

Dynamic Modeling of Critical Load Stabilizer

ECE FYP Final Report

Project Name: Dynamic modeling of critical load stabilizer

Project ID: TD1a-18

Supervisor: Professor Danny Tsang

Author: KAPUR, Akshay

Student ID: 

Date: 15th April 2019

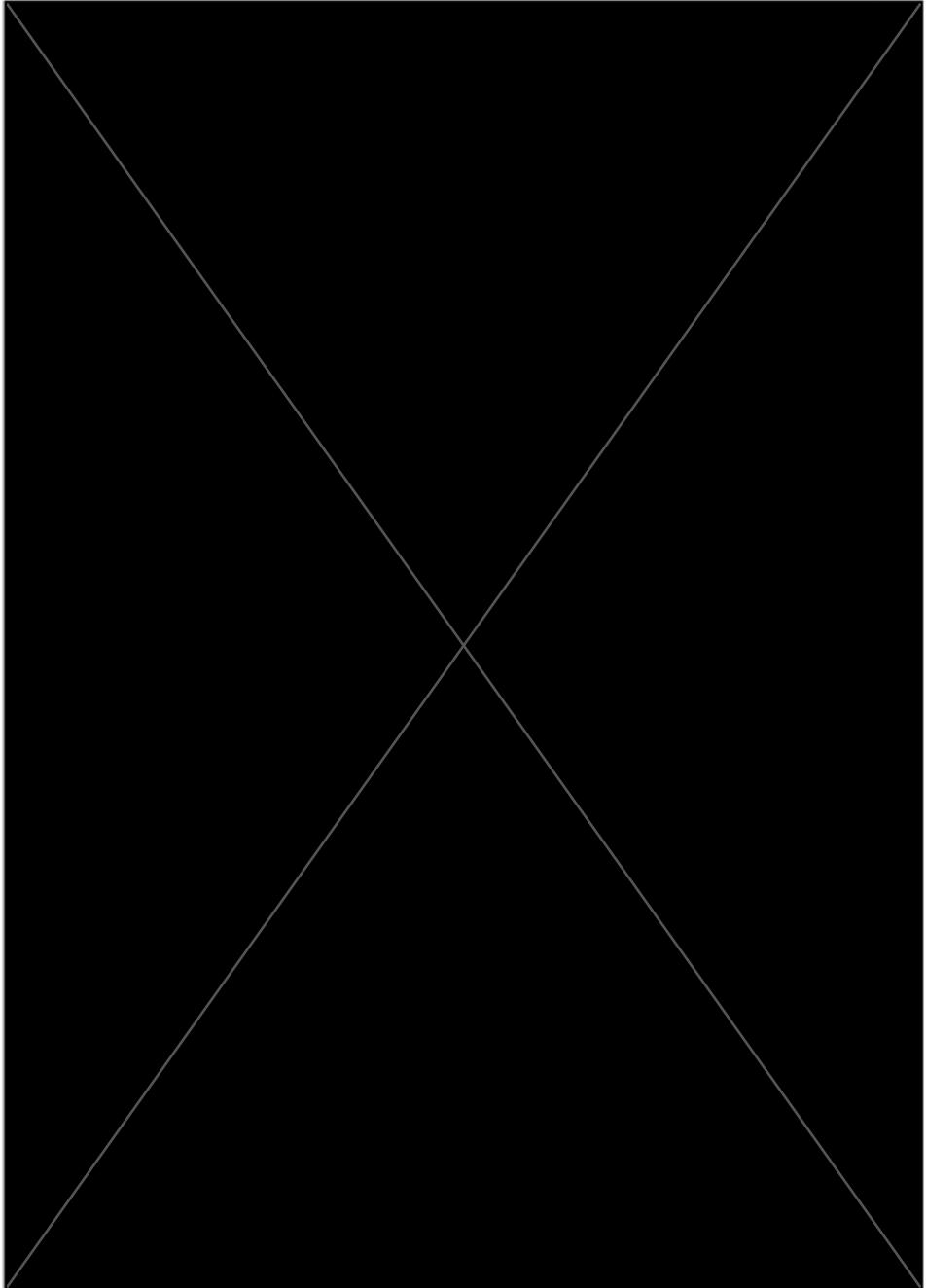
ACKNOWLEDGEMENT

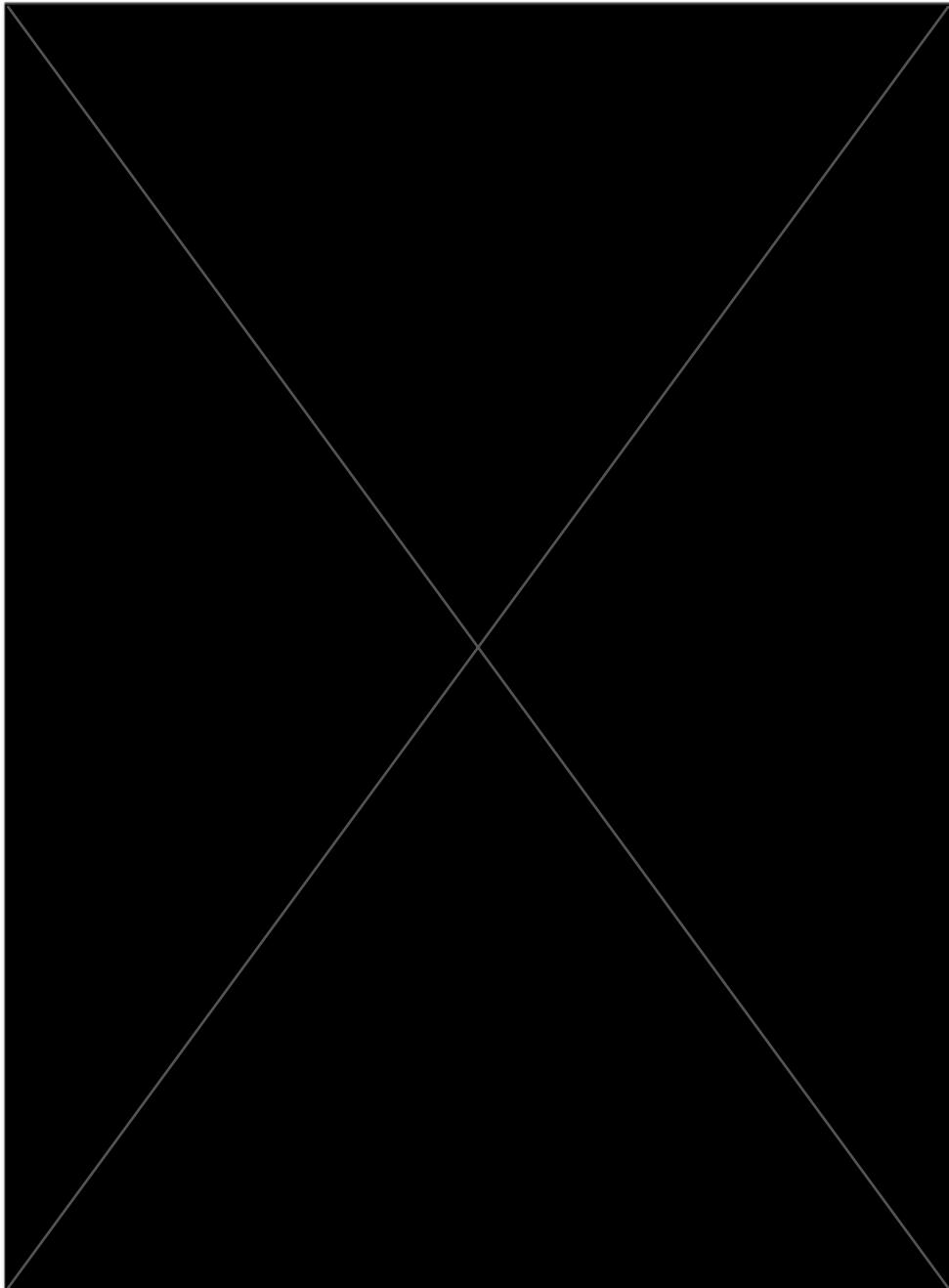
We would like to express immense pleasure by thanking everyone who has played a crucial role in helping us complete our project and this report. We are particularly grateful for the assistance given by our final year project supervisor, Professor Danny Tsang. His valuable and constructive suggestions during the planning and development of this project are greatly appreciated. We also wish to acknowledge the help provided by Ms. Tamie Konstas, who perused through different versions of this report and provided productive feedback.

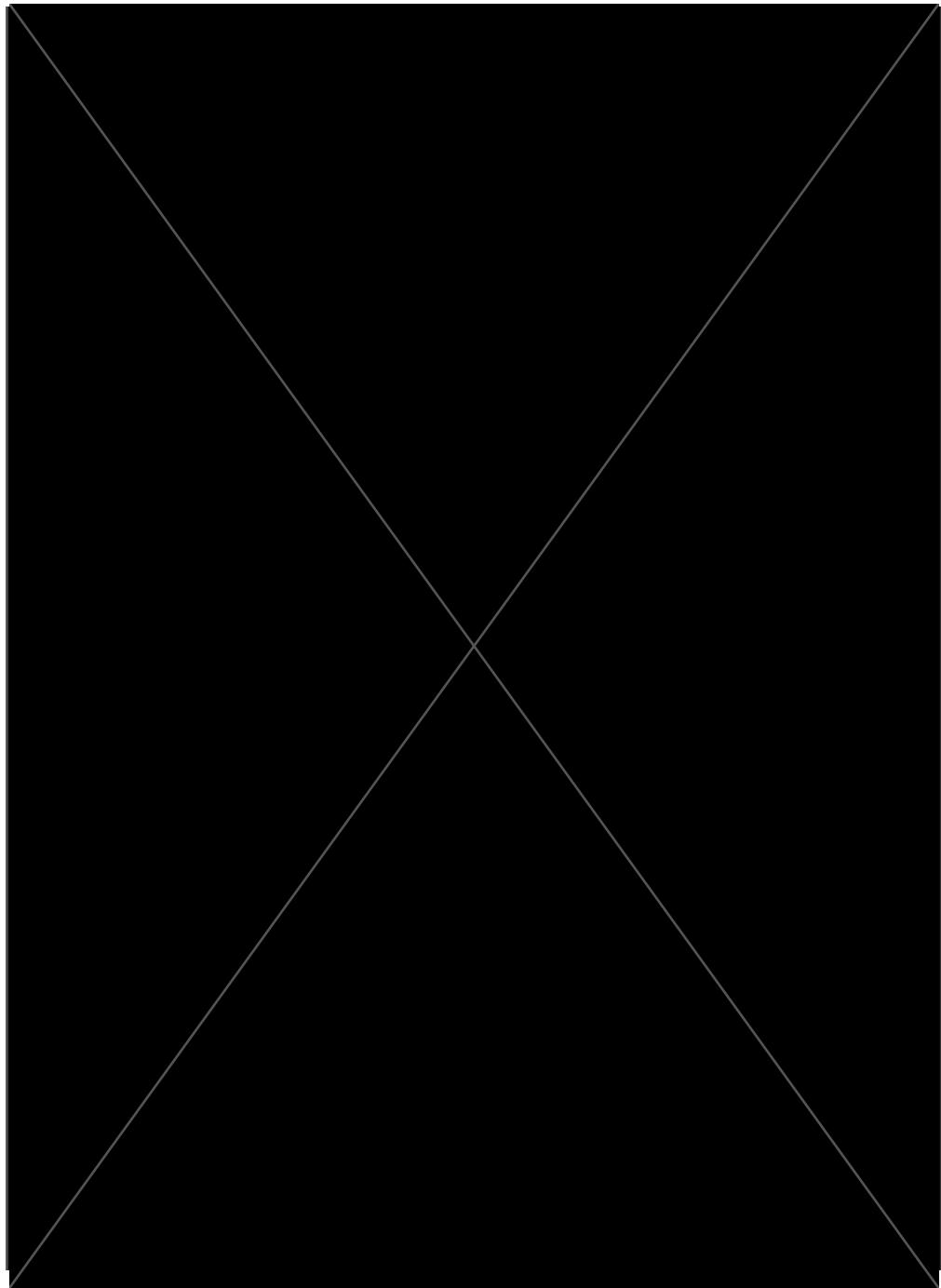
Finally, we would like to offer our special thanks to Mr. Shuwei Cao, PHD candidate in the Department of ECE at HKUST whose help and advice have played a major role in completion of this project.

ABSTRACT

Climate change and increasing electricity demand has called for the need of renewable energy sources. However, the intermittent nature of renewable energy sources has made it difficult to adopt this new shift. This requires a solution to stabilize grid voltage and protect critical load devices. This report introduces a simulation model of a critical load stabilizer that operates in response to grid voltage fluctuations. It converts the non-critical load into a smart load making it follow power generation thus reversing the traditional load-following control paradigm. The proposed model is built in MATLAB Simulink and its results are presented and evaluated in this report. Three different designs of critical load stabilizer are tested – pure reactive model, active-reactive model and branched-stabilizer model.







v



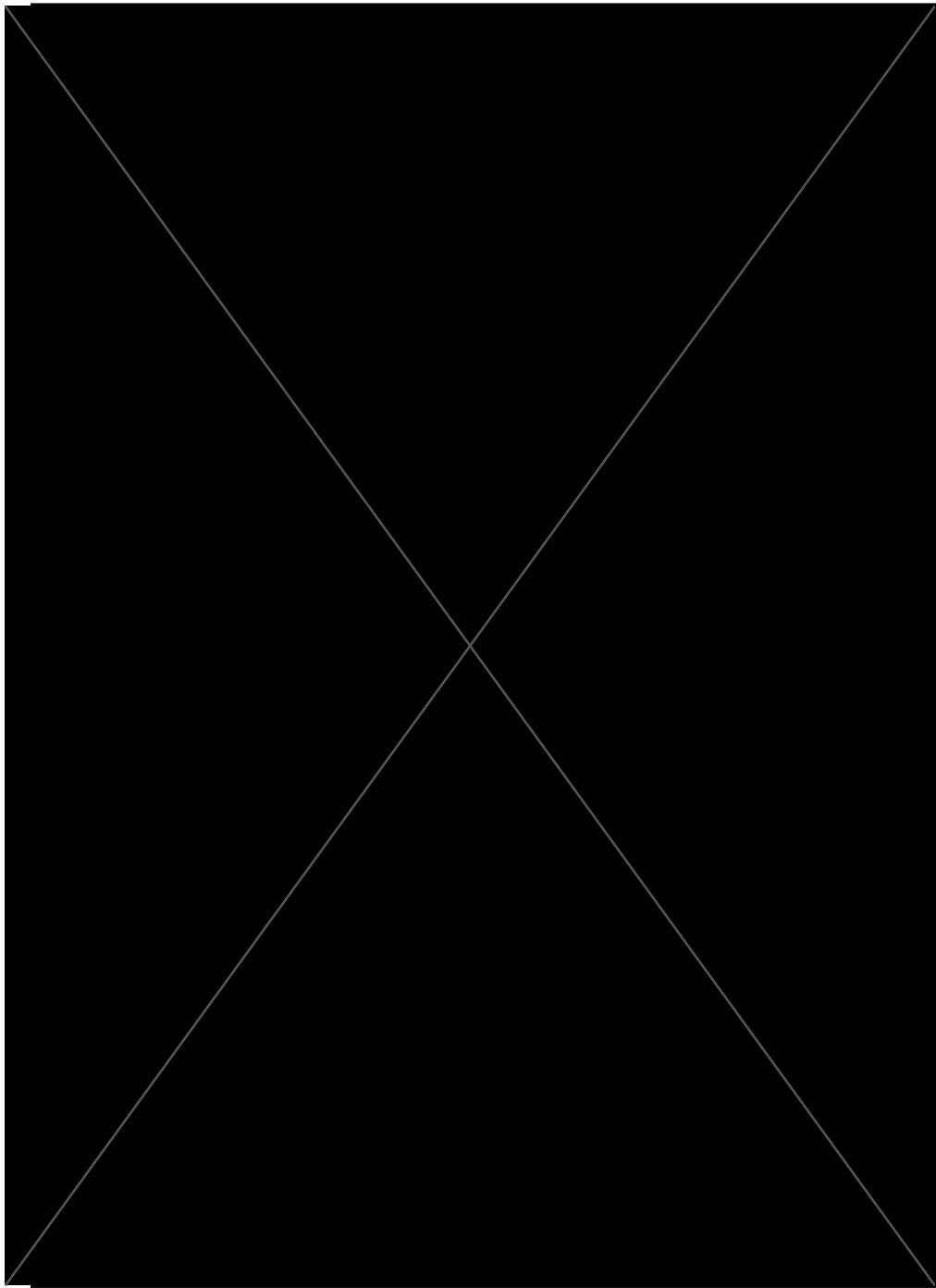




TABLE OF CONTENTS

ACKNOWLEDGEMENTS.....	i
ABSTRACT.....	ii
LIST OF FIGURES.....	x
LIST OF TABLES.....	xi
INTRODUCTION SECTION.....	1
CHAPTER 1 – INTRODUCTION.....	1
1.1 Project Introduction.....	1
1.1.1 Introduction.....	1
1.1.2 Literature Review.....	1
1.1.3 Group Project Objective.....	3
1.1.4 Sub-project Objective.....	3
1.2 Group Project Details.....	3
1.2.1 Group Project Description.....	3
1.2.2 Schedule of Responsibilities.....	4
1.3 Sub-Project Details.....	4
1.3.1 Sub-project Description.....	4
1.3.2 Components (hardware and/or software).....	4
1.3.3 System Block Diagram(s).....	5
1.3.4 Sub-Project Tasks.....	6
1.3.5 Sub-Project Schedule.....	6
1.3.6 Expected Technical Challenges.....	6
1.3.7 Budget.....	7
1.4 Report Outline.....	7
METHODOLOGY SECTION.....	8
CHAPTER 2 – DESIGN ANALYSIS AND TECHNICAL BACKGROUND.....	8
2.1 Project Design Decisions.....	8
2.2 Technical Background.....	9
2.2.1 Proportional-Integral controller.....	10
2.2.2 Phase locked loop.....	18
2.2.3 LC low pass filter.....	20
CHAPTER 3 – MATHEMATICAL MODEL OF CRITICAL LOAD STABILIZER.....	22
3.1 Circuit analysis with no Renewable energy source.....	22
3.2 Circuit analysis with Renewable energy source.....	25
3.3 Circuit analysis with Black box.....	27
3.4 Circuit analysis with Critical load stabilizer.....	30
CHAPTER 4 – IMPLEMENTING PI CONTROLLER IN MATLAB SIMULINK.....	33
CHAPTER 5 – IMPLEMENTING PHASE-LOCKED LOOP IN MATLAB SIMULINK.....	36
CHAPTER 6 – IMPLEMENTING LOW PASS FILTER IN MATLAB SIMULINK.....	38
CHAPTER 7 – GROUP PROJECT IMPLEMENTATION.....	41

GROUP PROJECT RESULTS & EVALUATION SECTIONS.....	44
CHAPTER 8 – GROUP PROJECT RESULTS & EVALUATION.....	44
8.1 Results and Evaluation.....	44
8.2 Future Improvements.....	48
CHAPTER 9 – CONCLUSION.....	50
REFERENCES.....	51

LIST OF FIGURES

Figure 1: System block diagram.....	5
Figure 2: Grid circuit with critical load stabilizer.....	8
Figure 3: Feedback control loop.....	10
Figure 4: Block diagram of Proportional controller.....	10
Figure 5 : Significance of K_p on controller output.....	11
Figure 6: Physical understanding of K_p	12
Figure 7 : Block diagram of P controller with 2 nd order system.....	12
Figure 8: Simulink model of P controller with 2 nd order system.....	14
Figure 9: Simulink output of process variable for different values of K_p	14
Figure 10: Magnified version of Simulink output of process variable.....	14
Figure 11: Block diagram of Integral controller.....	15
Figure 12: Input-output relationship of Integral controller.....	16
Figure 13: Block diagram of PI controller.....	16
Figure 14: Block diagram of PI controller with 2nd order system.....	17
Figure 15: Phase difference detection in PLL.....	19
Figure 16: Block diagram of PLL system.....	19
Figure 17: Circuit diagram of LC low pass filter.....	20
Figure 18: Bode diagram of LC low pass filter.....	21
Figure 19: Grid circuit diagram with no renewable energy.....	22
Figure 20: Simulink model of grid with no renewable energy source.....	24
Figure 21: Simulink results of grid circuit model with no renewable energy source.....	24
Figure 22: Grid circuit diagram with renewable energy source.....	25
Figure 23: Simulink model of grid with renewable energy source.....	26
Figure 24: Simulink results of grid circuit model with renewable energy source.....	27
Figure 25: Grid circuit diagram with black box.....	27
Figure 26: Simulink model of grid with resistor-capacitor black box.....	29
Figure 27: Simulink results of grid circuit model with resistor-capacitor black box.....	29
Figure 28: Grid circuit diagram with critical load stabilizer.....	30
Figure 29: Simulink model of grid with critical load stabilizer.....	31
Figure 30: Simulink results of grid circuit model with critical load stabilizer.....	32
Figure 31: Simulink model of PI controller used in critical load stabilizer.....	33
Figure 32: Input of PI controller with $K_p = 50$ and $K_I = 0$	34
Figure 33: Output of PI controller with $K_p = 50$ and $K_I = 0$	34
Figure 34: Input of PI controller with $K_p = 50$ and $K_I = 5$	35
Figure 35: Output of PI controller with $K_p = 50$ and $K_I = 5$	35
Figure 36: Simulink model of PLL.....	36
Figure 37: Simulink PLL block parameters.....	36
Figure 38: Simulink output of PLL block.....	37
Figure 39: Circuit diagram of half-bridge inverter and low pass filter.....	38
Figure 40: Simulink model of LC low pass filter.....	39
Figure 41: Input of LC low pass filter.....	40
Figure 42: Output of LC low pass filter.....	40

