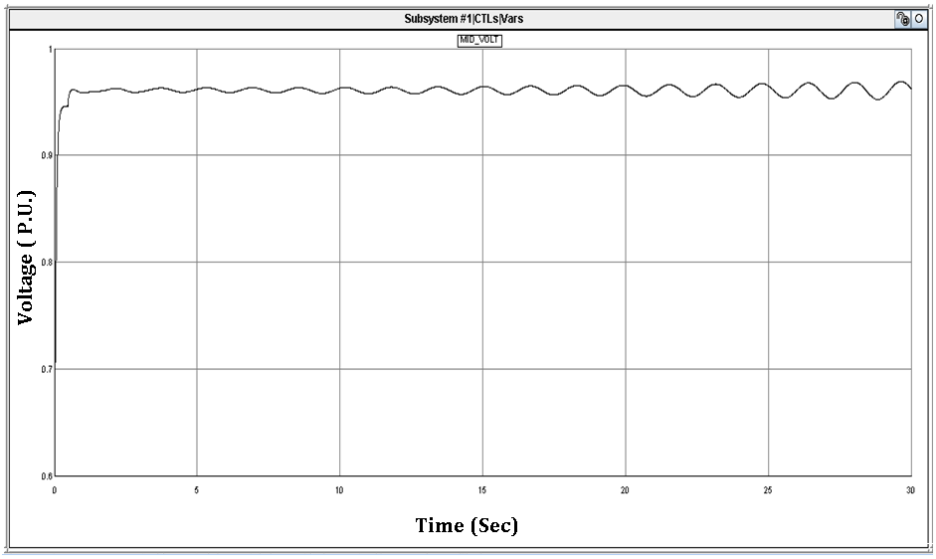


Figure 5.11Tie-Line Voltage (Midpoint),Time Response without Fault & No PSS (Inter-Area Oscillation phenomena)

Adding SVC, without damping controller, to the midpoint of the tie-line does not help that much in eliminating the oscillation. It only delays the oscillation as shown in Figures 5.12 and 5.13. This issue is expected as the SVC will react only to the voltage Variation without considering the power oscillation.

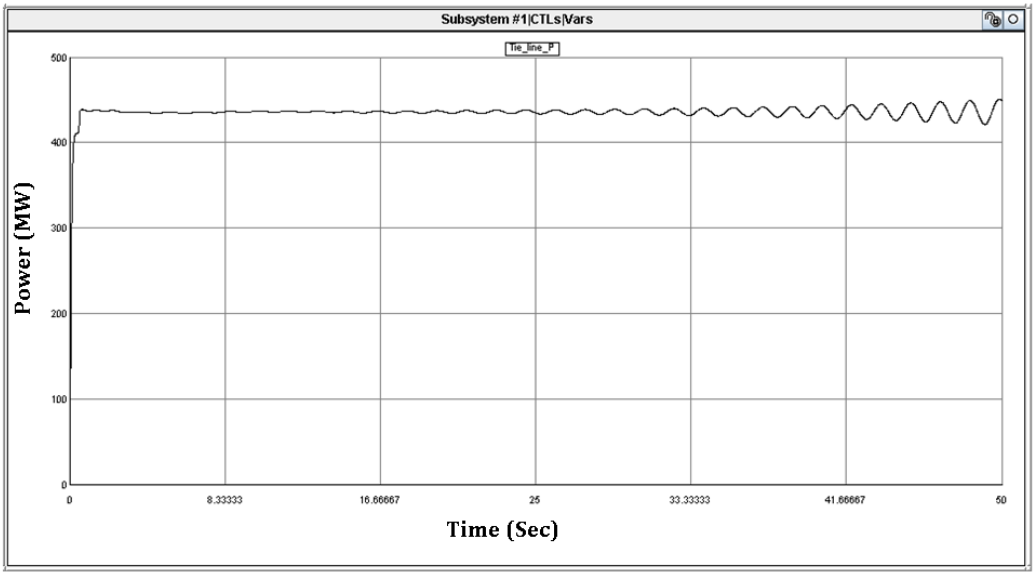


Figure5.12 Tie-Line Power, Time Response without Fault, No PSS, with SVC only (Inter-Area Oscillation phenomena)

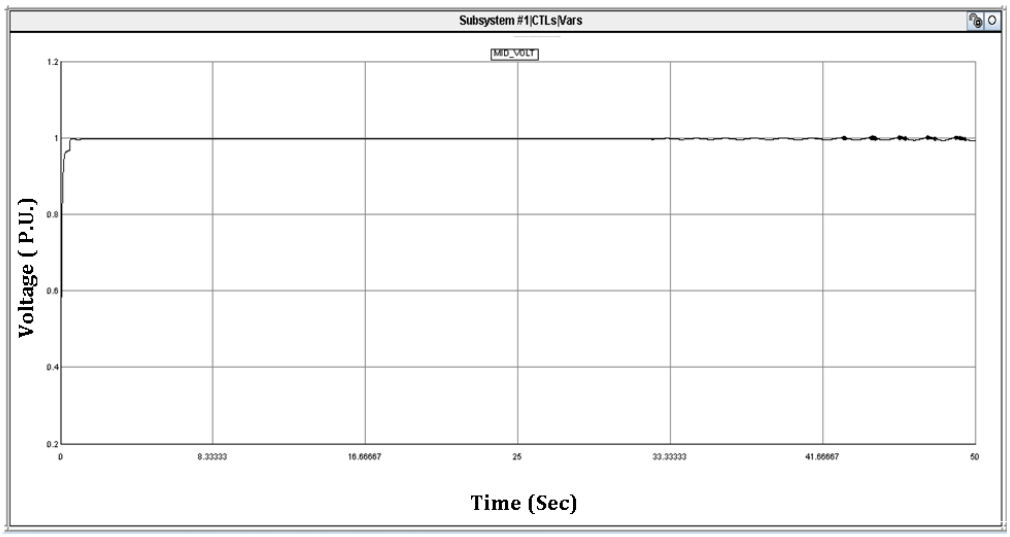


Figure5.13 Tie-Line Voltage (Midpoint),Time Response without Fault, No PSS, with SVC only (Inter-Area Oscillation

On the other hand, adding damping controller, that uses speed deviation as an input, will add another dimension to the SVC which is the reaction against the power oscillation. This issue will help in increasing the system oscillation damping, it also could be considered as the PSS effect on the AVR/ excitation system, as shown in Figures5.14& 5.15.

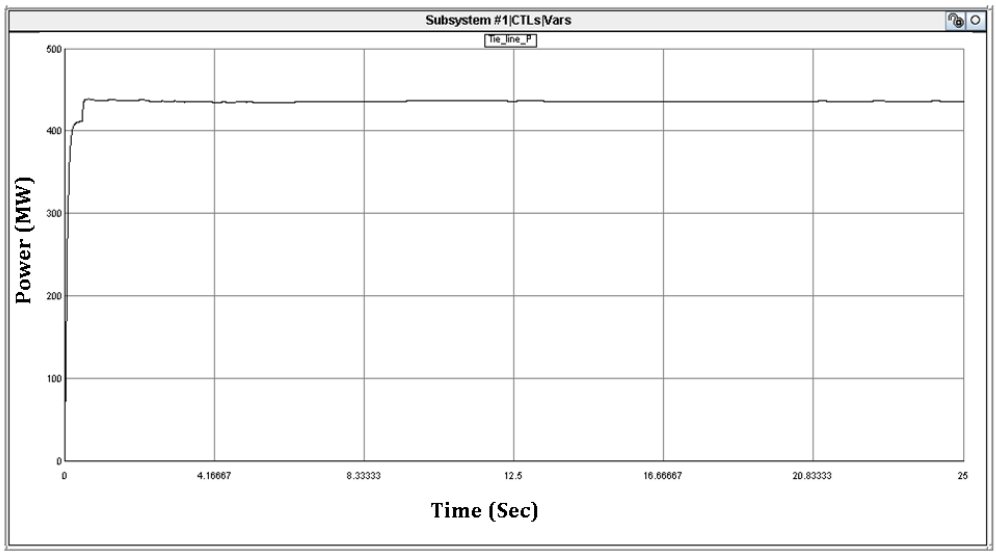


Figure5.14Tie-Line Power, Time Response without Fault, No PSS, with SVC &FL Damping Controller (Inter-Area Oscillation phenomena)

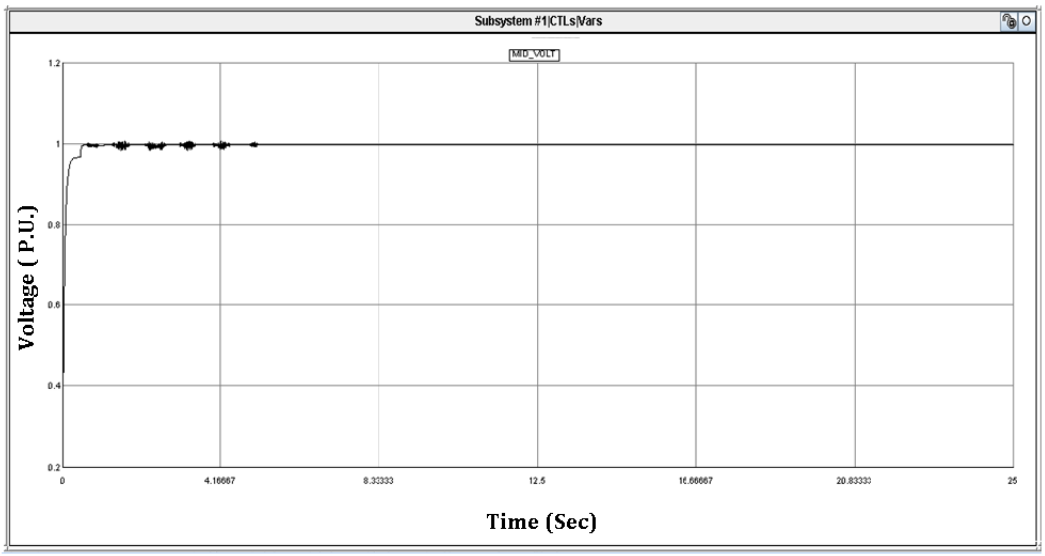


Figure5.15Tie-Line Voltage (Midpoint),Time Response without Fault, No PSS, with SVC &FL Damping Controller(Inter-Area Oscillation phenomena)

System Performance Without Damping Controller

In order to evaluate the controller performance along with big disturbance, a three phase short circuit test with duration of 200ms (self cleared) has been applied to midpoint bus of the tie-line, where the SVC is installed.

The same scenarios applied, before, in MATLAB will be applied here (with two more) which are:

1. Having three phase short circuit without the presence of both PSS and SVC.
2. Having three phase short circuit with SVC only.
3. Having three phase short circuit with SVC (100MVAr) and damping controller.
4. Having three phase short circuit with the presence of PSS & SVC (100MVAr) & damping controller.

Scenario 1: 200 ms 3 Phase Fault without PSS & Without SVC

Both Figures 5.16 and 5.17 show the power flowing and the voltage behaviors in the tie-line midpoint. As can be seen the system is unstable.

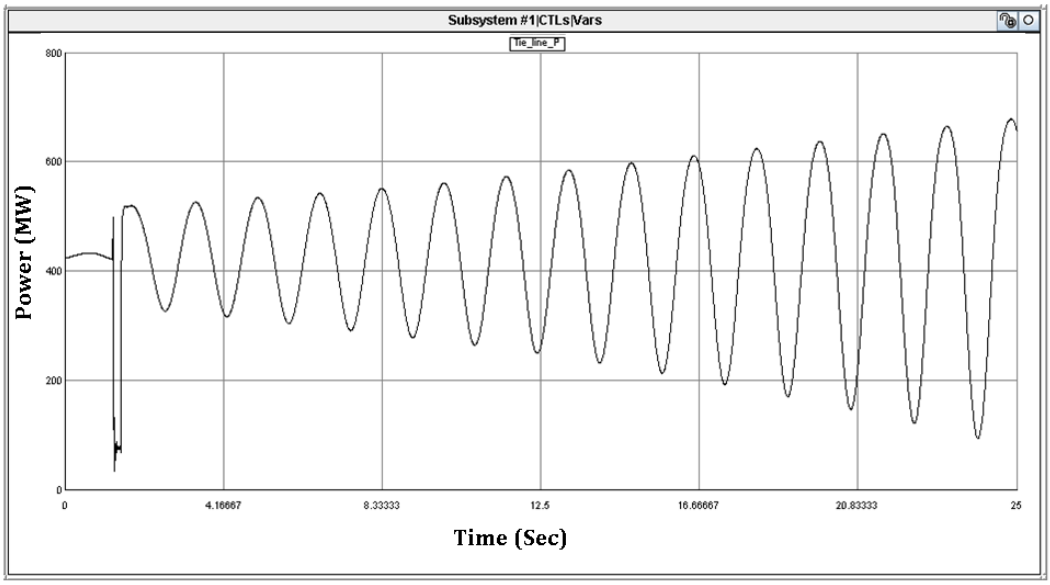


Figure5.16 Tie-Line Power, Time Response Under 200 ms Fault @ Bus 8 (No PSS)

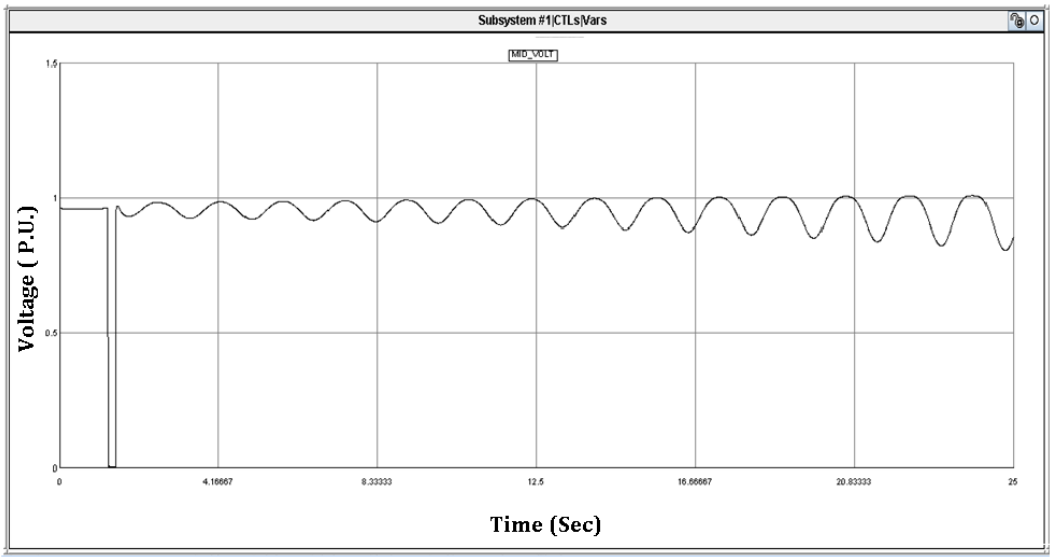


Figure5.17 Tie-Line Midpoint Voltages, Time Response Under 200 ms Fault @ Bus 8 (No PSS)

Scenario 2: 200 ms 3 Ph Fault without PSS & With SVC

The presence of SVC does not eliminate the oscillation, it only helps in regulating pre-fault voltage to be around 1 P.U. The SVC presence contributes in making the power oscillation worse, as it leads the system to be unstable faster as shown in Figures 5.18 and 5.19.

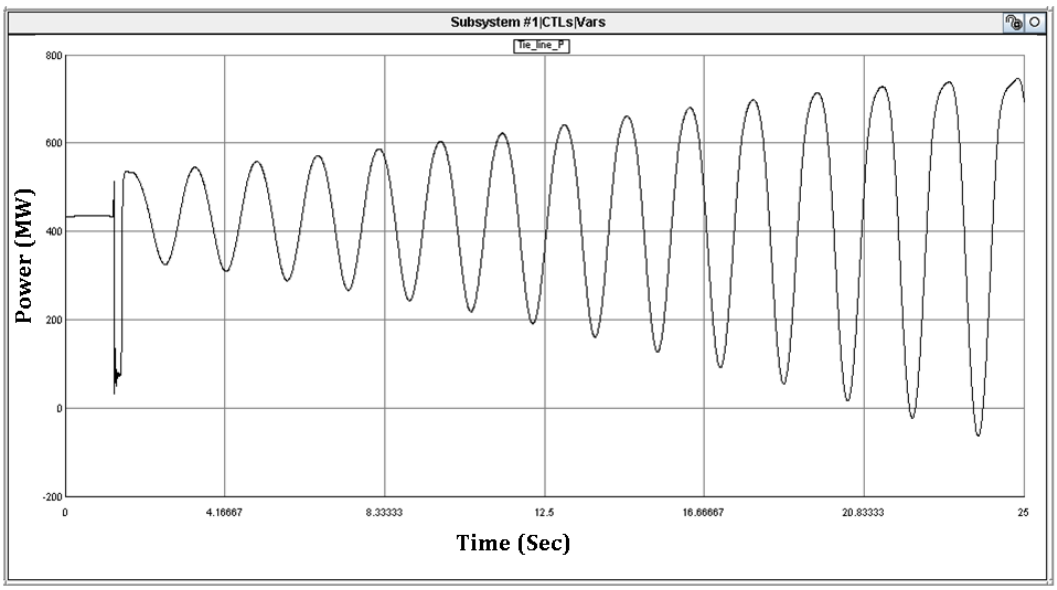


Figure5.18 Tie-Line Power, Time Response Under 200 ms Fault @ Bus 8 (No PSS, with SVC)

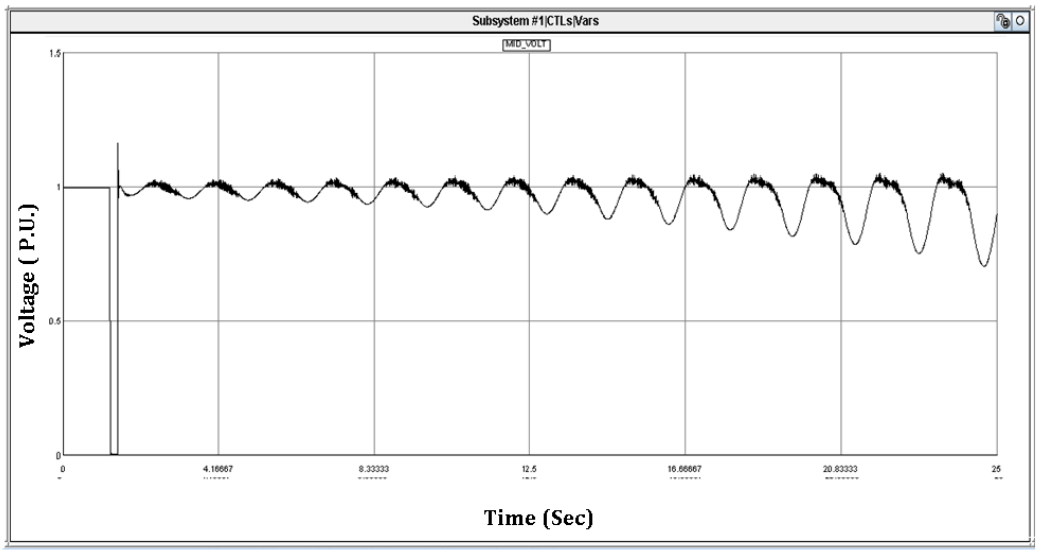


Figure 5.19Tie-Line Midpoint Voltages, Time Response Under 200 ms Fault @ Bus 8 (No PSS, with SVC)

Scenario 3: 200 ms 3 Ph Fault without PSS & With SVC & Damping Controller

In order to evaluate the damping controller performance, the input sources has to chosen first.

Choosing Source Inputs of FL Damping Controller

As stated before the speed deviation will be used as input to the controller, but from which generation/ substation? Our strategy is based on starting with single source and then moving toward a combination of sources as shown in the followingTable5.1.

Table5.1 Controller Input Combination

#	Source	Results	Re mark
1	Gen. 1	Stable	Gen. 1 is better than Gen. 2
2	Gen. 2	Stable	
3	Gen. 3	Unstable	
4	Gen. 4	Unstable	
5	Gen. 1 - Gen. 3	Stable	(Gen. 1 - Gen. 3) is better
6	Gen. 2- Gen. 4	Stable	than (Gen. 2- Gen. 4)
7	Gen. 1 + Gen. 3	Unstable	
8	Gen. 2 + Gen. 4	Unstable	
Overall			Gen. 1- Gen. 2 is the best

PMU Measurements VS Generation Measurements

The measurement of speed deviation at generator no.1 terminal using PMU is fairly similar to the measure speed deviation at the same generator, as shown in Figure5.20, where black curve is PMU measurement and red curve generator measured speed deviation. There are some noises in the PMU signal because of PMU's refresh rate 60 times per second, which we do believe it is not as fast as it should be. Conventionally, the speed deviation signal is taken from the generators directly, but regardless of the small noises introduced by the small refresh rate, using PMU adds another dimension to the system which is the mobility as the PMU can be placed anywhere, as well as the main advantage of having an alternative controller's input from single or multiple places at the same time.

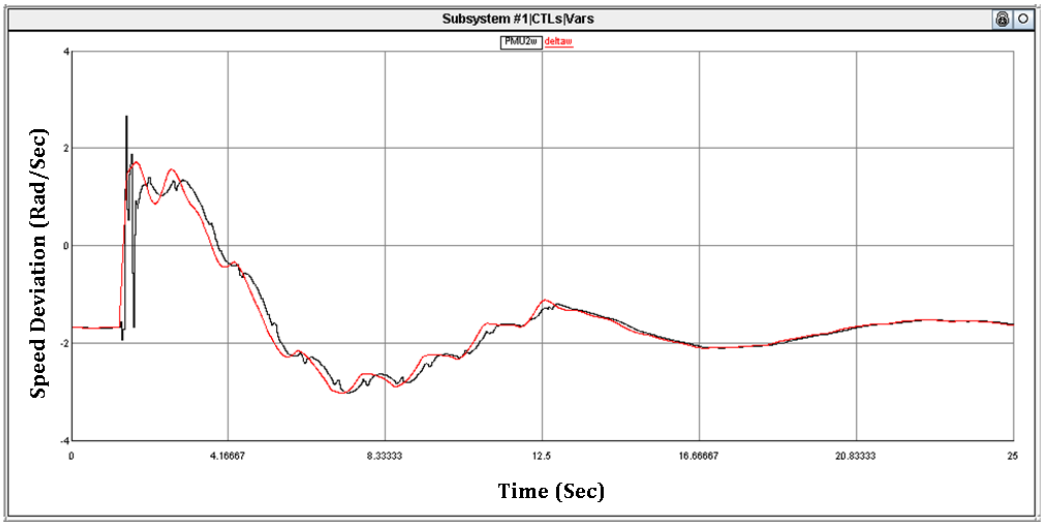


Figure5.20 PMU Measured Signal VS Gen. 1 Measured Signal

Single Source Signal

Regarding a single source FL damping controller's input, Gen. 1 speed deviation was the best as it helps in damping the oscillation in about 20 seconds as shown below in Figure5.21, where both PMU measured value & Gen 1 Speed value are used.

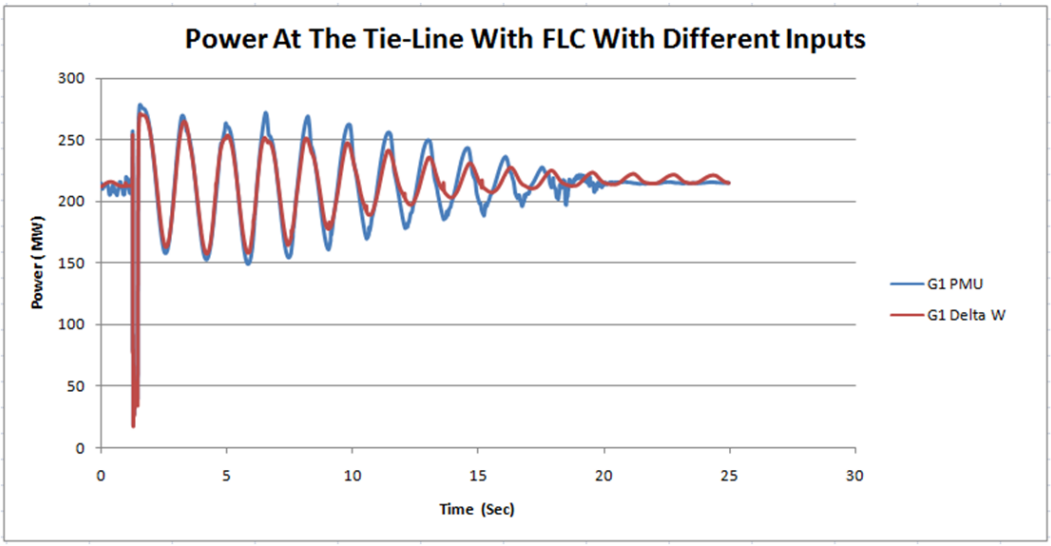


Figure5.21 Tie-Line Power, Time Response Using Different Inputs Measured at Gen 1 Terminal Under Fault Conditions

As can be seen inFigure5.21, using PMU signal is better than generation measured signal as it is settled in 20sec compared to more than 25sec for the other one. It can be noticed that they are somehow similar which could lead us to a conclusion that they are interchangeable. In other words, each one of them can be used as a replacement for each other. Hence, we can propose an input selector mechanism/ strategy which will sense the absence of one of them and then shift to the other one automatically.

Combined Source Signal

Using the difference between the speed deviation signals from Generator -1 located in area one & Generator-3 located in area two (for the FL damping controller) helps in damping the oscillation in about10 second which is far better than using the speed deviation of generator no.1 alone (settled in more than 25sec) as shown in Figures 5.22& 5.23 .

