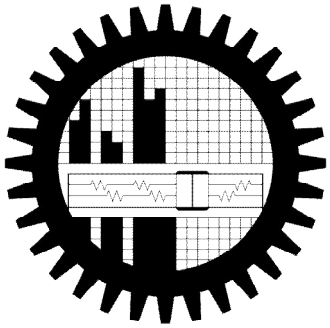
CLOSED LOOP ĈUK TOPOLOGY BASED SINGLE PHASE HIGH PERFORNACE AC-DC CONVERTER

By

Md. Ismail Hossain



DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY

2013

CLOSED LOOP ĈUK TOPOLOGY BASED SINGLE PHASE HIGH PERFORNACE AC-DC CONVERTER

By

Md. Ismail Hossain

A thesis Submitted to the Department of Electrical and Electronic Engineering of Bangladesh University of Engineering and Technology in partial Fulfillment of the Requirement for the Degree of

MASTER OF SCIENCE IN ELECTRICAL AND ELECTRONIC ENGINEERING

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY

September 2013

APPROVAL

This thesis titled **"Closed loop Ĉuk topology based single phase high performance AC-DC converter"** submitted by Md. Ismail Hossain, Roll No. 0411062239P, Session: April, 2011 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of **Master of Science in Electrical and Electronic Engineering** on 1 September, 2013.

BOARD OF EXAMINERS

1. _____
Dr. Mohammad Jahangir Alam — **Chairman**
Professor — **(Supervisor)**
Department of EEE
BUET, Dhaka

2. _____
Dr. Pran Kanai Saha — **Member**
Professor and Head — **(Ex-officio)**
Department of EEE
BUET, Dhaka

3. _____
Dr. Mohammad Ali Choudhury — **Member**
Professor
Department of EEE
BUET, Dhaka

4. _____
Dr. Md. Raju Ahmed — **Member**
Assistant Professor — **(External)**
Department of EEE
DUET, Gazipur

CANDIDATE'S DECLARATION

This is hereby declared that this thesis titled **"Closed loop Ĉuk topology based single phase high performance AC-DC converter"** or any part of it has not been submitted elsewhere for the award of any degree or diploma.

Signature of the candidate

_____

Md. Ismail Hossain

DEDICATED

TO

MY PARENTS AND MY WIFE

NAZNIN AKTER

TABLE OF CONTENTS

Chapter 3: Real time Program Algorithm and Simulation Using Proteus software through DSP processor

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATION

SMPS	= Switch mode power supply
MOSFET	= Metal oxide semiconductor field effect transistor
PWM	= Pulse Width Modulation
OPAMP	= Operational Amplifier
rms	= Root mean square
THD	= Total Harmonic Distortion
PF	= Power Factor
DSP	= Digital signal processor
DSC	= Digital signal controller
ADC	= Analog to Digital Converter
PI	= Proportional Integral
IGBT	= Insulated Gate Bipolar Transistor
SSA	= State Space Averaging
PFC	= Power Factor Correction
MIPS	= Million Instructions per Second

ACKNOWLEDGEMENT

I would like to express my sincere appreciation to my supervisor, Dr. Mohammad Jahangir Alam, Professor Department of Electrical and Electronic Engineering, BUET, Dhaka for his guidance, encouragement and continuous support.

I also extend my appreciation to **Professor Dr. Mohammad Ali Choudhury** Department of Electrical and Electronic Engineering, BUET for his valuable suggestion, comments and discussions. My thanks are to all staffs of BUET central Library and EEE departmental Library for their cooperation and material support.

Finally I acknowledge to my parents and my wife Naznin Akter for their constant inspiration towards the completion of this work.

ABSTRACT

Single phase rectifiers are commonly used in low and medium power application due to their low cost and ease of control. Rectifiers are non-linear in nature and draw non sinusoidal input current from the AC power sources. This causes a number of problems in sensitive electronics equipments and in the power distribution networks. Moreover it increases in reactive power drawn and reduce the input power factor. To reduce these problems switch mode regulated rectifiers are used. This thesis focuses on the study and digital implementation of a power factor correction (PFC) converter using close loop Ĉuk topology. This thesis work controls output voltage of a rectifier by inner current control based power factor correction (PFC). It provides high efficiency (more than 80%), less input current THD, nearly unity power factor and better output voltage regulation of AC-DC converter under variable input voltage, frequency and output load. Small signal model analysis is carried out through state space averaging technique and mathematical expression is developed for obtaining frequency response and checking the stability of the system. An algorithm is developed for digital implementation of proportional integral control. It presents a customized simulink model to represent the whole system and then an algorithm for digital implementation is developed. It also shows the real time simulation of simulink model through voltage sensor, current sensor and Digital signal controller. Finally real time C and assembly code for programming the Digital signal controller is presented.

CHAPTER -1

INTRODUCTION

1.1 Introduction

Uses of Computer and telecommunications equipment have become important in our society. Electric vehicle is rapidly increasing in use due to its environment friendly technology. DC power supply is the heart of these devices. DC supplies of these mostly have come from AC supply. The efficiency, total harmonic distortion of input current, input power factor and regulated output voltage etc are the main concern of these AC-DC converters.

Uncontrolled diode rectifiers followed by L-C smoothing filters are widely used as a cheap power supply. Un-controlled charging of DC filter capacitor results in fifty hertz pulsed ac current waveform at the input of the rectifier. Several power quality problems arise at the source side, which includes poor power factor, high input current total harmonic distortion (THD), failure of transformers due to overheating and harmonic pollution on grid [1, 2] etc. Grid disturbances may result in malfunction or damage of electrical devices. Many methods for elimination of harmonic pollution in the power system are in use and new methods are being investigated. Restrictions on current and voltage harmonics are maintained in many countries through IEEE 519-1992 and IEC 61000-3-2/IEC 61000-3-4[3] standards. The restrictions are associated with the idea of “clean power”. The power factor correction (PFC) converter topology using active wave shaping techniques can overcome the problem in line current. The PFC converter forces the line to draw near sinusoidal AC current in phase with its voltage.

1.2 Background

Most of the single phase AC to DC conversion PFC works are done with Buck, Boost, and buck-boost or fly-back topologies [4-14] between the source and the load. Very few works are done with Ĉuk topology and the reported ones are based on open loop fixed duty cycle [15-19] control.

Regardless of the input line voltage and output load variations, input current drawn by the buck or buck-boost converter is always discontinuous [4-14]. The input current in Ĉuk converter is comparable to boost converter's input current [6]. Since in Boost topology large value capacitance is required to minimize the output voltage ripple therefore initial inrush current is higher than Ĉuk converter [20-21]. Ĉuk converters require low value intermediate capacitance to transfer energy to output capacitance and load [22-23].

As for applications, the Ĉuk converters is widely used in the industries such as wind energy [24], photovoltaic power system [24-27], electrical vehicle [27-28], radar transmission and receiving [29], light emitting diode driver [30], telecommunication systems [31], energy harvesting from exercise bicycles [32], and compressor and motor controllers [33-34].

Open loop Ĉuk topology provides ripple and variation in output voltage and also phase differences between input voltage and input current under variable input voltage [15-19, 35-37]. In this thesis work output voltage and input current control variable duty cycle Ĉuk converter has been proposed for minimizing output voltage variation due to line voltage and load change and obtaining near sinusoidal AC mains current.

1.3 Brief Review of the Digital Signal controller

Digital signal controller (DSC) is widely used in the motor control, inverter, converter and many embedded systems because of its ability to handle deterministic operation and deal with interruptions, and its strong computation ability.

As the technology advances, the cost continues to drop and more and more functionalities have been integrated. Analog-to-digital converter (ADC) and pulse-width-modulation (PWM) are standard for most motor control-oriented DSC.

Currently DSCs are being marketed as green technologies for their potential to reduce power consumption in electric motors and power supplies. In order of market share, the top three DSC vendors are Texas Instruments, Freescale, and Microchip Technology, according to market research firm Forward Concepts 2007. These three companies dominate the DSC market, with

other vendors such as Infineon and Renesas taking a smaller slice of the pie. Table 1 shows current DSC Microchip product feature and application [39].

Table 1 Microchip product feature and application

Enabling Features of the microchip dsPIC DSC	**Applications**
• 16 bit operation • 1 or 2 fault pins • 28-, 40-, 64-, 80- and 100-pin variants • 6 or 8 motor PWM output • Complementary or independent PWM • Center-aligned or edge-aligned PWM • A/D sampling synchronized to PWM cycle • 10-bit, 1 to 2.2 Msps A/D converter • 2, 4 or 8 simultaneous A/D samples • 5V native operation for noisy environments (dsPIC30F) • On-chip Quadrature Encoder Interface (QEI) • Motor control algorithm reference designs • Up to 2 programmable dead time settings • One or two CAN/ECAN 2.0B modules • LIN support through UART and software • SPI™, I2C™ and UART communication ports • Visual design guide motor control GUI (MPLAB plug-in) • 8 channel DMA controller • 40 MIPS @3.3V • -40° to 85°C (extended temp. planned) • Up to 256 KB Program Flash • Data EEPROM emulation in Program Flash	•Heating, ventilation and air conditioning • Absolute encoders and resolvers • Blowers and lawn equipment •Electronic Power Steering •Industrial gate openers •Seat belt tensioners •Exercise equipment •Washing machines •Sewing machines •Industrial pumps •Stability control •Power tools • Refrigeration • Printing machines •Power and environment monitor in servers •Power management for equipment •Circuit breakers •Arc fault detection •Auxiliary power unit •Electric vehicles • AC to DC converters •DC to DC converters •Power factor correction •Inverters • Online UPS •Welding machines • Electrically assisted hydraulic steering • Electronic clutch and gearboxes • Roll and stability controllers • Seat belt pretensioners • Electronic power steering • Cabin noise cancellation • Advanced battery monitors • Airbag main controllers • Ignition controllers • Side impact airbags • Occupant sensors • Fuel pressure controls

The Texas product TMS320C28x generation of digital signal controllers is the industry's first 32-bit DSP-based controllers with on-board Flash memory and performance up to 150 MIPS. They target industrial control, optical networking, and automotive control applications. The C28x core is a high performance control optimized controller and offers up to 150 MIPS of computational bandwidth to handle numerous sophisticated control algorithms in real-time, such as sensorless speed control, random PWM, and power factor correction[40]. Due to rapid advances of Digital signal controller and lower cost it is better to use it in PFC controller implementation as an alternative of analog circuit for getting more reliable and high performance.

1.4 Goal of Thesis

The goal of this thesis is to design the control strategy of a single phase AC to DC converter using close loop ĈUK topology which will ensure high power factor, less input current THD, high efficiency and regulated DC voltage for variable line voltage, frequency and load. Matlab simulink is to be used to analyze the result and performance. Finally real time program algorithm is to be designed for digital implementation and its effectiveness will be evaluated through Proteus software.

1.5 Thesis Organization

Chapter-1 is the introduction to this thesis, and provides a summary of the background of AC-DC converter and Digital signal controller. This chapter also includes the objective of the thesis work.

Chapter-2 surveys the performance of full bridge diode rectifier and open loop Ĉuk rectifier by considering efficiency, input current THD, PF and output voltage regulation. This chapter deals with the goal of this thesis work and provides detail description of Close loop Ĉuk based AC-DC converter. It presents the Matlab simulink model of proposed system and simulation result. It provides customized user friendly simulink model to reduce the system complexity and quick parameters changing option. It also presents a mathematical model for stability analysis using bode plot of outer voltage loop and inner current loop. The small signal transfer function for Ĉuk converter through state space averaging technique is described in this chapter.

Chapter-3 describes the real time program algorithm for digital implementation, analog to digital conversion, PWM generation and complete circuit diagram for mixed signal (Analog and digital) simulation. This chapter includes the performance result, graphical simulation under the variable input voltage, input frequency and output load. The chapter also provides the relative performance comparison of full bridge diode rectifier, open loop Ĉuk rectifier and proposed close loop Ĉuk rectifier.

Chapter-4 concludes the thesis with summary and suggestion of future work. Real time program mixing with C and Assembly language has been given in Appendix and also Matlab program is given in Appendix for stability analysis.

CHAPTER -2

CONVENTIONAL AND ĈUK OPEN/CLOSED LOOP SINGLE PHASE AC-DC RECTIFIER

2.1 Full bridge Diode rectifier

Full wave diode rectifier is versatile and is used AC to DC converter. It's efficiency and input power factor are low with high input current THD. Since it is uncontrolled rectifier so output voltage varies with change of input voltage and load. Fig. 2.1 shows a full bridge diode rectifier based AC-DC converter.

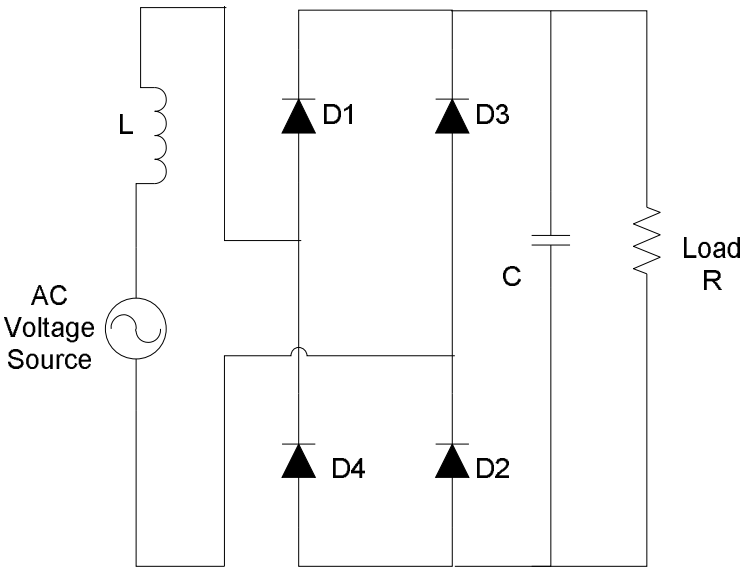


Fig 2.1 AC to DC converter using Full bridge Diode rectifier

L= 1mH and C= 1000uF are used for simulation. Fig. 2.2 to Fig. 2.4 show input voltage/current, FFT component and output voltage respectively for 400V reference output voltage at 400Ω load resistance, 410V (peak) input voltage and 50Hz supply frequency. Similarly Fig. 2.5 to Fig. 2.7 for 267Ω load resistance and Fig. 2.8 to Fig. 2.10 for 200Ω load resistance. Fig. 2.11 to Fig. 2.13 show the output voltage for different input voltages and 400Ω load resistance.

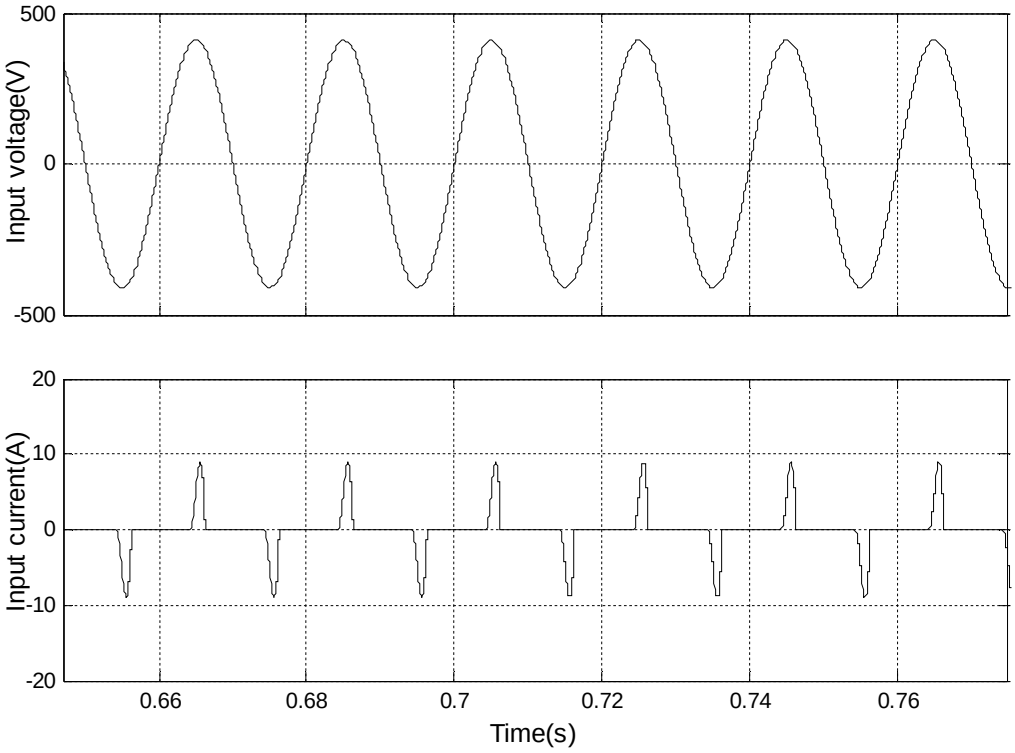


Fig 2.2 Input current and input voltage for 400Ω load resistance

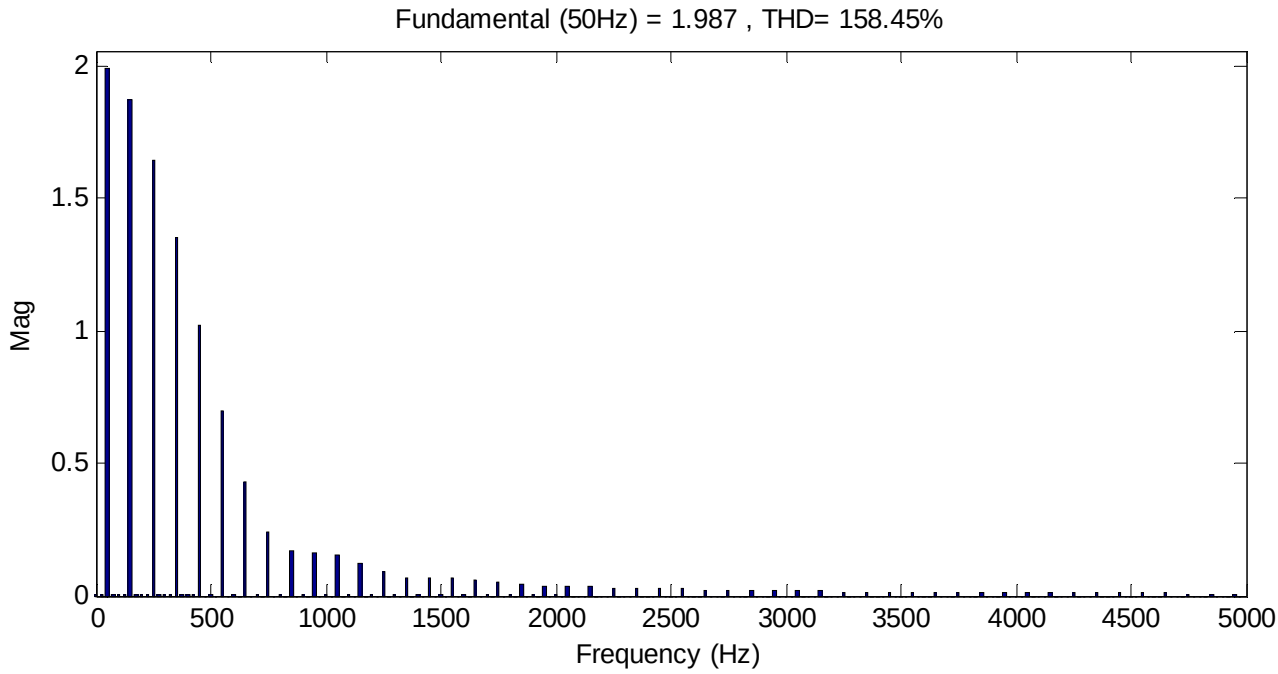


Fig 2.3 FFT component of input current for 400Ω load resistance and 410V (peak) input voltage.

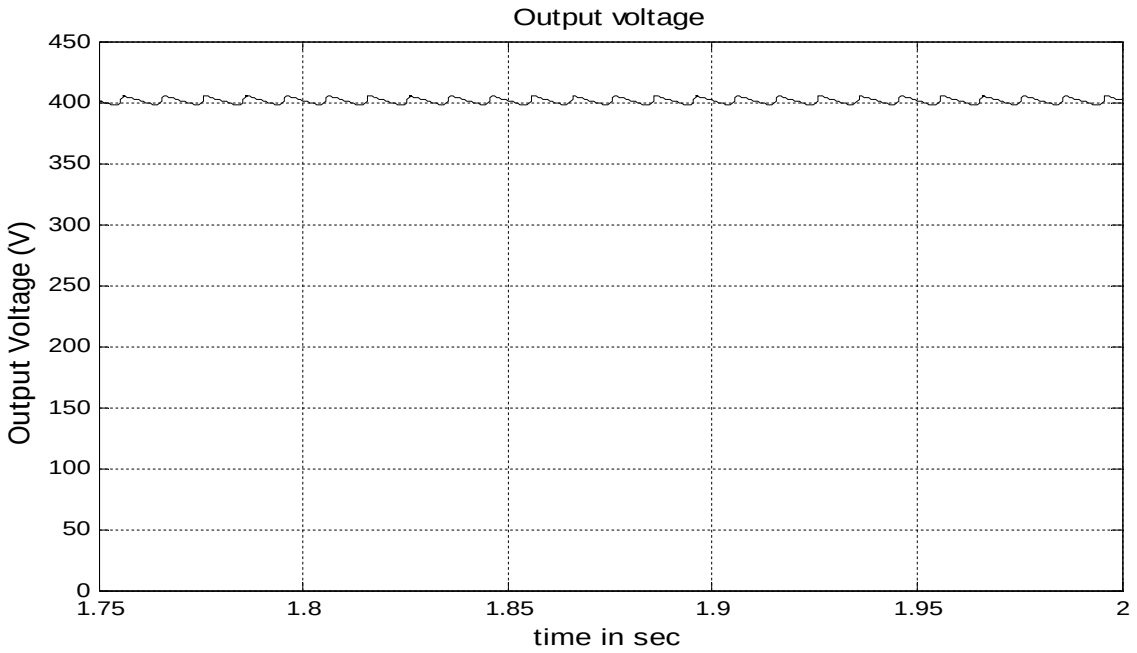


Fig 2.4 Output voltage for 400Ω load resistance and 410V (peak) input voltage.

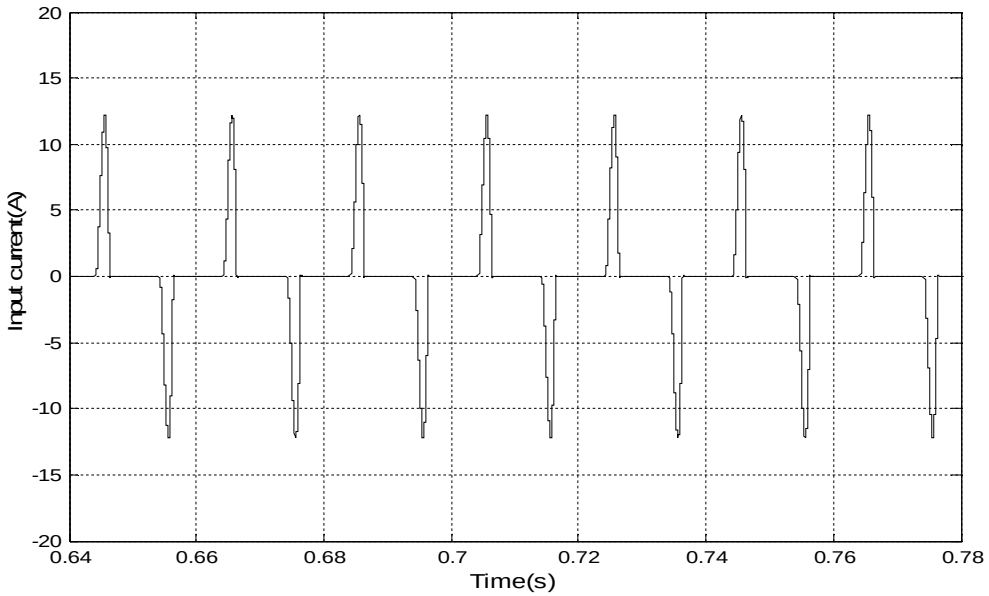


Fig 2.5 Input current for 267Ω load resistance and 410V (peak) input voltage.

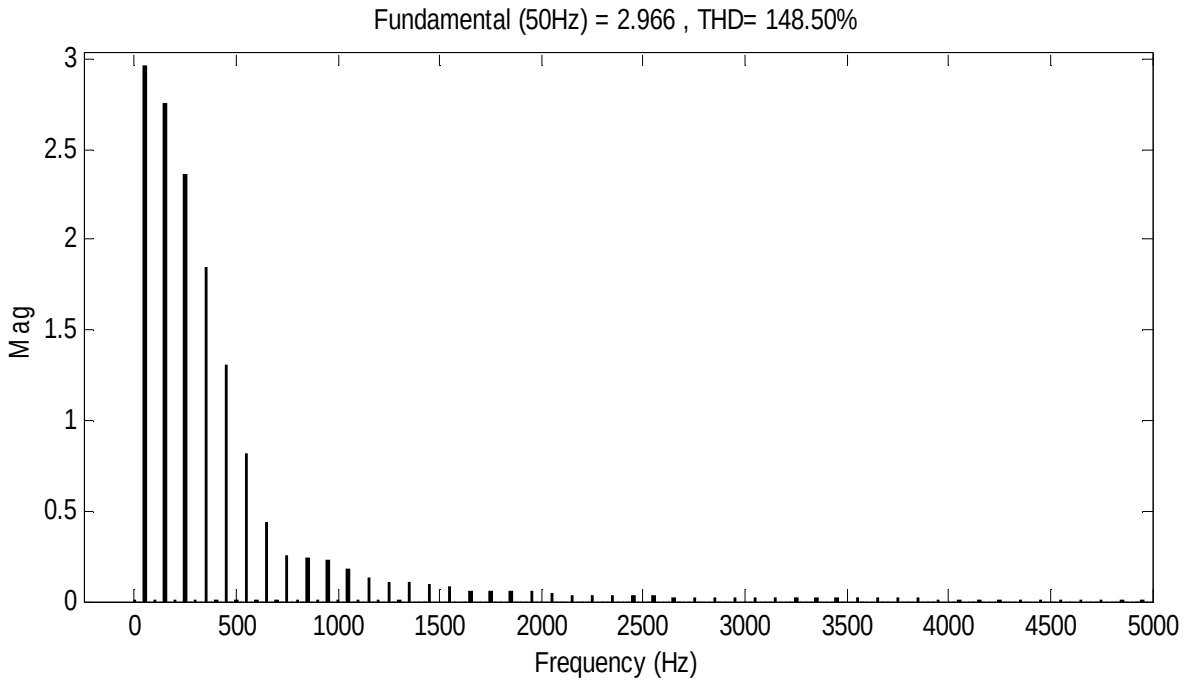


Fig 2.6 FFT component of input current for 267Ω load resistance and 410V (peak) input voltage.

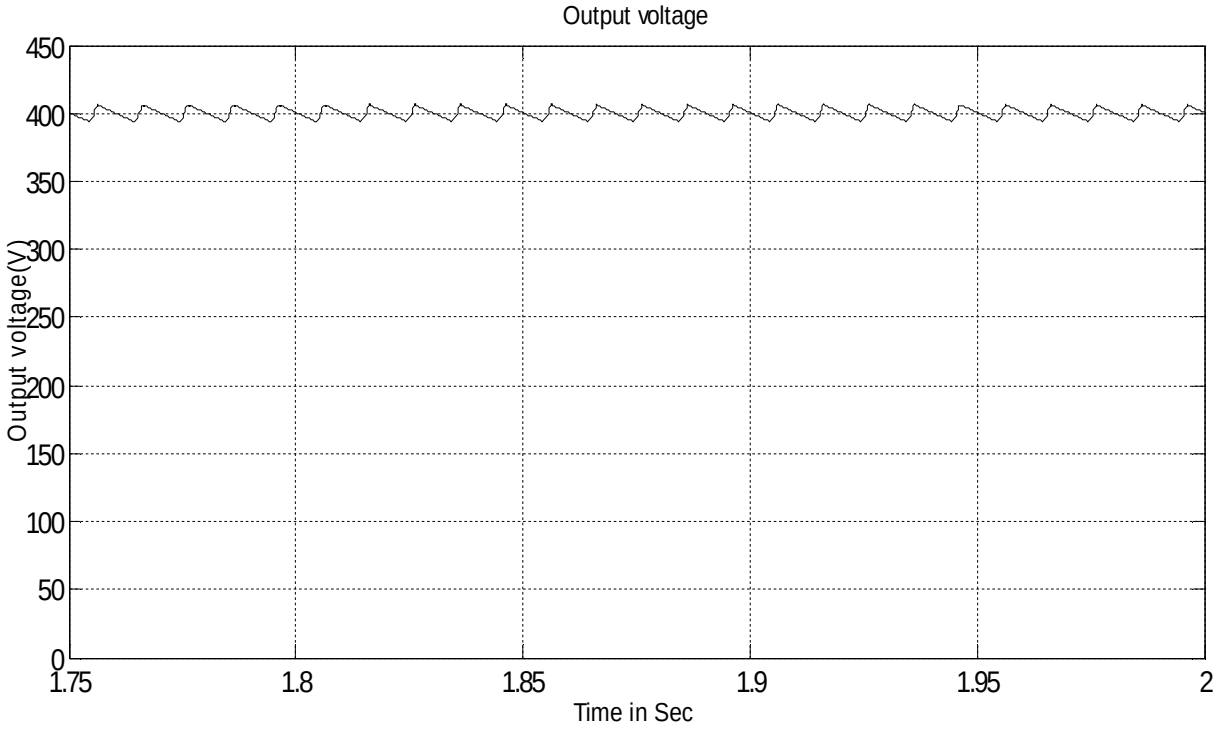


Fig 2.7 Output voltage for 267Ω load resistance and 410V (peak) input voltage.

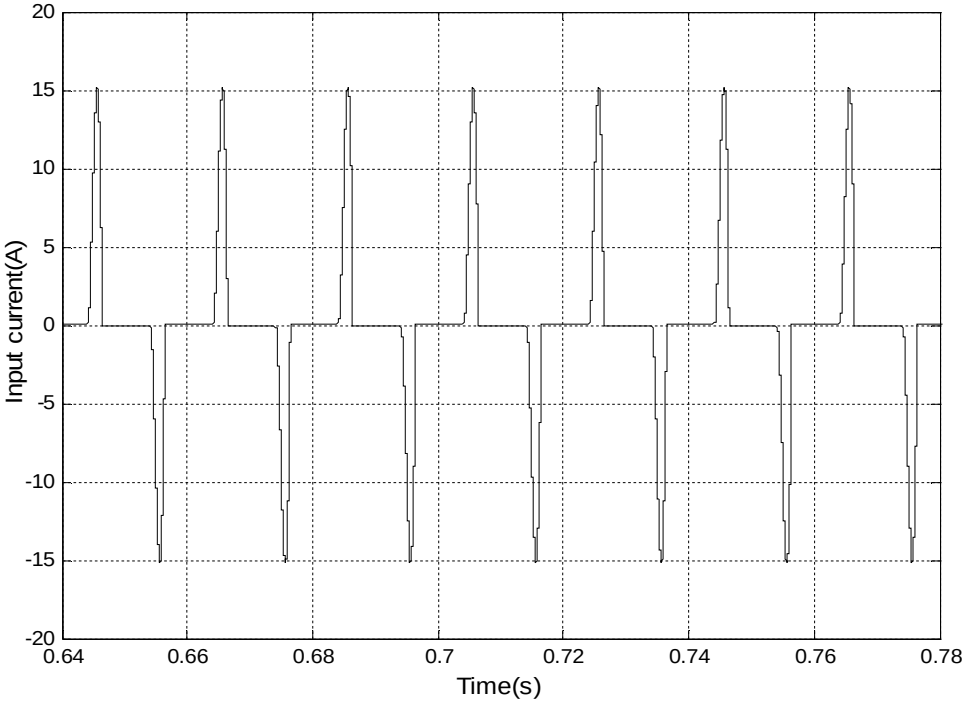


Fig 2.8 Input current for 200Ω load resistance and 410V (peak) input voltage.

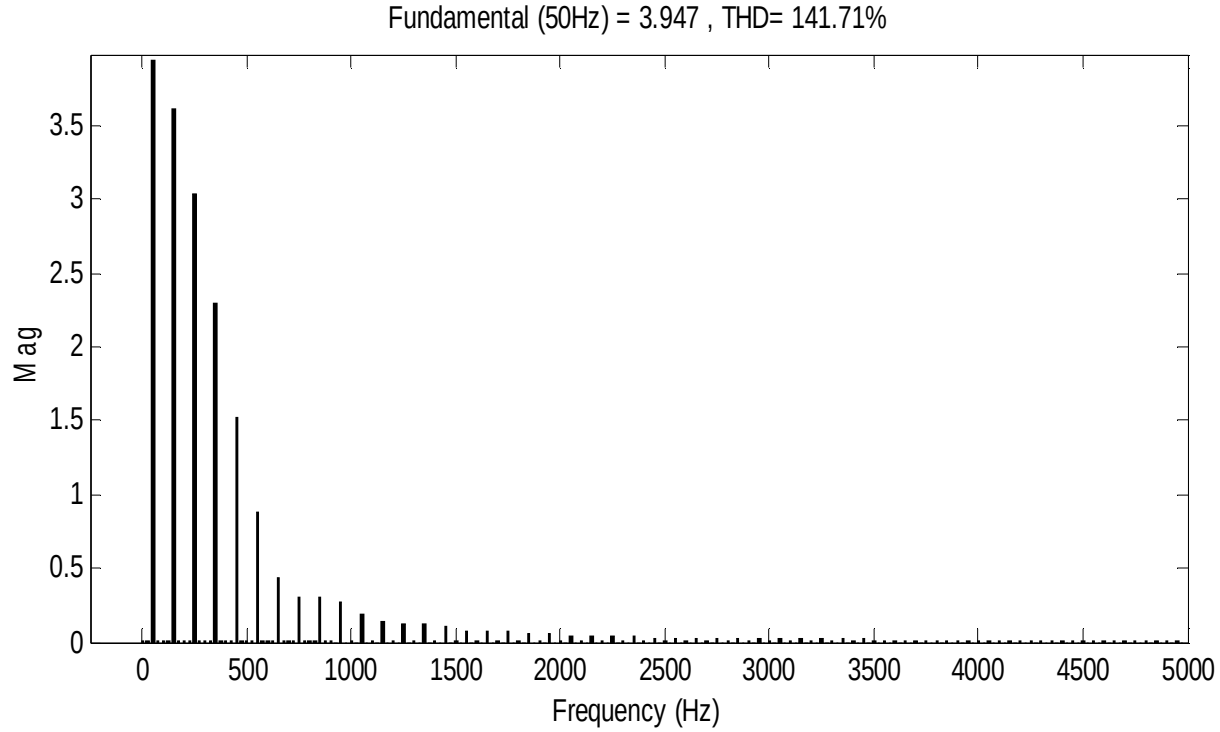


Fig 2.9 FFT component of input current for 200Ω load resistance and 410V (peak) input voltage.

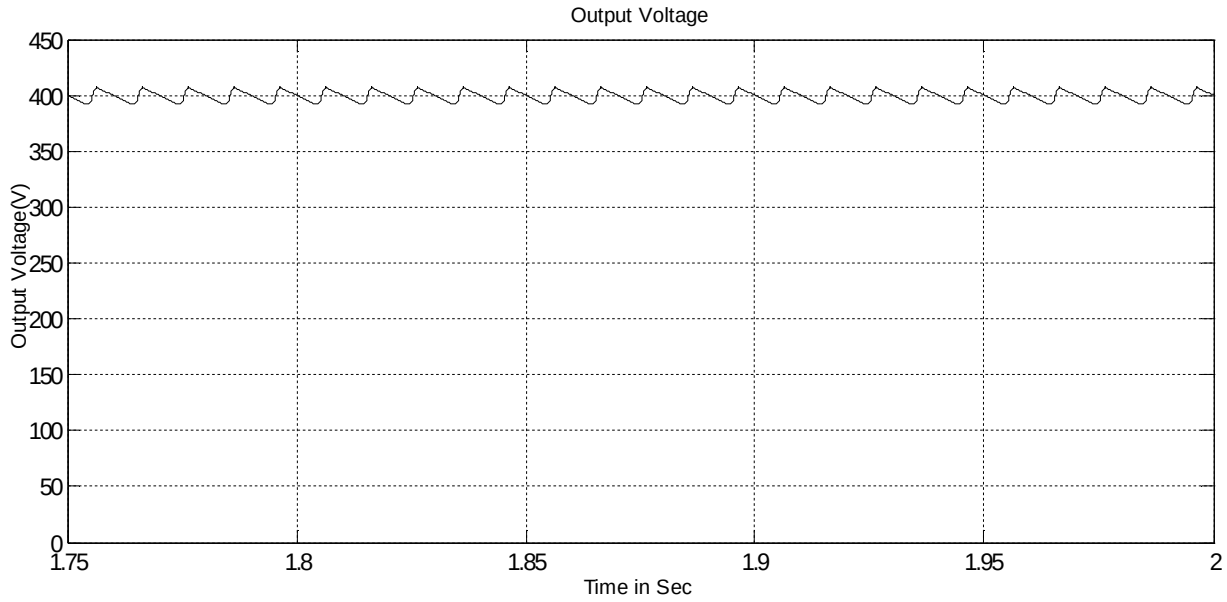


Fig 2.10 Output voltage for 200Ω load resistance and 410V (peak) input voltage.

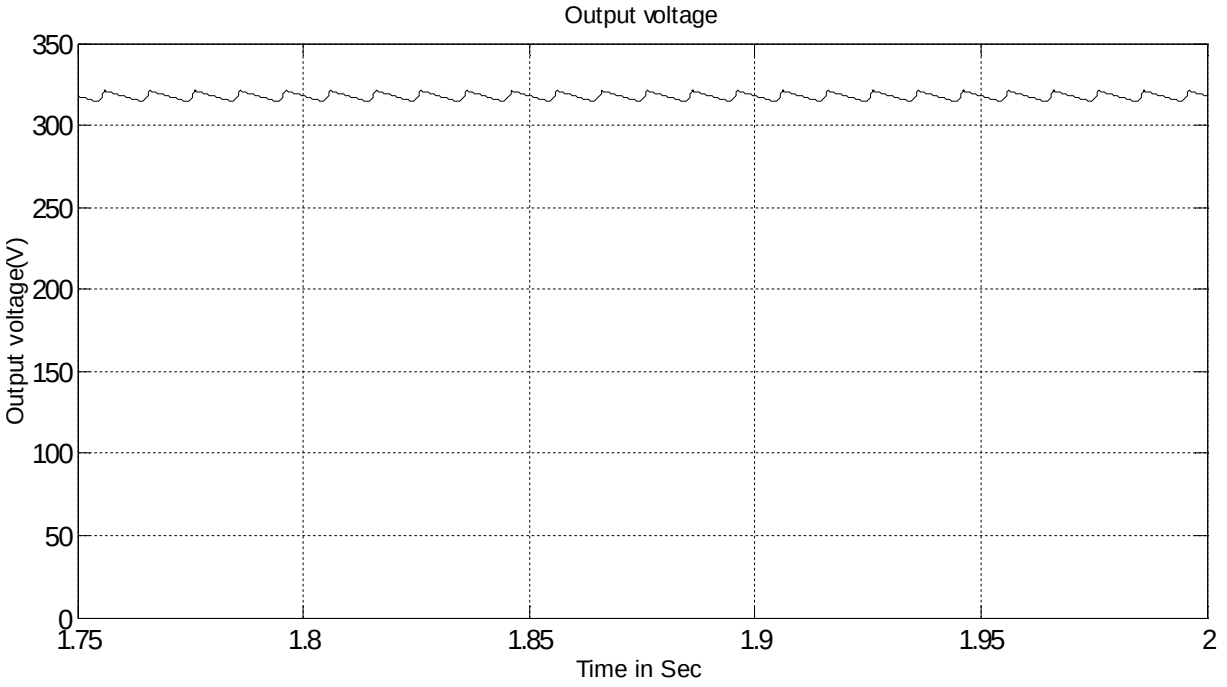


Fig 2.11 Output voltage for 400Ω load resistance and 325V (peak) input voltage

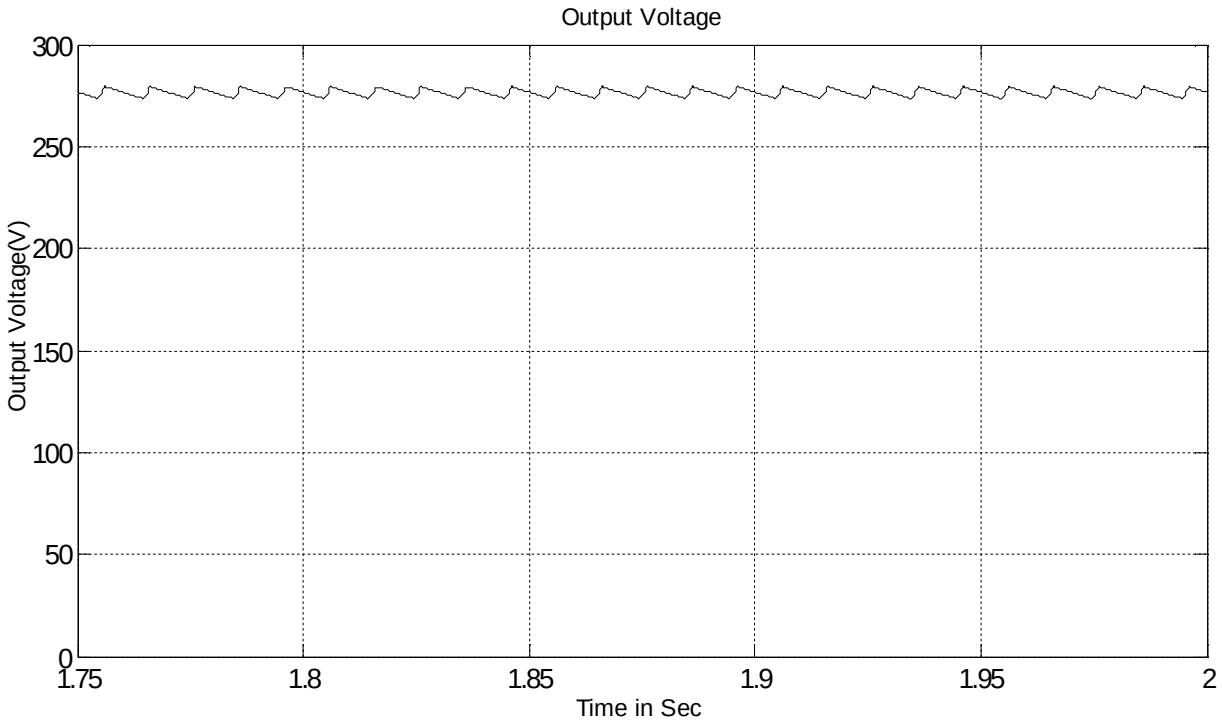


Fig 2.12 Output voltage for 400Ω load resistance and 282.8V (peak) input voltage

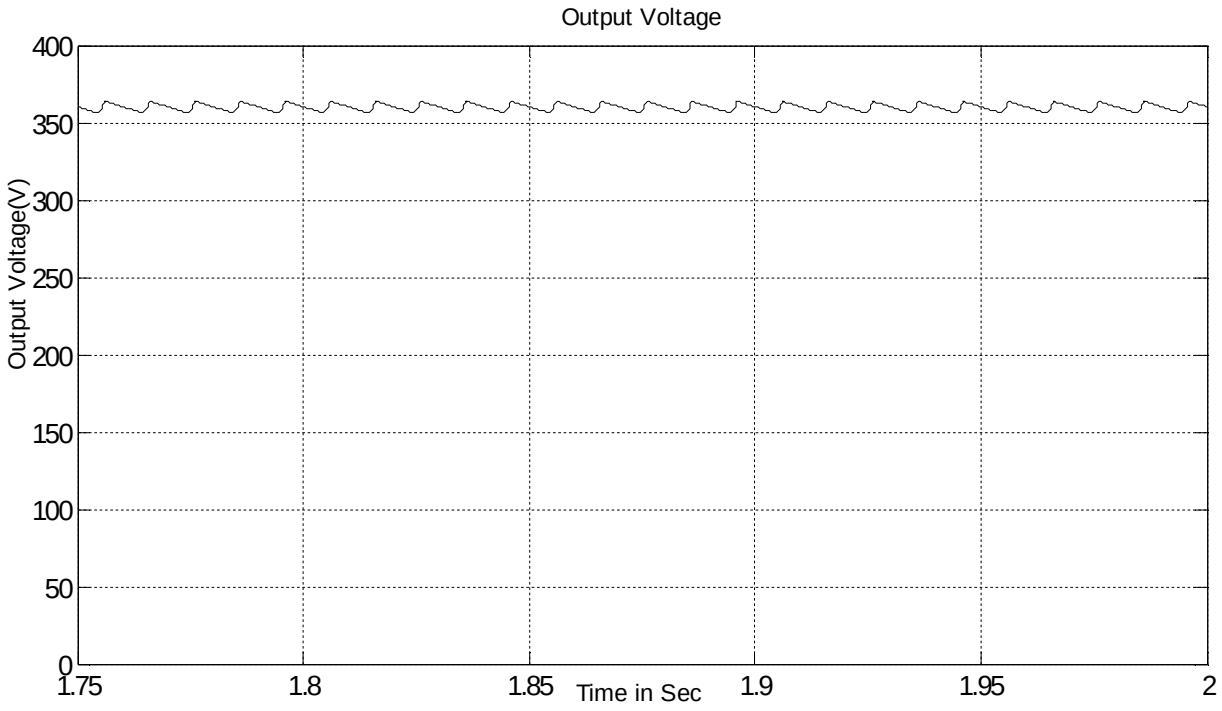


Fig 2.13 Output voltage for 400Ω load resistance and 368V (peak) input voltage

Table 2.1 summarizes the result from Fig. 2.2 to Fig. 2.10. Fig. 2.14 shows the output voltage at different input voltage. Fig. 2.15 to Fig. 2.18 show the input current THD, efficiency, output voltage and input power factor respectively for different load.

Table 2.1: Data for 400Ω to 200Ω load resistance at 410V (peak) input voltage, 50Hz supply frequency and 400V reference output voltage

Resistive load in Ohm	Input rms current(A)	Input rms Voltage(V)	Output DC voltage(V)	Load current (A)	Input THD in %	Input PF	Efficiency In %
400	2.63	290	401	1	158.45	0.53	52.7
267	3.75	290	400	1.5	148.5	0.56	55.4
200	4.84	290	399	2	141.71	0.58	56.7

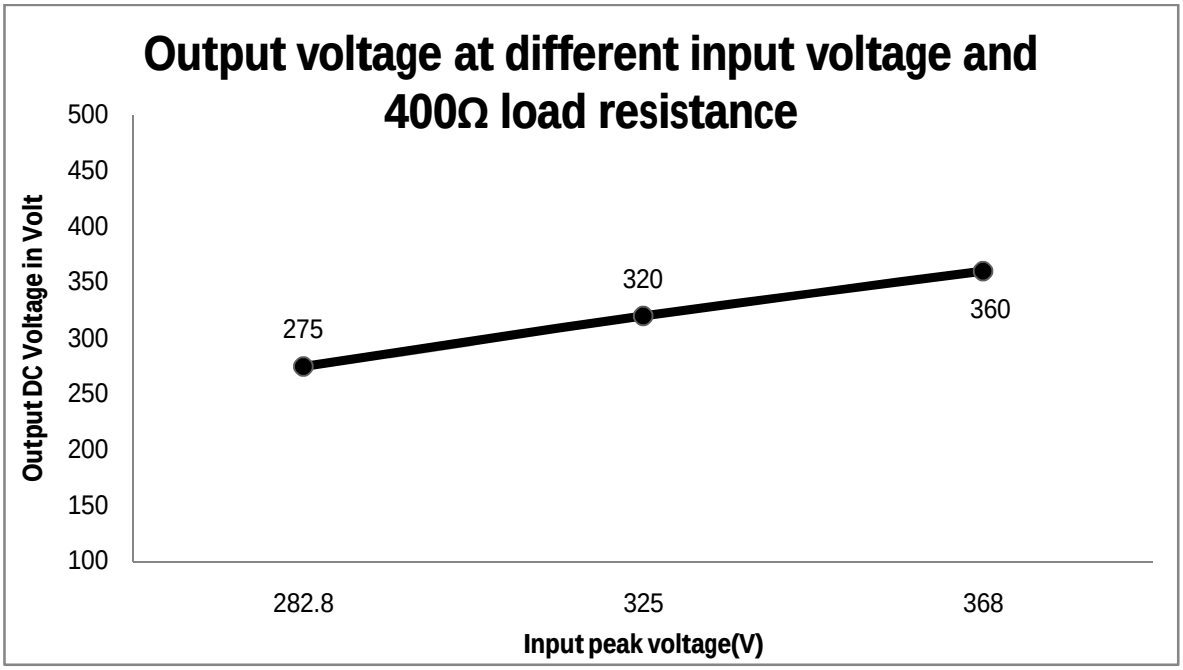


Fig 2.14 Output voltage at different input voltages with 400Ω load resistance.

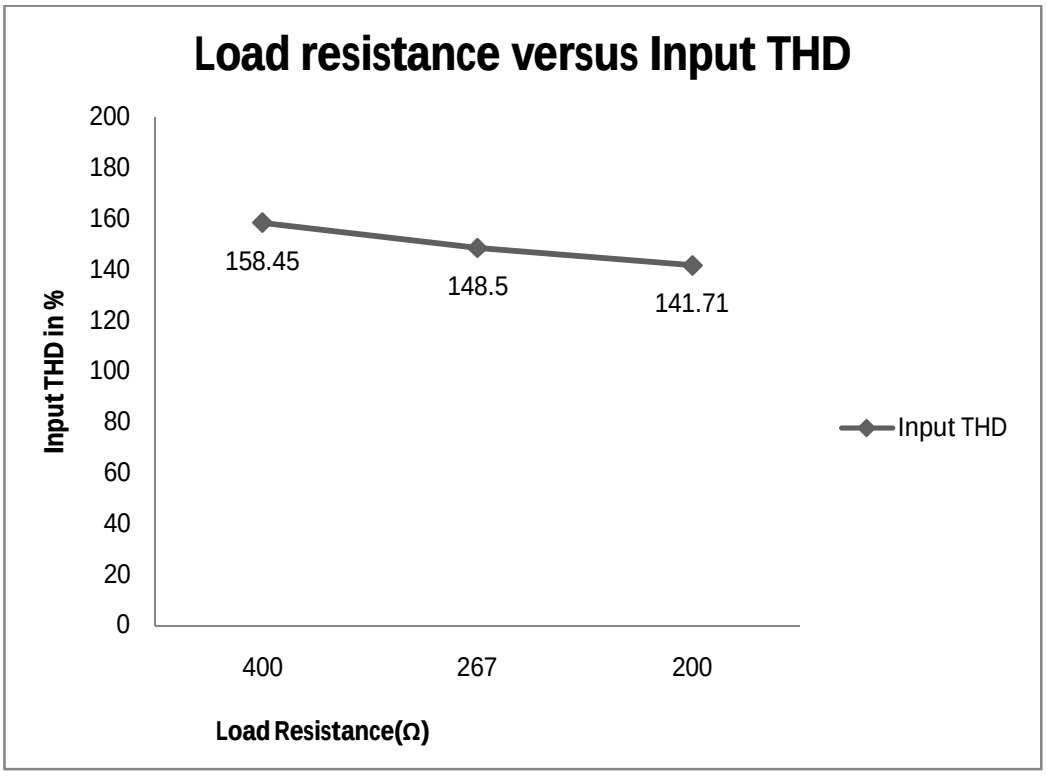


Fig 2.15 Input THD at different loads and 410V (peak) input voltage.

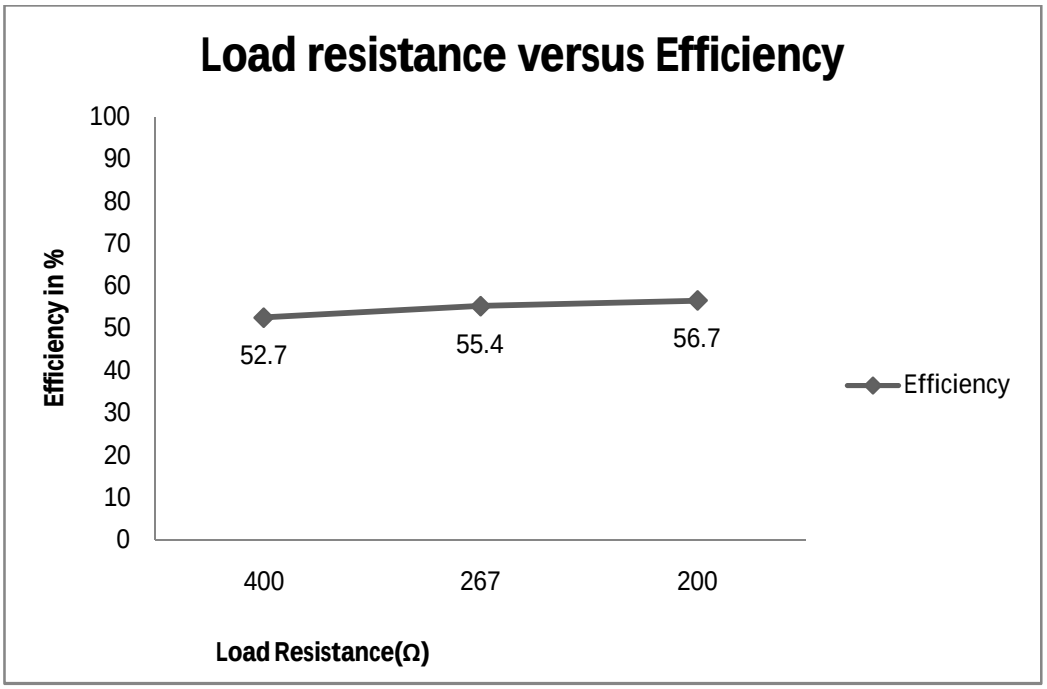


Fig 2.16 Efficiency at different loads and 410V (peak) input voltage.