Figure 43: Simulink model of renewable energy source.....	41
Figure 44: Power profile of renewable energy source.....	42
Figure 45: Grid voltage fluctuations due to renewable energy source.....	42
Figure 46: Simulink model of grid circuit with PV penetration and critical load stabilizer.....	43
Figure 47: Simulink model of critical load stabilizer.....	43
Figure 48: Power profile of renewable energy source under inductive mode.....	44
Figure 49: Critical load voltage under inductive mode.....	44
Figure 50: Power profile of renewable energy source under capacitive mode.....	45
Figure 51: Critical load voltage under capacitive mode.....	45
Figure 52: Graph displaying capacitance vs critical load voltage.....	47
Figure 53: Critical load voltage under inductive mode (active-reactive stabilizer).....	47
Figure 54: Critical load voltage under capacitive mode (active-reactive stabilizer).....	48
Figure 55: Critical load voltage under capacitive mode (branched reactive stabilizer).....	49

LIST OF TABLES

Table 1: Group project Gantt chart.....	4
Table 2: Sub-project Gantt chart.....	6

INTRODUCTION SECTION

CHAPTER 1 – INTRODUCTION

1.1 Project Introduction

1.1.1 Introduction

All over the world, climate change is a huge concern. With global sea levels rising between 16-20 cm since 1900, climate change is undeniably influencing oceans, land services, ice caps and weather pattern across the globe [1]. It is common knowledge that climate change is caused by atmospheric gases such as Carbon dioxide and Methane. Tracing these gases reveal an obvious correlation between human activity and climate change. To tackle the problem of climate change, huge investments are being made in the field of alternative energy sources, specifically in renewable energy sources. Increasing population and high demand are also major contributors to the need of renewable energy sources. Renewable energy is generated from sources that naturally replenish themselves and do not run out. The most common sources are solar, hydro, wind, geothermal and biomass [2]. These do not produce any greenhouse gas emissions during production therefore do not contribute to climate change or pollution. Due to their replenishable characteristic, they are a reliable source of energy. However, they do have some disadvantages. Renewable energy sources, especially solar and wind, are intermittent in nature. This means that their electricity generation capacity is highly unpredictable and is dependent on the intensity of sunshine or wind speed.

These renewable energy sources are built and interconnected with the existing transmission lines, producing and injecting electricity directly onto the grid system. The traditional power grid system works under the control paradigm where power generation follows power demand. Variable weather conditions make it complicated to calculate how much output from renewable energy sources are needed to follow the above-mentioned control paradigm of load following. Rapid fluctuations in output of solar or wind energy sources contribute to unstable grid voltage ultimately damaging the grid system. Fluctuating grid voltage can further damage the critical load installed at home such as desktop computers and servers. Thus, the unstable power supply from wind and solar energy sources have raised concerns about grid power system stability [3]. Several researchers have led studies to improve the unsteady characteristic of renewable energy sources and reverse the control paradigm of load following. Despite the numerous efforts and solutions put forward by the Power Electronics community, the problem still persists and requires further research and development.

This report introduces a research field which has huge potential in solving the issue of unstable power system due to sporadic nature of renewable energy source. This helps to change the current fossil fuel-powered grid system to a hybrid renewable energy-based grid system which is crucial to tackle climate change and increasing electricity demand. In this report, a simulation model of a critical load stabilizer is designed which is capable of stabilizing the fluctuations in grid voltage caused by intermittent renewable energy source and thus protecting the critical load. It also reverses the current control paradigm of load following to demand-side management.

1.1.2 Literature Review

To address the issue of increasing energy demand and intermittent nature of renewable energy source, several solutions have been implemented by the utility companies. Some of these solutions will be

discussed and analyzed in this chapter to understand the benefits and weaknesses associated with each one of them.

1. Direct Load Control

Direct Load Control (DLC) programs have been around since the 1970s and are one of the most common demand response programs. DLC demand response is opted by the utility companies as a form of load management control. DLC technology allows the utility companies to remotely manage the electricity demand by accessing and modifying the consumer devices – typically pool pumps, air conditioner and electric hot water systems. Usually, DLC programs allow the utility company to install equipment that helps them to control specific appliances during peak hours. Using the “remote appliance controller”, the companies can switch certain appliances on and off for short period of time to alleviate high electricity demand. In return for their participation, the companies usually offer financial incentives to customers such as electricity bill discount, one-off payment, ongoing annual payment or free hardware installations. Despite the numerous benefits offered by the utility companies, the consumer uptake for opting DLC technology has been extremely low. For instance, a survey conducted by the Federal Energy Regulatory Commission in 2012 reported that the customer enrollment in DLC program was a mere 0.11% in Texas Reliability Entity Region [4]. The consumers are unwilling to participate for various reasons such as concerns over disruption in daily lifestyle, lack of control and distrust between the utility companies and consumers.

2. Dynamic pricing

Dynamic pricing is a demand management technique used by the utility companies where they charge different prices at different times according to demand. Since the cost of generation to meet peak demand is high, therefore, the companies tend to charge higher prices at peak time compared to off-peak times. This information is useful for customers in monetary savings as they can shift their high energy consumption tasks to off-peak times. The utility company also find this extremely useful as it is highly feasible and results in reduced peak energy consumption and better planned operations. However, the drawback of this technique is that it might leave customers irritated. The customer will have to manually keep track of prices every day and manage their daily chores accordingly. The utility company might also face the problem of load synchronization where the electricity demand is moved from a typical peak hour to non-peak hour. This would lead to additional price change and the customer would need to keep track of the prices again. In some cases, the customers might stop keeping track of dynamic pricing and use their appliances according to their own need. This would lead to no change in peak hour demand and would make this technique completely useless.

3. Energy storage device

Due to intermittent nature of renewable energy, it is a difficult task to estimate the energy generation. To tackle this situation, an energy storage device is installed near the renewable energy generator to store the generated energy for future use. During days with high wind speed or sunshine, surplus energy is generated and can be stored in the storage device. On other days when energy generation is not enough, this extra energy stored in the storage device can be used to meet the needs of the consumers. However, high cost of installing energy storage devices reduce the profit margin of the utility companies. Integrating energy storage devices require additional infrastructure and space as well as complex energy tracking technologies. Therefore, these drawbacks make it extremely hard for the companies to opt for energy storage devices as a solution to intermittent nature of renewable energy sources.

The solutions mentioned above possess their shortcomings: Direct Load Control raise concern among customers over lack of control, Dynamic pricing relies heavily on the customer's participation and energy storage devices are extremely costly and complicated to install. In this report, a low-cost, accessible solution is proposed to stabilize the fluctuations in grid voltage and resolve the issue of demand side management. A simulation model of a Critical Load Stabilizer is designed that can be installed on the consumer side to provide stable energy from the grid and protect the critical load devices.

1.1.3 Group Project Objective

In this project, a simulation model of critical load stabilizer is designed to stabilize the fluctuating grid voltage due to intermittent nature of renewable energy source. It will convert non-critical load into a smart load and reverse the control paradigm by making non-critical load adjust to fluctuations in the grid voltage. This device solves all the problems of demand side management and intermittent behavior of renewable energy without any shortcomings possessed by other alternatives.

1.1.4 Sub-project Objective

In this sub-project, various sub-systems crucial in building critical load stabilizer are designed. These sub-systems include – Proportional-Integral (PI) compensation controller, Phase-locked loop (PLL) system and low pass filter. A PI controller with fast response and low steady state error is designed to minimize voltage error on critical load. A Phase-locked loop system is designed to record the phase of non-critical load branch current and provide maximum reactive power compensation. Finally, a low pass filter is designed to inject a smooth sinusoidal wave onto the critical load stabilizer capacitor for voltage stability.

1.2 Group Project Details

1.2.1 Group Project Description

In this project, a simulation model of Critical Load Stabilizer is developed in MATLAB Simulink, which can stabilize any voltage and frequency fluctuations produced by the utility grid, thus protecting devices at home. It is divided into 5 sub-systems – Compensation controller, Synchronization Network, PWM generator, PWM inverter and Output filter. The grid voltage is measured and fed into the compensation controller to minimize critical load voltage error and produce a controlled output. The frequency and phase of non-critical load current is measured by Synchronization Network.

The output of compensation controller and synchronization network are used to calculate additional circuit parameters. The PWM generator outputs a custom PWM signal which drives the PWM inverter to produce the necessary voltage for stabilization. Finally, the output filter takes input from the PWM inverter and outputs a smooth sinusoidal wave similar to AC grid voltage. All the above sub-systems work together to provide active-reactive compensation for grid stability. Figure 1 clearly depicts the relationship of each sub-system of critical load stabilizer with each other. Further details of each sub-system are explained in later chapters.

1.2.2 Schedule of Responsibilities

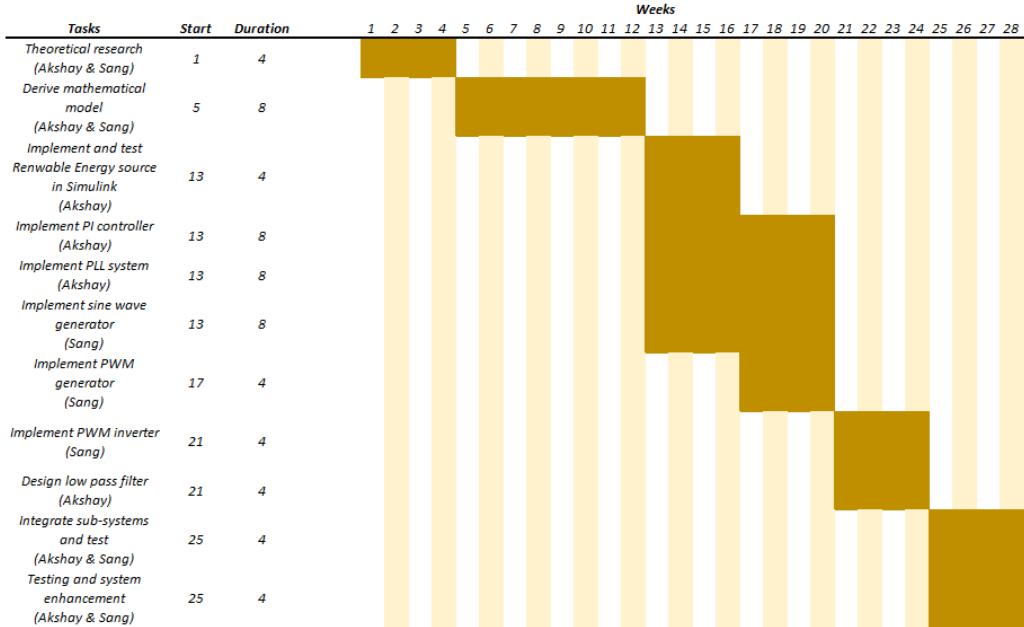


Table 1: Group project schedule Gantt chart

1.3 Sub-Project Details

1.3.1 Sub-project Description

In this sub-project, various sub-systems of the critical load stabilizer are implemented in MATLAB Simulink. First, a PI controller is tuned and implemented to drive critical load voltage error to zero. It is the main control system of critical load stabilizer and ensures that no error is present. The inputs to compensation controller are critical load voltage and a reference voltage which is desired on the critical load. The next subsystem implemented in this sub-project is PLL system. It is the most important system to ensure reactive power compensation as it calculates the phase of non-critical load branch current and ensures the critical load stabilizer's voltage is either leading or lagging the branch current by $\frac{\pi}{2}$. The PI controller and PLL block together calculate the parameters for sinusoidal wave responsible for driving PWM inverter.

At last, the output low pass filter is implemented using a LC combination. It discriminates against high frequency harmonic components and outputs a low frequency sinusoidal wave. The low pass filter is connected to output of PWM inverter and acts as the final stage of critical load stabilizer. Chapter 2 describes the fundamentals of each of these sub-systems to gain better understanding of their implementation. Further chapters will discuss implementation details and evaluate the results from each of these subsystems. Finally, all the subsystems are connected, and final results of the critical load stabilizer are evaluated.

1.3.2 Components

1. Standard PC/laptop with the following specifications:

- a. Processor – dual core 2.4 GHz+
- b. RAM – 8 GB+
- c. Hard Drive – 256 GB+
- d. Operating System – Windows 8.1+

2. MATLAB
3. Simulink

1.3.3 System Block Diagram

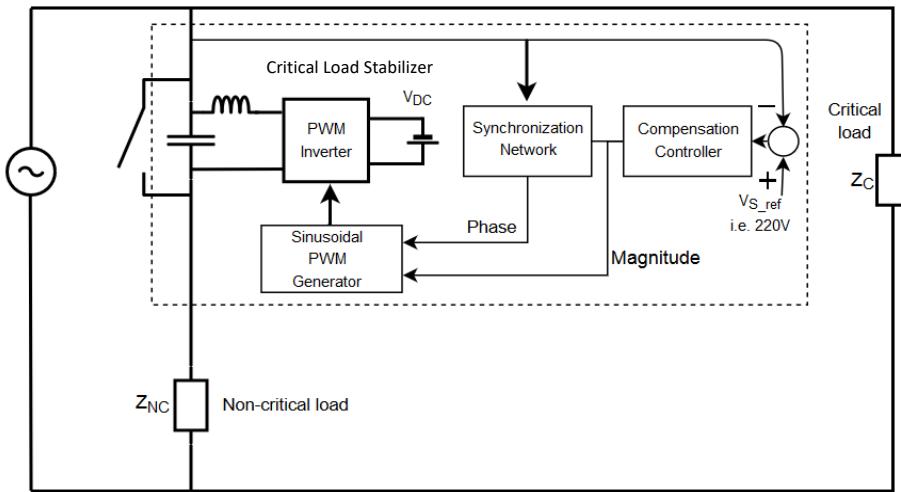


Figure 1: System block diagram

- The critical load error voltage is input to the compensation controller.
- The controller output is used to calculate parameters of the sinusoidal wave.
- The synchronization block measures the phase of non-critical load current and performs calculations on it to produce phase angle for sinusoidal wave.
- The sinusoidal PWM generator generates a high frequency PWM signal based on the custom sinusoidal wave.
- The PWM signal operates PWM inverter to translate the DC link voltage v_{DC} to output of the inverter as AC voltage.
- The low pass filter finally produces a low frequency sinusoid signal from output of the PWM inverter.
- The low frequency sinusoidal signal produced by low pass filter stabilizes voltage on the critical load.

A detailed description for each of the control blocks will be given in the upcoming chapters.

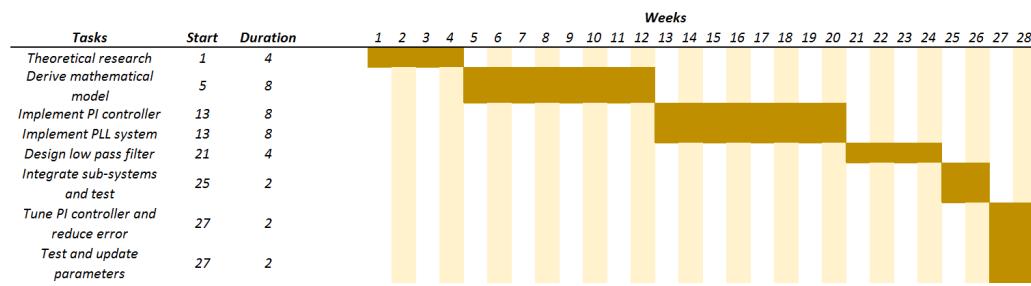
1.3.4 Sub-project Tasks

This sub-section will describe major tasks involved in this sub-project. The tasks to create sub-system simulation models are as follows:

1. Create a mathematical model of the critical load stabilizer.
2. Implement a PI controller for the above plant model.
3. Implement a PLL block and evaluate the results.
4. Design a low pass LC filter with appropriate cutoff frequency.
5. Integrate all sub-systems to test the critical load stabilizer.
6. Observe the error and tune PI controller parameters.
7. Tweak other necessary circuit parameters such as DC link voltage, PWM frequency and magnitude of sinusoidal wave.
8. Test and evaluate the final results.

1.3.5 Sub-Project Schedule

Table 2 shows the schedule of tasks.



1.3.6 Technical Challenges

Due to complexity of the system, a linear model could not be derived. As there were no transfer functions for the system, tuning the PI controller was a challenging task. Ziegler-Nichols method and Cohen-Coon method were used to reach an initial point for controller parameters. Thereafter, manual tuning was performed by looking at the controller response and tweaking the appropriate tuning parameter. For example, if output produced a steady-state error, then value of K_I (*integral constant*) was changed. Similarly, if output response was too slow, then value of K_p (*proportional constant*) was tweaked. Although this was a challenging task, but it helped to attain an intuitive understanding of PI controller. Reading several papers on control theory eased the process of controller design. Chapter 2 describes fundamentals of PI controller and is important in gaining a clearer understanding of the controller. Chapter 4 will explain the implementation of PI controller in MATLAB Simulink and evaluate the results.

The other challenge faced while designing critical load stabilizer was the derivation of mathematical model. Intensive knowledge of circuit design was required to create the model. Our PhD mentor, Shuwei Cao, helped us create this mathematical model which was the foundation of critical load stabilizer. Chapter 3 explains in detail the entire mathematical model along with its important derivations.

1.3.7 Budget

All the materials were readily available at HKUST and were used at no cost.

1.4 Report Outline

Chapter 2 will discuss the design decisions made. It also explains fundamental knowledge needed to understand the design of critical load stabilizer. Chapter 3 explains how the mathematical model of critical load stabilizer was derived. All the necessary equations are derived and explained as well. Chapter 4 explains how to implement the relationships derived in chapter 3 on MATLAB Simulink. It explains implementation and tuning of PI controller in MATLAB Simulink and evaluates the results. Chapter 5 implements the PLL system in Simulink and evaluates the results. Chapter 6 explains the design of LC low pass filter in Simulink and verifies the result obtained. Chapter 7 describes integration of all sub-systems to design the critical load stabilizer. In chapter 8, results obtained from testing the critical load stabilizer are evaluated and future improvements are discussed. Finally, chapter 9 concludes the project.

METHODOLOGY SECTION

CHAPTER 2 – DESIGN ANALYSIS AND TECHNICAL BACKGROUND

2.1 Project Design Decisions

In this chapter, the design decisions to complete this project will be discussed. MATLAB Simulink was chosen to design the simulation. To model the critical load stabilizer in MATLAB Simulink, Simscape library was essential to use. It contained all the required electrical and power system components needed for the project. Other simulation software such as Octave and Scilab were initially considered but were later rejected due to their shortcomings. Octave is a free software which uses a scientific programming language with built-in plotting and visualizing tools. It is largely compatible with MATLAB but lacks a simulation software similar to Simulink. Scilab is also a free and open source software for scientific computation and includes a model-based design software called Xcos. However, Simulink contains a much more extensive library than Xcos and was readily available on the university's computer free of cost, thereby making it the best choice.

Intermittent nature of renewable energy is the reason for fluctuations in grid voltage and frequency. Using circuit analysis, it was found that to stabilize the voltage on critical load Z_c , a device with variable impedance must be connected in series with non-critical load Z_{nc} . The mathematical derivation and computation associated with this circuit analysis are shown in chapter 3. It is not practical to replace the stabilizing device every time to change impedance across the non-critical load branch. So, the current design of critical load stabilizer is chosen as it injects variable voltage at an appropriate angle through the non-critical load branch to produce the same effect as variable impedance in stabilizing the voltage across Z_c .