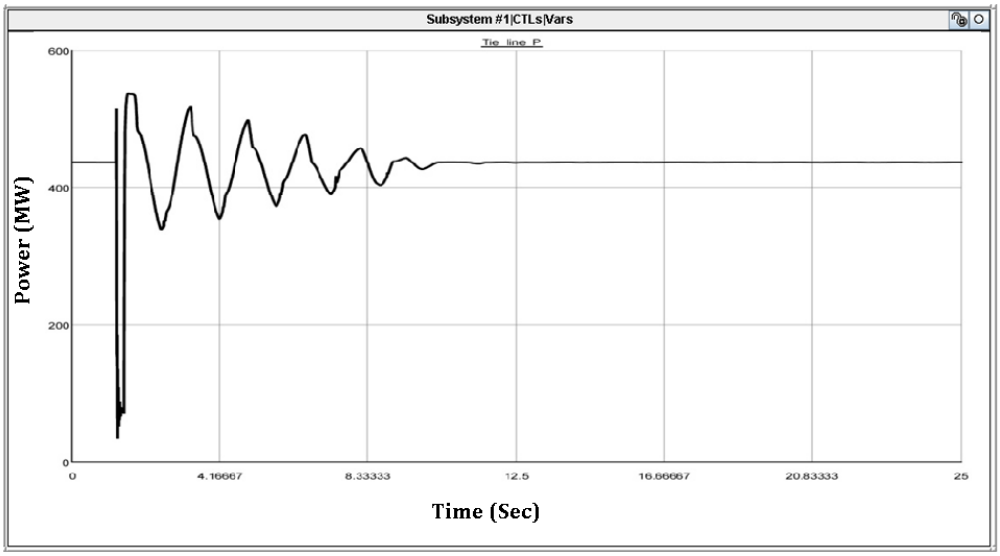


Figure5.22Tie-Line Power, Time Response Using(G1-G3) as Signal Under Fault Condition

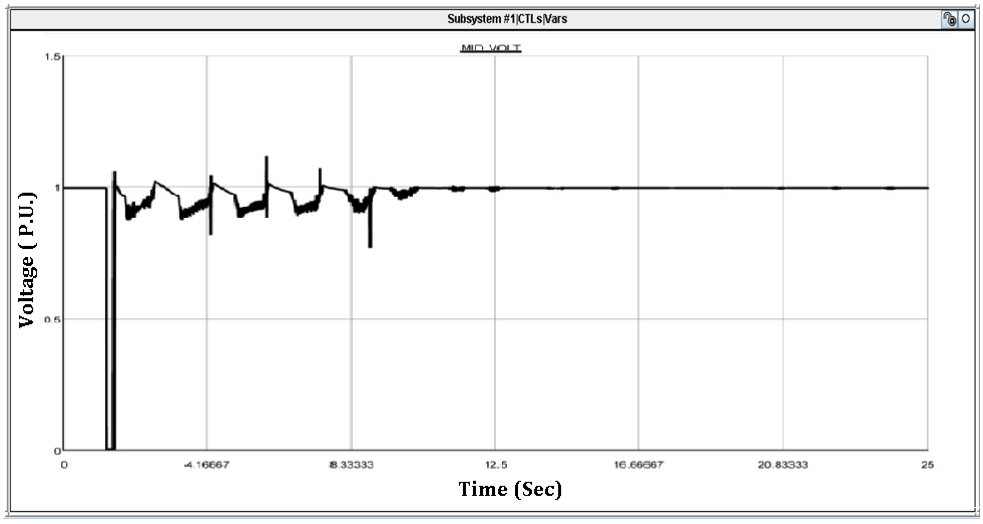


Figure5.23Tie-Line Midpoint Voltages, Time Response Using (G1-G3) Signal Under Fault Condition

Based on above the combination of (Gen.1 - Gen.3)speed deviation will be used as an input to the damping controller.

The Effect of changing the load

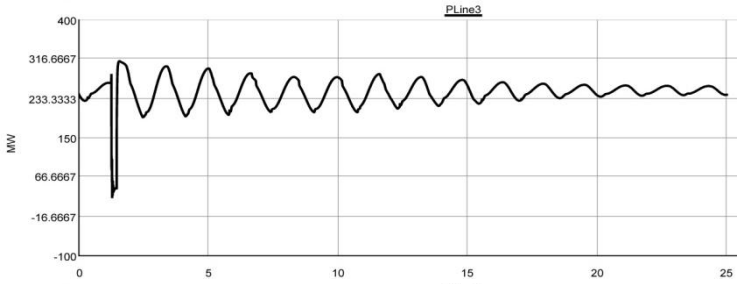
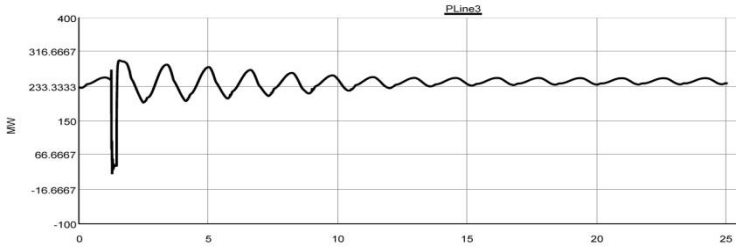
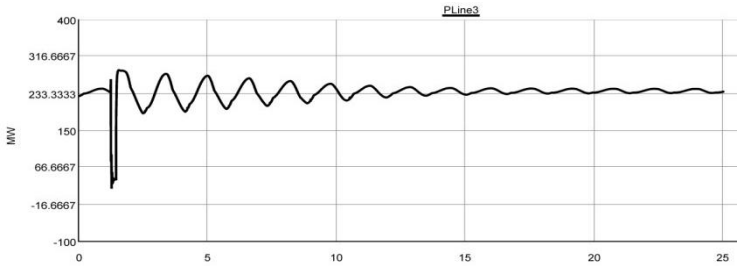
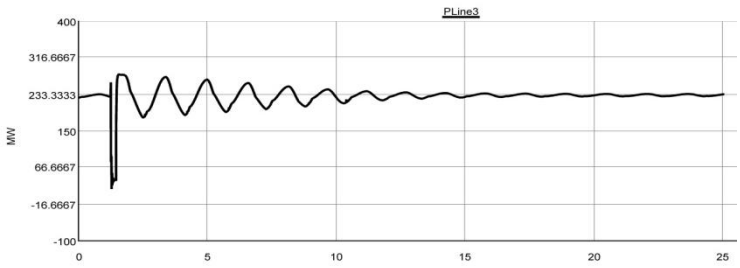
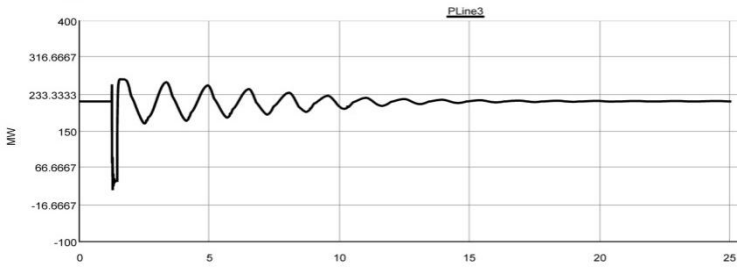
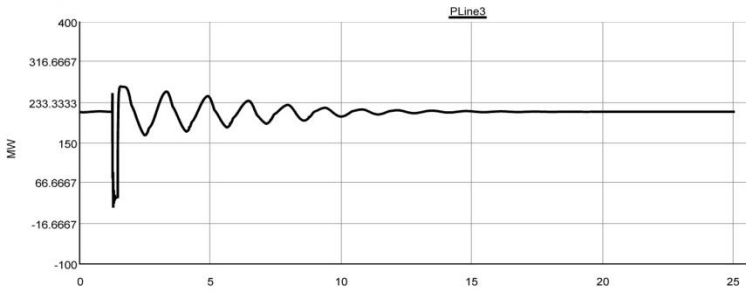
As it is known, the electrical demand changes all over the year and also the global demand grow every year. For instance, in Qatar the gap is very clear between summer and winter, in addition to the average annual load growth of about 10 %. Hence, the effect of load changing against the controller performance has to be studied.

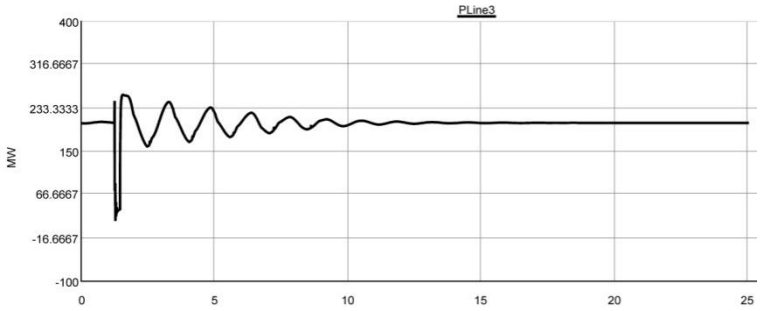
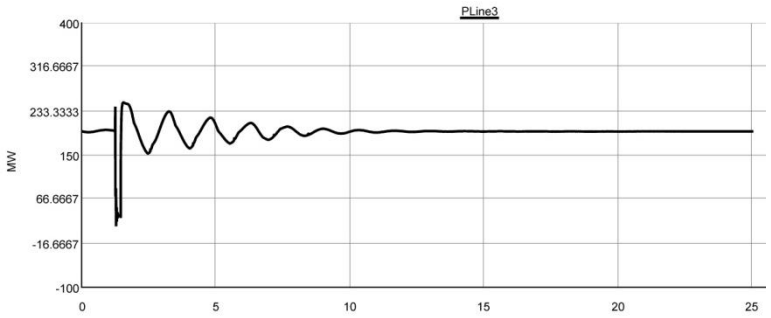
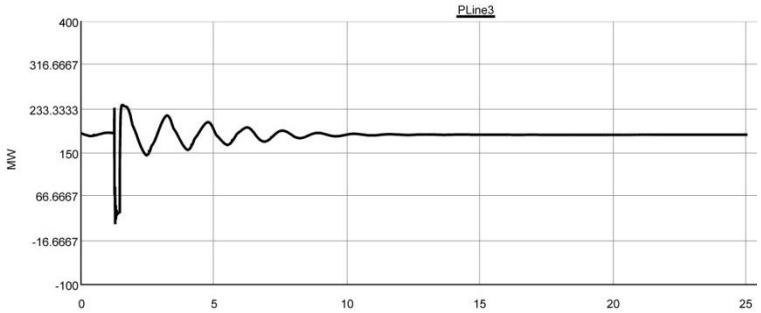
The proposed Variation is $\pm 20\%$ in the base case load in both areas. The used step of increment/decrement is 5%, based on that there will be nine cases as the following:

1. Base Case plus 20% (increment) in load in both areas
2. Base Case plus 15% (increment) in load in both areas
3. Base Case plus 10% (increment) in load in both areas
4. Base Case plus 5% (increment) in load in both areas
5. Base Case
6. Base Case minus 5% (decrement) in load in both areas
7. Base Case minus 10% (decrement) in load in both areas
8. Base Case minus 15% (decrement) in load in both areas
9. Base Case minus 20% (decrement) in load in both areas

Table5.2 below shows the controller performance with (100MVAr SVC)

Table5.2 Load Increment / Decrement Effect

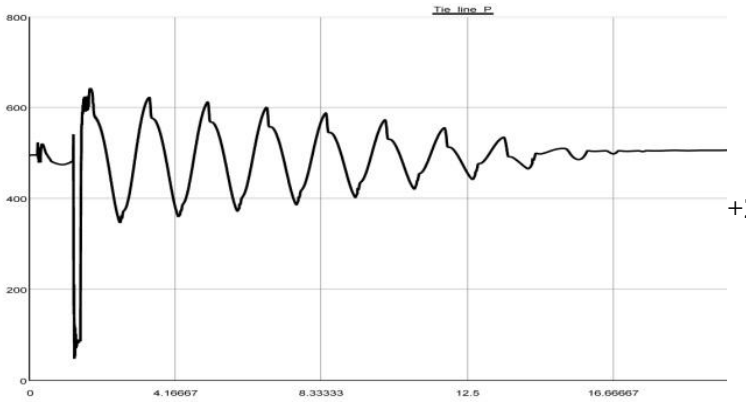
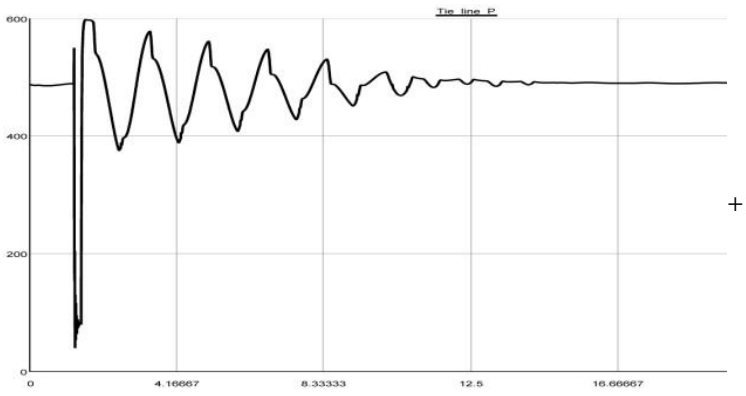
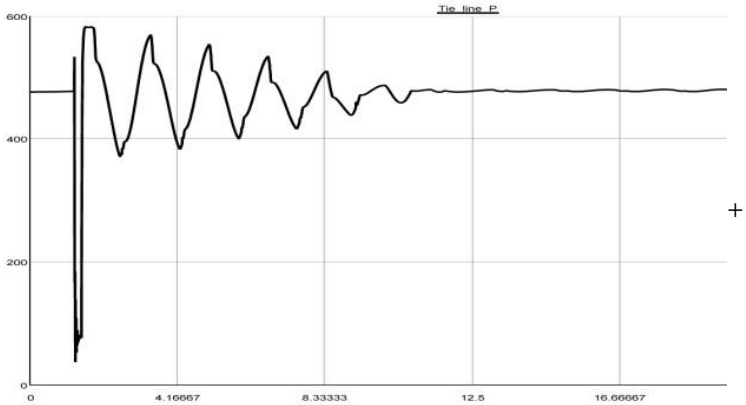
Case #	System response (Tie-line Power Flow)	Load Inc / Dec
1		+20%
2		+15%
3		+10%
4		+5%
5		0%
6		-5%

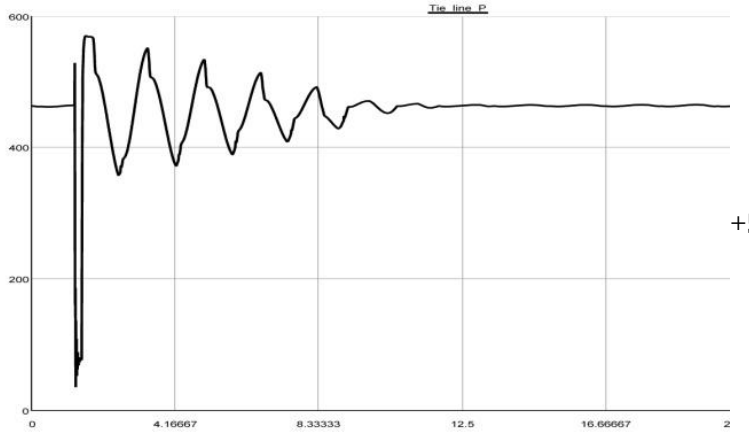
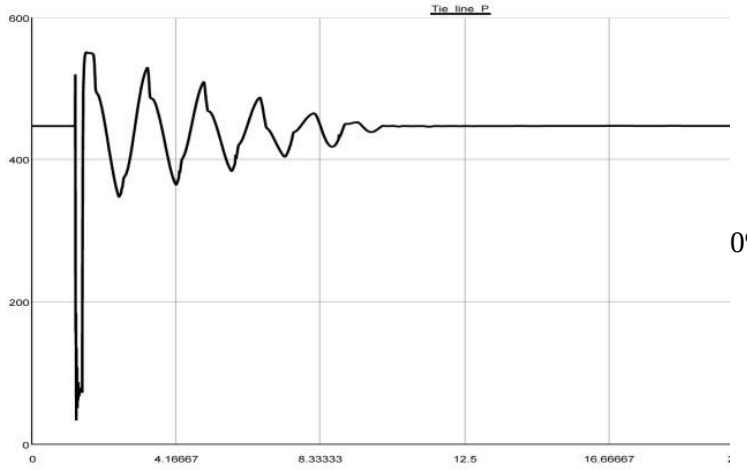
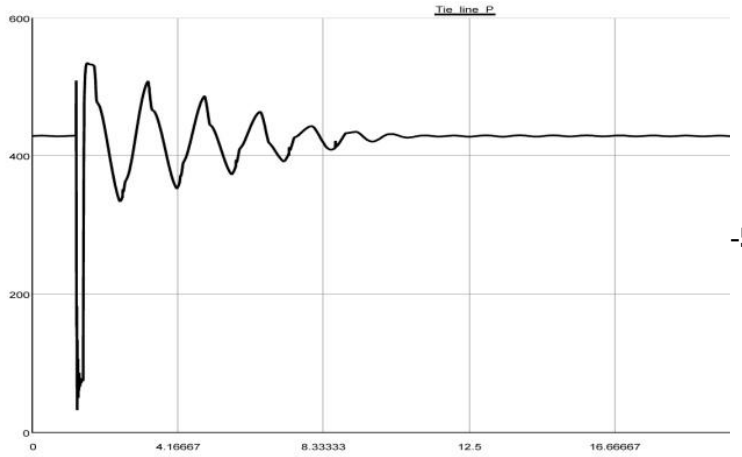
Case #	System response (Tie-line Power Flow)	Load Inc / Dec
7		-10%
8		-15%
9		-20%

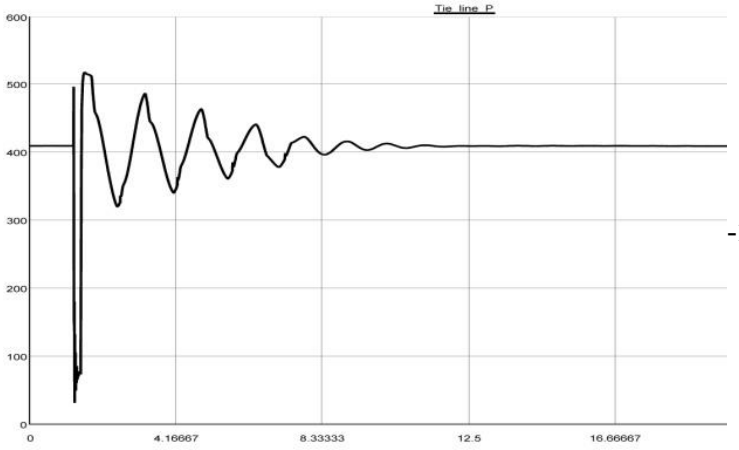
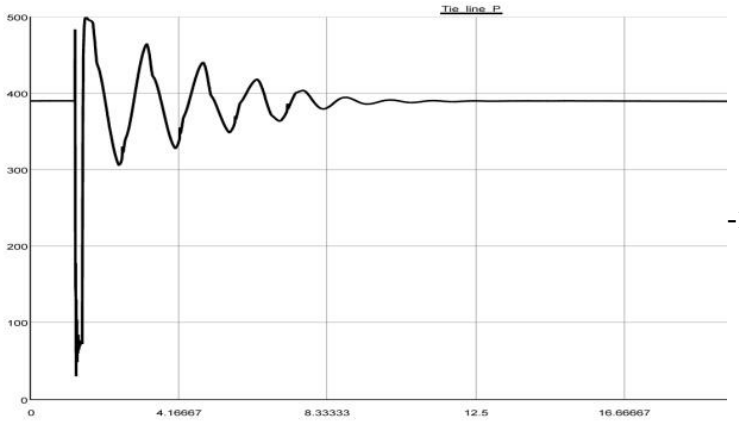
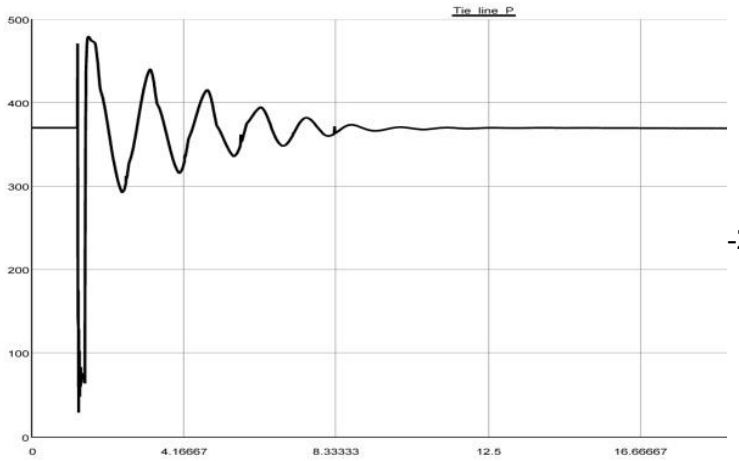
As shown above, the performance of the controller deteriorates as the load grows up. It becomes the worst at 20% increment of the base load (settled in more than 25sec) while the best at -20% (settled in about 10sec). Hence, our controller should be intelligent enough to have an adaptation mechanism to adapt itself according to the tie-line loading conditions. Therefore, the proposed gain adaptation mechanism, presented in chapter 4, has been added to the Fuzzy

controller to resolve this problem. The Table5.3 below shows the results of the introduced adaptation mechanism.

Table5.3 The Effect of adding Adaptive Controller

Case #	System response (Tie-line Power Flow)	Load Inc / Dec
1		+20%
2		+15%
3		+10%

Case #	System response (Tie-line Power Flow)	Load Inc / Dec
4		+5%
5		0%
6		-5%

Case #	System response (Tie-line Power Flow)	Load Inc / Dec
7		-10%
8		-15%
9		-20%

As can be seen above the performance becomes much better than the previous controller as the gain values are changing according to the tie-line

loading conditions. Table 5.4 shows the enhancement in setting time per each case with and without the presence of adaptation mechanism.

Table 5.4 Settling Time With and Without The Presence of Adaptation Mechanism

Type	load								
	-20%	-15%	-10%	-5%	0%	5%	10%	15%	20%
Without Adaptation	10sec	11sec	12sec	13sec	16sec	20sec	>25sec	>25sec	>25sec
With Adaptation	10sec	10sec	11sec	11sec	11sec	12sec	13sec	14sec	16sec
Enhancement	0%	10%	9%	18%	45%	66%	>92%	>78%	>56%

Scenario 4: 200 ms 3 Phase Fault with All PSS & SVC & Damping Controller

Figures 5.24&5.25 show a comparison of the tie-line flowing powers and tie-line midpoint voltages response when:

1. With only Power System Stabilizer is only there.
2. With Power System Stabilizer and SVC.
3. With Power System Stabilizer, SVC and Damping Controller.

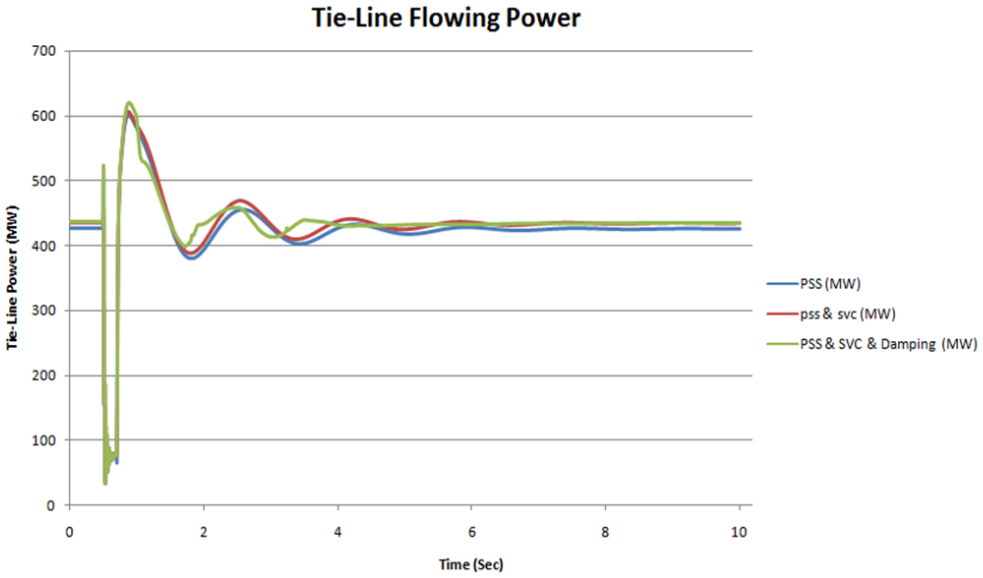


Figure5.24 Damping Controller Performance Compared To Other Scenarios (Power)

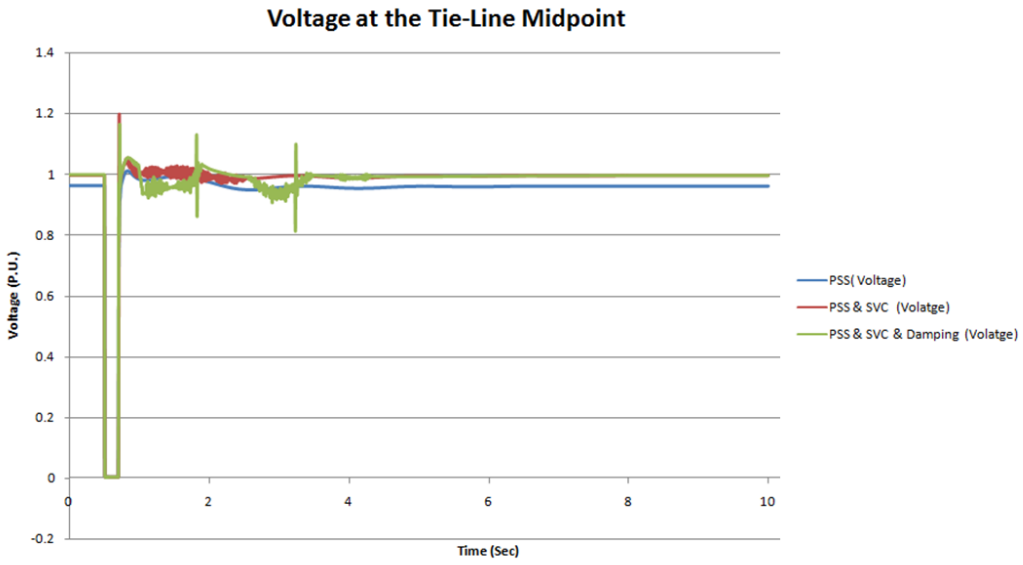


Figure 5.25 Damping Controller Performance Compared To Other Scenarios (Voltages)

As can be seen above, adding the SVC damping controller helps in damping the oscillation in the tie-line system 4 sec faster than both PSS only and PSS with SVC which both damp in 8 sec(without damping controller).

The Effect of increasing the SVC size from 100MVAr to 200 MVAr

Increasing the SVC size from 100MVAR to 200MVAR, improve the controller damping performance as shown Figure5.26to be damped 2 to 3 sec faster.