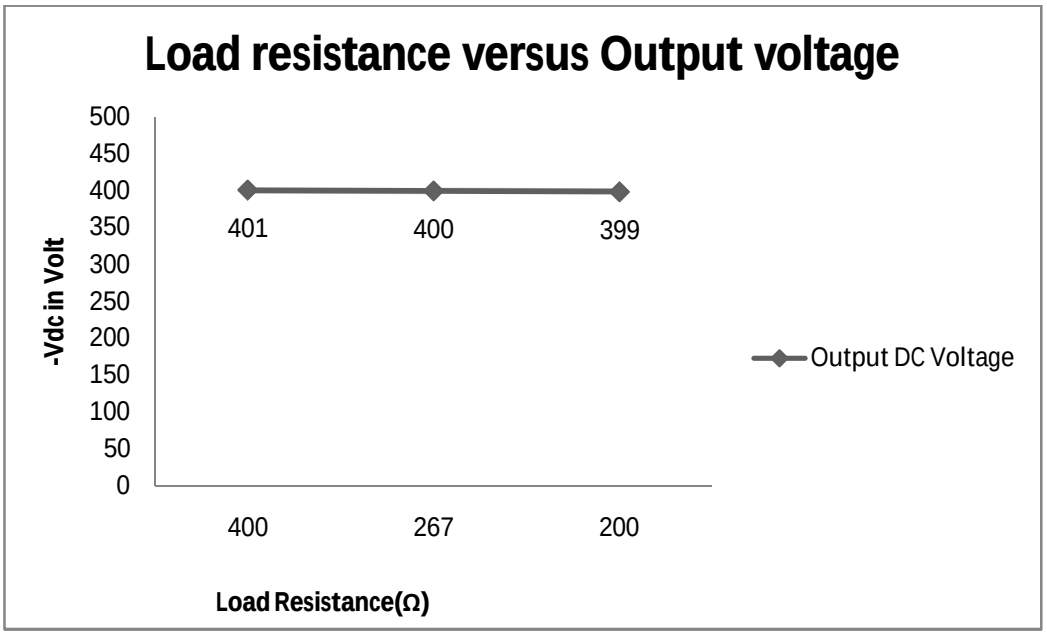


Fig 2.17 Output voltage at different loads and 410V (peak) input voltage.

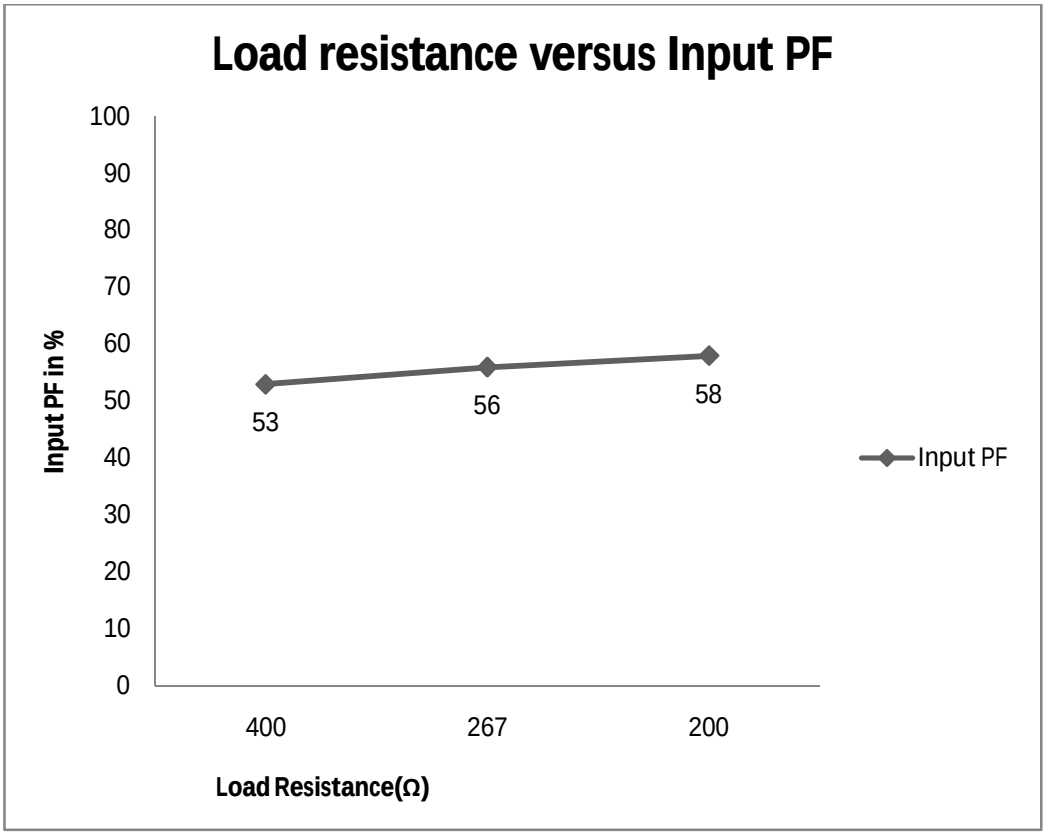


Fig 2.18 Input PF at different loads and 410V (peak) input voltage.

2.2 Open loop Ĉuk AC-DC converter

The circuit arrangement of the open loop Ĉuk regulator [10] is shown in Fig. 2.19. It provides an output voltage that is less than or greater than the input voltage and of opposite polarity from its input voltage. The circuit operation can be divided into two modes. Mode-1 is shown in Fig. 2.20 begins when transistor (IGBT) is turned on. The current through inductor L1 rises. At the same time the voltage of capacitor C1 reverse biases diode D and turns it off. The capacitor C1 discharges its energy to the circuit formed by C1, C2, the load and L2. Mode-2 is shown in Fig. 2.21 begins when transistor (IGBT) is turned off. The current through inductor L1 falls. The capacitor C1 is charged from the input supply and the energy stored in the inductor L2 is transferred to the load. The diode D and transistor (IGBT) provide a synchronous switching action. The capacitor C1 is the medium for transferring energy from the source to the load.

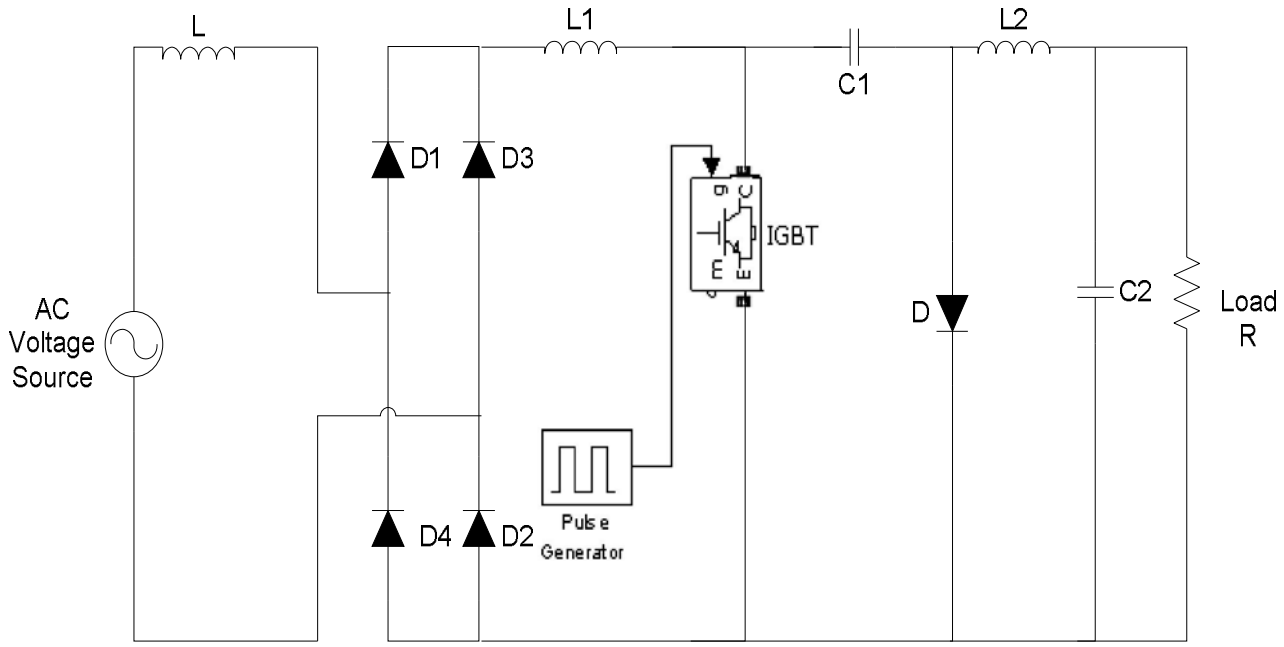


Fig 2 .19 Open loop Ĉuk regulator based full bridge single phase rectifier

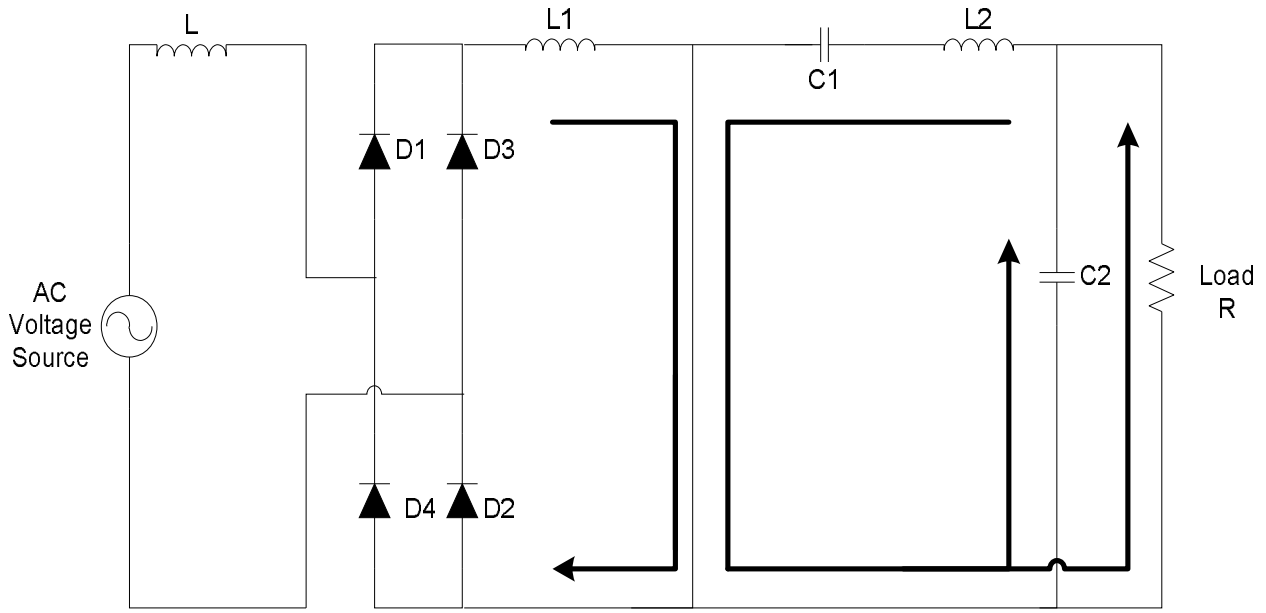


Fig 2 .20 Mode-1 when Switch is closed

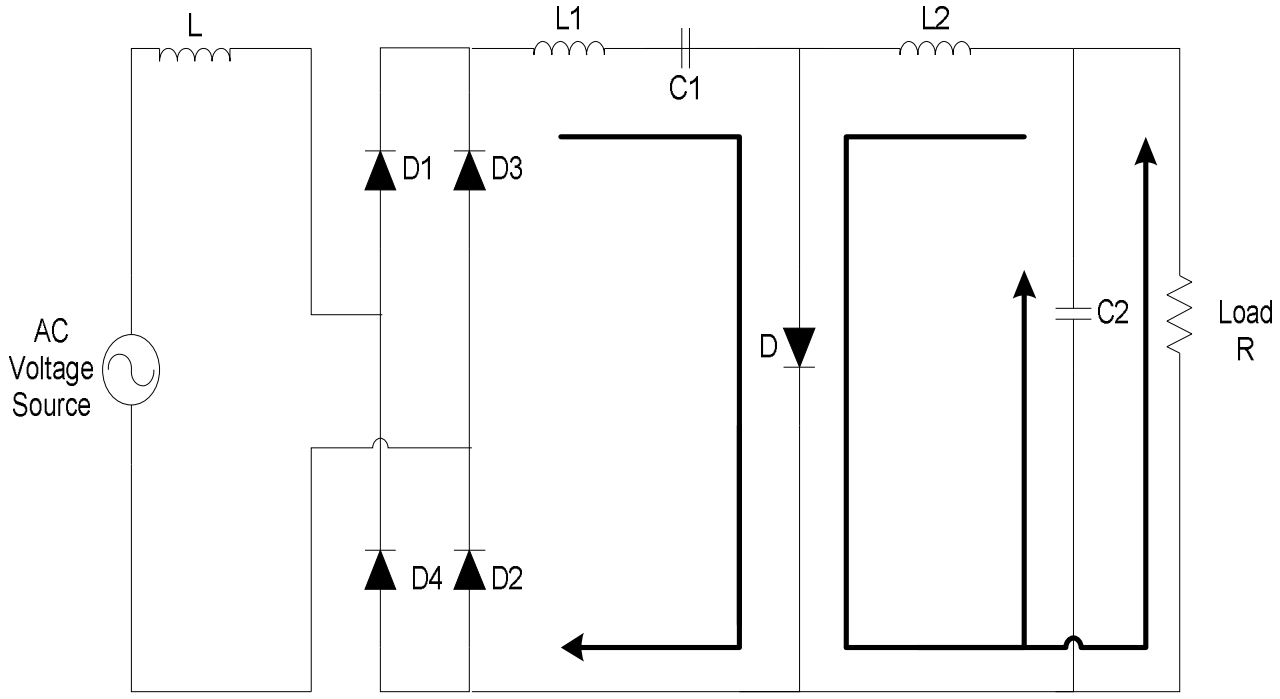


Fig 2 .21 Mode-2 when Switch is opened

The ideal input/output voltage and current relations are defined by the following equations [22]

$$V_{out} = -\frac{kV_{in}}{1-k} \quad \text{..... (2.1)}$$

$$I_{out} = \frac{(1-k)I_{in}}{k} \quad \text{..... (2.2)}$$

Where,

k is the duty cycle = $\frac{T_{on}}{T}$

T_{on} = on time of the switch

T_{off} = off time of the switch

$T = T_{on} + T_{off}$ = period of the switch

V_{in} = Average value of the input rectified AC voltage

V_{out} = Output voltage across the load

I_{in} = input current through the inductor and I_{out} = output current through the load

The condition for continuous inductor current and capacitor voltage is given by the following equation [22]

$$L1 = \frac{(1-k)^2 R}{2kf} \quad \text{..... (2.3)}$$

$$L2 = \frac{(1-k)R}{2f} \quad \text{..... (2.4)}$$

$$C1 = \frac{k}{2fR} \quad \text{..... (2.5)}$$

$$C2 = \frac{k}{8fR} \quad \text{..... (2.6)}$$

Where,

f = Switching frequency and R= Load resistance

Fig. 2.22 to Fig. 2.35 are obtained for f=80000Hz, k=0.65, L=100uH, L1=10mH, L2=1.8mH, C1=0.1uF, C2=2000uF. Fig. 2.22 to Fig. 2.24 show input voltage/current, FFT component and output voltage respectively for -400V reference output voltage at 400Ω load resistance and 410V

(peak) input voltage and 50Hz supply frequency using open loop Ĉuk rectifier. Similarly Fig. 2.25 to Fig. 2.27 for 800Ω load resistance, Fig. 2.28 to Fig. 2.30 for 267Ω load resistance and Fig. 2.31 to Fig. 2.33 for 200Ω load resistance. Fig. 2.34 to Fig. 2.35 show the output voltage for 400Ω load resistance and different input voltages.

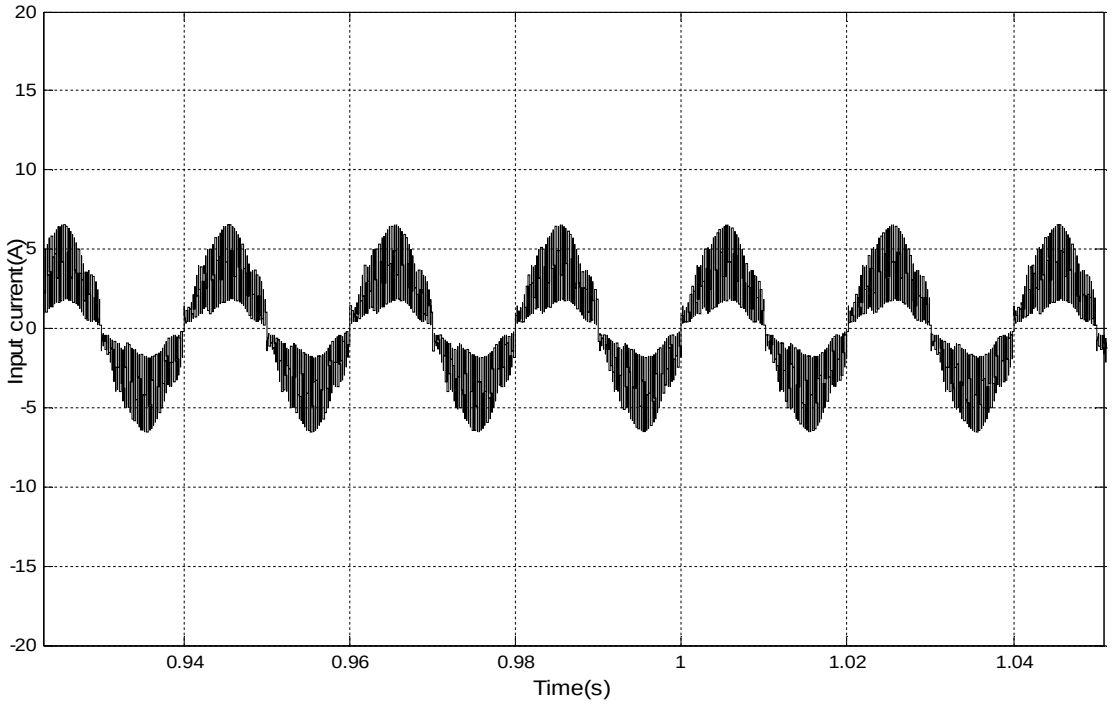


Fig 2.22 Input current for 400Ω load resistance and 325V (peak) input voltage

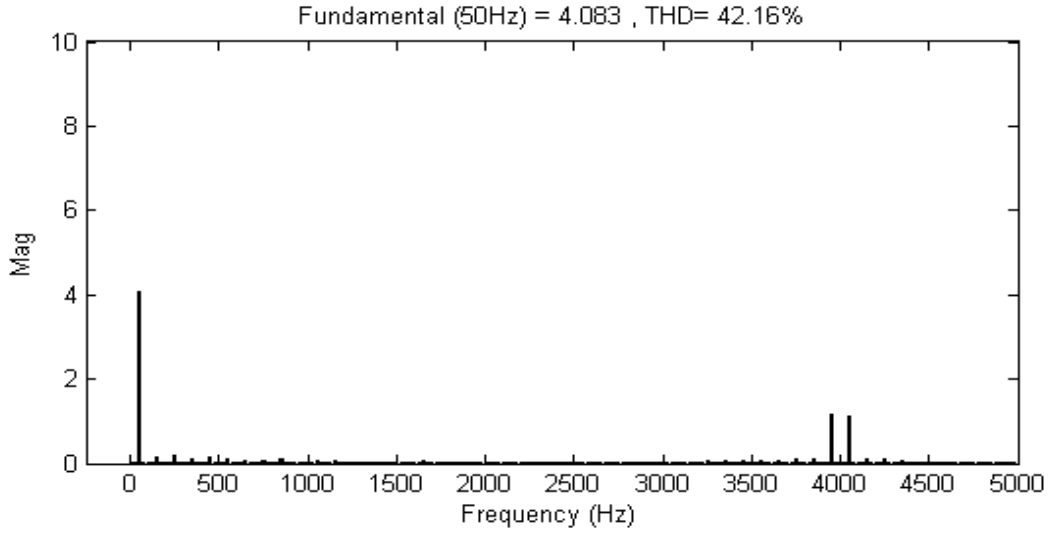


Fig 2.23 FFT component of input current for 400Ω load resistance and 325V(peak) input voltage

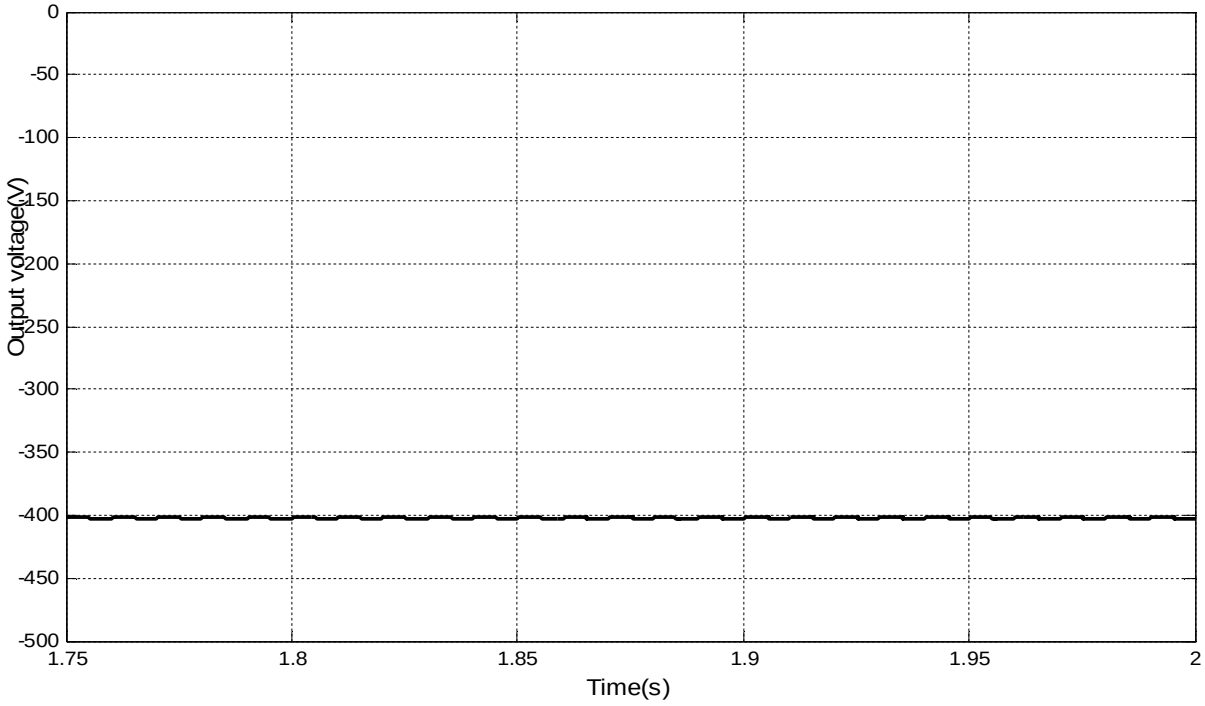


Fig 2.24 Output voltage for 400Ω load resistance and 325V (peak) input voltage

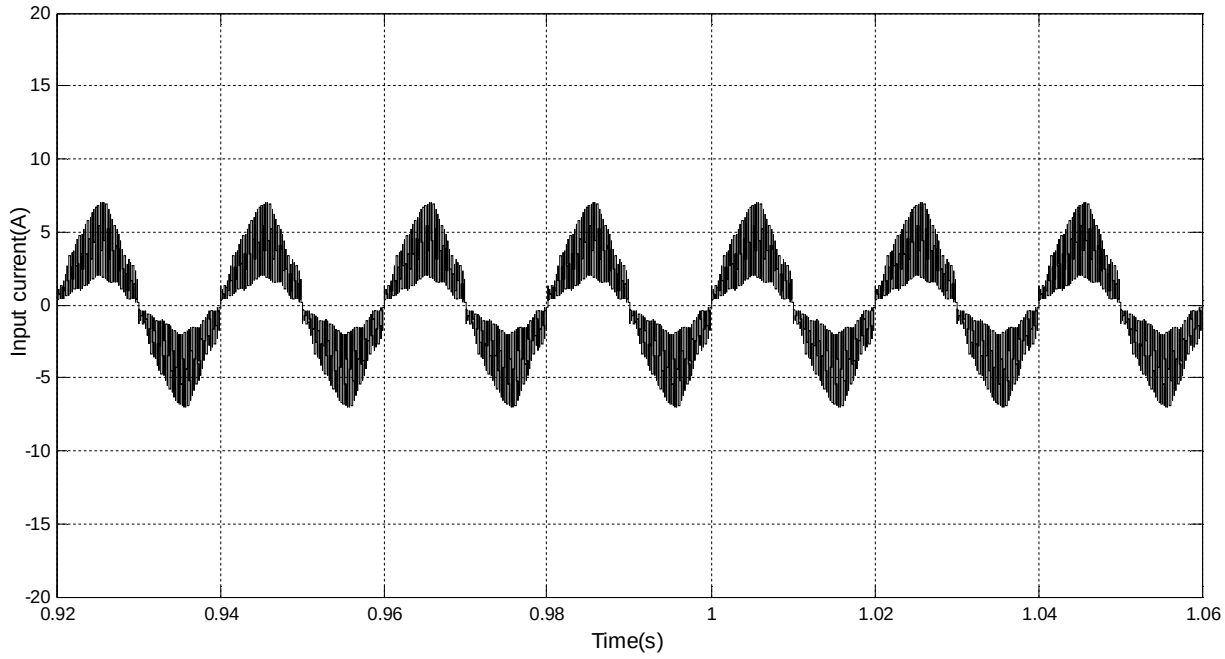


Fig 2.25 Input current for 267Ω load resistance and 325V (peak) input voltage

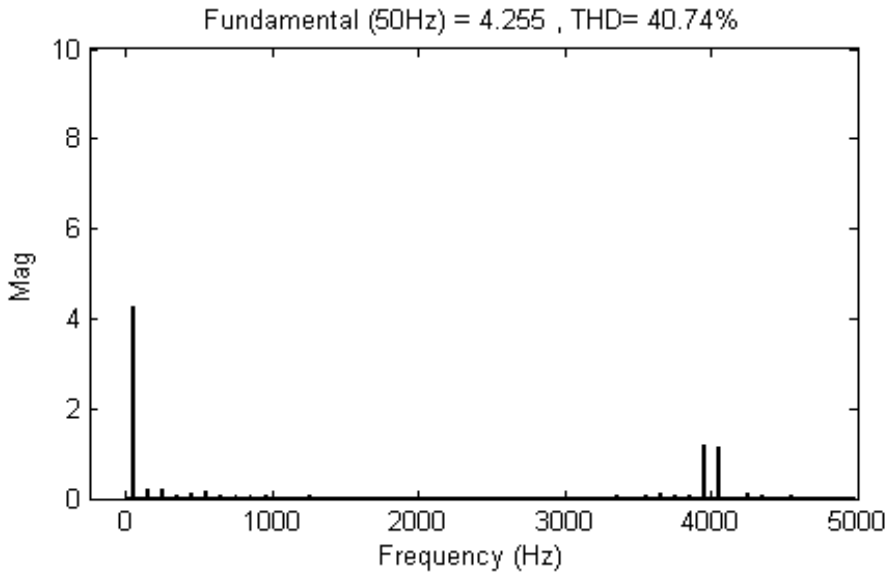


Fig 2.26 FFT component of input current for 267Ω load resistance and 325V(peak) input voltage

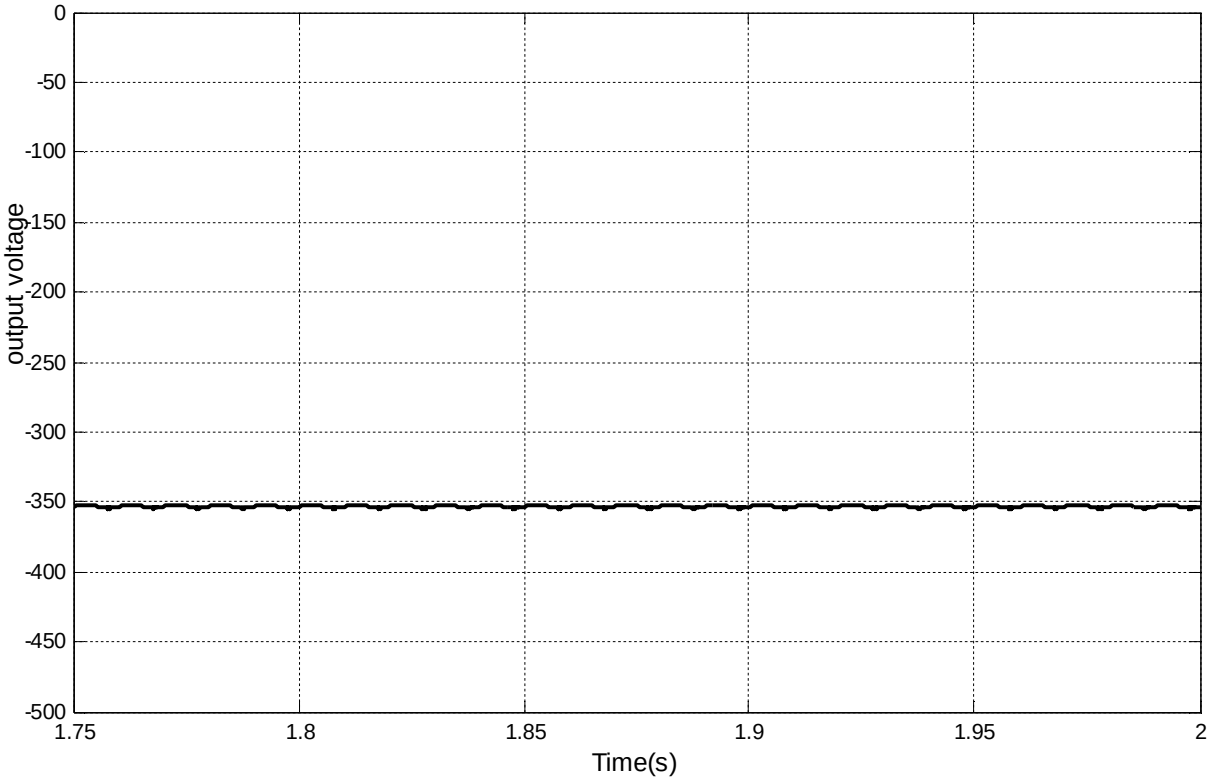


Fig 2.27 Output voltage for 267Ω load resistance and 325V (peak) input voltage

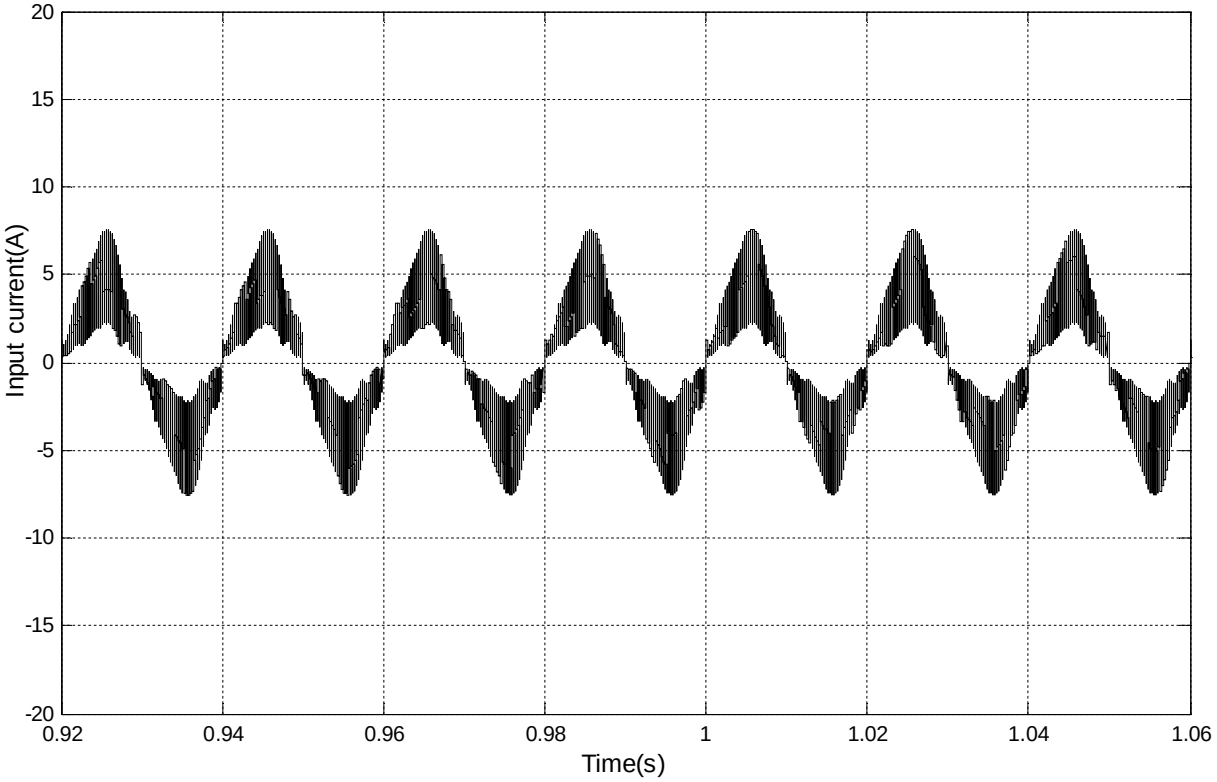


Fig 2.28 Input current for 200Ω load resistance and 325V (peak) input voltage

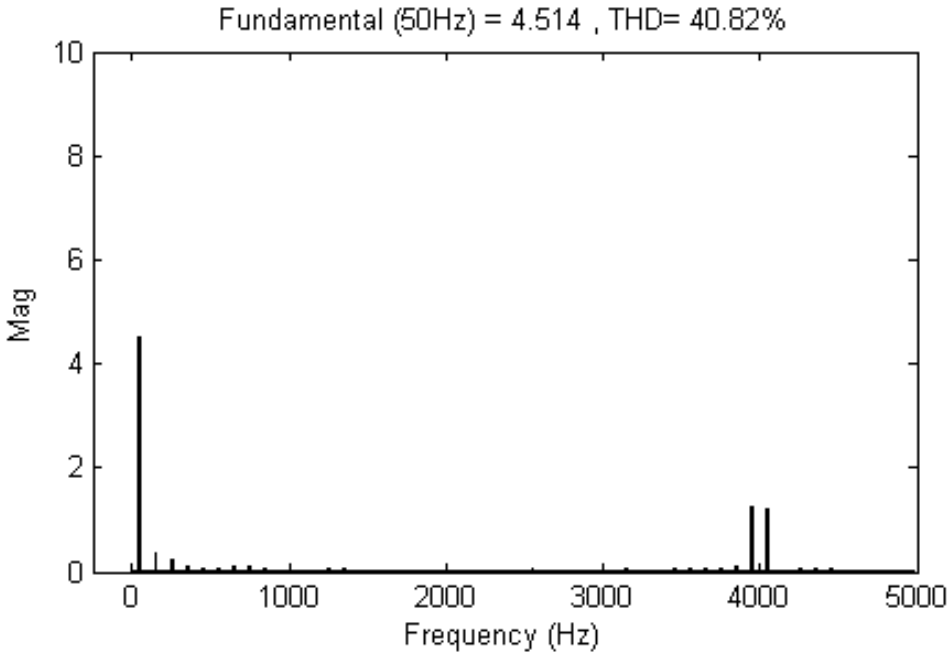


Fig 2.29 FFT component of input current for 200Ω load resistance and 325V(peak) input voltage

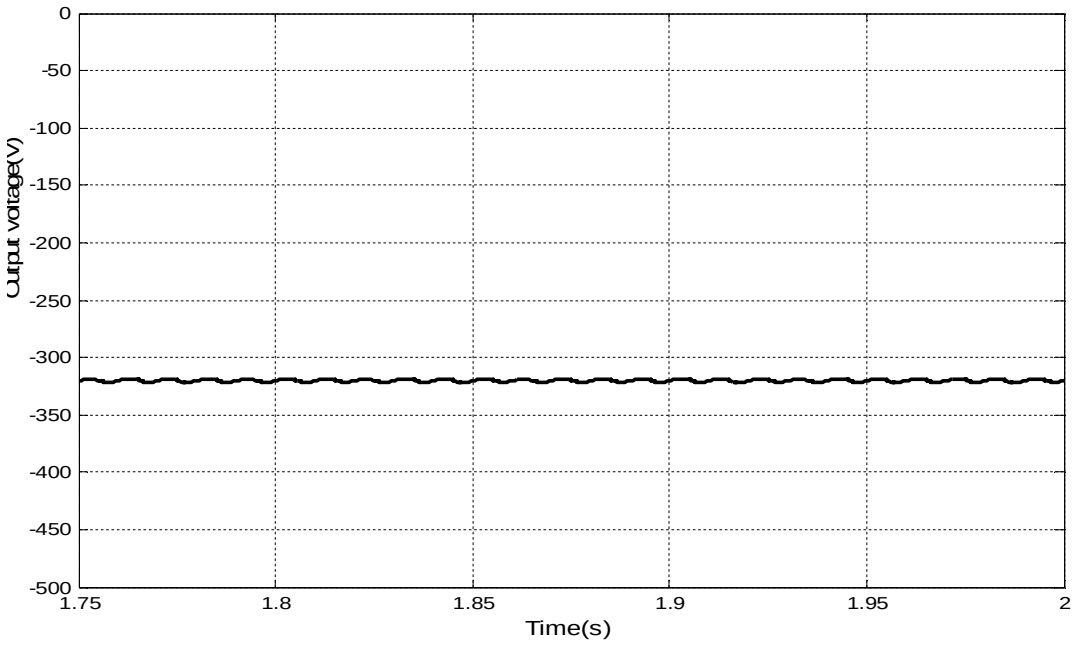


Fig 2.30 Output voltage for 200Ω load resistance and 325V (peak) input voltage

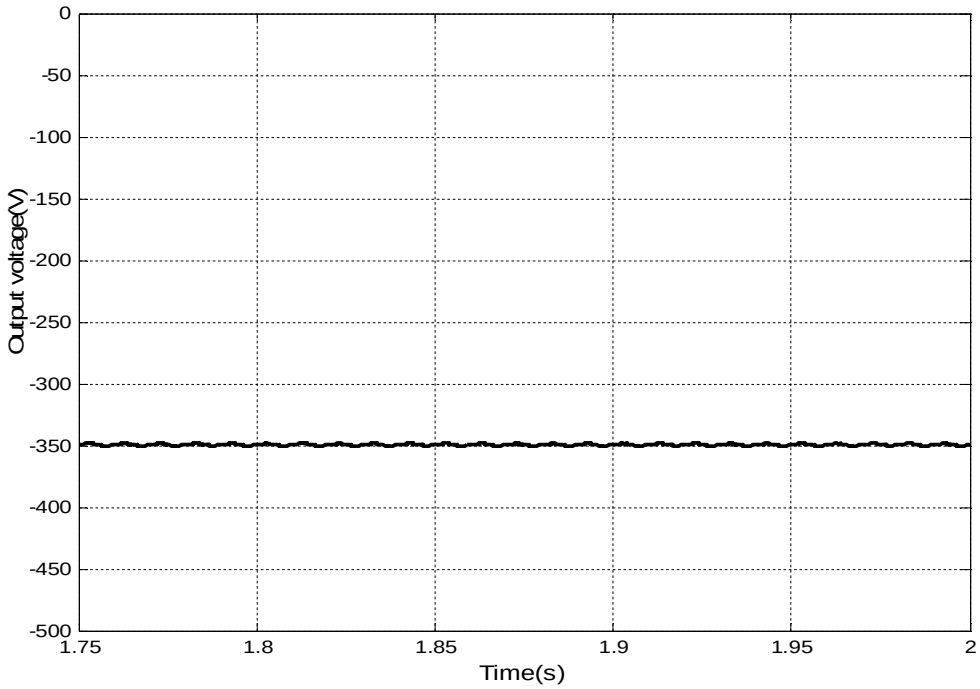


Fig 2.31 Output voltage for 400Ω load resistance and 282.8V (peak) input voltage

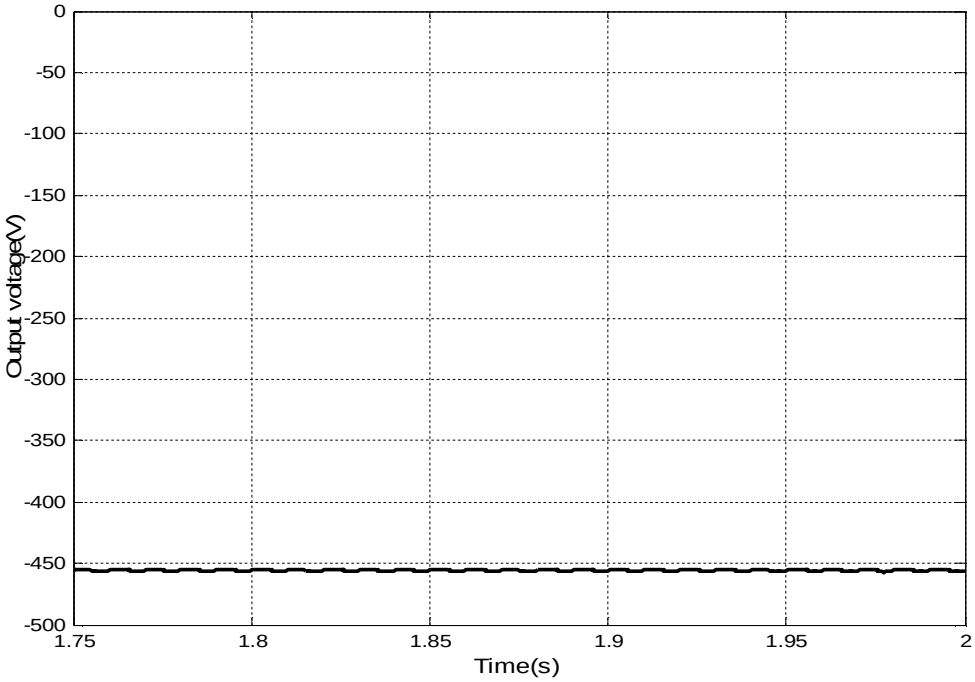


Fig 2.32 Output voltage for 400Ω load resistance and 367.69V (peak) input voltage

Table 2.2 summarizes the result from Fig. 2.22 to Fig. 2.30. Fig. 2.33 shows the output voltage at different input voltage. Fig. 2.34 to Fig. 2.37 show the output voltage, Input power factor, input current THD and efficiency respectively for different load.

Table 2.2: Data for 800Ω to 200Ω load resistance at 325V (peak) input voltage, 50Hz supply frequency and -400V reference output voltage

Resistive load in Ohm	Input rms current(A)	Input rms Voltage(V)	Output DC voltage(V)	Load current (A)	Input THD in %	Input PF	Efficiency In %
400	3.13	230	403	1.01	42.16	0.91	57
267	3.25	230	353.16	1.32	40.74	0.9247	63
200	3.45	230	320.17	1.60	40.82	0.9239	65

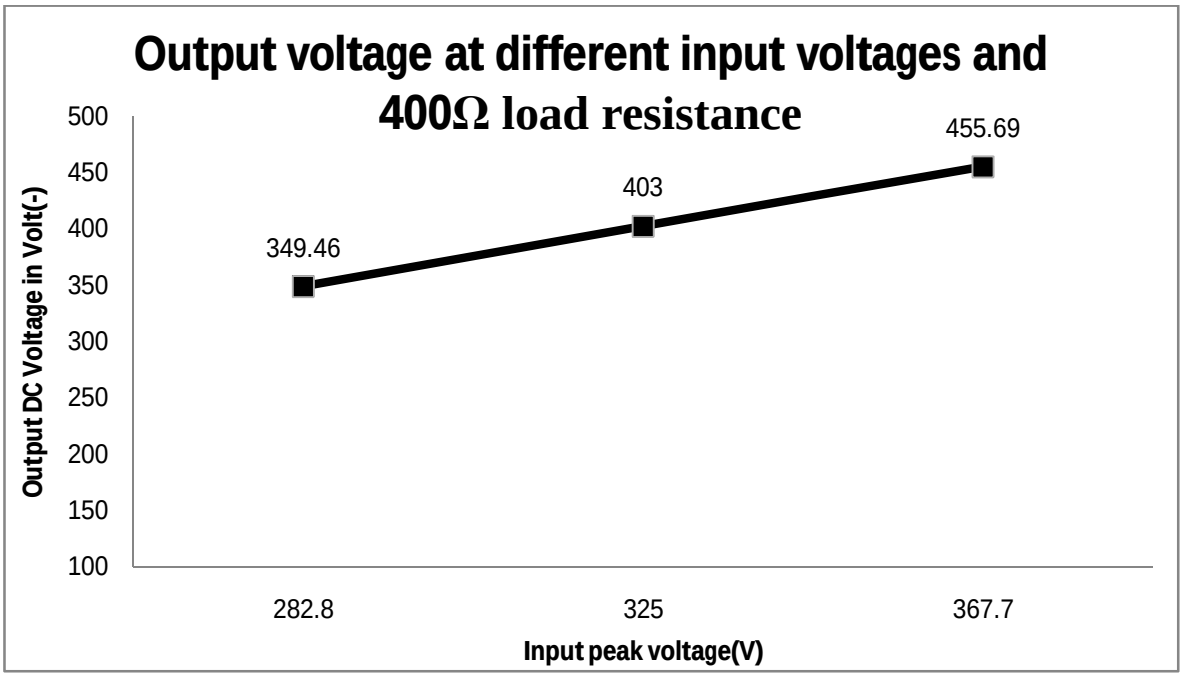


Fig 2.33 Output voltage at different input voltage with 400Ω load resistance.

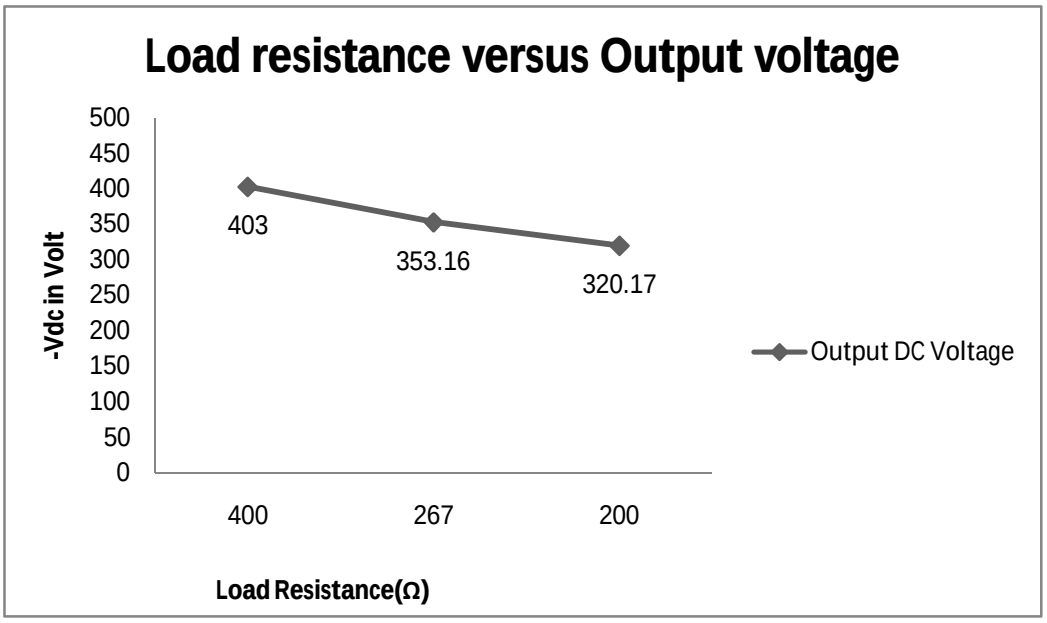


Fig 2.34 Output voltage at different load and 325V (peak) input voltage.

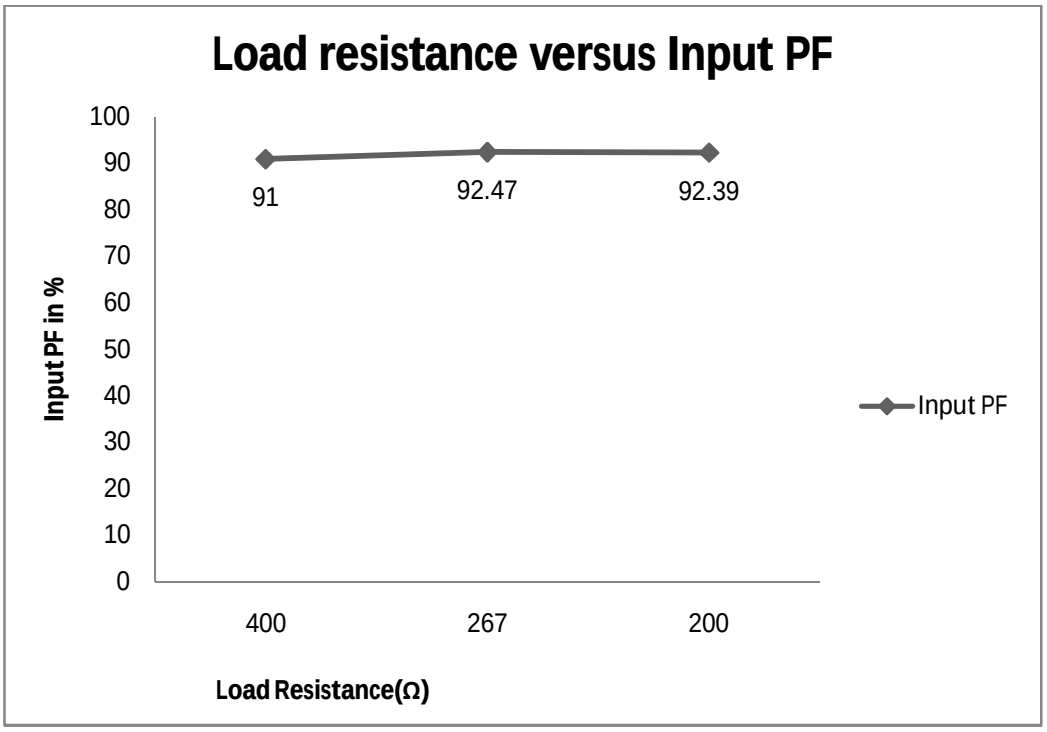


Fig 2.35 Input PF at different load and 325V (peak) input voltage.

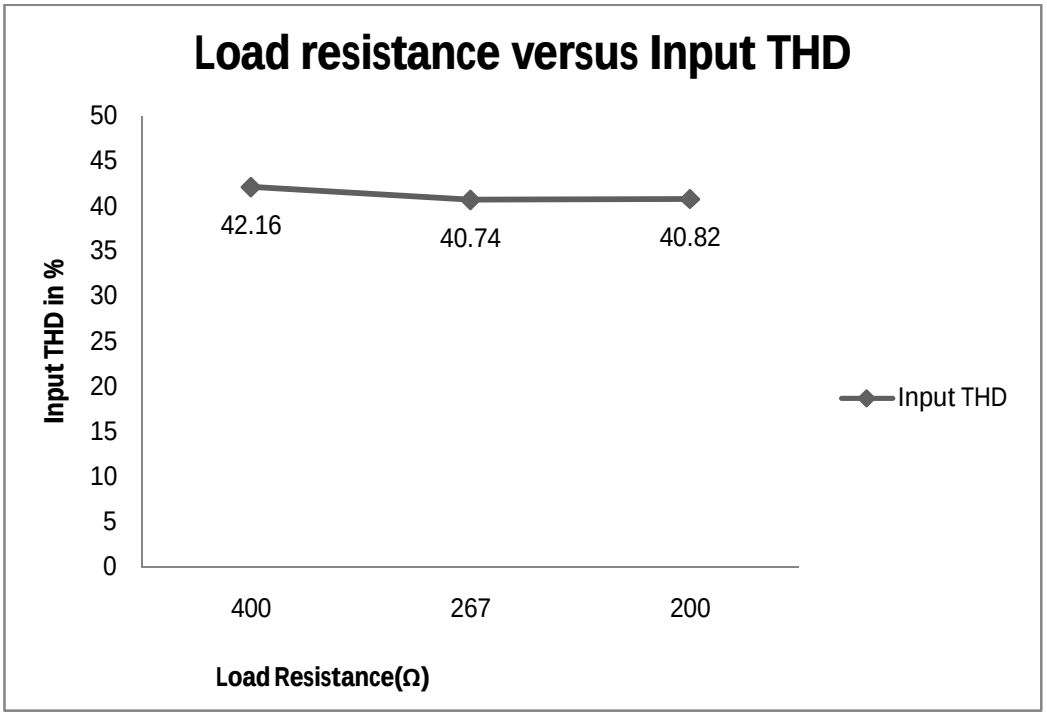


Fig 2.36 Input THD at different load and 325V (peak) input voltage.

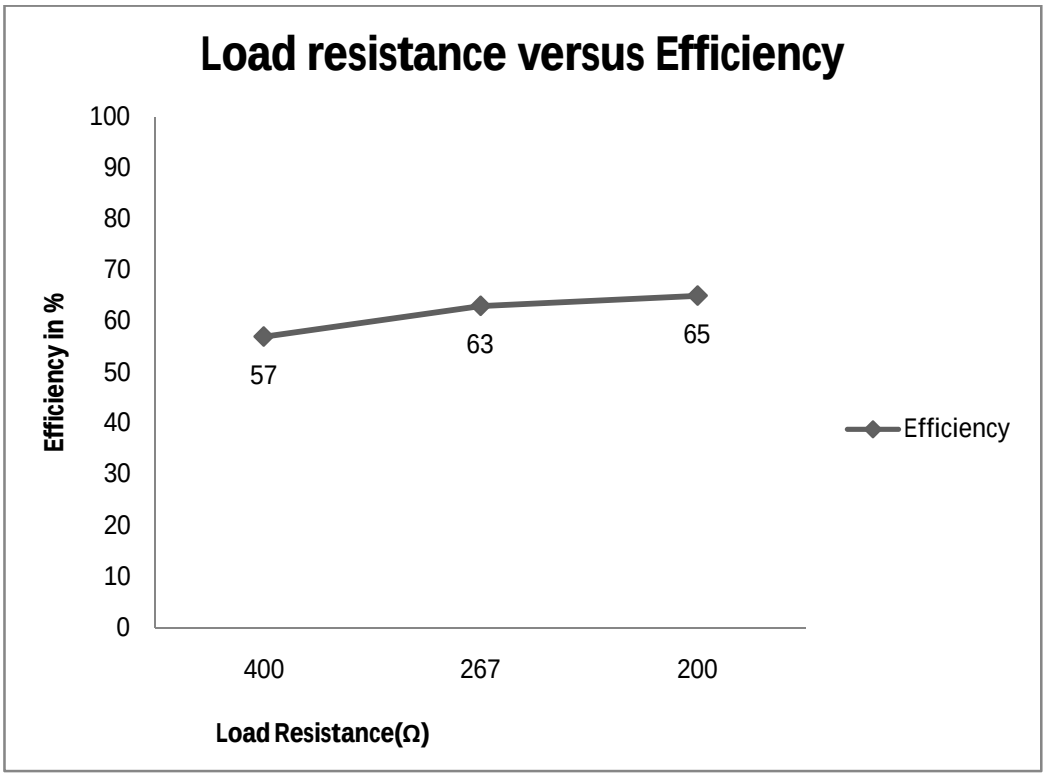


Fig 2.37 Efficiency at different load and 325V (peak) input voltage.