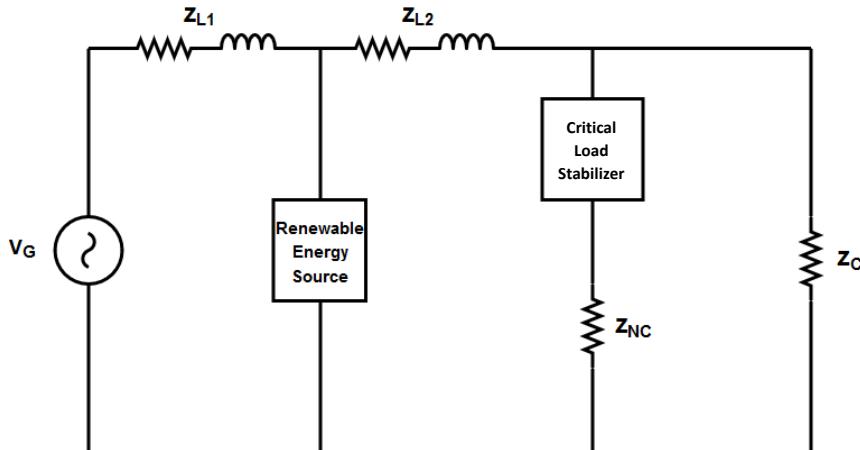


Figure 2: Grid circuit with critical load stabilizer

Figure 1 shows the control block diagram of the critical load stabilizer. The compensation controller is responsible for driving the error on critical load voltage v_c to zero. It consists of a PI controller which has

low steady-state error, improved damping and zero offset which is suitable for our design. In the initial stage, the controller was a Proportional (P) controller but with steady-state error. To eliminate steady-state error, Integral (I) controller was added to P. As PI controller's output was in acceptable range, the Derivative (D) controller was not needed. Tuning a Proportional-Integral-Derivative (PID) controller was a complex task and the result did not lead to a much better output. Sub-chapter 2.2 will describe in detail about PI controllers along with their mathematical equations. Chapter 4 will talk about how PI controller was designed and tested for this project.

Synchronization Network block is used to measure the phase angle of current passing through non-critical load branch. This phase is later used to generate a sinusoidal signal which creates a PWM signal to drive single-phase half bridge inverter. The output of PWM inverter drives voltage on the critical load stabilizer capacitor responsible for stabilizing the voltage across critical load Z_c . So, to implement the Synchronization Network block, a PLL system was chosen. As PLL generates an output signal whose phase is exactly same as its input signal, it was an appropriate choice. An extension of PLL called charge-pump PLL (CP-PLL) requires the addition of charge-pump between the phase-detector and loop filter of PLL. CP-PLL was an overkill for this project so it was not chosen. Sub-chapter 2.2 will discuss the fundamentals of PLL and how they work. Chapter 5 will describe how PLL block was designed and implemented in MATLAB.

The next decision was made regarding the design of low pass filter at the output of PWM inverter. The terminal voltage at the output of single-phase half-bridge inverter is a square wave. It is composed of high frequency harmonic components which are undesirable. To remove those high frequency harmonic components, a low pass filter was designed and used. The first choice was made to use a passive filter instead of an active filter. A passive filter is a kind of filter which is made up of passive elements such as resistors, inductors, capacitors and transformers requiring no external power source. On the contrary, an active filter consists of one or more active components requiring an external power source. A passive filter has the following advantages over an active filter:

- No/Low power consumption (apart from power taken out of the signal)
- Low cost
- Guaranteed stability

Second decision was made to use a low pass LC filter instead of a low pass RC filter. An LC filter has the following advantages over a RC filter:

- Dissipates less power (resulting in less heat) due to the replacement of resistor with an inductor
- Better noise rejection as it is a second order filter

So, looking at the advantages stated above, an LC low pass filter was implemented to filter out the high frequency harmonic components from the output of single-phase half-bridge inverter. Sub-chapter 2.2 will introduce the fundamentals of low pass filter with focus on low pass LC filter. Chapter 6 will describe how LC low pass filter was designed and evaluate the results.

2.2 Technical Background

In this sub-chapter, the fundamentals of various technologies such as PI controller, Phase-locked loop and low pass filter are introduced. Pertinent laws, principles and equations are used to explain and derive the

relationship between component parameters for better understanding of technical theory behind the project.

2.2.1 Proportional-Integral Controller

The basic idea of a control system is to figure out how to generate an appropriate actuating signal so that the output of system/plant will reach the desirable setpoint (SP). In a feedback controller, the output of plant also known as process variable (PV) is fed back into the control system and compared with SP[5]. The difference between the SP and the PV is known as error term $e(t)$. If PV is exactly equal SP, then $e(t)$ would become zero which is the desired outcome. So, a controller needs to be designed which converts $e(t)$ into a suitable actuator command driving the error term to zero over time. Figure 3 shows the block diagram of a feedback control system.

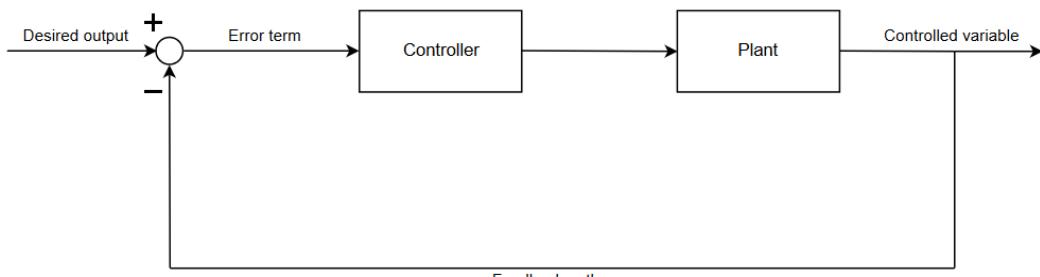


Figure 3: Feedback control loop

$$e(t) = SP - PV$$

where,

$e(t)$ = error term

SP = desired setpoint

PV = process variable/plant output

The simplest controller is a Proportional (P) controller which has a gain K_p . Based on the magnitude of $e(t)$, K_p is multiplied with it to produce a manipulated output. Figure 4 depicts the relationship between input and output of a P controller.

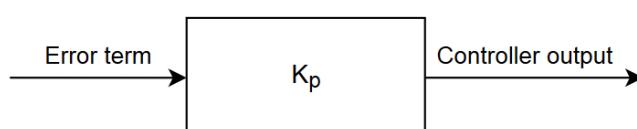


Figure 4: Block diagram of Proportional controller

$$m(t) \propto e(t)$$

$$\rightarrow m(t) = K_p * e(t)$$

where,

$m(t)$ = Controller output

K_p = Proportional gain

For higher values of K_p , the gain of error term will be large, resulting in smaller rise time t_{rs} . Smaller values of t_{rs} indicate faster response of controller output to reach the desired setpoint. However, high value of K_p can also lead to maximum peak overshoot m_p . Conversely, low values of K_p lead to high t_{rs} but decreased value of m_p . Figure 5 shows the significance of Proportional controller gain K_p .

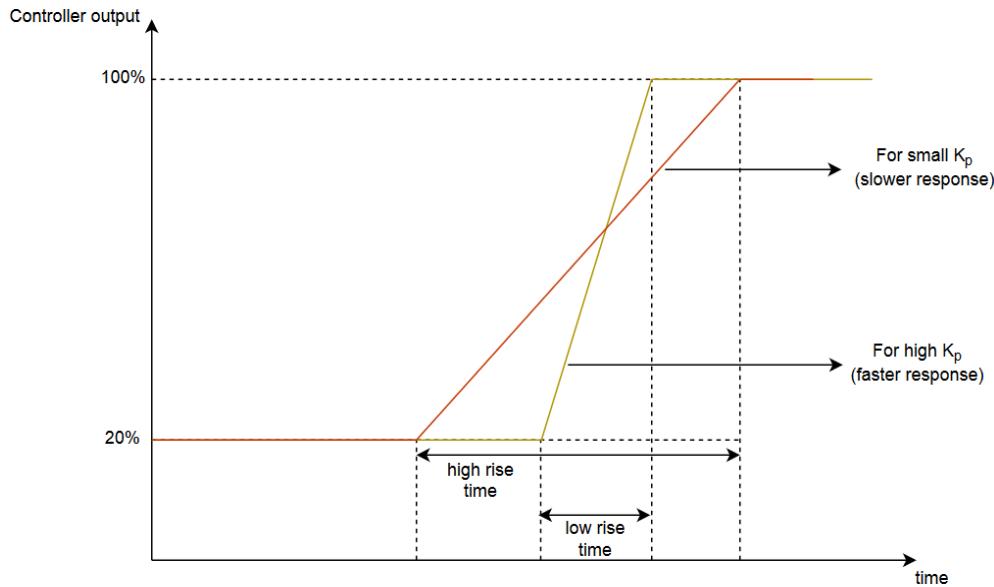


Figure 5: Significance of K_p on controller output

Figure 6 gives a physical understanding of a proportional controller. K_p is applied to input error signal and output signal response is shown.

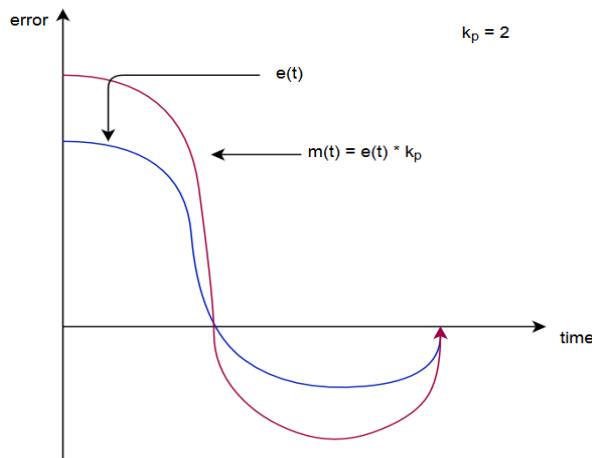


Figure 6: Physical understanding of K_p

To further understand P controller, the block diagram of P controller with 2nd order system is examined. Transfer functions are used to explain the effect of P controller.

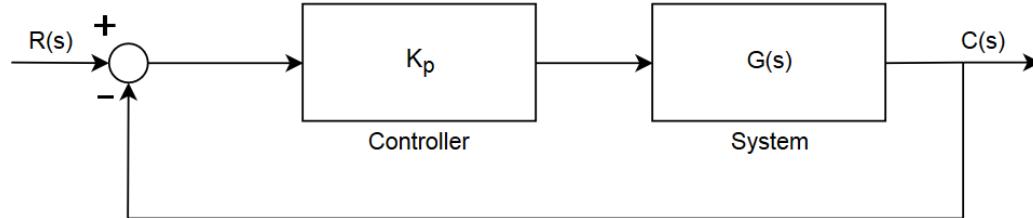


Figure 7: Block diagram of P controller with 2nd order system

$G(s)$ is the transfer function of a 2nd order system.

$$G(s) = \frac{\omega_n^2}{s(s + 2 * \zeta * \omega_n)} \quad (1)$$

where,

ω_n = natural frequency

ζ = dampening frequency

Transfer function for P controller with 2nd order system is represented as $\frac{C(s)}{R(s)}$.

$$\frac{C(s)}{R(s)} = \frac{K(s)}{1 + K(s) * H(s)} \quad (2)$$

where,

$$K(s) = K_p * G(s)$$

Replacing G(s) in the above equation with equation (1),

$$\rightarrow K(s) = \frac{K_p * \omega_n^2}{s(s + 2 * \zeta * \omega_n)}$$

and,

$$H(s) = 1$$

So, after replacing K(s) and H(s) with their respective values, equation (2) expands to,

$$\frac{C(s)}{R(s)} = \frac{\frac{K_p * \omega_n^2}{s(s + 2 * \zeta * \omega_n)}}{1 + \frac{K_p * \omega_n^2}{s(s + 2 * \zeta * \omega_n)}}$$

Simplifying the above equation yields,

$$\rightarrow \frac{C(s)}{R(s)} = \frac{K_p * \omega_n^2}{s^2 + (2 * \zeta * \omega_n * s) + (K_p * \omega_n^2)} \quad (3)$$

By comparing equation (3) and equation (1), it is evident that a P controller affects the natural frequency of a 2nd order standard system. After introducing a P controller to a 2nd order system, the natural frequency term ω_n^2 has a gain of K_p multiplied to it. As the magnitude of K_p is increased, it would lead to a faster response inferring higher frequency, and based on increased frequency, the slope will increase. The drawback of P controller is the presence of steady state error. The reason for steady state error is that, as the system approaches the setpoint, the error signal becomes smaller and smaller. The control system works by multiplying K_p to the error signal and eventually the error signal is small enough that the controlled signal is not enough to force PV to SP. So, the error keeps on getting smaller but does not converge to zero. Figure 9 shows PV for different values of K_p proving that the error fails to converge to zero and produces a steady state error. The input is a step signal with initial value of 0, step time of 5 seconds and final value of 100. It might seem in figure 9 that PV is exactly equal SP, but figure 10 displays the magnified version. Figure 8 shows the P control system implemented in MATLAB Simulink to observe the effect of different values of K_p on system response.

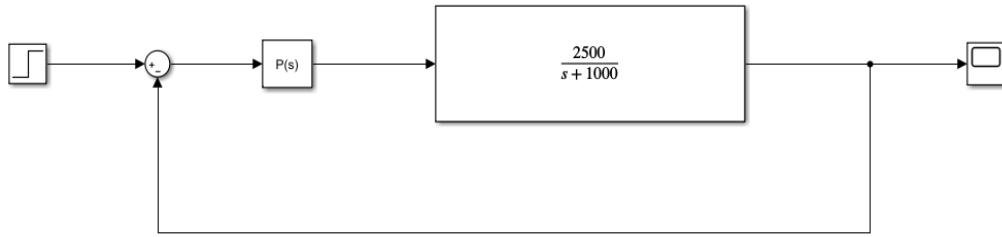


Figure 8: Simulink model of P controller with 2nd order system

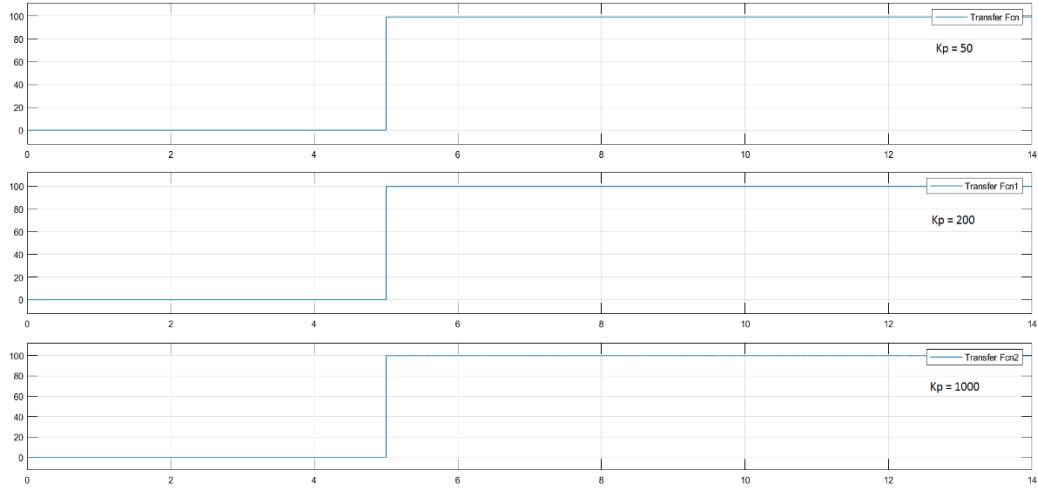


Figure 9: Simulink output of process variable for different values of K_p

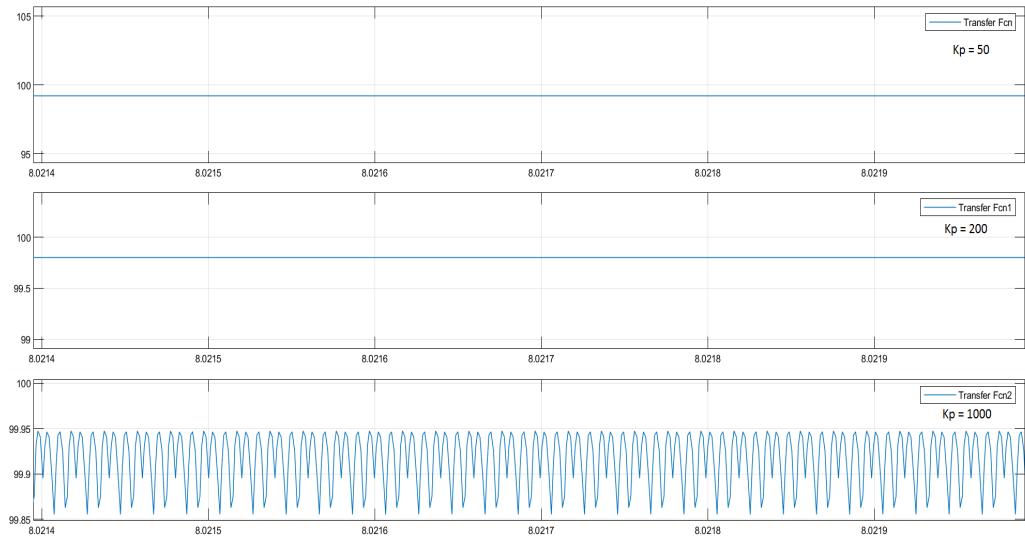


Figure 10: Magnified version of Simulink output of process variable

To solve the issue of steady state error, a new type of controller is needed which is an Integral (I) controller. An Integral controller provides an output which is integral of the input error signal. Figure 11 shows the relationship between the input and output of an I controller.

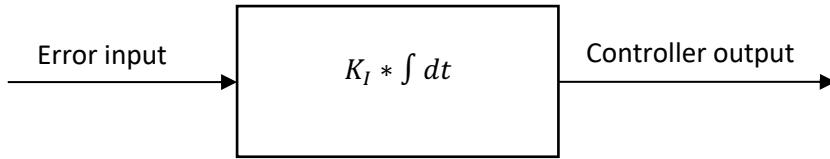


Figure 11: Block diagram of Integral controller

$$\begin{aligned}
 m(t) &\propto e(t) \\
 \rightarrow m(t) &= K_I \int e(t) dt \quad (1)
 \end{aligned}$$

where,

K_I = integral constant

By taking Laplace transform of equation (1),

$$\begin{aligned}
 \rightarrow M(s) &= K_I \cdot \frac{E(s)}{s} \\
 \rightarrow \frac{M(s)}{E(s)} &= \frac{K_I}{s} \quad (2)
 \end{aligned}$$

Figure 12 gives a physical understanding of an Integral controller. At the beginning of the graph, the input has a high positive value, so the output increases rapidly as it is an integral of the input value. Later, the input slowly decreases but is still positive. Now, the output value is still increasing but at a decreased rate. This is because integration is simply the addition of current value with the previous value. When the input value becomes negative, the output graph starts to decrease. As the magnitude of negative value increases, the rate at which output value decreases becomes larger represented by a greater dip in graph. To sum up, if the input is positive, the output will increase positively and conversely, if the input is negative, the output will decrease negatively. From equation (2), we observe that due to Integral controller, a pole is added at the origin which increase the type of the system. Hence, it helps in decreasing steady state error.

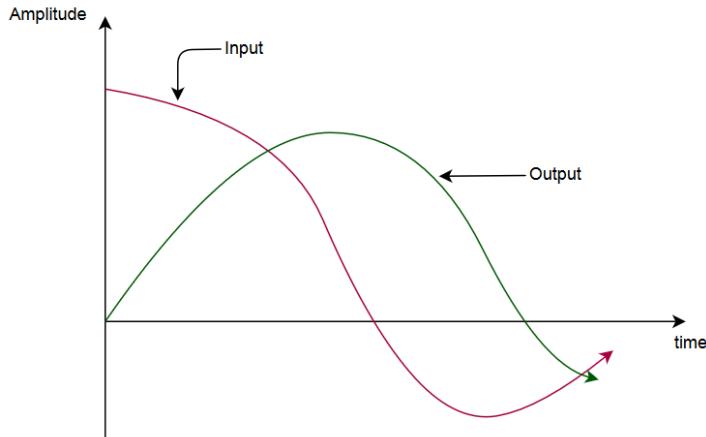


Figure 12: Input-output relationship of Integral controller

In this project, a Proportional-Integral (PI) controller is used which is a combination of P controller and I controller. The output of this controller is proportional to error signal and the integral of error signal. The equation of controlled output in time and frequency domain is shown below. The significance of PI controller is that, when the input error signal is zero, there exists a steady output to keep the error signal at zero due to the existence of I controller. Thus, it eliminates offset.