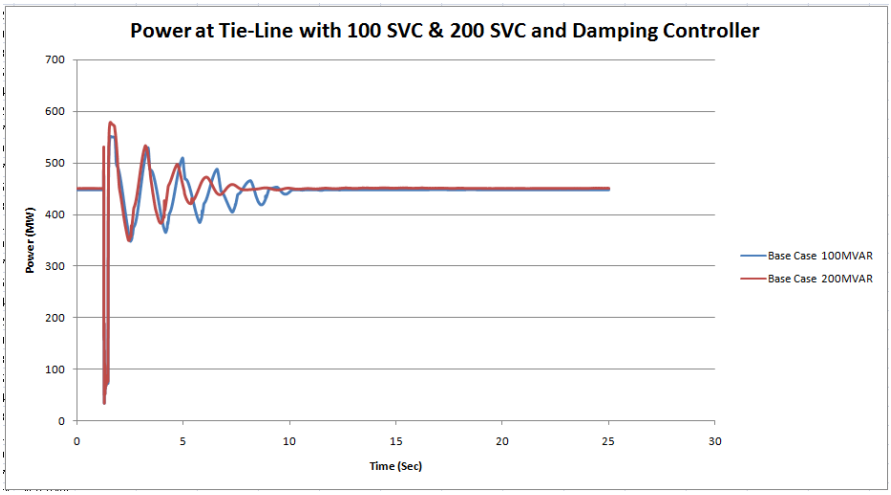
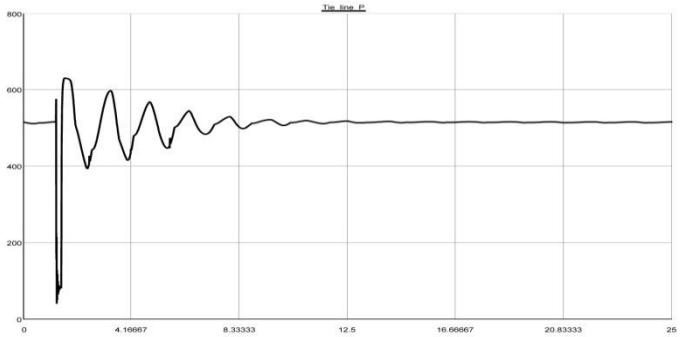
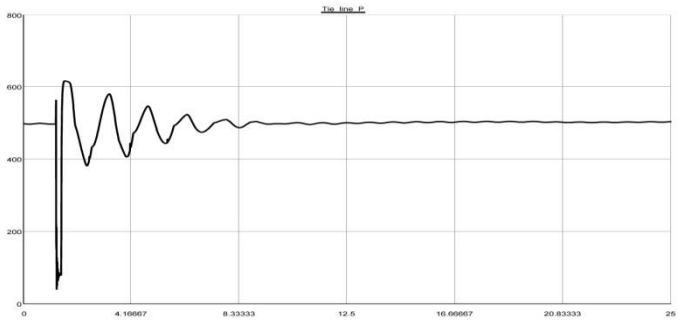
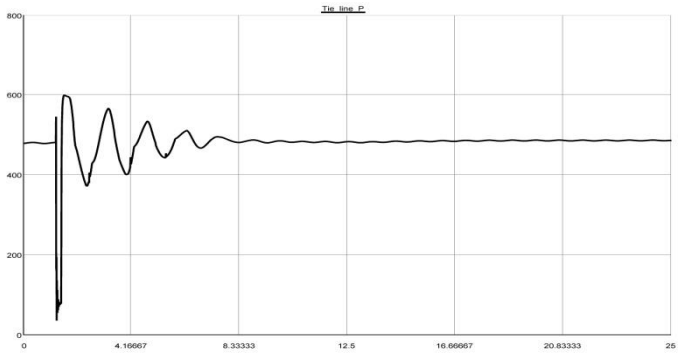
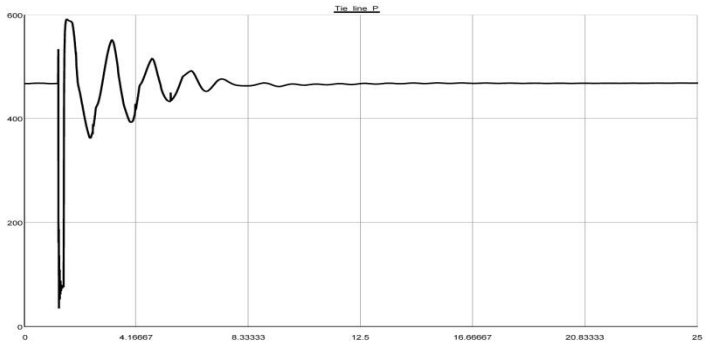
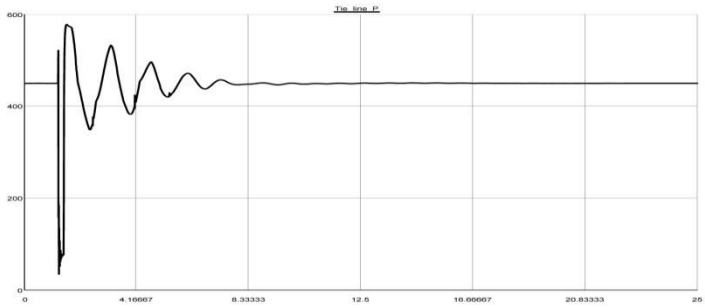
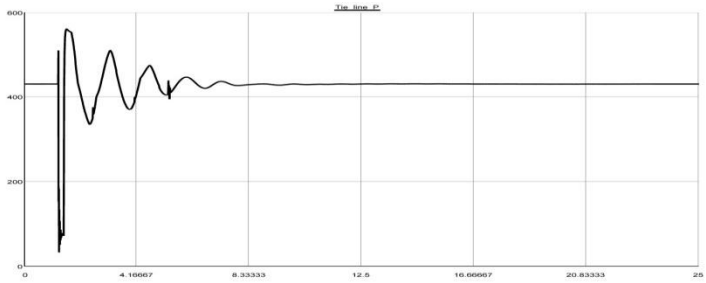
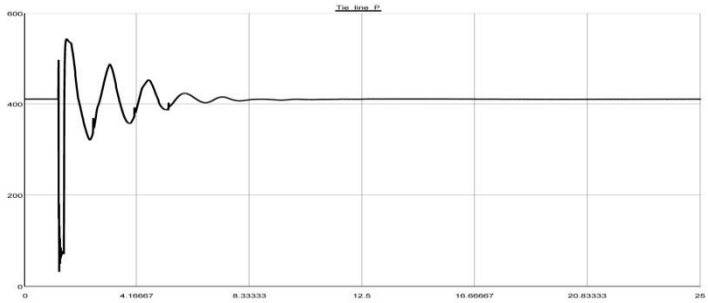
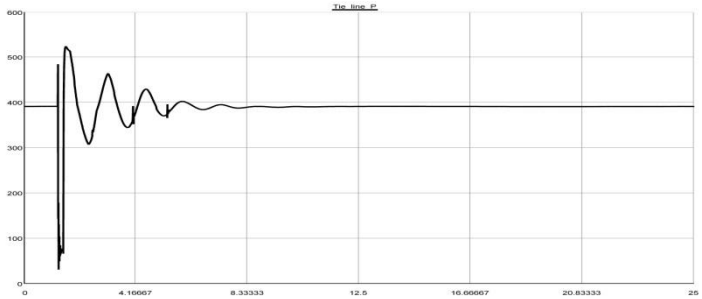
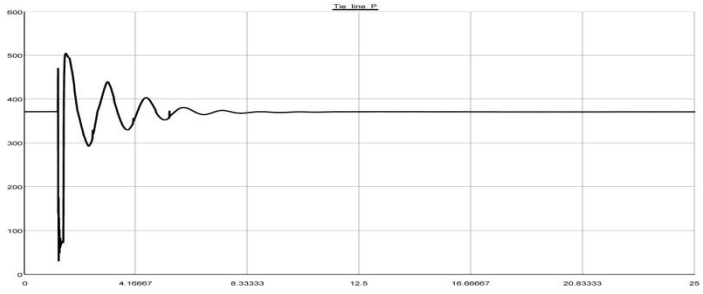


Figure5.26 The effect of increasing the SVC capacity from 100 MVAr to 200 MVAr

The results below in Table 5.4 show the effect of the new developed adaptive mechanism with different loading conditions (200MVArSVC).

Table5.5 Power at the Tie-line with different loading scenarios (200MVAr SVC)

Case #	System response (Tie-line Power Flow)	Load Inc / Dec
1		+20%
2		+15%
3		+10%
4		+5%

Case #	System response (Tie-line Power Flow)	Load Inc / Dec
5		0%
6		-5%
7		-10%
8		-15%
9		-20%

As it can be seen above, the settling time average is about 8 sec compared to 10 sec for100MVAr SVC. This issue implies that as the SVC capacity goes up as the performance become better, but it is found that the damping controller output gain has to be Variable as the SVC capacity becomes more than 200MVAr which introduce another complexity to the controller would be addressed in the future work.

Table 5.6 Settling Time With100MVAr & 200MVAr SVC

SVC Capacity	load								
	-20%	-15%	-10%	-5%	0%	5%	10%	15%	20%
100 MVAr	10sec	10sec	11sec	11sec	11sec	12sec	13sec	14sec	16sec
200 MVAr	7	7	7	7	7	8	9	10	12
Enhancement	42%	42%	57%	57%	57%	50%	44%	40%	33%

Figure5.27 shows a comparison of the power at the tie-line with the presence of SVC capacities (100MVAr and 200MVAr) along with all PSS.

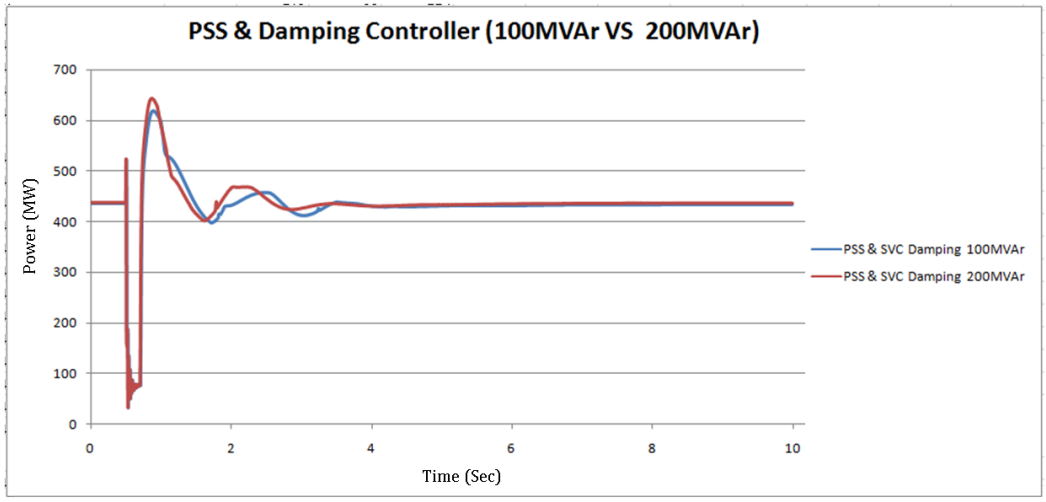


Figure5.27 The effect of using 100MVAr and 200MVAr on Power flowing through the tie-line during the Three Phase Short Circuit

As can be seen above the performance when adding the 200MVAr SVC damps in 3.5sec compared to 4.5sec for 100MVAr SVC.

Chapter 6. FLC Hardware Implementation

In this chapter, the fuzzy controller platform preparation, Fuzzy Controller Design Helper Tool (FCDHT), and the implementation results will be presented.

6.1. FLC Platform Preparation

In order to implement the fuzzy logic damping controller in real-time, an appropriate platform has to prepared / design for the sake of being able to take the input signals from the RTDS & and then send back the output signal to the RTDS. This can be achieved by:

1- Choosing a proper microcontroller
2- Choosing proper RTDS input & output signals within the microcontroller & RTDS limits.

The platform contains a microcontroller as a core & other interface peripherals. Generally, both RTDS and microcontroller can handle inputs & outputs in either analog or digital format. The main difference between them is the voltages ranges. RTDS analog input/ output voltage range is ($\pm 10V$) while, the microcontroller is either 0-5V, or 0-3.3V depending on its core type. As the real-time implementation is one of our targets, analog input/ output format has been chosen. To overcome the issue of difference in input / output voltages ranges, interface peripherals modules have been designed. Their main functions are to change the RTDS output from bipolar voltage to unipolar voltage, and to change the microcontroller output from unipolar to bipolar voltage.Figure6.1 shows the main blocks of the FLC platform.

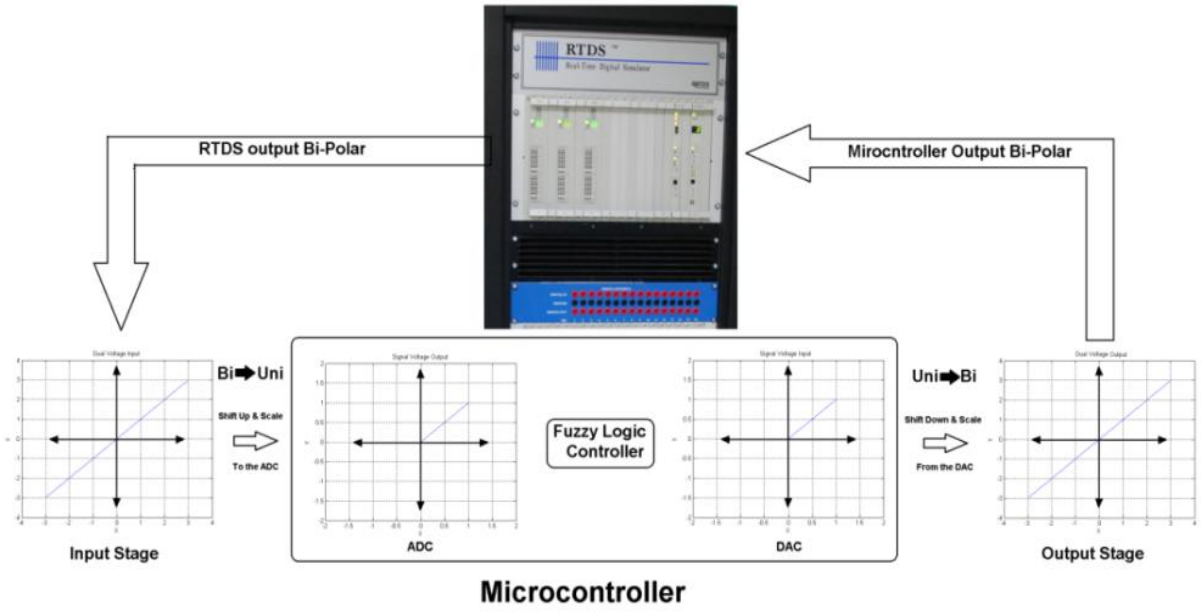


Figure6.1 The FLC Platform

6.1.1. Microcontroller Choosing

Since the FLC is going to be realized and implemented as Hardware In the Loop, we have to choose a microcontroller that is capable to handle the FLC algorithm computations in real-time. Based on that three categories of microcontrollers8-bit microcontroller, 32-bit microcontroller, and 32-bit microcontroller with Floating Point Unit (FPU) have been tested, which are PIC18f45K22, PIC32FMX460F512L, andSTM32F407 respectively. A performance test, by running the core code, has been conducted in order to find out the capable one. The Table 6.1 below shows the results as well as some of the microcontroller features.

Table 6.1 Microcontrollers Features & Comparison

Criteria	**PIC 18F45k22**	**PIC32MX450F512**	**STM32F407**
Speed	16 MIPS	80 MIPS	168 MIPS
FPU	No	No	Yes
ADC	10 bit	10 bit	12 bit
DAC	-	-	12 bit
Core Code execution Speed	1 KHz	10 KHz	80 KHz

Note: The microcontroller used in this thesis is STM32F407.

6.1.2. Interface Peripherals (Uni-Polar To Bi-Polar & Bi-Polar To Uni-Polar)

In order to connect RTDS to the microcontroller two modules have been designed, built and calibrated, and they are Bi-Polar to Uni-Polar Voltage module and Uni-Polar to Bi-Polar Voltage module. More details will be presented in the following paragraphs.

Bi-Polar to Uni-Polar Voltage module:

The main objective of this module is to change the RTDS dual output voltage from ($\pm 10V$)(it will be controller through the RTDS model to be $\pm 3V$) to the maximum of 0-3V suiTable for the microcontroller.Figure6.2 shows the elestrative main objective of the module.

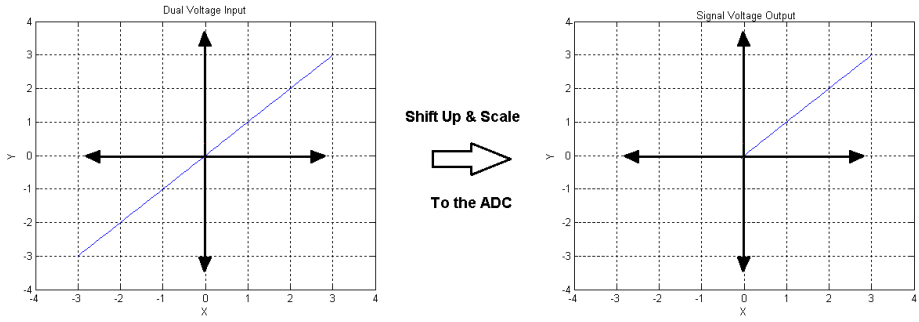


Figure6.2 Bipolar to Unipolar

Based on that non-inverted summation OP-AMP configuration can be used (shown in Figure6.3), and the following equation is used to calculate the resistors values.

$$V_o = (V_1 + V_2)\left(1 + \frac{R4}{R3}\right) = 2(V_1 + V_2) \qquad 6.1$$

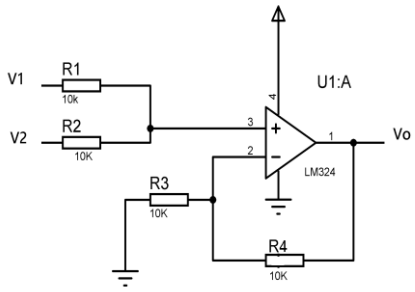


Figure6.3Core schematic

Note: full schematic & PCB are available in Appendix A.

The above circuit has been calibrated in order to achieve more accurate readings and the Figure6.4 shows the calibration curve along with the correlation equation

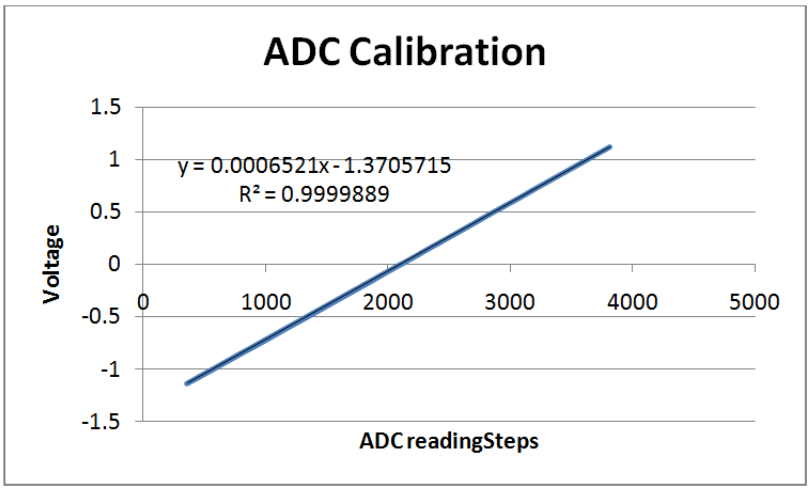


Figure6.4 Calibration graph & Equation

Uni-Polar to Bi-Polar Voltage module:

The main purpose of this module is to change the microcontroller output voltage from 0-3Vto the maximum of±3V to be adequate for the RTDS system. Figure6.5. shows raphlly the elestrative main objective of the module.

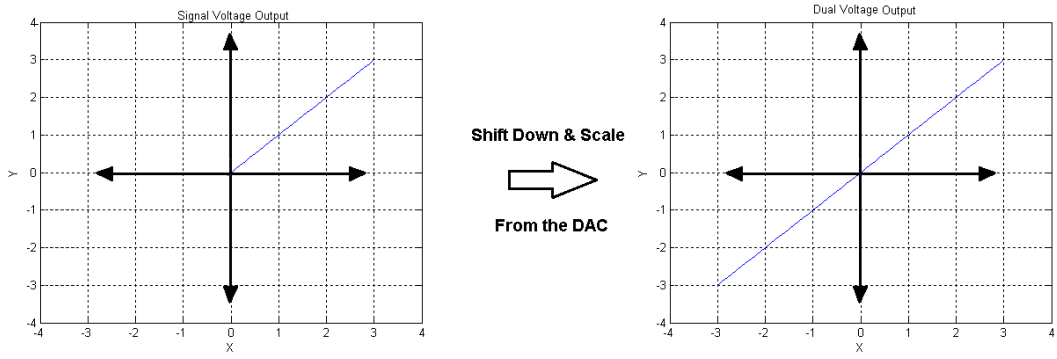


Figure6.5 Unipolar to Bipolar

In order to achieve that inverted & non-inverted summation difference OP-AMP configuration can be used. The following equation 6.2 is used for identifying the required resistors.

$$V_o = \left(1 + \frac{R3}{R4} + \frac{R1}{R4}\right) V1 - \frac{R3}{R4} * 2.5V \quad 6.2$$

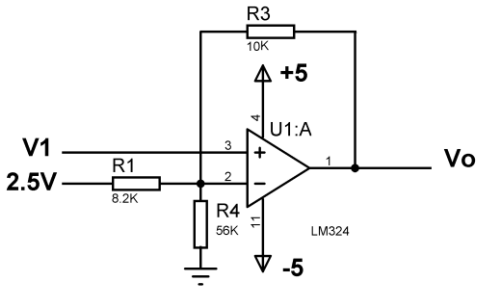


Figure6.6 Core schematic

Note: full schematic & PCB are available in Appendix A.

Above circuit, in Figure6.6, has been calibrated in order to achieve more accurate readings andFigure6.7 shows the calibration curve along with the correlation equation.Figure6.8shows the entire platform

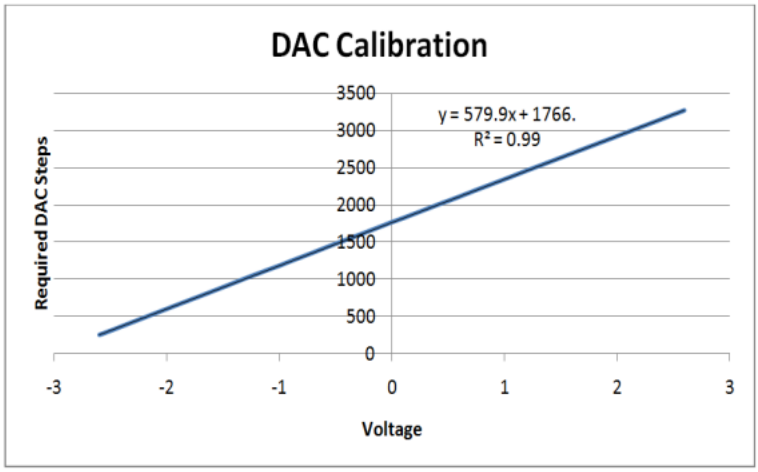


Figure 6.7 Calibration graph & Equation

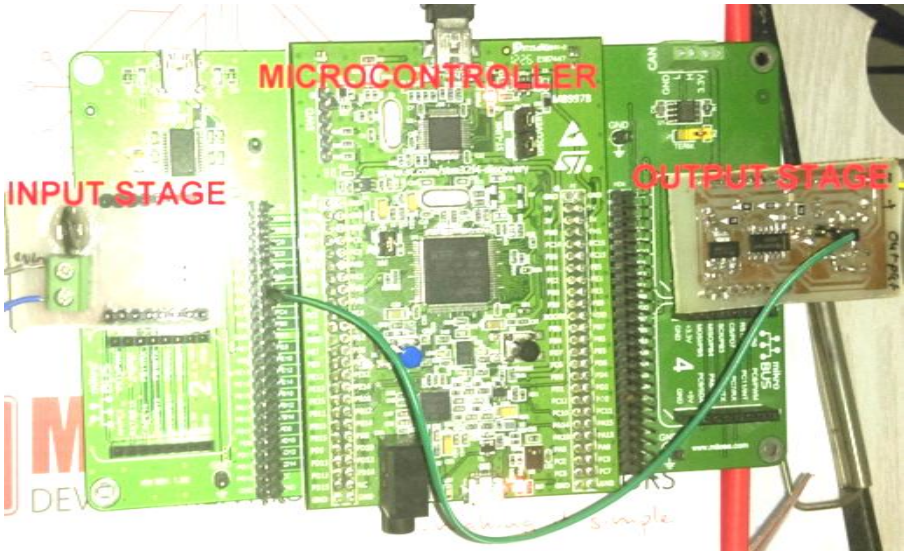


Figure6.8 FLC Platform

6.1.3. Data Pre-Processing (Moving Average Filter)

Both read data , Speed Deviation and Tie-Line power, are passed through moving average filter, which is considered as special type of Finite Impulse Response (FIR) filter. The number of chosen points are 31 for speed deviation and 100 for tie-line power.

6.1.4. Microcontroller programming

As mentioned above, the 32- bit microcontroller used in this thesis is STM32F407 microcontroller, and the compiler used is the Mikro-C produced by Mikroelektronika Company. The developed Fuzzy Controller Design Helper Tool, explained in the next section, has been used to generate the microcontroller core code and Figure6.9shows the microcontroller's program flow chart.

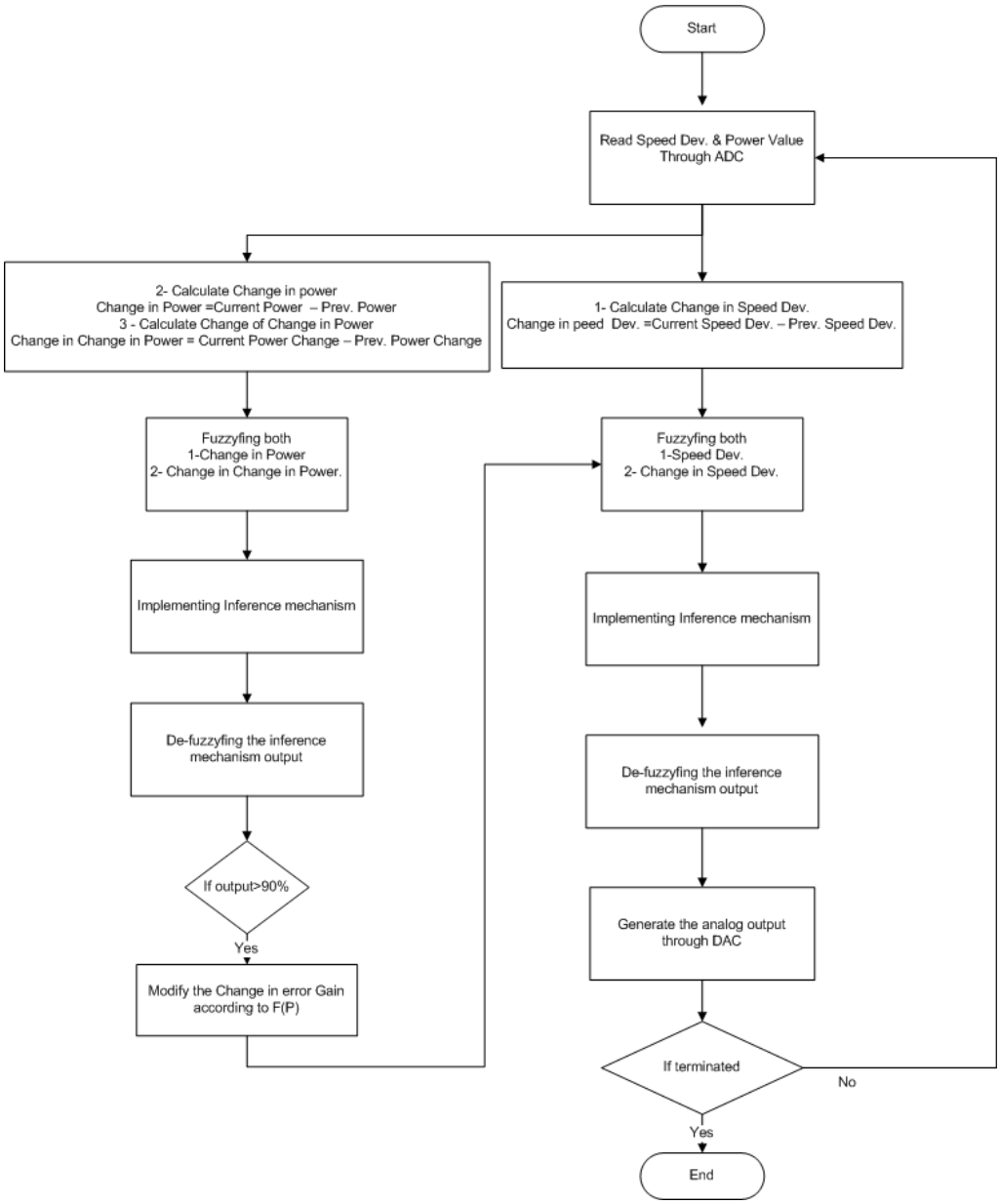


Figure6.9 Microcontroller Program Flow Chart

6.2. **Fuzzy Controller Design Helper Tool (FCDHT)**

This tool has mainly been developed for the sake of helping in designing Fuzzy logic controller. It has been designed for making a controller that has two inputs and one

output with triangular membership functions. The used Rule Base is 7x7.The tool has been developed because of many perspectives such as controller developing side and microcontroller programming side.