2.3 Proposed close loop Ĉuk AC-DC converter

2.3.1 Close loop Ĉuk AC-DC converter

From the previous chapter we have seen that both Normal diode rectifier and open loop Ĉuk rectifier have the problem of output voltage variation due to its input voltage changes, Low power factor and higher input THD. To minimize this problem and obtain better performance close loop Ĉuk regulator based AC-DC converter which is shown in Fig. 2.38 is proposed. The technique used here is the Average Current Mode control. In Average Current Mode control, the output voltage is controlled by varying the average value of the current amplitude signal. The output voltage is controlled by the output voltage error compensator and input current is controlled by the input current error compensator. The Voltage feed forward compensator controls the input voltage variation in such way that if the input voltage reduces then the output of Voltage feed forward compensator increases and vice versa. The actual output DC voltage is sensed and compared with a reference voltage then the voltage error is processed through the proportional integral controller. The output of the proportional integral controller is multiplied with the rectified input voltage and output of voltage feed forward compensator to make a reference current in phase with rectified input voltage. The real current is forced to track the reference current through current error compensator. The error between the actual current and reference current is processed through the proportional integral controller and then its output is compared with the Saw-tooth wave to generate the required PWM signal. Matlab simulink presentation of the proposed system is shown in Fig. 2.39. Fig. 2.40 to Fig. 2.44 show the customized matlab simulink block presentation of the proposed system.

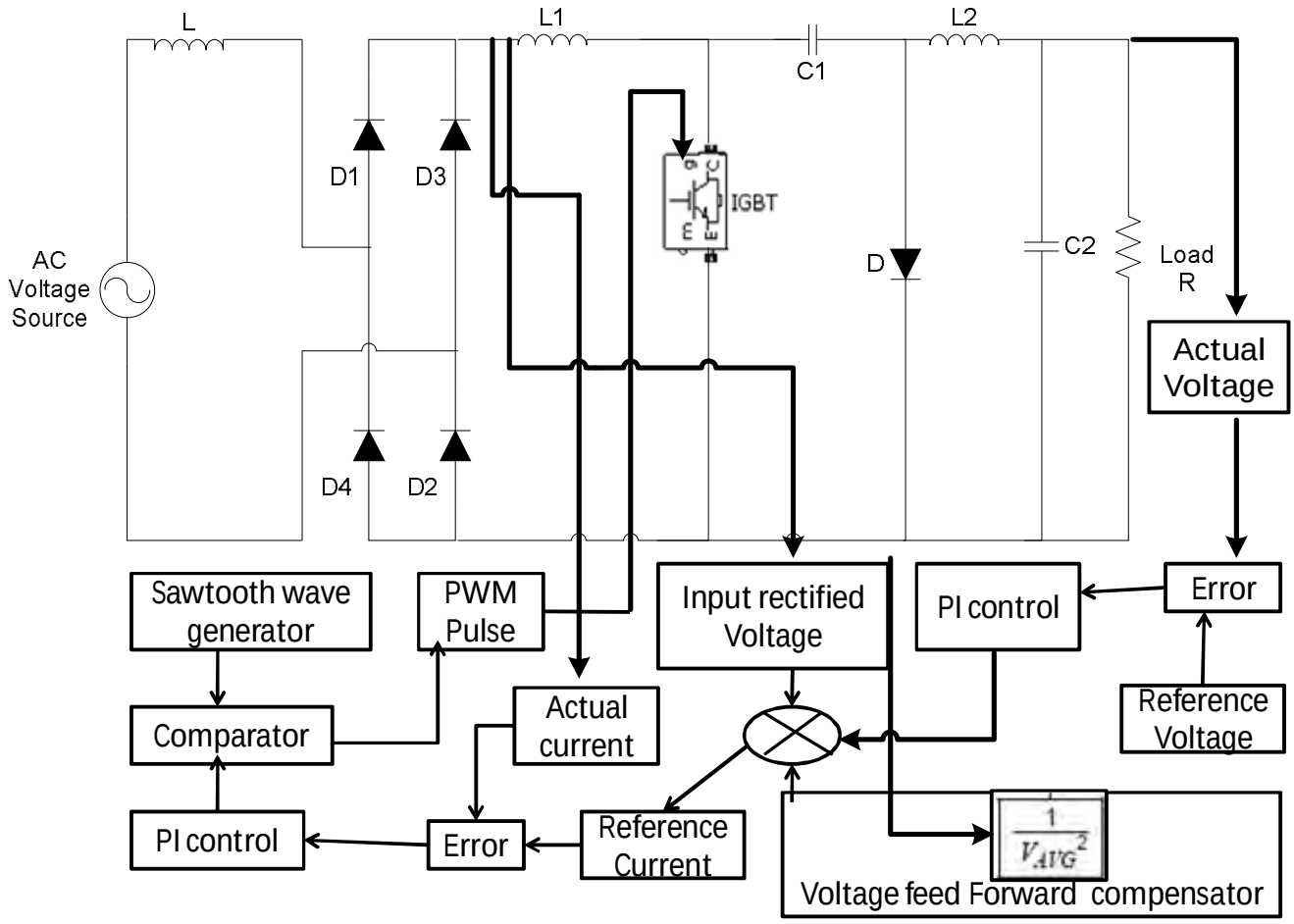


Fig 2.38 Proposed close loop ĈUK regulator based AC-DC converter.

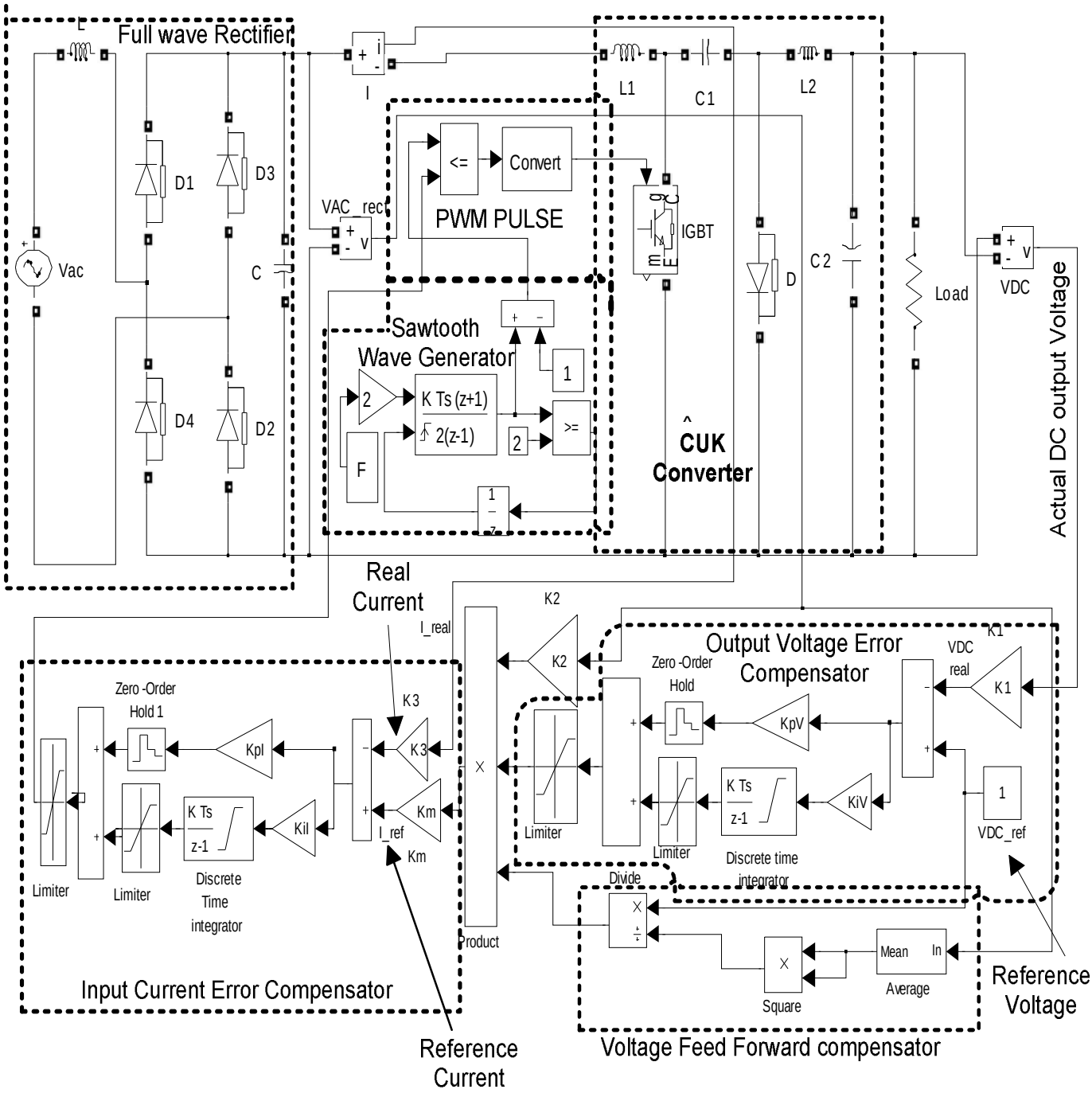


Fig 2.39 Proposed close loop ĈUK AC-DC converter in Matlab simulink.

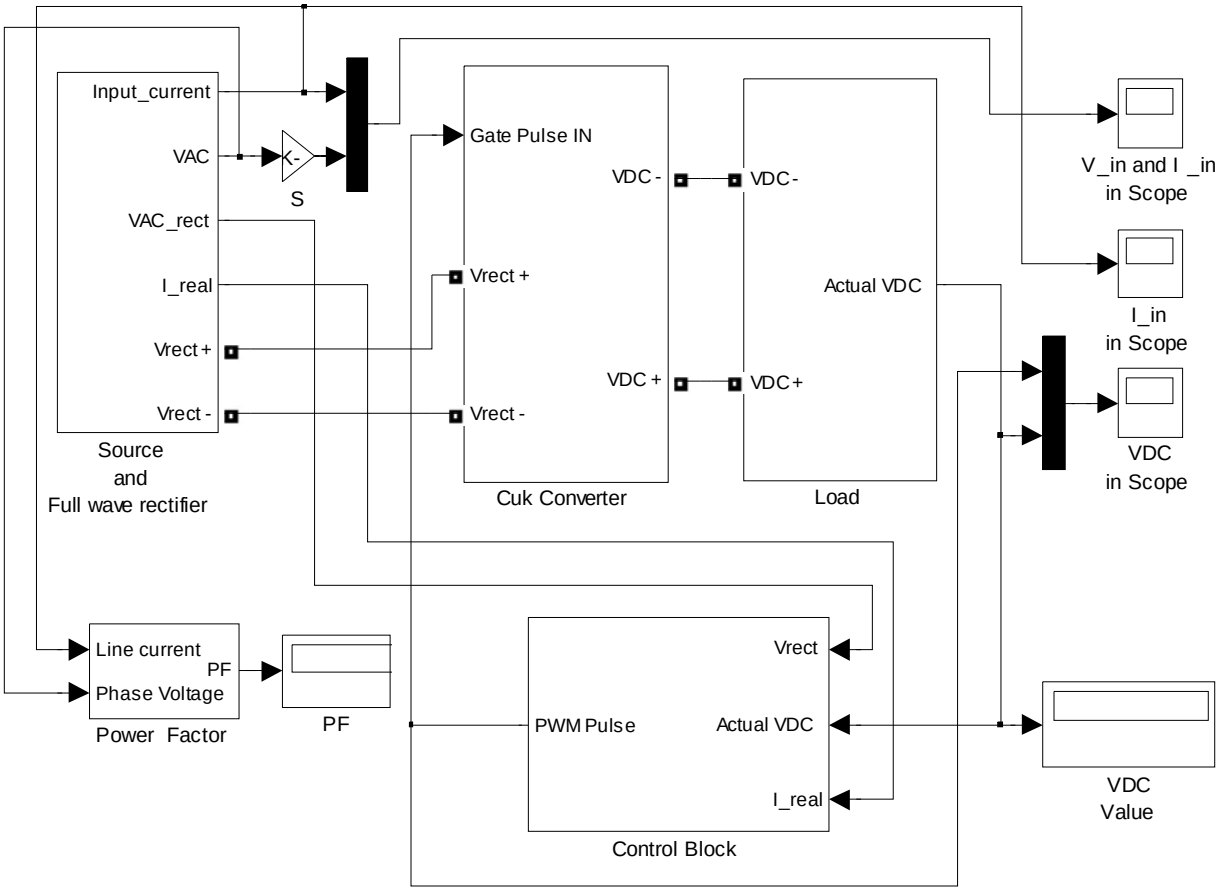


Fig 2.40 Customized representation of proposed close loop ĈUK AC-DC converter in Matlab simulink

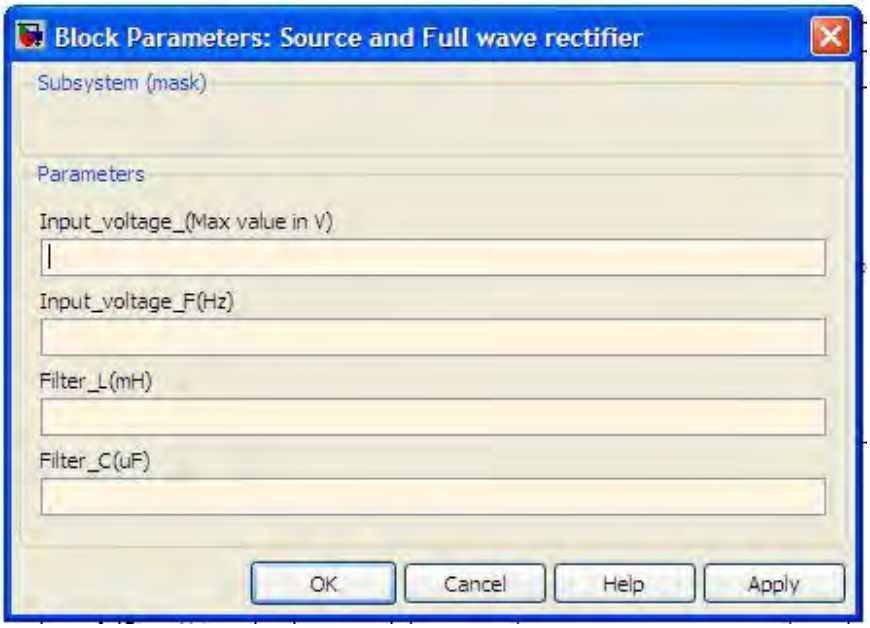


Fig 2.41 Input parameter of source and Full wave rectifier

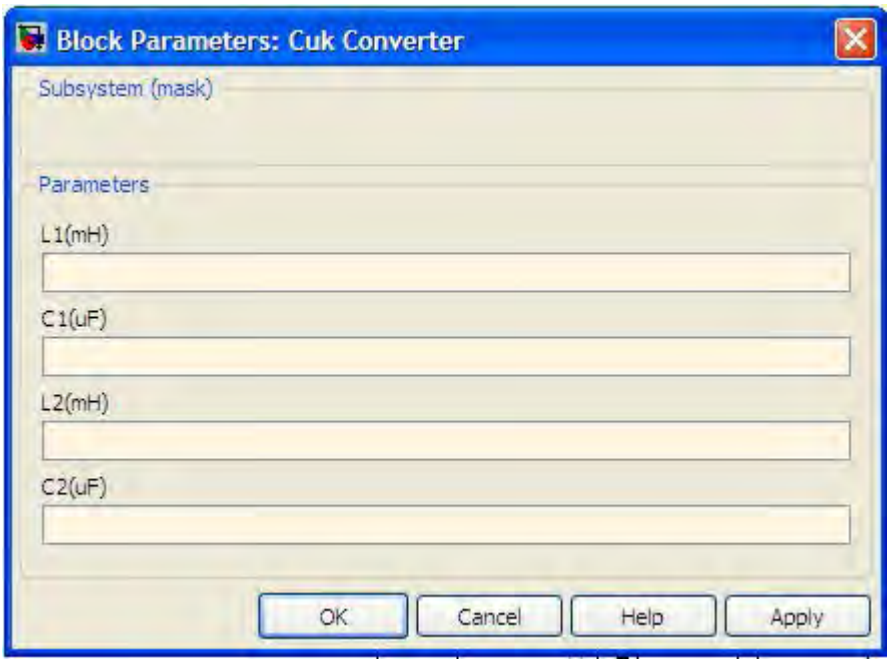


Fig 2.42 Input parameter of Ĉuk Converter

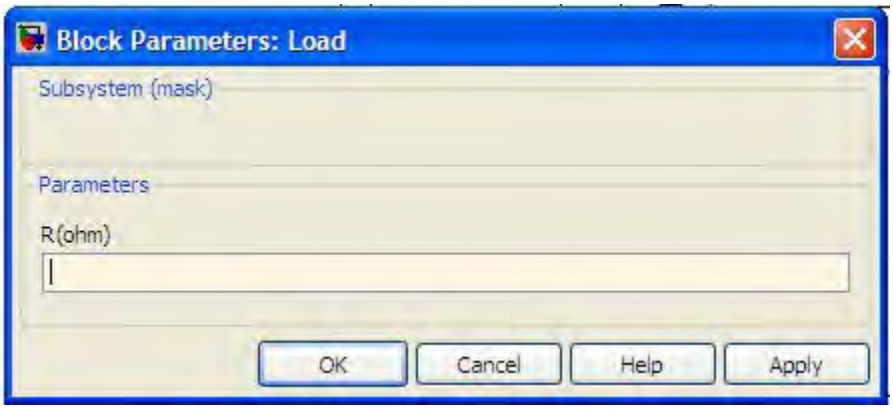


Fig 2.43 Input parameter of load block

Fig 2.44(a) Input parameter of control block

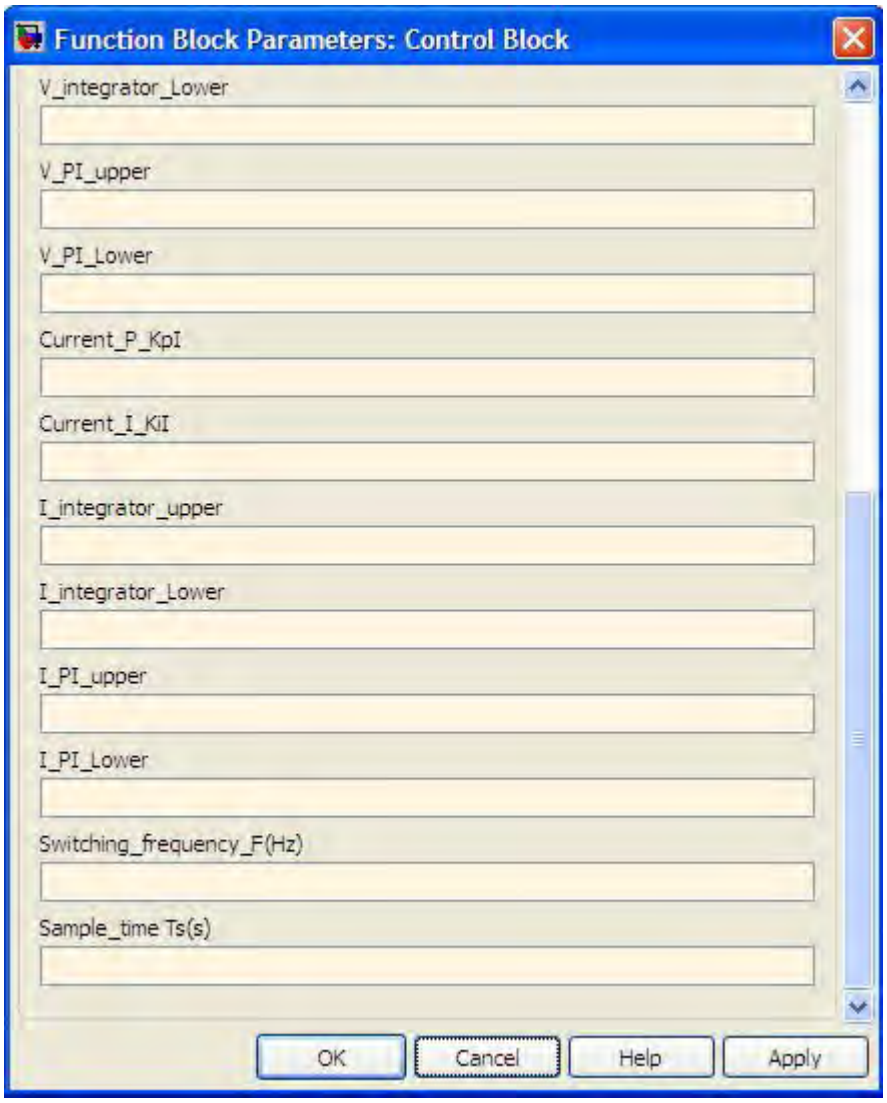


Fig 2.44(b) Input parameter of control block

Since Matlab has no block to determine power factor directly. Fig. 2.45 shows the way to determine the power factor.

We know the Power factor = Displacement Factor × Distortion Factor

Where Displacement Factor = $\cos \emptyset$

And Distortion Factor = $\sqrt{\frac{1}{1+(\frac{I_2}{I_1})^2+(\frac{I_3}{I_1})^2+\cdots+(\frac{I_n}{I_1})^2}} = \frac{I_1}{I} = \frac{1}{\sqrt{1+THD^2}}$

Here $\emptyset$ is the angle between the fundamental component of input current and input voltage

I_2, I_3 ... ... I_n are the harmonics component of input current

I is the rms value of input current and I_1 is the rms value of fundamental component of input current

Efficiency is calculated in the following way

$$\text{Efficiency} = \frac{Output\ power}{input\ apparent\ power} = \frac{\frac{output\ voltage^2}{load\ resistance}}{input\ rms\ voltage \times input\ rms\ current}$$

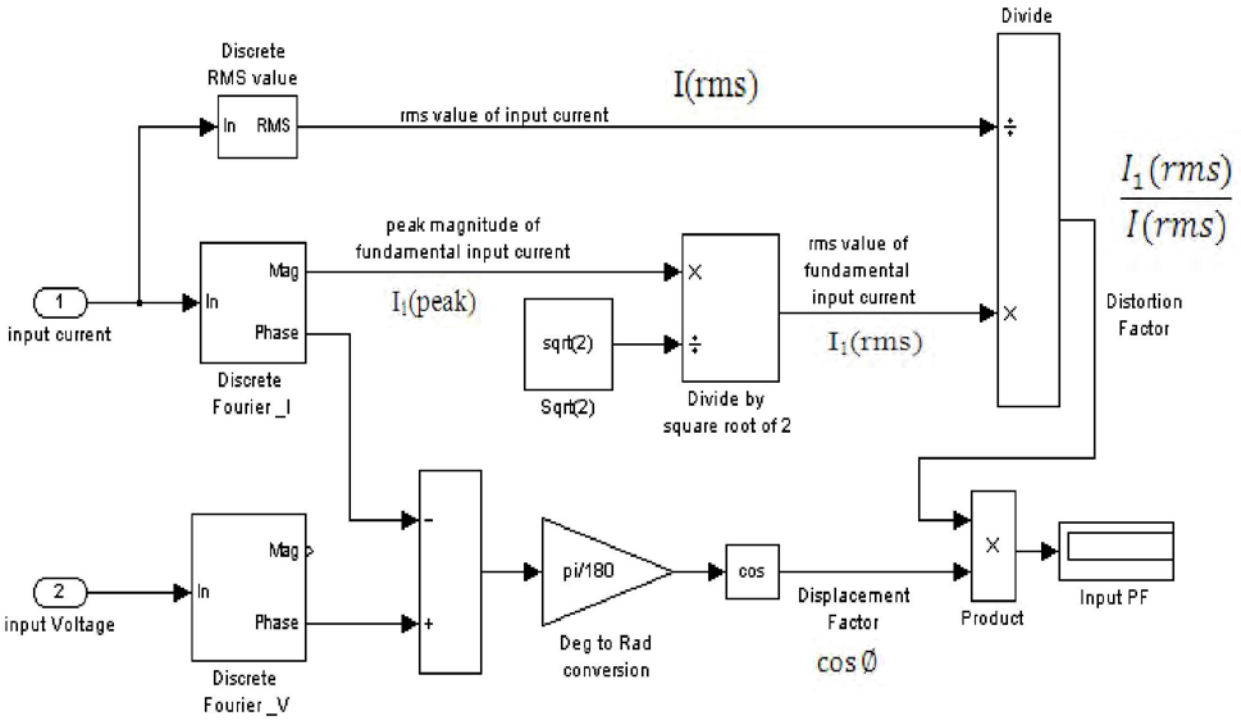


Fig 2.45 Determination of power factor in matlab simulink

2.3.2 Close loop response analysis of AC –DC Ĉuk converter using State space Averaging Technique

For analysis the stability of the proposed control system using bode plot open loop transfer function is needed. For continuous conduction mode (CCM), dc-dc converters operate in two circuit states in one switching period, (i) when switch is on for a time interval dT and (ii) when switch is off for a time interval (1-d) T, where d is a duty cycle and T is switching period. According to Leverrier.s Algorithm modeling of dc-dc converters using SSA method needs three steps in general as discussed below.

a. During each circuit state, the linear circuit is described by the state variable vector x.

Generally inductor currents and Capacitor voltages are chosen as state variables. The state space equations of two circuit states, in standard form, are obtained as

When switch is on during time interval dT, state-space equations of converter can be written as

$\frac{dx(t)}{dt} = A_1x(t) + B_1u(t)$ and

$y(t) = C_1x(t) + E_1u(t)$ .....(2.7)

The matrices A_1, B_1, C_1 and E_1 describe the network connections during the time interval dT

x(t), u(t) and y(t) are the state variable, input variable and output variable respectively.

When switch is off during time interval (1-d)T, state-space equations of converter can be written as

$\frac{dx(t)}{dt} = A_2x(t) + B_2u(t)$ and

$y(t) = C_2x(t) + E_2u(t)$.....(2.8)

The matrices A_2, B_2, C_2 and E_2 describe the network connections during the time interval 1(-d)T

b. Equation (2.7) and (2.8) are time weighted and averaged over one switching period as

$\frac{dx(t)}{dt} = A_{12}x(t) + B_{12}u(t)$ and

$y(t) = C_{12}x(t) + E_{12}u(t)$……(2.9)

Where

$A_{12} = A_1 d + A_2(1-d)$

$B_{12} = B_1 d + B_2(1-d)$

$C_{12} = C_1 d + C_2(1-d)$

$E_{12} = E_1 d + E_2(1-d)$

c. Now Linearization by introducing small ac perturbation around a DC operating point. To obtain a small signal ac model around a quiescent operating point, the following small perturbation as shown in equation (2.10) is added to state space model represented by equation (2.9),

x(t) = X+ $\tilde{x}(t)$

y(t) = Y+ $\tilde{y}(t)$

u(t) = U+ $\tilde{u}(t)$

d(t) = D+ $\tilde{d}(t)$……(2.10)

The capital letter represents a DC value. Small signal linearization is justified under the following condition

X > $\tilde{x}(t)$

Y > $\tilde{y}(t)$

U > $\tilde{u}(t)$

D > $\tilde{d}(t)$

Now replacing equation (2.10) in equation (2.9) we can get small signal state space model [41] as

$$\frac{d\tilde{x}(t)}{dt} = A\tilde{x}(t) + B\tilde{u}(t) + B_d\tilde{d}(t)$$

$$\tilde{y}(t) = C\tilde{x}(t) + E\tilde{u}(t) + E_d\tilde{d}(t) \quad \text{.....(2.11)}$$

Where

$$A = A_1D + A_2(1-D)$$

$$B = B_1D + B_2(1-D)$$

$$C = C_1D + C_2(1-D)$$

$$E = E_1D + E_2(1-D)$$

$$B_d = [(A_1 - A_2)X + (B_1 - B_2)U]$$

$$E_d = [(C_1 - C_2)X + (E_1 - E_2)U]$$

Taking Laplace transform of equation (11) we have

$$\tilde{X}(s) = (sI - A)^{-1}[B\tilde{u}(s) + B_d\tilde{d}(s)] \quad \text{.....(2.12)}$$

$$\tilde{Y}(s) = C(sI - A)^{-1}[B\tilde{u}(s) + B_d\tilde{d}(s)] + E\tilde{u}(s) + E_d\tilde{d}(s) \quad \text{.....(2.13)}$$

Using equation (2.14) and (2.15) for DC value of input voltage $V_{IN,}$ output Voltage V_O and duty Cycle D, the control-to-output and the input-to-output small signal transfer functions of the converter are respectively given as

$$\frac{\widetilde{V_O}(s)}{\tilde{d}(s)} = C(sI - A)^{-1}B_d + E_d \quad \text{.....(2.14)}$$

$$\frac{\widetilde{V_O}(s)}{\widetilde{V_{IN}}(s)} = C(sI - A)^{-1}B + \text{E} \quad \text{.....(2.15)}$$

Now for Ĉuk converter in Fig. 2.48 the parameter $A_{1,}$ $B_{1,}$ C_1 and E_1 is derived by the following way

When switch is on for dT time then from Fig 2.46(a) we have

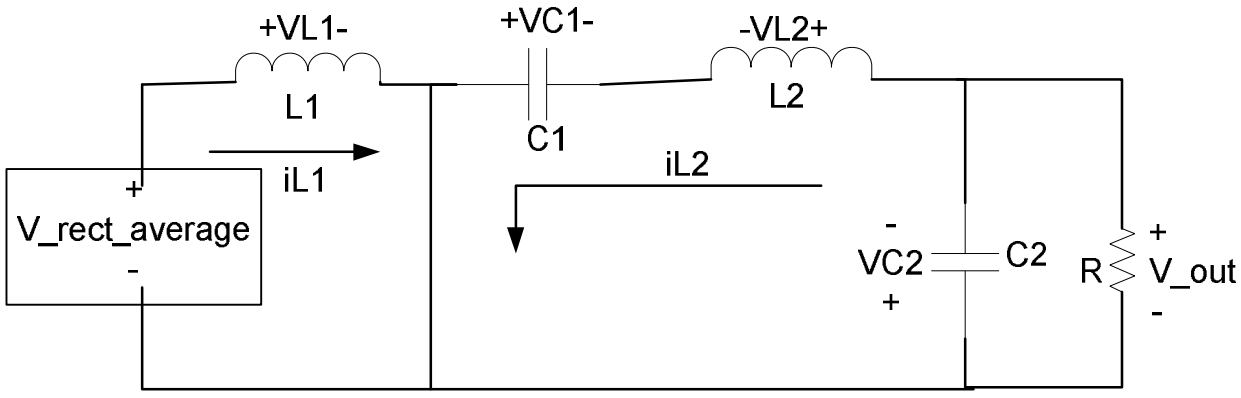
$$\frac{diL1}{dt} = \frac{V_{rect\_average}}{L1} \quad \text{..... (2.16)}$$

$$\frac{diL2}{dt} = \frac{VC1-VC2}{L2} \quad \text{..... (2.17)}$$

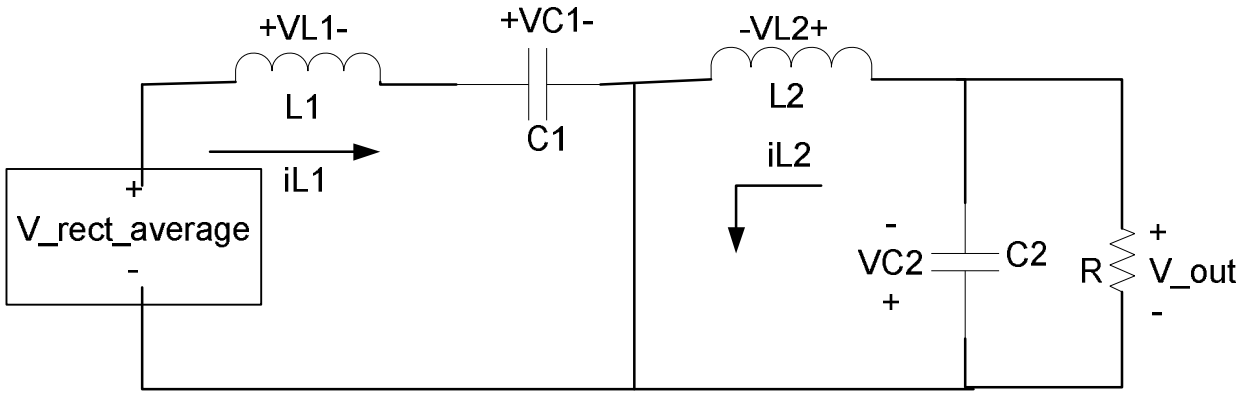
$$\frac{dVC1}{dt} = \frac{-iL2}{C1} \quad \text{..... (2.18)}$$

$$\frac{dVC2}{dt} = \frac{iL2-\frac{VC2}{R}}{C2} \quad \text{..... (2.19)}$$

V_out = -VC2 ..... (2.20)



(a)



(b)

Fig 2.46 Ĉuk converter (a) when switch is closed (b) when switch is opened

When switch is off for (1-d)T time then from Fig 2.46(b) we have

$$\frac{diL1}{dt} = \frac{-VC1 + V_{rect\_average}}{L1} \quad \text{..... (2.21)}$$

$$\frac{diL2}{dt} = \frac{-VC2}{L2} \quad \text{..... (2.22)}$$

$$\frac{dVC1}{dt} = \frac{iL1}{C1} \quad \text{..... (2.23)}$$

$$\frac{dVC2}{dt} = \frac{iL2 - \frac{VC2}{R}}{C2} \quad \text{..... (2.24)}$$

V_out = -VC2 ..... (2.25)

Therefore

$$A_1 = \begin{vmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{L2} & \frac{-1}{L2} \\ 0 & \frac{-1}{C1} & 0 & 0 \\ 0 & \frac{1}{C2} & 0 & \frac{-1}{RC2} \end{vmatrix}, \quad B_1 = \begin{vmatrix} \frac{1}{L1} \\ 0 \\ 0 \\ 0 \end{vmatrix}, \quad C_1 = |0 \quad 0 \quad 0 \quad -1| \text{ and } E_1 = [0] \text{ .....(2.26)}$$

$$A_2 = \begin{vmatrix} 0 & 0 & \frac{-1}{L1} & 0 \\ 0 & 0 & 0 & \frac{-1}{L2} \\ \frac{1}{C1} & 0 & 0 & 0 \\ 0 & \frac{1}{C2} & 0 & \frac{-1}{RC2} \end{vmatrix}, \quad B_2 = \begin{vmatrix} \frac{1}{L1} \\ 0 \\ 0 \\ 0 \end{vmatrix}, \quad C_2 = |0 \quad 0 \quad 0 \quad -1| \text{ and } E_2 = [0] \text{ .....(2.27)}$$

So using the state space Averaging method we have

$$A = \begin{vmatrix} 0 & 0 & \frac{-D'}{L1} & 0 \\ 0 & 0 & \frac{D}{L2} & \frac{-1}{L2} \\ \frac{D'}{C1} & \frac{-D}{C1} & 0 & 0 \\ 0 & \frac{1}{C2} & 0 & \frac{-1}{RC2} \end{vmatrix}, \quad B = \begin{vmatrix} \frac{1}{L1} \\ 0 \\ 0 \\ 0 \end{vmatrix}, \quad C = |0 \quad 0 \quad 0 \quad -1|, \; E = [0] \ldots\ldots\ldots\ldots (2.28)$$

$$B_d = \begin{vmatrix} \frac{V\_rect\_average}{D'L1} \\ \frac{V\_rect\_average}{D'L2} \\ \frac{-V\_rect\_averageD}{RC1D'^2} \\ 0 \end{vmatrix} \text{ and } E_d = [0] \ldots\ldots\ldots\ldots (2.29)$$

2.4 Stability analysis of the controller using Bode plot

The main objective of the control system is to draw a sinusoidal current, in phase with the input voltage. The reference inductor current $i_L^*(t)$ as shown in Figure 2.47 is of the full wave rectified form. The requirements on the form and the amplitude of the inductor current lead to two control loops as shown in Figure 2.47 to pulse width modulate the switch of the Ĉuk converter. The inner current loop ensures the form of $i_L^*(t)$ based on the input voltage. The outer voltage loop determines the amplitude $\grave{I}_L^*$ of $i_L^*(t)$ based on the output voltage feedback. If the inductor current is insufficient for a given load supplied by the control system, the output voltage will drop below its pre-selected reference valueV_d^*. By measuring the output voltage and using it as the feedback signal the voltage loop adjust the inductor current amplitude to bring the output voltage to its reference value. In addition to determining the inductor current amplitude, this voltage feedback control acts to regulate the output voltage to the preselected dc voltage. The inner current loop is shown in Figure 2.46 within the dotted box. In order to follow the reference with as little THD as possible, an average current mode control is used with a high bandwidth, where the error between the reference $i_L^*(t)$ and the measured inductor current $i_L(t)$ is amplified by a current controller to produce the control voltage $v_c(t)$. This control voltage is compared with the saw-tooth $v_r(t)$ with a peak of V_r at the switching frequency to produce the switching signal. Now we

can separate the control loop into two loops and analysis its stability. The current control loop and voltage control loop are shown in Fig 2.48 and Fig 2.49 respectively.

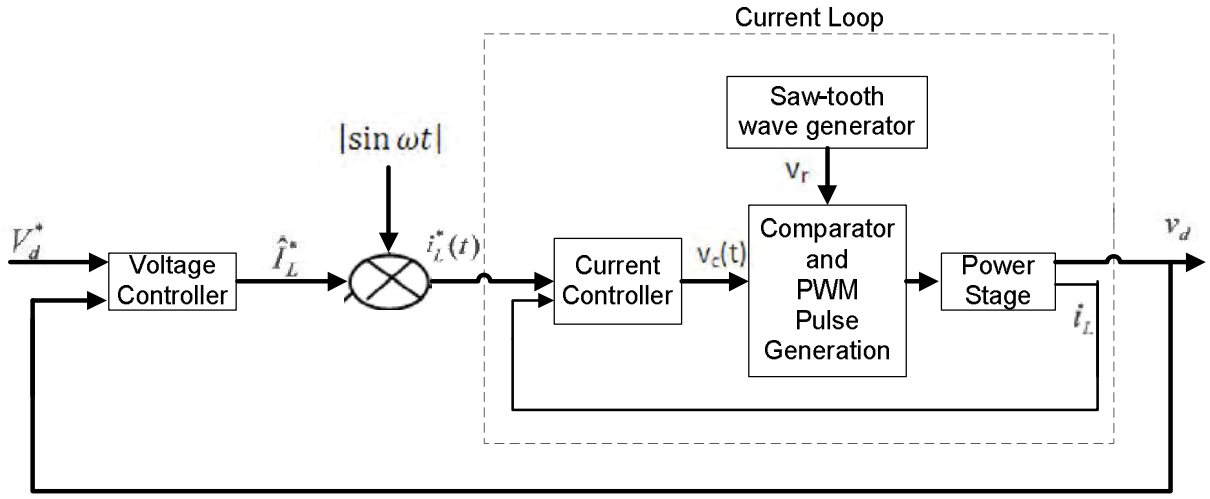


Fig 2.47 control loop of the proposed control system.

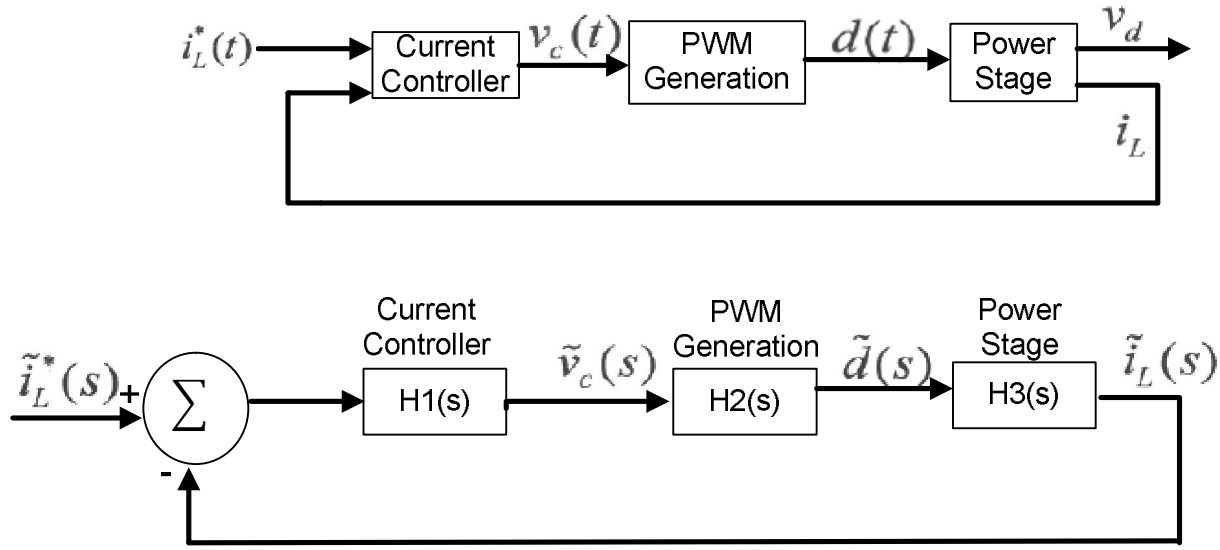


Fig. 2.48 current loop

Open loop transfer function is given by the following equation

$T_I(s)$ = H1(s) H2(s) H3(s)……..(2.30)

Where,

H1(s) = $K_{pI} + \frac{K_{iI}}{S}$ [PI controller]

H2(s) = $\frac{1}{V_r}$

Form Appendix 2

H3(s) = $\frac{2.121\times10^{-12}S^3-8.094\times10^{7}S^2+2.576\times10^{12}S-2.76\times10^{16}}{S^4+1.667S^3+9.849\times10^{8}S^2+1.641\times10^{9}S+1.55\times10^{13}}$

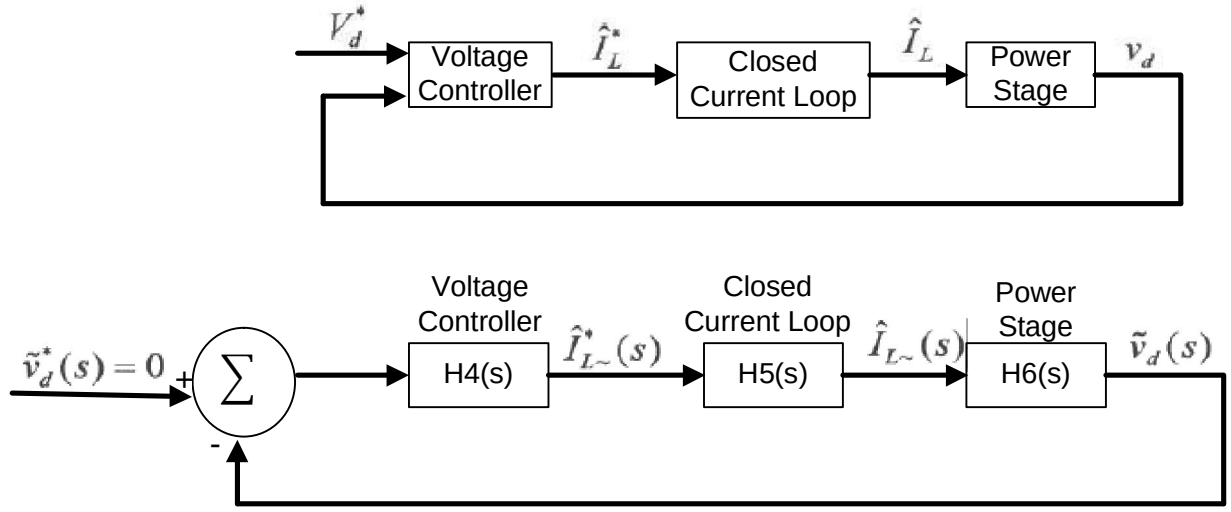


Fig. 2.49 voltage loop

Open loop transfer function is given by the following equation

$T_V(s)$ = H4(s) H5(s) H6(s)…….(2.31)

Where

H4(s) = $K_{pV} + \frac{K_{iV}}{S}$ [PI controller]

H5(s) = 1

Form Appendix 3

$$H6(s) = \frac{2.196\times10^{-12}S^7 - 2.961\times10^{7}S^6 - 2.965\times10^{12}S^5 - 2.192\times10^{17}S^4 - 2.471\times10^{22}S^3 + 2.3\times10^{26}S^2 + 3.997\times10^{25}S + 3.198\times10^{30}}{2.221\times10^{4}S^7 + 1.073\times10^{9}S^6 + 2.71\times10^{14}S^5 + 8.943\times10^{18}S^4 + 7.154\times10^{23}S^3 + 3.7\times10^{24}S^2 + 9.947\times10^{27}S + 3.314\times10^{28}}$$

Bode plot of transfer function H3(s) with PI control and without PI control is shown in Fig. 2.50 and Phase margin and gain margin of current control loop with PI control is shown in Fig. 2.51. Bode plot of transfer function H6(s) with PI control and without PI control is shown in Fig. 2.52 and Phase margin and gain margin of voltage control loop with PI control is shown in Fig. 2.53. Fig. 2.54 shows the overall system bode plot and Fig. 2.55 shows Phase margin and gain margin of the overall system. For the closed loop system to be stable if its gain cross over frequency is less than the phase cross over frequency in open loop transfer function or at the cross over frequency the phase delay introduced by the loop transfer function is less than 180 degree. From the bode plot it is clearly shown that the control loop fulfill the stability criteria.

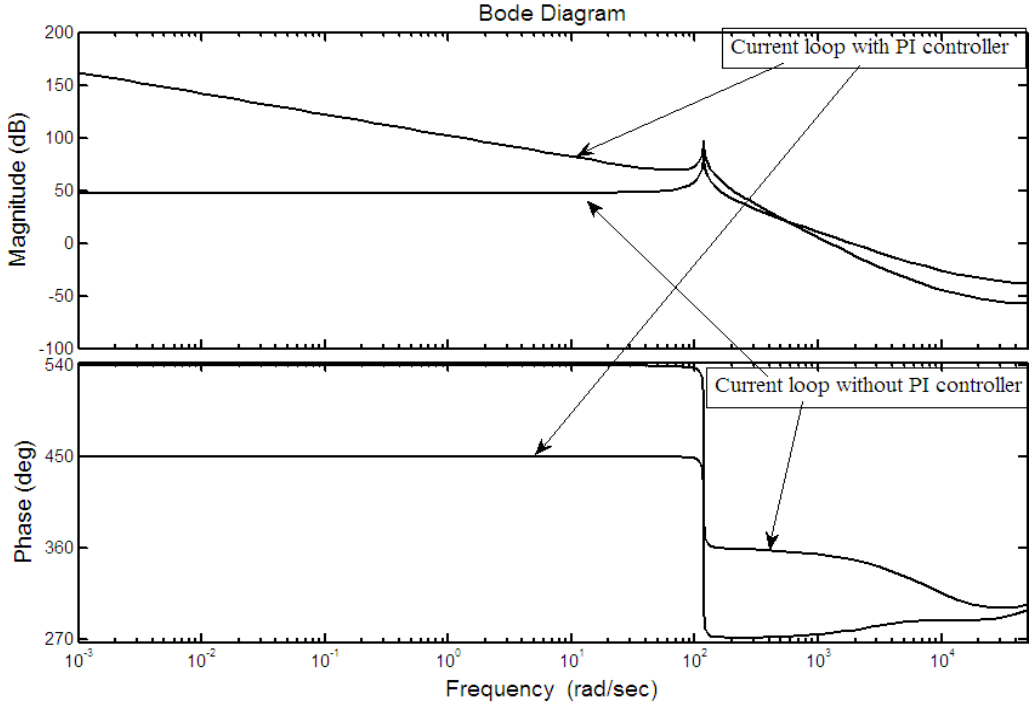


Fig. 2.50 Stability analysis of current loop using bode plot

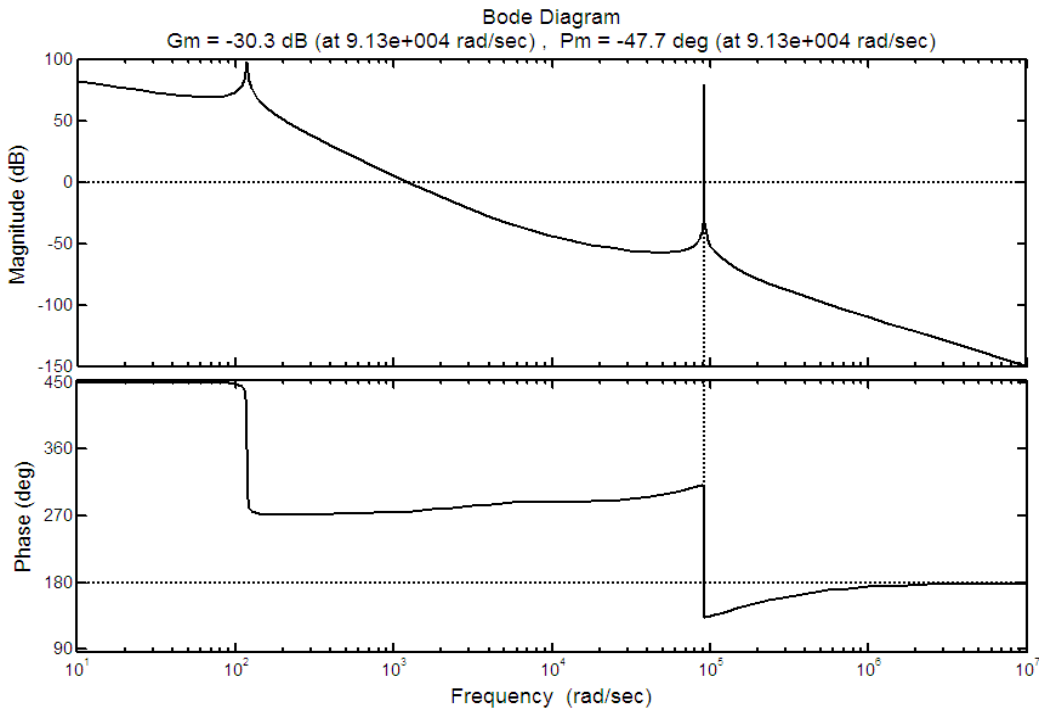


Fig. 2.51 Phase margin and gain margin of current loop using bode plot with PI controller

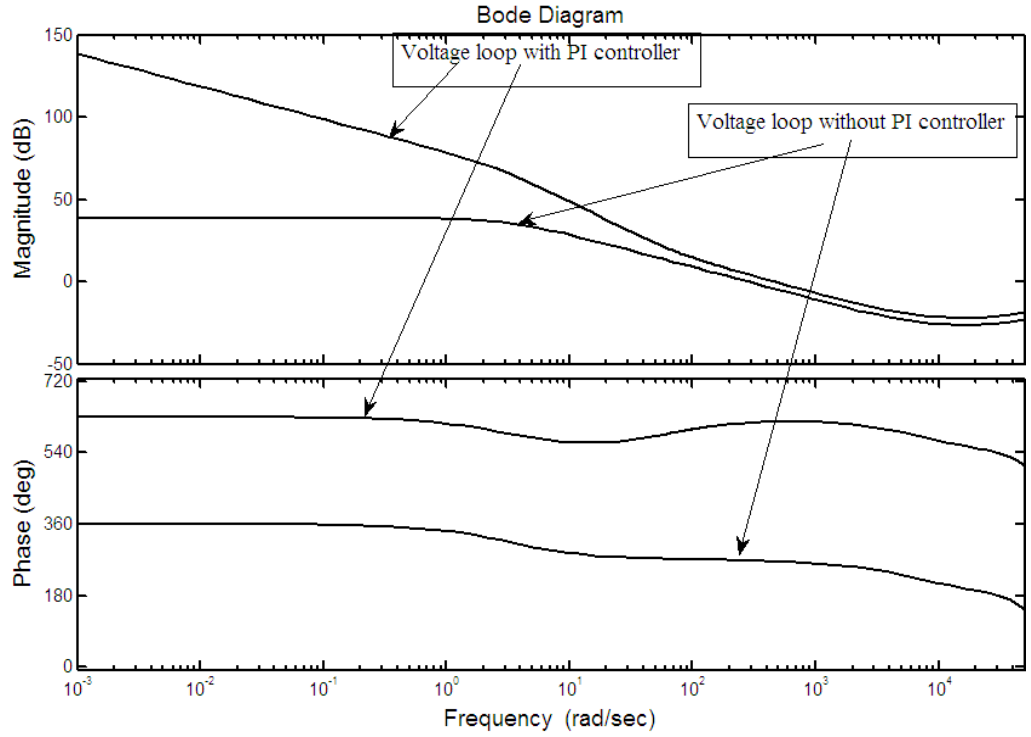


Fig. 2.52 Stability analysis of voltage loop using bode plot

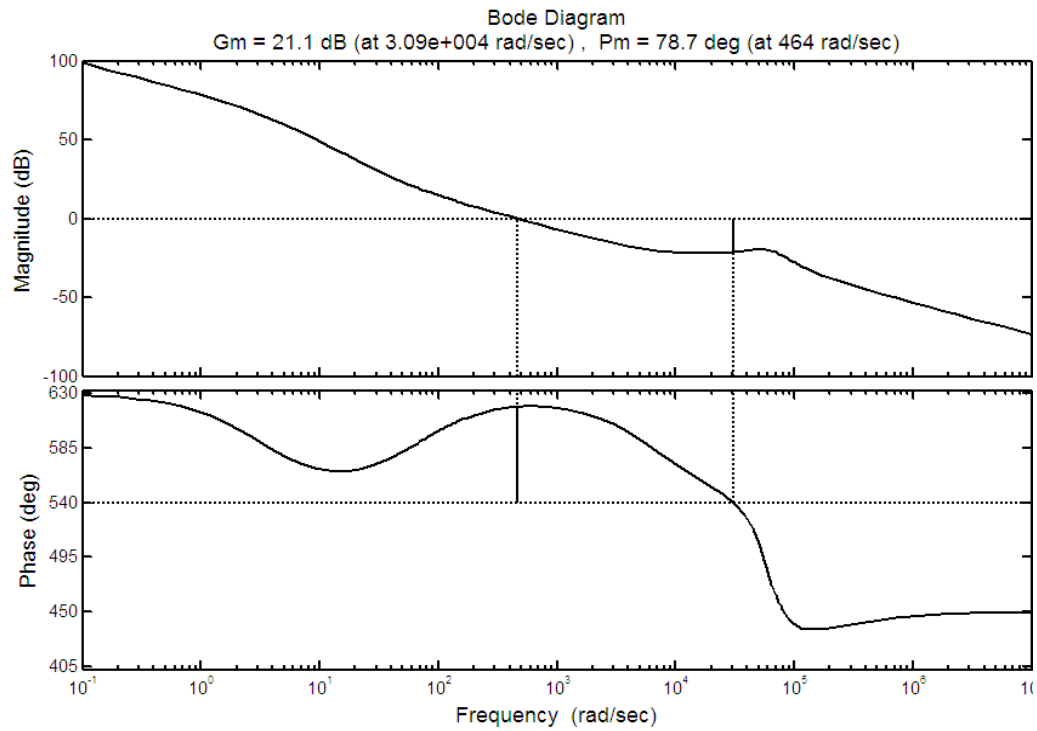


Fig. 2.53 Phase margin and gain margin of voltage loop using bode plot with PI controller

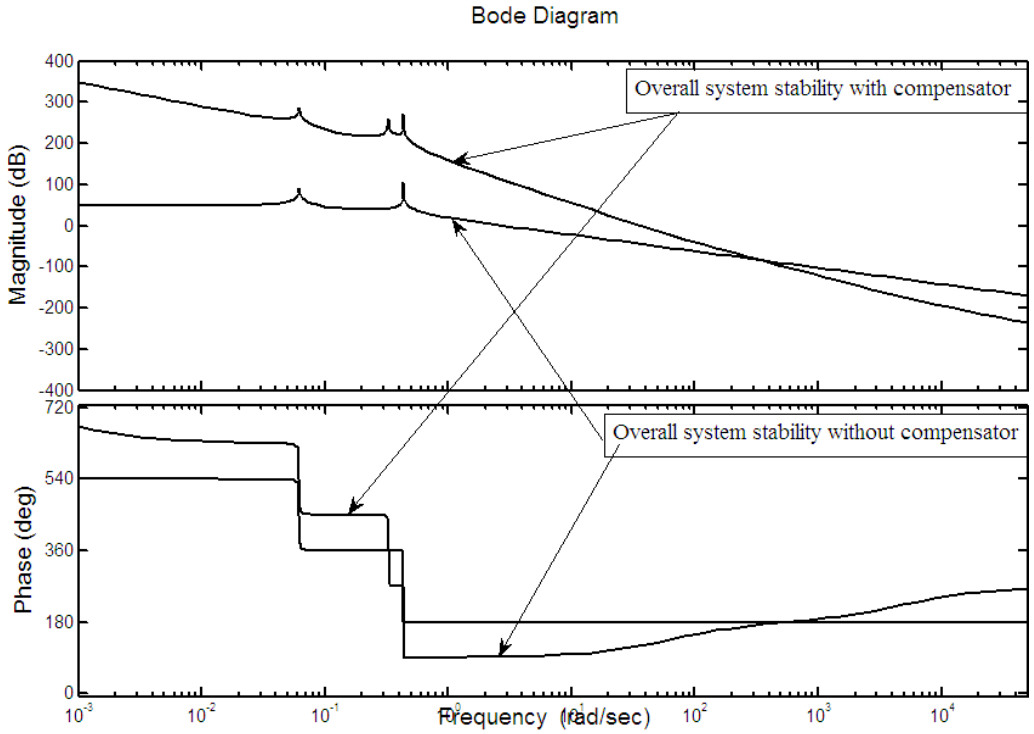


Fig. 2.54 Stability analysis of overall system (output voltage to duty ratio) using bode plot

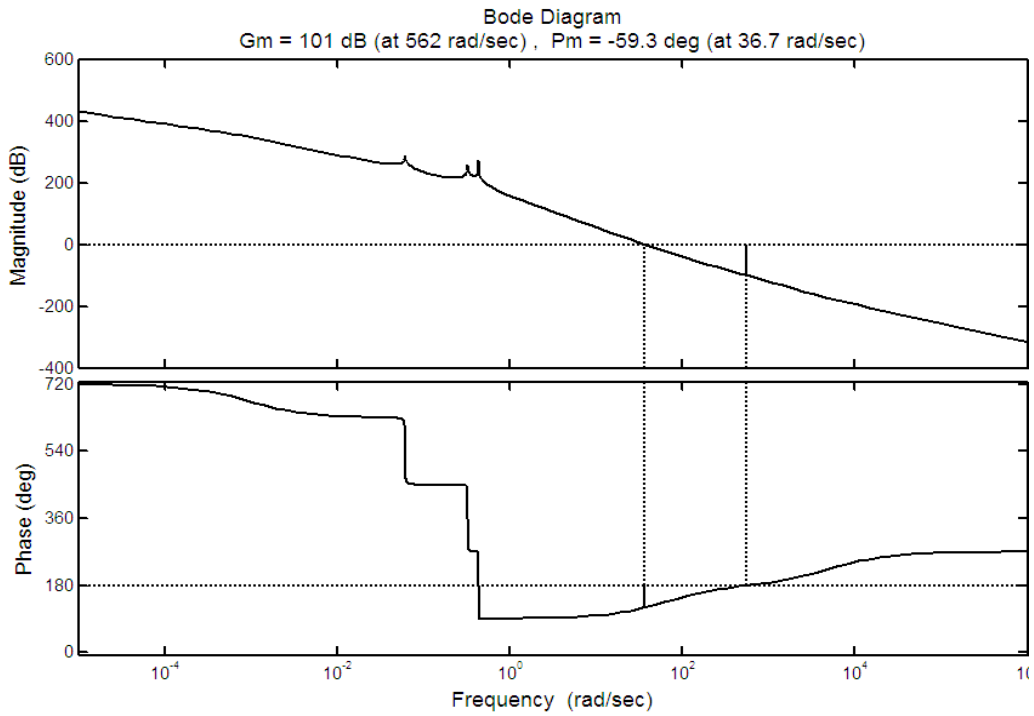


Fig. 2.55 Phase margin and gain margin of overall system (output voltage to duty ratio) with compensator

Fig. 2.56 to Fig. 2.63, Fig. 2.65 to Fig. 2.66 and Fig. 2.68 to Fig. 2.69 are simulated using proposed system. The following parameter value is used

L= 100uH, L1= 10mH, L2=5mH, C1=0.1uF, C2=1500uF and switching frequency = 80000Hz

For voltage loop $K_p = 27$ and $K_i = 0.1$ and for current loop $K_p = 0.2$ and $K_i = 0.4$ are used. The K_p and K_i are chosen by trial and error method to get better performance. Fig. 2.56 to Fig. 2.61 show the input voltage/current, real current and reference current, output voltage, input current THD, power factor and efficiency respectively for 255Ω load resistance and -400V reference voltage under 325V(peak) input voltage and 50Hz supply frequency. Fig. 2.62 and Fig. 2.63 shows the PWM pulse and output voltage for 400Ω and 800Ω load resistance respectively.