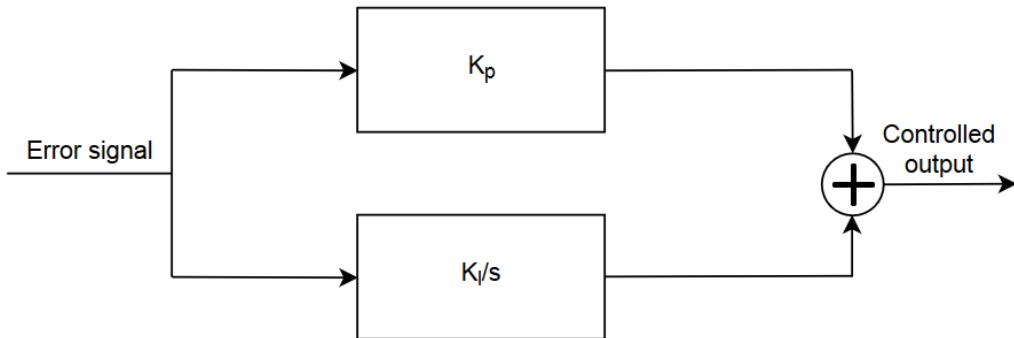


Figure 13: Block diagram of PI controller

$$m(t) = K_p * e(t) + K_I * \int e(t) dt \quad (1)$$

Using Laplace transform in equation (1) gives,

$$M(s) = K_p * E(s) + K_I \frac{E(s)}{s}$$

Simplifying the above equation,

$$M(s) = \left(K_p + \frac{K_I}{s} \right) * E(s)$$

To further understand a PI controller, the block diagram of PI controller with 2nd order system is examined. Transfer functions are used to explain the effect of PI controller.

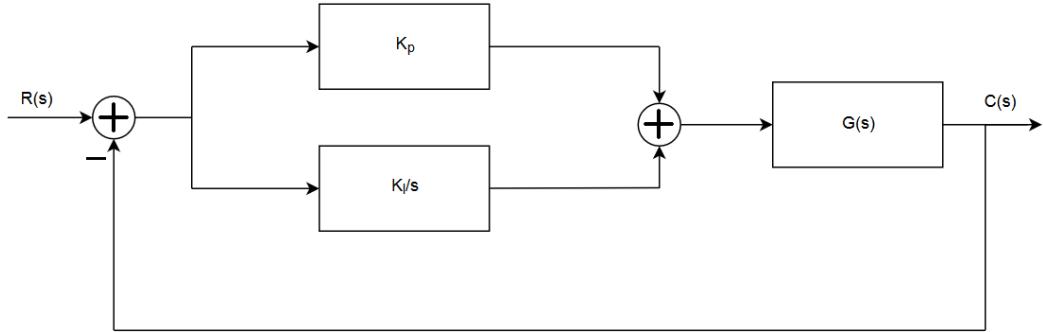


Figure 14: Block diagram of PI controller with 2nd order system

$G(s)$ is the transfer function of a 2nd order system.

$$G(s) = \frac{\omega_n^2}{s(s + 2 * \zeta * \omega_n)} \quad (1)$$

where,

ω_n = natural frequency

ζ = dampening constant

Transfer function for P controller with a 2nd order system is represented as $\frac{C(s)}{R(s)}$.

$$\frac{C(s)}{R(s)} = \frac{K(s)}{1 + K(s) * H(s)} \quad (2)$$

where,

$$K(s) = \left(K_p + \frac{K_I}{s} \right) * G(s)$$

Replacing $G(s)$ in the above equation with equation (1),

$$\rightarrow K(s) = \left(K_p + \frac{K_I}{s} \right) * \left(\frac{\omega_n^2}{s(s + 2 * \zeta * \omega_n)} \right)$$

and,

$$H(s) = 1$$

So, after replacing $K(s)$ and $H(s)$ with their respective values, equation (2) expands to,

$$\frac{C(s)}{R(s)} = \frac{\left(K_p + \frac{K_I}{s}\right) * \left(\frac{\omega_n^2}{s(s+2*\zeta*\omega_n)}\right)}{1 + \left(K_p + \frac{K_I}{s}\right) * \left(\frac{\omega_n^2}{s(s+2*\zeta*\omega_n)}\right)}$$

Simplifying the above equation yields,

$$\rightarrow \frac{C(s)}{R(s)} = \frac{(K_p * s + K_I) * \omega_n^2}{s^3 + (2 * \zeta * \omega * s^2) + (K_p * \omega_n^2 * s) + (K_I * \omega_n^2)} \quad (3)$$

Equation (3) shown above is the transfer function of PI controller with standard 2nd order system. By manipulating K_p , the speed of the system can be controlled. K_I can be changed to improve damping and together with K_p , PI controller can eliminate steady state error. The information conveyed in this sub-chapter was important to understand why PI controller was implemented. It provides enough knowledge to understand chapter 4 where the implementation of PI controller in MATLAB Simulink will be explained.

2.2.2 Phase-locked loop

A Phase-locked loop is a non-linear feedback system that tracks the phase of input signal and minimizes phase error at local oscillator [6]. It is combined of a phase detector and a voltage-controlled oscillator (VCO) such that the VCO maintains a constant phase angle relative to input signal. As the VCO operates at high frequency, it is susceptible to temperature and noise which make the output frequency unstable. Therefore, a feedback system such as PLL is needed to stabilize the frequency of VCO. The working principle of PLL is to compare the phase of VCO output signal with the input reference signal and drive the phase difference to zero to achieve same frequency. If the two signals have different frequencies, then the phase difference across multiple time instances will vary. This is due to the fact that the time taken for each cycle is different and hence they are moving at different rates. So, the control loop measures the phase difference multiple times and if the phase difference remains constant, it can be inferred from the above explanation that the two signals have same frequency. It is important to note that the signals may have a phase difference between them. But due to constant phase difference, it is known that one signal is either leading or lagging the other by same amount. Hence, they will have equal frequencies. Figure 15 gives a physical understanding of how signals are measured for phase difference and how constant phase difference translates to equal frequency.

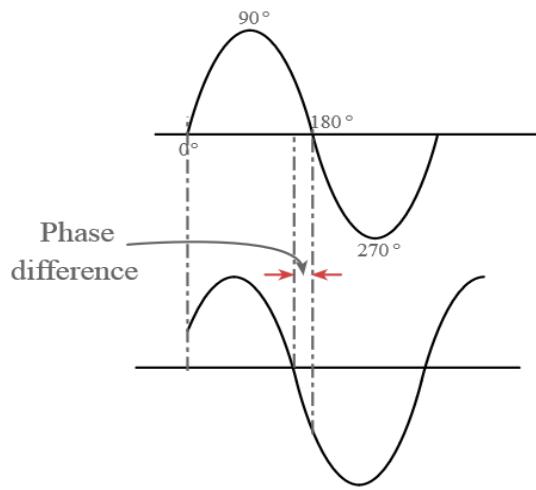


Figure 15: Phase difference detection in PLL

Figure 16 shows the block diagram of a standard PLL system. It has 3 main components – a phase detector, low pass filter and a VCO. First, the phase detector has 2 main inputs – reference input signal with frequency f_{in} and VCO output with frequency f_{out} . It will generate an output error signal with 2 frequency components $f_{in} + f_{out}$ and $f_{in} - f_{out}$ based on the phase difference of its inputs. The high frequency component $f_{in} + f_{out}$ is considered as noise and rejected. So, the low pass filter rejects the high frequency component and results in a DC or minimum frequency signal. This minimum frequency signal is fed into the VCO to generate an output signal of frequency f_{out} . The output signal frequency f_{out} is directly proportional to the amplitude of the input signal generated by the low pass filter. Sometimes, a frequency divider of factor N is used in the feedback path to generate a frequency $\frac{f_{out}}{N}$ which is a multiple of the input signal frequency f_{in} . The major function of each component is summarized below.

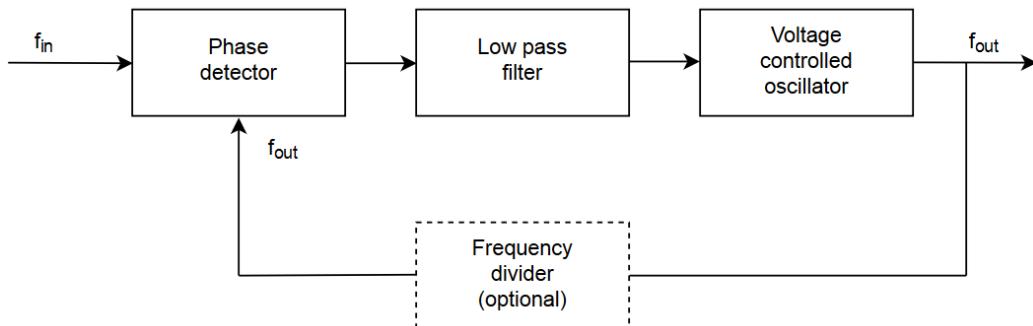


Figure 16: Block diagram of PLL system

- **Phase detector** – It compares the input signal phase with VCO output signal phase and produces an output proportional to the phase error.

- **Low pass filter** – It filters out the high frequency noise to output a minimum frequency signal or a DC signal. This component can be implemented either as a passive low pass filter or an active low pass filter.
- **Voltage-controlled oscillator** – The instantaneous VCO frequency is controlled by the input voltage fed from the low pass filter.

$$f_o = f + k * \omega_m$$

Here, f is constant frequency and k depends on the input voltage. Thus, the frequency of VCO, f_o , is directly controlled by the DC or minimum frequency signal.

The operation of PLL system goes through 3 states.

- **Free running** – During the absence of f_{in} , PLL is in free running state. The output of phase detector and low pass filter is zero and thus, the VCO operates at its free running frequency $f_o = f$.
- **Capture** – When the input f_{in} is applied, VCO output starts to change and PLL is said to be in capture mode.
- **Phase lock** – Once, the input frequency and VCO frequency are equal, PLL is said to be in phase lock state.

$$f_{in} - f_o = 0 \quad (\text{phase lock state})$$

The above explanation of PLL is appropriate to understand the use of it in building the critical load stabilizer. Chapter 5 will discuss the design and implementation of PLL in building critical load stabilizer.

2.2.3 LC low pass filter

A simple LC low pass filter is built using an inductor in series and a capacitor in parallel. Figure 17 shows the circuit diagram of an LC low pass filter. The impedance of an inductor is directly proportional to frequency and hence, it discriminates against high frequencies. A capacitor, on the other hand, favors high frequencies as its impedance is inversely proportional to frequency. However, in an LC circuit the capacitor is connected in parallel and connected to ground. So as the frequency increases, impedance decreases, and the signal is driven to ground. Thus, a capacitor in LC circuit also discriminates against high frequencies.

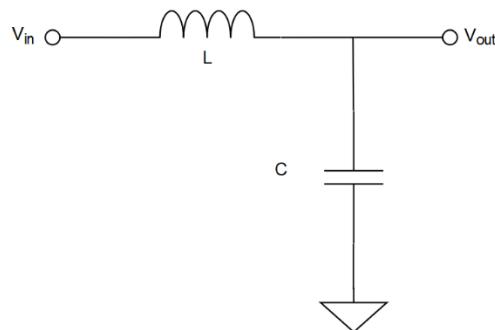


Figure 17: Circuit diagram of LC low pass filter

The impedance of an inductor is given as follows,

$$Z_L = j(2 * \pi * f * L) \Omega \quad (1)$$

The impedance of a capacitor is given as follows,

$$Z_C = \frac{1}{j(2 * \pi * f * C)} \Omega \quad (2)$$

Resonance occurs when the circuit is driven at an angular frequency where capacitor impedance is equal to inductor impedance.

$$Z_L = Z_C \quad (3)$$

In equation (3), replacing Z_L with equation (1) and Z_C with equation (2) gives,

$$j(2 * \pi * f * L) = \frac{1}{j(2 * \pi * f * C)}$$

Rearranging the above equation gives the cutoff frequency for low pass filter which is an important equation for designing a filter.

$$f_c = \frac{1}{2 * \pi * \sqrt{LC}} \text{ Hz}$$

At frequencies smaller than the cutoff frequency, the circuit operates in capacitive mode and at frequencies greater than the cutoff frequency, the circuit operates in inductive mode. Thereby, the filter allows frequencies less than f_c to pass through, filters out the frequencies greater than f_c and provides a resonant peak at frequency f_c [7]. Figure 18 displays the frequency response of a low pass LC filter with the following specification –

$$L = 0.15 \text{ Henries}$$

$$C = 13.2 \mu\text{F}$$

$$f_c = 113.1 \text{ Hz}$$

BodeDiagram

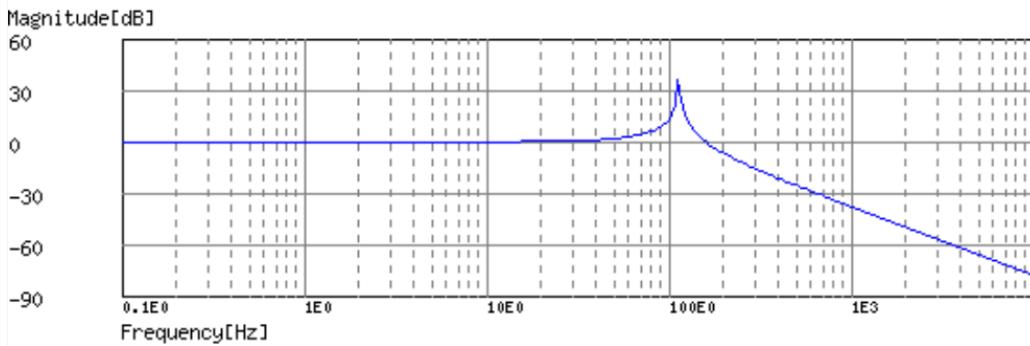


Figure 18: Bode diagram of LC low pass filter

CHAPTER 3 – MATHEMATICAL MODEL OF CRITICAL LOAD STABILIZER

This chapter will describe how the general design of critical load stabilizer was formulated by creating a mathematical model for it. The model was implemented and tested on MATLAB Simulink for verification and evaluation of results.

3.1 Circuit analysis with no Renewable energy source

First, a simple circuit analysis was conducted with no renewable energy source and a simple capacitor in place of critical load stabilizer in the non-critical load branch. The circuit diagram is shown below.

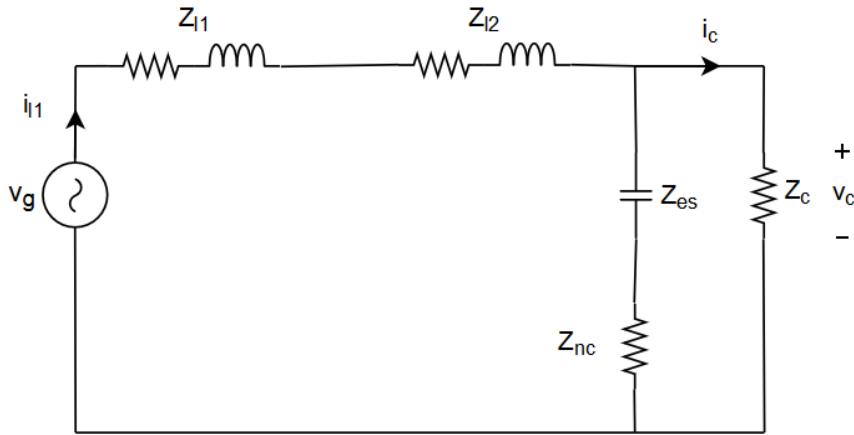


Figure 19: Grid circuit diagram with no renewable energy

Where,

V_g = AC grid voltage source

Z_{l_1} = Grid line impedance 1

Z_{l_2} = Grid line impedance 2

Z_{es} = Critical load stabilizer impedance (in this case, a simple capacitor)

Z_{nc} = Non – critical load impedance

Z_c = Critical load impedance

i_{l_1} = Grid line current

i_c = Critical load branch current

The motive of this circuit analysis was to obtain ideal circuit parameters in absence of renewable energy source. These parameters were later used as reference for stabilization when the system was under the influence of renewable energy source fluctuation. First, the equivalent load impedance is calculated which consists of a parallel combination of non-critical and critical load branch impedance.

$$Z_{load} = (Z_{es} + Z_{nc}) \parallel Z_c$$

$$\rightarrow Z_{load} = \frac{Z_c * (Z_{es} + Z_{nc})}{Z_{es} + Z_{nc} + Z_c}$$

After calculating Z_{load} , the voltage across critical load v_c can be calculated using the below formula,

$$v_c = v_g * \frac{Z_{load}}{Z_{load} + Z_{l_1} + Z_{l_2}} \quad (1)$$

Using v_c , the current i_c can be calculated using Ohm's law,

$$i_c = \frac{v_c}{Z_c}$$

Replacing v_c in the above equation with equation (1),

$$i_c = \frac{v_g * Z_{load}}{Z_c * (Z_{load} + Z_{l_1} + Z_{l_2})} \quad (2)$$

The critical load power S_c is calculated using equation (1) and (2),

$$S_c = v_c * i_c$$

$$\rightarrow S_c = \frac{v_g^2 * Z_{load}^2}{Z_c * (Z_{load} + Z_{l_1} + Z_{l_2})^2} \quad (3)$$

To compute v_c , i_c and S_c , the following circuit parameters were used,

$$v_g = 221 V$$

$$Z_{l_1} = Z_{l_2} = 0.1 + j(\omega * 1.22 * 10^{-3}) \Omega$$

$$Z_c = Z_{nc} = 50 \Omega$$

Using equation (1), (2) and (3), the following results were obtained,

$$v_c = 220.69 - j(3.83) V$$

$$\rightarrow |v_c| = 220.7225 V$$

$$\rightarrow \angle v_c = -0.959^\circ$$

$$i_c = 4.41 - j(0.07) A$$

$$\rightarrow |i_c| = 4.414 A$$

$$\rightarrow \angle i_c = -0.959^\circ$$

$$S_c = 974.37 + j(3.32 * 10^{-15}) W$$

The above results were verified by implementing the model on MATLAB Simulink. The results are shown below.

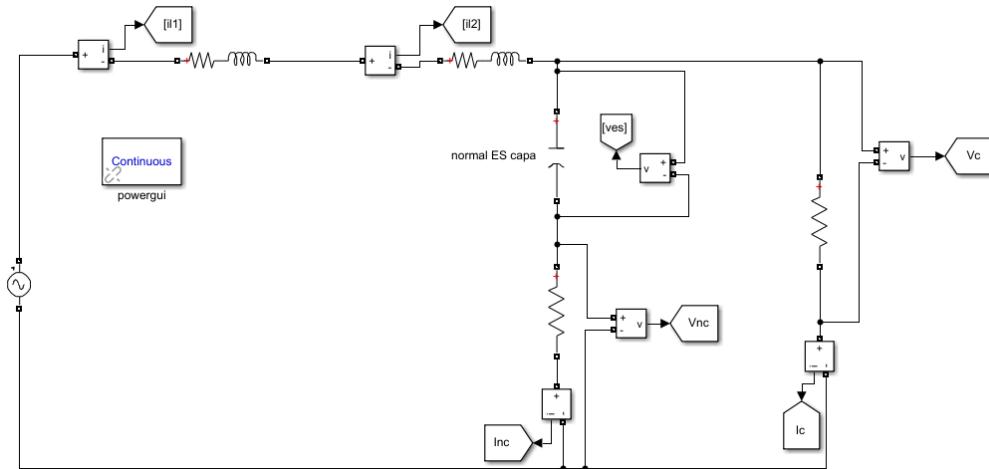


Figure 20: Simulink model of grid with no renewable energy source

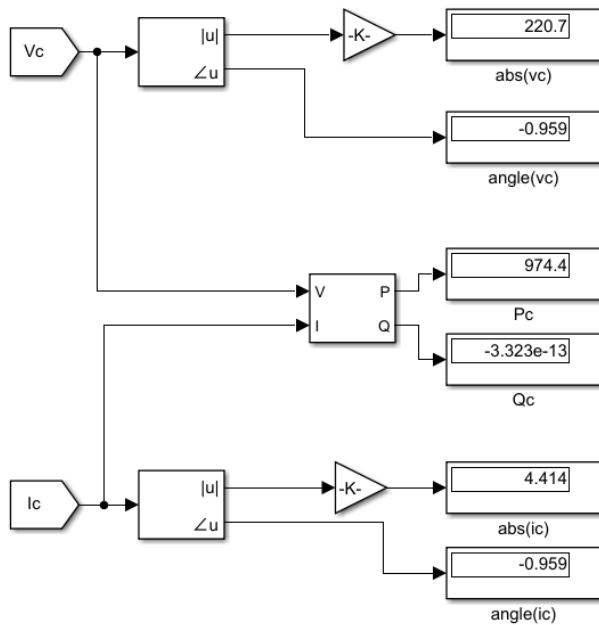


Figure 21: Simulink results of grid circuit model with no renewable energy source

The simulated results matched the calculated results proving the correctness of circuit analysis.