6.2.1. Controller Developing Side

The FLC, described in Chapter 4, consists of fuzzification stage, rule base & inference mechanism stage, and de-fuzzification stage, each one those stages has many Variables that may cause minor or major change in the FLC programming (especially for the microcontroller) . For instance, fuzzification & de-fuzzification stages have many Variables within the membership functions (the centers, amplitudes and widths) that are needed to be chosen carefully. Furthermore, the rule base matrix which is the heart of the controller is needed to be chosen properly. In our case we have 21 membership Variables and functions in addition to 49 rule bases. Based on above, there is a need for a tool that helps in developing the controller model in MATLAB - SIMULINK & M-File, RTDS, and Microcontroller.

In MATLAB

The Developed tool helps in:

1. Giving the opportunity of choosing the proper values of the membership function limits, amplitude as well as the rule base with a possibility of observing the direct effect.
2. Generating MATLAB Fuzzy Inference System file (FIS) that can used directly in fuzzy model within SIMULINK environment. Currently, there is a toolbox in the MATLAB called Fuzzy Logic that helps in developing and designing the controller, but it's main drawback is the manner of entering the rule base, as it has

to be entered one by one without having a facility to enter all of them as Table and this issue could lead to :

I. Increase the possibility of making mistakes

II. Consume a lot of time.

In this tool, you will have a simple 7x7 Table for the rule base where it can be modified easily with little mistakes and little time consumption. Table6.2 below shows a sample of Table used.

Table6.2 sample of rule base Table

		Rule Base 3						
		Err						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Derr	E_BN	BN	BN	BN	MN	MN	SN	Z
	E_MN	BN	BN	MN	MN	SN	Z	SP
	E_SN	BN	MN	MN	SN	Z	SP	MP
	E_Z	MN	MN	SN	Z	SP	MP	MP
	E_SP	MN	SN	Z	SP	MP	MP	LP
	E_MP	SN	Z	SP	MP	MP	LP	LP
	E_BP	Z	SP	MP	MP	LP	LP	LP

3. Generate a MATLAB M-file for online Adaptive Fuzzy controller based on Model Reference Learning Method, as it is hard to be implemented through MATLAB fuzzy logic toolbox

In RTDS

In RTDS, there is no such facility as FIS Toolbox in the MATLAB for making the controller. Therefore, in order to build a fuzzy model inside the RTDS, Component Builder facility has to be used, it mainly has two parts the graphical part and the coding

part. As we stated above, FLC has a huge probability of changing the code as it has many Variables and many rules, based on that there is need for a tool that helps in generating the code that is needed for RTDS Fuzzy Controller Model.

6.2.2. Microcontroller Programming Side

Since the FLC will be realized in microcontroller, there is a need for a tool that can convert the fuzzy logic controller stages into a code that the microcontroller can understand. It could be true that it can be programmed once, but as we are in the developing stage of the microcontroller based FLC, there is a high probability of frequent code changing that would consume a lot of time and effort .Based on above there is a need for a tool that can generate a Mikro-C code for the microcontroller.

6.2.3. Main Structure

The tool is Microsoft Excel based that has been programmed using VBA code. Figure6.10 below shows the main structure

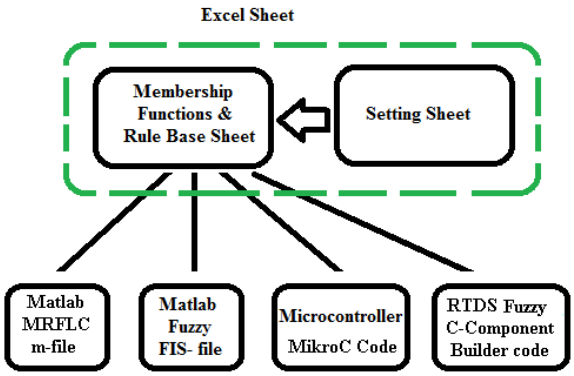


Figure6.10 Main FCDHT Tool Structure

All of the functions shown in Figure6.10(MATLAB MRFLC m-file, MATLAB Fuzzy FIS file, Microcontroller Mikeo-C code, RTDS Fuzzy C- Component Builder Code) are using the following flow chart Figure6.11.

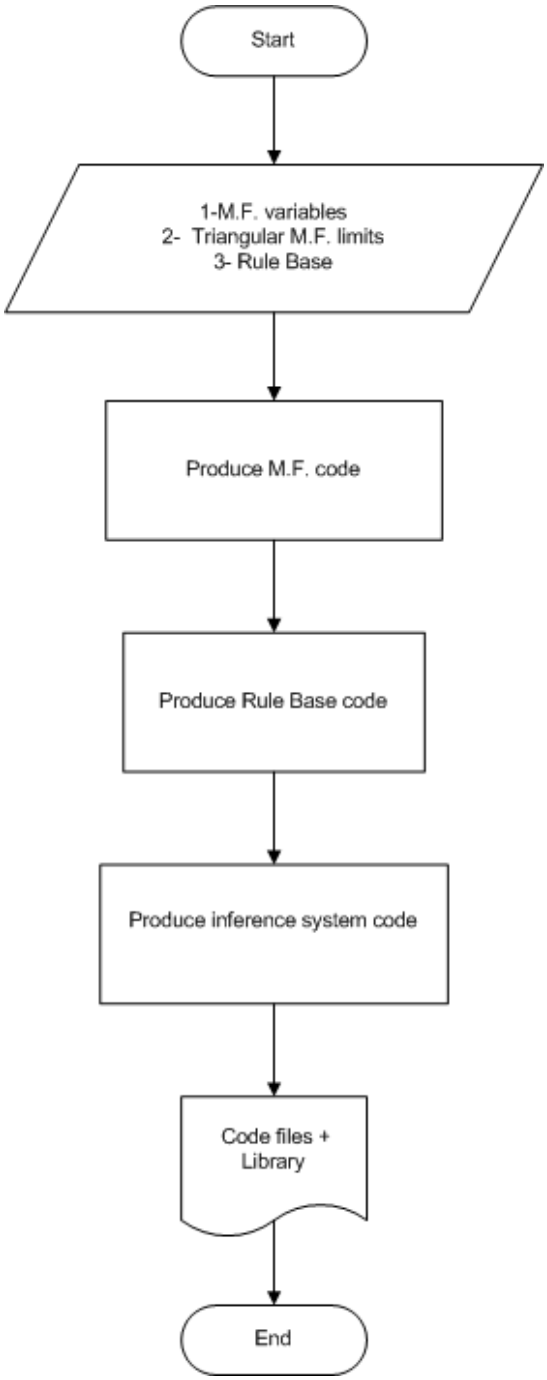


Figure6.11 FCDHT Flow Chart

Figures 6.12&6.13shows the settings & Fuzzy 7x7 sheets in the FCDHT. In Figure6.12 the inputs & output Variables are places, while inFigure6.13 the membership function

limits & height are placed in addition to the rule base. .

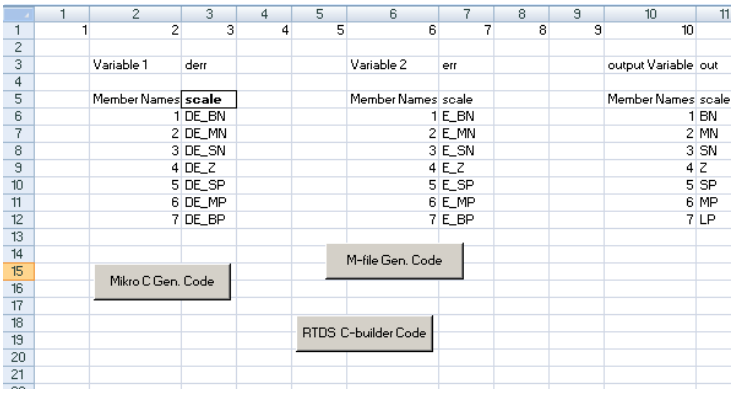


Figure6.12 Setting Sheet

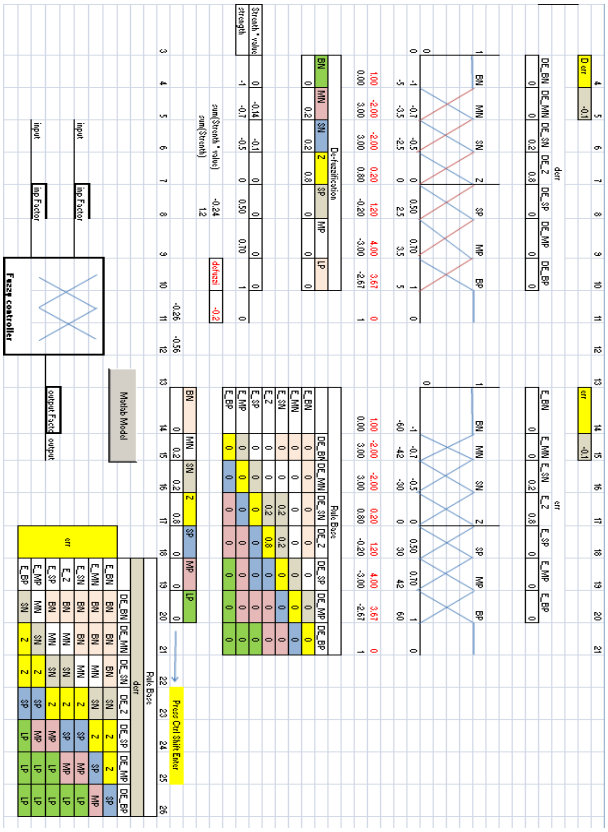


Figure6.13 Membership Functions and Rule Base Sheet

6.3. **Implementation Results:**

After connecting the FLC platform with the RTDS and preparing the experimental setup as shown in Figure6.14, three phase disturbance (200 ms, self cleared) at the midpoint of the tie-line has been simulated.

Figure6.14 Experimental Setup

Figures 6.15to 6.25show the results of the following scenarios:

1- Without the presence of any PSS and with SVC capacities 100MVAr& 200MVAr.

2- With the presence of all PSS, all PSS & SVC, and all PSS & SVC & Damping Controller(100MVAr& 200MVAr).

6.3.1. Without the presence of any PSS

a) (100MVAr SVC)

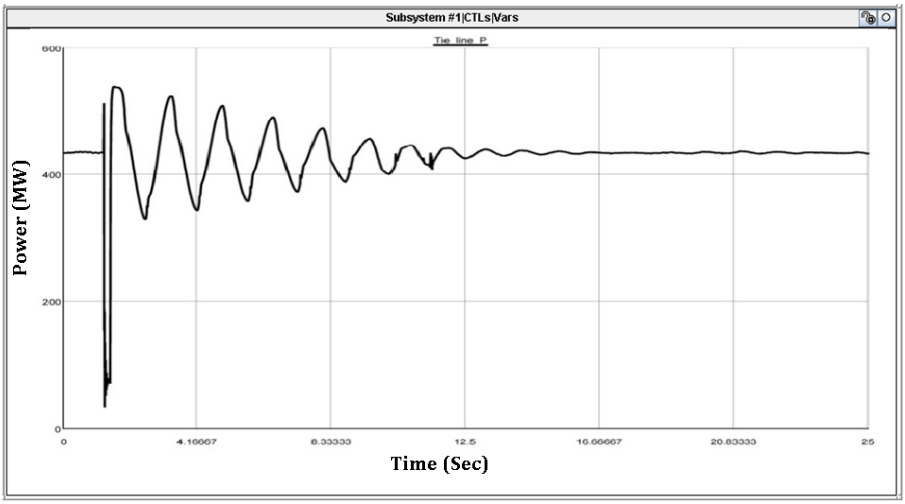


Figure6.15 Power at the Tie-Line with 100MVAr

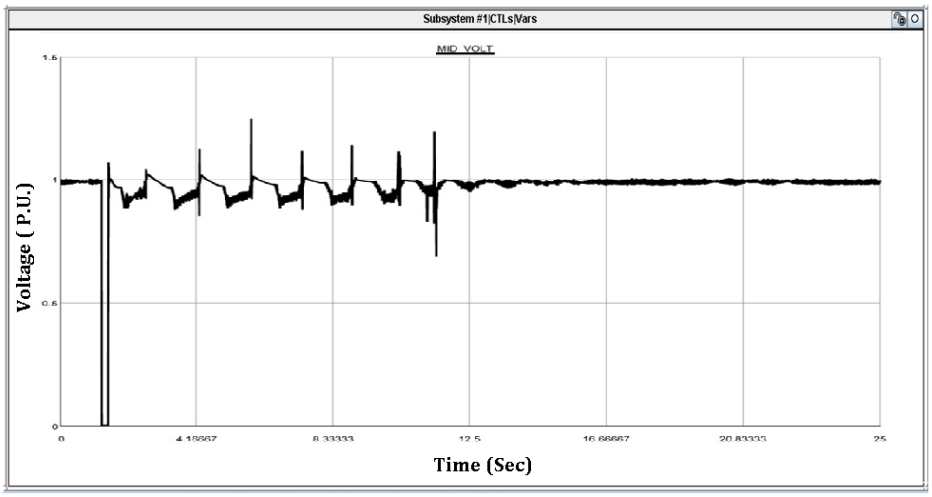


Figure6.16 Voltage at the Midpoint of the Tie-Line (100MVAr)

b) (200MVAr SVC)

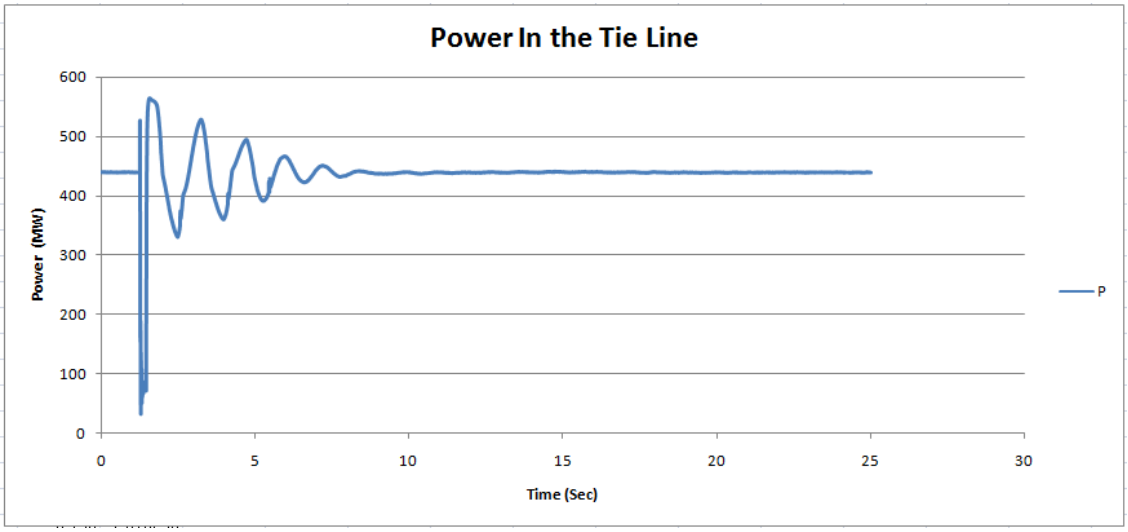


Figure6.17 Power at the Tie-Line with 200MVAr

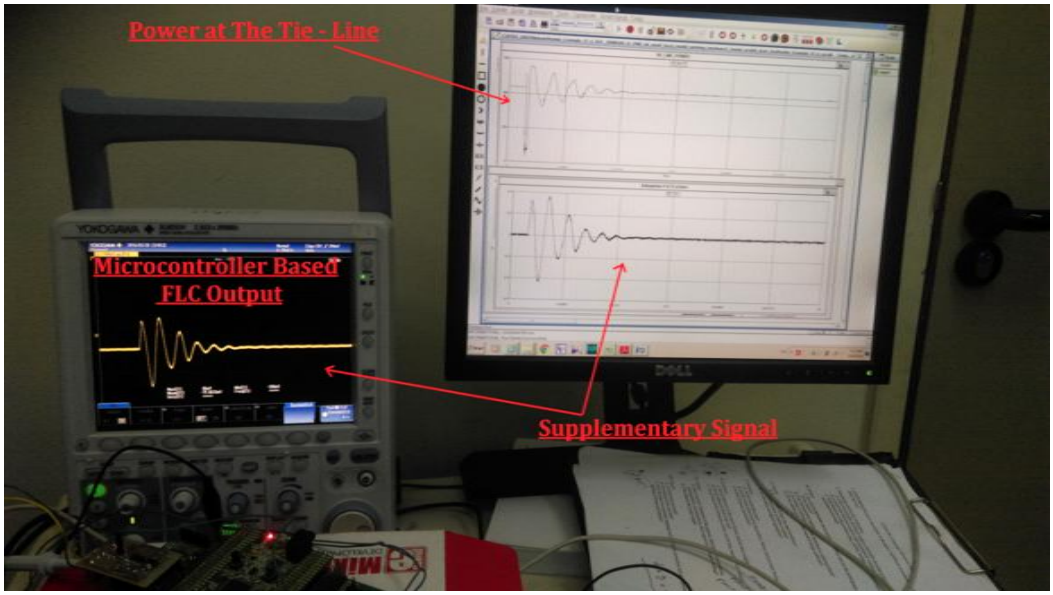


Figure6.18 Power at the Tie-Line & the Supplementary Signal

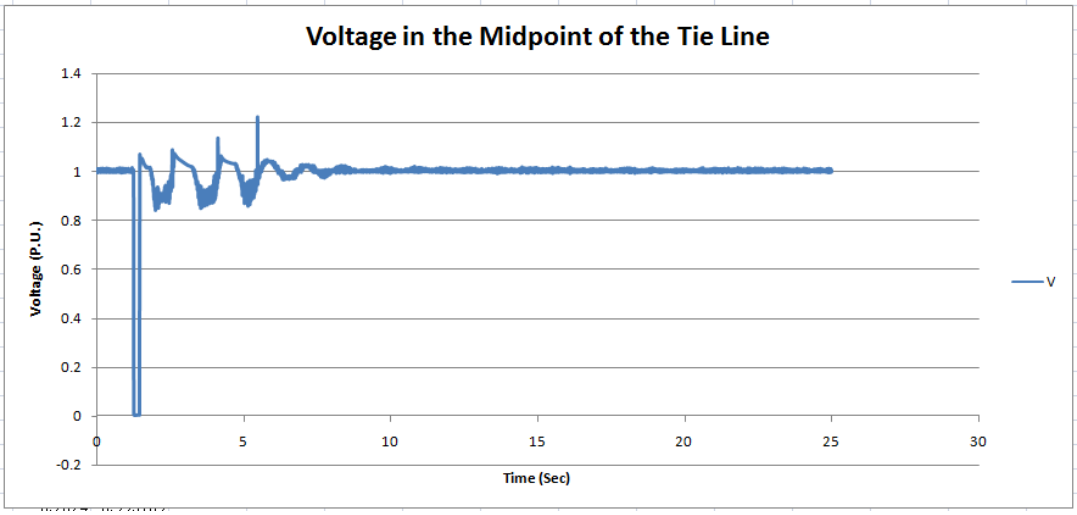


Figure6.19 Voltage at the Midpoint of the Tie-Line (200MVAr)

As can be seen in Figures 6.16to6.20, the system becomes stable in about 16sec with 100MVAr SVC and in less than 10 sec with 200 MVAr SVC. It also can be noticed that there are some distortion in both voltages and power due to noise introduce by ADC /DAC quantization error as well as the other electro-magnetic interference around the realized FLC.

6.3.2. With The Presence Of All PSS, All PSS &SVC, And All PSS &SVC &Damping Controller

A) (100MVArSVC)

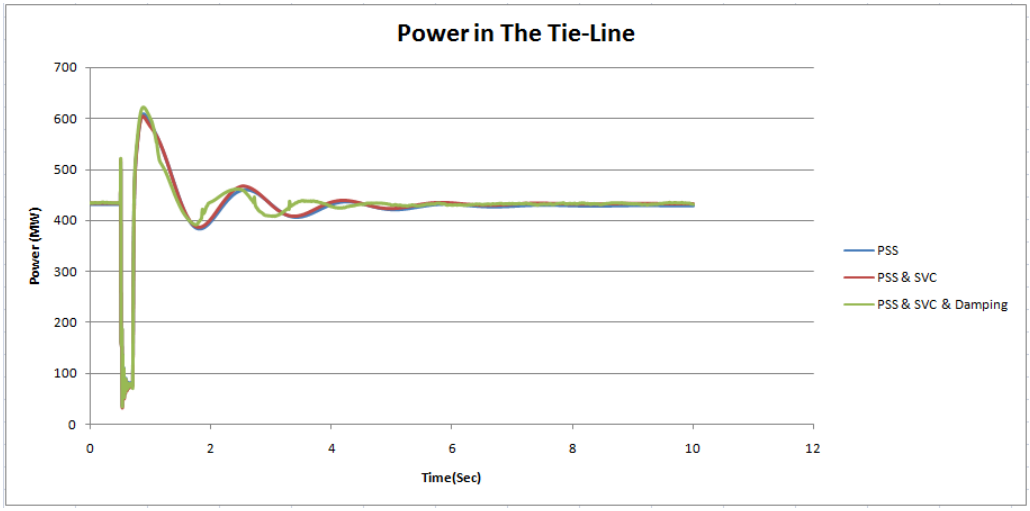


Figure6.20 Comparing the power at Tie-Line with presence of All PSS, All PSS & SVC, and All PSS & SVC & Damping Controller(100MVAr SVC)

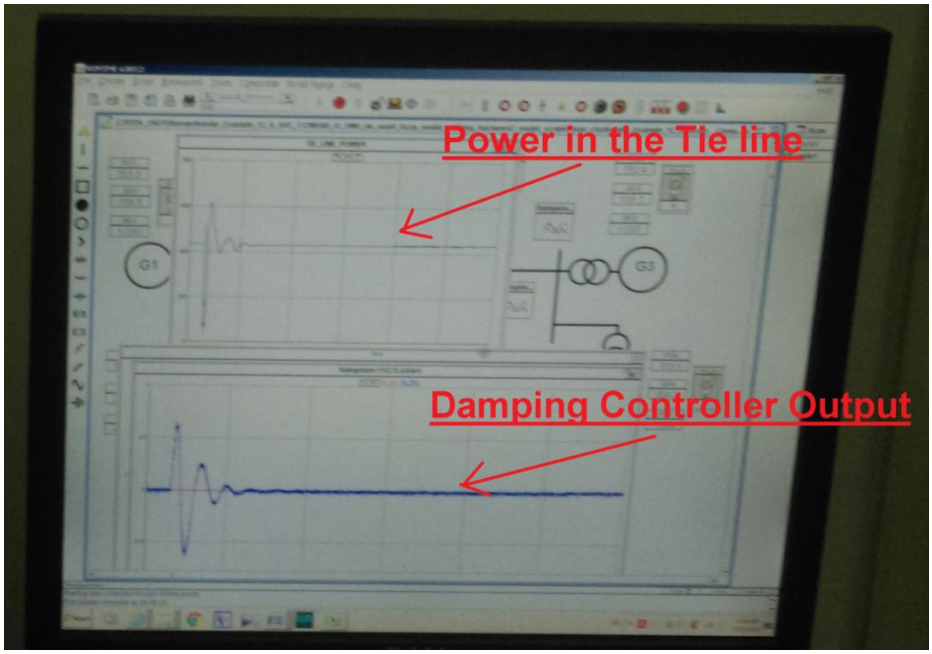


Figure6.21 Power at the Tie-Line & the Supplementary Signal

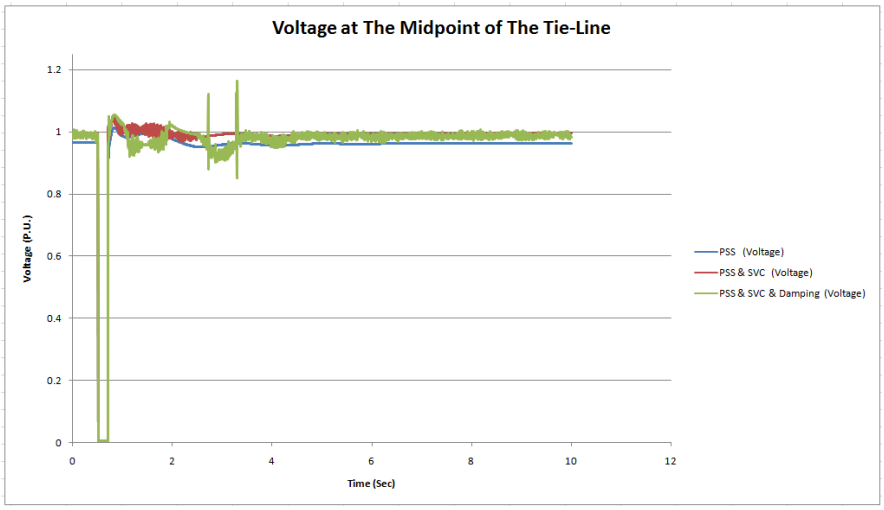


Figure6.22 Comparing the voltage at the midpoint of the Tie-Line with presence of All PSS, All PSS & SVC, and All PSS & SVC & Damping Controller(100MVAr SVC)

B) (200MVArSVC)

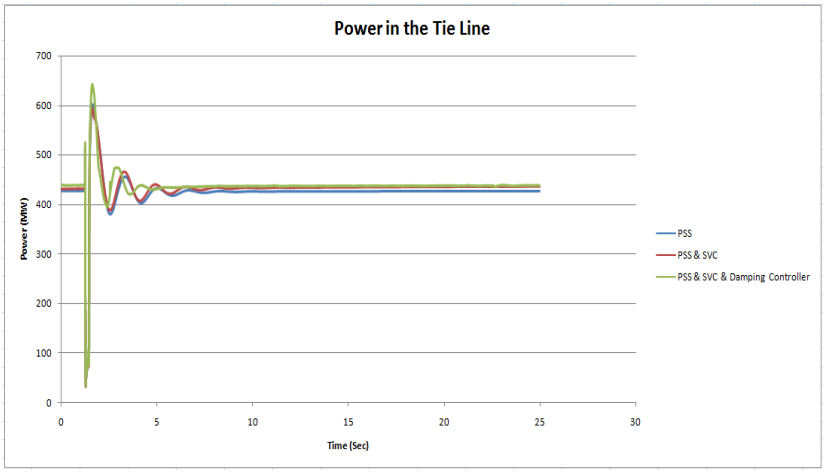


Figure6.23 Comparing the power at Tie-Line with presence of All PSS, All PSS & SVC, and All PSS & SVC & Damping Controller (200MVAr SVC)

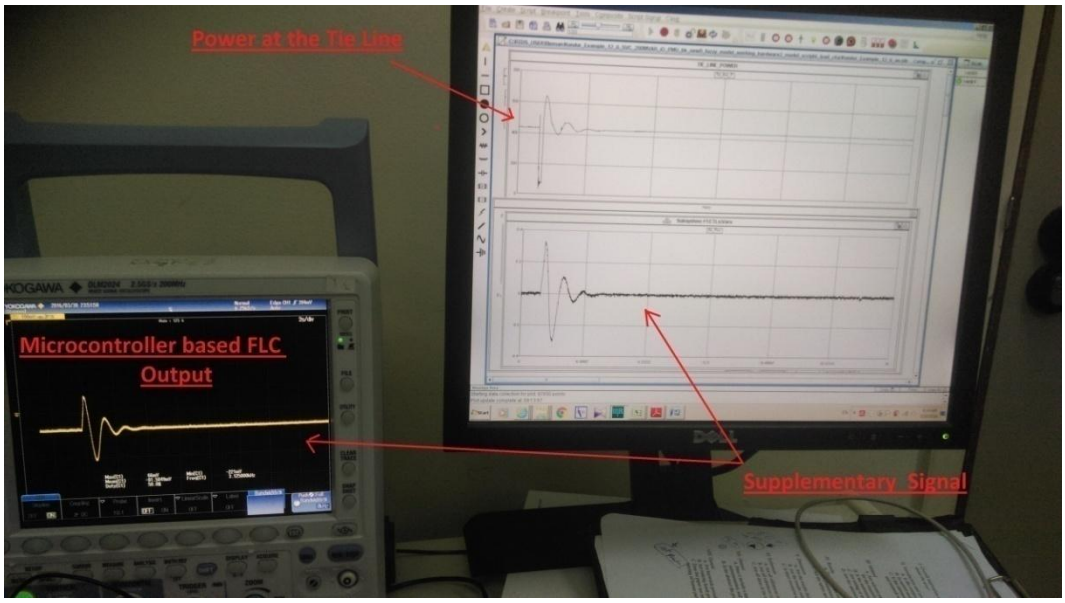


Figure6.24 Power at the-Tie Line & the Supplementary Signal

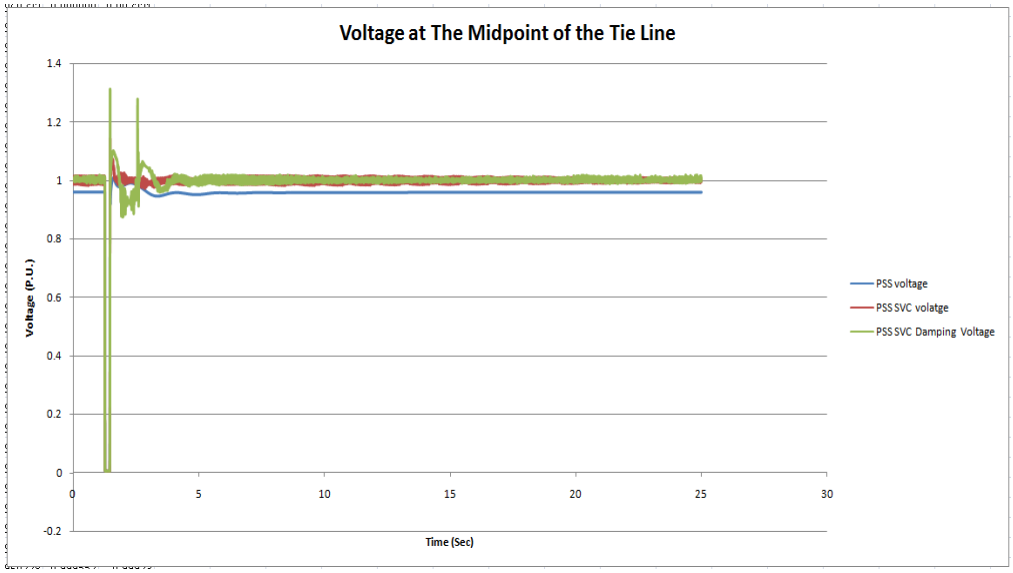


Figure6.25 Comparing the voltage at the midpoint of the Tie-Line with presence of PSS, PSS & SVC, and PSS & SVC & Damping Controller (100MVAr SVC)

As shown above in Figures 6.21 and 6.26, for 100MVAr the damping controller reduces the time needed for eliminating the inter-area oscillation by about one second, while for 200MVAr is about 4 sec.