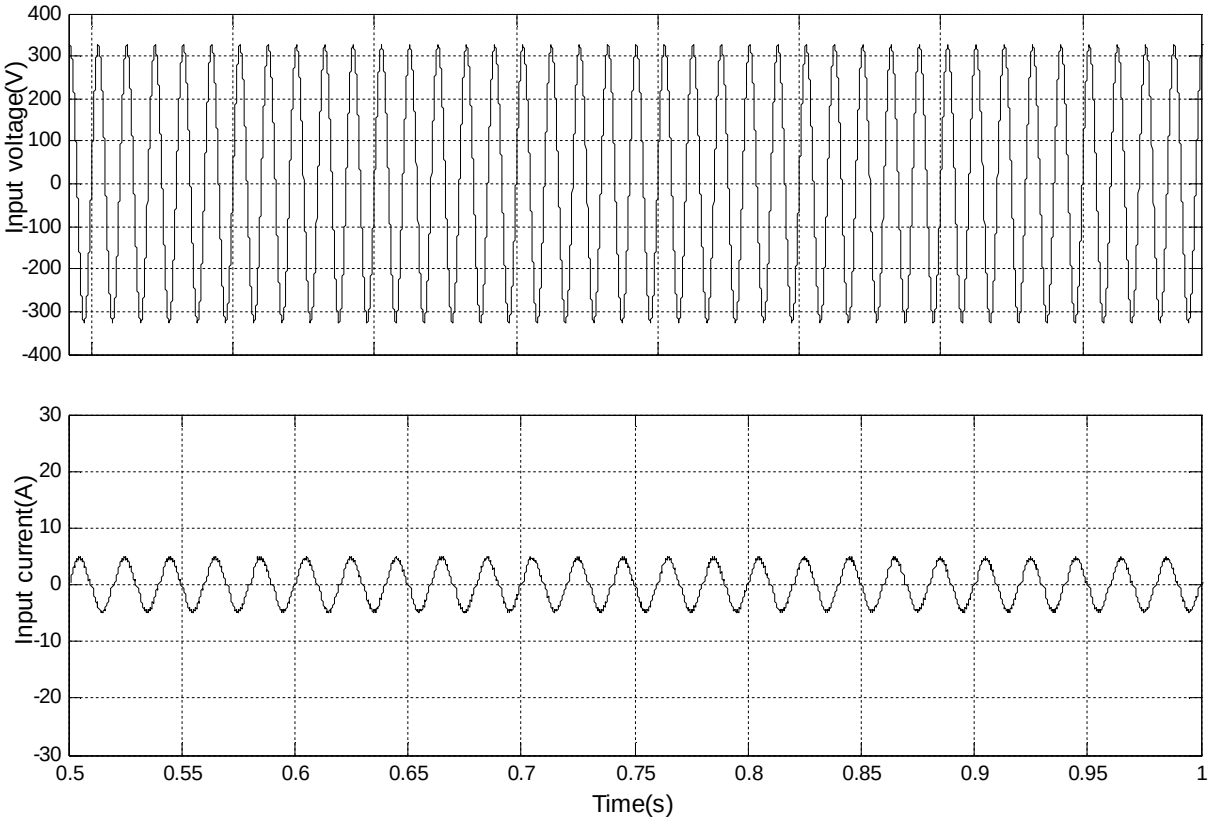


Fig. 2.56 Input voltage 325V (peak) and input current for 255Ω load and -400V output voltage reference

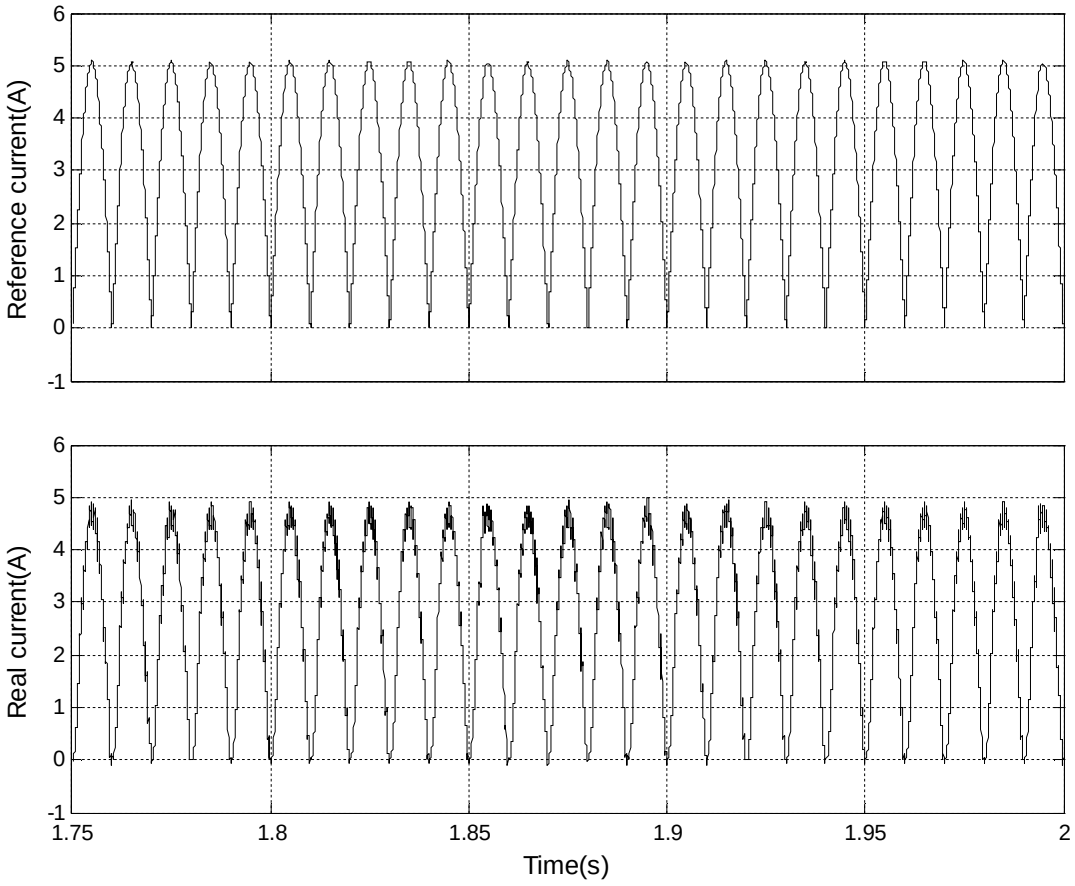


Fig. 2.57 Reference current and real current for 255Ω load and -400V output voltage

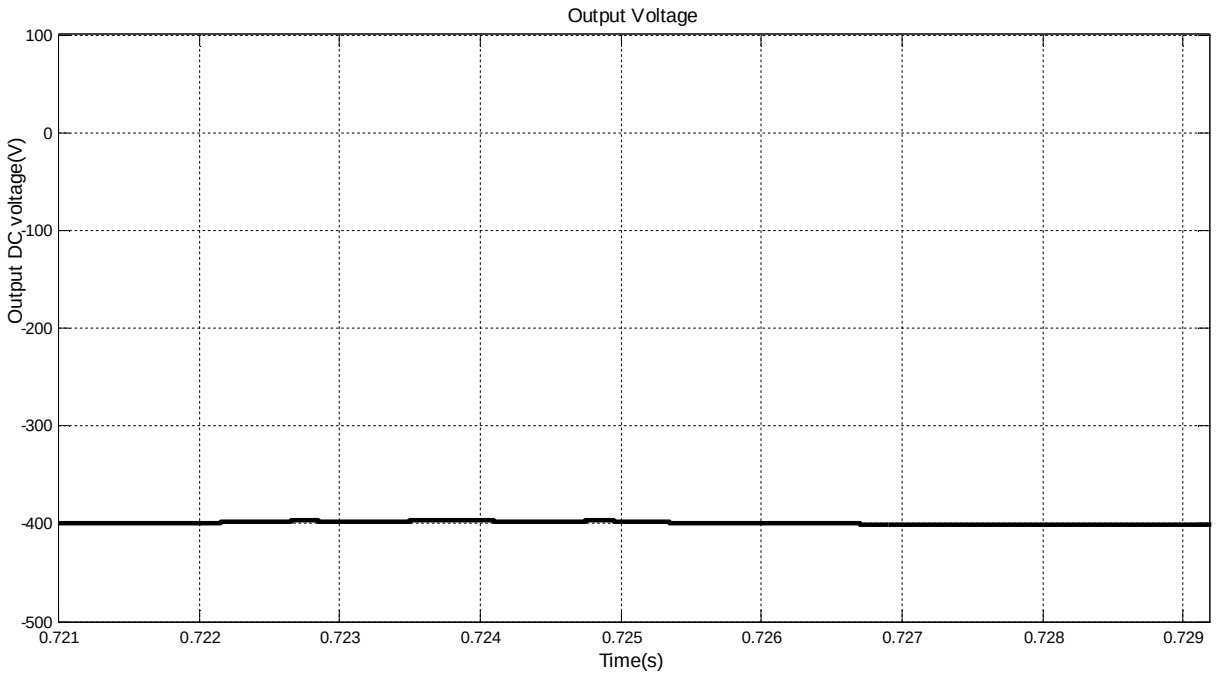


Fig. 2.58 Output voltage for 255Ω load and 325V (peak) input voltage

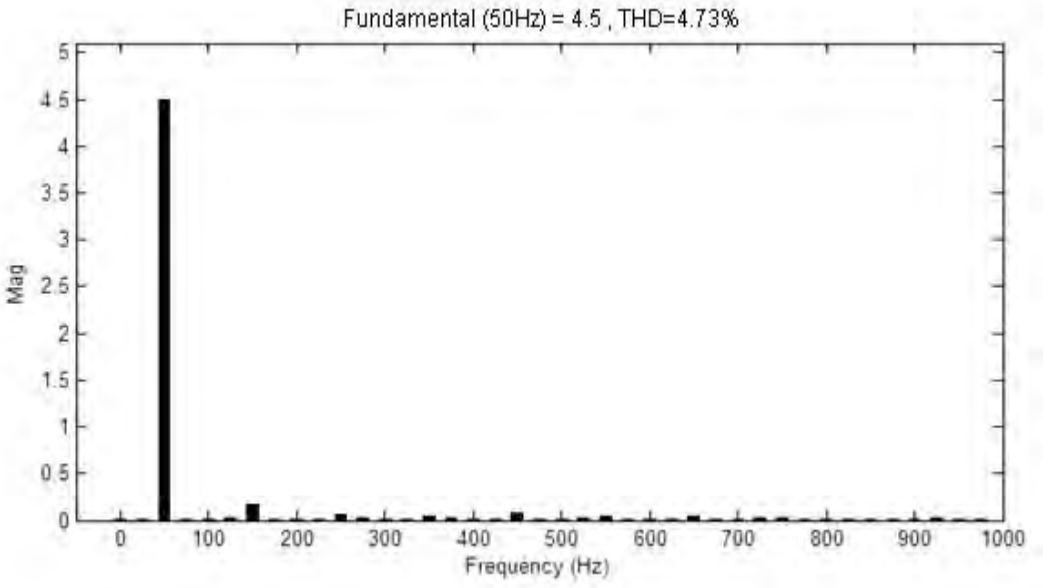


Fig 2.59 THD of input current for 255Ω load and 325V (peak) input voltage

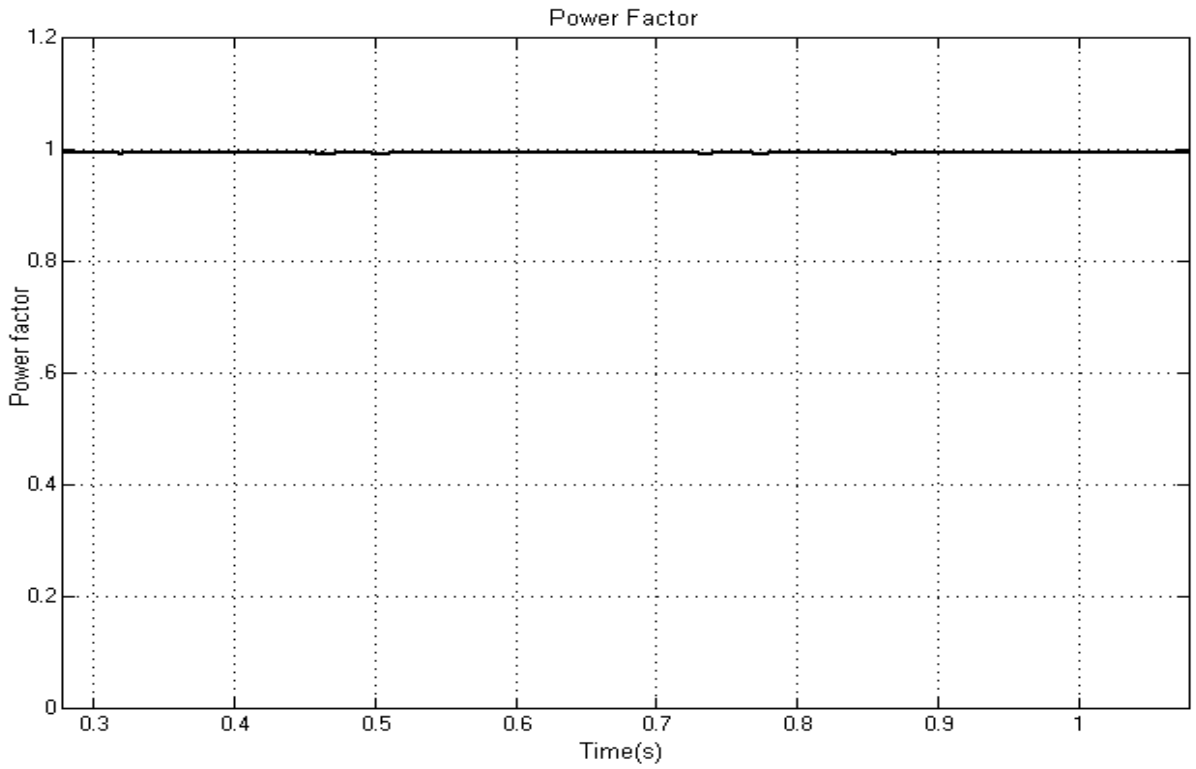


Fig 2.60 Input power factor for 255Ω load and 325V (peak) input voltage

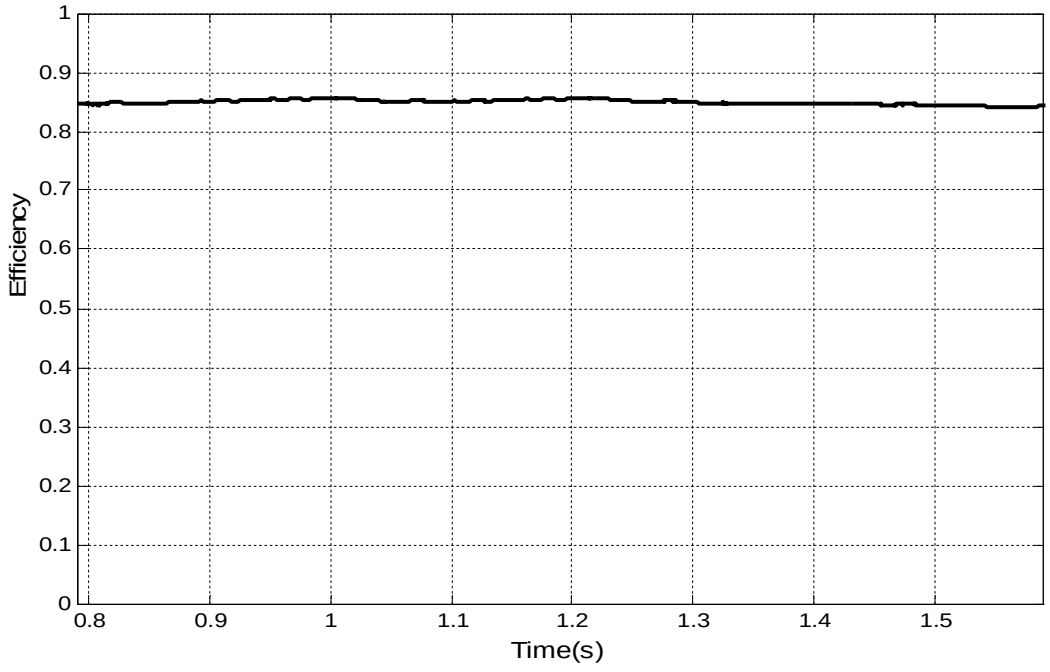


Fig 2.61 Efficiency for 255Ω load and 325V (peak) input voltage

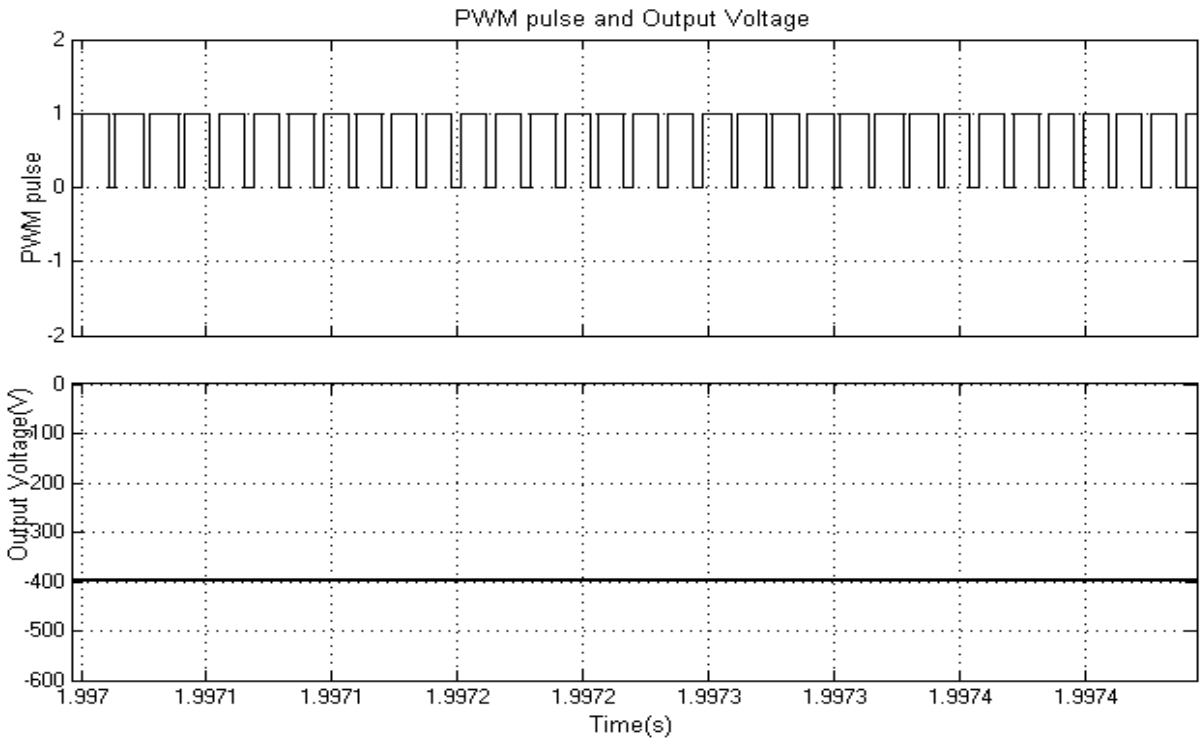


Fig 2.62 PWM pulse and output voltage for 400Ω load resistance

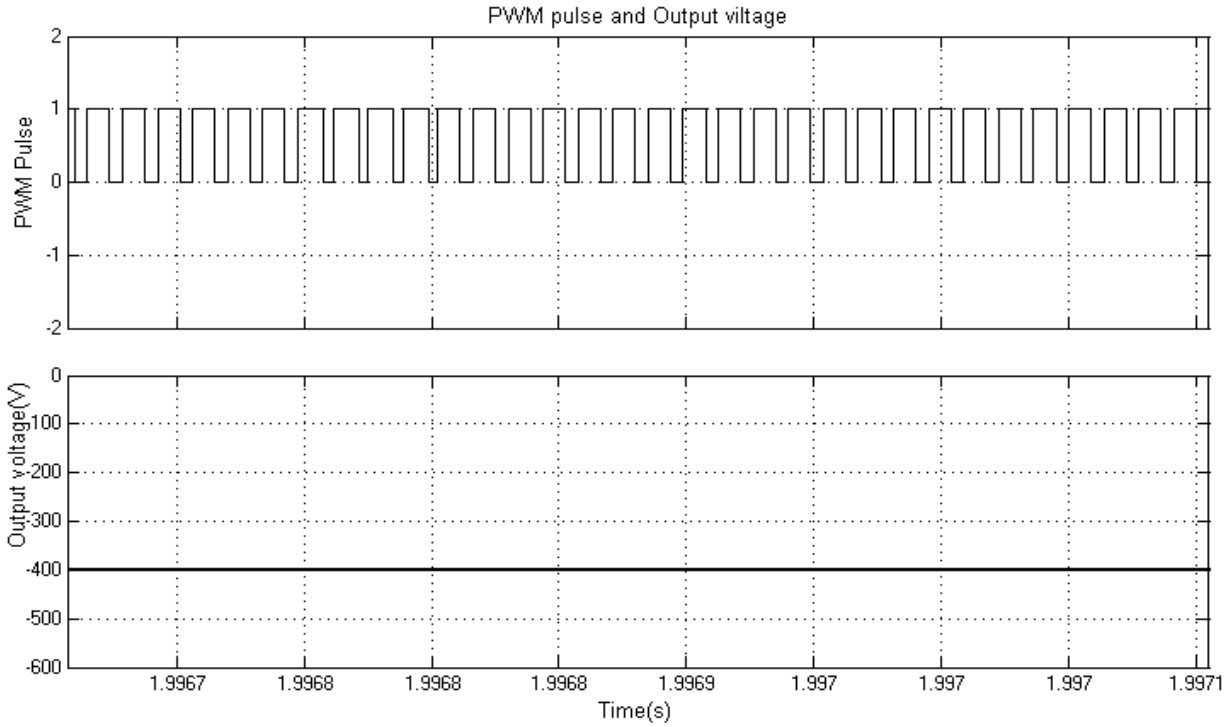


Fig 2.63 PWM pulse and output voltage for 800Ω load resistance

2.5 Simulation under load change

For observing the dynamic response of the proposed system under load changed the technique as shown in Fig 2.64 in matlab simulink is used. Here breaker will close after 0.5s and Load2 will connect in parallel with Load1. Fig. 2.65 to Fig. 2.66 show the effect of load change from 200W to 400W on output voltage and reference/real current respectively.

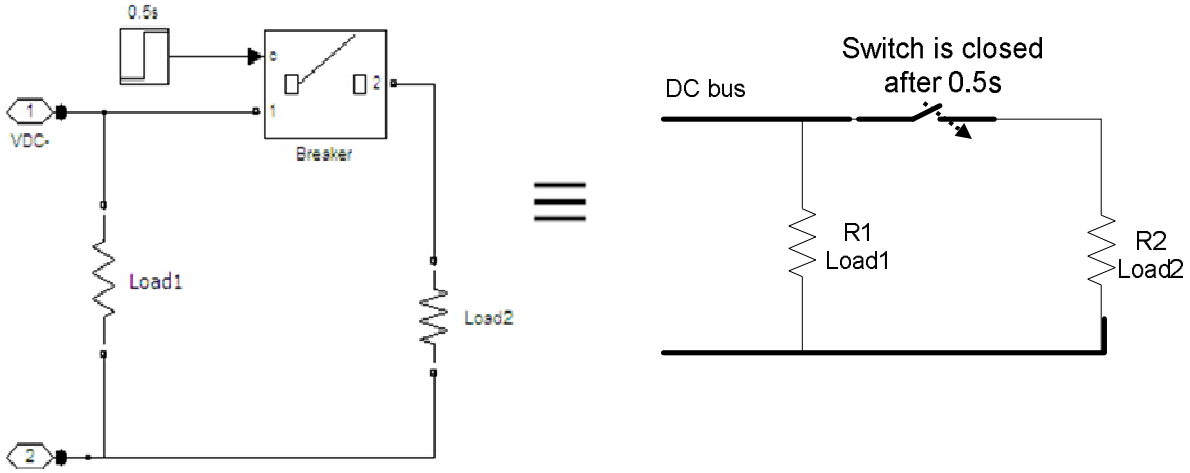


Fig 2.64 Load changing after 0.5s

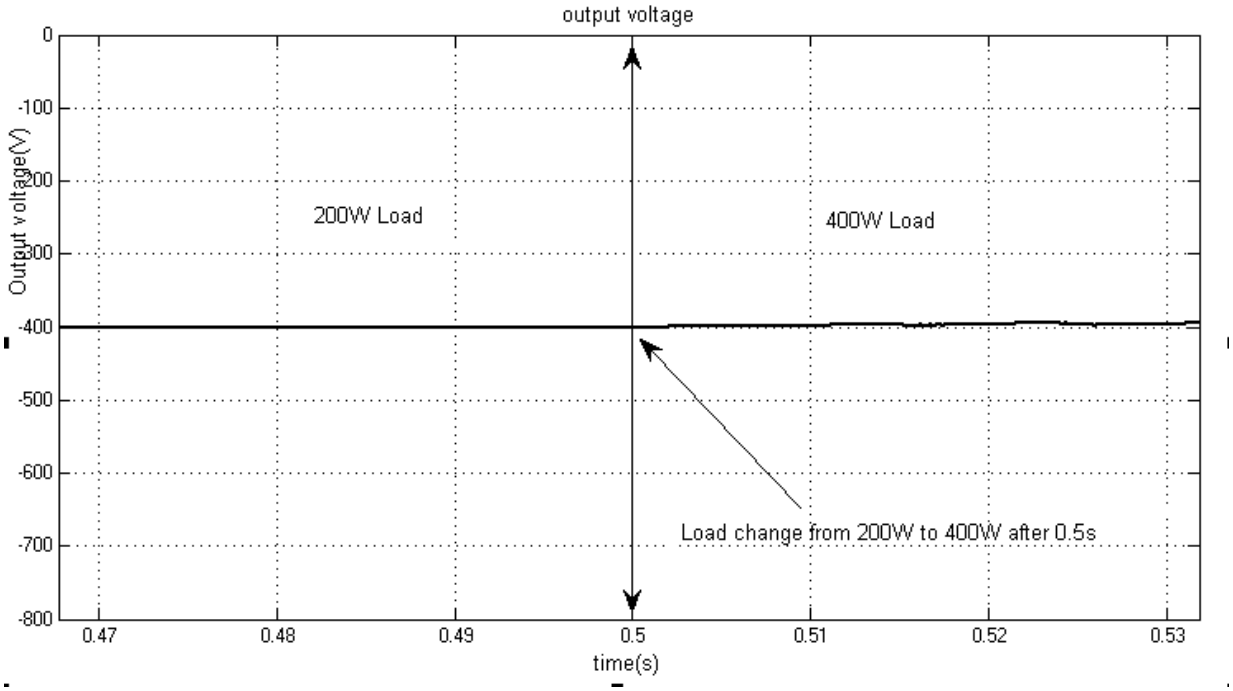


Fig 2.65 Output voltage under load change from 200W to 400W

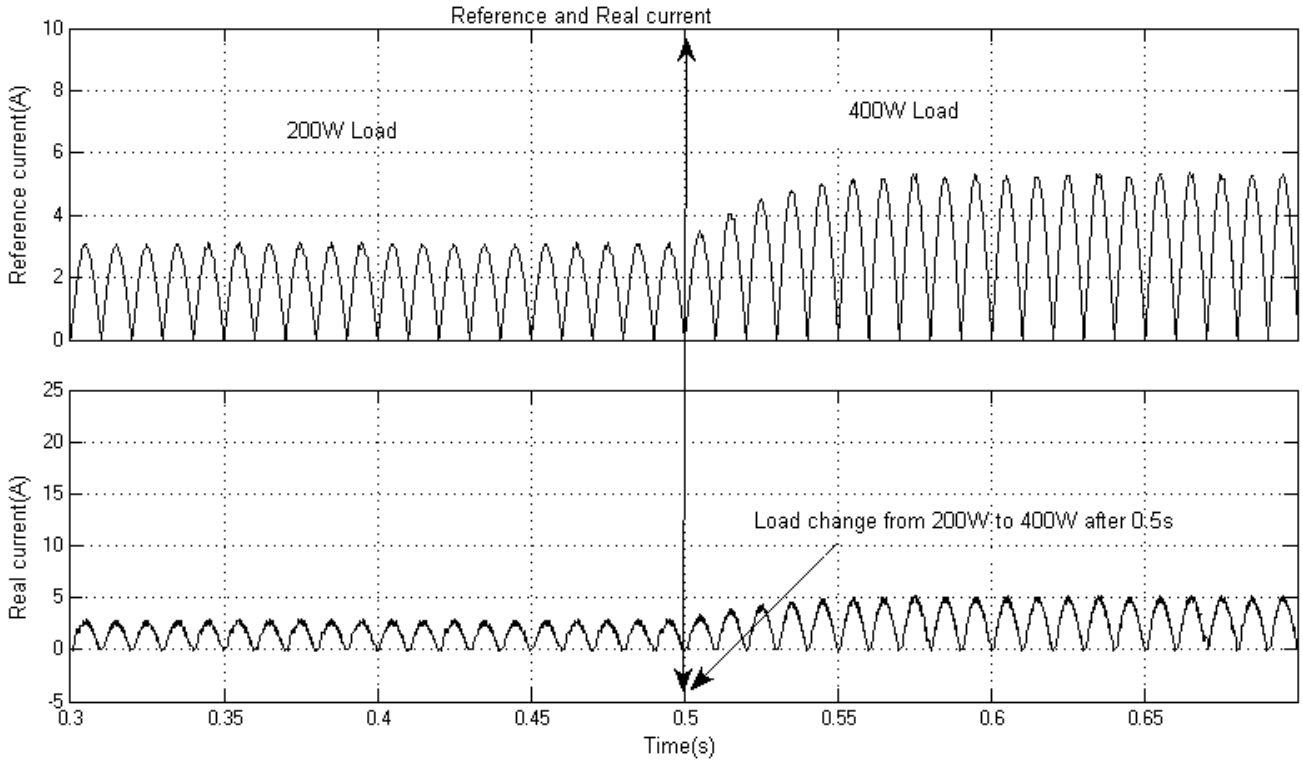


Fig 2.66 Reference and real current under load change from 200W to 400W

2.6 Simulation under input voltage change

For observing the dynamic response of the proposed system under input voltage changed the technique as shown in Fig 2.67 in matlab simulink is used. Here ideal switch will close after 0.6s and ideal switch1 will open after 0.6s. As a result the line voltage will switch from 200V (rms) to 230V (rms).

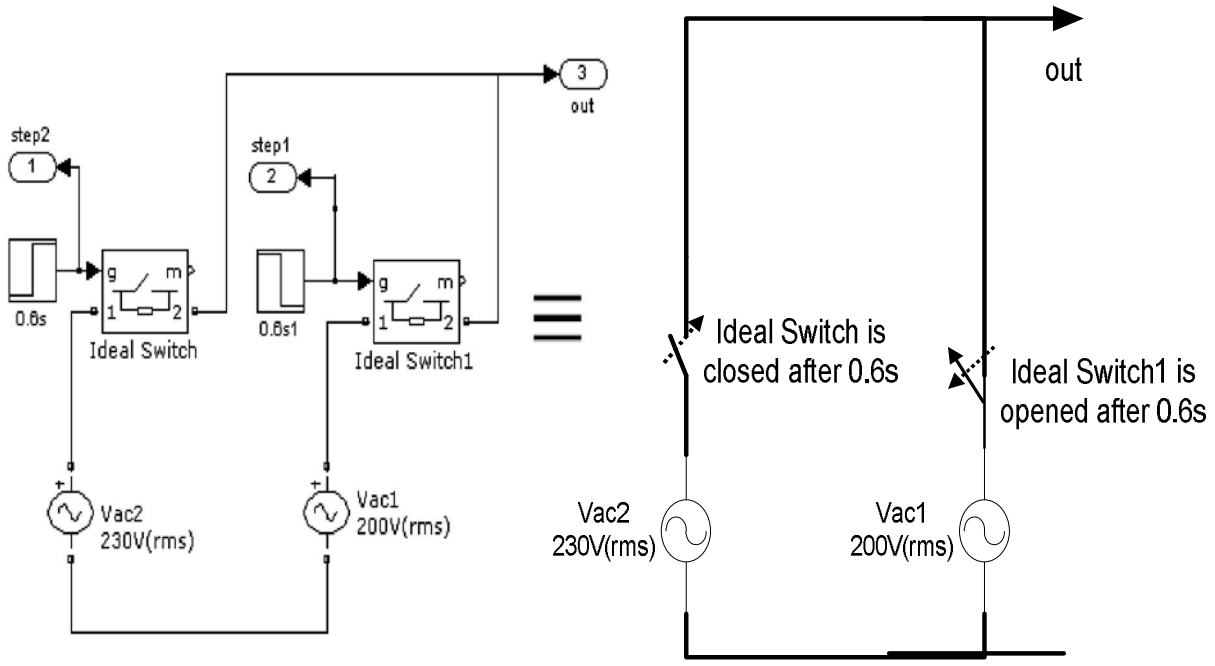


Fig 2.67 Input voltage changing arrangement

Fig. 2.68 and Fig. 2.69 show the effect of input voltage changes from 283V (peak) to 325V (peak) and 311V (peak) to 367V (peak) respectively on output voltage change.

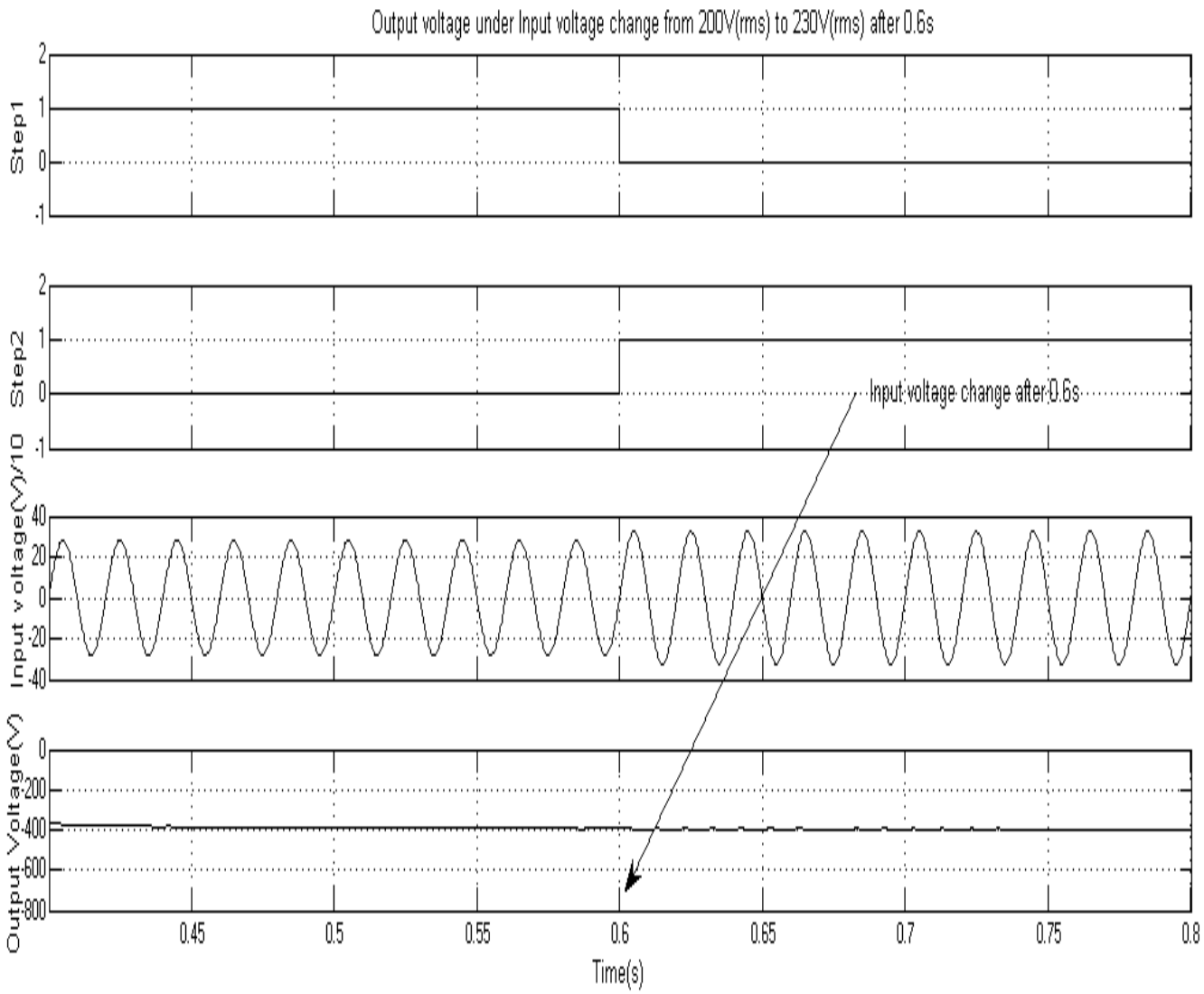


Fig 2.68 Output voltage under input voltage change from scaled 283V (peak) to scaled 325V (peak) after 0.6s

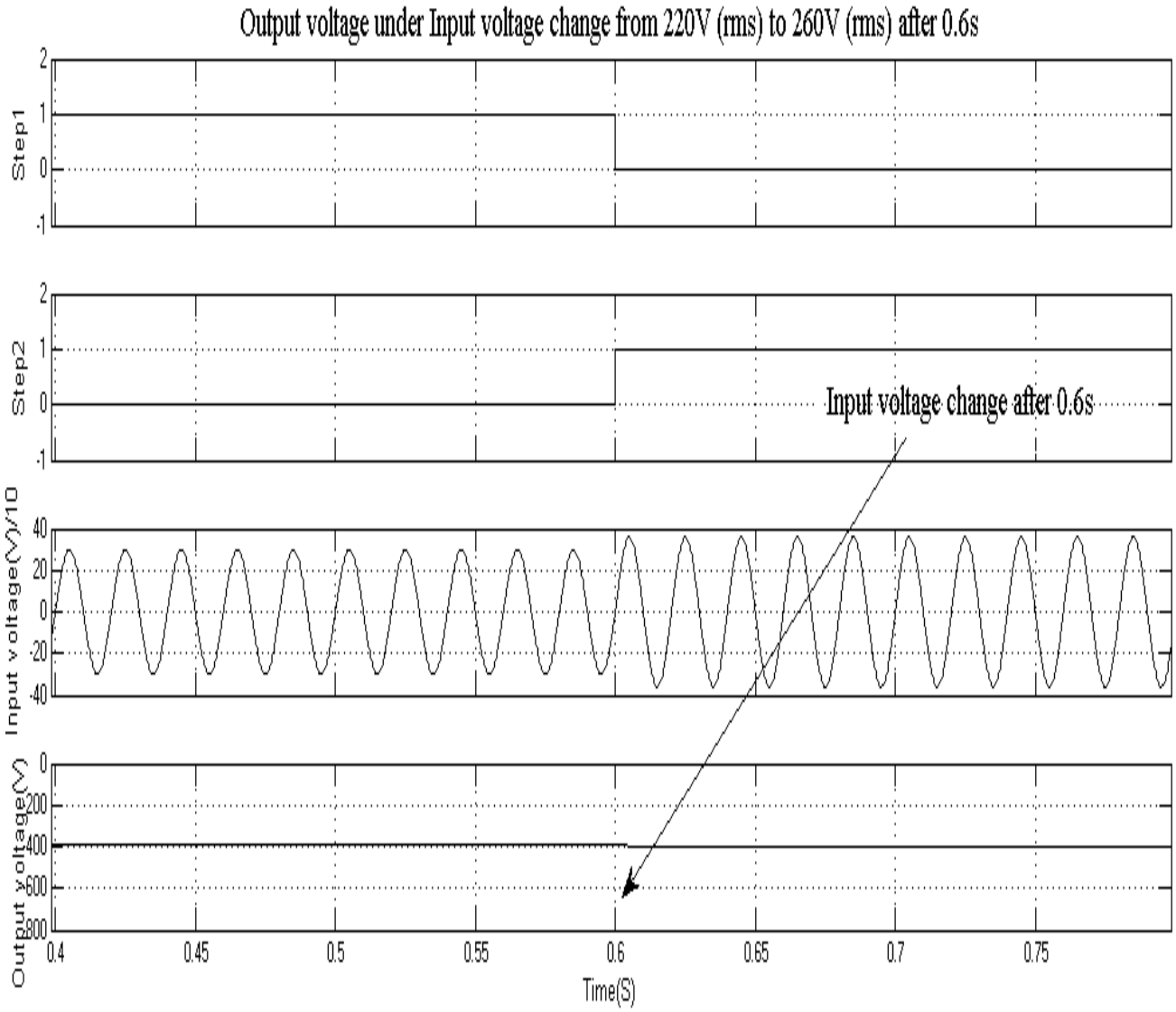


Fig 2.69 Output voltage under input voltage change from scaled 311V (peak) to scaled 367V (peak) after 0.6s

CHAPTER 3

REAL TIME PROGRAM ALGORITHM AND SIMULATION USING PROTEUS SOFTWARE THROUGH DSP PROCESSOR

From the previous chapter we have seen how the proposed system is designed and modeled in Matlab simulink. In this chapter the model system will be implemented and simulated in the real circuit. Since the input rectified AC voltage, current and output DC voltage is sensed so proper sensing network is needed to interface with the low voltage operated DSP processor. Program algorithm for implementing proportional integral control block along with all sensing parameter and data processing will be discussed in detail. DSP processor has been chosen to carry out all of the operation due to its capability to convert several analog signals to digital signal simultaneously over other microcontrollers.

3.1 PROTEUS SIMULATION SOFTWARE

Proteus Virtual System Modeling (VSM) combines mixed mode SPICE circuit simulation, animated components and microprocessor models to facilitate co-simulation of complete microcontroller based designs. At the heart of Proteus VSM is ProSPICE. This is an established product that combines uses a SPICE3f5 analogue simulator kernel with a fast event-driven digital simulator to provide seamless mixed-mode simulation. Proteus VSM includes a number of virtual instruments including an Oscilloscope, Logic Analyser, Function Generator, Pattern Generator, Counter Timer and Virtual Terminal as well as simple voltmeters and ammeters. The most exciting and important feature of Proteus VSM is its ability to simulate the interaction between software running on a microcontroller and any analog or digital electronics connected to it. The micro-controller model sits on the schematic along with the other elements of our design. It simulates the execution of our object code (machine code), just like a real chip. If the program code writes to a port, the logic levels in circuit change accordingly, and if the circuit changes the state of the processor's pins, this will be seen by our program code, just as in real life.

The VSM CPU models fully simulate I/O ports, interrupts, timers, USARTs and all other peripherals present on each supported processor. It is anything but a simple software simulator since the interaction of all these peripherals with the external circuit is fully modeled down to

waveform level and the entire system is therefore simulated. VSM can even simulate designs containing multiple CPUs, since it is a simple enough matter to place two or more processors on a schematic and wire them together.

3.2 Sensing network design.

3.2.1 Input rectified AC voltage and output DC voltage.

Input rectified AC voltage is first downsized using a resistor dividing network to accepted value. The accepted value of the ADC pin of DSP processor is 3.3V or below. The following network as shown in Fig. 3.1 ensures that in any circumstances the input voltage in ADC pin will not exceed the 3.3V because diode D11 will be forward biased.

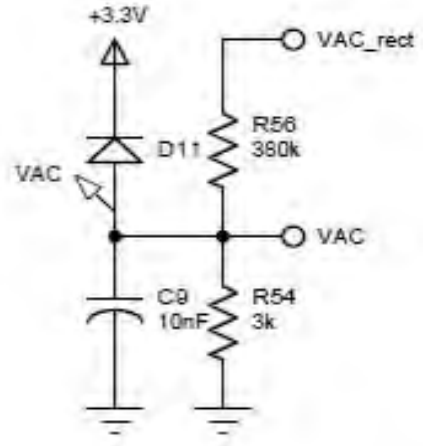


Figure 3.1 Voltage dividing resistor network

Let the input voltage 260V (rms) then the voltage across the resistor R54 is

$$\frac{3}{380 + 3} \times \sqrt{2} \times 260 = 2.88V$$

Similarly the following voltage dividing resistor network and inverting network as shown in Fig. 3.2 is used to sense the output voltage.

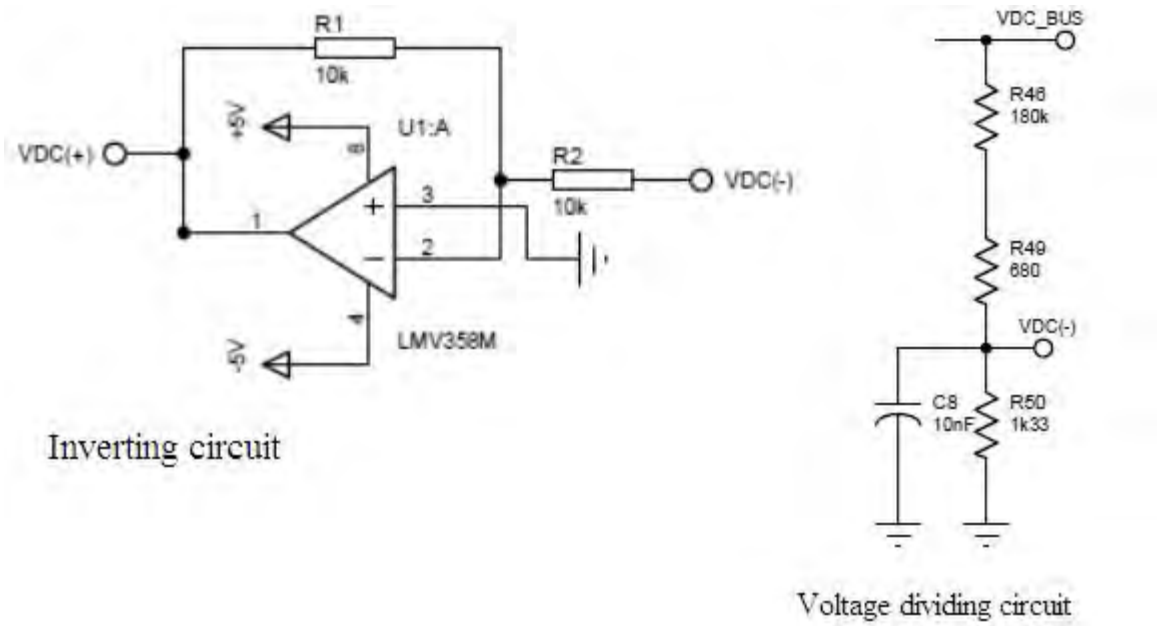


Figure 3.2 Voltage dividing resistor network and inverting network

Resistor dividing network downsizes the DC Bus voltage and inverting network inverts negative voltage to positive voltage. Notice that ADC input voltage must be positive voltage.

Let the output DC voltage -400V so the voltage across the resistor R50

$$-\frac{1.33}{180 + 0.68 + 1.33} \times 400 = -2.92V$$

And Op-amp output voltage =

$$-\frac{10}{10} \times (-2.92) = 2.92V$$

3.2.2 Input rectified Current.

Hall type current sensor (Model LTS6_np) is used to sense the input rectified DC Current. The following circuit as shown in Fig. 3.3 is used in this purpose.

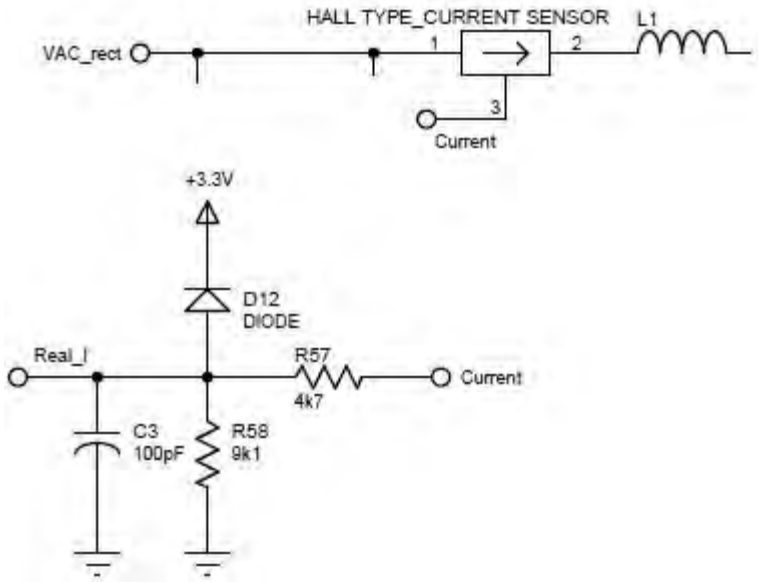


Fig. 3.3 Current sensing circuit

From the following characteristic curve as shown in Fig. 3.4 of LTS6_np we can easily see its output voltage is varied linearly in the range 2.5 ± 0.625V with respect to its input current.