3.2 Circuit analysis with Renewable energy source

Next, a renewable energy source was introduced in the circuit and its effect was observed on circuit parameters. The renewable energy source was modeled as a component parallel to grid voltage source with impedance Z_{res} .

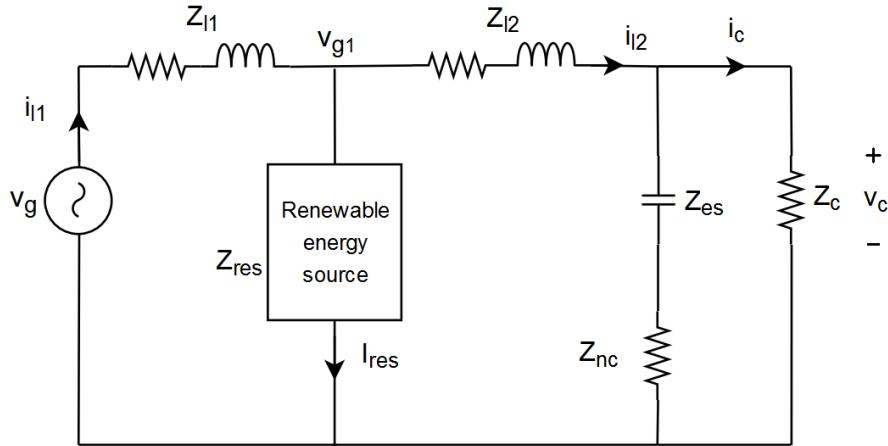


Figure 22: Grid circuit diagram with renewable energy source

First, line current i_{l_1} is calculated,

$$i_{l_1} = \frac{v_g}{Z'_{load}}$$

where,

$$Z'_{load} = Z_{l_1} + (Z_{res} \parallel (Z_{l_2} + Z_{load}))$$

Next, voltage at renewable energy source is calculated by subtracting the voltage drop caused by Z_{l_1} from v_g .

$$\begin{aligned} v_{g_1} &= v_g - i_{l_1} * Z_{l_1} \\ \rightarrow v_{g_1} &= \frac{v_g * (Z_{res} \parallel (Z_{l_2} + Z_{load}))}{Z'_{load}} \quad (4) \end{aligned}$$

The line current i_{l_2} is calculated,

$$\begin{aligned} i_{l_2} &= \frac{v_{g_1}}{Z_{l_2} + Z_{load}} \\ \rightarrow i_{l_2} &= \frac{v_g * Z_{res}}{Z_{l_1} * (Z_{l_2} + Z_{load} + Z_{res}) + Z_{res} * (Z_{l_2} + Z_{load})} \quad (5) \end{aligned}$$

Finally, the voltage across critical load is calculated to observe the effect of renewable energy source Z_{res} .

$$v'_c = v_{g_1} - i_{l_2} * Z_{l_2}$$

$$\rightarrow v'_c = \frac{v_g * Z_{res} * Z_{load}}{Z_{l_1} * (Z_{l_2} + Z_{load} + Z_{res}) + Z_{res} * (Z_{l_2} + Z_{load})} \quad (6)$$

To calculate v'_c , all the previous parameters were used and Z_{res} was modeled using a 0.15 H inductor.

So, the calculated Z_{res} equals to,

$$Z_{res} = j(\omega * 0.15)$$

$$\rightarrow Z_{res} = j(47.12) \Omega$$

Using equations (4), (5) and (6), the voltage and current of critical load was calculated.

$$v'_c = 218.88 - j(3.19) V$$

$$\rightarrow |v'_c| = 218.9463 V$$

$$\rightarrow \angle v'_c = -0.8346^\circ$$

$$i'_c = 4.38 - j(0.06) A$$

$$\rightarrow |i'_c| = 4.38 A$$

$$\rightarrow \angle i'_c = -0.8346^\circ$$

The Simulink model and results obtained are shown below.

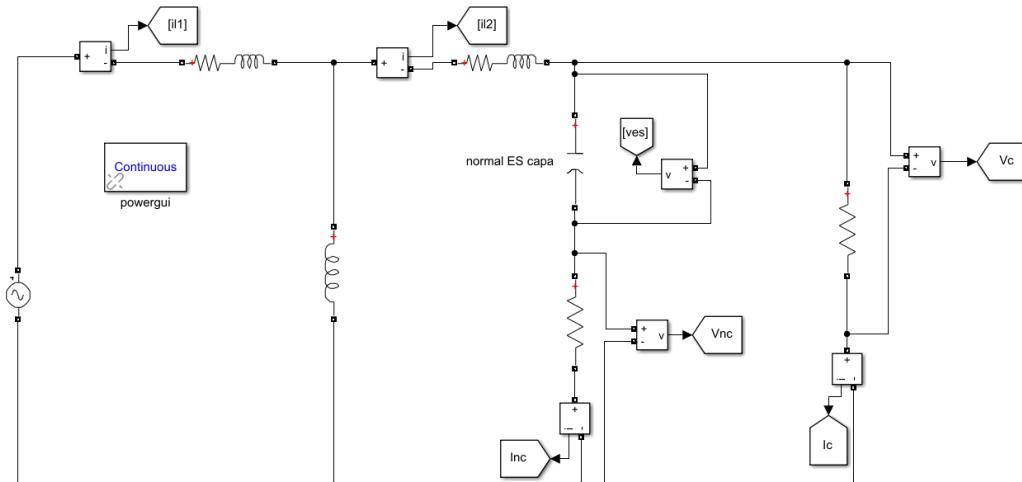


Figure 23: Simulink model of grid with renewable energy source

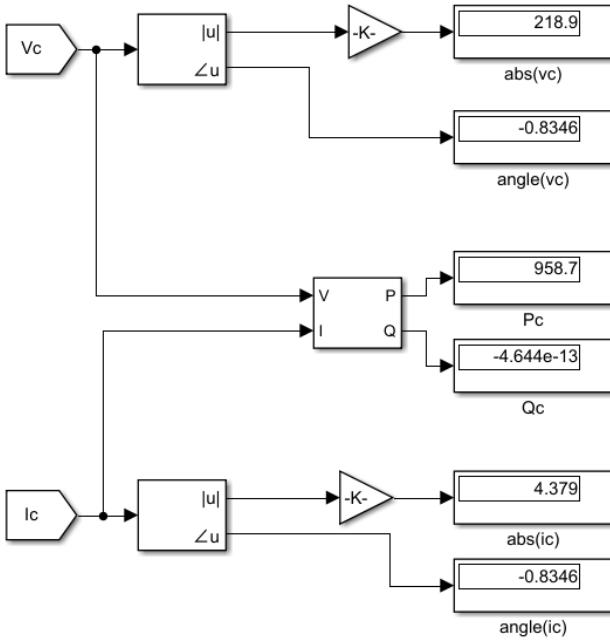


Figure 24: Simulink results of grid circuit model with renewable energy source

The results obtained above verify the correctness of the mathematical model. Furthermore, it can be observed that the renewable energy source causes a voltage drop of 1.8 V at the critical load. Thus, there is a need to stabilize the critical load voltage to avoid damage from fluctuations.

3.3 Circuit analysis with Black box

To eliminate fluctuations, the capacitor of critical load stabilizer was replaced by a black box. The objective of upcoming circuit analysis was to calculate the impedance of this black box that will stabilize the voltage $v'_c = 218.9$ V to the original voltage $v_c = 220.7$ V.

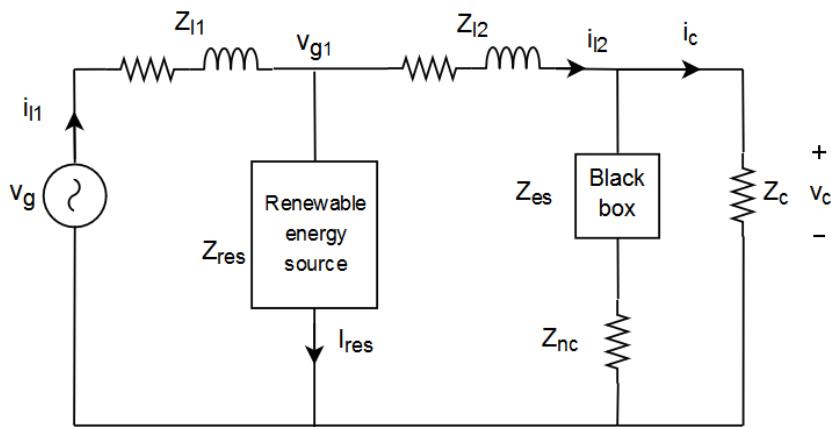


Figure 25: Grid circuit diagram with black box

To calculate the black box impedance, replace Z_{load} in equation (6) with Z''_{load} to get v''_c .

$$v''_c = \frac{v_g * Z_{res} * Z''_{load}}{Z_{l_1} * (Z_{l_2} + Z''_{load} + Z_{res}) + Z_{res} * (Z_{l_2} + Z''_{load})} \quad (7)$$

As the new voltage v''_c needs to be stabilized and brought back to original voltage v_c , equation (7) is equated with equation (1).

$$\begin{aligned} v_g * \frac{Z_{load}}{Z_{load} + Z_{l_1} + Z_{l_2}} &= \frac{v_g * Z_{res} * Z''_{load}}{Z_{l_1} * (Z_{l_2} + Z''_{load} + Z_{res}) + Z_{res} * (Z_{l_2} + Z''_{load})} \\ \rightarrow Z''_{load} &= \frac{Z_{load} * ((Z_{l_1} * Z_{res}) + (Z_{l_1} * Z_{l_2}) + (Z_{l_2} * Z_{res}))}{((Z_{l_1} * Z_{res}) + (Z_{l_2} * Z_{res}) - (Z_{load} * Z_{l_1}))} \end{aligned} \quad (8)$$

and,

$$Z''_{load} = \frac{Z_c * (Z''_{es} + Z_{nc})}{Z''_{es} + Z_{nc} + Z_c}$$

Rearranging the above equation yields,

$$Z''_{es} = \frac{((Z_c * Z_{nc}) - (Z''_{load} * Z_c) - (Z''_{load} * Z_{nc}))}{Z''_{load} - Z_c} \quad (9)$$

The current through non-critical load branch was calculated using Ohm's law.

$$i''_{nc} = \frac{v''_c}{Z''_{es} + Z_{nc}}$$

Voltage across non-critical load is given by,

$$v''_{nc} = i''_{nc} * Z_{nc}$$

The black box voltage is given by the following equation,

$$v''_{es} = Z''_{es} * i''_{nc}$$

Finally, the black box power is as follows,

$$S''_{es} = v''_{es} * i''_{nc}$$

Using circuit parameters in sub-chapters 3.1 and 3.2, the following results were calculated,

$$Z''_{es} = -46.5850 - j(68.5626) \Omega$$

$$|i''_{nc}| = 3.2153 A$$

$$\angle i''_{nc} = 86.1895^\circ$$

$$|v''_{es}| = 266.5206 V$$

$$\angle v''_{es} = -38.0047^\circ$$

The black box was implemented using a resistor-capacitor series component having the same impedance as calculated above. The capacitance value was calculated as follows,

$$C_{es} = \frac{1}{j(\omega * \text{imag}(Z_{es}))}$$

$$\rightarrow C_{es} = 4.6426 * 10^{-5} \text{ F}$$

In Simulink, a resistor-capacitor combination was implemented in place of a black box to achieve active-reactive power compensation.

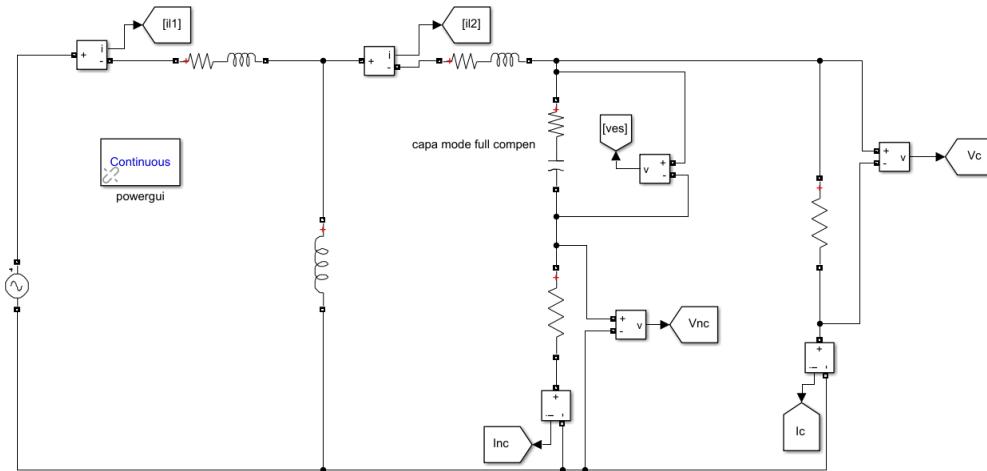


Figure 26: Simulink model of grid with resistor-capacitor black box

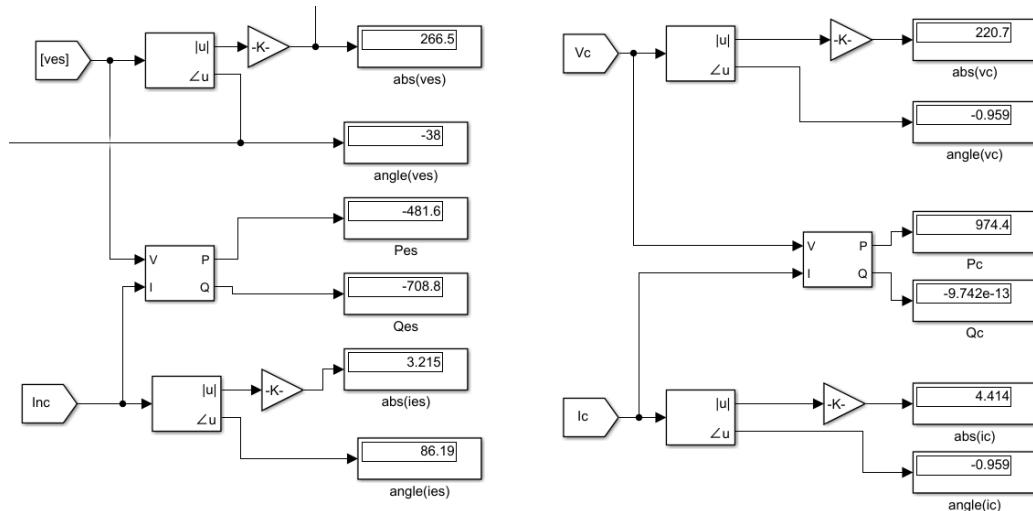


Figure 27: Simulink results of grid circuit model with resistor-capacitor black box

The Simulink results verify the calculation and prove that a resistor-capacitor combination can achieve full active-reactive power compensation. Voltage drop of 1.8V caused by the renewable energy source was fully compensated by the RC combination. However, in reality Z_{res} keeps on changing and therefore, Z''_{es} changes to compensate various degrees of fluctuation. It is impossible to modify resistor-capacitor combination physically to modify impedance every time, so an alternate solution needed to be created but on the same working principle. The critical load stabilizer works by injecting voltage with certain magnitude and angle to give the same effect as the resistor-capacitor combination would but without the need to modify any component for impedance change.

3.4 Circuit analysis with Critical load stabilizer

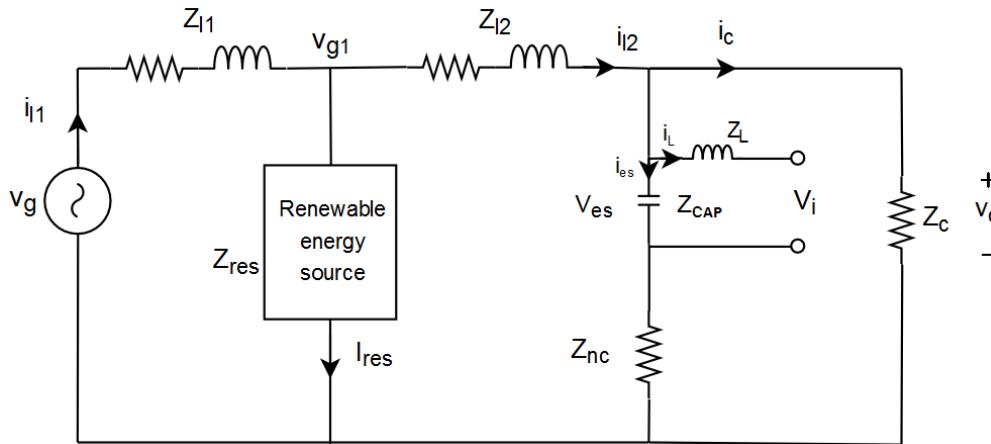


Figure 28: Grid circuit diagram with critical load stabilizer

The black box used in the previous sub-chapter was replaced by the original capacitor used in sub-chapter 3.1. Voltage v_i was designed to drive the capacitor in such a manner that it imitated the black box impedance.

The current passing through the capacitor is no longer equal to i_c due to inductor branch connecting v_i and v_{es} . Using Ohm's law, the current passing through the capacitor is,

$$i_{es} = \frac{v_{es}}{Z_{cap}}$$

where,

Z_{cap} = capacitor impedance

The current passing through the inductor is given by,

$$i_L = i_{nc} - i_{es}$$

Using Ohm's law, the voltage across the inductor is given by,

$$v_L = i_L * Z_L$$

And, the voltage v_i is expressed as,

$$v_i = v_{es} - v_L$$

The following results from sub-chapter 3.3 are used to calculate the correct value of v_i that drives the capacitor to stabilize voltage fluctuations caused by renewable energy source.

$$Z_{es}'' = -46.5850 - j(68.5626) \Omega$$

$$|i_{nc}''| = 3.2153 A$$

$$\angle i_{nc}'' = 86.1895^\circ$$

$$|v_{es}''| = 266.5206 V$$

$$\angle v_{es}'' = -38.0047^\circ$$

Inputting the above results into the equations derived in this sub-chapter provides the following result,

$$|v_i| = 266.7660 V$$

$$\angle v_i = -37.95^\circ$$

The critical load stabilizer model mentioned in this sub-chapter was implemented in MATLAB Simulink with the above calculated value of v_i to compensate voltage fluctuations caused by a 0.15 H inductor.