Chapter 7. Conclusion& Future Work

7.1. **Conclusion**

In this thesis, the designs of SVC fuzzy controllers to enhance the damping of inter-area oscillation in power systems are presented. The design starts with fuzzy logic controller for a range of operating points, as this is one of the main FLC advantage over the conventional controller, and then this range has been extended by adding an adaptation mechanism to the main controller for adapting the controller gains based on the tie-line loading conditions. The range of robust operation point has been extended to 20% more than the original operation point, hence it becomes more robust.

Fifteen rule bases have been tested, and what could be considered as the best has been chosen. The same thing has been implemented for finding membership functions' centers.

Using tuning steps, presented in chapter 4, for the input-output scaling factors helps in finding a reasonable controller performance.

RTDS has been used for modeling, simulation & implementation, as the RTDS is mainly designed for simulating and testing the power systems. Hence, it has more detailed models than the MATLAB has. RTDS still has lacks some features compared to MATLAB such as the fuzzy toolbox and other toolboxes in addition to the FLC model, which could be considered as one of the motivations of making Fuzzy Controller Design Helper Tool. Fuzzy Controller Design Helper

Tool helps in many fields such as making the RTDS Fuzzy controller model, generating Mikro-C code for the microcontrollers, making FIS file for the MATLAB fuzzy model, as well as generating MATLAB m-code for adaptive fuzzy controller based on Fuzzy Model Reference Learning Method.

Adding Damping controller to SVC helps in damping the inter-area oscillation in the tie-line even without the presence of any PSS, and also it helps in damping the oscillation faster with the presence of PSS.

The adaptation mechanism helps in increasing the operation range to cover up to 20% more the original operation point (base case).

Increasing the SVC capacity from 100MVAr to 200MVAr helps in damping the oscillation faster, which implies that as the SVC capacity goes up as the performance becomes better, but this issue is valid up to certain limit.

PMU Signals measurement can be used main as a main controller input source or as an alternative to the Generation measured speed deviation, hence, input selection mechanism can be used to chose the proper one (in case of the absence of the other one).

Using PMU adds a possibility of using a combination of signals from multiple sources, hence it widening the observable area, and improves the stability.

Hardware In the Loop test has been conducted using microcontroller based fuzzy logic controller, and the results was fairly similar to the simulated one

except with some distortion introduced by the ADC & DAC quantization error as well as the electro-magnetic interference.

As a benefit of this study, it may be extended to be used in GCC interconnected network in order to improve the power transfer capability between the GCC countries.

7.2. **Future Work:**

- Using Tuning technique such as Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and other techniques in order to auto tune the fuzzy logic limits and gains
- Adaptive methods, such as ANFIS or FMRLC technique can be used to make adaptive controller in order to enhance the performance.
- Address the issue performance enhancement by related SVC sizing with Variable output controller gain. Using Variable output gain can enhance the controller performance.
- Address the delay issue that is associated with WAMS & PMU data communication which could range from 100 ms to 700 ms depending on the type of communication used.
- Addressing the WAMS communication failure issue. Damping Controller's Source Selection Strategy between the Local Measured Value & PMU measured values can be used
- Testing the designed Fuzzy logic damping controller on STATCOM.
- Implementing the designed Fuzzy logic damping controller on the GCC interconnected systems.

References

[1] "Proposed Terms and Definitions for Power System Stability Task Force on Terms & Definitions System Dynamic Performance Subcommittee Power System Engineering Committee," *IEEE Power Engineering Review IEEE Power Eng. Rev.*, vol. PER-2, no. 7, pp. 28–28, 1982.

[2] "Definition and Classification of Power System Stability IEEE/CIGRE Joint Task Force on Stability Terms and Definitions," *IEEE Trans. Power Syst. IEEE Transactions on Power Systems*, vol. 19, no. 3, pp. 1387–1401, 2004.

[3] "Electricity," *Arab Fund for Economic and social development*. [Online]. Available at: http://www.arabfund.org/default.aspx?pageid=454. [Accessed: 03-Apr-2016].

[4] M. Klein, G. Rogers, and P. Kundur, "A fundamental study of inter-area oscillations in power systems," *IEEE Trans. Power Syst. IEEE Transactions on Power Systems*, vol. 6, no. 3, pp. 914–921, 1991.

[5] P. Kundur, N. J. Balu, and M. G. Lauby, Power system stability and control. New York: McGraw-Hill, 1994.

[6] J. Shea, "Understanding FACTS-concepts and technology of flexible AC transmission systems [Book Review]," IEEE Electrical Insulation Magazine IEEE Electr. Insul. Mag., vol. 18, no. 1, pp. 46–46, 2002.

[7] S. D. Oliveira, "Synchronizing and damping torque coefficients and power system steady-state stability as affected by static VAr compensators," IEEE Trans. Power Syst. IEEE Transactions on Power Systems, vol. 9, no. 1, pp. 109–119, 1994.

[8] G. Taranto and J. Chow, "A robust frequency domain optimization technique for tuning series compensation damping controllers," IEEE Trans. Power Syst. IEEE Transactions on Power Systems, vol. 10, no. 3, pp. 1219–1225, 1995.

[9] E. Acha, Power electronic control in electrical systems. Oxford: Newnes, 2002.

[10] "Static VAr compensator models for power flow and dynamic performance simulation," IEEE Trans. Power Syst. IEEE Transactions on Power Systems, vol. 9, no. 1, pp. 229–240,

1994.

[11] Modeling of SVC in Power System Studies, ABB Power Systems AB, information NR 500-026E April 1996.

[12] M. L. Wittrock, H. Johannsson, and R. Garcia-Valle, “An implementation and test platform for wide area stability assessment methods,” IEEE PES ISGT Europe 2013, 2013.

[13] F. Hu, K. Sun, A. D. Rosso, E. Farantatos, and N. Bhatt, “Measurement-Based Real-Time Voltage Stability Monitoring for Load Areas,” IEEE Trans. Power Syst. IEEE Transactions on Power Systems, pp. 1–12, 2015.

[14] A. G. Phadke and J. S. Thorp, Synchronized phasor measurements and their applications. New York: Springer, 2008.

[15] M. Klein, G. Rogers, and P. Kundur, “A fundamental study of inter-area oscillations in power systems,” *IEEE Trans. Power Syst. IEEE Transactions on Power Systems*, vol. 6, no. 3, pp. 914–921, 1991.

[16] G. Andersson, P. Donalek, R. Farmer, N. Hatziargyriou, I. Kamwa, P. Kundur, N. Martins, J. Paserba, P. Pourbeik, J. Sanchez-Gasca, R. Schulz, A. Stankovic, C. Taylor, and V. Vittal, “Causes of the 2003 Major Grid Blackouts in North America and Europe, and Recommended Means to Improve System Dynamic Performance,” *IEEE Trans. Power Syst. IEEE Transactions on Power Systems*, vol. 20, no. 4, pp. 1922–1928, 2005.

[17] H. Ghasemi, C. Canizares, and A. Moshref, “Oscillatory Stability Limit Prediction Using Stochastic Subspace Identification,” IEEE Trans. Power Syst. IEEE Transactions on Power Systems, vol. 21, no. 2, pp. 736–745, 2006.

[18] F. D. Marco, N. Martins, and J. C. R. Ferraz, “An automatic method for power system stabilizers phase compensation design,” *2013 IEEE Power & Energy Society General Meeting*, 2013.

[19] C.-C. Chu and H.-C. Tsai, “Application of Lyapunov-based adaptive neural network upfc damping controllers for transient stability enhancement,” *2008 IEEE Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century*, 2008.

[20] S. Shojaeian, J. Soltani, and G. A. Markadeh, “Damping of Low Frequency Oscillations of

Multi-Machine Multi-UPFC Power Systems, Based on Adaptive Input-Output Feedback Linearization Control," *IEEE Trans. Power Syst. IEEE Transactions on Power Systems*, vol. 27, no. 4, pp. 1831–1840, 2012

[21] R. Goldoost, Y. Mishra, and G. Ledwich, "Utilizing Wide-Area Signals for off-center SVCs to damp interarea oscillations," *2013 IEEE Power & Energy Society General Meeting*, 2013.

[22] S. P. Azad, R. Iravani, and J. E. Tate, "Damping Inter-Area Oscillations Based on a Model Predictive Control (MPC) HVDC Supplementary Controller," IEEE Trans. Power Syst. IEEE Transactions on Power Systems, vol. 28, no. 3, pp. 3174–3183, 2013.

[23] L. Harnefors, N. Johansson, L. Zhang, and B. Berggren, "Interarea Oscillation Damping Using Active-Power Modulation of Multiterminal HVDC Transmissions," IEEE Trans. Power Syst. IEEE Transactions on Power Systems, vol. 29, no. 5, pp. 2529–2538, 2014.

[24] G. Gurrala and I. Sen, "Power System Stabilizers Design for Interconnected Power Systems," *IEEE Trans. Power Syst. IEEE Transactions on Power Systems*, vol. 25, no. 2, pp. 1042–1051, 2010.

[25] M. Aboul-Ela, A. Sallam, J. Mccalley, and A. Fouad, "Damping controller design for power system oscillations using global signals," *IEEE Trans. Power Syst. IEEE Transactions on Power Systems*, vol. 11, no. 2, pp. 767–773, 1996.

[26] M. R. Younis and R. Iravani, "Wide-area damping control for inter-area oscillations: A comprehensive review," *2013 IEEE Electrical Power & Energy Conference*, 2013

[27] I. Kamwa, R. Grondin, and G. Trudel, "IEEE PSS2B Versus PSS4B: The Limits of Performance of Modern Power System Stabilizers," IEEE Trans. Power Syst. IEEE Transactions on Power Systems, vol. 20, no. 2, pp. 903–915, 2005.

[28] Y. Yasaei, M. Karimi-Ghartemani, A. Bakhshai, and M. Parniani, "Design of a nonlinear power system stabilizer," *2010 IEEE International Symposium on Industrial Electronics*, 2010.

[29] A. Dysko, W. Leithead, and J. O'reilly, "Enhanced Power System Stability by Coordinated PSS Design," *IEEE Trans. Power Syst. IEEE Transactions on Power Systems*, vol. 25, no. 1, pp. 413–422, 2010.

[30] M. Mokhtari, F. Aminifar, D. Nazarpour, and S. Golshannavaz, "Wide-area power oscillation damping with a fuzzy controller compensating the continuous communication delays," *IEEE Trans. Power Syst. IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 1997–2005, 2013.

[31] X. Xie, J. Xiao, C. Lu, and Y. Han, "Wide-area stability control for damping interarea oscillations of interconnected power systems," IEE Proc., Gener. Transm. Distrib. *IEE Proceedings - Generation, Transmission and Distribution, vol. 153, no. 5, p. 507, 2006*.

[32] K. Uhlen, L. Vanfretti, M. M. D. Oliveira, V. H. Aarstrand, and J. O. Gjerde, "Wide-Area Power Oscillation Damper implementation and testing in the Norwegian transmission network," *2012 IEEE Power and Energy Society General Meeting*, 2012

[33] W. Yao and L. Jiang, "Wide-area damping controller of FACTS devices for inter-area oscillations considering communication time delays," *2014 IEEE PES General Meeting | Conference & Exposition*, 2014.

[34] J. He, X. Wu, P. Li, C. Lu, and J. Wu, "Design and experiment of wide area HVDC supplementary damping controller considering time delay in China southern power grid," IET Generation, Transmission & Distribution, vol. 3, no. 1, pp. 17–25, Jan. 2009.

[35] Y. Chang, H. Chen, and F. Zhang, "Design of Multi-objective Robust HVDC Supplementary Controller Based on Global Signal," *2006 IEEE PES Power Systems Conference and Exposition*, 2006.

[36] F. Liu, R. Yokoyama, Y. C. Zhou, and M. Wu, "Design of robust damping controllers of FACTs devices with considering time-delay of wide-area signals," *2011 IEEE Trondheim PowerTech*, 2011

[37] J. Ma, Z. Wang, S. Wang, X. Gao, and T. Wang, "Classification and regression tree-based adaptive damping control of inter-area oscillations using wide-area signals," *IET Generation, Transmission & Distribution*, vol. 8, no. 6, pp. 1177–1186, Jan. 2014.

[38] J. Arif, S. Ray, and B. Chaudhuri, "MultiVAriable Self-Tuning Feedback Linearization Controller for Power Oscillation Damping," *IEEE Transactions on Control Systems Technology IEEE Trans. Contr. Syst. Technol.*, vol. 22, no. 4, pp. 1519–1526, 2014.

[39] R. Preece and J. V. Milanovic, "Tuning of a Damping Controller for Multiterminal VSC-HVDC Grids Using the Probabilistic Collocation Method," IEEE Transactions on Power

Delivery IEEE Trans. Power Delivery, vol. 29, no. 1, pp. 318–326, 2014.

[40] R. Goldoost-Soloot, Y. Mishra, and G. Ledwich, “Wide-area damping control for inter-area oscillations using inverse filtering technique,” *IET Generation, Transmission & Distribution*, vol. 9, no. 13, pp. 1534–1543, Jan. 2015.

[41] W. Yao, L. Jiang, J. Wen, Q. Wu, and S. Cheng, “Wide-Area Damping Controller for Power System Interarea Oscillations: A Networked Predictive Control Approach,” *IEEE Transactions on Control Systems Technology IEEE Trans. Contr. Syst. Technol.*, vol. 23, no. 1, pp. 27–36, 2015.

[42] Harikrishna, D., R. S. Dhekekar, and N. V. Srikanth. "A Novel Approach to Dynamic Stability Enhancement Using PID Damped Fuzzy Susceptance Controlled SVC."*2011 IEEE/PES Power Systems Conference and Exposition* (2011).

[43] Arzeha, N. A., M. W. Mustafa, and R. Mohamad Idris. "Fuzzy-based Static VAR Compensator Controller for Damping Power System Disturbances." 2012 *IEEE International Power Engineering and Optimization Conference (2012).*

[44] Arunprasanth, S., U. D. Annakkage, and Rick Kuffel. "Real Time Digital Simulation of a Static Var Compensator with Fuzzy Supervisory Control." *2013 IEEE 8th International Conference on Industrial and Information Systems (2013).*

[45] Lu, J., M.h. Nehrir, and D.a. Pierre. "A Fuzzy Logic-based Adaptive Damping Controller for Static VAR Compensator." *Electric Power Systems Research 68.2 (2004): 113-18.*

[46] Fang, D.z., Y. Xiaodong, T.s. Chung, and K.p. Wong. "Adaptive Fuzzy-Logic SVC Damping Controller Using Strategy of Oscillation Energy Descent." *IEEE Trans. Power Syst. IEEE Transactions on Power Systems 19.3 (2004): 1414-421.*

[47] Ellithy, K., and A. Al-Naamany. "A Hybrid Neuro-fuzzy Static Var Compensator Stabilizer for Power System Damping Improvement in the Presence of Load Parameters Uncertainty." *Electric Power Systems Research 56.3 (2000): 211-23*

[48] Magaji, Nuraddeen, M.w. Mustafa, and Z. Bint Muda. "Power System Damping Using GA Based Fuzzy Controlled SVC Device." *TENCON 2009 - 2009 IEEE Region 10 Conference (2009).*

[49] P. Kundur, N. J. Balu, and M. G. Lauby, *Power system stability and control*. New York: McGraw-Hill, 1994.

[50] P. W. Sauer and M. A. Pai, *Power system dynamics and stability*. Upper Saddle River, NJ: Prentice Hall, 1998.

[51] M. Klein, G. Rogers, and P. Kundur, "A fundamental study of inter-area oscillations in power systems," *IEEE Trans. Power Syst. IEEE Transactions on Power Systems*, vol. 6, no. 3, pp. 914–921, 1991.

[52] E. V. Larsen and D. A. Swann, "Applying Power System Stabilizers. Part I: General Concepts," *IEEE Power Engineering Review IEEE Power Eng. Rev.*, vol. PER-1, no. 6, pp. 62–63, 1981.

[53] E. V. Larsen and D. A. Swann, "Applying Power System Stabilizers Part II: Performance Objectives and Tuning Concepts," *IEEE Power Engineering Review IEEE Power Eng. Rev.*, vol. PER-1, no. 6, pp. 63–63, 1981.

[54] "IEEE Recommended Practice for Excitation System Models for Power System Stability Studies," 2015.

[55] K. Ellithy, "Tuning of SVC Stabilizers over a Wide Range of Load Parameters Using Pole-Placement Technique" *The Journal of Engineering Research, vol. 5, no. 1, pp. 71–78, 2008.*

56] L. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.

[57] K. M. Passino and S. Yurkovich, Fuzzy control. Menlo Park, CA: Addison-Wesley, 1998.

[58] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Transactions on Systems, Man, and Cybernetics IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, no. 1, pp. 116–132, 1985.

[59] J. Saade and H. Diab, "Defuzzification techniques for fuzzy controllers," *IEEE Trans. Syst., Man, Cybern. B IEEE Transactions on Systems, Man and Cybernetics, Part B*

(Cybernetics), vol. 30, no. 1, pp. 223–229, 2000.

[60] F. Smith, "Selecting inference and defuzzification techniques for fuzzy logic control," *UKACC International Conference on Control (CONTROL '98)*, 1998.

Appendix-A Schematics & PCBs

A)Bi-Polar to Unipolar Module

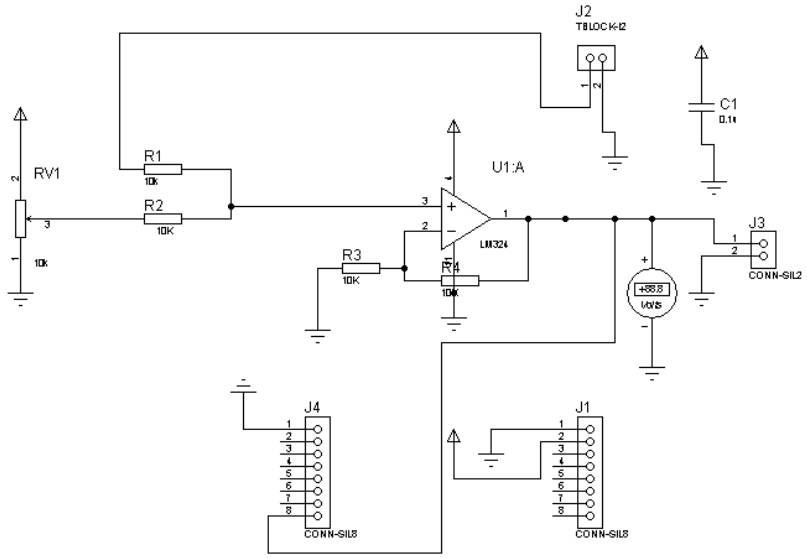


Figure8.1

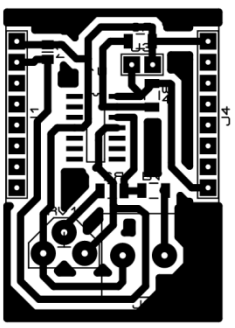


Figure8.2 PCB

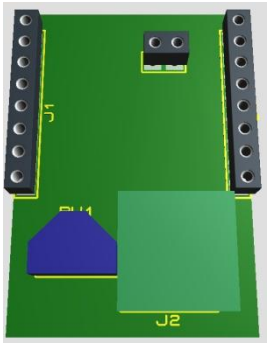


Figure8.3 3D Front Side

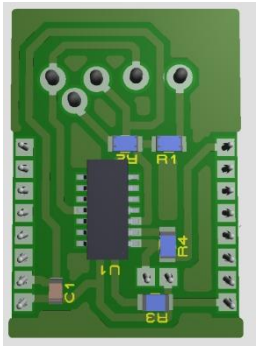
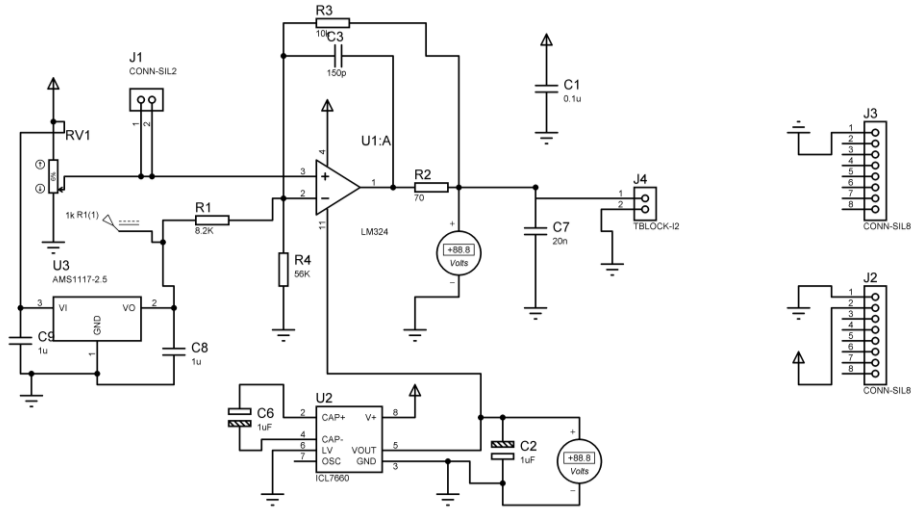
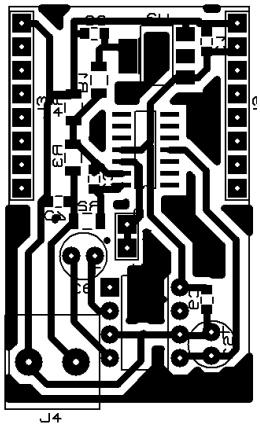
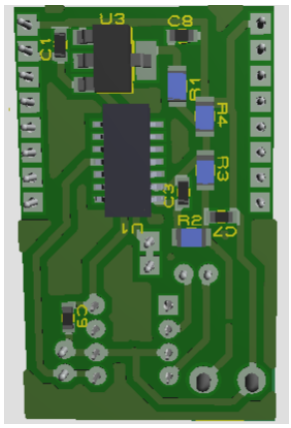


Figure8.4 3D Back Side

B)Unipolarto Bi-Polar Module







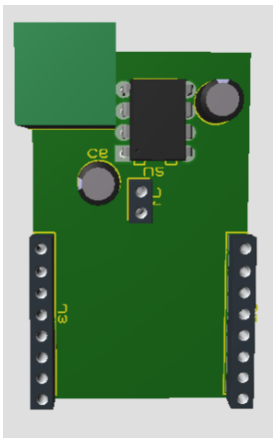
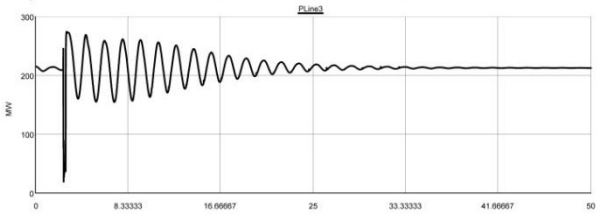
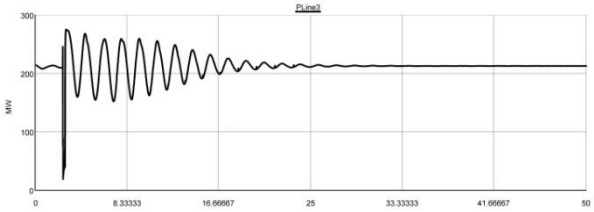
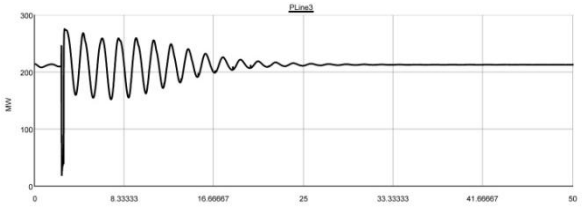
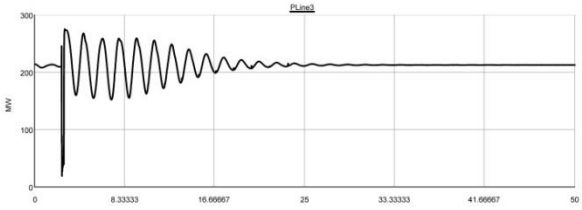
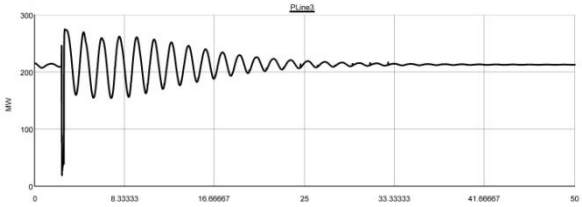
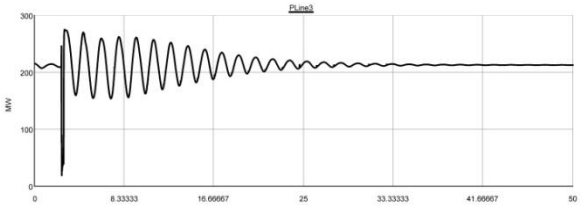
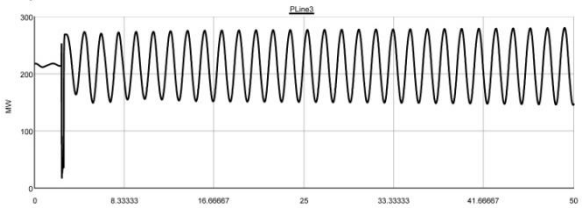