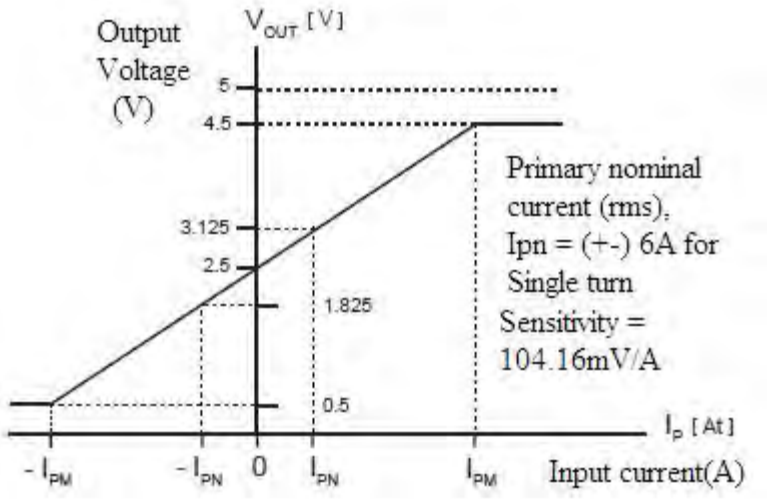


Fig 3.4 Input current versus output voltage characteristic curve [43]

So maximum voltage across the Resistor R58 is

$$\frac{9.1}{9.1 + 4.7} \times 3.125 = 2.06V$$

And minimum voltage across the Resistor R58 is

$$\frac{9.1}{9.1 + 4.7} \times 2.5 = 1.65V$$

3.3 Gate drive circuit

Since the voltage level of PWM signal is 3.3V so it cannot drive MOSFET directly though the MOSFT position is in low side in the circuit and also the MOSFET capacitor charging current can damage the controller. So it is better to use gate drive circuit. The following gate drive circuit as shown in Fig. 3.5 is used as a level converter.

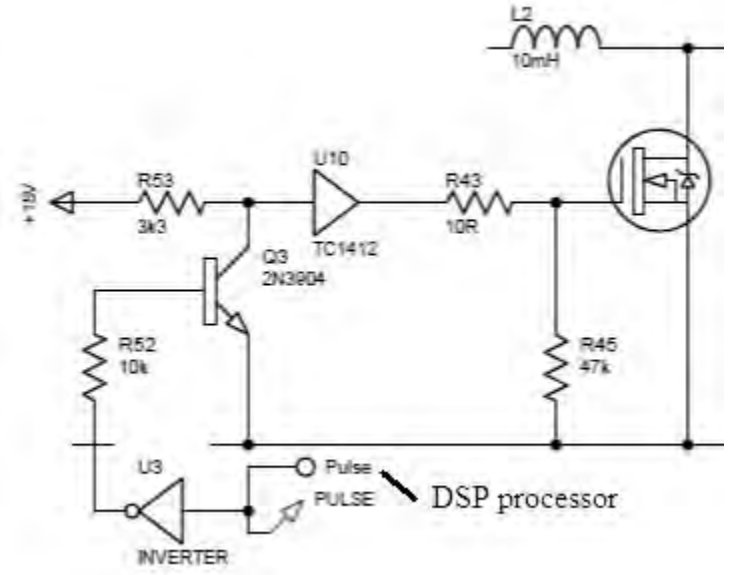


Fig. 3.5 Gate drive circuit

The high portion of the PWM signal makes the inverter output Low and transistor operates in cut off mode due to it's approximately zero base current. The voltage between collector and emitter is +15V and can easily drive the MOSFET through the TC1412 voltage buffer IC. Buffer IC is used to overcome the loading effect of the Transistor.

3.4 Complete circuit diagram

The following real time hardware circuit as shown in Fig. 3.6 is used to implement the Matlab simulink model of proposed closed loop Ĉuk based AC-DC converter.

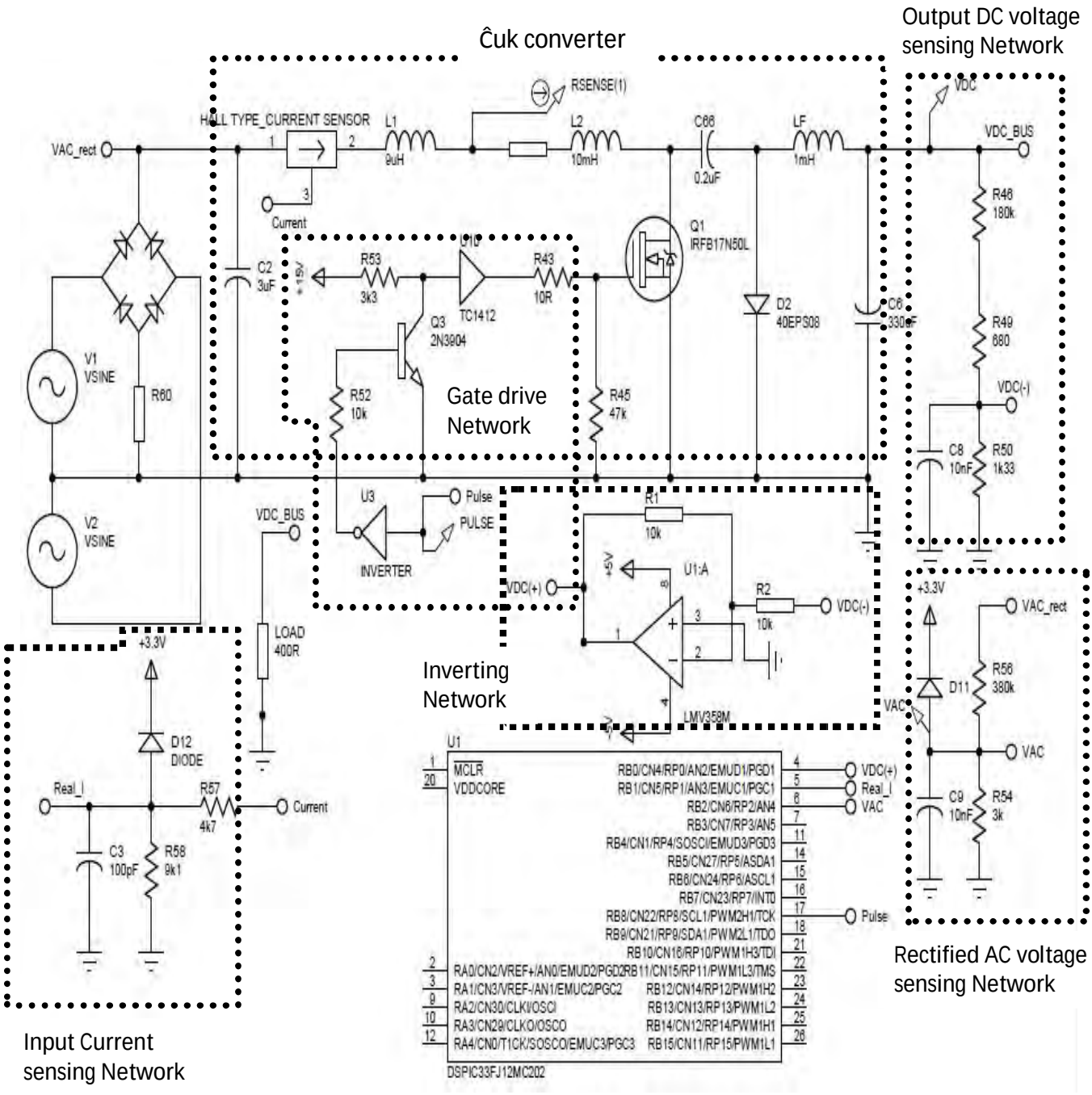


Fig. 3.6 Complete circuit diagram of the proposed closed loop Ĉuk AC-DC converter.

3.5 Average voltage computation in digital domain

In the analog domain, the continuous form of the average Voltage is

$$V_{AVG} = \frac{1}{T}\int_{t}^{t+T} V_{ac}.dt \quad \text{..... (3.1)}$$

Where,

V_{ac} = the instantaneous AC input voltage

T= time period depending on the frequency of the AC input voltage

In the digital domain the discrete form of this equation is

$$i = n+\frac{T}{T_S}$$

Therefore $$V_{AVG} = \frac{1}{T}\sum_{i=n} V_{AC}(i).T_S \quad \text{..... (3.2)}$$

Or $$V_{AVG} = \sum_{i=n} V_{AC}(i).\frac{1}{\frac{T}{T_S}} \quad \text{..... (3.3)}$$

Or $$V_{AVG} = \frac{\sum V_{AC}(i)}{N} \quad \text{..... (3.4)}$$

Where,

V_{AC} = Input voltage at the i^{th} sample.

N = Number of samples taken

To calculate N, which is given by $N = T/T_S$, the input line frequency, F = $1/T$, has to be computed with the control loop frequency $F_S = 1/T_S$. The PFC is implemented with a control loop frequency of 40 kHz running inside the ADC Interrupt Service Routine (ISR). A control loop frequency of 40 kHz is chosen to track the input voltage precisely and to shape the inductor current accurately. Based on this, the sampling time is as shown in the following equation

$$T_s = \frac{1}{40kHz} = 25us$$

The PFC software is designed for a line frequency range of 45Hz to 55Hz, as shown in the following Equation

F_{MIN} = 45Hz therefore $T_{MIN} = \frac{1}{45Hz}$ = 22.22ms and

F_{MAX} = 55Hz therefore $T_{MAX} = \frac{1}{55Hz}$ = 18.18ms

Given the previous calculations, the value of N has the range shown in the following Equation

$N_{MAX} = \frac{T_{MIN}}{T_S} = \frac{22.22mS}{25uS} = 888$ to

$N_{MIN} = \frac{T_{MAX}}{T_S} = \frac{18.18mS}{25uS} = 727$

However, since the rectified AC input voltage is at twice the line frequency, the sample count may be anywhere between 363 and 444 with the nominal value being 400, corresponding to a line frequency of 50 Hz. Fig. 3.7 shows how to compute the number of samples N in the rectified AC input voltage (the zero crossing points need to be monitored). Monitoring of zero crossing points demands more complexity in analog circuitry. Instead, the method used is to fix a minimum reference point for the input voltage, as shown in Fig. 3.8. A counter starts when the sampled value of input AC voltage from ADC rises above V_{MINREF}, and stops when the voltage falls below V_{MINREF} in the next cycle. The count value at that point would give the value of sample count N. Fig. 3.9 shows the program flow chart of average voltage computation in digital domain.

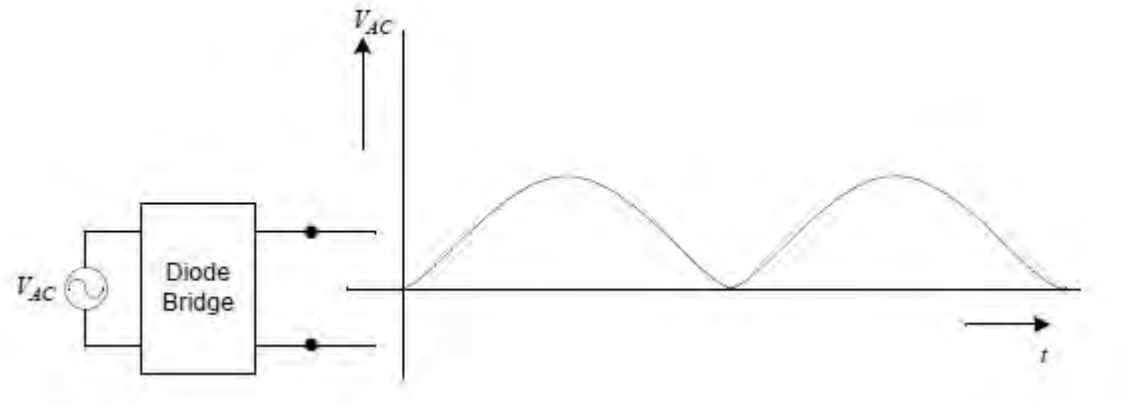


Fig 3.7 Rectified AC voltage

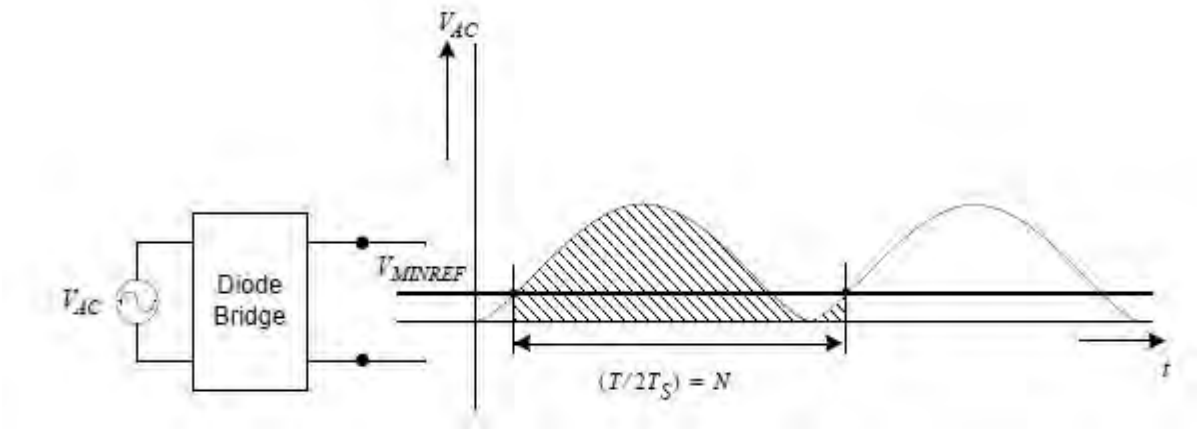


Fig 3.8 Calculation of average AC voltage

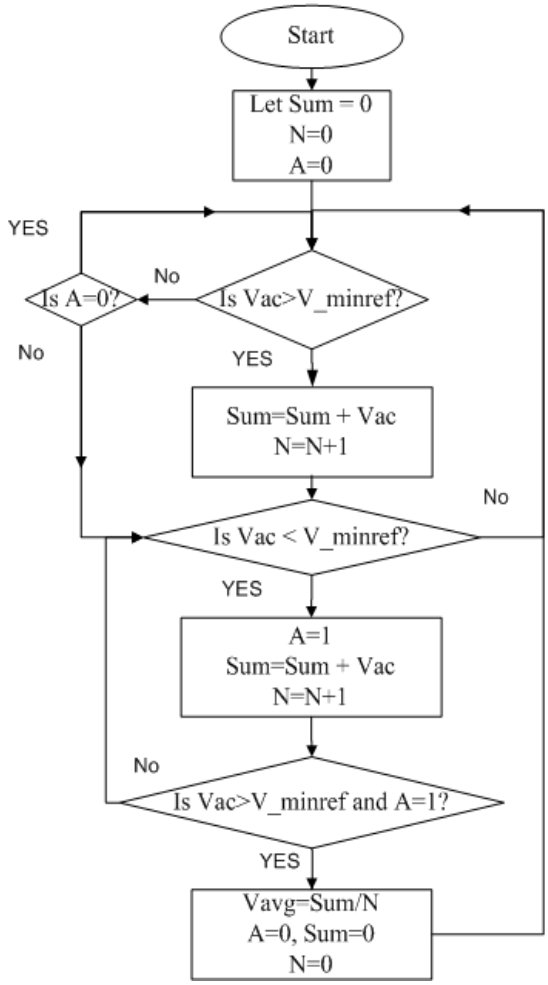


Fig 3.9 Program Flow chart of average voltage computation in digital domain

3.6 Analog to digital conversion

The analog to digital conversion is the most important operation in the Digital signal processing based implementation of proposed close loop Ĉuk active PFC Matlab simulink model. Since we need the information of input rectified AC voltage, output DC voltage and input rectified inductor current to generate PWM gate signal in MOSFET so analog to digital conversion is the foremost work that has to be done to fetch this analog signals in digital domain. Every embedded Digital signal processor has more than one built in ADC. The used digital controller dsPIC33FJ12MC202 device has up to 6 ADC module input channels. This ADC modules can be configured as either a 10-bit or a 12- bit ADC. Since our operation needs 3 ADC and simultaneous conversion features so 10-bit ADC mode is selected for its capability to fulfill our needs.

The 10-bit ADC configuration has the following key features:

• Successive Approximation (SAR) conversion
• Conversion speeds of up to 1.1 Msps
• Up to 6 analog input pins
• External voltage reference input pins
• Simultaneous sampling of up to four analog input pins
• Automatic Channel Scan mode
• Selectable conversion trigger source
• Selectable Buffer Fill modes
• Four result alignment options (signed/unsigned, fractional/integer)
• Operation during CPU Sleep and Idle modes
• 16-word conversion result buffer

A block diagram of the ADC is shown in Fig. 3.10 and ADC conversion clock period block diagram is shown in Fig. 3.11. To configure the ADC module we have to follow the following steps

1. Select port pins as analog inputs
2. Select voltage reference source to match expected range on analog inputs.
3. Select the analog conversion clock to match the desired data rate with the processor clock.
4. Determine how many sample-and-hold channels will be used.
5. Select the appropriate sample/conversion sequence.

6. Select the way conversion results are presented in the buffer.

7. Turn on the ADC module.

8. Configure ADC interrupt (if required).

9. Clear the ADxIF bit.

On the basis of the above steps and our needs the configuration of ADC Module is described through the table 3.1.

Table 3.1: ADC Module configuration

Configuring bits in different register	Function description
AD1CON1 = 0	Continue module operation in Idle mode ADC sample-and-hold amplifiers are holding
AD1CON1bits.FORM = 3	Signed Fractional Results
AD1CON1bits.AD12B = 0	Configured for 10 bit operation
AD1CON1bits.SSRC = 5	MC PWM2 module to trigger ADC
AD1CON1bits.SIMSAM = 1	Simultaneous Sampling enabled
AD1CON1bits.ASAM = 1	Auto Sampling enabled
AD1CON2 = 0	Converter Voltage Reference ADREF+ =AVDD and ADREF- =AVSS Clock derived from system clock Always starts filling buffer from the beginning Always uses channel input selects for Sample A
AD1CON2bits.CHPS = 3	Convert CH0, CH1, CH2 and CH3
AD1CON2bits.SMPI = 1	Interrupt on second sample/convert sequence
AD1CON2bits.CSCNA = 1	Channel Scanning Enabled
AD1CON3bits.SAMC = 8	Auto Sample time = 8×Tad
AD1CON3bits.ADCS = 4	AD Conversion time = 8×Tcy
AD1CHS0bits.CH0NA = 0	Channel CH0 negative reference is Vref
AD1CHS123bits.CH123NA = 0	Channels CH1, CH2 and CH3 negative reference is Vref

AD1CHS123bits.CH123SA = 1	Convert AN3, AN4 and AN5 on CH1, CH2 and CH3 positive reference respectively
AD1PCFGL = 0xFFFF	Port pin in Digital mode, port read input enabled, ADC input multiplexer connected to AVSS
AD1PCFGLbits.PCFG2 = 0	AN2 pin in analog mode used for output DC voltage
AD1PCFGLbits.PCFG3 = 0	AN3 pin in analog mode used for input rectified DC inductor current
AD1PCFGLbits.PCFG4 = 0	AN4 pin in analog mode used for input rectified AC Voltage
AD1PCFGLbits.PCFG5 = 0	Buffer5 is activated
AD1CSSL = 0	Skip ANx (x= 0 to 5) for input scan
AD1CSSLbits.CSS2 = 1 AD1CSSLbits.CSS4 = 1	Select AN2 for input scan Select AN4 for input scan The ADC module is used in Channel Scanning mode. The two voltages, output DC voltage (AN2) and input rectified AC Voltage (AN4) are sampled and converted on alternate triggers, while the input rectified DC inductor current (AN3) is sampled and converted on every trigger.
AD1CON1bits.ADON = 1	Turn-ON the ADC Module

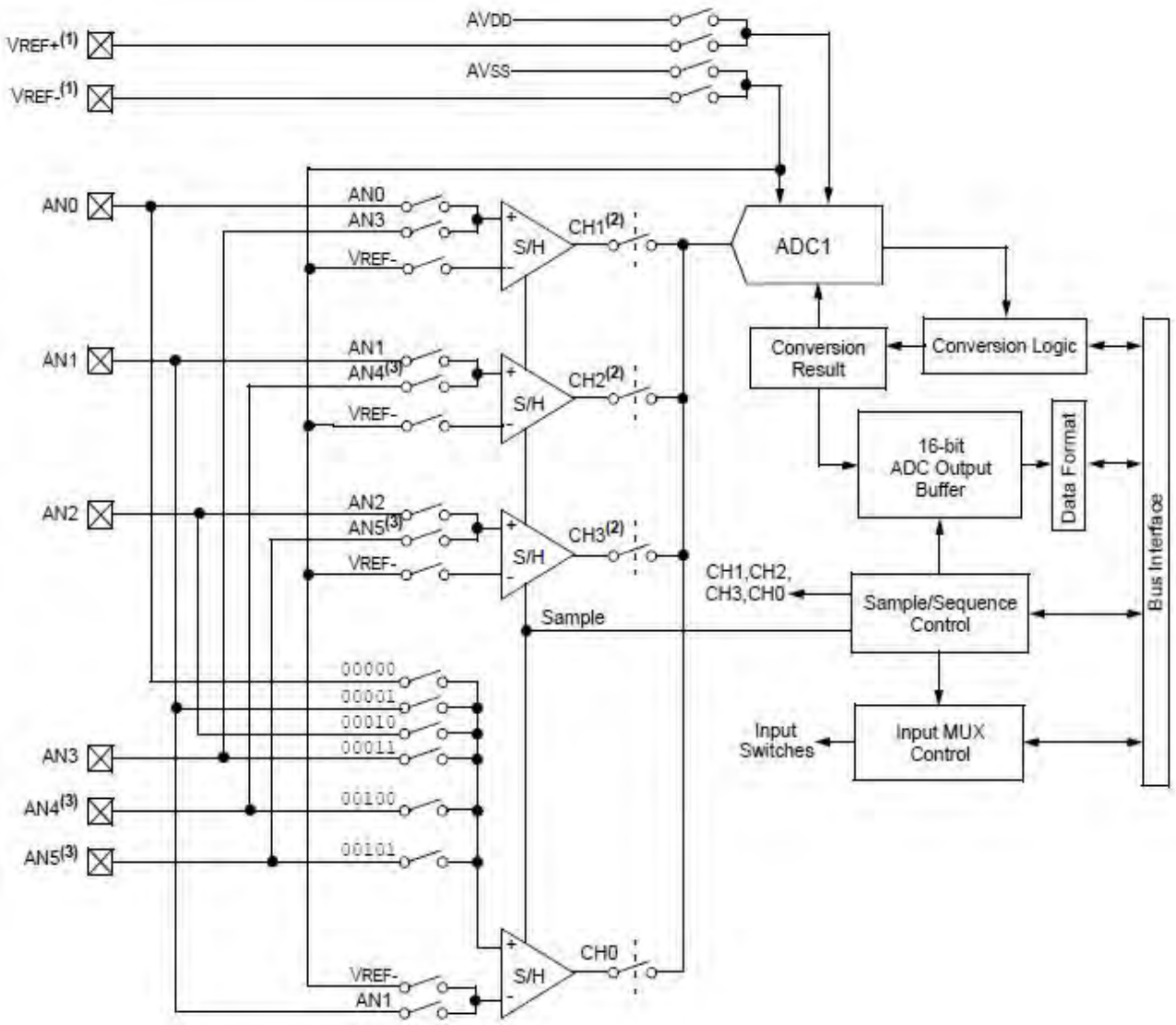


Fig 3.10 ADC Module block diagram [42]

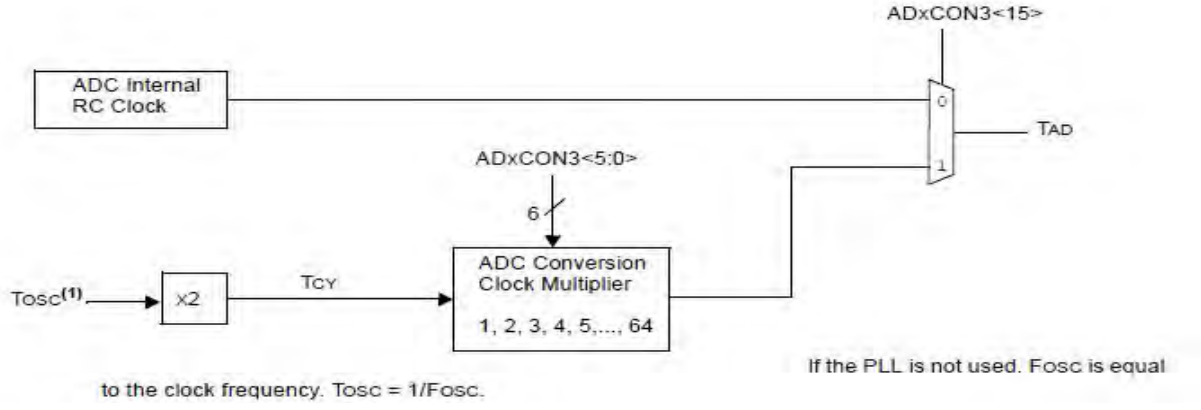


Fig 3.11 ADC conversion clock period block diagram [42]

3.7 PWM signal generation

The only output from the Digital signal controller device is firing pulses to the Cuk converter switch to control the output DC voltage in addition to presenting a resistive load to the AC line.

Every Digital signal processor has the PWM signal generation feature. The used dsPIC33FJ12MC202 device supports up to two dedicated Pulse Width Modulation (PWM) modules. The PWM1 module is a 6-channel PWM generator, and the PWM2 module is a 2-channel PWM generator. 2-Channel PWM module block diagram (PWM2) is shown in Fig 3.12 and Edge Aligned Block diagram is shown in Fig 3.13.

The PWM module has the following features:

• Up to 16-bit resolution

• On-the-fly PWM frequency changes

• Edge and Center-Aligned Output modes

• Single Pulse Generation mode

• Interrupt support for asymmetrical updates in Center-Aligned mode

• Output overrides control for Electrically Commutative Motor (ECM) operation or BLDC

• Special Event comparator for scheduling other peripheral events

• Fault pins to optionally drive each of the PWM output pins to a defined state

• Duty cycle updates configurable to be immediate or synchronized to the PWM time base

In our application PWM2 module has been used to generate PWM signal and its configuration is described through the table 3.2.

Table 3.2 PWM2 module configuration

Configuring bits in different register	Function description
P2TCON = 0	PWM time base operates in a Free-Running mode PWM time base input clock period is TCY (1:1 prescale) and 1:1 postscale [Tcy = 1/Fcy and Fcy= Fosc/2] Fosc = Fin×M/(N1×N2) PWM time base runs in CPU Idle mode
P2TMR = 0	PWM time base is counting up Edge-Aligned mode
P2TPER = PWM period	The period value corresponds to a frequency

P2SECMPbits.SEVTCMP = PWM period	Special Event Trigger for ADC is at PWM Timer reset
PWM2CON1bits.PMOD1 = 1	Independent output Mode of PWM operation
PWM2CON1bits.PEN1H = 1	PWM output pin is enabled for PWM output
PWM2CON2 = 0	Updates from Duty Cycle and Period Buffer registers are enabled Output overrides via the P2OVDCON register occur on next TCY boundary
PWM2CON2bits.SEVOPS = 0	Trigger ADC on every PWM cycle
PWM2CON2bits.IUE = 1	Immediate updates of duty cycle is enabled
P2OVDCONbits.POVD1H = 1	PWM Output pin is controlled by PWM Generator
P2DC1 = 0	PWM duty cycle register value
IFS4bits.PWM2IF = 0	Clear the PWM2 interrupt flag (Interrupt request has not occurred)
IEC4bits.PWM2IE = 0	Disable PWM2 interrupts (Interrupt request not enabled)

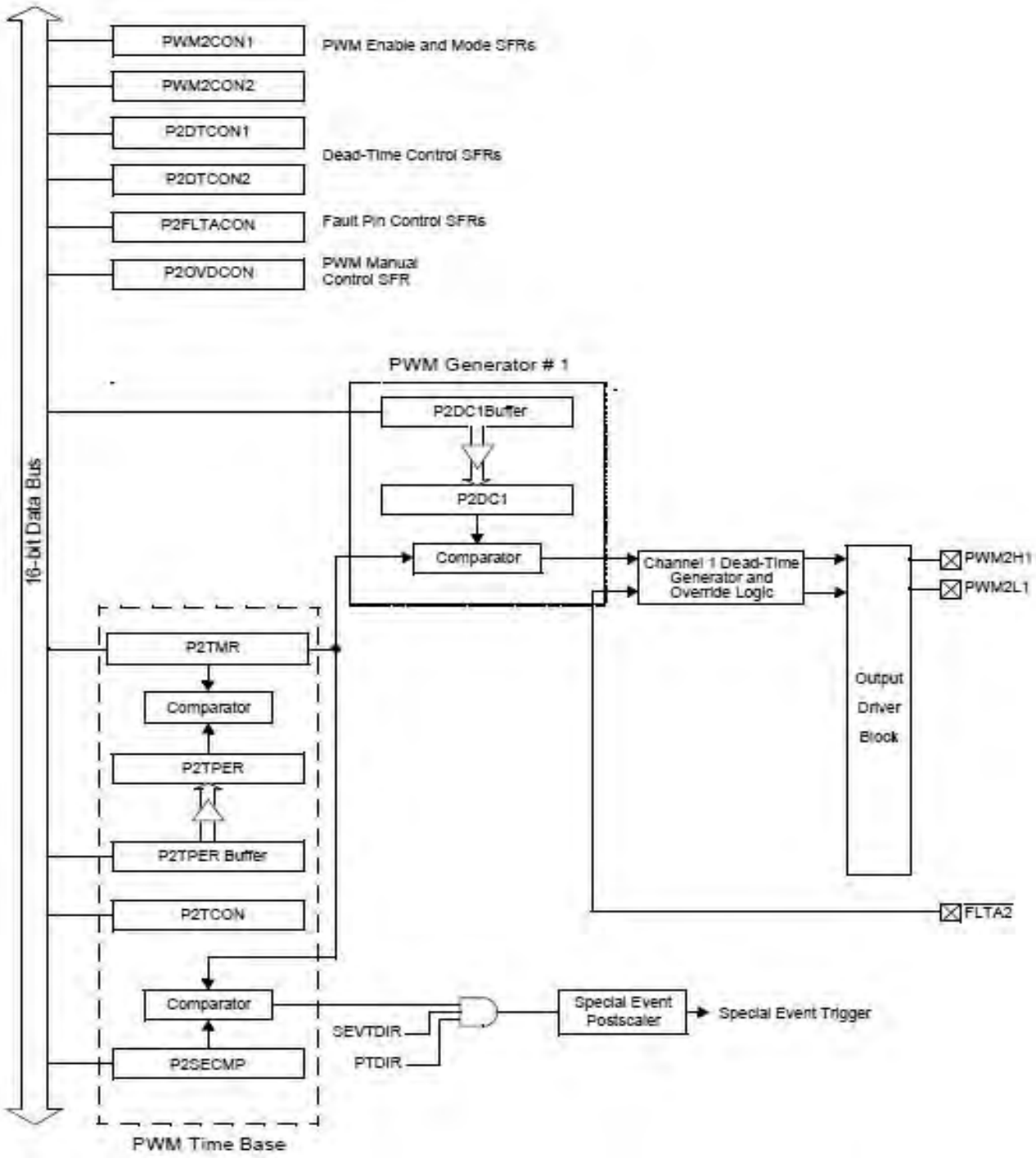


Fig 3.12 2-Channel PWM Module block diagram (PWM2) [42]

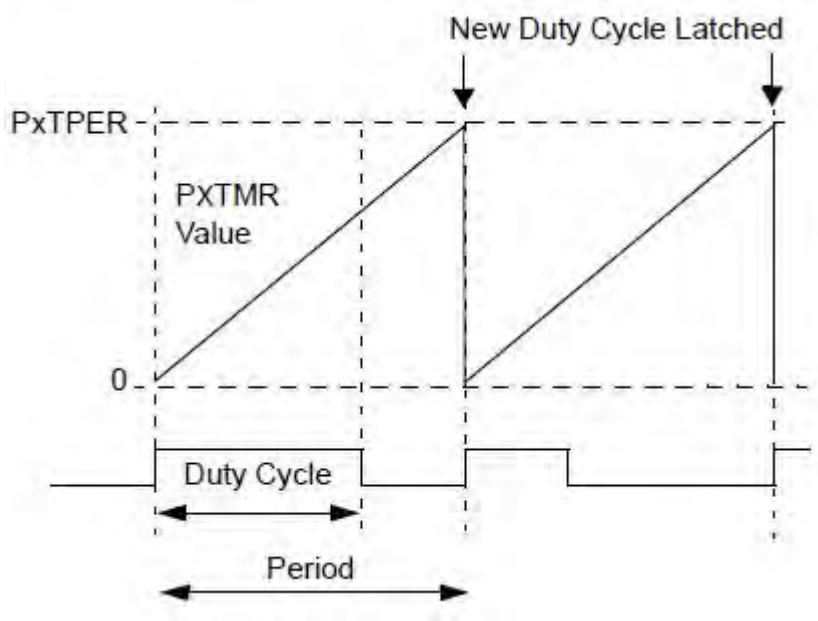


Fig. 3.13 Edge aligned PWM [42]

PWM period is calculated by the following equation

TPWM = TCY × (PXTPER + 1) × (PXTMR Prescale Value)

3.8 Digital implementation of proportional integral control

The expression of proportional integral control in continuous time domain is defined by the following equation

$C(t) = K_P \times E(t) + K_I \int_0^t E(t)dt$ ..... (3.5)

Where

C(t) corresponds to the output signal of the PI controller

E(t) =Reference – Actual(t) corresponds to the input error signal of the PI controller

K_P corresponds to the proportional factor

K_I corresponds to the integral constant

And the discrete form of PI expression

$C(n) \approx K_P \times E(n) + K_I T_S \sum_0^N E(n)$ ..... (3.6)

$C(n) \approx K_P \times E(n) + \frac{T_S}{T_I} \sum_0^N E(n)$ ..... (3.7)

$C(n) \approx K \times E(n) + \frac{1}{T_I} \sum_0^N E(n)$ ..... (3.8)

Where,

K_P = K and $K_I = \frac{K}{T_I}$

T_S is the sample time

T_I is the integral time constant

E(n)= Reference-Actual(n) input error at n^{th} sample

The program flow chart of PI control is given by Fig. 3.14

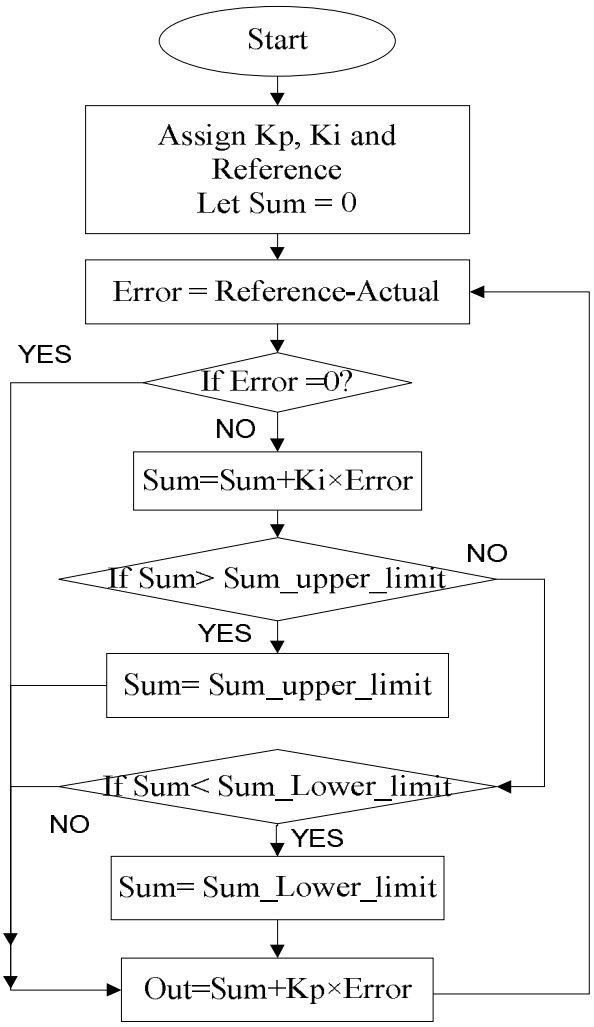


Fig 3.14 Program Flow chart of PI control

3.9 Timing logic for the software implementation

PWM pulse runs at a frequency of 80 kHz while supplying a trigger to the ADC every period (80 kHz). An ADC interrupt occurs every 2 timer periods (40 kHz). The ADC module is used in Channel Scanning mode. The two voltages, output DC voltage and input rectified AC voltage are sampled and converted on alternate triggers, while the input rectified DC current is sampled and converted on every trigger. The following analog channels and buffers are used on the dsPIC33FJ12MC202 device:

• The first ADC interrupt is generated after converting channels:

- AN2 - output DC voltage (ADCBUF2)
- AN3 - input rectified DC current (ADCBUF3)

• The second ADC interrupt is generated after converting channels:

- AN4 - input rectified AC voltage (ADCBUF4)
- AN3 - input rectified DC current (ADCBUF5)

Analog inputs AN2 and AN4 are in channel scanning alternately.

• At any point in the control loop reading from:

- ADCBUF2 gives the output DC voltage
- ADCBUF3 and ADCBUF5 give the input rectified DC current
- ADCBUF4 gives the input rectified AC voltage

PWM2 module generates the PWM pulses to drive the gate of the MOSFET. These events, along with the A/D interrupt generation sequence, are shown in the timing logic in Figure 3.15

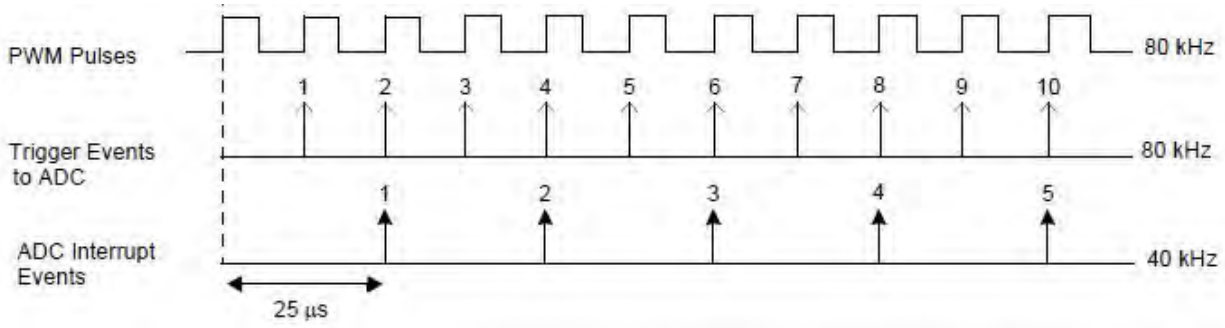


Fig 3.15 Timing logic [42]

3.10 Program Flowchart of the system

The main program flow is shown in Fig 3.16. After a Reset, when the program is executed, all the variables are initialized and peripherals are configured. The PI parameter values are defined for the control loop compensators. The PWM module is switched ON to operate at a frequency of 80 kHz, and all the interrupts are enabled. The ADC module waits for a PWM module special event interrupt. On every period match, the PWM generates a trigger to the ADC to start sampling the signals and converting them. On a timer trigger, the ADC samples and converts the voltages and currents and later generates an ADC interrupt. The power factor correction routines run inside the ADC Interrupt Service Routine (ISR). A power-on delay is allowed for the capacitors to charge to the DC bus voltage. After the power-on delay time (approximately 100 ms) completion, the control loops begin to execute. During the process of power-on delay, the voltage samples are accumulated and frequency is calculated. This enables the average voltage calculation to be done in the first iteration of the control loop itself, as the average voltage is already available for a period corresponding to one line voltage cycle. The voltage error compensators execute the voltage PI controllers having the measured value of output DC voltage. The average value of input voltage, squaring and dividing routines, execute in sequence from the measured value of input AC voltage thereby giving the voltage feed-forward compensator output. This output is used in conjunction with the voltage error compensator output to calculate the reference value of I_ac_ref. Having I_ac_ref and the measured value of the inductor current, the current error compensator executes the current PI controllers to produce the new duty cycle for the PWM pulse.

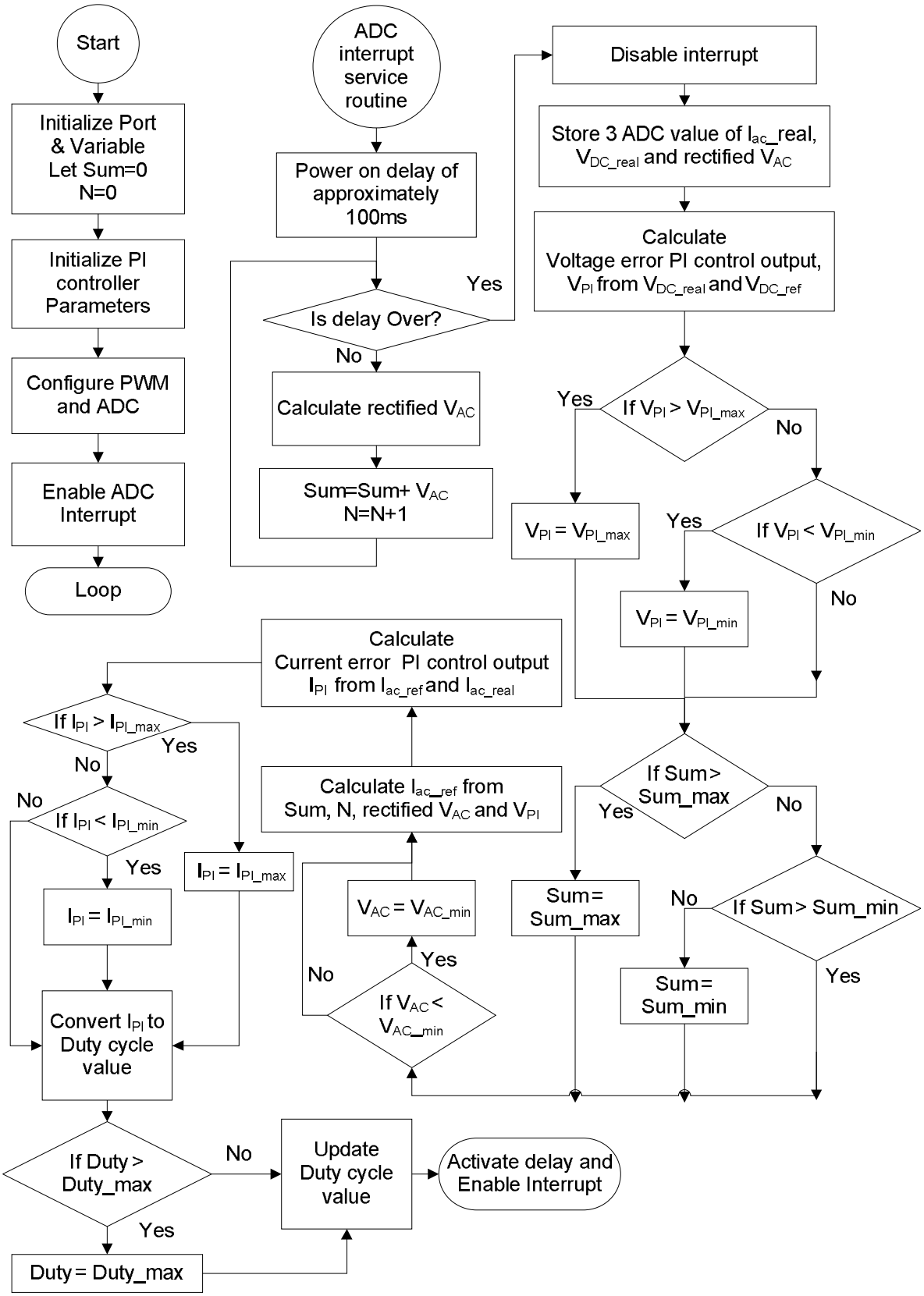


Fig 3.16 Program Flow chart of the system

3.11 Simulation

Fig. 3.17 to Fig. 3.32 show the input rectified voltage/current, output voltage and FFT component of input current at different load resistance and -400V reference output voltage under 325V (peak) input voltage and 50Hz supply frequency using proposed close loop Ĉuk rectifier in Proteus software.

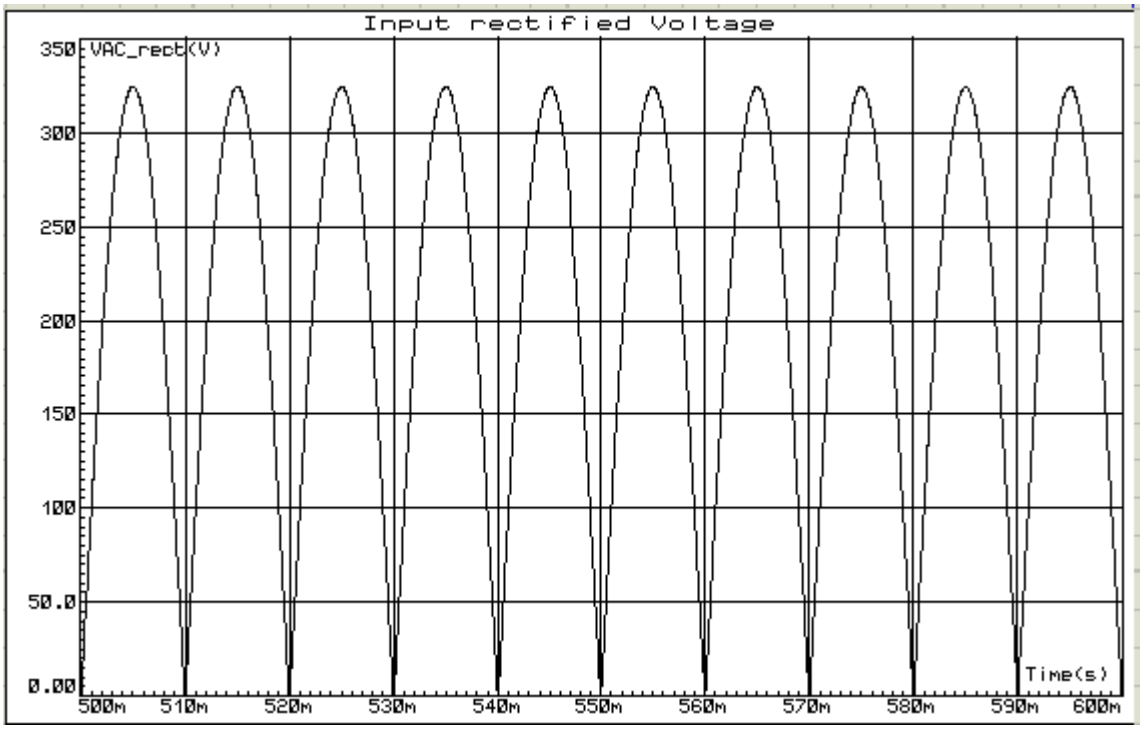


Fig 3.17 Input rectified voltage 325V (peak)

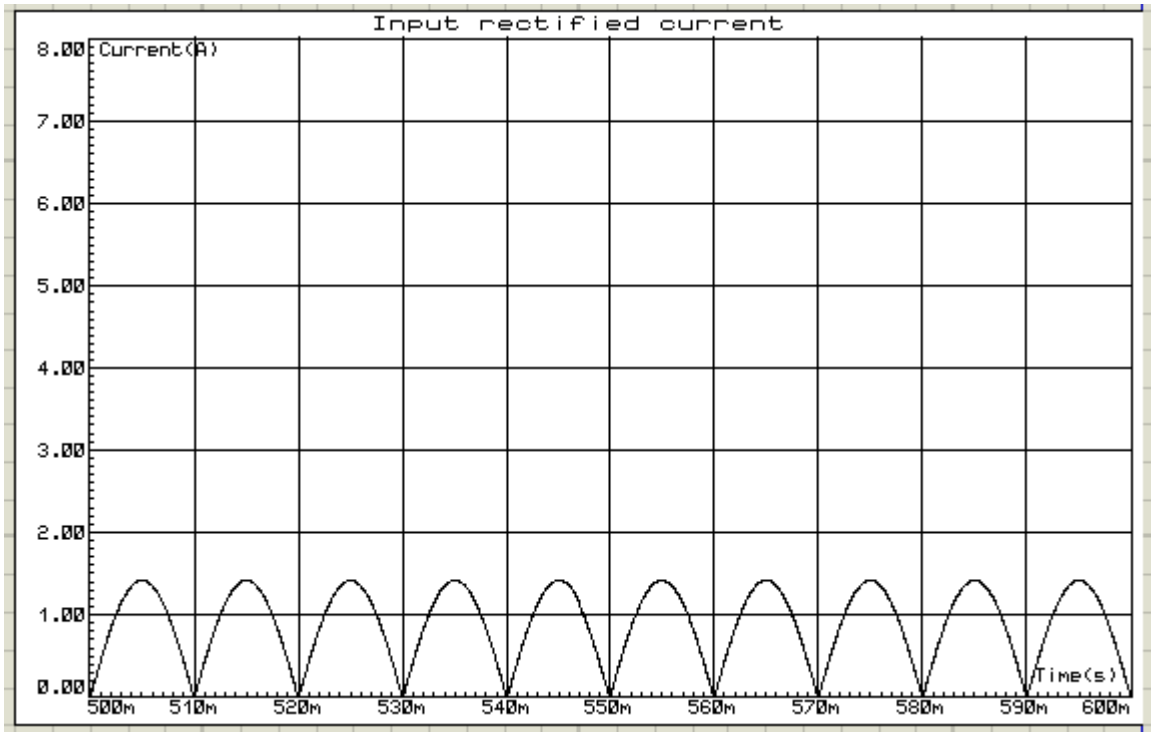


Fig 3.18 Input rectified current for 800Ω load resistance and 325V (peak) input voltage

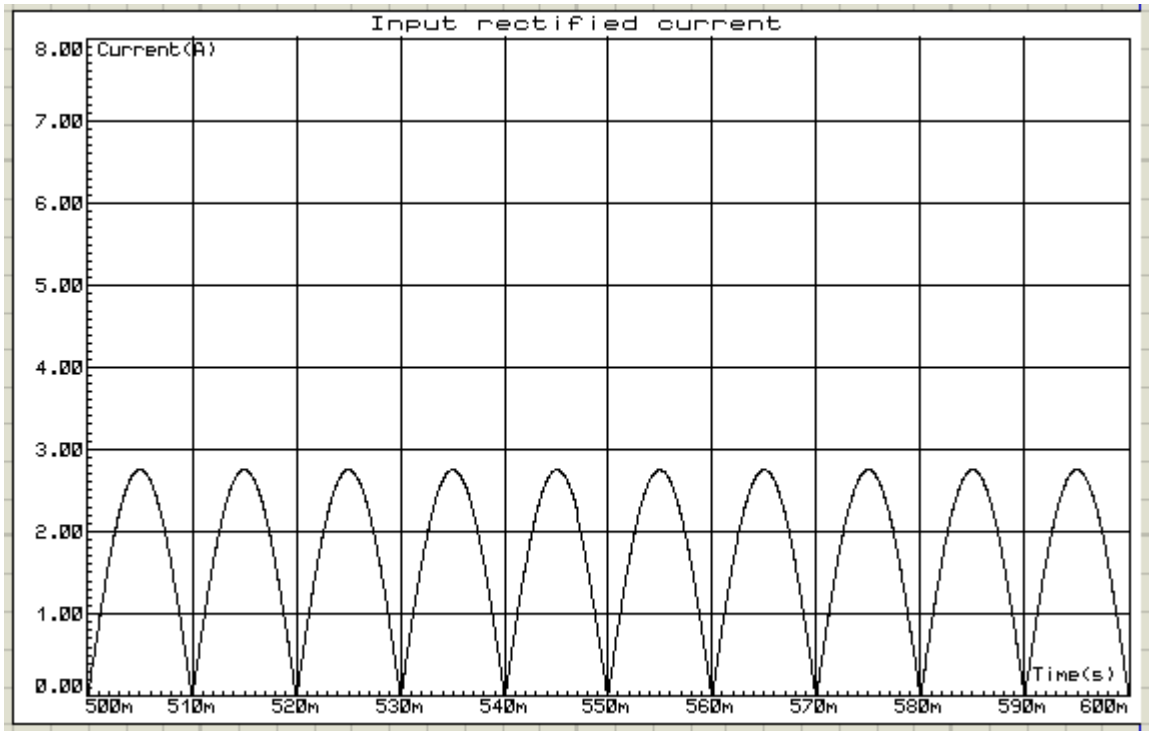


Fig 3.19 Input rectified current for 400Ω load resistance and 325V (peak) input voltage

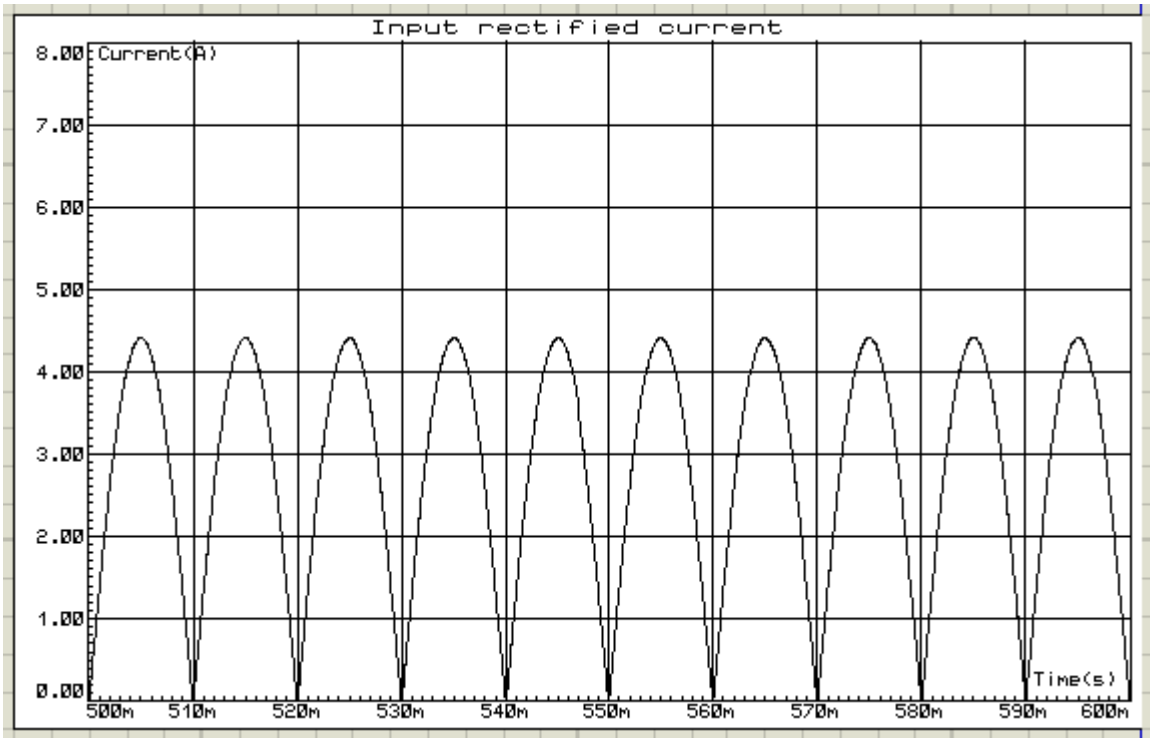


Fig 3.20 Input rectified current for 267Ω load resistance and 325V (peak) input voltage