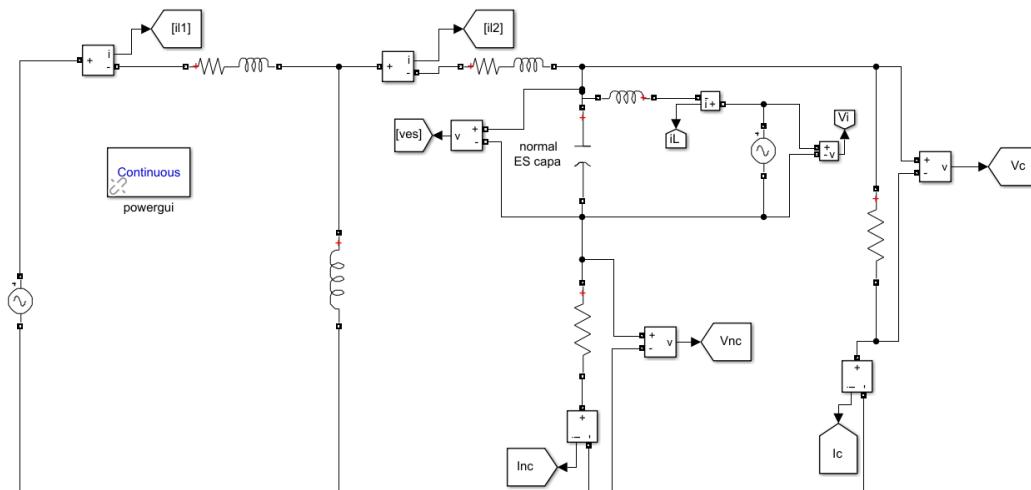


Figure 29: Simulink model of grid with critical load stabilizer

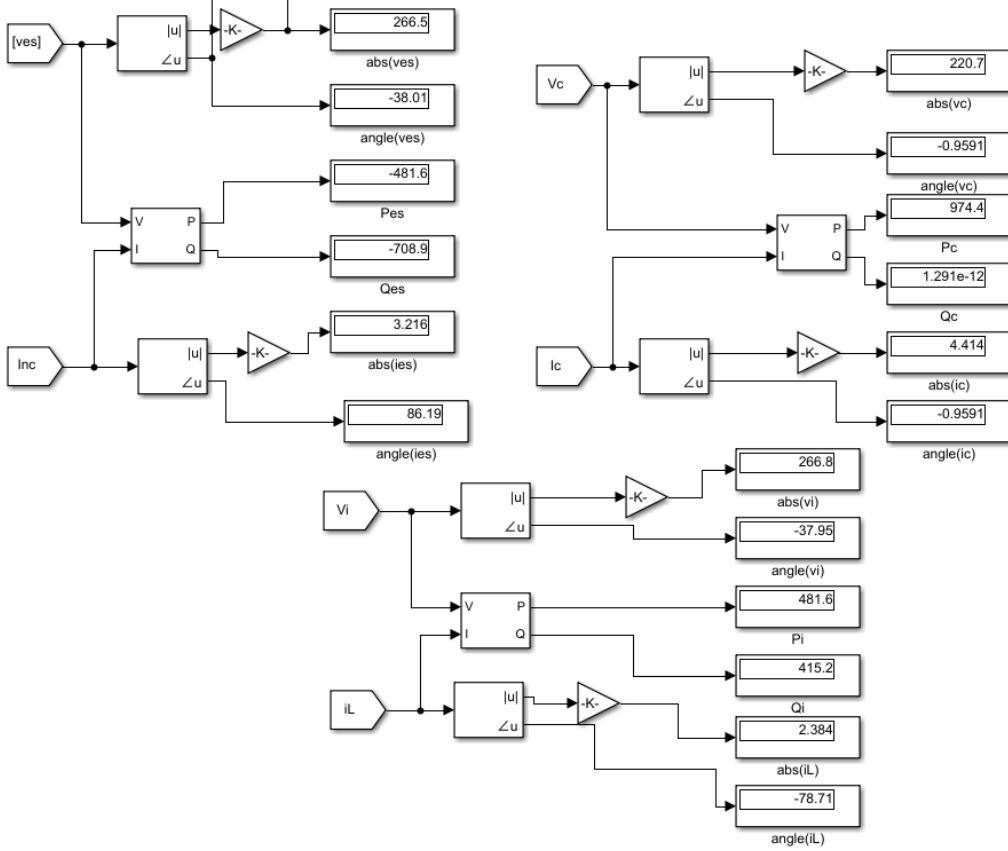


Figure 30: Simulink results of grid circuit model with critical load stabilizer

The above model proves that any fluctuation can be stabilized by providing v_i of accurate magnitude and phase. v_i can be calculated using the equations derived in this chapter and therefore, this model provides active-reactive compensation. As mentioned before, Z_{res} is an unknown and unstable value which represents random fluctuations by renewable energy source. Therefore, Z_{res} cannot be used to calculate v'_c . Instead, v'_c can be measured from the critical load and Z_{res} can be derived as a function of v'_c .

$$Z_{res} = v'_c * \frac{Z_{l_1} * (Z_{l_2} + Z_{load})}{(v_g * Z_{load}) - (v'_c * (Z_{l_1} + Z_{l_2} + Z_{load}))}$$

This value of Z_{res} can be used to calculate v_i . The next few chapters will describe how various sub-systems were designed to translate the above calculations into a circuit. PI controller was used to minimize critical load voltage error, PLL was used to inject correct phase into the critical load stabilizer and LC low pass filter was used to filter high frequency harmonic components from single phase half bridge inverter output v_i . Chapter 4 will describe how PI controller was implemented and evaluate the results.

CHAPTER 4 – IMPLEMENTING PI CONTROLLER IN MATLAB SIMULINK

The PI controller was implemented in Simulink using “PID controller” block available in the Simulink library under “Simulink/Continuous”. The input to PI controller is the difference between voltage on critical load v_c and reference voltage v_{c_ref} . In ideal case, the input must be zero in which case, critical load voltage v_c would have reached the desired setpoint $v_{c_ref} = 220V$. The error signal is fed into PI controller and output signal is used to generate a sine signal. The generated sine signal is later used to drive a PWM inverter injecting voltage onto a low pass filter capacitor for voltage stabilization. The figure below shows the Simulink model of PI controller used in building the critical load stabilizer.

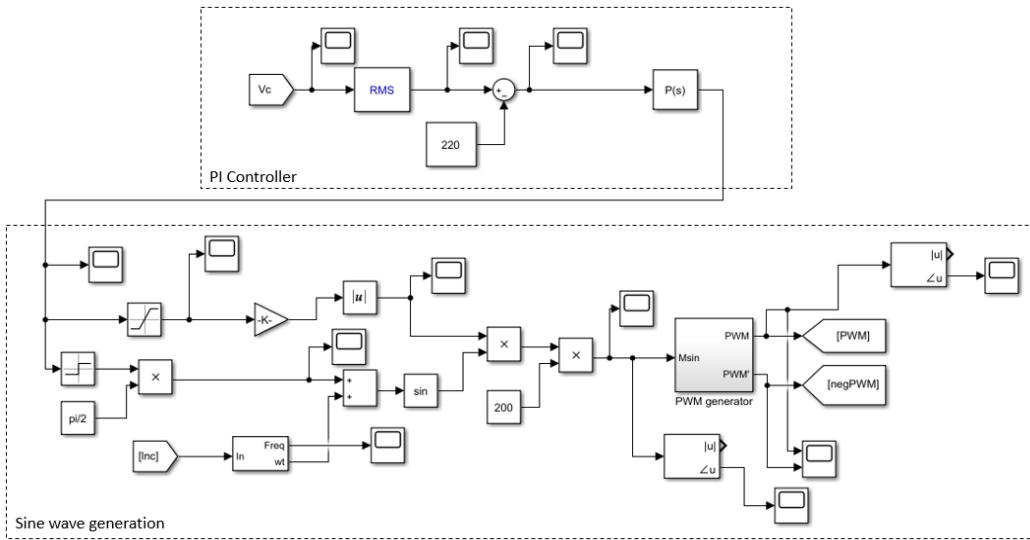


Figure 31: Simulink model of PI controller used in critical load stabilizer

The grid model used in this project was complicated and had no transfer functions available for it. It was a non-linear model and therefore, regular linear model tuning methods could not be used. Various heuristic methods such as Ziegler-Nichols method and Cohen-Coon tuning method were used for PI controller tuning [8][9]. Following the Ziegler-Nichols method, initially, the controller was converted to a P-only compensator and the proportional coefficient K_p was tuned until the controller output was stable with consistent oscillations. The value of K_p was set to 1 and then slowly increased until it reached a value known as ultimate gain. In this simulation, K_p was tuned to value of 50 where the output achieved consistent oscillations. The input and output of the controller are shown in the graphs below. The input refers to error in the critical load voltage.

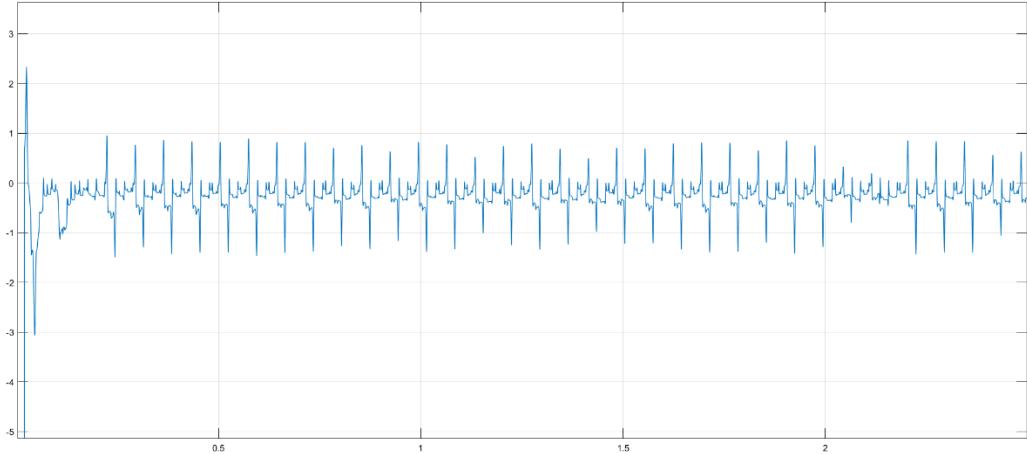


Figure 32: Input of PI controller with $K_p = 50$ and $K_I = 0$

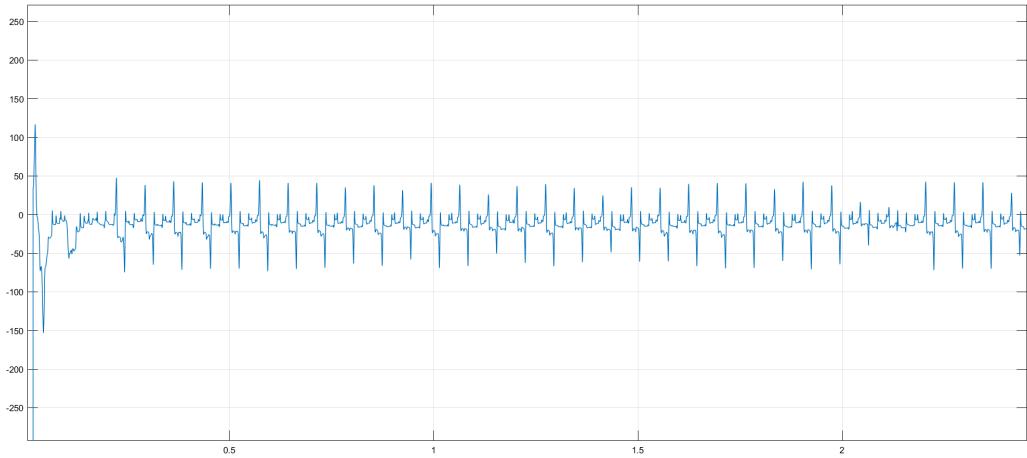


Figure 33: Output of PI controller with $K_p = 50$ and $K_I = 0$

Exactly like a traditional P controller, the output is directly proportional to input error signal. It can also be observed from the above results that there are clear oscillations present with a small magnitude of steady state error. To remove steady state error and dampen oscillations, coefficient K_I was tuned. Higher value of K_I led to high oscillations which were unacceptable for critical load stabilizer design. Large K_I also increases controller effort which needs to be limited for practical purpose. For instance, extremely high values of K_p and K_I might be able to decrease system error but they might require a huge output from controller. Such huge output might not be possible from an actual device. Therefore, a low value of K_I had to be chosen to fulfill design requirements. K_I was initially set to 1 and slowly increased until a proper balance between steady-state error and error fluctuations was achieved. The input and output of the controller are shown in the graphs below.

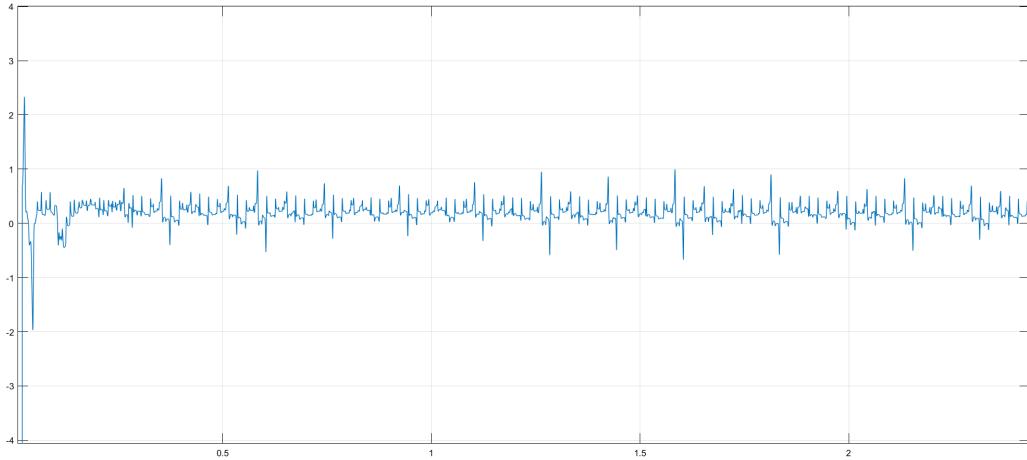


Figure 34: Input of PI controller with $K_p = 50$ and $K_I = 5$

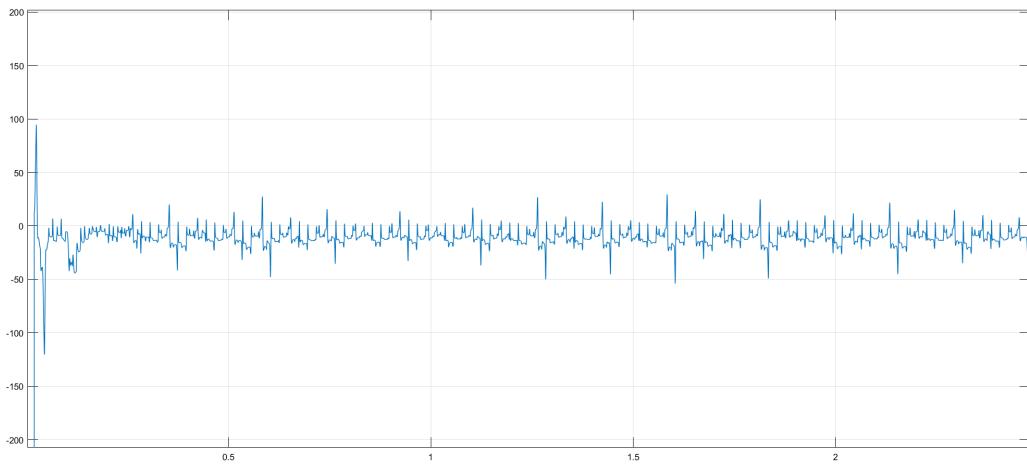


Figure 35: Output of PI controller with $K_p = 50$ and $K_I = 5$

Comparing figure 32 and figure 34 shows the reduction in steady state error and oscillation magnitude. The output of PI controller is closer to zero with oscillations within the range $\pm 0.4V$ compared to P only controller with oscillations within the range $\pm 1V$. The output still has fluctuations but that is due to unstable non-linear plant system. The magnitude of oscillations is extremely small and thus the following PI configuration fulfills the design requirements. Different values of K_p were also tested but resulted in increased steady state error or fluctuations. $K_p = 50$ achieved the best balance between optimal steady state error and magnitude of oscillation. Thereafter, $K_I = 5$ fulfilled the design goals by reducing the steady state error to almost zero and maintaining low oscillations. The next chapter will discuss the implementation of Phase-locked loop and evaluate its results.

CHAPTER 5 – IMPLEMENTING PHASE-LOCKED LOOP IN MATLAB SIMULINK

The main purpose of PLL in critical load stabilizer is to calculate the phase angle of non-critical load branch current i_{nc} . The calculated phase is used to generate sine wave for voltage stabilization. The input of PLL is i_{nc} and the output is ω_t . PLL was implemented in Simulink using “PLL” block available in the Simscape library under “Simscape/Power Systems/Specialized Technology/Control & Measurements”. The Simulink model is shown below.

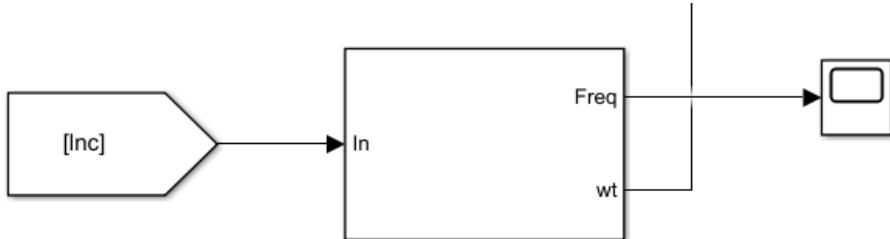


Figure 36: Simulink model of PLL

The figure below shows parameter values used to implement PLL block in Simulink. The minimum frequency was set as 45 Hz because the ideal grid frequency was marked at 50 Hz.

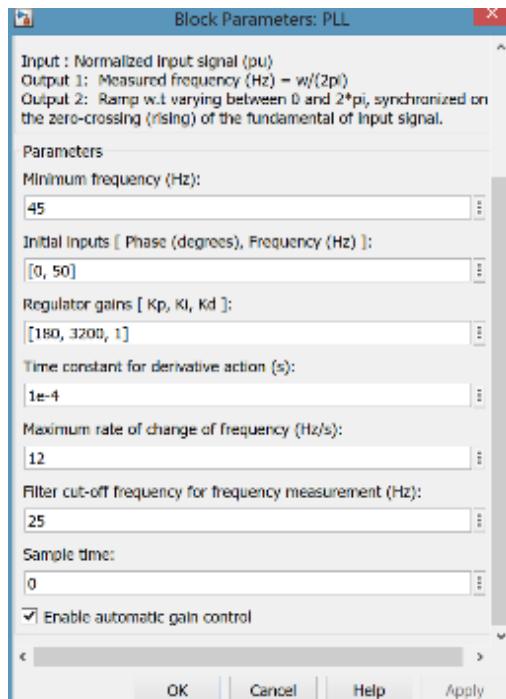


Figure 37: Simulink PLL block parameters

The output of PLL is shown in the graph below.

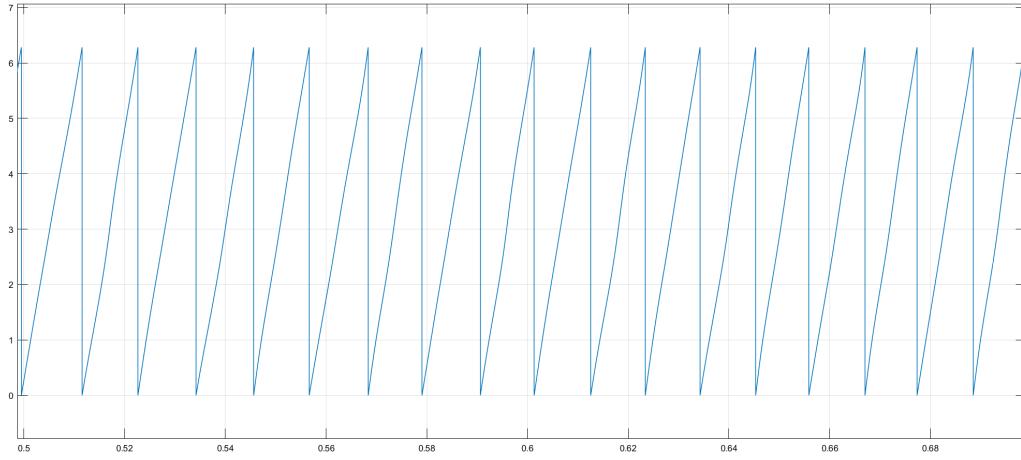


Figure 38: Simulink output of PLL block

The PLL block outputs an angle ωt , varying between 0 and 2π . It is synchronized on the rising zero-crossing of the fundamental of input signal. The resultant phase from PLL will be summed with $\pm \frac{\pi}{2}$ to generate a sine wave. The magnitude of sine wave was calculated from the output of PI controller. Thus, PLL holds a vital role in the critical load stabilizer system. The next chapter will explain how LC low pass filter was implemented in Simulink and evaluate the results.

CHAPTER 6 – IMPLEMENTING LOW PASS FILTER IN MATLAB SIMULINK

A low pass filter is implemented at the end of critical load stabilizer system. The output of single-phase half-bridge inverter is connected to low pass filter. This filter converts the square wave output of inverter to sine wave by blocking high frequency harmonic components. The diagram below shows the schematic of single-phase half-bridge inverter.