Figure8.5 PCB Figure8.6 3D Front Side Figure8.7 3D Back Side

Appendix-B Tested Rule Bases

#	Rule Base	System Response
1		
2		
3		
4		
5		
6		
7		

Rule Base 1

		Err						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Derr	E_BN	BN	BN	BN	MN	MN	SN	Z
	E_MN	BN	MN	MN	MN	SN	Z	SP
	E_SN	BN	MN	SN	SN	Z	SP	MP
	E_Z	MN	MN	SN	Z	SP	MP	MP
	E_SP	MN	SN	Z	SP	SP	MP	LP
	E_MP	SN	Z	SP	MP	MP	MP	LP
	E_BP	Z	SP	MP	MP	LP	LP	LP

Rule Base 2

		Err						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Derr	E_BN	BN	BN	BN	MN	MN	SN	Z
	E_MN	BN	MN	MN	MN	SN	Z	SP
	E_SN	BN	MN	MN	SN	Z	SP	MP
	E_Z	MN	MN	SN	Z	SP	MP	MP
	E_SP	MN	SN	Z	SP	MP	MP	LP
	E_MP	SN	Z	SP	MP	MP	MP	LP
	E_BP	Z	SP	MP	MP	LP	LP	LP

Rule Base 3

		Err						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Derr	E_BN	BN	BN	BN	MN	MN	SN	Z
	E_MN	BN	BN	MN	MN	SN	Z	SP
	E_SN	BN	MN	MN	SN	Z	SP	MP
	E_Z	MN	MN	SN	Z	SP	MP	MP
	E_SP	MN	SN	Z	SP	MP	MP	LP
	E_MP	SN	Z	SP	MP	MP	LP	LP
	E_BP	Z	SP	MP	MP	LP	LP	LP

Rule Base 4

		Err						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Derr	E_BN	BN	BN	MN	MN	MN	SN	Z
	E_MN	BN	MN	MN	MN	SN	Z	SP
	E_SN	MN	MN	MN	SN	Z	SP	MP
	E_Z	MN	MN	SN	Z	SP	MP	MP
	E_SP	MN	SN	Z	SP	MP	MP	MP
	E_MP	SN	Z	SP	MP	MP	MP	LP
	E_BP	Z	SP	MP	MP	MP	LP	LP

Rule Base 5

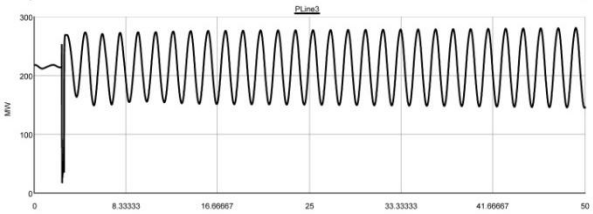
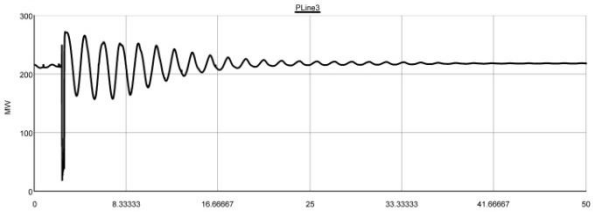
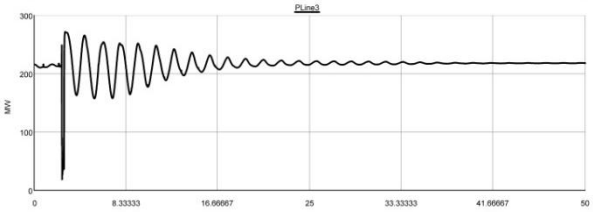
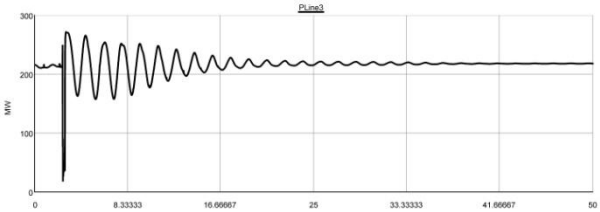
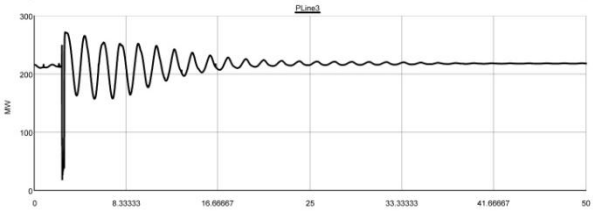
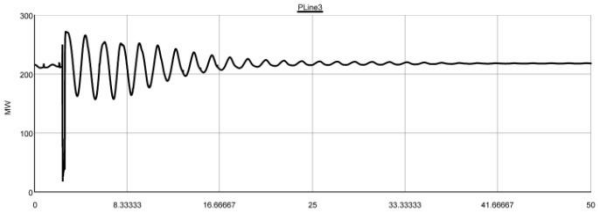
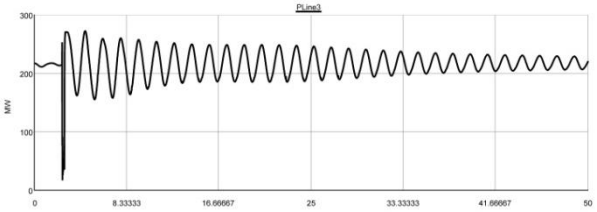
		Err						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Derr	E_BN	BN	BN	MN	MN	MN	SN	Z
	E_MN	BN	MN	MN	MN	SN	Z	SP
	E_SN	MN	MN	SN	SN	Z	SP	SP
	E_Z	MN	SN	SN	Z	SP	SP	MP
	E_SP	SN	SN	Z	SP	SP	MP	MP
	E_MP	SN	Z	SP	MP	MP	MP	LP
	E_BP	Z	SP	MP	MP	MP	LP	LP

Rule Base 6

		Err						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Derr	E_BN	BN	BN	MN	MN	SN	SN	Z
	E_MN	BN	MN	MN	SN	SN	Z	SP
	E_SN	MN	MN	SN	SN	Z	SP	SP
	E_Z	MN	SN	SN	Z	SP	SP	MP
	E_SP	SN	SN	Z	SP	SP	MP	MP
	E_MP	SN	Z	SP	SP	MP	MP	LP
	E_BP	Z	SP	SP	MP	MP	LP	LP

Rule Base 7

		Err						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Derr	E_BN	BN	BN	MN	SN	SN	SN	Z
	E_MN	BN	MN	SN	SN	SN	Z	Z
	E_SN	MN	SN	SN	Z	Z	Z	SP
	E_Z	SN	SN	Z	Z	Z	SP	SP
	E_SP	SN	Z	Z	Z	SP	SP	MP
	E_MP	Z	Z	SP	SP	SP	MP	LP
	E_BP	Z	SP	SP	SP	MP	LP	LP

#	Rule Base	System Response
8	Rule Base 8	
9	Rule Base 9	
10	Rule Base 10	
11	Rule Base 11	
12	Rule Base 12	
13	Rule Base 13	
14	Rule Base 14	

Rule Base 8

		Err						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Derr	E_BN	BN	MN	MN	SN	SN	SN	Z
	E_MN	MN	MN	SN	SN	SN	Z	Z
	E_SN	MN	SN	SN	Z	Z	Z	SP
	E_Z	SN	SN	Z	Z	Z	SP	SP
	E_SP	SN	Z	Z	Z	SP	SP	MP
	E_MP	Z	Z	SP	SP	SP	MP	MP
	E_BP	Z	SP	SP	SP	MP	MP	LP

Rule Base 9

		Err						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Derr	E_BN	BN	BN	BN	SN	Z	Z	SP
	E_MN	BN	BN	MN	SN	Z	SP	MP
	E_SN	BN	BN	MN	Z	SP	MP	LP
	E_Z	BN	MN	SN	Z	SP	MP	LP
	E_SP	BN	MN	SN	Z	MP	LP	LP
	E_MP	MN	SN	Z	SP	MP	LP	LP
	E_BP	SN	Z	Z	SP	LP	LP	LP

Rule Base 10

		Err						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Derr	E_BN	BN	BN	BN	MN	Z	Z	SP
	E_MN	BN	BN	MN	SN	Z	SP	MP
	E_SN	BN	BN	MN	Z	SP	MP	LP
	E_Z	BN	MN	SN	Z	SP	MP	LP
	E_SP	BN	MN	SN	Z	MP	LP	LP
	E_MP	MN	SN	Z	SP	MP	LP	LP
	E_BP	SN	Z	Z	MP	LP	LP	LP

Rule Base 11

		Err						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Derr	E_BN	BN	BN	BN	MN	Z	Z	SP
	E_MN	BN	BN	MN	SN	Z	SP	MP
	E_SN	BN	MN	MN	Z	SP	SP	LP
	E_Z	BN	MN	SN	Z	SP	MP	LP
	E_SP	BN	SN	SN	Z	MP	MP	LP
	E_MP	MN	SN	Z	SP	MP	LP	LP
	E_BP	SN	Z	Z	MP	LP	LP	LP

Rule Base 12

		Err						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Derr	E_BN	BN	BN	BN	MN	Z	Z	SP
	E_MN	BN	BN	MN	SN	Z	SP	MP
	E_SN	BN	MN	MN	Z	SP	SP	MP
	E_Z	BN	MN	SN	Z	SP	MP	LP
	E_SP	MN	SN	SN	Z	MP	MP	LP
	E_MP	MN	SN	Z	SP	MP	LP	LP
	E_BP	SN	Z	Z	MP	LP	LP	LP

Rule Base 13

		Err						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Derr	E_BN	BN	BN	BN	MN	Z	Z	SP
	E_MN	BN	BN	MN	SN	Z	SP	SP
	E_SN	BN	MN	MN	Z	SP	SP	MP
	E_Z	MN	MN	SN	Z	SP	MP	MP
	E_SP	MN	SN	SN	Z	MP	MP	LP
	E_MP	SN	SN	Z	SP	MP	LP	LP
	E_BP	SN	Z	Z	MP	LP	LP	LP

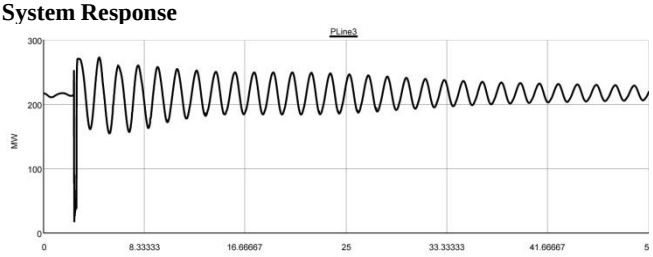
Rule Base 14

		Err						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Derr	E_BN	BN	BN	BN	MN	Z	Z	SP
	E_MN	BN	BN	MN	SN	Z	SP	SP
	E_SN	BN	MN	SN	Z	SP	SP	MP
	E_Z	MN	MN	SN	Z	SP	MP	MP
	E_SP	MN	SN	SN	Z	SP	MP	LP
	E_MP	SN	SN	Z	SP	MP	LP	LP
	E_BP	SN	Z	Z	MP	LP	LP	LP

#	Rule Base	System Response
15		

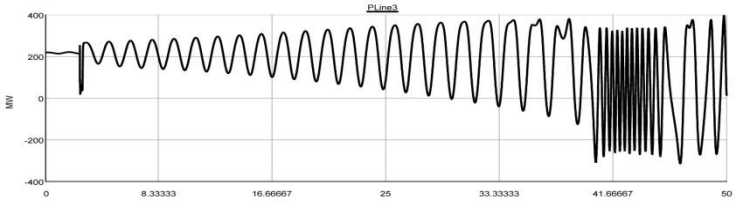
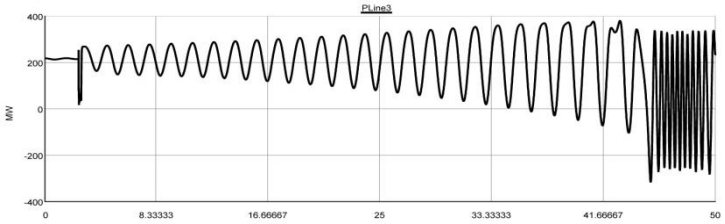
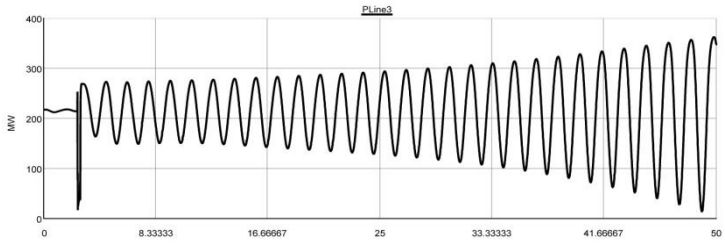
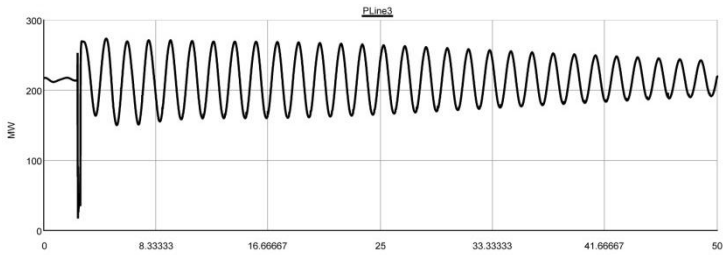
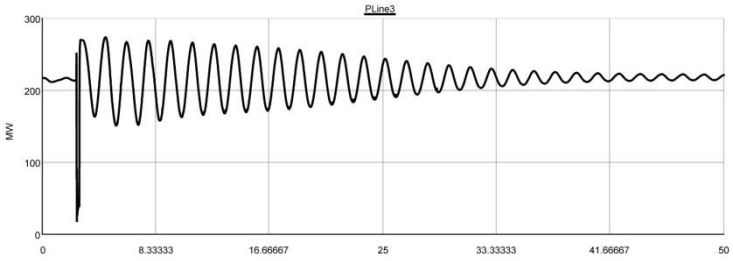
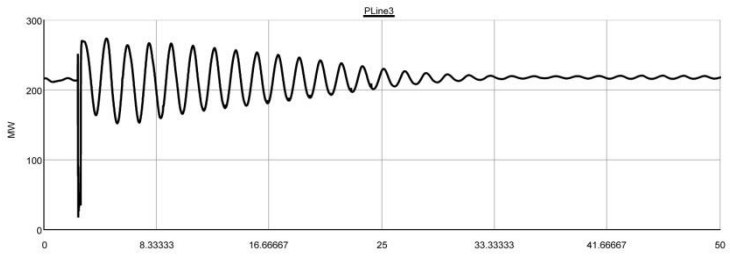
Rule Base 15

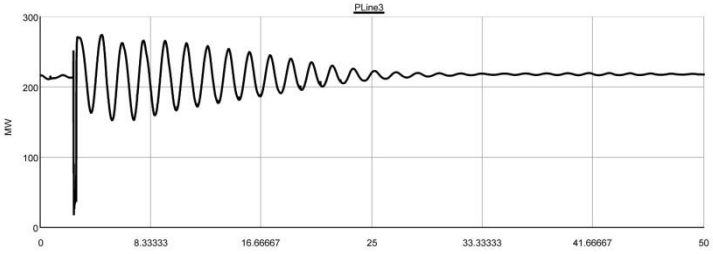
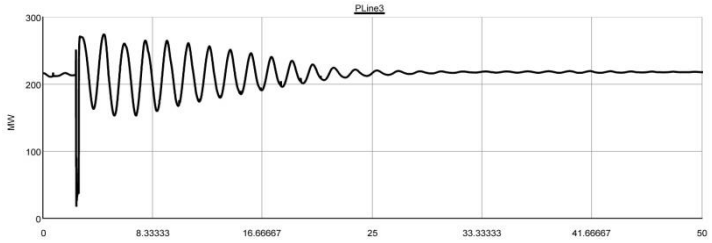
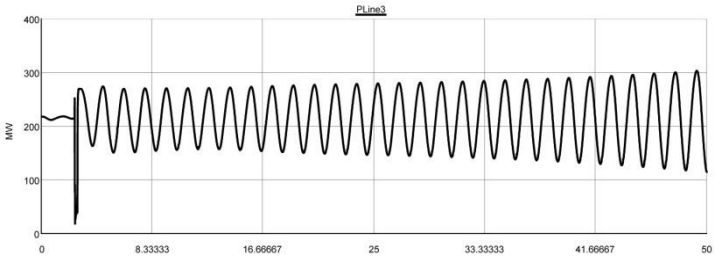
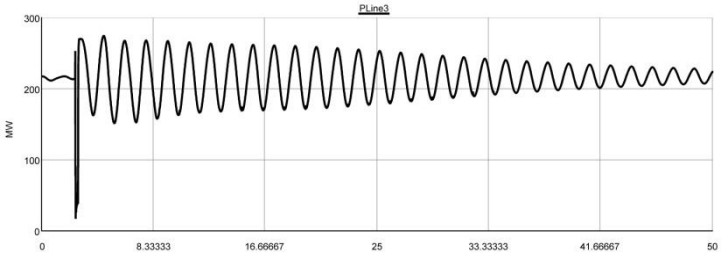
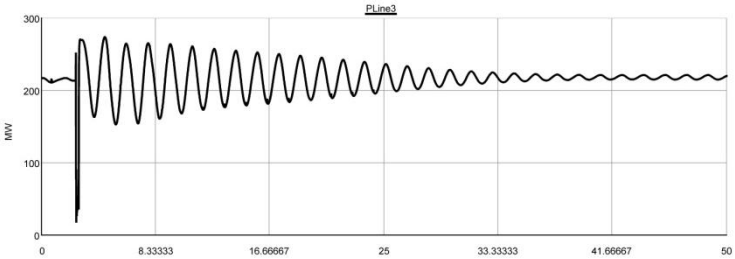
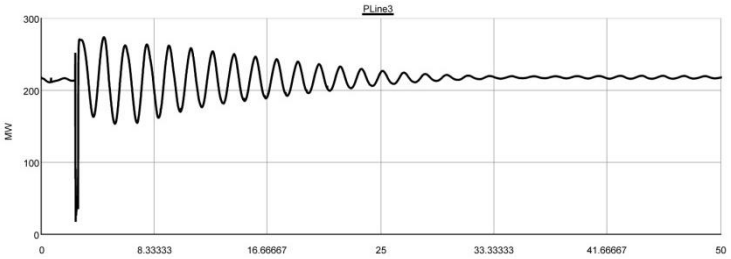
		Err						
		DE_BN	DE_MN	DE_SN	DE_Z	DE_SP	DE_MP	DE_BP
Derr	E_BN	BN	BN	MN	MN	Z	Z	SP
	E_MN	BN	BN	MN	SN	Z	SP	SP
	E_SN	BN	MN	SN	Z	Z	SP	MP
	E_Z	MN	MN	SN	Z	SP	MP	MP
	E_SP	MN	SN	Z	Z	SP	MP	LP
	E_MP	SN	SN	Z	SP	MP	LP	LP
	E_BP	SN	Z	Z	MP	MP	LP	LP

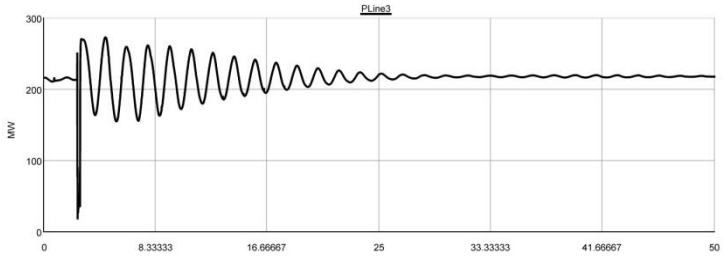
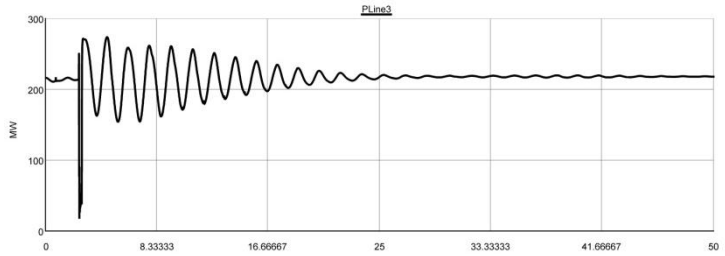
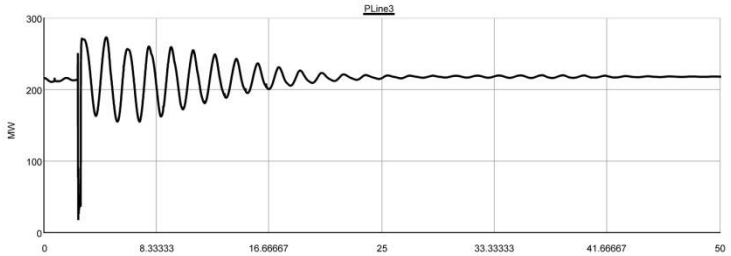
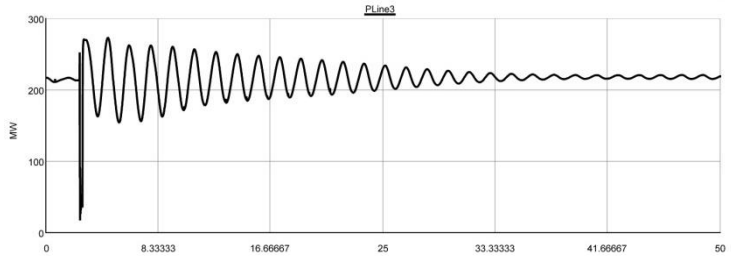
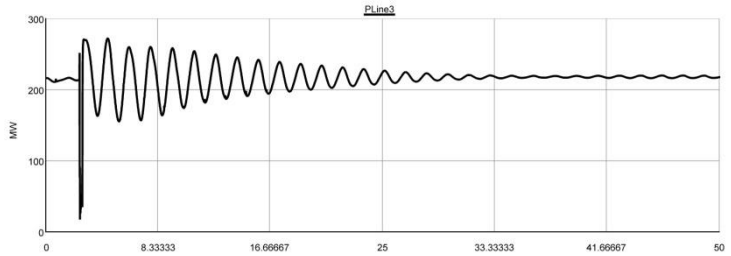
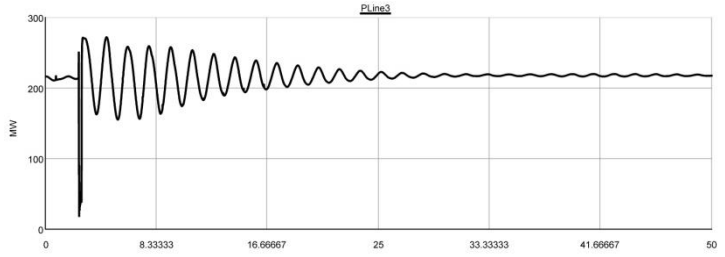


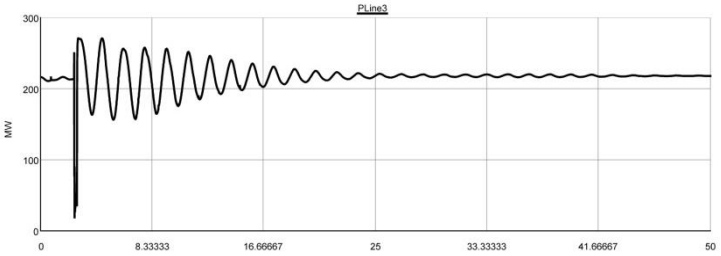
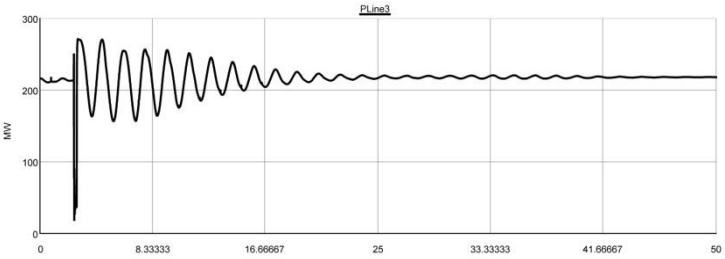
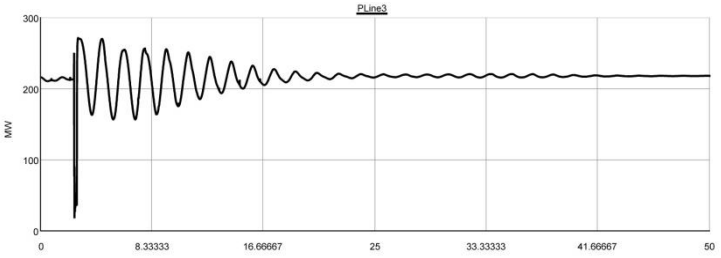
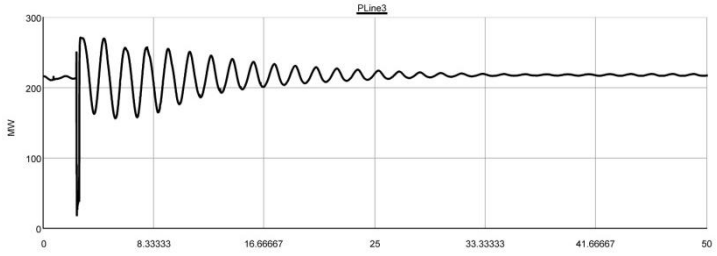
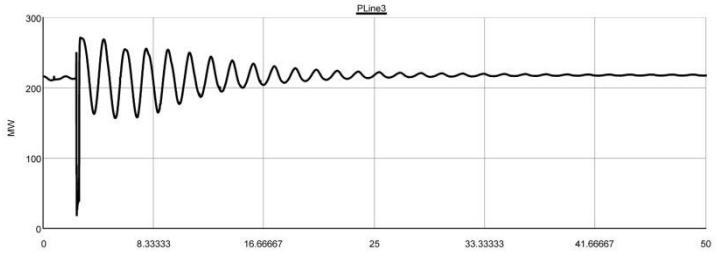
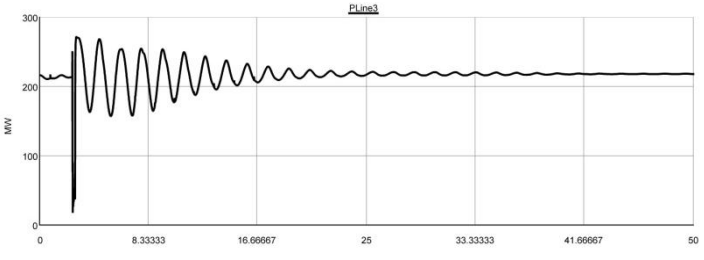
Appendix-C Tested MF Limits