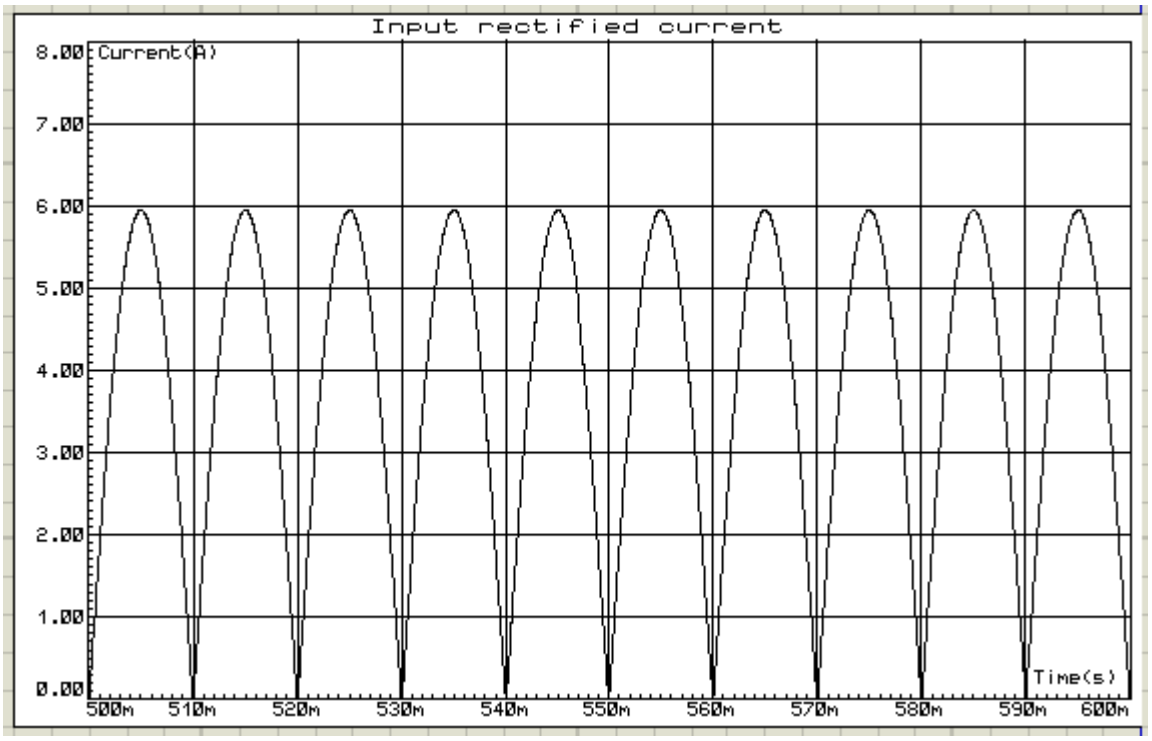


Fig 3.21 Input rectified current for 200Ω load resistance and 325V (peak) input voltage

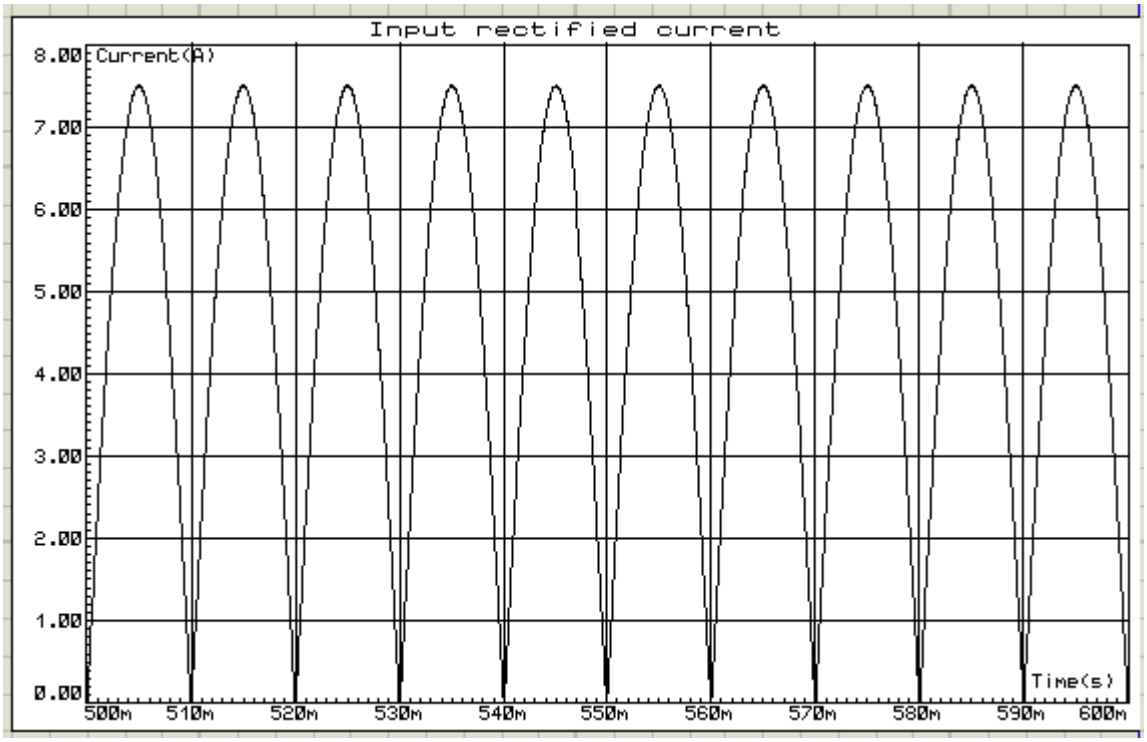


Fig 3.22 Input rectified current for 160Ω load resistance and 325V (peak) input voltage

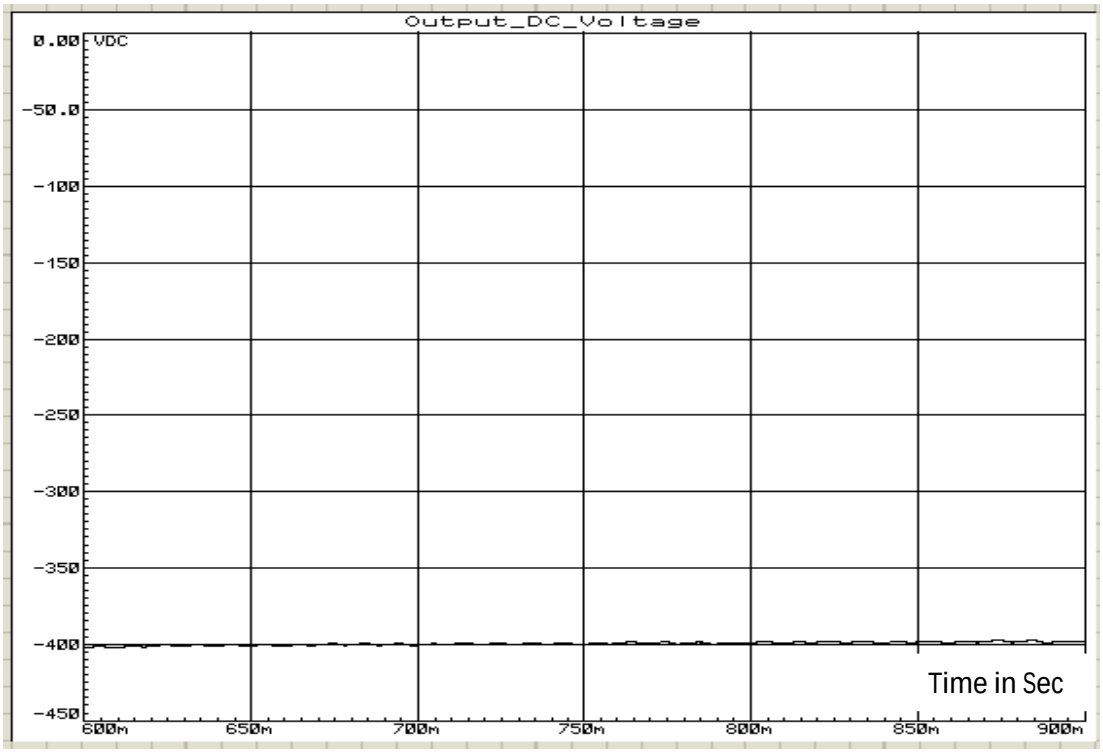


Fig 3.23 Output voltage for 800Ω load resistance and 325V (peak) input voltage

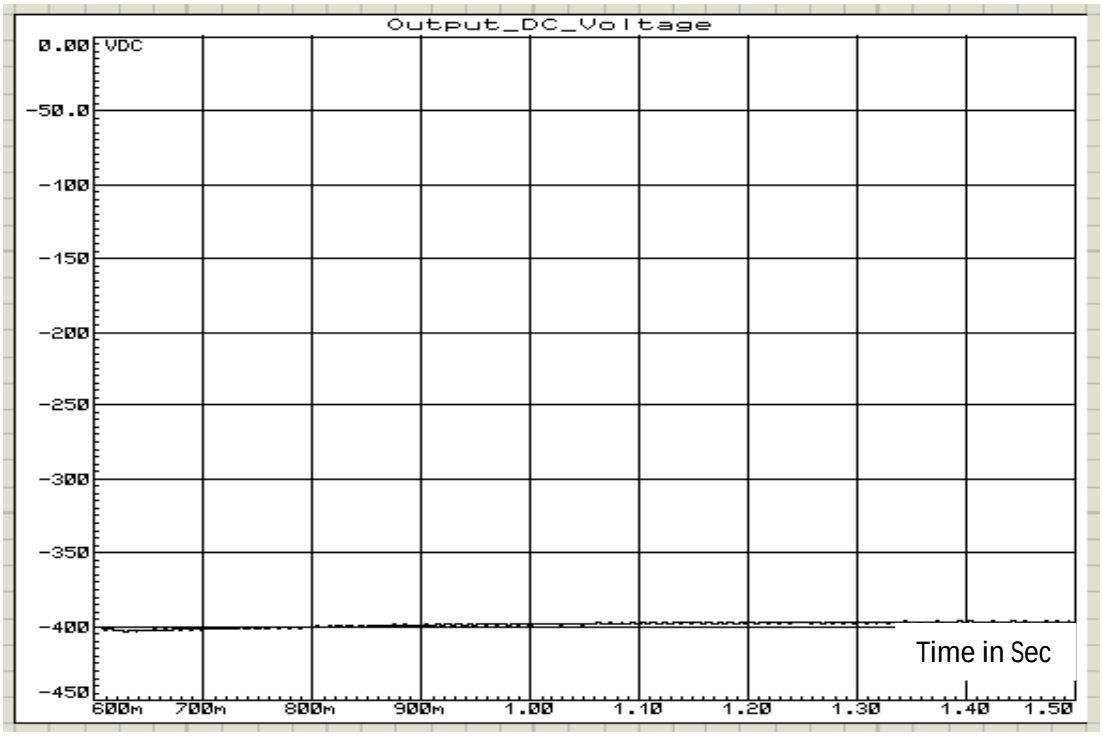


Fig 3.24 Output voltage for 400Ω load resistance and 325V (peak) input voltage

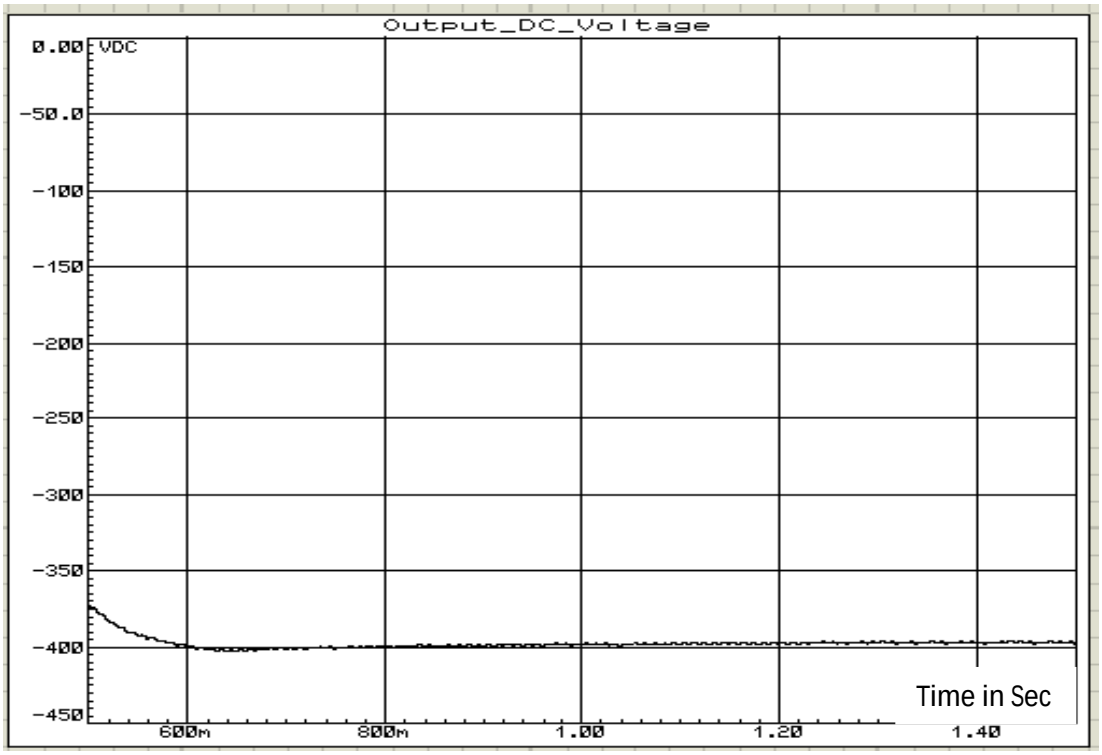


Fig 3.25 Output voltage for 267Ω load resistance and 325V (peak) input voltage

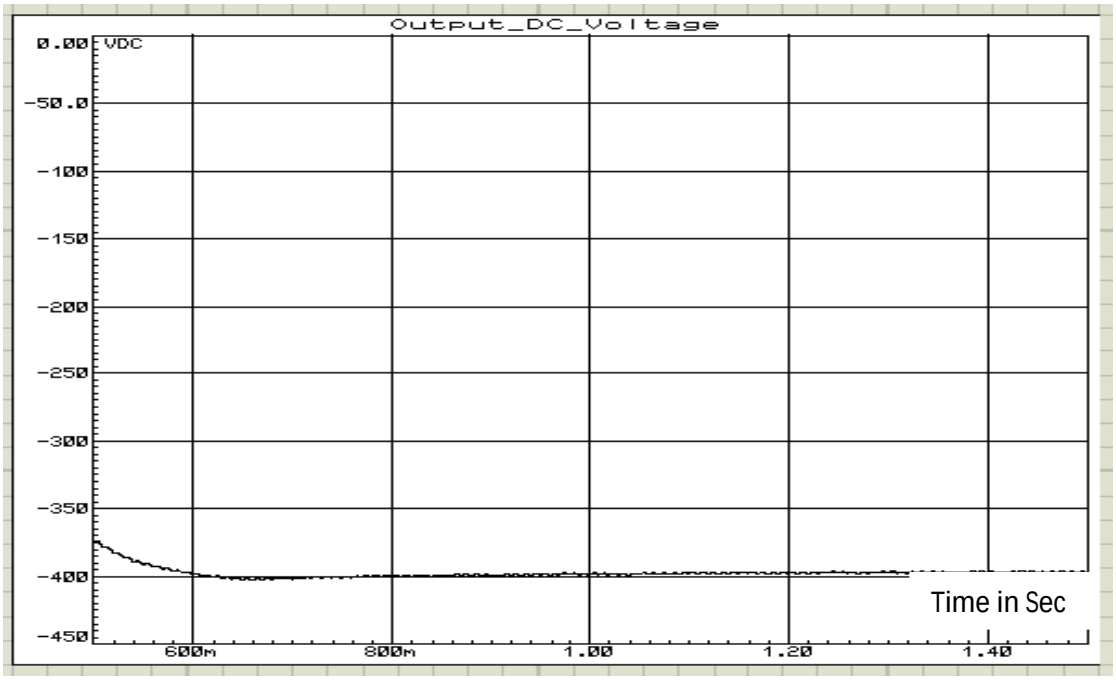


Fig 3.26 Output voltage for 200Ω load resistance and 325V (peak) input voltage

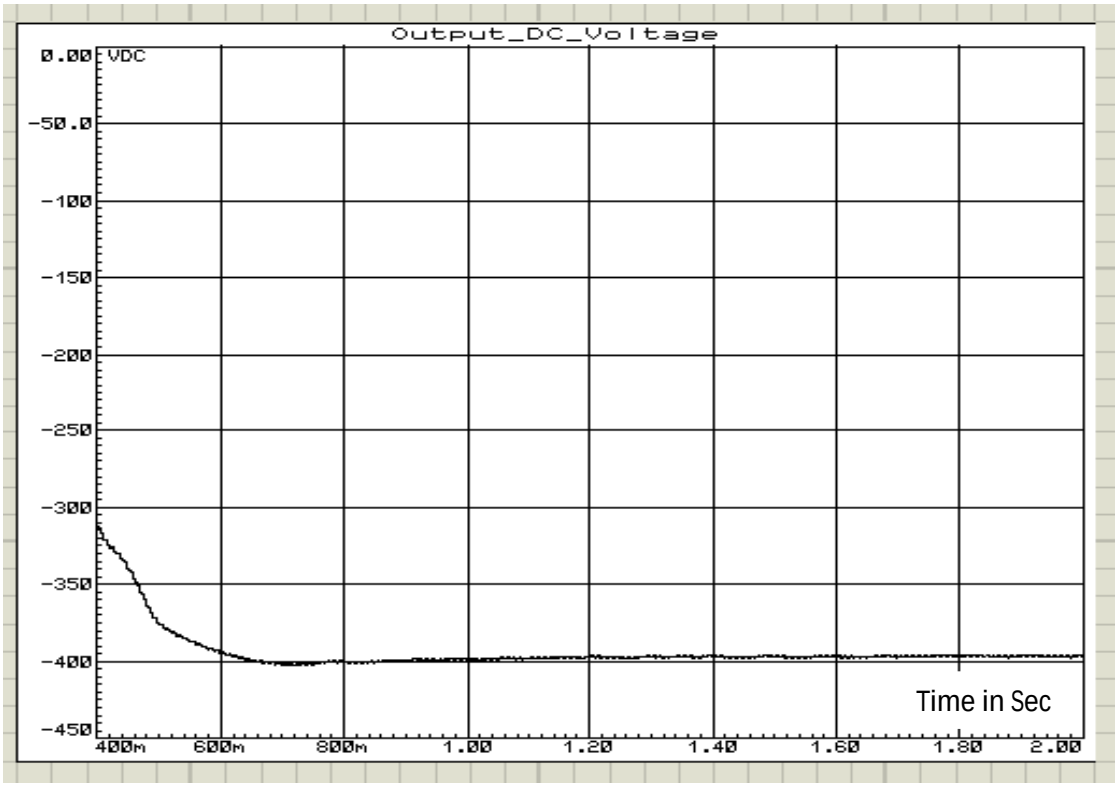


Fig 3.27 Output voltage for 160Ω load resistance and 325V (peak) input voltage

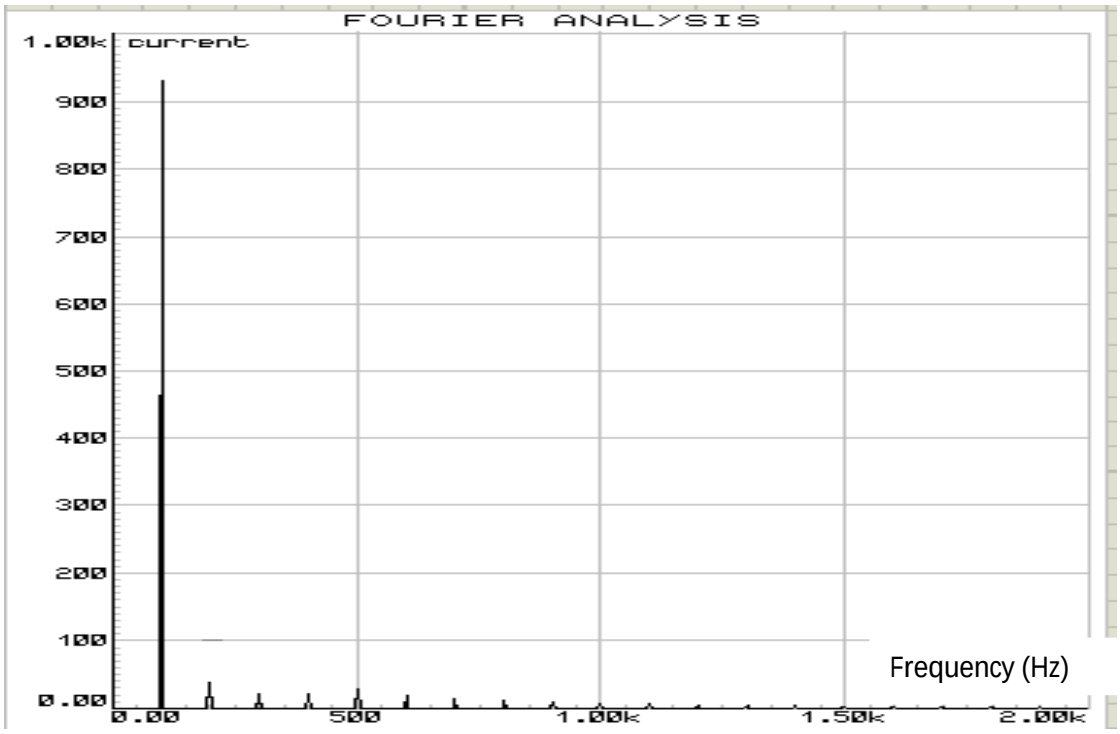


Fig 3.28 FFT component of input current for 800Ω load resistance and 325V (peak) input voltage

Fig 3.29 FFT component of input current for 400Ω load resistance and 325V (peak) input voltage

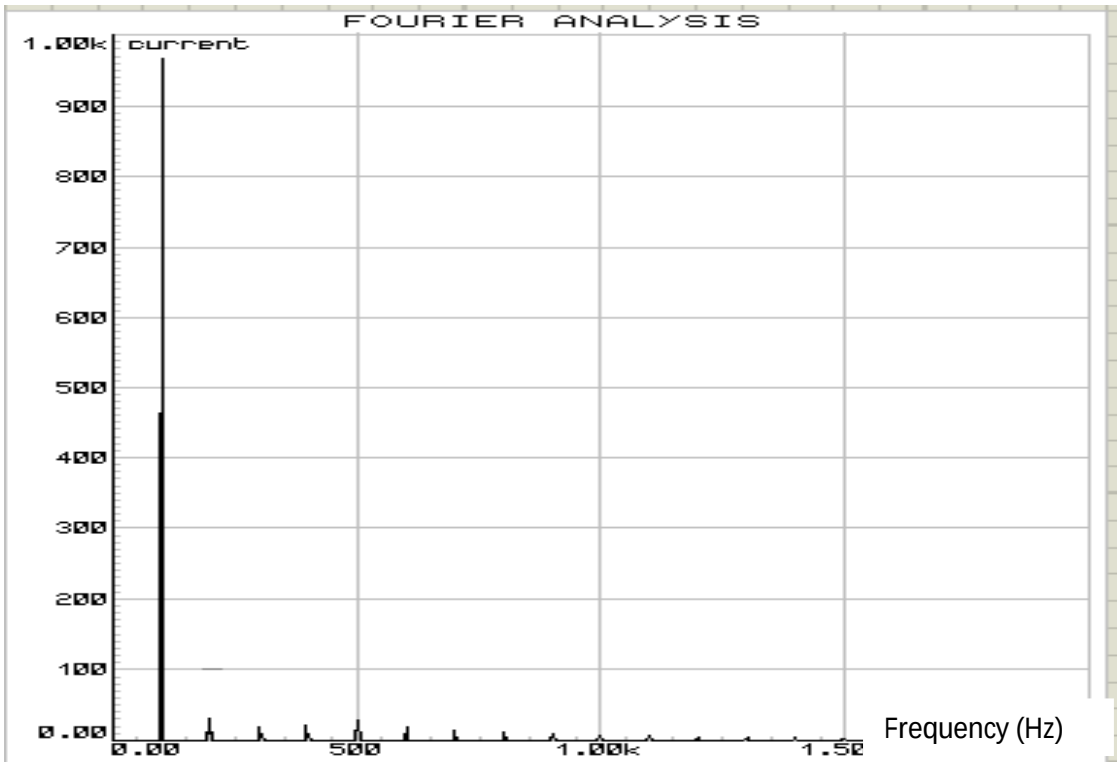


Fig 3.30 FFT component of input current for 267Ω load resistance and 325V (peak) input voltage

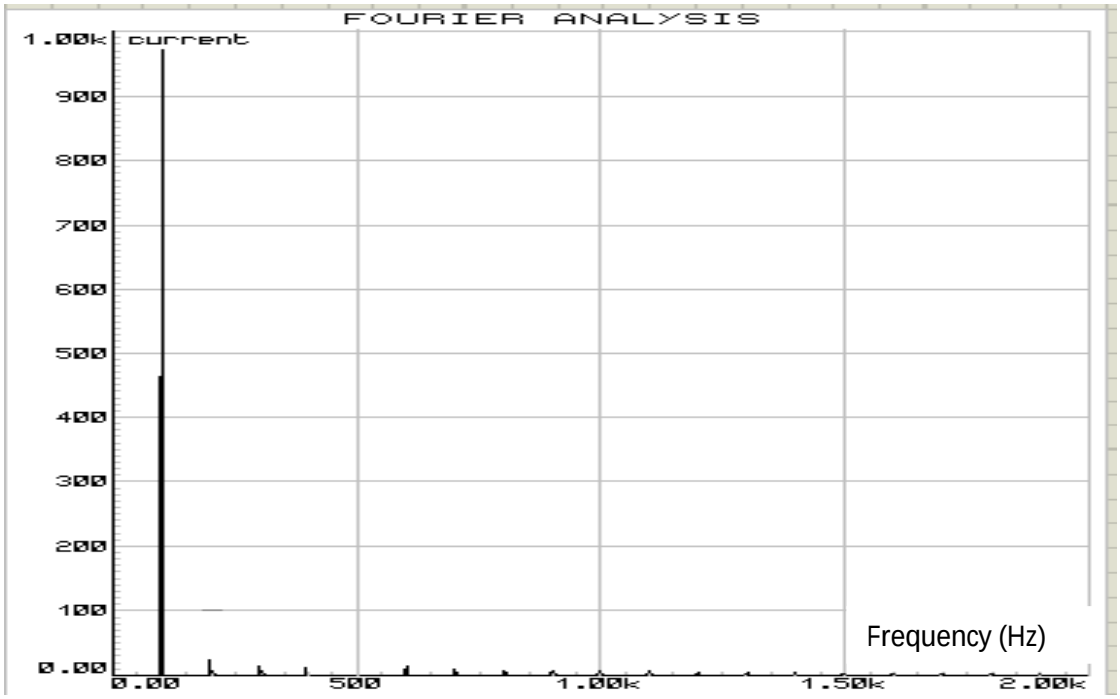


Fig 3.31 FFT component of input current for 200Ω load resistance and 325V (peak) input voltage

Fig 3.32 FFT component of input current for 160Ω load resistance and 325V (peak) input voltage

Table 3.3 shows the summarize data that is obtained from Fig. 3.17 to Fig. 3.32.

Calculation of input THD, input PF and efficiency is carried out using following equation

Input THD = $\frac{1}{I_1}\sqrt{{I_2}^2 + {I_3}^2 + \cdots + {I_n}^2}$ ………………………………………………… (3.9)

Here,

I_1 is the rms value of fundamental component of input current

I_2, I_3 … … … I_n are the harmonics component of input current

Input PF = Displacement Factor × Distortion Factor = $\frac{Cos\emptyset}{\sqrt{1+THD^2}}$

$\emptyset$ = angle between the fundamental component of input voltage and input current

Efficiency = $\frac{\frac{output\ voltage^2}{Load\ resistance}}{input\ rms\ voltage \times input\ rms\ current}$

Table 3.3 Data for 800Ω to 160Ω load resistance at 325V (peak) input voltage, 50Hz supply frequency and -400V output voltage

Resistive load in Ohm	Input rms current(A)	Input rms Voltage(V)	Output -DC Voltage(V)	Input THD in %	Input PF	Efficiency In %
800	1.005	230	400	7.9	0.997	86.5
400	1.95	230	399	7.4	0.9972	88.7
267	3.13	230	398	6.12	0.998	82.4
200	4.213	230	399	4.4	0.999	82.15
160	5.305	230	398	4	0.9992	81.14

Fig. 3.33 to Fig. 3.36 show the efficiency, output voltage, input PF and input THD respectively at different load resistances under the supply frequency 50Hz and input voltage 325V(peak).

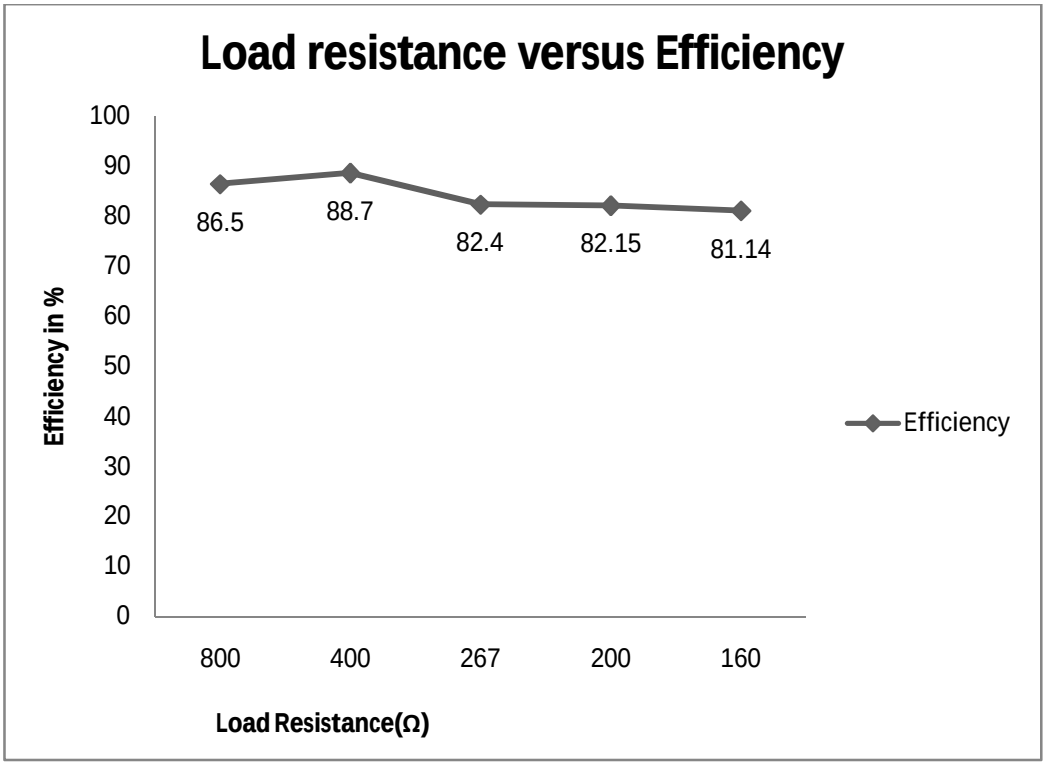


Fig 3.33 Efficiency at different loads and 325V (peak) input voltage

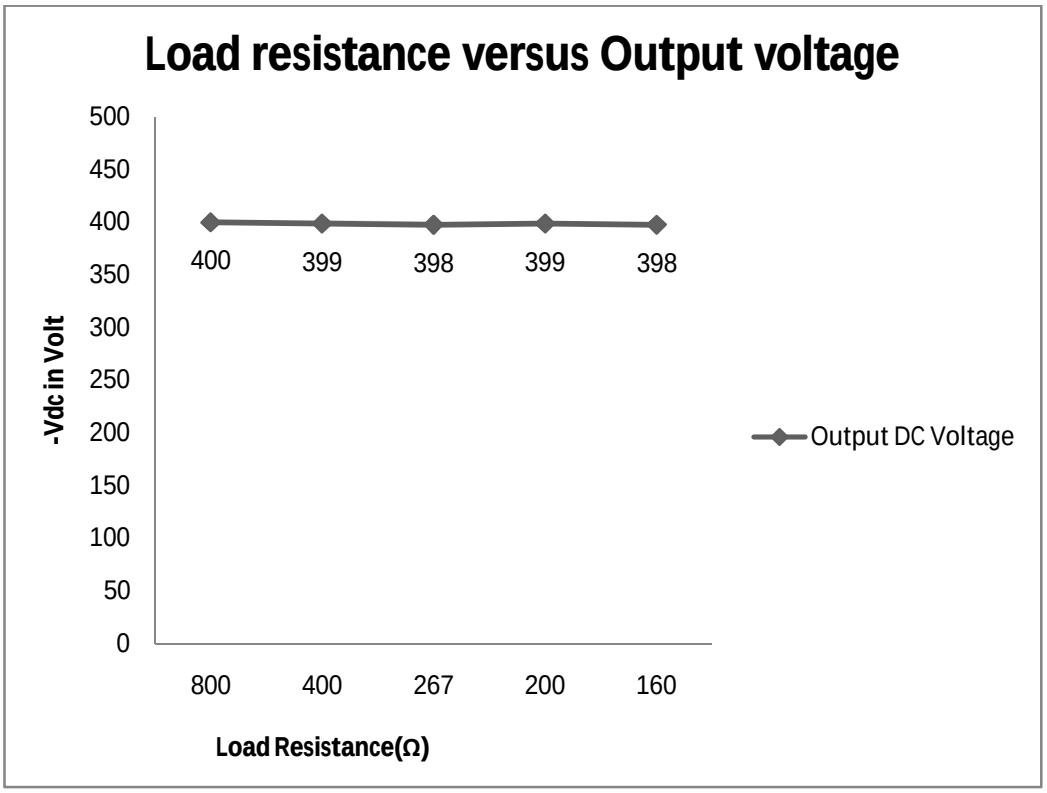


Fig 3.34 Output voltage at different loads and 325V (peak) input voltage

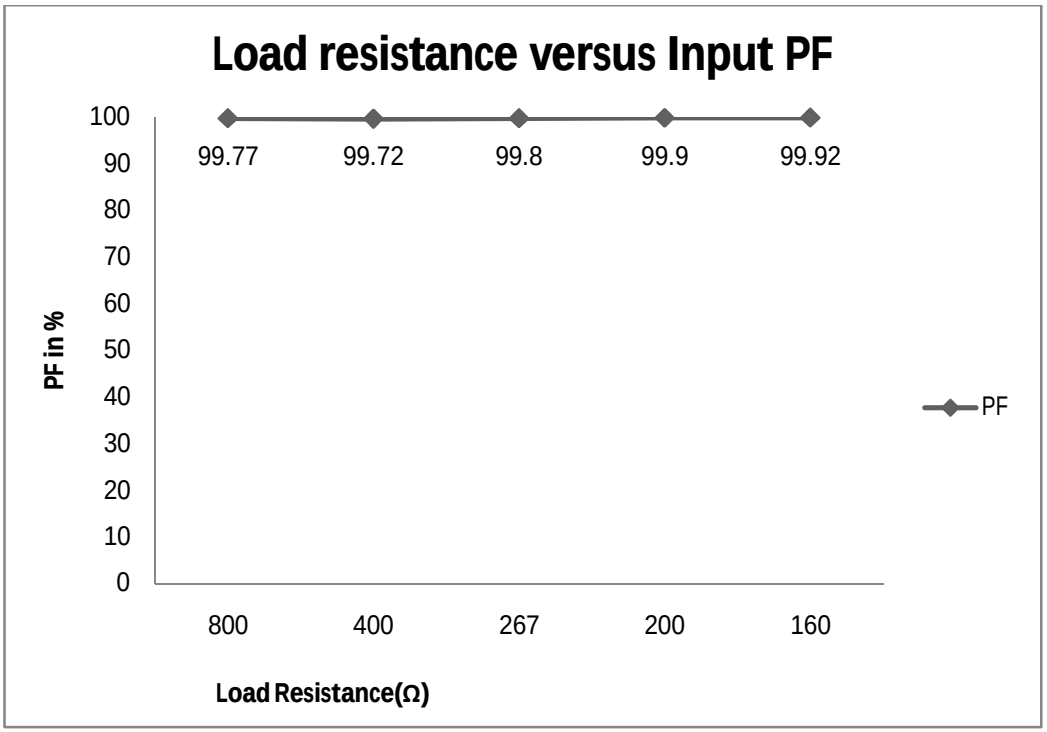


Fig 3.35 Input PF at different loads and 325V (peak) input voltage

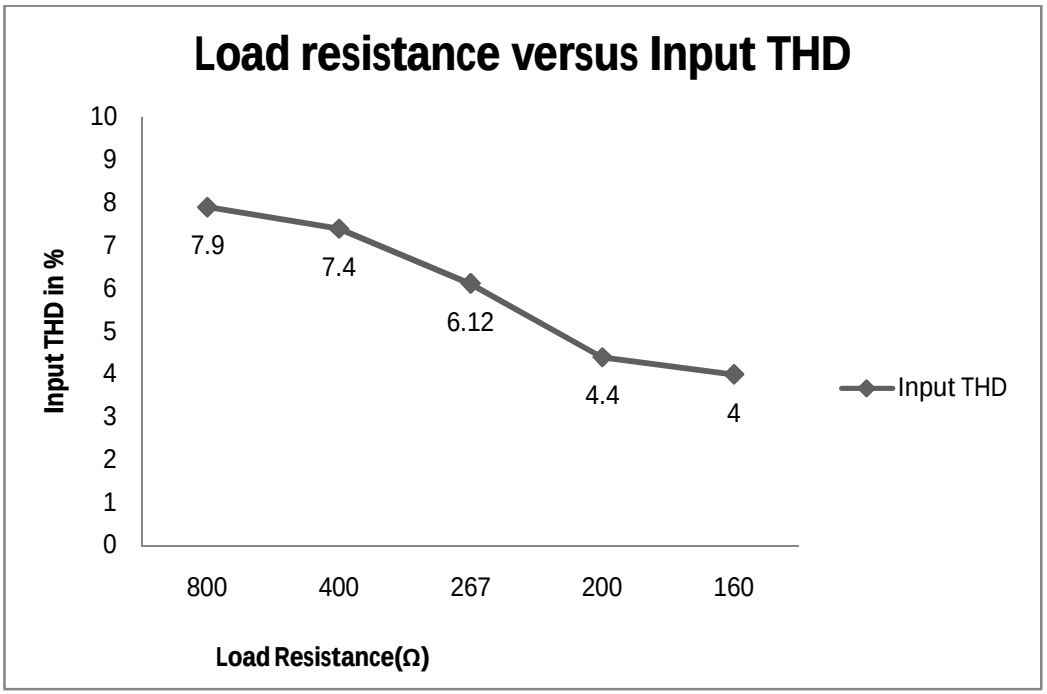


Fig 3.36 Input THD at different loads and 325V (peak) input voltage

Fig. 3.37 to Fig. 3.43 show the input rectified voltage/current, output voltage and FFT component of input current at different load resistance and -400V reference output voltage under 200V (rms) input voltage and 50Hz supply frequency using proposed close loop Ĉuk rectifier in Proteus software.

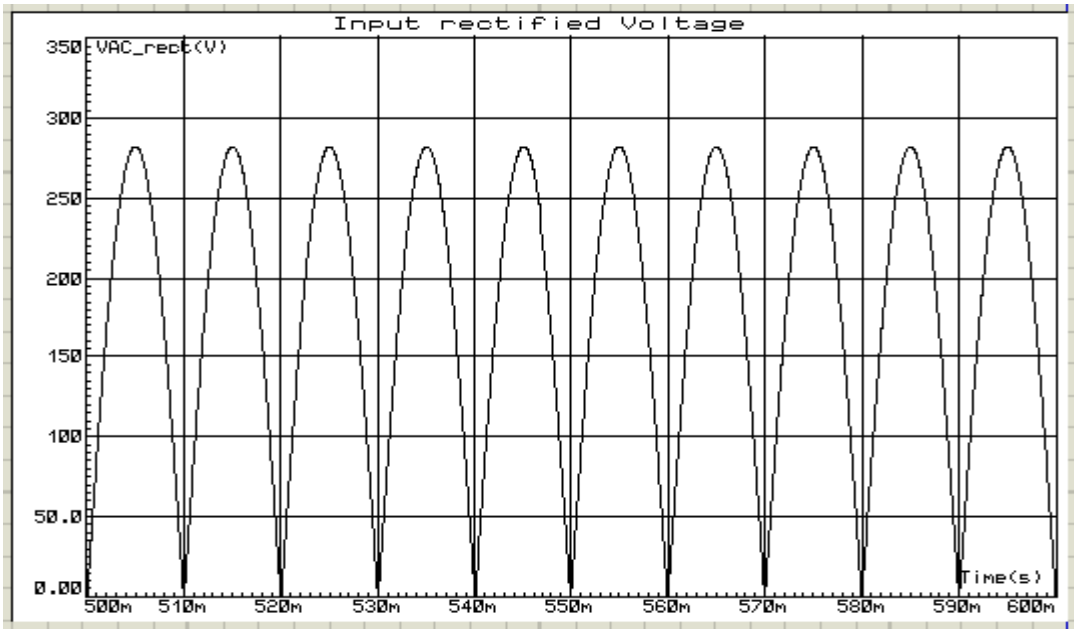


Fig 3.37 Input rectified voltage 283V (peak)

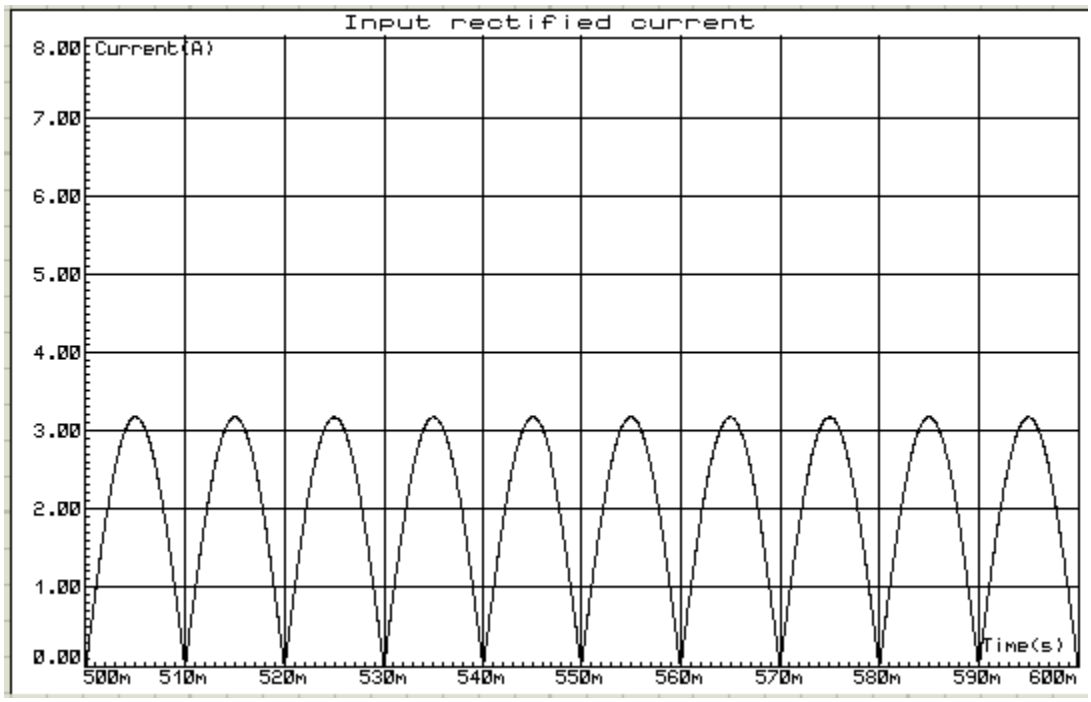


Fig 3.38 Input rectified current for 400Ω load resistance and 283V (peak) input voltage

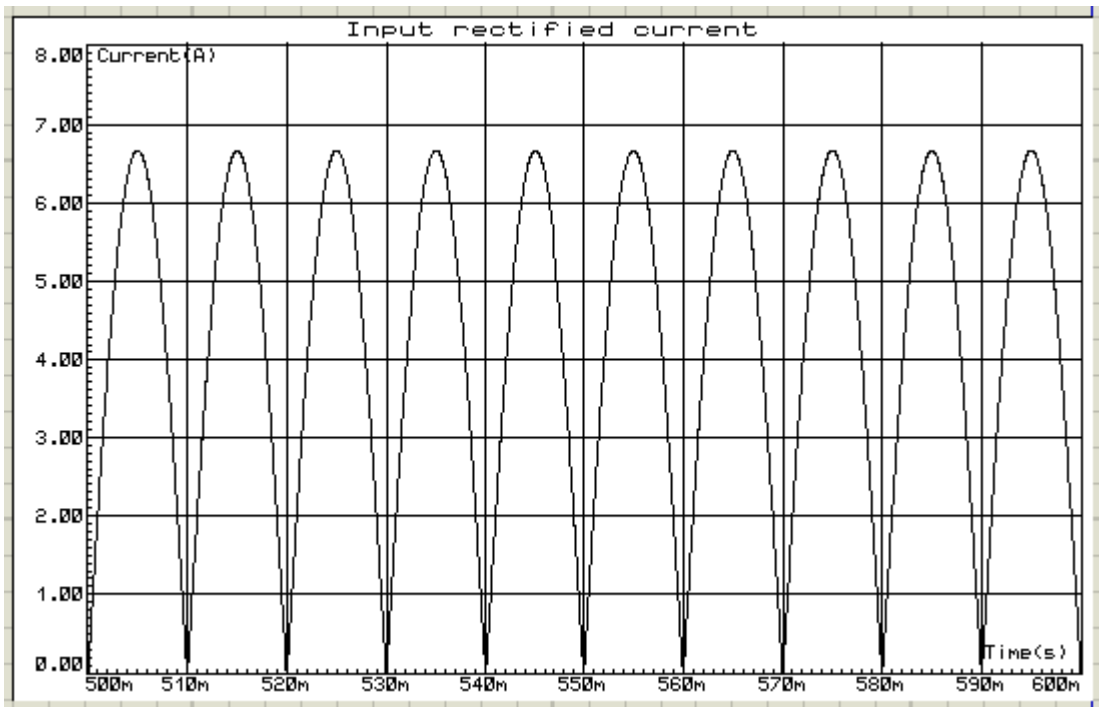


Fig 3.39 Input rectified current for 200Ω load resistance and 283V (peak) input voltage

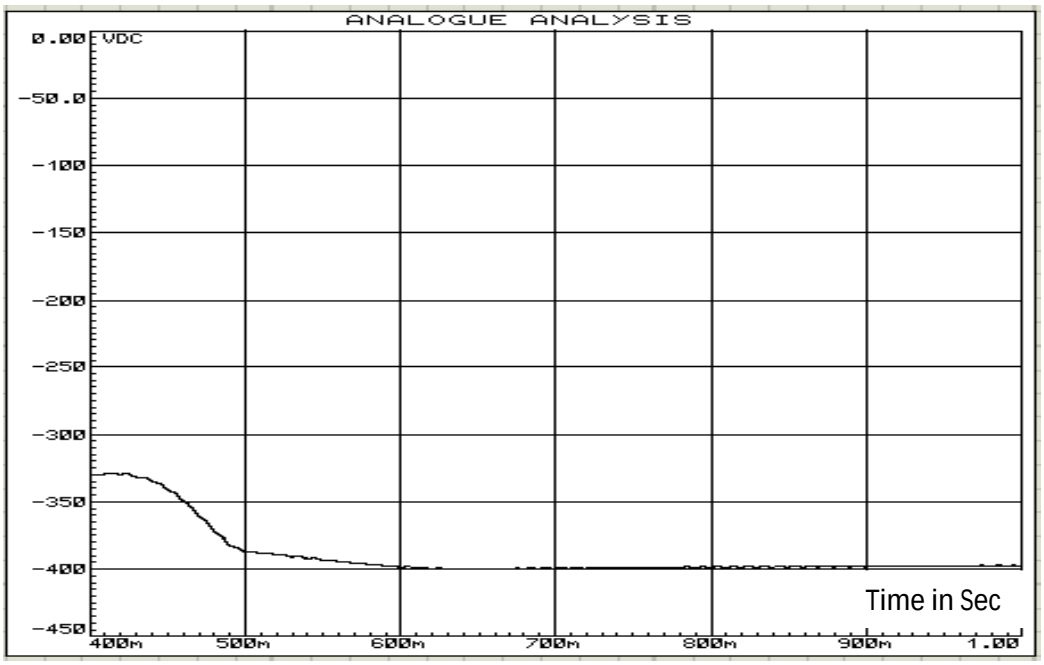


Fig 3.40 Output voltage for 400Ω load resistance and 283V (peak) input voltage

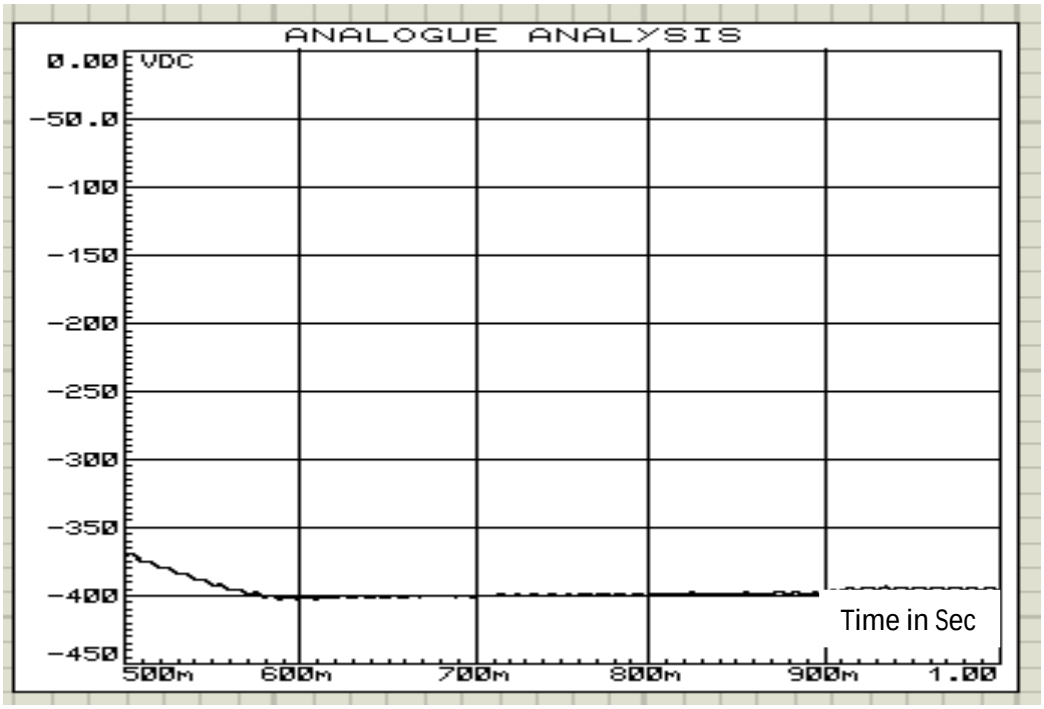


Fig 3.41 Output voltage for 200Ω load resistance and 283V (peak) input voltage

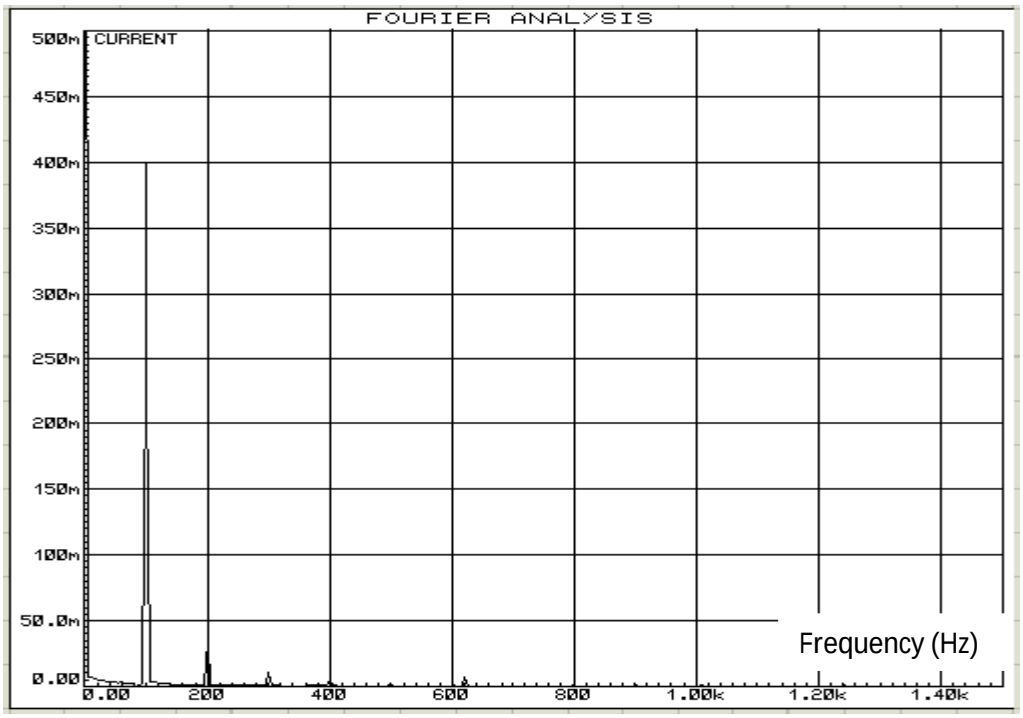


Fig 3.42 FFT component of input current for 400Ω load resistance and 283V (peak) input voltage

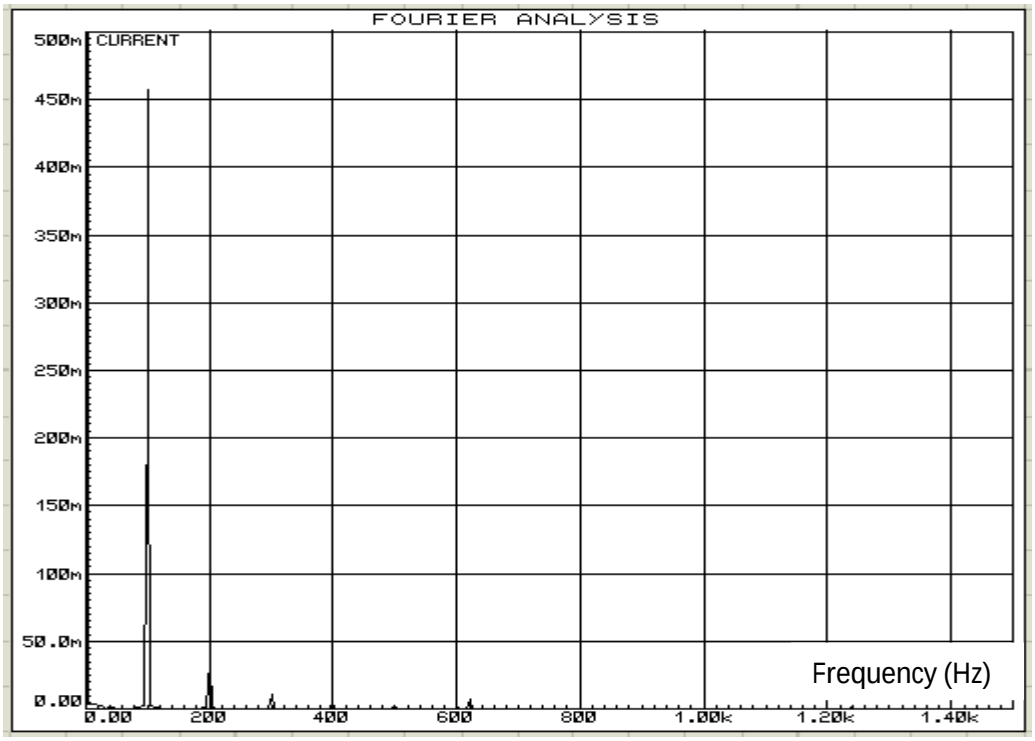


Fig 3.43 FFT component of input current for 200Ω load resistance and 283V (peak) input voltage

Table 3.4 shows the summarize data that is obtained from Fig. 3.37 to Fig. 3.43.