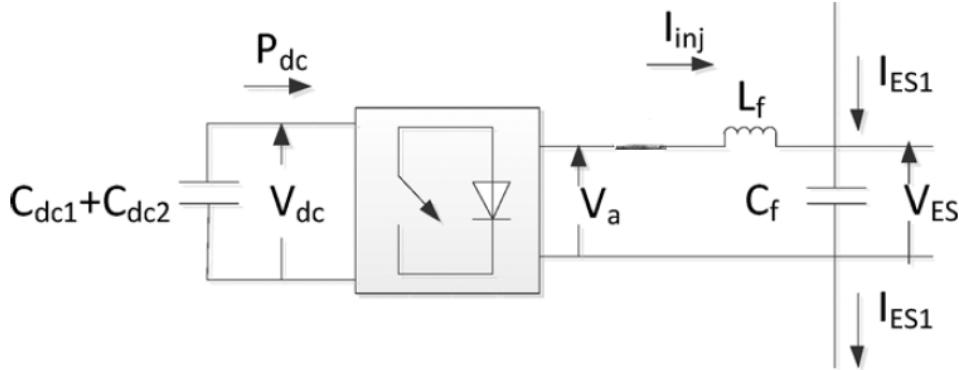


Figure 39: Circuit diagram of half-bridge inverter and low pass filter [10]

Applying KVL on the output of inverter,

$$v_a - v_{es} = L_f * \frac{dI_{inj}}{dt}$$

And applying KCL on the output gives,

$$I_{ES1} + I_{inj} = C_f * \frac{dv_{es}}{dt}$$

The terminal voltage v_a at the output of half bridge inverter can be written as,

$$v_a(t) = \left(\frac{v_{dc}}{2}\right) * PWM - \left(\frac{v_{dc}}{2}\right) * \overline{PWM}$$

Here, PWM is a switching function forcing the switch to change its state. $v_a(t)$ can be further decomposed into fundamental and high frequency harmonic components.

$$v_a(t) = v_{1a}(t) + \sum_{h=1}^{\infty} a_h * \cos\left(\frac{2 * \pi * h}{T_s} * t\right) + b_h * \sin\left(\frac{2 * \pi * h}{T_s} * t\right)$$

$v_a(t)$ averaged over one switching period can be written as

$$\bar{v}_a(t) = \frac{1}{T_s} * \int_0^{T_s} v_{1a}(\tau) d\tau + \sum_{h=1}^{\infty} a_h * \cos\left(\frac{2 * \pi * h}{T_s} * t\right) + b_h * \sin\left(\frac{2 * \pi * h}{T_s} * t\right)$$

Looking at the above equation, there is a need to filter out high frequency components so that PWM inverter can be represented by its fundamental frequency.

$$v_a(t) = \left(\frac{v_{dc}}{2}\right) * m(t)$$

where,

$$m(t) = A * \sin(2 * \pi * 50 * t + \theta)$$

v_{dc} = Inverter DC link voltage

Figure 40 shows the Simulink model of LC low pass filter. The output of the inverter is connected to the filter's inductor which drives the voltage on the filter capacitor. Using the equations derived in chapter 2.2, an inductor of value 0.15 H and a $13.2 \mu F$ capacitor were used. The cutoff frequency of the filter was 113.1 Hz.

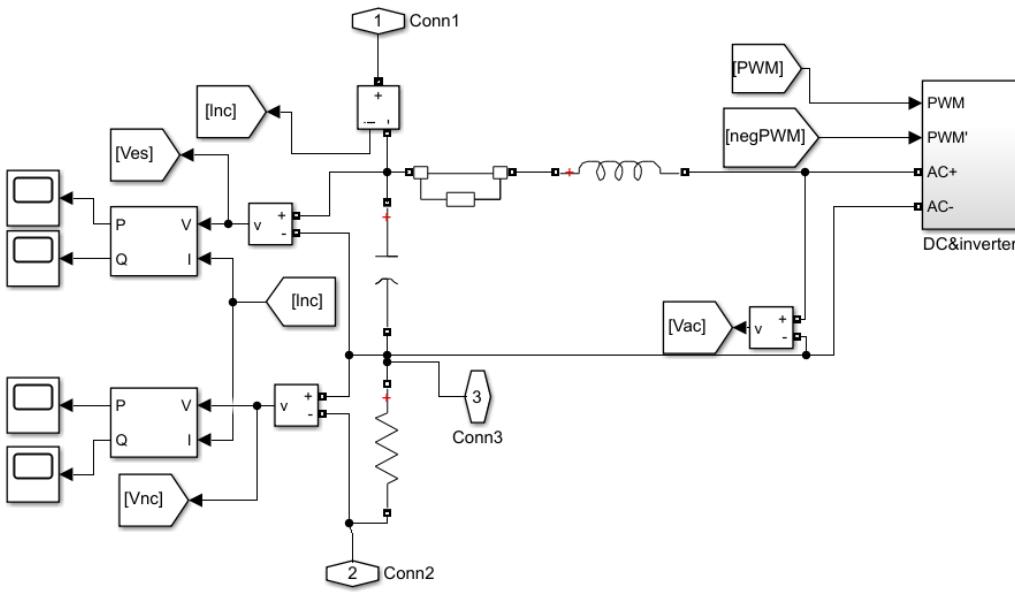


Figure 40: Simulink model of LC low pass filter

The input signal to LC filter is a square wave and the output is a sine wave due to low pass filter implemented above. The output signal resembles a low frequency sine wave but does contain small fluctuations. The reason for fluctuating sine signal is the phase-varying PWM signal driving the half bridge inverter. Based on the critical load voltage error, the phase of PWM signal changes. As a result, the output of PWM inverter contains a varying phase. The input and output signals of LC filter are shown below. The input square wave has a much higher frequency due to the high frequency 20kHz PWM signal operating the half bridge inverter.

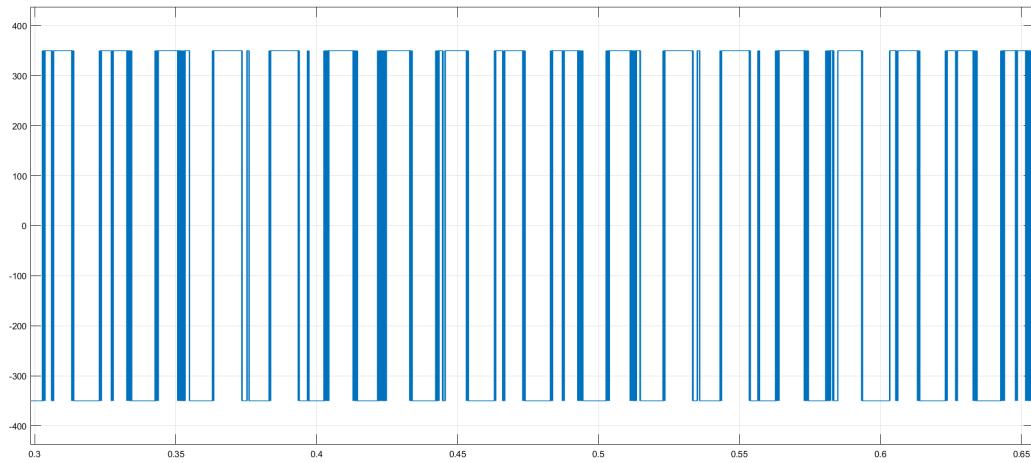


Figure 41: Input of LC low pass filter

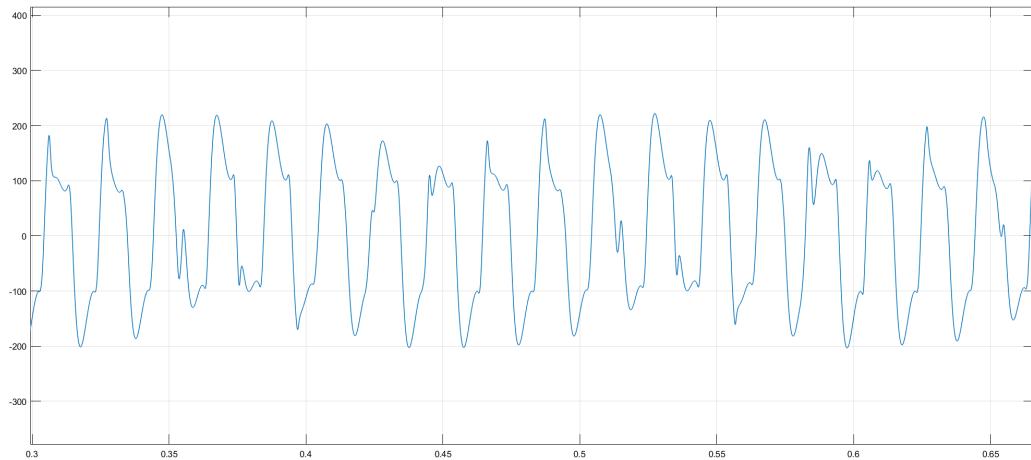


Figure 42: Output of LC low pass filter

CHAPTER 7 – GROUP PROJECT IMPLEMENTATION

In this chapter, the integration of various sub-systems such as PI Controller, PLL and low pass filter in critical load stabilizer will be discussed. It will also describe the implementation and functioning of critical load stabilizer in Simulink.

The first task was to implement renewable energy source in Simulink to imitate random fluctuating behavior of grid voltage. This was achieved by connecting a parallel circuit of inductor and capacitor with grid voltage source v_g . A timed switch was installed in series to alternate between parallel connection of capacitor and inductor. The capacitor was responsible for boosting grid voltage by supplying reactive power whereas the inductor reduced grid voltage by consuming reactive power. The figure below shows the Simulink model of renewable energy source.

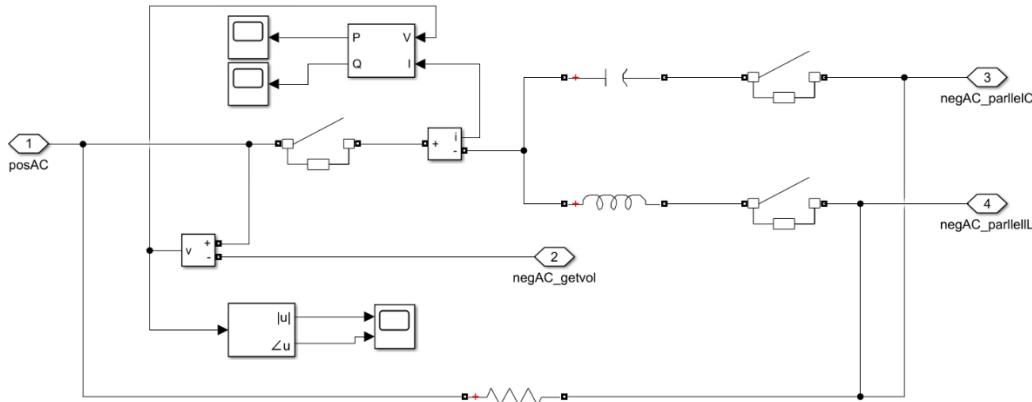


Figure 43: Simulink model of renewable energy source

In the above circuit, the following parameters are used

$$\begin{aligned} C &= 127 \mu F \\ L &= 75 mH \\ R &= 0.1 \mu \Omega \end{aligned}$$

Initially, the LC circuit was disconnected, and no reactive power flowed through the branch maintaining grid voltage at 221.7V. At $t = 0.1s$, the inductor was connected parallel to grid voltage. It absorbed 2000Var causing voltage to drop to 218.2V. At $t = 0.25s$, the inductor branch was disconnected, and the capacitor was connected parallel to grid voltage. The capacitor injected 2000Var boosting the grid voltage to 225.1V. Finally, at $t = 0.4s$, the parallel LC connection was disconnected, and the system operated at its original grid voltage of 221.7V. Therefore, a fluctuation of $\pm 3.5V$ was achieved by injecting $\pm 2000\text{Var}$ in the circuit. The graphs below show the reactive power injected by LC branch and the resulting fluctuation on grid voltage.

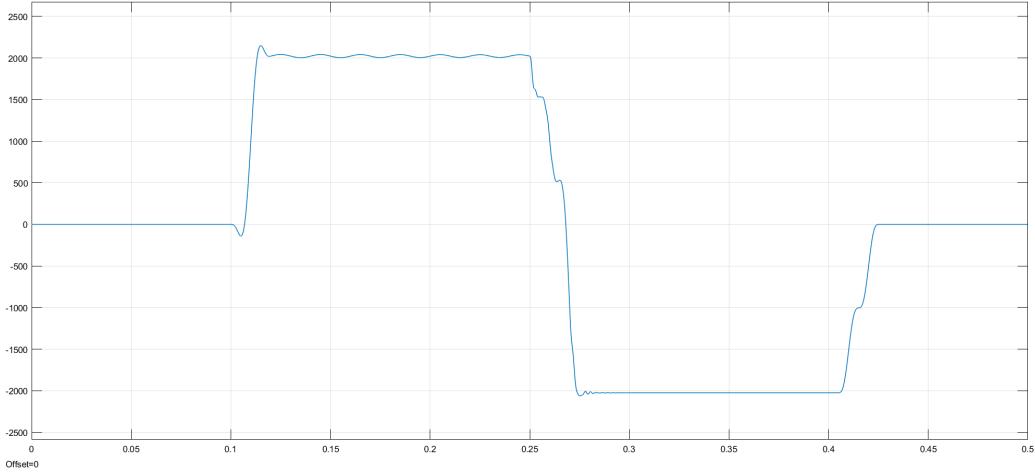


Figure 44: Power profile of renewable energy source

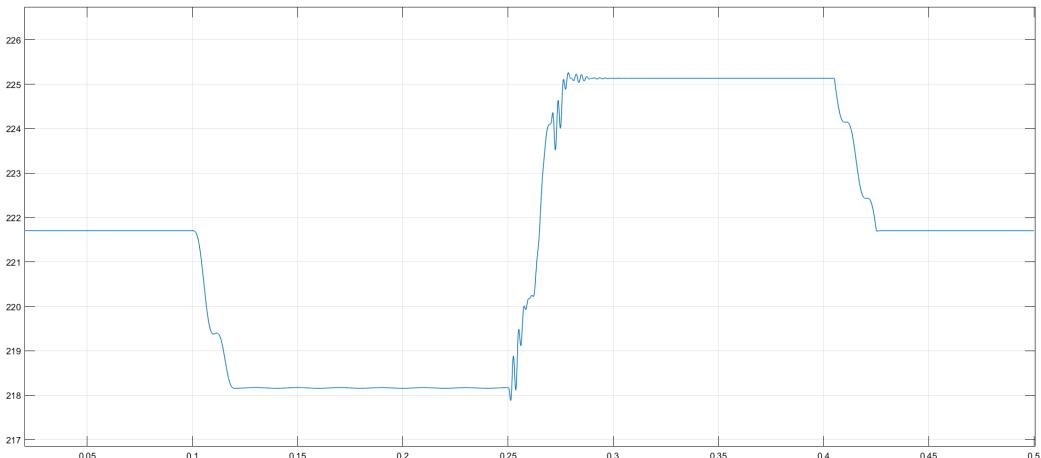


Figure 45: Grid voltage fluctuations due to renewable energy source

To stabilize these fluctuations, a critical load stabilizer was designed and installed in series with non-critical load. The critical load stabilizer system is subdivided into 6 sub-systems – PI compensation controller, PLL block, Sine wave generator, PWM generator, PWM inverter and low pass filter. At the beginning, the error between critical load voltage and reference critical load voltage is calculated and fed as input into the PI controller. The controller is responsible for generating a modulated output which is aimed at driving the error back to zero. The modulated output passes through a limiter to avoid large controller effort and is finally used as magnitude for sine wave. The PLL block measures ωt of non-critical load branch current. The output of PLL block is summed with $\pm \frac{\pi}{2}$ to generate the appropriate phase for sine signal. $\pm \frac{\pi}{2}$ is added to PLL output to generate a sine wave that is either leading or lagging the non-critical load voltage by $\frac{\pi}{2}$. This phase difference ensures that the critical load stabilizer provides full reactive compensation. Thus, the sine wave generator takes in magnitude and phase as input and generates an appropriate sine wave. The PWM generator takes the custom sine wave as input and uses it

as a modulating wave. A triangular wave of 20kHz frequency is used as a carrier wave and is compared with the modulating wave to generate a PWM output. The output of PWM generator, PWM and \overline{PWM} , are treated as inputs for PWM inverter. The PWM inverter comprises of a single-phase half-bridge inverter which is driven by the PWM signal. The DC link voltage is converted to AC square wave at the output of inverter. The square wave passes through an LC low pass filter that blocks all high frequency harmonic components. This results in a low frequency sine wave resembling to that of AC grid voltage. Note that the critical load stabilizer's sine wave is either lagging or leading the non-critical load voltage by $\frac{\pi}{2}$. Therefore, it provides full reactive compensation and stabilizes the critical load voltage. Due to this system being a feedback system, the voltage on critical load is again measured and compared with reference critical load voltage to calculate error signal. The critical load stabilizer system works on this error signal until it is driven down to zero. Figure 46 displays the Simulink model of the entire grid circuit with critical load stabilizer installed.

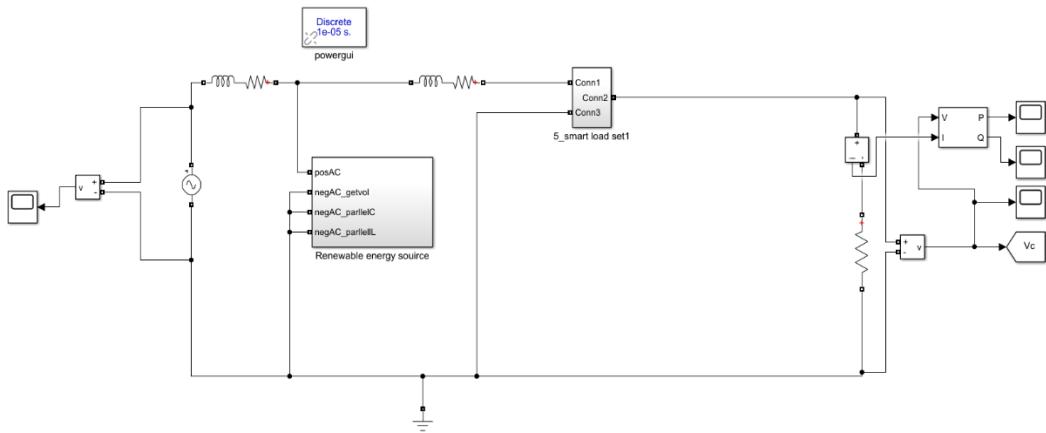


Figure 46: Simulink model of grid circuit with PV penetration and critical load stabilizer

Figure 47 shows the inside of smart load branch consisting of critical load stabilizer and non-critical load branch connected in series.

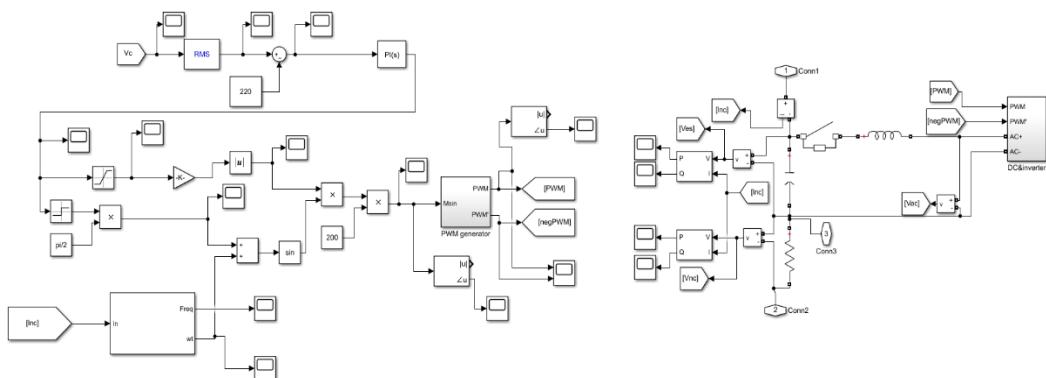


Figure 47: Simulink model of critical load stabilizer

GROUP PROJECT RESULTS & EVALUATION SECTION

CHAPTER 8 – GROUP PROJECT RESULTS AND EVALUATION

8.1 Results and Evaluation

The first design of critical load stabilizer was implemented to provide pure reactive power compensation. A pure reactive model was crucial in delivering an energy-efficient solution. In this model, the critical load stabilizer voltage was always either leading or lagging the non-critical load voltage by $\frac{\pi}{2}$. To test this model, a single critical load stabilizer was connected in series with a pure resistive non-critical load. The reference voltage for critical load was set to be 220V. The initial grid voltage was set to be 221V. Due to voltage drop through transmission line, the initial critical load voltage was 220V. Figure 48 shows the power profile of renewable energy source under inductive mode. At 0.2s, a 0.15H inductor was connected parallel to grid voltage drawing 1000Var from the system.