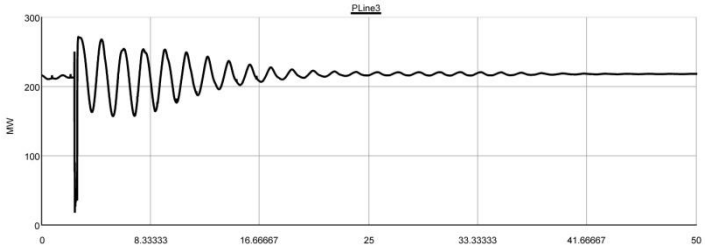
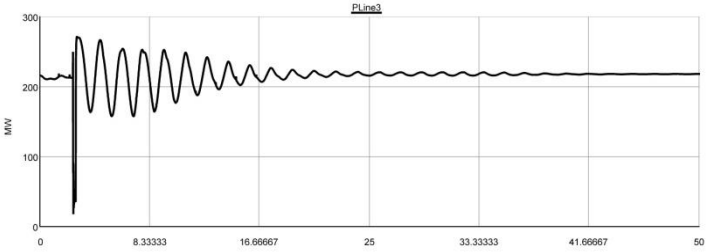
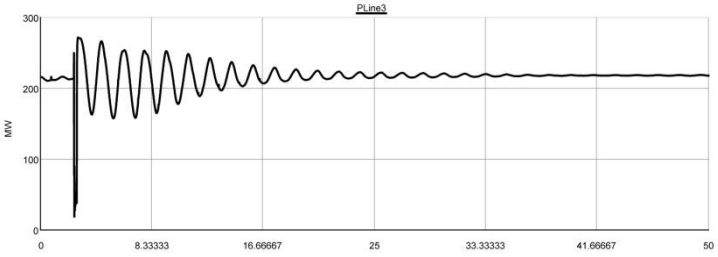
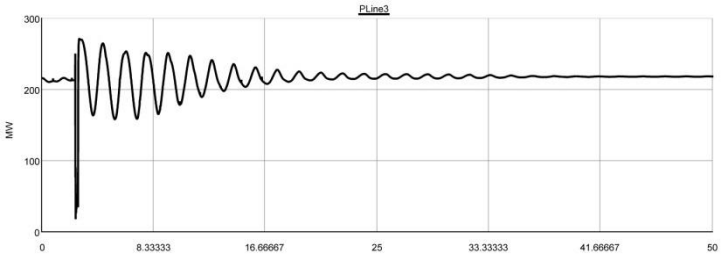
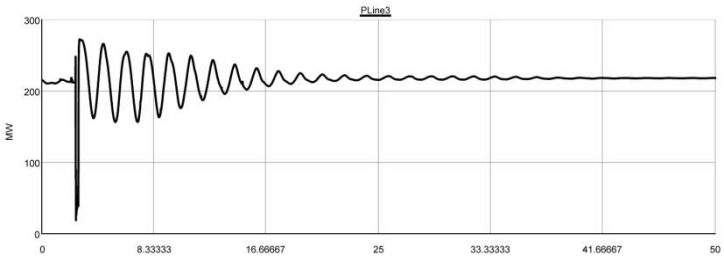
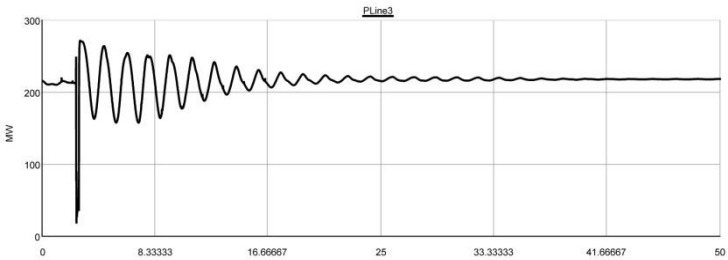
Table 10.1 Membership Function Tested Limits

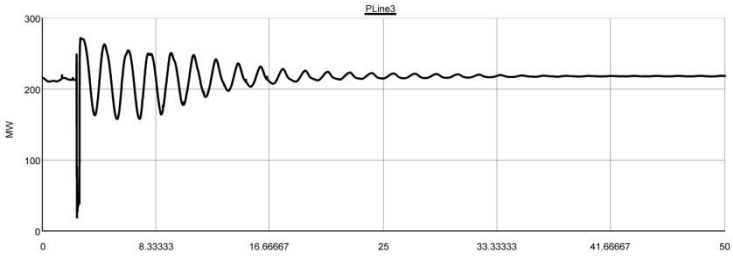
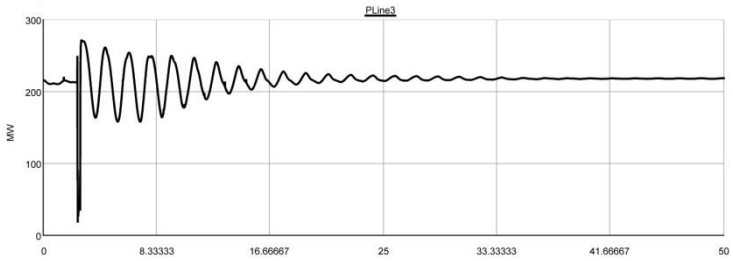
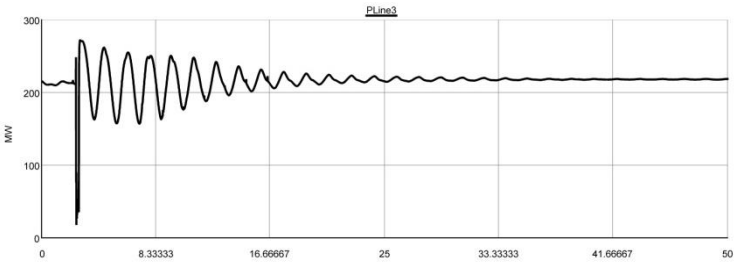
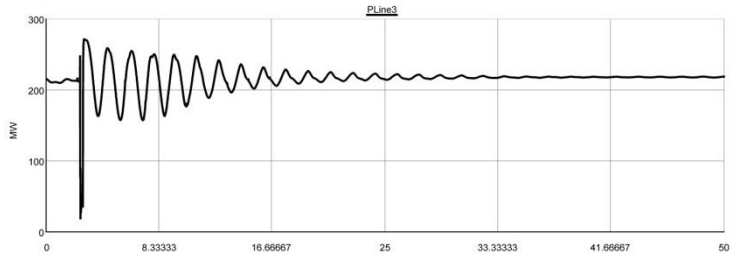
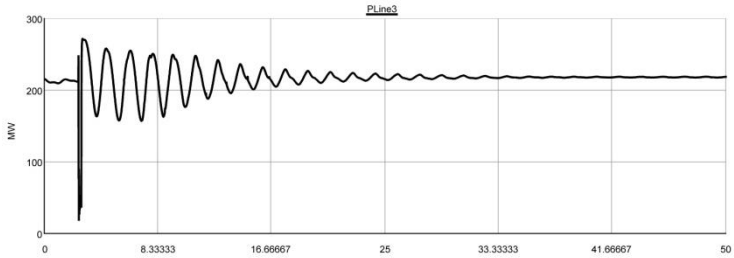
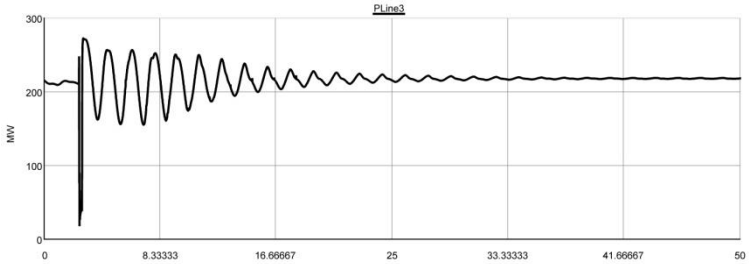
#	BN	MN	SN	Z	SP	MP	BP	
1	-1	-0.2	-0.1	0	0.1	0.2	1	
2	-1	-0.3	-0.1	0	0.1	0.3	1	
3	-1	-0.4	-0.1	0	0.1	0.4	1	
4	-1	-0.5	-0.1	0	0.1	0.5	1	
5	-1	-0.6	-0.1	0	0.1	0.6	1	
6	-1	-0.7	-0.1	0	0.1	0.7	1	

#	BN	MN	SN	Z	SP	MP	BP	
7	-1	-0.8	-0.1	0	0.1	0.8	1	
8	-1	-0.9	-0.1	0	0.1	0.9	1	
9	-1	-0.3	-0.2	0	0.2	0.3	1	
10	-1	-0.4	-0.2	0	0.2	0.4	1	
11	-1	-0.5	-0.2	0	0.2	0.5	1	
12	-1	-0.6	-0.2	0	0.2	0.6	1	

#	BN	MN	SN	Z	SP	MP	BP	
13	-1	-0.7	-0.2	0	0.2	0.7	1	
14	-1	-0.8	-0.2	0	0.2	0.8	1	
15	-1	-0.9	-0.2	0	0.2	0.9	1	
16	-1	-0.4	-0.3	0	0.3	0.4	1	
17	-1	-0.5	-0.3	0	0.3	0.5	1	
18	-1	-0.6	-0.3	0	0.3	0.6	1	

#	BN	MN	SN	Z	SP	MP	BP	
19	-1	-0.7	-0.3	0	0.3	0.7	1	
20	-1	-0.8	-0.3	0	0.3	0.8	1	
21	-1	-0.9	-0.3	0	0.3	0.9	1	
22	-1	-0.5	-0.4	0	0.4	0.5	1	
23	-1	-0.6	-0.4	0	0.4	0.6	1	
24	-1	-0.7	-0.4	0	0.4	0.7	1	

#	BN	MN	SN	Z	SP	MP	BP	
25	-1	-0.8	-0.4	0	0.4	0.8	1	
26	-1	-0.9	-0.4	0	0.4	0.9	1	
27	-1	-0.6	-0.5	0	0.5	0.6	1	
28	-1	-0.7	-0.5	0	0.5	0.7	1	
29	-1	-0.8	-0.5	0	0.5	0.8	1	
30	-1	-0.9	-0.5	0	0.5	0.9	1	

#	BN	MN	SN	Z	SP	MP	BP	
31	-1	-0.7	-0.6	0	0.6	0.7	1	
32	-1	-0.8	-0.6	0	0.6	0.8	1	
33	-1	-0.9	-0.6	0	0.6	0.9	1	
34	-1	-0.8	-0.7	0	0.7	0.8	1	
35	-1	-0.9	-0.7	0	0.7	0.9	1	
36	-1	-0.9	-0.8	0	0.8	0.9	1	

Appendix-D Microcontroller Code

```
void main() {
int adc_value;

// VAriable def. Part
float derr_scale,derr_DE_BN,derr_DE_MN,derr_DE_SN,derr_DE_Z,derr_DE_SP,derr_DE_MP,derr_DE_BP;
float err_scale,err_E_BN,err_E_MN,err_E_SN,err_E_Z,err_E_SP,err_E_MP,err_E_BP;
float out_scale,out_BN,out_MN,out_SN,out_Z,out_SP,out_MP,out_LP;

float derr,err,out;
float P_read,P_hist,P_err,P_err_his,P_derr,P_calc,P_enable;
float output_sc, ERR_sc, DERR_sc,his_err,v_his_err, P_ERR_sc, P_DERR_sc;
float inp_volt;

int input_reading, count,i,xx;
unsigned int ADC_in[250];
unsigned int P_ADC_in[250];

float DAC_out[100];
unsigned long xxx;


    set_init();
            // 04/01/2015 //
    output_sc=2;  //1; //  5      //  1.3        ;

    ERR_sc=1;     //1;   //  25;     //  30   ///50
    DERR_sc=0.1;   //0.08; //  25      //  60
    count =0;
    P_ERR_sc=1;
    P_DERR_sc=1;

    i=0;
    his_err=0;
    v_his_err=0;

    xxx=0;
    P_hist=0;
    P_err_his=0;
    uart2_write_text("Start The Project");
    for (i=0;i<90;i++) ADC_in[i]= ADC1_Read(11);
    for (i=0;i<90;i++) DAC_out[i]=0;

    for (i=0;i<90;i++) P_ADC_in[i]= ADC1_Read(12);



 while(1)
 {
      // read power with moving average
      P_read= M_avg(&P_ADC_in,100,ADC1_Read(12));
      // convert the input voltage into power real value
      P_read=inpt_power(P_read);
      // calculate P error
      P_err= P_read -P_hist;
      // Save P History
      P_hist= P_read;
      // Calculate P Delta Error
      P_derr= P_err - P_err_his;
      // Save P Error History
      P_err_his = P_err;

      P_err=P_err/P_ERR_sc;
      P_derr=P_derr/P_DERR_sc;

      // Fuzzyfication of the P Delta Error
      fuzzification3 (P_derr ,&derr_DE_BN,&derr_DE_MN,&derr_DE_SN,&derr_DE_Z,&derr_DE_SP,&derr_DE_MP,&derr_DE_BP);
      // Fuzzyfication of the P Error
      fuzzification4 (P_err,&err_E_BN,&err_E_MN,&err_E_SN,&err_E_Z,&err_E_SP,&err_E_MP,&err_E_BP);
      // Calculate the Rule Base

rulebase2(derr_DE_BN,derr_DE_MN,derr_DE_SN,derr_DE_Z,derr_DE_SP,derr_DE_MP,derr_DE_BP,err_E_BN,err_E_MN,err_E_SN,err_E_Z,err_E_SP,err_E_MP,err_E_BP,&out_BN,&out_M
N,&out_SN,&out_Z,&out_SP,&out_MP,&out_LP);
      // Calculate the Scaling_factor
      P_enable=defuzzification (out_BN,1 ,out_MN,0.8 ,out_SN,0.2 ,out_Z,0 ,out_SP,0.2 ,out_MP,0.8 ,out_LP,1);

      if (P_enable>.9)
      {
       // Calculate Delta (Speed Deviation)
         if ((P_read<(413*1.2))&&((P_read>(413*0.8))))
            derr_sc=Calc_Delta_sc (P_read,-0.00546,3.77583);

      }


      derr_sc=Calc_Delta_sc (P_calc,-0.00546,3.77583);

      // read Error (Speed Deviation) with moving average
      err= M_avg(&ADC_in,100,ADC1_Read(11));
      // convert the input voltage into Error value
      err=inpt_voltage(err);
      // Calculate Delta Error
      derr=err-his_err;
      // Save Error History
```

```
        his_err=err;

        // Save Error Scalling
        err=err/err_sc;
        // Save Delta Error Scalling
        derr=derr/derr_sc;
        // Fuzzyfication of the Delta Error
        fuzzification1 (derr,&derr_DE_BN,&derr_DE_MN,&derr_DE_SN,&derr_DE_Z,&derr_DE_SP,&derr_DE_MP,&derr_DE_BP);
        // Fuzzyfication of the Error
        fuzzification2 (err,&err_E_BN,&err_E_MN,&err_E_SN,&err_E_Z,&err_E_SP,&err_E_MP,&err_E_BP);
        // Calculate the Rule Base

rulebase(derr_DE_BN,derr_DE_MN,derr_DE_SN,derr_DE_Z,derr_DE_SP,derr_DE_MP,derr_DE_BP,err_E_BN,err_E_MN,err_E_SN,err_E_Z,err_E_SP,err_E_MP,err_E_BP,&out_BN,&out_MN,
&out_SN,&out_Z,&out_SP,&out_MP,&out_LP);
        // Calculate the output
        out=defuzzification (out_BN,-1 ,out_MN,-0.7 ,out_SN,-0.5 ,out_Z,0 ,out_SP,0.5 ,out_MP,0.7 ,out_LP,1);

         out = out *  output_sc;
         out= M_avg_f(&DAC_out,30,out);

         if (out>= 1) out =1;
         else if (out<= -1) out=-1;
              out=out;

 ////////////////////////////////
      output_voltage(out);

     GPIOA_ODR.b6=~GPIOA_ODR.b6;

 }
}
```

#include"other_functions.c"

```
void set_init()
{
  GPIO_Digital_Output(&GPIOA_BASE, _GPIO_PINMASK_ALL); // Set PORTA as digital output
  GPIO_Digital_Output(&GPIOB_BASE, _GPIO_PINMASK_ALL); // Set PORTB as digital output
  GPIO_Digital_Output(&GPIOC_BASE, _GPIO_PINMASK_ALL); // Set PORTC as digital output
  GPIO_Digital_Output(&GPIOD_BASE, _GPIO_PINMASK_ALL); // Set PORTD as digital output
  GPIO_Digital_Output(&GPIOE_BASE, _GPIO_PINMASK_ALL); // Set PORTE as digital output
  GPIO_Digital_Output(&GPIOD_BASE, _GPIO_PINMASK_12); // Set PORTD as digital output
  GPIO_Digital_Output(&GPIOD_BASE, _GPIO_PINMASK_13); // Set PORTD as digital output
  GPIO_Digital_Output(&GPIOD_BASE, _GPIO_PINMASK_14); // Set PORTD as digital output
  GPIO_Digital_Output(&GPIOD_BASE, _GPIO_PINMASK_15); // Set PORTD as digital output
  GPIO_Digital_Output(&GPIOA_BASE, _GPIO_PINMASK_6); // Set PORTD as digital output
  ADC_Set_Input_Channel(_ADC_CHANNEL_10); // Set ADC channel 10 as an analog input
  ADC_Set_Input_Channel(_ADC_CHANNEL_11); // Set ADC channel 11 as an analog input

 ADC1_Init();                          // Initialize ADC module
 dac1_init(_DAC_CH1_ENABLE);
 uart2_init(9600);   // this one is working TX : PA2 , while RX: PA3
}

float limiter ( float inp, float lower,float upper)
{
  if (inp >= upper) return upper;
  else if (inp < lower) return lower;
  else return inp;
}

float f_min (float a, float b)
{
if (a>=b) return b;
else return a;
}
```

#include"fuzzy_functions.c"

```
float defuzzification (float BN, float BN_w , float MN , float MN_w , float SN , float SN_w , float Z , float Z_w , float SP , float SP_w , float MP , float MP_w , float LP , float LP_w)
{
return (( BN* BN_w +MN*MN_w +SN*SN_w +Z*Z_w +SP*SP_w +MP*MP_w +LP*LP_w)/(BN +MN +SN +Z +SP +MP +LP));
}

float f_max2 (float a1, float a2, float a3, float a4, float a5, float a6, float a7, float a8, float a9, float a10, float a11, float a12 ,float a13, float a14, float a15, float a16, float a17, float a18, float a19, float a20, float a21, float a22, float a23, float a24, float a25, float a26, float a27, float a28, float a29, float a30, float a31, float a32, float a33, float a34, float a35, float a36, float a37, float a38, float a39, float a40, float a41, float a42, float a43, float a44, float a45, float a46, float a47, float a48, float a49)
{
float fmax;
fmax=a1;
if (a2>fmax) fmax=a2;
if (a3>fmax) fmax=a3;
if (a4>fmax) fmax=a4;
if (a5>fmax) fmax=a5;
if (a6>fmax) fmax=a6;
if (a7>fmax) fmax=a7;
if (a8>fmax) fmax=a8;
if (a9>fmax) fmax=a9;
if (a10>fmax) fmax=a10;
if (a11>fmax) fmax=a11;
if (a12>fmax) fmax=a12;
if (a13>fmax) fmax=a13;
if (a14>fmax) fmax=a14;
if (a15>fmax) fmax=a15;
if (a16>fmax) fmax=a16;
if (a17>fmax) fmax=a17;
if (a18>fmax) fmax=a18;
if (a19>fmax) fmax=a19;
if (a20>fmax) fmax=a20;
if (a21>fmax) fmax=a21;
if (a22>fmax) fmax=a22;
if (a23>fmax) fmax=a23;
if (a24>fmax) fmax=a24;
if (a25>fmax) fmax=a25;
if (a26>fmax) fmax=a26;
if (a27>fmax) fmax=a27;
if (a28>fmax) fmax=a28;
if (a29>fmax) fmax=a29;
if (a30>fmax) fmax=a30;
if (a31>fmax) fmax=a31;
if (a32>fmax) fmax=a32;
if (a33>fmax) fmax=a33;
if (a34>fmax) fmax=a34;
if (a35>fmax) fmax=a35;
if (a36>fmax) fmax=a36;
if (a37>fmax) fmax=a37;
if (a38>fmax) fmax=a38;
if (a39>fmax) fmax=a39;
if (a40>fmax) fmax=a40;
if (a41>fmax) fmax=a41;
if (a42>fmax) fmax=a42;
if (a43>fmax) fmax=a43;
if (a44>fmax) fmax=a44;
if (a45>fmax) fmax=a45;
if (a46>fmax) fmax=a46;
if (a47>fmax) fmax=a47;
if (a48>fmax) fmax=a48;
if (a49>fmax) fmax=a49;
return fmax;
}

void fuzzification1 (float inp , float *VAr1 , float *VAr2 , float *VAr3 , float *VAr4 , float *VAr5 , float *VAr6 , float *VAr7)
{
 *VAr1=0; *VAr2=0; *VAr3=0; *VAr4=0; *VAr5=0; *VAr6=0; *VAr7=0;
if(inp<= -1 ) *VAr1=1;
else if ((inp<=-0.7)&& (inp>=-1))
{
      *VAr1=-3.33333333333333 *inp +(-2.33333333333333);
      *VAr2=3.33333333333333*inp + (3.33333333333333);
}
else if ((inp<=-0.5)&& (inp>-0.7))
{
      *VAr2=-5 *inp +(-2.5);
      *VAr3=5*inp + (3.5);
}

else if ((inp<=0)&& (inp>-0.5))
{
      *VAr3=-2 *inp +(0);
      *VAr4=2*inp + (1);
}
else if ((inp<=0.5)&& (inp>0))
{
      *VAr4=-2 *inp +(1);
      *VAr5=2*inp + (0);
}
else if ((inp<=0.7)&& (inp>0.5))
{
      *VAr5=-5 *inp +(3.5);
      *VAr6=5*inp + (-2.5);
}
else if ((inp<=1)&& (inp>0.7))
{
      *VAr6=-3.33333333333333 *inp +(3.33333333333333);
      *VAr7=3.33333333333333*inp + (-2.33333333333333);
}
```

```
else if (inp>1) *VAr7=1;
}

void fuzzification2 (float inp , float *VAr1 , float *VAr2 , float *VAr3 , float *VAr4 , float *VAr5 , float *VAr6 , float *VAr7)
{
 *VAr1=0; *VAr2=0; *VAr3=0; *VAr4=0; *VAr5=0; *VAr6=0; *VAr7=0;
if(inp<= -1 ) *VAr1=1;
else if ((inp<=-0.7)&& (inp>=-1))
{
	*VAr1=-3.33333333333333 *inp +(-2.33333333333333);
	*VAr2=3.33333333333333*inp + (3.33333333333333);
}
else if ((inp<=-0.5)&& (inp>-0.7))
{
	*VAr2=-5 *inp +(-2.5);
	*VAr3=5*inp + (3.5);
}
else if ((inp<=0)&& (inp>-0.5))
{
	*VAr3=-2 *inp +(0);
	*VAr4=2*inp + (1);
}
else if ((inp<=0.5)&& (inp>0))
{
	*VAr4=-2 *inp +(1);
	*VAr5=2*inp + (0);

}
else if ((inp<=0.7)&& (inp>0.5))
{
	*VAr5=-5 *inp +(3.5);
	*VAr6=5*inp + (-2.5);
}
else if ((inp<=1)&& (inp>0.7))
{
	*VAr6=-3.33333333333333 *inp +(3.33333333333333);
	*VAr7=3.33333333333333*inp + (-2.33333333333333);
}
else if (inp>1) *VAr7=1;
}

// The RuleBase Part
float rulebase (float derr_DE_BN , float derr_DE_MN , float derr_DE_SN , float derr_DE_Z , float derr_DE_SP , float derr_DE_MP , float derr_DE_BP , float err_E_BN , float err_E_MN , float err_E_SN , float err_E_Z , float err_E_SP , float err_E_MP , float err_E_BP , float *BN , float *MN , float *SN , float *Z , float *SP , float *MP , float *LP)
{
float BN1,BN2,BN3,BN4,BN5;
float MN1,MN2,MN3,MN4,MN5,MN6,MN7,MN8,MN9;
float SN1,SN2,SN3,SN4,SN5,SN6,SN7;
float Z1,Z2,Z3,Z4,Z5,Z6,Z7;
float SP1,SP2,SP3,SP4,SP5,SP6,SP7;
float MP1,MP2,MP3,MP4,MP5,MP6,MP7,MP8,MP9;
float LP1,LP2,LP3,LP4,LP5;
  // Rule base of BN
BN1=f_min(err_E_BN,derr_DE_BN);
BN2=f_min(err_E_BN,derr_DE_MN);
BN3=f_min(err_E_BN,derr_DE_SN);
BN4=f_min(err_E_MN,derr_DE_BN);
BN5=f_min(err_E_SN,derr_DE_BN);
*BN=f_max2(BN1 ,BN2 ,BN3 ,BN4 ,BN5 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);

  // Rule base of MN
MN1=f_min(err_E_BN,derr_DE_Z);
MN2=f_min(err_E_BN,derr_DE_SP);
MN3=f_min(err_E_MN,derr_DE_MN);
MN4=f_min(err_E_MN,derr_DE_SN);
MN5=f_min(err_E_MN,derr_DE_Z);
MN6=f_min(err_E_SN,derr_DE_MN);
MN7=f_min(err_E_Z,derr_DE_BN);
MN8=f_min(err_E_Z,derr_DE_MN);
MN9=f_min(err_E_SP,derr_DE_BN);
*MN=f_max2(MN1 ,MN2 ,MN3 ,MN4 ,MN5 ,MN6 ,MN7 ,MN8 ,MN9 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);

  // Rule base of SN
SN1=f_min(err_E_BN,derr_DE_MP);
SN2=f_min(err_E_MN,derr_DE_SP);
SN3=f_min(err_E_SN,derr_DE_SN);
SN4=f_min(err_E_SN,derr_DE_Z);
SN5=f_min(err_E_Z,derr_DE_SN);
SN6=f_min(err_E_SP,derr_DE_MN);
SN7=f_min(err_E_MP,derr_DE_BN);
*SN=f_max2(SN1 ,SN2 ,SN3 ,SN4 ,SN5 ,SN6 ,SN7 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);

  // Rule base of Z
Z1=f_min(err_E_BN,derr_DE_BP);
Z2=f_min(err_E_MN,derr_DE_MP);
Z3=f_min(err_E_SN,derr_DE_SP);
Z4=f_min(err_E_Z,derr_DE_Z);
Z5=f_min(err_E_SP,derr_DE_SN);
Z6=f_min(err_E_MP,derr_DE_MN);
Z7=f_min(err_E_BP,derr_DE_BN);
*Z=f_max2(Z1 ,Z2 ,Z3 ,Z4 ,Z5 ,Z6 ,Z7 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);


  // Rule base of SP
SP1=f_min(err_E_MN,derr_DE_BP);
SP2=f_min(err_E_SN,derr_DE_MP);
SP3=f_min(err_E_Z,derr_DE_SP);
SP4=f_min(err_E_SP,derr_DE_Z);
SP5=f_min(err_E_SP,derr_DE_SP);
SP6=f_min(err_E_MP,derr_DE_SN);
SP7=f_min(err_E_BP,derr_DE_MN);
```

```
*SP=f_max2(SP1 ,SP2 ,SP3 ,SP4 ,SP5 ,SP6 ,SP7 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);

  // Rule base of MP
MP1=f_min(err_E_SN,derr_DE_BP);
MP2=f_min(err_E_Z,derr_DE_MP);
MP3=f_min(err_E_Z,derr_DE_BP);
MP4=f_min(err_E_SP,derr_DE_MP);
MP5=f_min(err_E_MP,derr_DE_Z);
MP6=f_min(err_E_MP,derr_DE_SP);
MP7=f_min(err_E_MP,derr_DE_MP);
MP8=f_min(err_E_BP,derr_DE_SN);
MP9=f_min(err_E_BP,derr_DE_Z);
*MP=f_max2(MP1 ,MP2 ,MP3 ,MP4 ,MP5 ,MP6 ,MP7 ,MP8 ,MP9 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);


  // Rule base of LP
LP1=f_min(err_E_SP,derr_DE_BP);
LP2=f_min(err_E_MP,derr_DE_BP);
LP3=f_min(err_E_BP,derr_DE_SP);
LP4=f_min(err_E_BP,derr_DE_MP);
LP5=f_min(err_E_BP,derr_DE_BP);
*LP=f_max2(LP1 ,LP2 ,LP3 ,LP4 ,LP5 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);
}

void fuzzification3 (float inp , float *VAr1 , float *VAr2 , float *VAr3 , float *VAr4 , float *VAr5 , float *VAr6 , float *VAr7)
{
 *VAr1=0; *VAr2=0; *VAr3=0; *VAr4=0; *VAr5=0; *VAr6=0; *VAr7=0;
if(inp<= -1 ) *VAr1=1;
else if ((inp<=-0.8)&& (inp>=-1))
{
	*VAr1=-5 *inp +(-4);
	*VAr2=5*inp + (5);
}
else if ((inp<=-0.2)&& (inp>-0.8))
{
	*VAr2=-1.66666666666667 *inp +(-0.333333333333333);
	*VAr3=1.66666666666667*inp + (1.33333333333333);
}

else if ((inp<=0)&& (inp>-0.2))
{
	*VAr3=-5 *inp +(0);
	*VAr4=5*inp + (1);
}

else if ((inp<=0.2)&& (inp>0))
{
	*VAr4=-5 *inp +(1);
	*VAr5=5*inp + (0);
}

else if ((inp<=0.8)&& (inp>0.2))
{
	*VAr5=-1.66666666666667 *inp +(1.33333333333333);
	*VAr6=1.66666666666667*inp + (-0.333333333333333);
}

else if ((inp<=1)&& (inp>0.8))
{
	*VAr6=-5 *inp +(5);
	*VAr7=5*inp + (-4);
}

else if (inp>1) *VAr7=1;
}

void fuzzification4 (float inp , float *VAr1 , float *VAr2 , float *VAr3 , float *VAr4 , float *VAr5 , float *VAr6 , float *VAr7)
{
 *VAr1=0; *VAr2=0; *VAr3=0; *VAr4=0; *VAr5=0; *VAr6=0; *VAr7=0;
if(inp<= -1 ) *VAr1=1;
else if ((inp<=-0.8)&& (inp>=-1))
{
	*VAr1=-5 *inp +(-4);
	*VAr2=5*inp + (5);
}
else if ((inp<=-0.2)&& (inp>-0.8))
{
	*VAr2=-1.66666666666667 *inp +(-0.333333333333333);
	*VAr3=1.66666666666667*inp + (1.33333333333333);
}

else if ((inp<=0)&& (inp>-0.2))
{
	*VAr3=-5 *inp +(0);
	*VAr4=5*inp + (1);
}

else if ((inp<=0.2)&& (inp>0))
{
	*VAr4=-5 *inp +(1);
	*VAr5=5*inp + (0);
}

else if ((inp<=0.8)&& (inp>0.2))
{
	*VAr5=-1.66666666666667 *inp +(1.33333333333333);
	*VAr6=1.66666666666667*inp + (-0.333333333333333);
}

else if ((inp<=1)&& (inp>0.8))
{
```

```
        *VAr6=-5 *inp +(5);
        *VAr7=5*inp + (-4);
}
else if (inp>1) *VAr7=1;
}

// The RuleBase Part
float rulebase2 (float derr_DE_BN , float derr_DE_MN , float derr_DE_SN , float derr_DE_Z , float derr_DE_SP , float derr_DE_MP , float derr_DE_BP , float err_E_BN , float err_E_MN , float
err_E_SN , float err_E_Z , float err_E_SP , float err_E_MP , float err_E_BP , float *BN , float *MN , float *SN , float *Z , float *SP , float *MP , float *LP)
{
float Z1,Z2,Z3,Z4,Z5,Z6,Z7,Z8,Z9,Z10,Z11,Z12,Z13,Z14,Z15,Z16,Z17,Z18,Z19,Z20,Z21,Z22,Z23,Z24,Z25,Z26,Z27,Z28,Z29,Z30,Z31,Z32,Z33,Z34;
float SP1,SP2,SP3,SP4,SP5,SP6,SP7,SP8,SP9,SP10;
float LP1,LP2,LP3,LP4,LP5;

  // Rule base of BN
  // Rule base of MN
  // Rule base of SN
  // Rule base of Z
Z1=f_min(err_E_BN,derr_DE_BN);
Z2=f_min(err_E_BN,derr_DE_MN);
Z3=f_min(err_E_BN,derr_DE_SN);
Z4=f_min(err_E_BN,derr_DE_Z);
Z5=f_min(err_E_BN,derr_DE_SP);
Z6=f_min(err_E_BN,derr_DE_MP);
Z7=f_min(err_E_BN,derr_DE_BP);
Z8=f_min(err_E_MN,derr_DE_BN);
Z9=f_min(err_E_MN,derr_DE_MN);
Z10=f_min(err_E_MN,derr_DE_MP);
Z11=f_min(err_E_MN,derr_DE_BP);
Z12=f_min(err_E_SN,derr_DE_BN);
Z13=f_min(err_E_SN,derr_DE_MN);
Z14=f_min(err_E_SN,derr_DE_MP);
Z15=f_min(err_E_SN,derr_DE_BP);
Z16=f_min(err_E_Z,derr_DE_BN);
Z17=f_min(err_E_Z,derr_DE_MN);
Z18=f_min(err_E_Z,derr_DE_MP);
Z19=f_min(err_E_Z,derr_DE_BP);
Z20=f_min(err_E_SP,derr_DE_BN);
Z21=f_min(err_E_SP,derr_DE_MN);
Z22=f_min(err_E_SP,derr_DE_MP);
Z23=f_min(err_E_SP,derr_DE_BP);
Z24=f_min(err_E_MP,derr_DE_BN);
Z25=f_min(err_E_MP,derr_DE_MN);
Z26=f_min(err_E_MP,derr_DE_MP);
Z27=f_min(err_E_MP,derr_DE_BP);
Z28=f_min(err_E_BP,derr_DE_BN);
Z29=f_min(err_E_BP,derr_DE_MN);
Z30=f_min(err_E_BP,derr_DE_SN);
Z31=f_min(err_E_BP,derr_DE_Z);
Z32=f_min(err_E_BP,derr_DE_SP);
Z33=f_min(err_E_BP,derr_DE_MP);
Z34=f_min(err_E_BP,derr_DE_BP);
*Z=f_max2(Z1 ,Z2 ,Z3 ,Z4 ,Z5 ,Z6 ,Z7 ,Z8 ,Z9 ,Z10 ,Z11 ,Z12 ,Z13 ,Z14 ,Z15 ,Z16 ,Z17 ,Z18 ,Z19 ,Z20 ,Z21 ,Z22 ,Z23 ,Z24 ,Z25 ,Z26 ,Z27 ,Z28 ,Z29 ,Z30 ,Z31 ,Z32 ,Z33 ,Z34 , 0 , 0 , 0 , 0 , 0 , 0 ,
0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);

  // Rule base of SP
SP1=f_min(err_E_MN,derr_DE_SN);
SP2=f_min(err_E_MN,derr_DE_Z);
SP3=f_min(err_E_MN,derr_DE_SP);
SP4=f_min(err_E_SN,derr_DE_SN);
SP5=f_min(err_E_SN,derr_DE_SP);
SP6=f_min(err_E_SP,derr_DE_SN);
SP7=f_min(err_E_SP,derr_DE_SP);
SP8=f_min(err_E_MP,derr_DE_SN);
SP9=f_min(err_E_MP,derr_DE_Z);
SP10=f_min(err_E_MP,derr_DE_SP);
*SP=f_max2(SP1 ,SP2 ,SP3 ,SP4 ,SP5 ,SP6 ,SP7 ,SP8 ,SP9 ,SP10 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);

  // Rule base of MP
  // Rule base of LP
LP1=f_min(err_E_SN,derr_DE_Z);
LP2=f_min(err_E_Z,derr_DE_SN);
LP3=f_min(err_E_Z,derr_DE_Z);
LP4=f_min(err_E_Z,derr_DE_SP);
LP5=f_min(err_E_SP,derr_DE_Z);
*LP=f_max2(LP1 ,LP2 ,LP3 ,LP4 ,LP5 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);
}
```