Table 3.4: Data for 400Ω to 200Ω load resistance at 283V (peak) input voltage, 50Hz supply frequency and -400V output voltage

Resistive load in Ohm	Input rms current(A)	Input rms Voltage(V)	Output DC voltage(V)	Input THD in %	Input PF	Efficiency In %
400	2.24	200	399	7.2	0.9974	88.8
200	4.72	200	398	4.98	0.9987	83.9

Fig. 3.44 to Fig. 3.47 show the input PF, input THD, efficiency and output voltage respectively at different load resistances under the supply frequency 50Hz and input voltage 283V(peak).

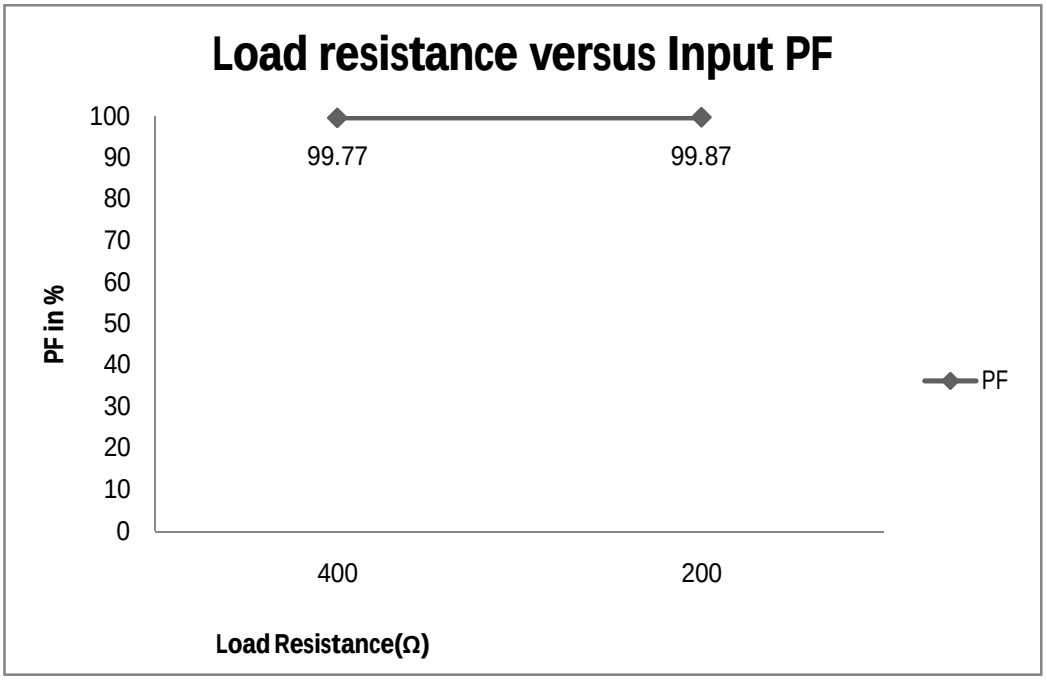


Fig 3.44 Input PF at different loads and 283V (peak) input voltage

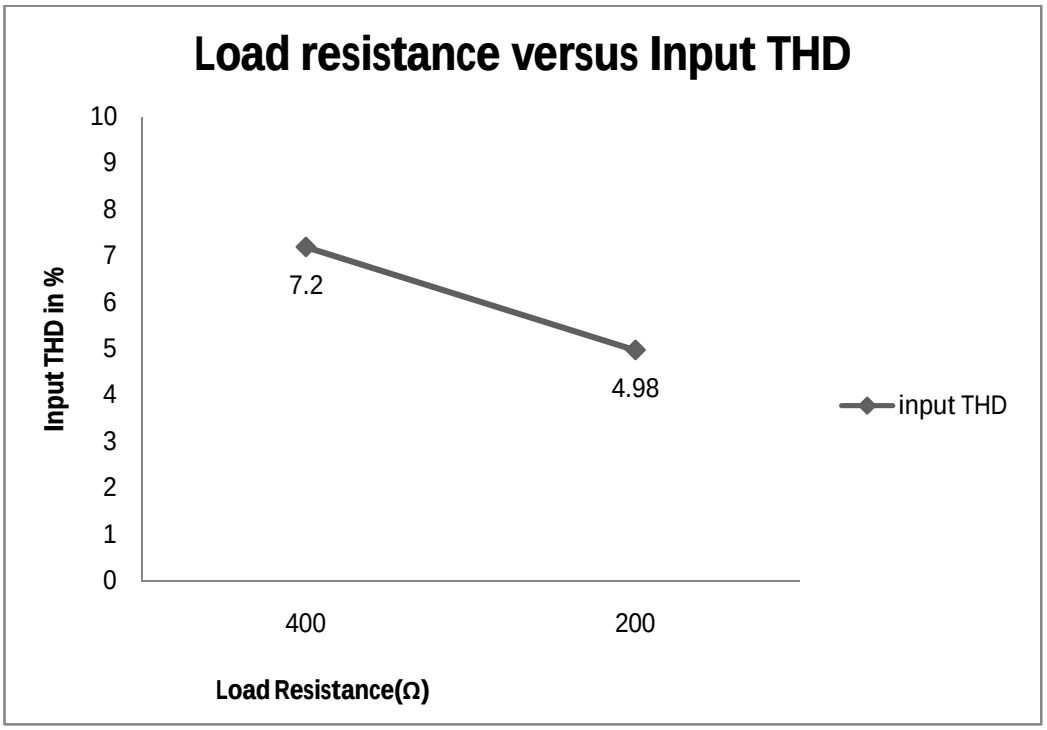


Fig 3.45 Input THD at different loads and 283V (peak) input voltage

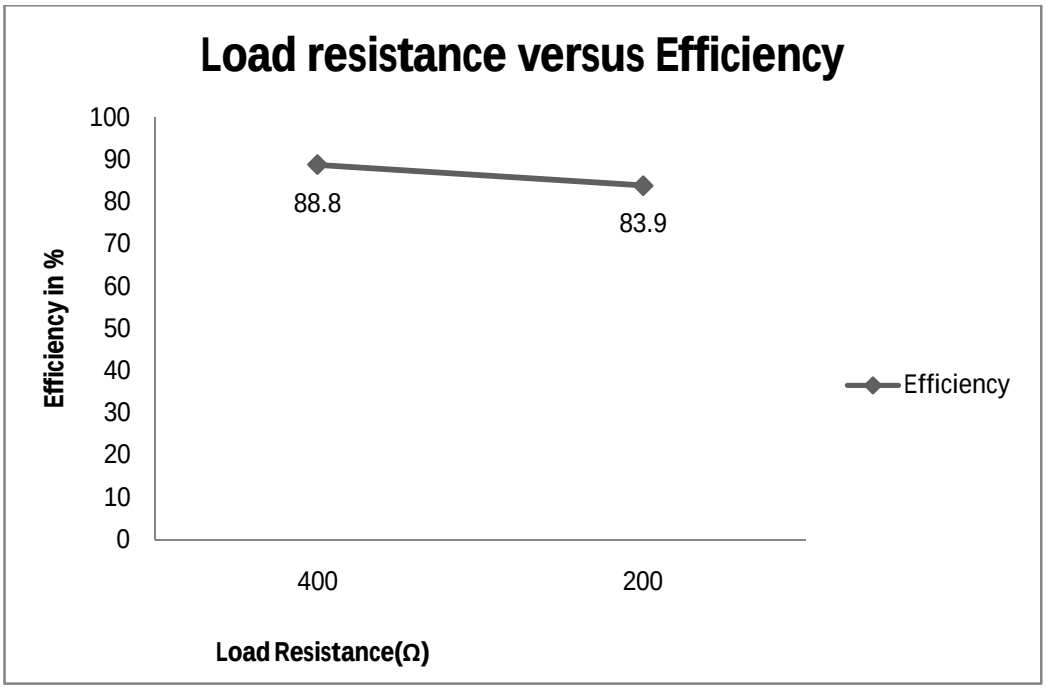


Fig 3.46 Efficiency at different loads and 283V (peak) input voltage

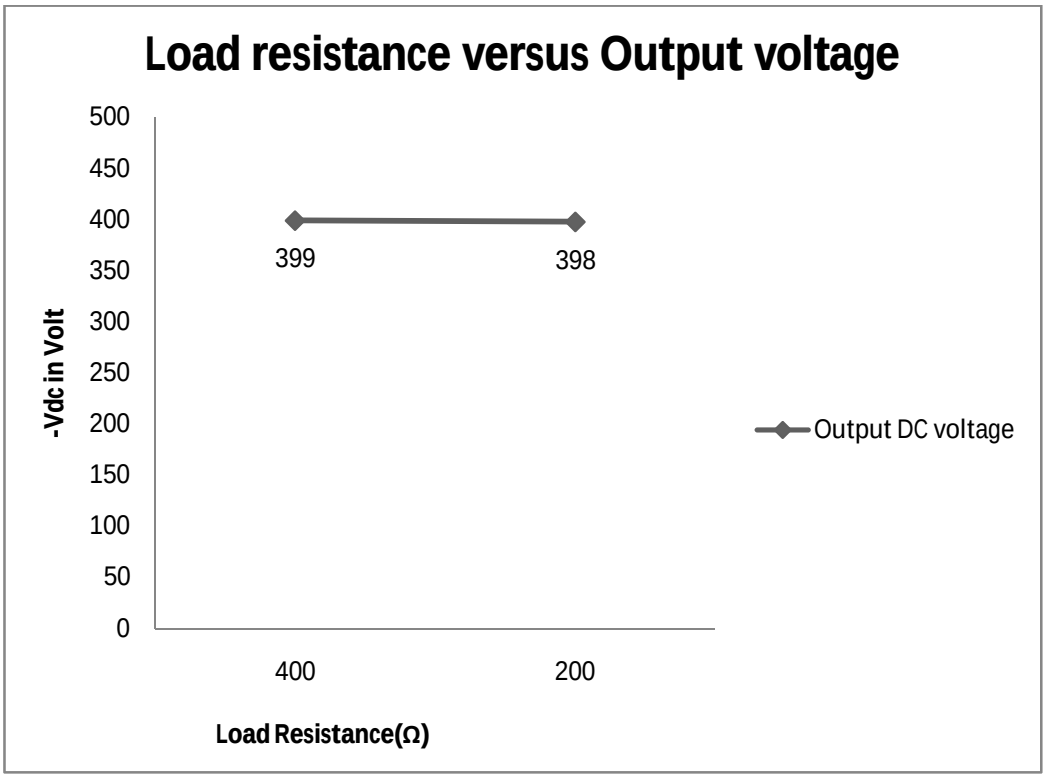


Fig 3.47 Output voltage at different loads and 283V (peak) input voltage

Fig. 3.48 to Fig. 3.54 show the input rectified voltage/current, output voltage and FFT component of input current at different load resistance and -400V reference output voltage under 367V (peak) input voltage and 50Hz supply frequency using proposed close loop Ĉuk rectifier in Proteus software.

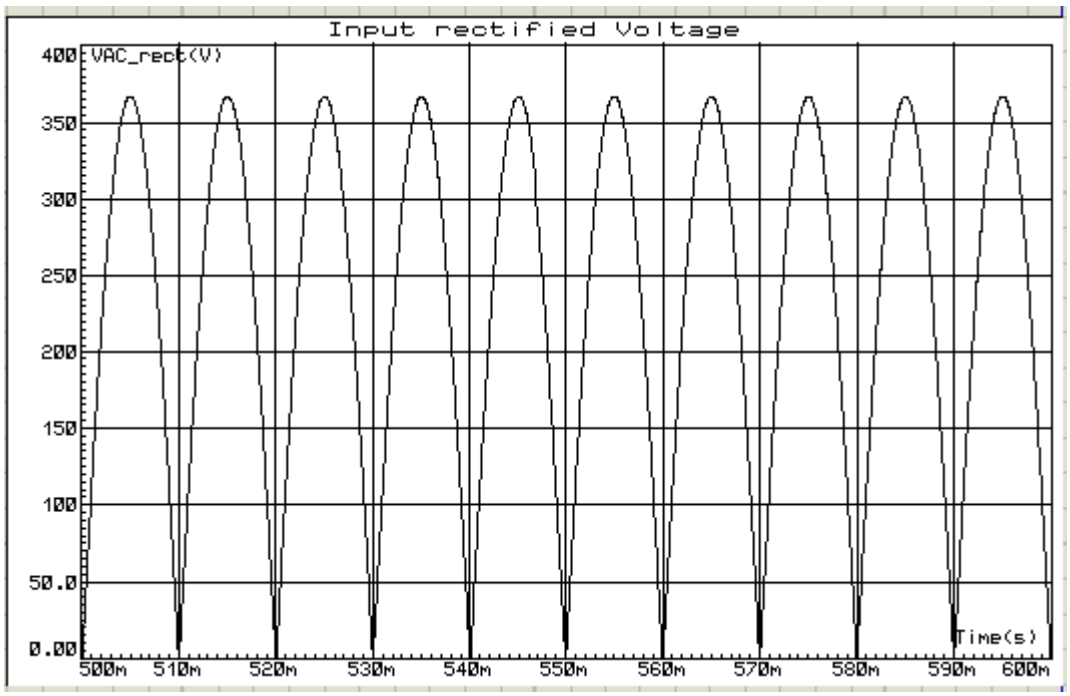


Fig 3.48 Input rectified voltage 367V (peak)

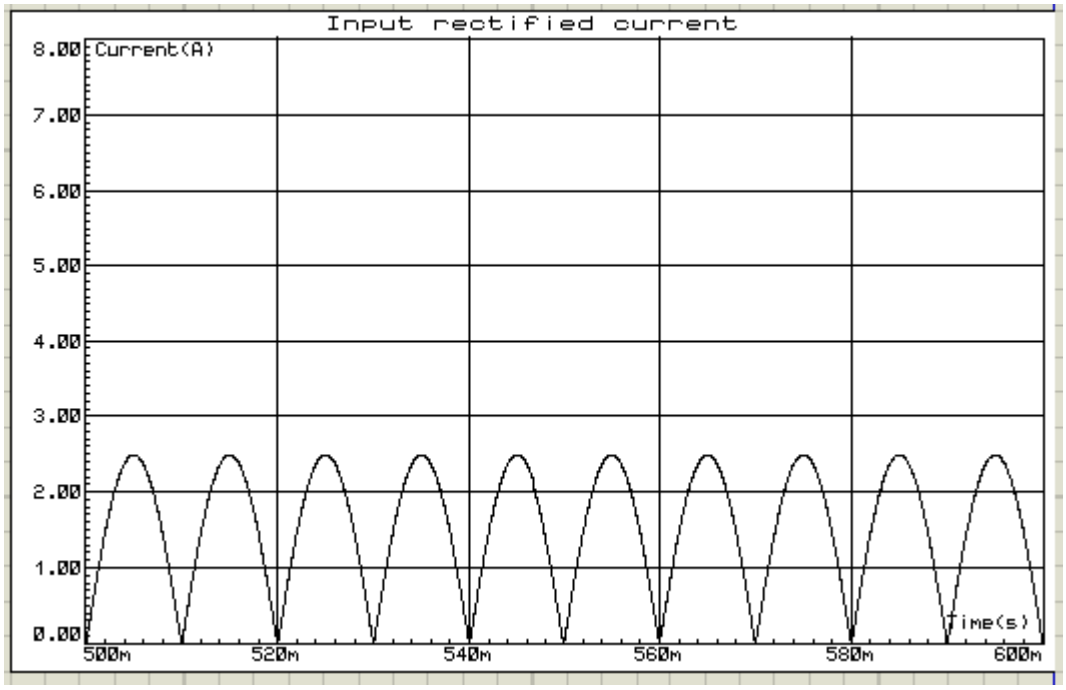


Fig 3.49 Input rectified current for 400Ω load resistance and 367V (peak) input voltage

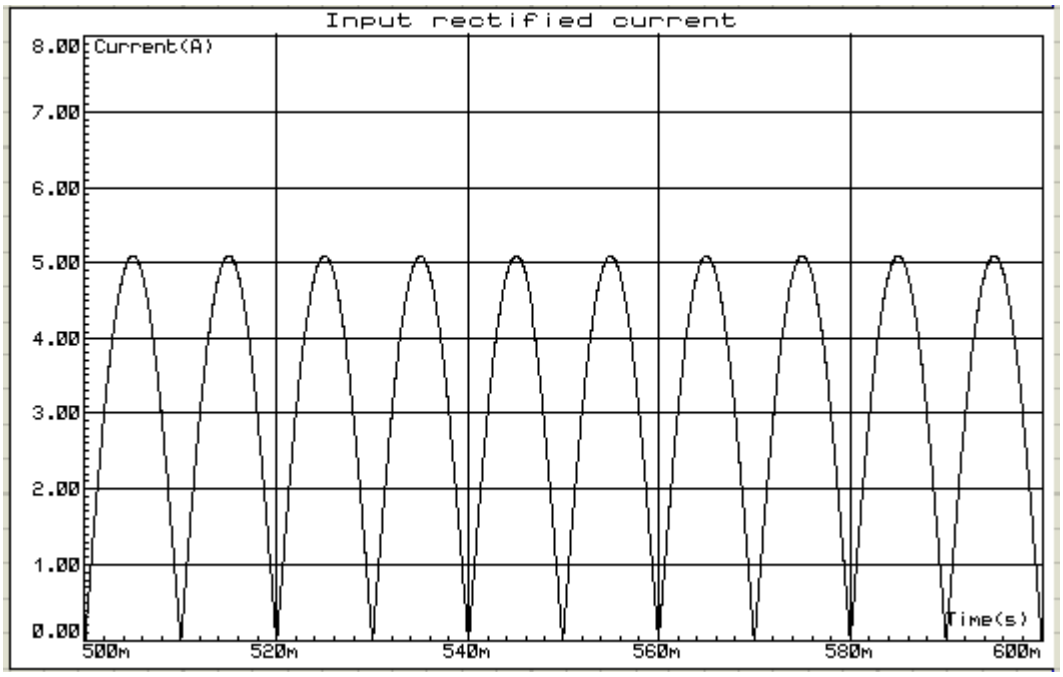


Fig 3.50 Input rectified current for 200Ω load resistance and 367V (peak) input voltage

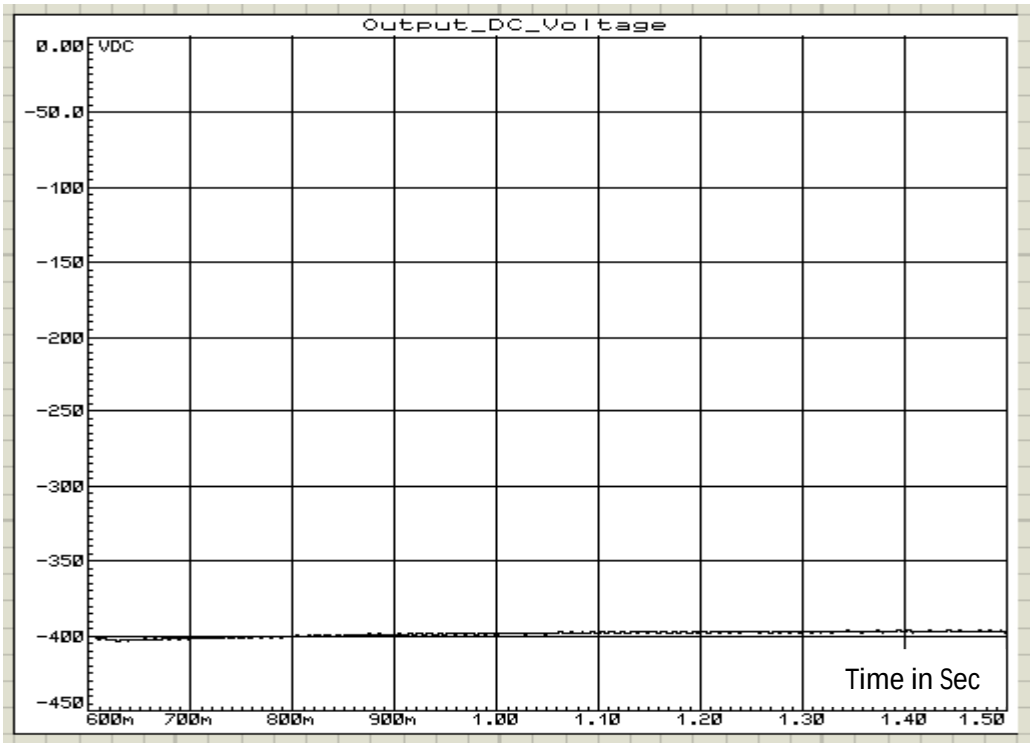


Fig 3.51 Output voltage for 400Ω load resistance and 367V (peak) input voltage

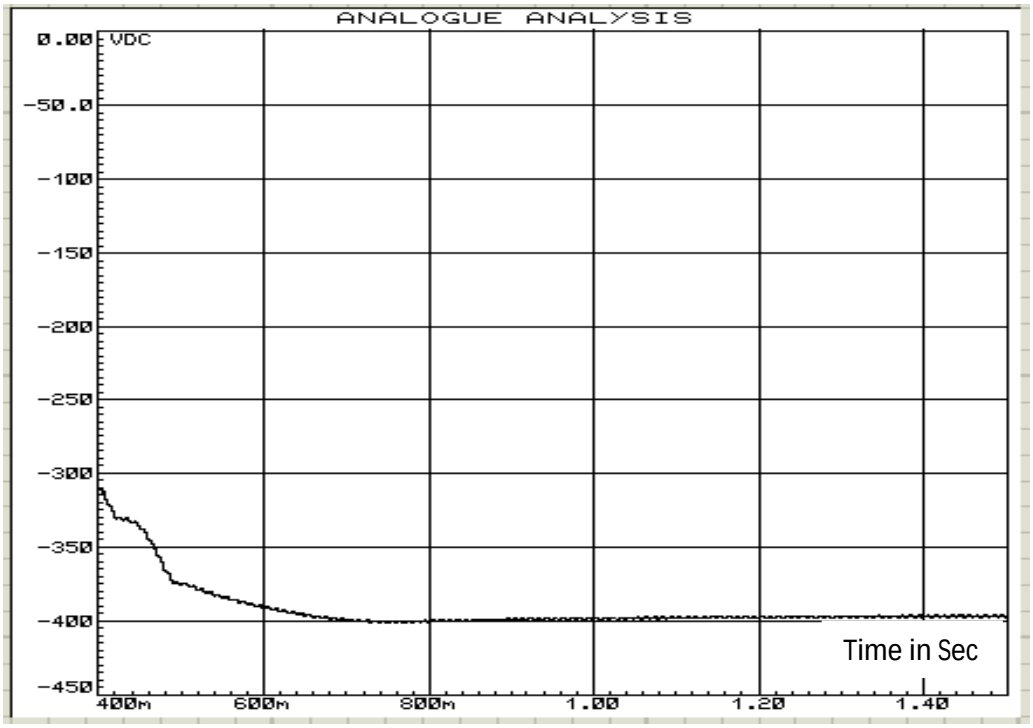


Fig 3.52 Output voltage for 200Ω load resistance and 367V (peak) input voltage

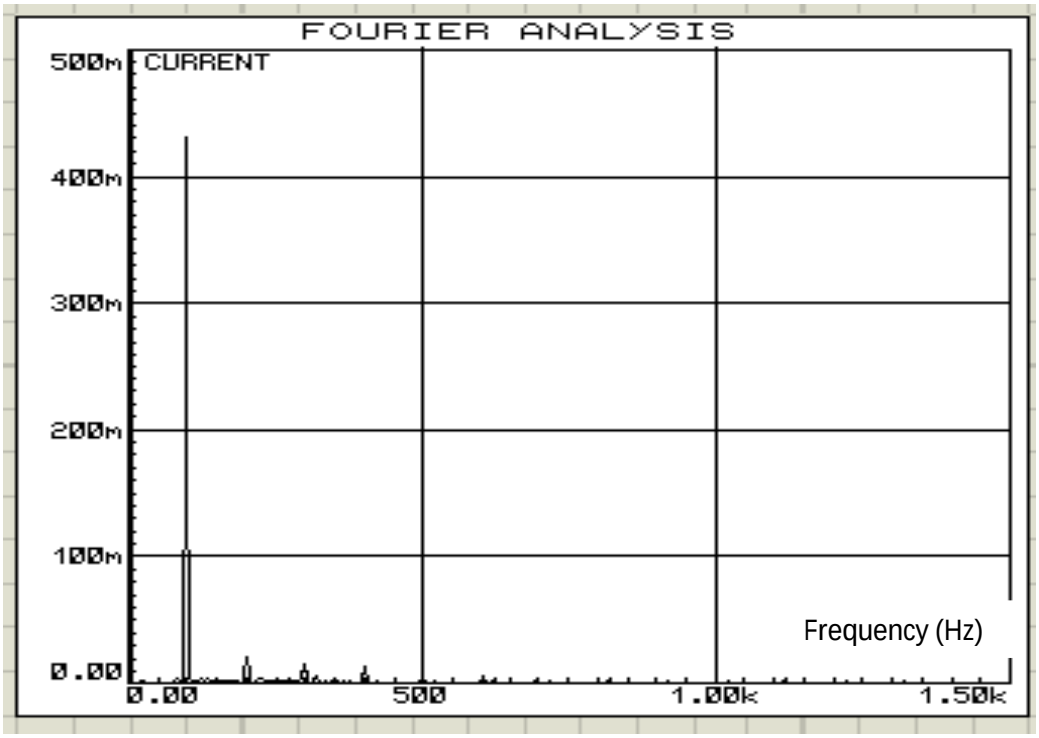


Fig 3.53 FFT component of input current for 400Ω load resistance and 367V (peak) input voltage

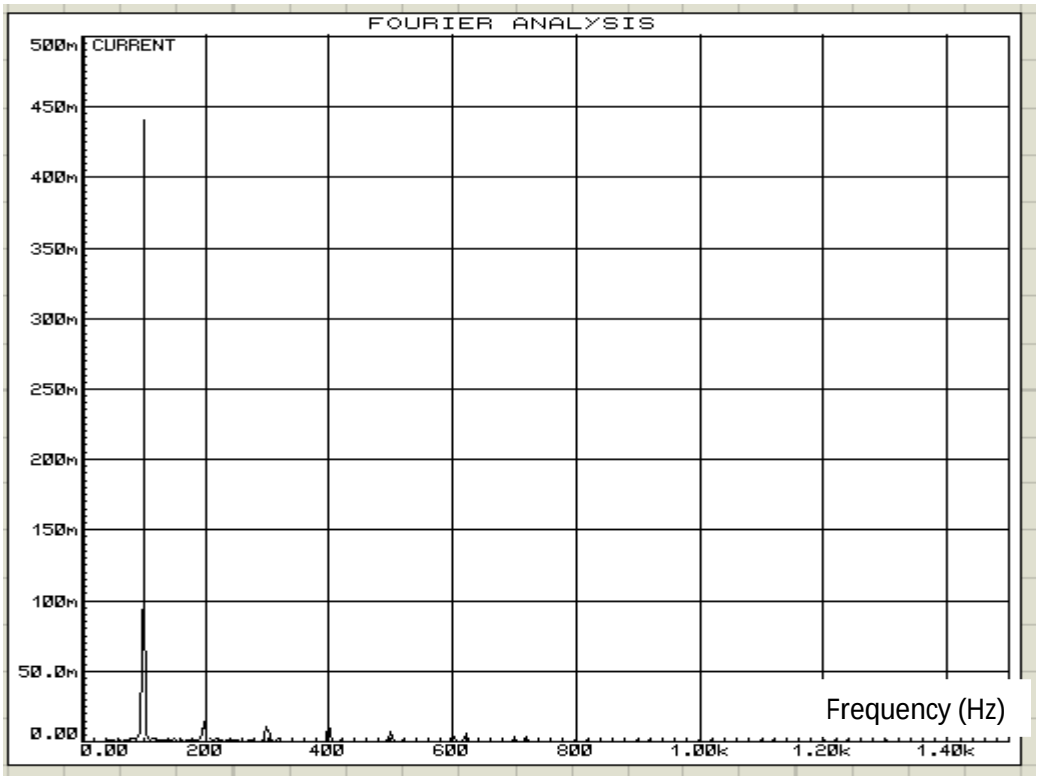


Fig 3.54 FFT component of input current for 200Ω load resistance and 367V (peak) input voltage

Table 3.5 shows the summarize data that is obtained from Fig. 3.48 to Fig. 3.54.

Table 3.5: Data for 400Ω to 200Ω load resistance at 367V (peak) input voltage, 50Hz supply frequency and -400V output voltage

Resistive load in Ohm	Rectified Average Input current(A)	Rectified Average Input Voltage(V)	Output DC voltage(V)	Input THD in %	Input PF	Efficiency In %
400	1.77	260	398	7.8	0.994	86.05
200	3.61	260	399	3.3	0.999	84.81

Fig. 3.55 to Fig. 3.58 show the input PF, input THD, efficiency and output voltage respectively at different load resistances under the supply frequency 50Hz and input voltage 260V(rms).

Fig. 3.59 shows the output voltage at different input voltages under same load resistance.

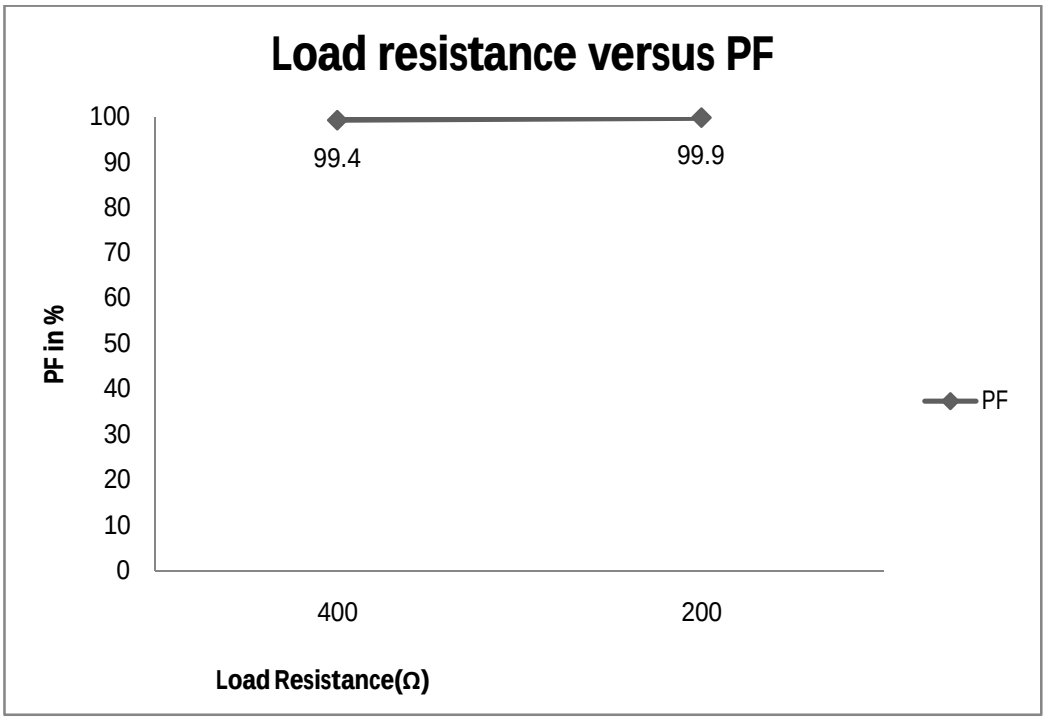


Fig 3.55 Input PF at different loads and 367V (peak) input voltage

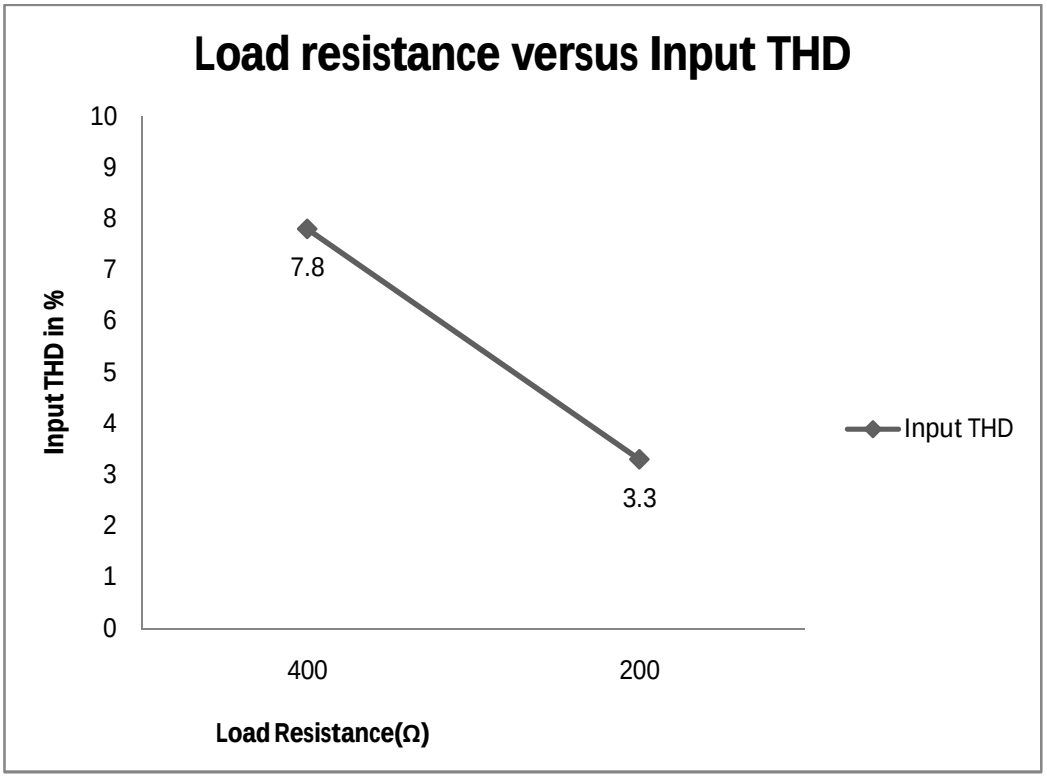


Fig 3.56 Input THD at different loads and 367V (peak) input voltage

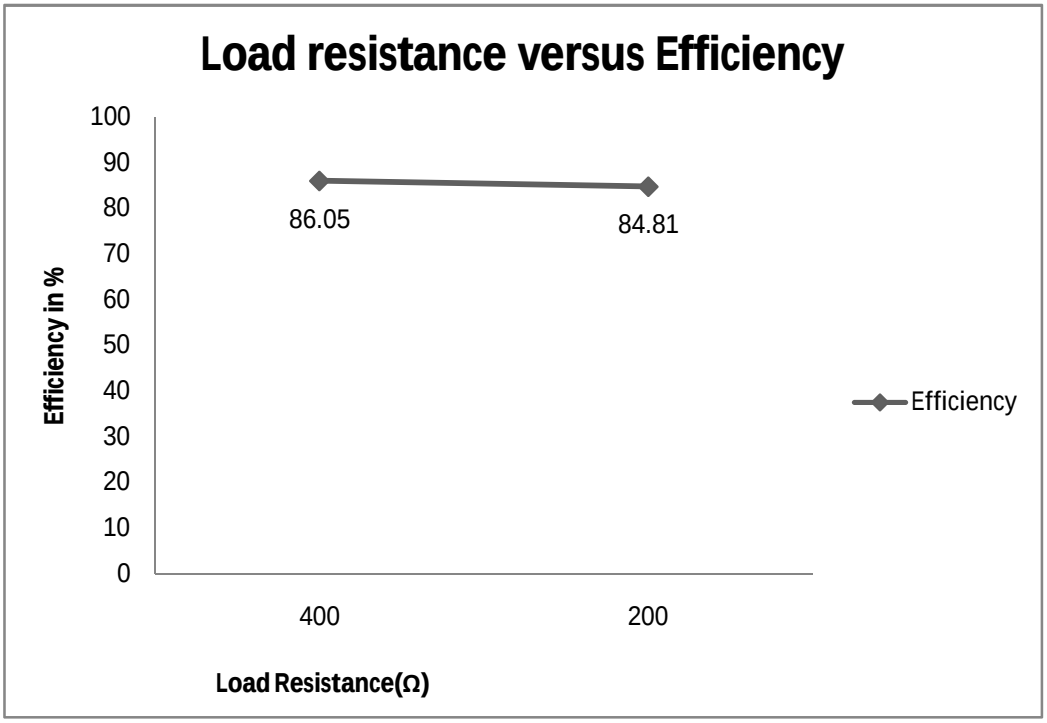


Fig 3.57 Efficiency at different loads and 367V (peak) input voltage

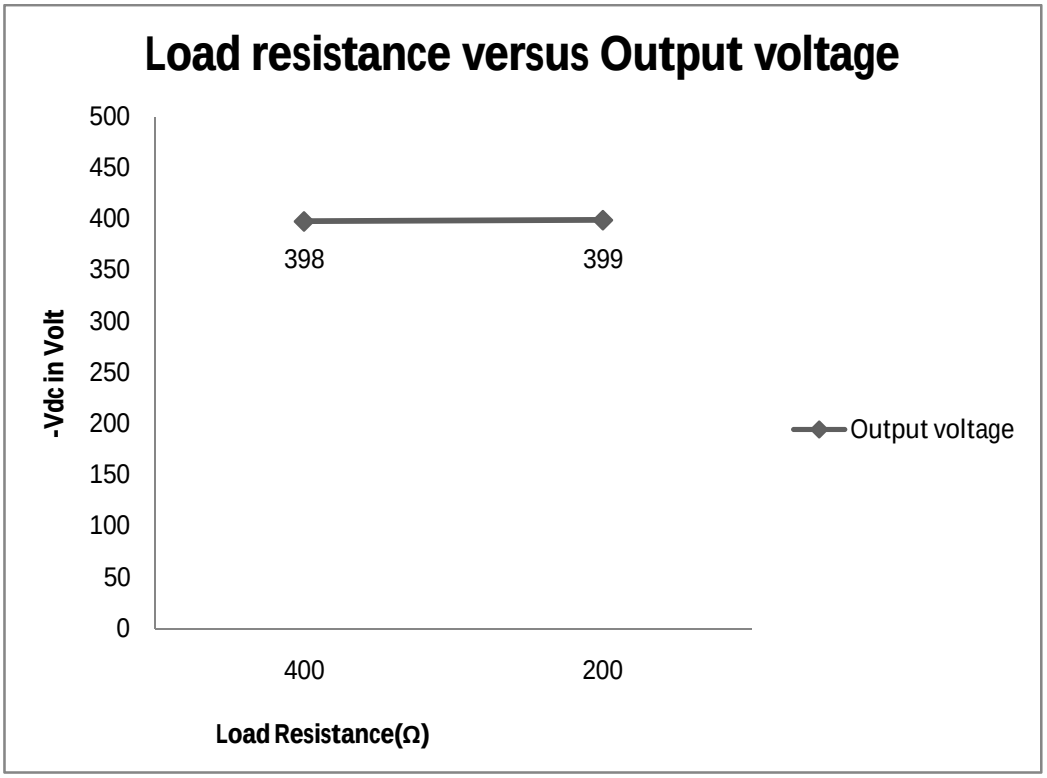


Fig 3.58 Output voltage at different loads and 367V (peak) input voltage

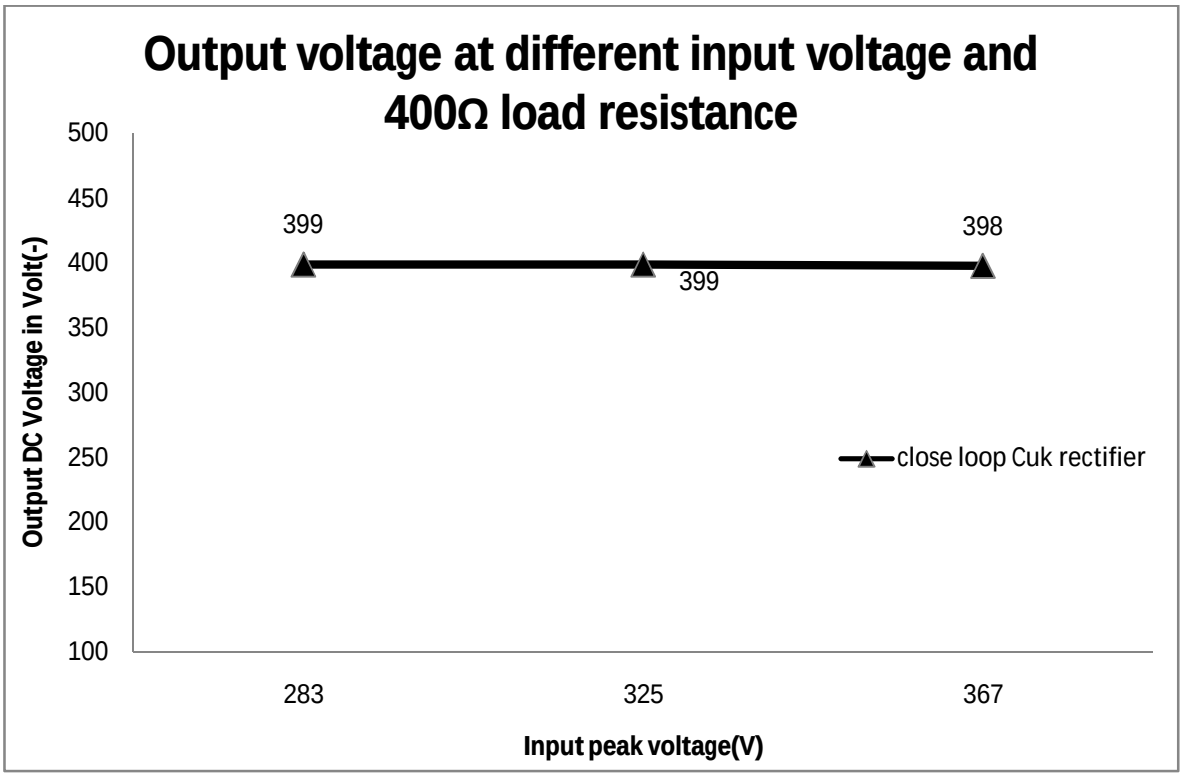


Fig 3.59 Output voltage at different input voltages

Fig. 3.60 to Fig. 3.62 shows the output voltage at different input supply frequency and 400Ω load resistance. Similarly Fig. 3.63 to Fig. 3.65 shows the output voltage at different input supply frequencies and 200Ω load resistance.

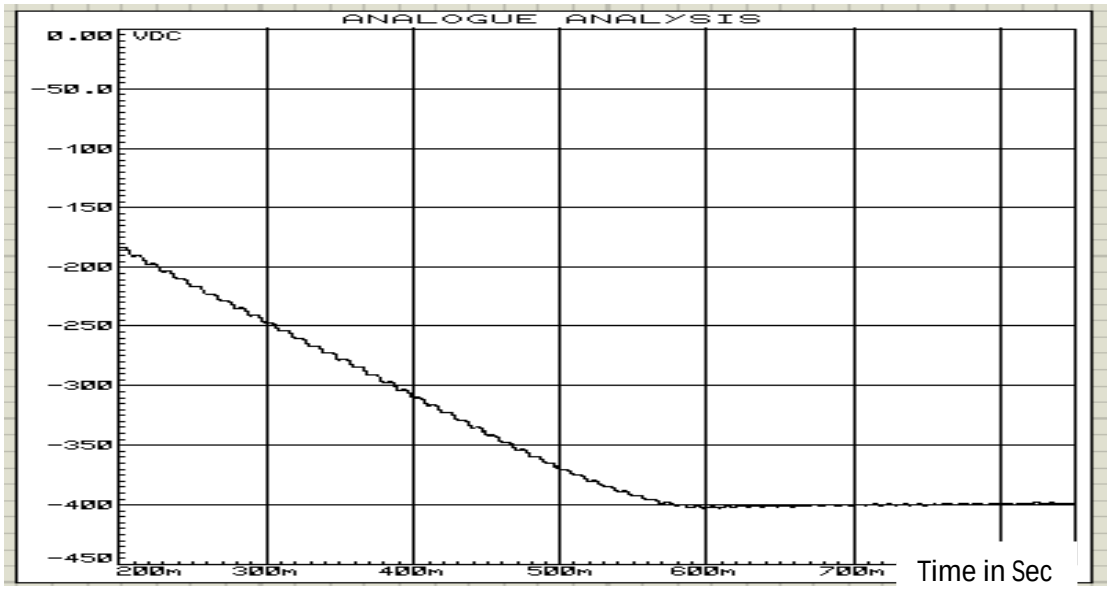


Fig 3.60 Output voltage for 400Ω load resistance and 45Hz input frequency

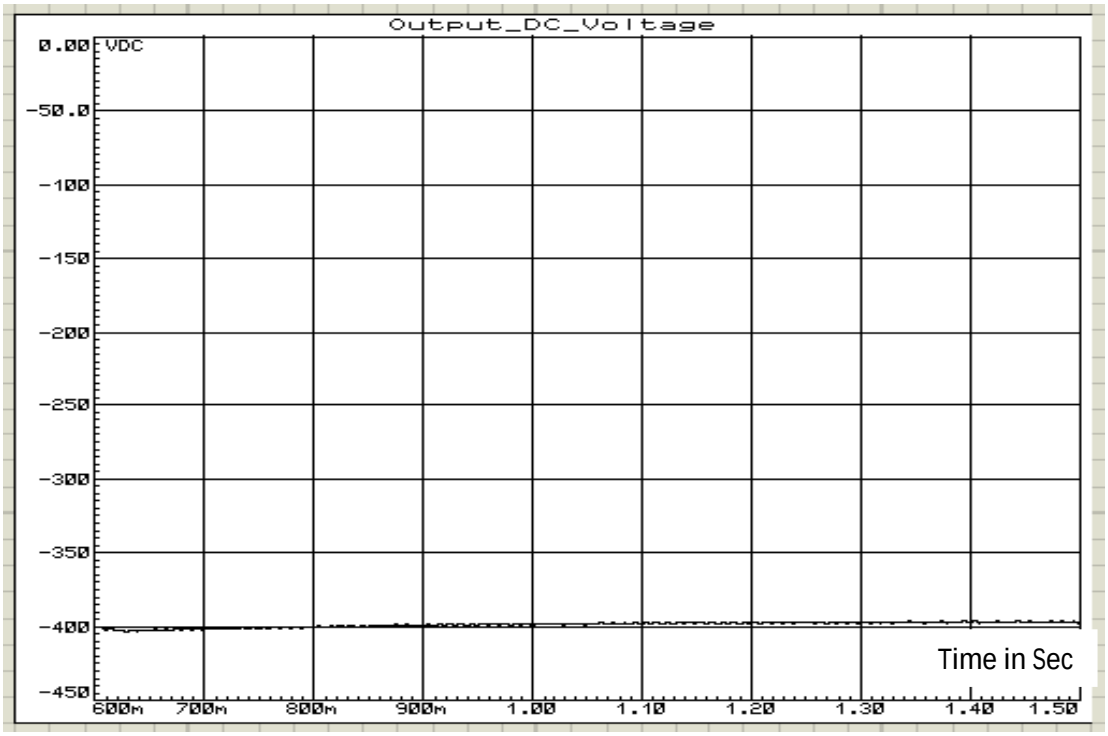


Fig 3.61 Output voltage for 400Ω load resistance and 50Hz input frequency

Fig 3.62 Output voltage for 400Ω load resistance and 55Hz input frequency

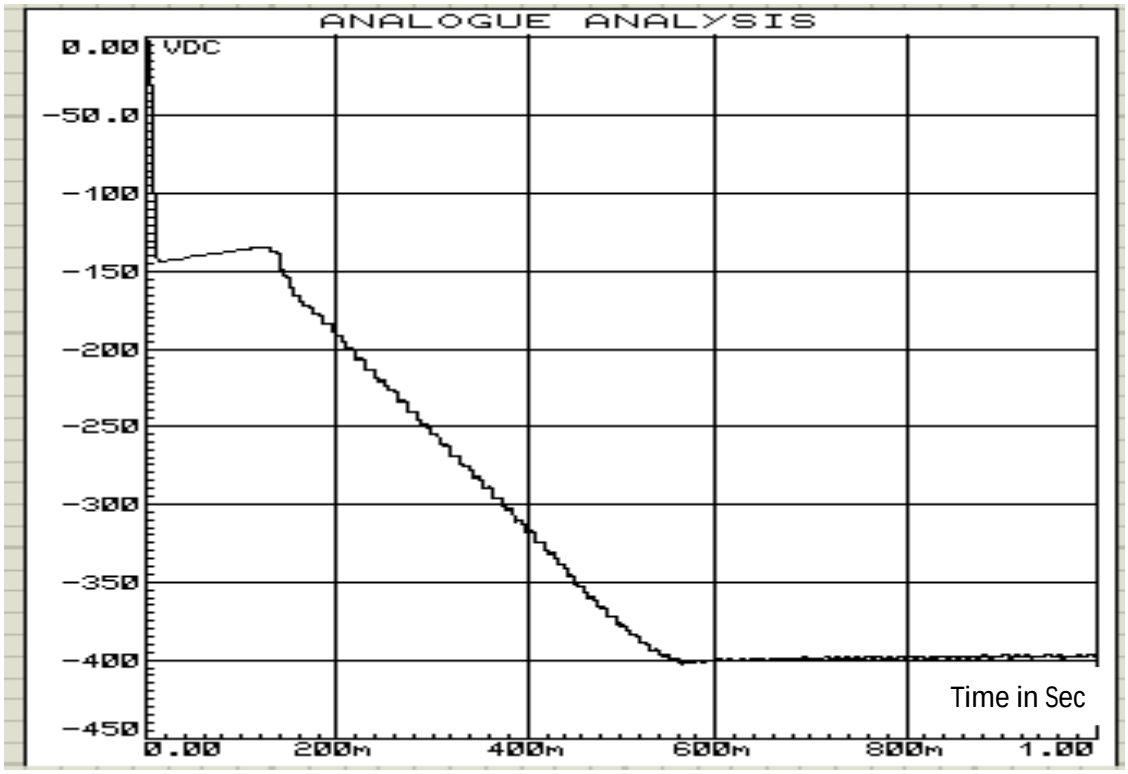


Fig 3.63 Output voltage for 200Ω load resistance and 45Hz input frequency

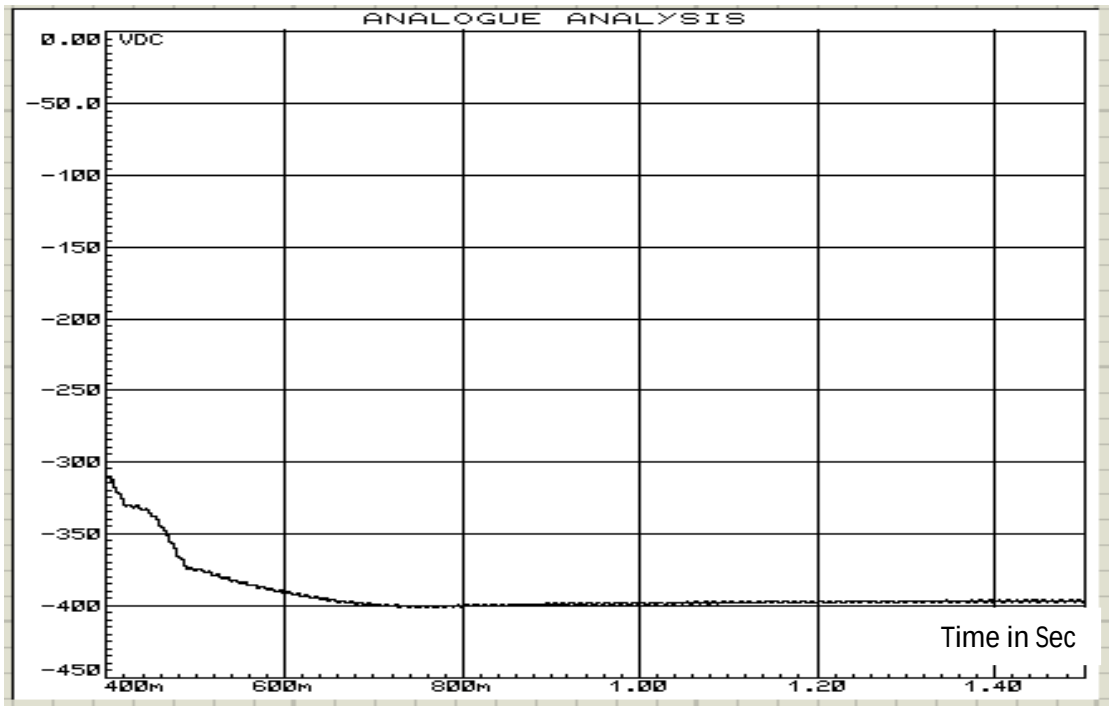


Fig 3.64 Output voltage for 200Ω load resistance and 50Hz input frequency

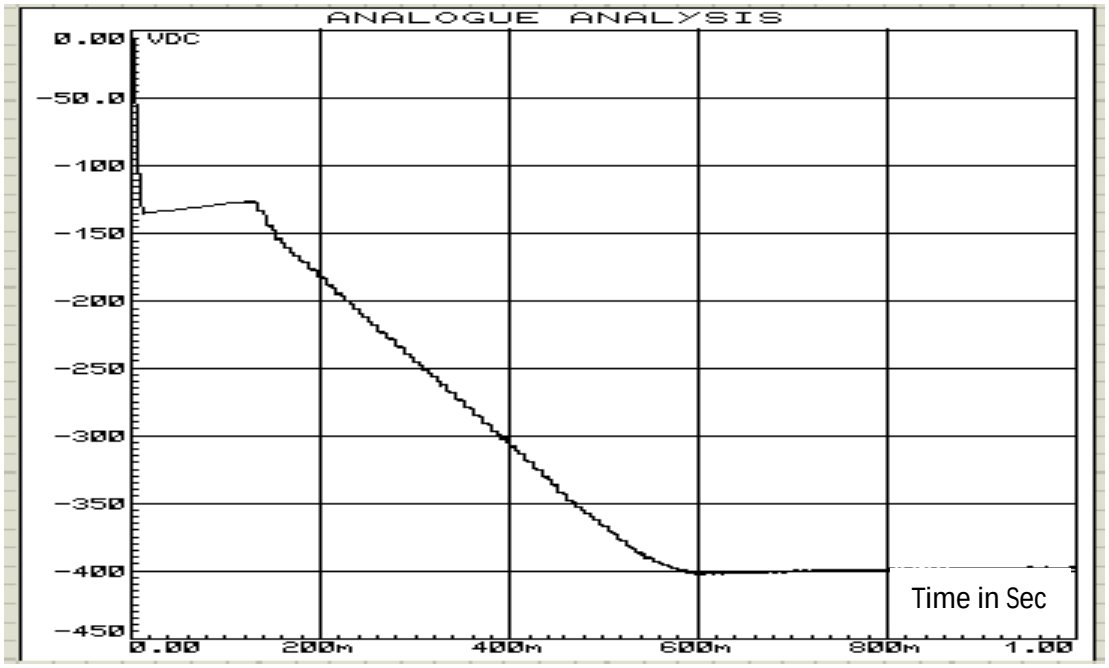


Fig 3.65 Output voltage for 200Ω load resistance and 55Hz input frequency

Table 3.6 shows the summarize data that is obtained from Fig. 3.60 to Fig. 3.65

Table 3.6: Data for 400Ω to 200Ω load resistance at 325V (peak) input voltage, and different input supply frequency and -400V output voltage reference

Frequency (Hz)	Output voltage(V)
For Load resistance R= 400Ω	
45	400
50	398
55	400
For Load resistance R= 200Ω	
45	398
50	398
55	399

Fig. 3.72 and Fig 3.73 show the effect of input supply frequency variation on output reference voltage at 400Ω and 200Ω load resistance respectively.