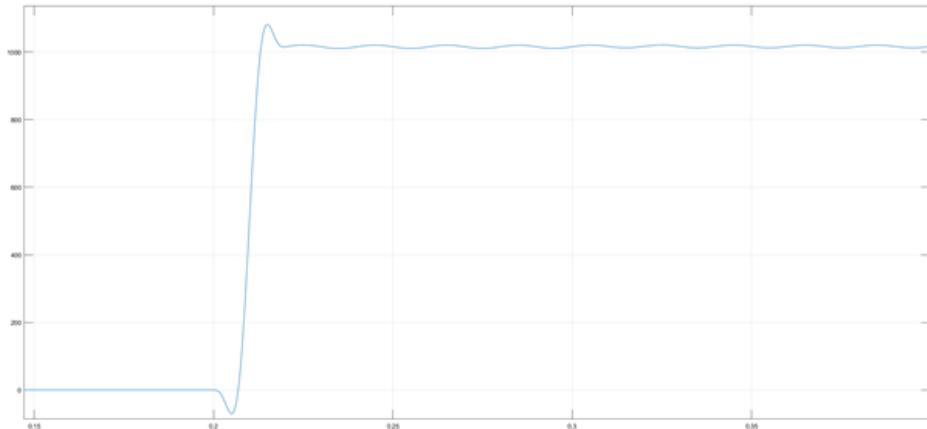


Figure 48: Power profile of renewable energy source under inductive mode

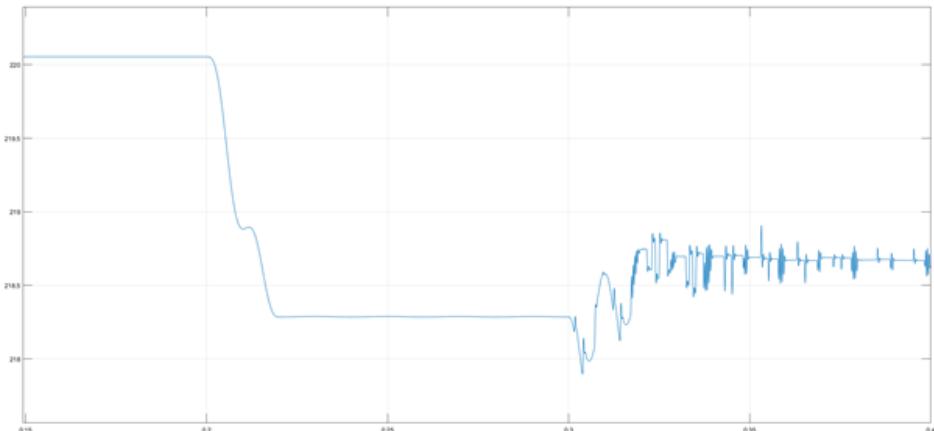


Figure 49: Critical load voltage under inductive mode

Figure 49 displays the critical load voltage in response to Var injection. This caused critical load voltage to drop to 218.2V in absence of critical load stabilizer. At 0.3s, the critical load stabilizer was connected in series with non-critical load and injected positive reactive power to provide voltage support. This resulted in a voltage boost and critical load voltage increased to 218.7V. Figure 50 shows the power profile of renewable energy source under capacitive mode. At 0.2s, a $63.5\mu F$ capacitor was connected parallel to the grid voltage injecting 1000Var into the system.

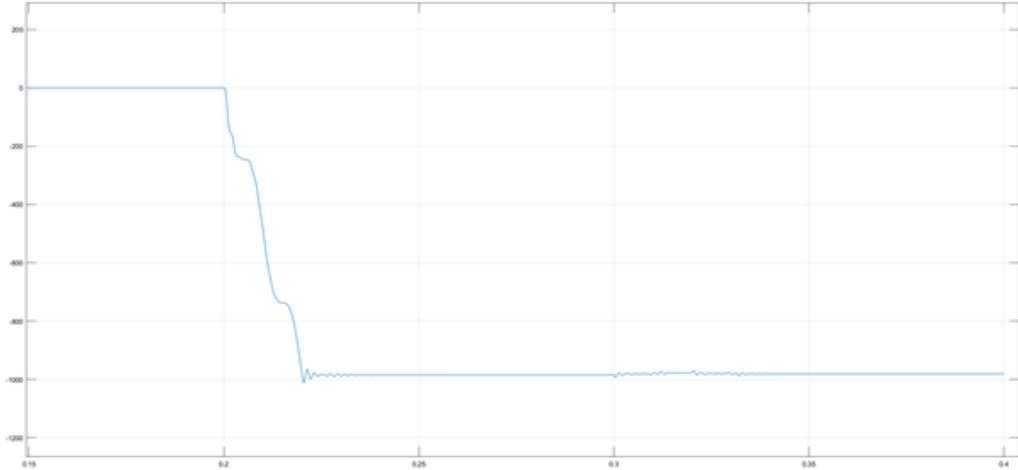


Figure 50: Power profile of renewable energy source under capacitive mode

Figure 51 displays the critical load voltage in response to Var injection. This caused critical load voltage to boost to 221.74V in the absence of critical load stabilizer. At 0.3s, the critical load stabilizer was connected in series with non-critical load and injected negative reactive power to stabilize high voltage. This resulted in a voltage drop and critical load voltage decreased to 220.82V.

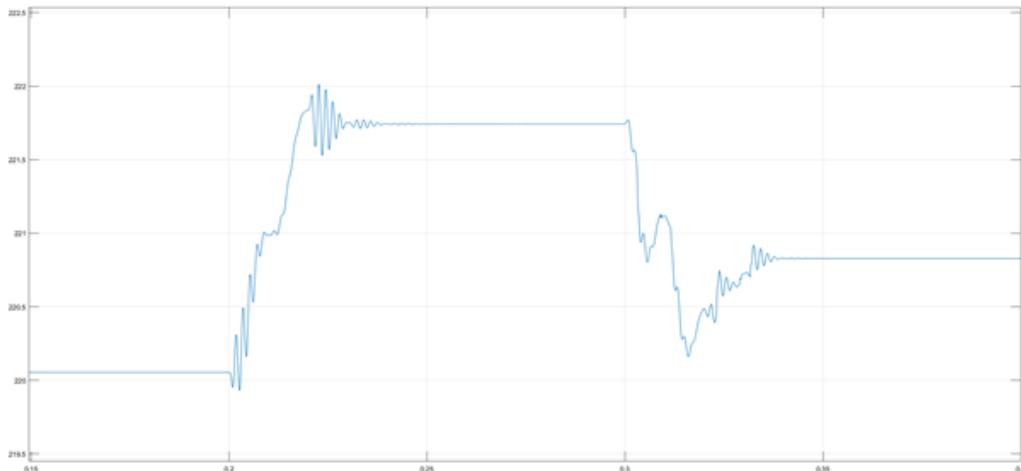


Figure 51: Critical load voltage under capacitive mode

Looking at the results confirmed that in absence of critical load stabilizer, critical load voltage fluctuated above acceptable values and required a system to stabilize it. However, a pure reactive critical load stabilizer was not able to fully stabilize critical load voltage back to original voltage of 220V.

For pure reactive mode, Z_{es} is a pure inductor or a capacitor. Therefore, Z_{es} is a pure imaginary value.

$$\rightarrow Z_{es} = jm \quad (1)$$

where 'm' is a real value.

Therefore, in inductive mode

$$jm = j\omega L$$

$$\rightarrow m = \omega L$$

$$\rightarrow L = \frac{m}{\omega}$$

In capacitive mode,

$$jm = \frac{1}{j\omega C}$$

$$\rightarrow m = -\frac{1}{\omega C}$$

$$\rightarrow C = -\frac{1}{\omega m}$$

From chapter 3, it is known that

$$Z_{load} = \frac{Z_c * (Z_{es} + Z_{nc})}{Z_{es} + Z_{nc} + Z_c} \quad (2)$$

Rearranging the above equation yields,

$$Z_{es} = \frac{Z_c * Z_{nc} - Z_{load} * Z_c - Z_{load} * Z_{nc}}{Z_{load} - Z_c}$$

Replacing Z_{es} in equation (2) with equation (1),

$$Z_{load} = \frac{Z_c * (jm + Z_{nc})}{jm + Z_{nc} + Z_c}$$

Using parameter values for Z_c and Z_{nc} from chapter 3,

$$Z_{load} = \frac{2500 + 50 * jm}{100 + jm}$$

From chapter 3, it is known that critical load voltage is a function of Z_{load}

$$v'_c = \frac{v_g * Z_{res} * Z_{load}}{Z_{l_1} * (Z_{l_2} + Z_{load} + Z_{res}) + Z_{res} * (Z_{l_2} + Z_{load})}$$

Thus, critical load voltage is limited by the capacitance or inductance value. The graph shown below displays the relationship between the capacitance value and critical load voltage. It clearly shows that full compensation by a pure reactive model is not achievable. The maximum compensation is up to 219.6V when critical load stabilizer's capacitor has a capacitance of $49.1\mu F$

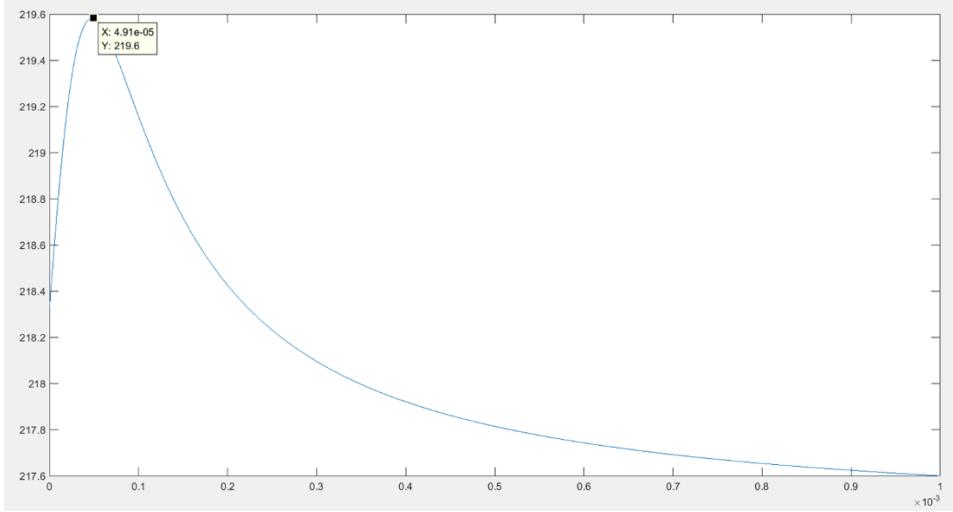


Figure 52: Graph displaying capacitance vs critical load voltage

To overcome the above issue, an active-reactive power compensation model was tested. In this circuit, the critical load stabilizer voltage leads or lags the circuit by any angle rather than only $\frac{\pi}{2}$. The same testing setup as above was used for this model. Initially, the system was injected with -1000Var using a 0.15H inductor parallel to grid voltage source. This caused a voltage drop in critical load voltage. To provide voltage support, the stabilizer injected positive active-reactive power into the system. Figure 53 displays critical load voltage in response to Var injection under inductive mode.

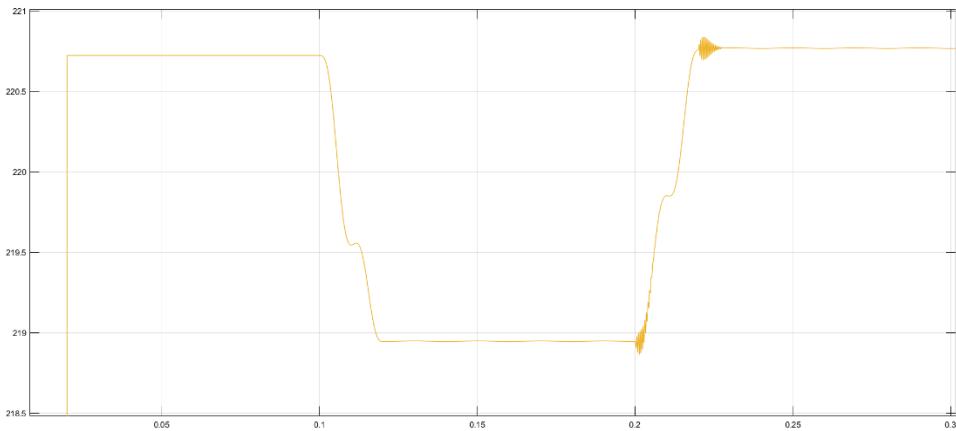


Figure 53: Critical load voltage under inductive mode (active-reactive stabilizer)

The system was also tested under capacitive mode where it was injected with 1000Var. This caused voltage to boost up to 222.4V. The critical load stabilizer injected negative active-reactive power to

stabilize the system. Figure 54 displays critical load voltage in response to Var injection under capacitive mode.

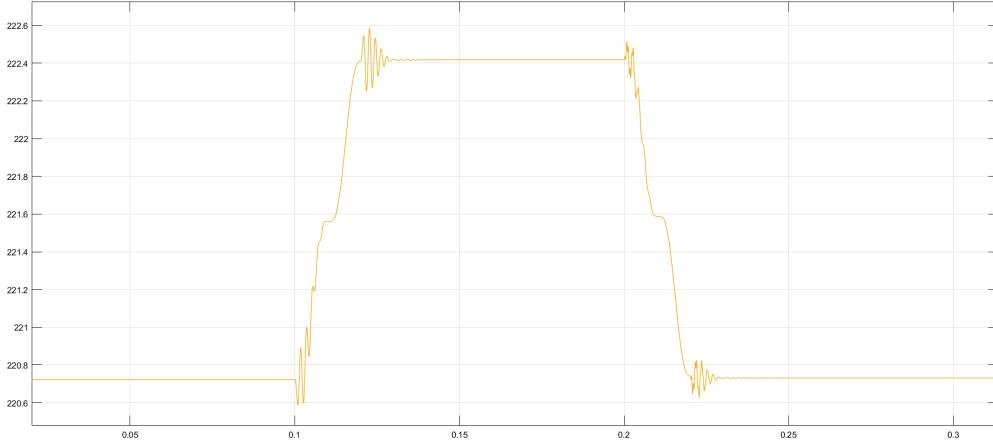


Figure 54: Critical load voltage under capacitive mode (active-reactive stabilizer)

As seen from the graphs above, an active-reactive power compensation model is capable of stabilizing critical load voltage back to original safe value. However, this model possesses a drawback. In order to fully stabilize the voltage, critical load stabilizer needs to provide a huge voltage. The PWM inverter requires a large DC link voltage to suppress fluctuations. In the above case, to stabilize a fluctuation of 1.5V, an 800V DC link voltage was required. The following equation explains the relation between DC link voltage, inverter output voltage and magnitude of sinusoidal signal.

$$v_i = \frac{m * v_{DC}}{2\sqrt{2}}$$

$$\rightarrow m = \frac{2\sqrt{2} * v_i}{v_{DC}}$$

$$\rightarrow v_{DC} = \frac{2\sqrt{2} * v_i}{m}$$

Chapter 3 explained the derivation of v_i i.e. PWM inverter AC voltage. In order to produce an appropriate v_i , magnitude of sine wave must be calculated using the above formula. As magnitude of sine wave, m , must always be between 0 and 1, the minimum value for v_{DC} needed to generate v_i must be $2\sqrt{2} * v_i$. In the above experiment, $v_i = 266.7V$ was required to compensate a voltage drop of around 1.5V. Therefore, a minimum DC link voltage of 754.34 was required. The above limitation forces the critical load stabilizer to possess a huge DC source to compensate large fluctuations.

8.2 Future Improvements

Observing the high DC source need of active-reactive model, a new model called branched reactive model was tested. In this model, a number of smart load branches, each containing a reactive critical load stabilizer were installed parallel to one another. The motive of this model was to provide full reactive compensation but with low DC source. As seen from previous experiments, a single branch of critical load

stabilizer requires high DC source to fully compensate fluctuations and a pure reactive model was not able to provide full support. Therefore, each branch contains a small DC source and therefore all the branches collectively possess enough power to fully compensate voltage fluctuations. In this experiment, 5 branches of smart load were connected in parallel to critical load. Each one of them contained a 200V DC source and followed a pure reactive phase control paradigm. The graph shown below displays the critical load voltage in branched reactive model under Var injection.

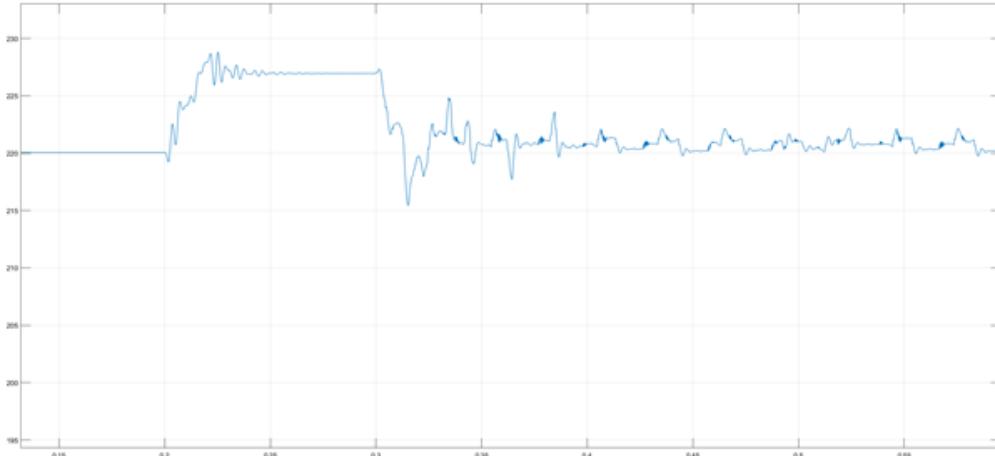


Figure 55: Critical load voltage under capacitive mode (branched reactive stabilizer)

-4000Var was injected into the system which led to a voltage rise of about 7V. After the branched stabilizer was turned on at 0.3s, the voltage dropped down to approximately 221V. It can be observed that voltage stabilization effect of this model is much better than a single-branch pure reactive stabilizer model. Further improvements can be made to this model. In the future, we would like to create a general mathematical model for an n-branched critical load stabilizer model. This model would explain the effects of different component parameters on the branched model. It would also help to determine the effect of adding smart load branches and achieve an ideal configuration for the branched model. Finally, it would lead to better pure reactive stabilization and provide a stable energy-efficient compensation solution.

CHAPTER 9 – CONCLUSION

A simulation model of critical load stabilizer capable of protecting critical load from fluctuations was developed in MATLAB Simulink. It converted non-critical load into a smart load making it follow grid power generation. In this paper, 3 different simulation models were implemented and tested to evaluate their performance.

First, a pure reactive power compensation model of critical load stabilizer was developed. A mathematical model was developed using circuit analysis and later implemented in Simulink for verification. Pure reactive compensation was achieved by maintaining the stabilizer voltage at either $\pm\frac{\pi}{2}$ with respect to non-critical load voltage. It was observed that the stabilization capability of this model was limited due to capacitance value of the stabilizer. Therefore, it was only capable of compensating small fluctuations in the power grid.

To alleviate the above problem, an active-reactive power compensation model of critical load stabilizer was developed. Using the mathematical model, the source of fluctuation in the grid was modeled as an impedance source. From the derived equations, unstable impedance source was observed as a function of critical load voltage. After detecting the correct impedance for renewable energy source, appropriate phase was calculated for the stabilizer. The limitation of this model was the requirement of huge DC source in PWM inverter system of the stabilizer. It was observed that an approximate 800V DC source was required to compensate a fluctuation of +1.5V.

Lastly, to improve upon the previous two models, a branched pure reactive power compensation model of critical load stabilizer was developed. It was tested using a -4000Var renewable energy source and 5 non-critical smart load branches. It achieved a compensation of approximately 6V displaying much better results than a single-branched pure reactive stabilizer model. In future, a mathematical model for an n-branched stabilizer system can be created and tested. It would help to derive the ideal configuration for the stabilizer and achieve the most energy-efficient compensation model. Observing the results from this report, the proposed simulation model can be used by researchers and companies to build and evaluate a physical model of critical load stabilizer for future smart grids with high PV penetration.

REFERENCES

- [1] <https://ocean.si.edu/through-time/ancient-seas/sea-level-rise> [Accessed April 1, 2019].
- [2] <http://www.altenergy.org/renewables/renewables.html> [Accessed April 1, 2019].
- [3] <https://oilprice.com/Energy/Energy-General/How-Intermittent-Renewables-Are-Harming-The-Electricity-Grid.html> [Accessed April 4, 2019].
- [4] <http://www.ferc.gov/legal/staff-reports/12-20-12-demand-response.pdf> [Accessed April 4, 2019].
- [5] <http://www.ewh.ieee.org/sb/iiee/new/tutorials/feedback.pdf> [Accessed April 6, 2019]
- [6] Someshwar C. Gupta, "Phase-Locked Loops", *Proc. IEEE*, vol. 63, No. 2, February 1975, pp. 291-302.
- [7] Hyosung Kim and Seung-Ki Sul, "Analysis on output LC filters for PWM inverters", *Proc. 2009 IEEE 6th International Power Electronics and Motion Control Conference*, Wuhan, China, May 2009.
- [8] G. Mallesham, S. Mishra and A. N. Jha, "Zeigler-Nichols Based controller parameters tuning for load frequency control in a microgrid", *Proc. IEEE 2011 International Conference on Energy, Automation and Signal*, Bhubaneshwar, Odisha, India, December 2011.
- [9] Asfaw Gezae Daful, "Comparative study of PID tuning methods for processes with large & small delay times", *Proc. IEEE 2018 Advances in Science and Engineering Technology International Conferences (ASET)*, Abu Dhabi, United Arab Emirates, June 2018.
- [10] Nilanjan Ray Chaudhari, Chi Kwan Lee, Balarko Chaudhari and S. Y. Ron Hui, "Dynamic Modeling of Electric Springs", *Proc. IEEE Transactions on Smart Grid*, vol. 5, No, 5, September 2014, pp. 2452-2543.