#include"DAC.c"

```
 void output_voltage (float x_volt)
{
  float y_volt;
  unsigned err;
  //x_volt=-2.5;
  y_volt = 579.9*x_volt + 1767 ;
  err=y_volt;
  DAC1_Ch1_Output(err);
}

 float inpt_voltage (float inp_volt)
{
     inp_volt= 0.0006543243*inp_volt - 1.3783422372;
     return inp_volt;
}


 float inpt_power (float inp_power)
{
     inp_power= 0.003*inp_power;   // 1000MW =3V
     return inp_power;
}

 float Calc_Delta_sc (float SC,float slope,float y_int)
{
     //SC= 0.0006543243*SC - 1.3783422372;
     SC= slope *SC +  y_int;
     //SC = -0.00546*SC + 3.77583;

     return SC;
}
```

#include"simple_filter.c"

```
float M_avg( unsigned int * ARR , int length, unsigned int new_value)
{
unsigned int i,sum;
float fsum;
   sum=0;
   fsum=0;

       for (i=1;i<length ; i++)
          {
             sum = *( ARR+i);
             fsum+=sum;
             *(ARR+i-1)=*( ARR+i);
           }
     *(ARR+(length-1))=new_value;
     fsum += new_value;
     fsum/=length;
     return fsum;

}


float M_avg_f( float * ARR , int length, float new_value)
{
unsigned int i;
float sum;
float fsum;

   sum=0;
   fsum=0;

       for (i=1;i<length ; i++)
          {
             sum = *( ARR+i);
             fsum+=sum;
             *(ARR+i-1)=*( ARR+i);
           }
     *(ARR+(length-1))=new_value;
     fsum += new_value;
     fsum/=length;
     return fsum;
}
```

Appendix-D RTDS Fuzzy Model Code

Fuzzy Model

```
VERSION:
3.001

MODEL_TYPE: CTL

#define PI        3.14159265358979323846264338327795   // definition of PI
#define TWOPI     6.28318530717958647692528676655 9    // definition of 2.0*PI
#define E         2.71828182845904523536028747135266  // definition of E
#define EINV      0.36787944117144232159552377016147  // definition of E Inverse (1/E)
#define RT2       1.41421356237309504880168872420 97   // definition of square root 2.0
#define RT3       1.73205080756887729352744634150 59   // definition of square root 3.0
#define INV_ROOT2 0.70710678118654752440084436210485

INPUTS:

 double derr;
 double err;

OUTPUTS:

 double out;


STATIC:


RAM_FUNCTIONS:


RAM:


CODE_FUNCTIONS:


float f_min (float a, float b)
{
if (a>=b) return b;
else return a;
}


float defuzzification (float BN, float BN_w , float MN , float MN_w , float SN , float SN_w , float Z , float Z_w , float SP , float SP_w , float MP , float MP_w , float LP , float LP_w)


{
return (( BN* BN_w +MN*MN_w +SN*SN_w +Z*Z_w +SP*SP_w +MP*MP_w +LP*LP_w)/(BN +MN +SN +Z +SP +MP +LP));


}


float f_max2 (float a1, float a2, float a3, float a4, float a5, float a6, float a7, float a8, float a9, float a10, float a11, float a12 ,float a13, float a14, float a15, float a16, float a17, float a18, float a19, float a20, float a21, float a22, float a23, float a24, float a25, float a26, float a27, float a28, float a29, float a30, float a31, float a32, float a33, float a34, float a35, float a36, float a37, float a38, float a39, float a40, float a41, float a42, float a43, float a44, float a45, float a46, float a47, float a48, float a49)

{
float fmax;
fmax=a1;
if (a2>fmax) fmax=a2;
if (a3>fmax) fmax=a3;
if (a4>fmax) fmax=a4;
if (a5>fmax) fmax=a5;
if (a6>fmax) fmax=a6;
if (a7>fmax) fmax=a7;
if (a8>fmax) fmax=a8;
if (a9>fmax) fmax=a9;
if (a10>fmax) fmax=a10;
if (a11>fmax) fmax=a11;
if (a12>fmax) fmax=a12;
if (a13>fmax) fmax=a13;
if (a14>fmax) fmax=a14;
if (a15>fmax) fmax=a15;
if (a16>fmax) fmax=a16;
if (a17>fmax) fmax=a17;
if (a18>fmax) fmax=a18;
if (a19>fmax) fmax=a19;
if (a20>fmax) fmax=a20;
if (a21>fmax) fmax=a21;
if (a22>fmax) fmax=a22;
if (a23>fmax) fmax=a23;
if (a24>fmax) fmax=a24;
if (a25>fmax) fmax=a25;
if (a26>fmax) fmax=a26;
if (a27>fmax) fmax=a27;
if (a28>fmax) fmax=a28;
if (a29>fmax) fmax=a29;
if (a30>fmax) fmax=a30;
if (a31>fmax) fmax=a31;
```

```
if (a32>fmax) fmax=a32;
if (a33>fmax) fmax=a33;
if (a34>fmax) fmax=a34;
if (a35>fmax) fmax=a35;
if (a36>fmax) fmax=a36;
if (a37>fmax) fmax=a37;
if (a38>fmax) fmax=a38;
if (a39>fmax) fmax=a39;
if (a40>fmax) fmax=a40;
if (a41>fmax) fmax=a41;
if (a42>fmax) fmax=a42;
if (a43>fmax) fmax=a43;
if (a44>fmax) fmax=a44;
if (a45>fmax) fmax=a45;
if (a46>fmax) fmax=a46;
if (a47>fmax) fmax=a47;
if (a48>fmax) fmax=a48;
if (a49>fmax) fmax=a49;
return fmax;
}


void fuzzification1 (float inp , float *VAr1 , float *VAr2 , float *VAr3 , float *VAr4 , float *VAr5 , float *VAr6 , float *VAr7)


{
 *VAr1=0; *VAr2=0; *VAr3=0; *VAr4=0; *VAr5=0; *VAr6=0; *VAr7=0;

if(inp<= -1 ) *VAr1=1;

else if ((inp<=-0.7)&& (inp>=-1))
{
      *VAr1=-3.33333333333333 *inp +(-2.33333333333333);
      *VAr2=3.33333333333333*inp + (3.33333333333333);
}
else if ((inp<=-0.5)&& (inp>-0.7))

{
      *VAr2=-5 *inp +(-2.5);
      *VAr3=5*inp + (3.5);
}

else if ((inp<=0)&& (inp>-0.5))

{
      *VAr3=-2 *inp +(0);
      *VAr4=2*inp + (1);
}

else if ((inp<=0.5)&& (inp>0))

{
      *VAr4=-2 *inp +(1);
      *VAr5=2*inp + (0);

}

else if ((inp<=0.7)&& (inp>0.5))

{
      *VAr5=-5 *inp +(3.5);
      *VAr6=5*inp + (-2.5);

}

else if ((inp<=1)&& (inp>0.7))

{
      *VAr6=-3.33333333333333 *inp +(3.33333333333333);
      *VAr7=3.33333333333333*inp + (-2.33333333333333);

}

else if (inp>1) *VAr7=1;

}


void fuzzification2 (float inp , float *VAr1 , float *VAr2 , float *VAr3 , float *VAr4 , float *VAr5 , float *VAr6 , float *VAr7)


{
 *VAr1=0; *VAr2=0; *VAr3=0; *VAr4=0; *VAr5=0; *VAr6=0; *VAr7=0;

if(inp<= -1 ) *VAr1=1;

else if ((inp<=-0.7)&& (inp>=-1))
{
      *VAr1=-3.33333333333333 *inp +(-2.33333333333333);
      *VAr2=3.33333333333333*inp + (3.33333333333333);
}
else if ((inp<=-0.5)&& (inp>-0.7))

{
      *VAr2=-5 *inp +(-2.5);
      *VAr3=5*inp + (3.5);
}

else if ((inp<=0)&& (inp>-0.5))

{
```

```
        *VAr3=-2 *inp +(0);
        *VAr4=2*inp + (1);
}

else if ((inp<=0.5)&& (inp>0))

{
        *VAr4=-2 *inp +(1);
        *VAr5=2*inp + (0);

}

else if ((inp<=0.7)&& (inp>0.5))

{
        *VAr5=-5 *inp +(3.5);
        *VAr6=5*inp + (-2.5);

}

else if ((inp<=1)&& (inp>0.7))

{
        *VAr6=-3.33333333333333 *inp +(3.33333333333333);
        *VAr7=3.33333333333333*inp + (-2.33333333333333);

}

else if (inp>1) *VAr7=1;

}


// The RuleBase Part


float rulebase (float derr_DE_BN , float derr_DE_MN , float derr_DE_SN , float derr_DE_Z , float derr_DE_SP , float derr_DE_MP , float derr_DE_BP , float err_E_BN , float err_E_MN , float err_E_SN , float err_E_Z , float err_E_SP , float err_E_MP , float err_E_BP , float *BN , float *MN , float *SN , float *Z , float *SP , float *MP , float *LP)

{
float BN1,BN2,BN3,BN4,BN5,BN6,BN7,BN8,BN9;
float MN1,MN2,MN3,MN4,MN5;
float SN1,SN2,SN3,SN4,SN5,SN6;
float Z1,Z2,Z3,Z4,Z5,Z6,Z7,Z8,Z9;
float SP1,SP2,SP3,SP4,SP5,SP6;
float MP1,MP2,MP3,MP4,MP5;
float LP1,LP2,LP3,LP4,LP5,LP6,LP7,LP8,LP9;


  // Rule base of BN

BN1=f_min(err_E_BN,derr_DE_BN);
BN2=f_min(err_E_BN,derr_DE_MN);
BN3=f_min(err_E_BN,derr_DE_SN);
BN4=f_min(err_E_MN,derr_DE_BN);
BN5=f_min(err_E_MN,derr_DE_MN);
BN6=f_min(err_E_SN,derr_DE_BN);
BN7=f_min(err_E_SN,derr_DE_MN);
BN8=f_min(err_E_Z,derr_DE_BN);
BN9=f_min(err_E_SP,derr_DE_BN);

*BN=f_max2(BN1 ,BN2 ,BN3 ,BN4 ,BN5 ,BN6 ,BN7 ,BN8 ,BN9 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);


  // Rule base of MN

MN1=f_min(err_E_MN,derr_DE_SN);
MN2=f_min(err_E_SN,derr_DE_SN);
MN3=f_min(err_E_Z,derr_DE_MN);
MN4=f_min(err_E_SP,derr_DE_MN);
MN5=f_min(err_E_MP,derr_DE_BN);

*MN=f_max2(MN1 ,MN2 ,MN3 ,MN4 ,MN5 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);


  // Rule base of SN

SN1=f_min(err_E_BN,derr_DE_Z);
SN2=f_min(err_E_MN,derr_DE_Z);
SN3=f_min(err_E_Z,derr_DE_SN);
SN4=f_min(err_E_SP,derr_DE_SN);
SN5=f_min(err_E_MP,derr_DE_MN);
SN6=f_min(err_E_BP,derr_DE_BN);

*SN=f_max2(SN1 ,SN2 ,SN3 ,SN4 ,SN5 ,SN6 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);


  // Rule base of Z

Z1=f_min(err_E_BN,derr_DE_SP);
Z2=f_min(err_E_BN,derr_DE_MP);
Z3=f_min(err_E_MN,derr_DE_SP);
Z4=f_min(err_E_SN,derr_DE_Z);
Z5=f_min(err_E_Z,derr_DE_Z);
Z6=f_min(err_E_SP,derr_DE_Z);
Z7=f_min(err_E_MP,derr_DE_SN);
Z8=f_min(err_E_BP,derr_DE_MN);
Z9=f_min(err_E_BP,derr_DE_SN);

*Z=f_max2(Z1 ,Z2 ,Z3 ,Z4 ,Z5 ,Z6 ,Z7 ,Z8 ,Z9 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);
```

```
  // Rule base of SP

SP1=f_min(err_E_BN,derr_DE_BP);
SP2=f_min(err_E_MN,derr_DE_MP);
SP3=f_min(err_E_SN,derr_DE_SP);
SP4=f_min(err_E_Z,derr_DE_SP);
SP5=f_min(err_E_MP,derr_DE_Z);
SP6=f_min(err_E_BP,derr_DE_Z);

*SP=f_max2(SP1 ,SP2 ,SP3 ,SP4 ,SP5 ,SP6 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);


  // Rule base of MP

MP1=f_min(err_E_MN,derr_DE_BP);
MP2=f_min(err_E_SN,derr_DE_MP);
MP3=f_min(err_E_Z,derr_DE_MP);
MP4=f_min(err_E_SP,derr_DE_SP);
MP5=f_min(err_E_MP,derr_DE_SP);

*MP=f_max2(MP1 ,MP2 ,MP3 ,MP4 ,MP5 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);


  // Rule base of LP

LP1=f_min(err_E_SN,derr_DE_BP);
LP2=f_min(err_E_Z,derr_DE_BP);
LP3=f_min(err_E_SP,derr_DE_MP);
LP4=f_min(err_E_SP,derr_DE_BP);
LP5=f_min(err_E_MP,derr_DE_MP);
LP6=f_min(err_E_MP,derr_DE_BP);
LP7=f_min(err_E_BP,derr_DE_SP);
LP8=f_min(err_E_BP,derr_DE_MP);
LP9=f_min(err_E_BP,derr_DE_BP);

*LP=f_max2(LP1 ,LP2 ,LP3 ,LP4 ,LP5 ,LP6 ,LP7 ,LP8 ,LP9 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0);


}

CODE:



// VAriable def. Part

float derr_scale,derr_DE_BN,derr_DE_MN,derr_DE_SN,derr_DE_Z,derr_DE_SP,derr_DE_MP,derr_DE_BP;
float err_scale,err_E_BN,err_E_MN,err_E_SN,err_E_Z,err_E_SP,err_E_MP,err_E_BP;
float out_scale,out_BN,out_MN,out_SN,out_Z,out_SP,out_MP,out_LP;


fuzzification1 (derr,&derr_DE_BN,&derr_DE_MN,&derr_DE_SN,&derr_DE_Z,&derr_DE_SP,&derr_DE_MP,&derr_DE_BP);
fuzzification2 (err,&err_E_BN,&err_E_MN,&err_E_SN,&err_E_Z,&err_E_SP,&err_E_MP,&err_E_BP);
rulebase(derr_DE_BN,derr_DE_MN,derr_DE_SN,derr_DE_Z,derr_DE_SP,derr_DE_MP,derr_DE_BP,err_E_BN,err_E_MN,err_E_SN,err_E_Z,err_E_SP,err_E_MP,err_E_BP,&out_BN,&out_MN,
&out_SN,&out_Z,&out_SP,&out_MP,&out_LP);
out=defuzzification (out_BN,-1 ,out_MN,-0.7 ,out_SN,-0.5 ,out_Z,0 ,out_SP,0.5 ,out_MP,0.7 ,out_LP,1);
```

Differentiation model

```
VERSION:
3.001

// Include file below is generated by C-Builder
// and contains the VAriables declared as -
// PARAMETERS, INPUTS, OUTPUTS . . .
#include "diff.h"


STATIC:




// ---------------------------------------------------
// VAriables declared here may be used in both the
// RAM: and CODE: sections below.
// ---------------------------------------------------
//    double dt;
                    double old_data;
                    double new_data;


// - E n d  o f  S T A T I C :  S e c t i o n -

RAM_FUNCTIONS:

// ---------------------------------------------------
// This section should contain any 'c' functions
// to be called from the RAM section (either
// RAM_PASS1 or RAM_PASS2). Example:
//
// static double myFunction(double v1, double v2)
// {
//     return(v1*v2);
// }
// ---------------------------------------------------

RAM:

// ---------------------------------------------------
// Place C code here which computes constants
// required for the CODE: section below.  The C
// code here is executed once, prior to the start
// of the simulation case.
// ---------------------------------------------------
//    dt= getTimeStep();
new_data=0;
old_data=0;

// ---- E n d  o f  R A M :  S e c t i o n ----

CODE:

// ---------------------------------------------------
// Place C code here which runs on the RTDS. The
// code below is entered once each simulation
// step.
// ---------------------------------------------------
new_data=IN;

out=new_data-old_data;
old_data=new_data;

// ---- E n d  o f  C O D E :  S e c t i o n ---
```

Max Min Model

```
VERSION:
3.001

// Include file below is generated by C-Builder
// and contains the VAriables declared as -
// PARAMETERS, INPUTS, OUTPUTS . . .


MODEL_TYPE: CTL

#define PI        3.14159265358979323846264338327795   // definition of PI
#define TWOPI     6.28318530717958647692528676655 9    // definition of 2.0*PI
#define E         2.71828182845904523536028747135266  // definition of E
#define EINV      0.36787944117144232159552377016147  // definition of E Inverse (1/E)
#define RT2       1.41421356237309504880168872420 97   // definition of square root 2.0
#define RT3       1.73205080756887729352744634150 59   // definition of square root 3.0
#define INV_ROOT2 0.70710678118654752440084436210485

INPUTS:
  double IN;
  double MAXI;
  double MINI;

OUTPUTS:
  double out;


STATIC:




// ---------------------------------------------------
// VAriables declared here may be used in both the
// RAM: and CODE: sections below.
// ---------------------------------------------------
//    double dt;
                  double old_data;
                  double new_data;


// - E n d  o f  S T A T I C :  S e c t i o n -



RAM_FUNCTIONS:

// ---------------------------------------------------
// This section should contain any 'c' functions
// to be called from the RAM section (either
// RAM_PASS1 or RAM_PASS2). Example:
//
// static double myFunction(double v1, double v2)
// {
//     return(v1*v2);
// }
// ---------------------------------------------------

RAM:

// ---------------------------------------------------
// Place C code here which computes constants
// required for the CODE: section below.  The C
// code here is executed once, prior to the start
// of the simulation case.
// ---------------------------------------------------
//    dt= getTimeStep();
new_data=0;
old_data=0;

// ---- E n d  o f  R A M :  S e c t i o n ----

CODE:

// ---------------------------------------------------
// Place C code here which runs on the RTDS. The
// code below is entered once each simulation
// step.
// ---------------------------------------------------
new_data=IN;

if(new_data>MAXI)
                  new_data=old_data;
else if (new_data<MINI)
                  new_data=old_data;
else
                  new_data=new_data;

out=new_data;
old_data=new_data;

// ---- E n d  o f  C O D E :  S e c t i o n ---
```

Appendix-E RTDS Model