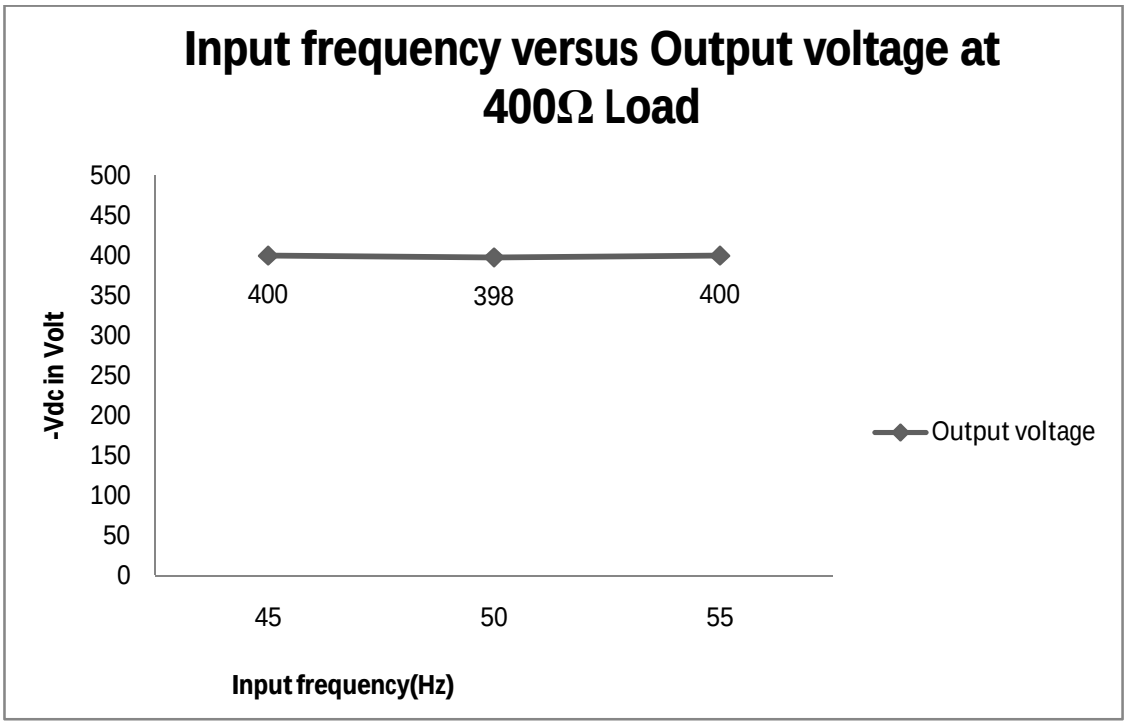


Fig 3.66 Output voltage at different input frequency and 400Ω load resistance

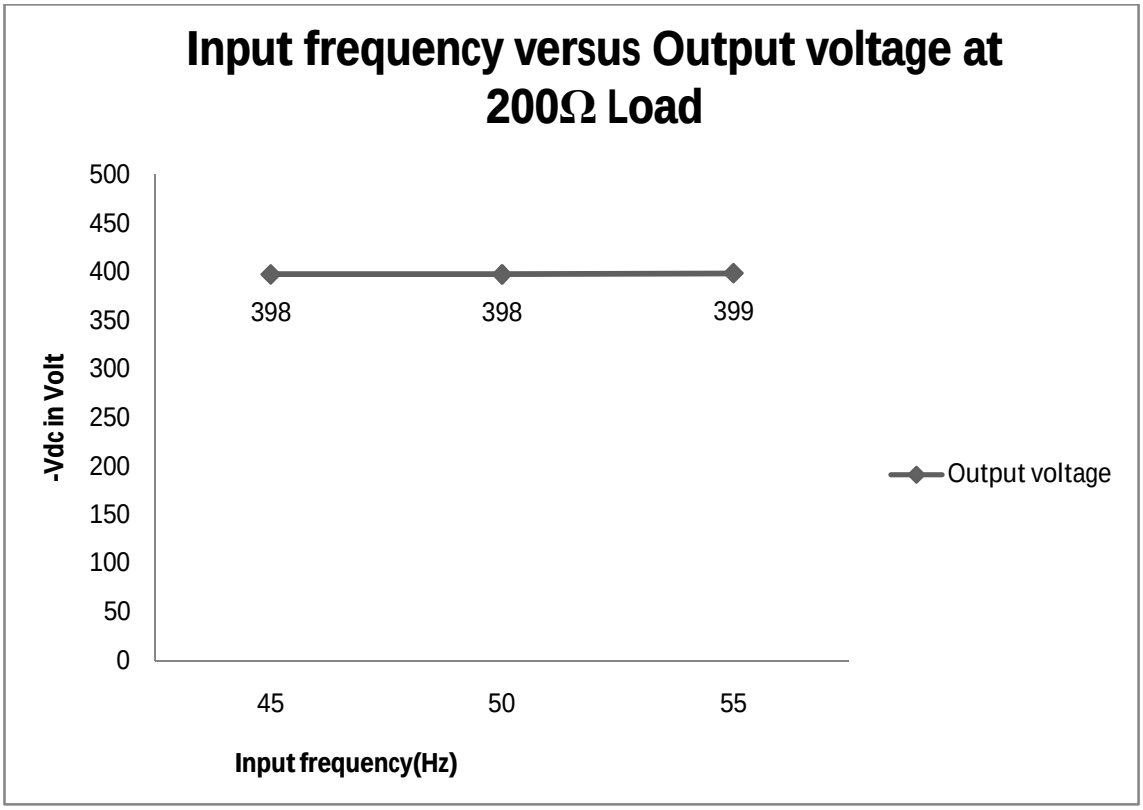


Fig 3.67 Output voltage at different input frequency and 200Ω load resistance

3.12 Relative performance between normal diode rectifier, open loop Ĉuk AC-DC converter and proposed close loop Ĉuk AC-DC converter.

Fig. 3.68 to Fig. 3.72 present the comparative performance of three systems in terms of input current THD, PF, Efficiency and output voltage regulation under changes of input voltage and output load.

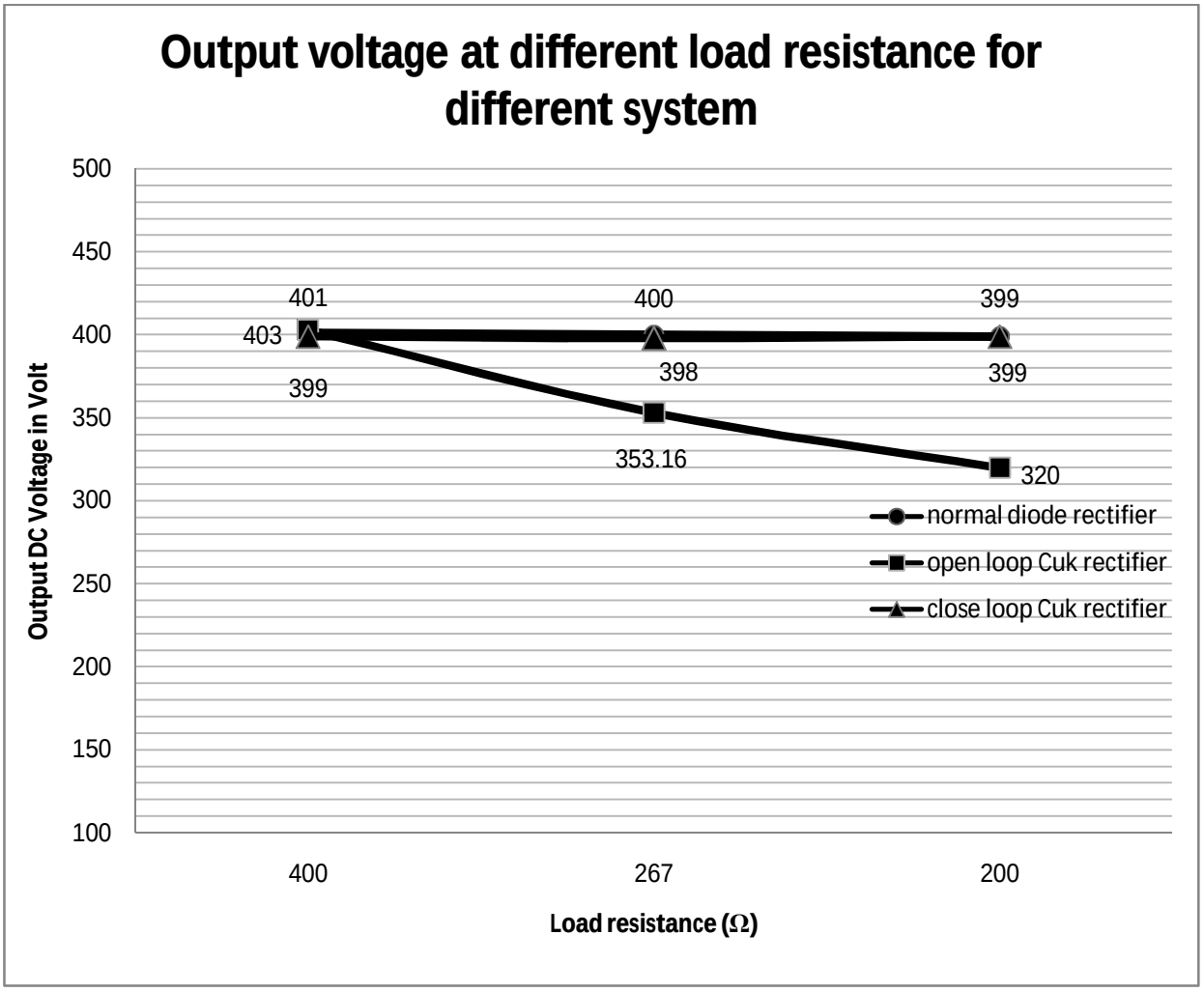


Fig 3.68 Output voltage at different loads and 325V (peak) input voltage

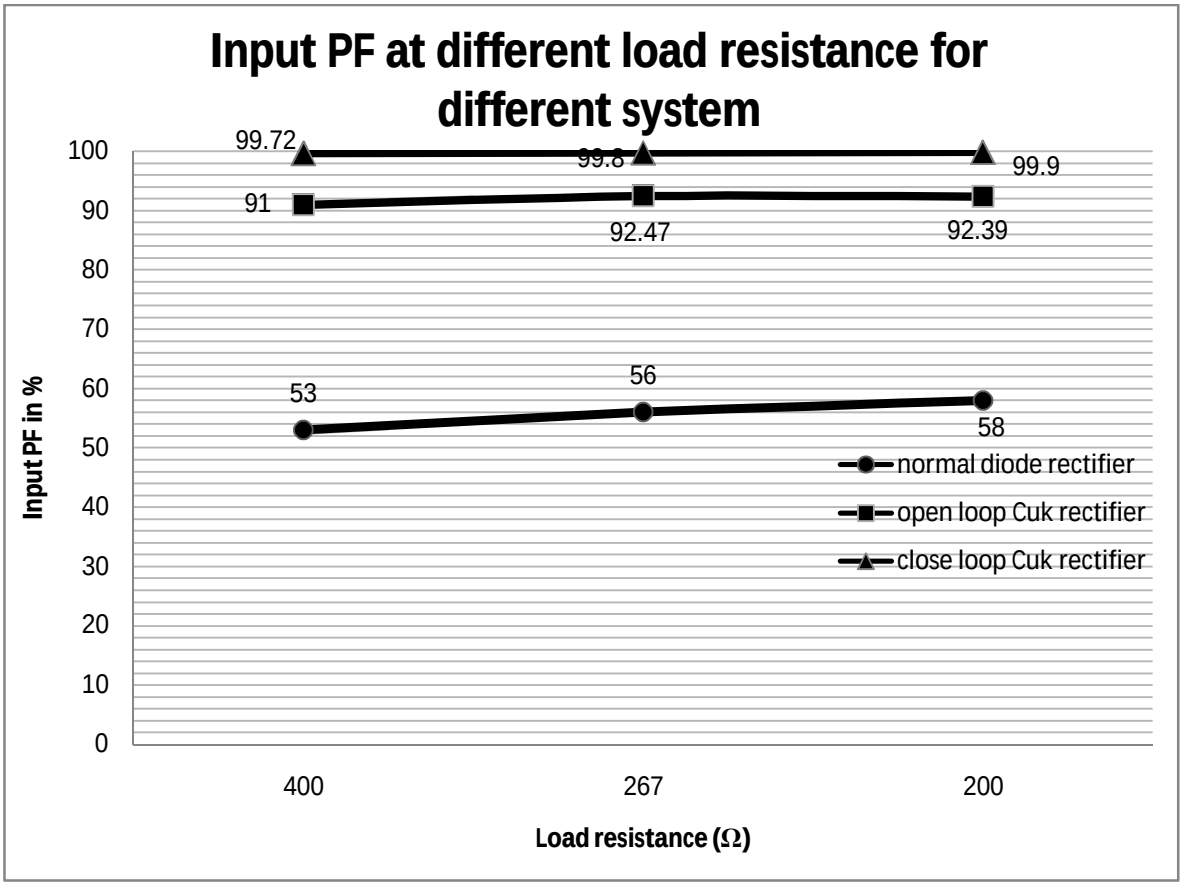


Fig 3.69 Input PF at different loads and 325V (peak) input voltage

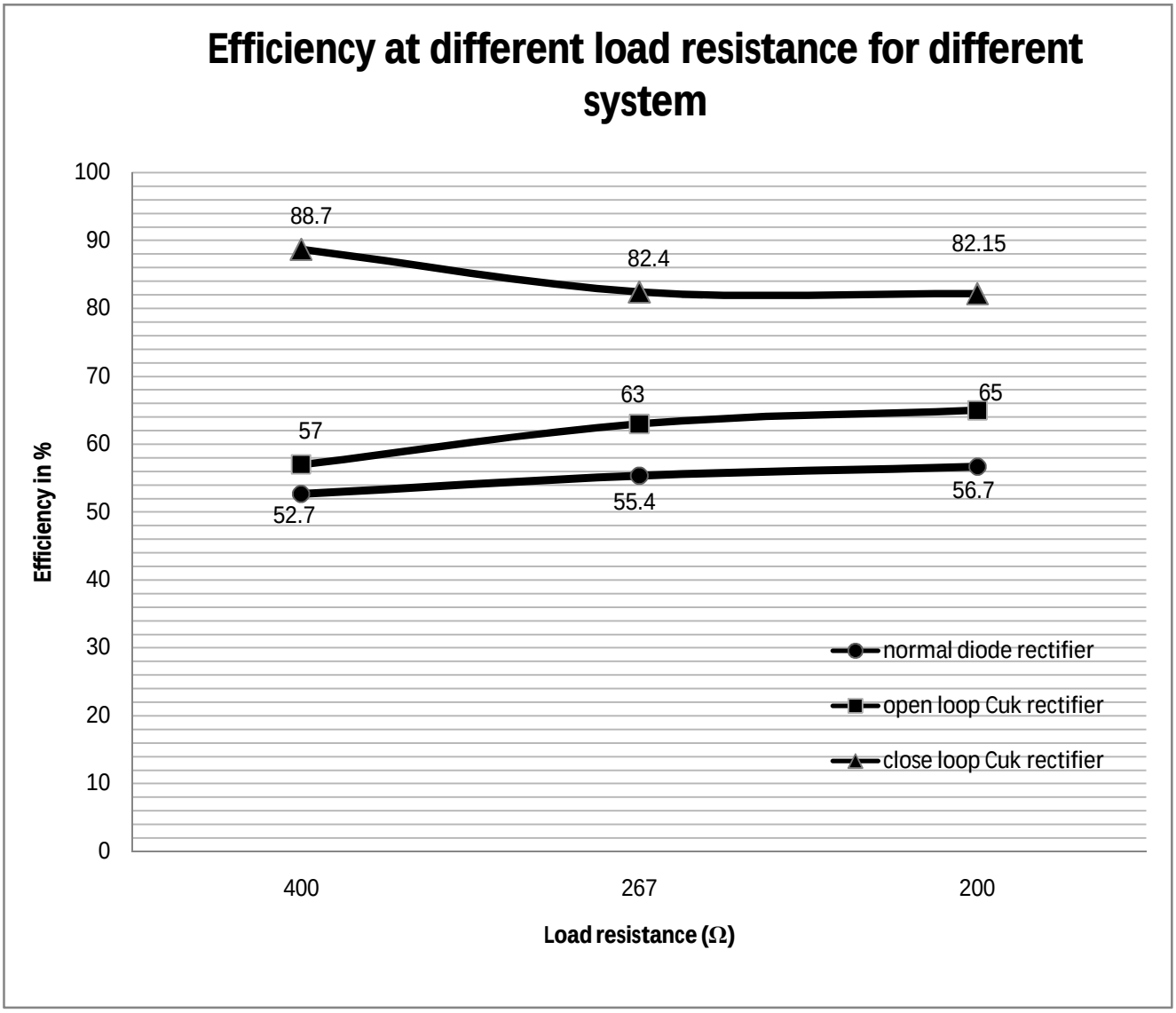


Fig 3.70 Efficiency at different loads and 325V (peak) input voltage

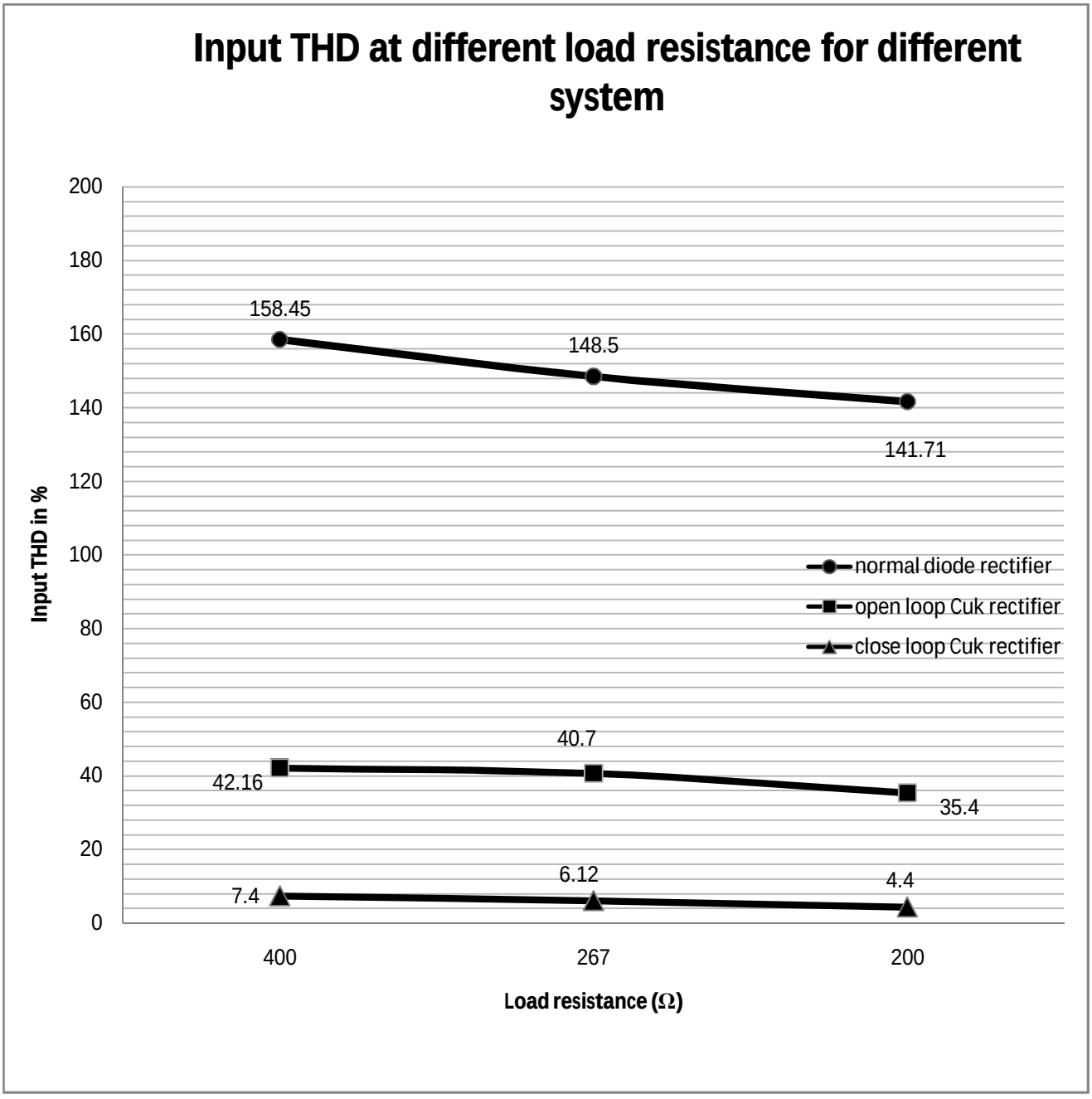


Fig 3.71 Input current THD at different loads and 325V (peak) input voltage

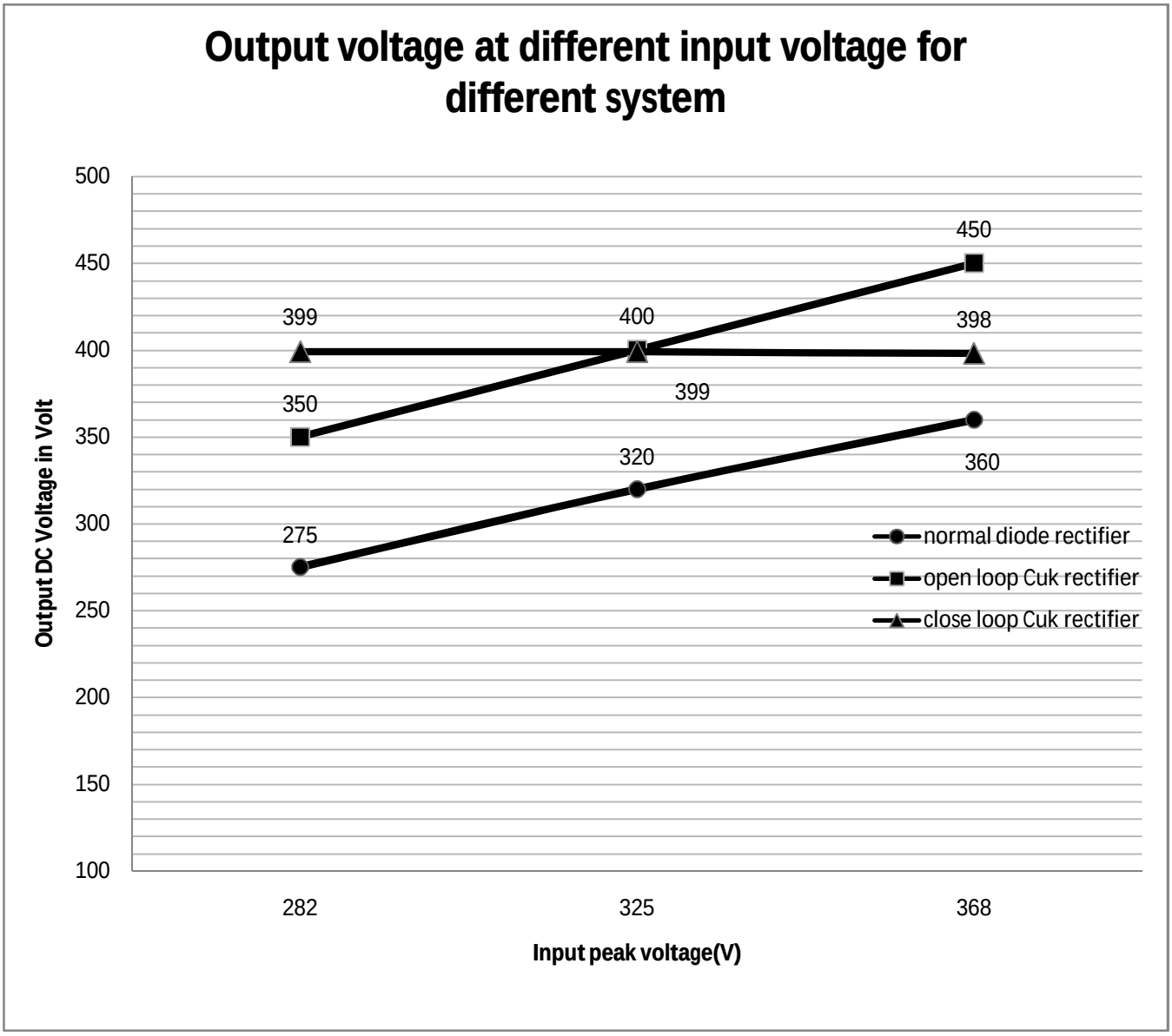


Fig 3.72 Output voltage at different input voltages with 400Ω load resistance

CHAPTER 4

CONCLUSION AND RECOMMENDATIONS FOR FUTURE WORK

4.1 Conclusion

The work presented in this thesis shows the development of a close loop Ĉuk based single phase AC-DC converter. It has seen that the performance is much better than the open loop Ĉuk based AC-DC converter in terms of input THD, input PF, efficiency and output voltage regulation. It has been observed that the output voltage is constant under the input voltage variation from 200V (rms) to 260V (rms), input frequency variation from 45Hz to 55 Hz and output load variation from 200W to 1000W in the proposed system compliance IEEE 519-1992 and IEC 61000-3-2/IEC 61000-3-4[3] standard regarding input THD, power factor and efficiency. The control model was relatively complex to implement through analog circuitry hence a DSC platform has been used with simple voltage and current sensors. The high speed DSC can easily handle the ADC, PI control and PWM generation etc.

.

4.2 Recommendations for Future Work

Though the proposed system has good performance regarding input THD, near unity power factor and output voltage regulation, the efficiency is still less than 90%. In the front end full wave diode rectifier has been used which increases loss due to its forward bias voltage drop 0.7V. So diode can be replaced with switch (IGBT, MOSFET etc). The resistance of inductor and capacitor has been neglected due to complexity to obtain its transfer function. As a result Kp and Ki tuning may have introduced error in the control system. Accurate Dynamic model can be investigated in power stage to ensure better Kp and Ki. The proposed system has been designed and simulated for resistive load other load type can be investigated instantaneously and the system may be investigated in real time implementation.

REFERENCES:

[1] Zixin Li, Yaohua Li, Ping Wang, Hai bin Zhu, Cong wei Liu and Wei Xu ,"Control of Three-Phase Boost-Type PWM Rectifier in Stationary Frame under Unbalanced Input Voltage" *IEEE Trans.* On Power Electronics, VOL. 25, NO.10, OCTOBER 2010, pp. 2521-2530

[2] Thomas Nussbaumer and Johann W.Kolar "Comparison of 3-Phase Wide Output Voltage Range PWM Rectifiers", *IEEE Tran* on Power electronics,VOL.54, NO.6, DECEMBER 2007. pp. 3422-3425

[3] H.Azizi and A.Vahedi, "Performance Analysis of Direct Power Controlled PWM Rectifier under Disturbed AC Line Voltage ", ICREPQ'05 , 15-16-17 March 2005, Zaragoza- Spain Serial 244, 2005 pp.1-6

[4] Mohammad Ziauddin Alamgir, "Single Switch Buck Boost AC-DC Converster" M.SC. Thesis EEE, BUET, 2011

[5] Ruma and Mohammad Ali Choudhury, "Power Factor Improvement of a Three Phase Rectifier by Boost Regulator" Journal of Electrical Engineering The Institution of Engineers, Bangladesh Vol. EE 36, No. II, December 2009

[6] K. Periyasamy, "Power Factor Correction Based On Fuzzy Logic Controller With Average Current-Mode For DC-DC Boost Converter", International Journal of Engineering Research and Applications (IJERA) Vol. 2, Issue 5, September- October 2012, pp.771-777

[7] Mahadev S. Patil, "Single phase buck type power factor corrector with lower harmonic contents in compliances with IEC 61000-3-2", International Journal of Engineering Science and Technology Vol. 2(11), 2010, pp.6122-6130

[8] Ray-Lee Lin and Rui-Che Wang, "Non-inverting Buck-Boost Power-Factor-Correction Converter with Wide Input-Voltage-Range Applications", IECON 2010 Page(s): 599 –604

[9] Jayanthi, D. "A novel buck converter for power factor correction using PWM technique", Advances in Engineering, Science and Management (ICAESM), 2012 Page(s): 309 – 314

[10] Kamran Rezaei, "A Control Scheme for an AC-DC Single-Stage Buck-Boost PFC Converter with Improved Output Ripple Reduction", M.SC. thesis ECE, The University of Western Ontario 2012

[11] Majid Jamil and Zahra Mehdi, "Power factor improvement of cascaded buck boost converter" National power electronics conference (NPEC-10) , Indian Institute of Technology Roorkee june-2010, pp.1-6

[12] Wang Wei, "A novel bridgeless buck-boost PFC converter" Power Electronics Specialists Conference, 2008. PESC 2008. IEEE Page(s): 1304 – 1308

[13] Grigore, V,"High power factor rectifier based on buck converter operating in discontinuous capacitor voltage mode" Power Electronics, IEEE Transactions on (Volume:15 , Issue: 6) 2000 Page(s): 1241 - 1249

[14] Z. S. Chen, J. G. Hu, and W. Z. Gao, "Closed-loop analysis and control of a non-inverting buck-boost converter," *Intl. Journal of Control*, vol. 83, no. 11, pp. 2294-2307, Oct. 2010.

[15] Md. Nazmul Hasan, "Input Current Improvement of a Three Phase Rectifier by Ĉuk Regulator " M.SC. thesis EEE,BUET, 2010

[16] Md. Ashfanoor Kabir, Amina Hasan Abedin, Samia Islam and M. A. Choudhury "Ĉuk topology based single phase AC-DC converter wih low THD and high power factor " IECON , 7-10 Nov. 2011,**pp.**1256 – 1261

[17] Ahmad J. Sabzali, Esam H. Ismail, and Mustafa A. Al-Saffar and Abbas A. Fardoun, "A New Bridgeless PFC Sepic and Ĉuk Rectifiers with Low Conduction and Switching Losses" conference of Power electronics and Drive system, 2009, page 550-556

[18] K.Kavitha, Dr. Ebenezer Jeyakumar, "A Synchronous Ĉuk Converter Based Photovoltaic Energy System Design and Simulation" International Journal of Scientific and Research Publications, Volume 2, Issue 10, October 2012, ISSN, pp.2250-3153

[19] Alomgir Hossain, "AC voltage regulation by Ĉuk switch mode power supply", M.Sc thesis ,EEE BUET, May 2005

[20] R. W. Erickson, and D. Maksimovic, *Fundamentals of Power Electronics*. Boston: Kluwer Academic Publishers, 2004.

[21] Ned Mohan "First course on power electronics and drives" Minneapolis USA: MNPERE 2003

[22] *Muhammad H. Rashid* "Power Electronics Handbook 3rd edition devices circuits and application" Elsevier Inc, Burlington USA, 2011

[23] Bimal k. Bose "Modern Power electronics and AC drives" Prentice hall PTR, Upper saddle river 2002

[24] Jacob, T, "Maximum Power Point Tracking of hybrid PV and wind energy systems using a new converter topology" Green Technologies (ICGT), 2012, Page(s): 280 - 287

[25] S. Gang, et al. , "Maximum power point tracing of photovoltaic cells with OIF-Elman network," *in 29th Chinese ControlConf.*, July 2010, pp. 4880-4884.

[26] H. S. H. Chung, , et al., "A novel maximum power point tracking technique for solar panels using a SEPIC or Ĉuk converter," *IEEE Trans. Power Electron.*, vol. 18, no. 3, pp. 717-724, May 2003.

[27] Y. Fan, L. M. Ge, and W. Hua, "Multiple-input DC-DC converter for the thermoelectric-photovoltaic energy system in Hybrid Electric Vehicles," *in 2010 IEEE Vehicle Power and Propulsion Conf.*, Aug. 2010, pp. 35-40.

[28] P. Jose and N. Mohan, "A novel ZVS bidirectional Ĉuk converter for dual voltage systems in automobiles," *in The 29thAnnual Conf. of the IEEE Ind. Electron. Society*, Sept. 2003, pp.1-5.

[29] B.R. McGee and R.M. Nelms,"Powering solid state radar T/R module arrays from a fuel cell using an isolated Ĉuk converter," *19th Annual IEEE Applied Power Electron. Conf. and Expo.*, vol. 3, Sept. 2004, pp. 1853-1857.

[30] J. R. de Britto, et al., "A proposal of Led Lamp Driver for universal input using Ĉuk converter," *in IEEE Power Electron.Specialists Conf.*, June 2008, pp. 2640-2644

[31] A. A. Aboulnaga, and A. Emadi, "High performance bidirectional Ĉuk converter for telecommunication system," *26th AnnualIntl. Telecom. Energy Conf.*, Sept. 2004, pp. 182-189

[32] S. Janjornmanit, S. Yachiangkam, and A. Kaewsingha, "Energy harvesting from exercise bicycle," *in 7th Intl.Conf. on Power Electron. and Drive Systems*, Nov. 2007, pp. 1138-1140.

[33] S. Singh, and B. Singh, "A voltage controlled PFC Ĉuk converter based PMBLDCM drive for air-conditioners," *in 2010IEEE Ind. Appl. Society Annual Meeting*, Oct. 2010, pp. 1-6.

[34] F. A. Himmelstoss, and C. M. Walter, "A simple Ĉuk converter derivated two-quadrant DC motor controller," *in Intl. Symp. Power Electron. Electric. Drives Automa. and Motion*, June 2010, pp. 1108-1112

[35] J. L. Flores, J. L. B. Avalos, and C. A. B. Espinoza, "Passivity-based controller and online algebraic estimation of the load parameter of the DC-to-DC power converter Ĉuk type," *IEEE Latin America Trans.*, vol. 9, no.1, pp. 50-57, Mar. 2011.

[36] J. Neely, R. DeCarlo, and S. Pekarek, "Real-time model predictive control of the Ćuk converter," *in IEEE 12th Workshop on Ctrl. and Modeling for Power Electron.*, June 2010, pp. 1-8.

[37] A. Safari, and S. Mekhilef, "Simulation and hardware implementation of incremental conductance MPPT with direct control method using Ĉuk converter," *IEEE Trans. Ind. Electron.*, vol. 58, no. 4, pp. 1154-1161, April 2011

[38] C.L. Phillips and H.T. Nagle, "Digital control system analysis and design," Prentice-Hall, Inc., Englewood Cliffs, N.J., 1984.

[39] Microchip technology Inc, 16 bit digital signal controller performance documentation http://ww1.microchip.com/downloads/en/DeviceDoc/DS-70095K.pdf

[40] Celebi Arama Kriteri., DSP tutorial chapter 4 http://www.byclb.com/TR/Tutorials/dsp_advanced/ch4_1.htm

[41] N.Mohan,T.M.Undeland and W.P.Robbins, Power Electronics: Converters, Applications and Design,3'd ed., John Wiley, 2003.

[42] Microchip technology Inc., dsPIC33FJ12MC202, 16-bit digital signal controller datasheet. http://ww1.microchip.com/downloads/en/DeviceDoc/70265E.pdf

[43] Euro power components Ltd. lts6-np, Hall type current transducer, http://www.europowercomponents.com/media/uploads/lts6-np.pdf

Appendix 01

Determination of L1, C1, L2 and C2 parameters

```
clc
clear all
close all
Vo=400; %output regular voltage
fs=80e3; %switching frequency
Ts=1/fs; m=0;
Del_Vo=3;
for Po=200:100:1000 % Considering all the output powers
for Vi=282:5:367 % Considering all the input voltages peak value
    Vi=Vi*0.637; % considering Average voltage
m=m+1;
Ro=Vo^2/Po;
k=Vo/Vi;
D=k/(k+1);
Del_iL1=Po/Vi*0.2;
Del_iL2=Po/Vo*0.5;
L1(m)=Vi*D*Ts/(2*Del_iL1);
L2(m)=Vo*(1-D)*Ts/(2*Del_iL2);
Vs1=Vi/(1-D);
Del_VC1=Vs1*0.01;
C1(m)=Vo*D*Ts/(2*Ro*Del_VC1);
C2(m)=Vo*(1-D)*Ts^2/(8*Del_Vo*L2(m));
end
end
L1=max(L1)
L2=max(L2)
C1=max(C1)
C2=max(C2)
```

Appendix 02

Determination of transfer function H3(s)

```
Vin=325*0.637;
Vo=400;
D=Vo/(Vo+Vin);
Dp=1-D;
R=400;
Acm=[0 0 -Dp/L1 0;
     0 0 D/L2 -1/L2;
    Dp/C1 -D/C1 0 0;
    0 1/C2 0 -1/(R*C2)];
Bcm=[1/L1 Vin/(Dp*L1);0 Vin/(Dp*L2);0 -Vin*D/(R*C1*Dp*Dp);0 0];
Ccm=[0 0 0 -1];
Dcm=[0 0];
iu=2;
[NUM,DEN]=ss2tf(Acm,Bcm,Ccm,Dcm,iu);
H3=tf(NUM,DEN)
bode(H3)
%-----------------------------------------
%Current loop PI Controller |
%-----------------------------------------
hold on
KP=10.777;
TC0=1/(2*pi*80);
PI_controller_i1=tf(KP*[TC0 1],[TC0 0]);
H3_PI=PI_controller_i1*H3;
w={0.001, 50000};
bode(H3_PI,w)
grid on
```

Appendix 03

Determination of transfer function H6(s)

```
Cvm=[0 0 0 -1];
Dvm=[0 0];
iu=2;
[NUM,DEN]=ss2tf(Acm,Bcm,Cvm,Dvm,iu);
Gvd=tf(NUM,DEN)
Ccm=[1 0 0 0];
Dcm=[0 0];
iu=2;
[NUM,DEN]=ss2tf(Acm,Bcm,Ccm,Dcm,iu);
Gid=tf(NUM,DEN)
H6=Gvd/Gid
bode(H6)
%Voltage loop PI Controller
hold on
KP=27;
TC0=1/(2*pi*10);
PI_controller_V1=tf(KP*[TC0 1],[TC0 0]);
H6_PI=PI_controller_V1 *H6;
hold on
w={0.001, 50000}
bode(H6_PI,w)
grid on
```

Appendix 04

Determination of transfer function of overall system (Output voltage to Duty)

```
Acm=[0 0 -Dp/L1 0;
     0 0 D/L2 -1/L2;
    Dp/C1 -D/C1 0 0;
    0 1/C2 0 -1/(R*C2)]
Bcm=[1/L1 Vin/(Dp*L1);0 Vin/(Dp*L2);0 -Vin*D/(R*C1*Dp*Dp);0 0];
Cvm=[0 0 0 -1];
Dvm=[0 0];
iu=2;
[NUM,DEN]=ss2tf(Acm,Bcm,Cvm,Dvm,iu);
Gvd=tf(NUM,DEN)
bode(Gvd)
set(findall(gcf,'Type','text'),'FontSize',14)
% Set data line width and color
set(findall(gcf,'Type','line'),'LineWidth',2,'Color','black')
% Set axes tick label font size, color, and line width
set(findall(gcf,'Type','axes'),'FontSize',11,'LineWidth',2,'XColor','black','YColor','black')
hold on
GVI_update_temp= H6_PI* H3_PI
w={0.001, 50000}
bode(GVI_update_temp,w)
set(findall(gcf,'Type','text'),'FontSize',14)
% Set data line width and color
set(findall(gcf,'Type','line'),'LineWidth',2,'Color','black')
% Set axes tick label font size, color, and line width
set(findall(gcf,'Type','axes'),'FontSize',11,'LineWidth',2,'XColor','black','YColor','black')
```

Appendix 05

```
//*************************** Main.c file*****************************//
#include "Header01.h"
//*****************************************************************
_FPOR( PWMPIN_ON & HPOL_ON & LPOL_ON  & ALTI2C_OFF & FPWRT_PWR128)
_FWDT( FWDTEN_OFF & WINDIS_OFF)
_FOSC( FCKSM_CSECMD & IOL1WAY_OFF & POSCMD_NONE )
_FOSCSEL( FNOSC_FRCPLL )
_FGS( GSS_OFF & GCP_OFF & GWRP_OFF )
_FBS( BWRP_WRPROTECT_OFF )
//*****************************************************************
volatile unsigned int Avg_Vac;
volatile unsigned int VpiCount=0,IrefCount=0;
volatile unsigned int Start_OnDelay = 150;
volatile unsigned int Counting_sample = SamplingFreq/(2*SupplyFrequency);
volatile unsigned int VacLow,VacHigh = 0x008E;
volatile int VacHighMinimum = 0x0024, SampleCountMin = 310;
//*****************************************************************
int main ( void )
{
        PLLFBD=2;
        CLKDIVbits.PLLPOST=0;
        CLKDIVbits.PLLPRE=0;
        initialisePorts();
        initialisePWM2();
        initialiseADC();
        IFS0bits.AD1IF = 0;
        IEC0bits.AD1IE = 1;
        P2TCONbits.PTEN = 1;
while(1);
}
```

```
//*****************************************************************
void Delay(void)
{
   Start_OnDelay--;
   VsumAndFreq();
}
//*****************************************************************
void initialisePorts( void )
{
      LATB  = 0x0000;
         TRISBbits.TRISB0 = 1;
         TRISBbits.TRISB1 = 1;
         TRISBbits.TRISB2 = 1;
         TRISBbits.TRISB8 = 0;
         TRISBbits.TRISB15 = 0;
}
//*****************************************************************
void __attribute__((interrupt , auto_psv)) _ADC1Interrupt(void)
{
   IFS0bits.AD1IF = 0;
   if(Start_OnDelay != 0)
   {
      Delay();
   }
   else
   {
                  LATBbits.LATB15 = 1;
                  Voltage_PI_control();
                  Iac_Ref();
                    Current_PI_control();
                  VsumAndFreq();
```

```
                LATBbits.LATB15 = 0;
    }
}
//*****************************************************************
void initialiseADC(void)
{
    AD1CON1 = 0;
    AD1CON1bits.FORM = 3;
    AD1CON1bits.AD12B = 0;
    AD1CON1bits.SSRC = 5;
    AD1CON1bits.SIMSAM = 1;
    AD1CON1bits.ASAM = 1;
    AD1CON2 = 0;
    AD1CON2bits.CHPS = 3;
    AD1CON2bits.SMPI = 1;
    AD1CON2bits.CSCNA = 1;
    AD1CON3bits.SAMC = 8;
    AD1CON3bits.ADCS = 4;
    AD1CHS0bits.CH0NA = 0;
    AD1CHS123bits.CH123NA = 0;
    AD1CHS123bits.CH123SA = 1;
    AD1PCFGL = 0xFFFF;
    AD1PCFGLbits.PCFG2 = 0;
    AD1PCFGLbits.PCFG3 = 0;
    AD1PCFGLbits.PCFG4 = 0;
    AD1PCFGLbits.PCFG5 = 0;
    AD1CSSL = 0;
    AD1CSSLbits.CSS2 = 1;
    AD1CSSLbits.CSS4 = 1;
    AD1CON1bits.ADON = 1;
}
```

```
void initialisePWM2(void)
{
        P2TCON = 0x0000;
        P2TMR = 0x0000;
        P2TPER = PWM_PERIOD;
        P2SECMPbits.SEVTCMP = PWM_PERIOD;
        PWM2CON1bits.PMOD1 = 1;
        PWM2CON1bits.PEN1H = 1;
        PWM2CON2 = 0x0000;
        PWM2CON2bits.SEVOPS = 0;
        PWM2CON2bits.IUE = 1;
        P2OVDCONbits.POVD1H = 1;
        P2DC1 = 0;
        IFS4bits.PWM2IF = 0;
        IEC4bits.PWM2IE = 0;
}
void __attribute__((__interrupt__ , auto_psv)) _MPWM2Interrupt(void)
{
IFS4bits.PWM2IF = 0;
}
void __attribute__((__interrupt__ , auto_psv)) _MathError(void)
{
      while(1);
}
void __attribute__((__interrupt__ , auto_psv)) _AddressError(void)
{
   while(1);
}
void __attribute__((__interrupt__ , auto_psv)) _StackError(void)
{
   while(1);
```

```
}
//*****************************Main.S file************************//
.include "inc/ Header01.inc"
.include "inc/ propotional_integral.inc"
.equiv flag1, 0x0000
.equiv flag2, 0x0001
.equiv flag3, 0x0002
.data
VPI_High:                .word   0x0000
IPI_Low:                 .word   0x0000
IPI_High:                .word   0x0000
Vdc:                     .word   0x0000
VPI_Low:                 .word   0x0000
flag:                    .word   0x0007
.bss
VacHighTemp_sum:  .space  2
AvgVacSqr:        .space  2
Ref_Vdc:          .space  2
VdcSoftStart:     .space  2
pfcVoltBA:    .space  30
pfcCurrBA:    .space  30
Sam_Count_temp:   .space  2
Vac_sum_temp:            .space  4
Vac:              .space  2
VacdivideVavgSqr:        .space  2
Iac:              .space  2
IacRef:           .space  2
; ========================================
.global Vdc
.global AvgVacSqr
.global VPI_High
```

```
.global VPI_Low
.global IPI_High
.global IPI_Low
.global IacRef
.global Vac
.global Vdc
.global Iac
.global VacdivideVavgSqr
.global VdcSoftStart
.global Ref_Vdc
.global Vavg_min
; ======================================
.equ MD_PFC, 866
.equ MR_volt, 200
.equ V_kp_volt, 30000
.equ AdcConvFactor, 0x8000
.equ SoftStartIncrement, 50
.equ Km1, 2
.equ V_ki_volt, 1600
.equ I_kp_PFC,          1400
.equ I_ki_PFC,          7000
.equ I_kc_PFC,          50
.equ I_outMa_PFC, 32767
.equ I_outMi_PFC, 800
.equ Vavg_min,3000
.equ DutyConversionFactor, 40
.equ V_kc_volt, 107
.equ V_outMa_PFC, 32767
.equ V_outMi_PFC,    0
.equ Ref_Vdc, 28736
.equ Km2, 1
```

```
.equ Pfc_Duty,                P2DC1
.section  .text
.global   _VsumAndFreq
.global   VsumAndFreq
_VsumAndFreq:
VsumAndFreq:
push.d w0
push.d w2
push.d w4
push.d w6
push.d w8
                      mov.w ADCBUF4, w0
                      mov.w #AdcConvFactor,w1
                      xor w0,w1,w0
                      lsr w0,#6,w0
compare:              mov.w #MR_volt, w2
                      cp w0,w2
                      bra N, tempsum
                      btss flag,#flag1
                      btsc flag,#flag2
                      bra case2
case1:                mov.w Sam_Count_temp, w0
                      mov.w _SampleCountMin, w1
                      cp w0,w1
                      bra N, SaturateMinCount
                      bra DontSaturate
SaturateMinCount:     mov.w _SampleCountMin, w0
                      bra continueSumVac
DontSaturate:         mov.w Sam_Count_temp, w0
continueSumVac:       mov.w w0,_Counting_sample
                      mov #Vac_sum_temp, w0
```

```
                    mov.d [w0],w2
                    mov w3,ACCAH
                    mov w2,ACCAL
                    sftac A,#-5
                    mov.w ACCAL,w1
                    mov.w ACCAH,w0
                    mov.w w1, _VacLow
                    mov.w w0, VacHighTemp_sum
                    mov.w _VacHighMinimum, w1
                    cp w0,w1
                    bra N, Sat_Min_Sum
                    bra not_Sat_Sum
Sat_Min_Sum:        mov.w _VacHighMinimum, w0
                    bra Loop_continue
not_Sat_Sum: mov.w VacHighTemp_sum,w0
Loop_continue:              mov.w w0,_VacHigh
                    bset flag, #flag1
                    bset flag, #flag2
                    clr w0
                    mov.w w0,Sam_Count_temp
                    clr A
                    clr w2
                    clr w3
                    mov.w #Vac_sum_temp, w4
                    mov.d w2, [w4]
                    bra outofloop
case2:              mov.w ADCBUF4, w0
                    mov.w #AdcConvFactor,w1
                    xor w0,w1,w0
                    lsr w0,#6,w0
                    mov.w Sam_Count_temp, w6
```

```
                mov.w  w0,w8
                mov #Vac_sum_temp, w0
                mov.d [w0],w2
                add w8,w2,w2
                addc #0,w3
                inc w6,w6
                mov.w w6, Sam_Count_temp
                mov.w #Vac_sum_temp, w4
                mov.d w2, [w4]
                bclr flag, #flag1
                bra outofloop
tempsum:        btsc flag, #flag1
                bra outofloop
                mov.w ADCBUF4, w0
                mov.w #AdcConvFactor,w1
                xor w0,w1,w0
                lsr w0,#6,w0
                mov.w w0,w8
                mov.w Sam_Count_temp, w6
                mov #Vac_sum_temp, w0
                mov.d [w0],w2
                add w8,w2,w2
                addc #0,w3
                inc w6,w6
                mov.w w6, Sam_Count_temp
                mov.w #Vac_sum_temp, w4
                mov.d w2, [w4]
                bclr flag, #flag2
outofloop:
pop.d w8
pop.d w6
```

```
pop.d w4
pop.d w2
pop.d w0
return
.global   _Iac_Ref
.global   Iac_Ref
_Iac_Ref:
Iac_Ref:
push.d w0
push.d w2
push.d w4
push.d w6
push.d w8
                    mov.w _VacLow, w0
                    mov.w _VacHigh,w1
                    mov.w _Counting_sample, w3
                    repeat #17
                    div.ud w0,w3
                    mov  w0,_Avg_Vac
Vavgcalc:           mov      w0,w4
                    mpy     w4*w4,A
                    mov     ACCAH,w3
                    mov w3,AvgVacSqr
                    mov.w ADCBUF4, w0
                    mov.w #AdcConvFactor,w1
                    xor w0,w1,w0
                    lsr w0,#1,w0
                    mov     w0,Vac
                    mov Vac, w0
                    lac       w0,A
                    sftac     A,#7
```

```
                    mov     ACCAL,w0
                    mov     ACCAH,w1
                    repeat #17
                    div.ud w0,w3
                    mov.w  w0,VacdivideVavgSqr
                    mov.w  VacdivideVavgSqr,w4
                    mov.w VPI_High, w5
                    mpy     w4*w5,A
                    mov.w ACCAH, w0
                    mov.w w0,w2
                    sl w2,#Km1,w2
                    sl w0,#Km2,w0
                    add w0,w2,w0
                    mov.w w0, IacRef
pop.d w8
pop.d w6
pop.d w4
pop.d w2
pop.d w0
return
.global   _Voltage_PI_control
.global   Voltage_PI_control
_Voltage_PI_control:
Voltage_PI_control:
push.d w0
push.d w2
push.d w4
push.d w6
push.d w8
push.d w10
                    mov.w  ADCBUF0, w0
```

```
                mov.w #AdcConvFactor,w1
                xor w0,w1,w0
                lsr w0,#1,w0
                mov.w  w0,Vdc
                btsc flag,#flag3
                mov.w w0, VdcSoftStart
                bclr flag,#flag3
                mov.w VdcSoftStart, w0
                mov.w #Ref_Vdc, w1
                cp w0,w1
                bra GE, SteadyState
                mov.w #SoftStartIncrement, w0
                mov.w VdcSoftStart, w1
                add w0,w1,w0
                mov.w w0, VdcSoftStart
                mov.w w0, Ref_Vdc
                bra PIController
SteadyState:    mov.w #Ref_Vdc, w0
                mov.w w0, Ref_Vdc
PIController:   mov.w #V_kp_volt, w0
                mov.w w0, pfcVoltBA+PI_qKp
                mov.w #V_ki_volt, w0
                mov.w w0, pfcVoltBA+PI_qKi
                mov.w #V_kc_volt, w0
                mov.w w0, pfcVoltBA+PI_qKc
                mov.w #V_outMa_PFC, w0
                mov.w w0, pfcVoltBA+PI_qOutMax
                mov.w #V_outMi_PFC, w0
                mov.w w0, pfcVoltBA+PI_qOutMin
                mov.w Ref_Vdc, w0
                mov.w w0, pfcVoltBA+PI_qInRef
```

```
                    mov.w Vdc, w0
                    mov.w w0, pfcVoltBA+PI_qInMeas
                    mov.w #pfcVoltBA, w0
                    call _Calculation_PI
                    mov.w pfcVoltBA+PI_qOut, w1
                    mov.w w1,VPI_High
back:
pop.d w10
pop.d w8
pop.d w6
pop.d w4
pop.d w2
pop.d w0
return
.global   _Current_PI_control
.global   Current_PI_control
_Current_PI_control:
Current_PI_control:
push.d w0
push.d w2
push.d w4
push.d w6
push.d w8
push.d w10
mov.w  ADCBUF1, w0
mov.w w0, Iac
clr w1
btss w0, #15
bra next
mov.w  w1,Iac
next:               mov.w #I_kp_PFC, w0
```

```
                mov.w w0,pfcCurrBA+PI_qKp
                mov.w #I_ki_PFC, w0
                mov.w w0,pfcCurrBA+PI_qKi
                mov.w #I_kc_PFC, w0
                mov.w w0,pfcCurrBA+PI_qKc
                mov.w #I_outMa_PFC, w0
                mov.w w0,pfcCurrBA+PI_qOutMax
                mov.w #I_outMi_PFC, w0
                mov.w w0,pfcCurrBA+PI_qOutMin
                mov.w IacRef, w0
                mov.w w0,pfcCurrBA+PI_qInRef
                mov.w Iac, w0
                sl w0,#1,w0
                mov.w w0,pfcCurrBA+PI_qInMeas
                mov.w #pfcCurrBA, w0
                call _Calculation_PI
                mov.w pfcCurrBA+PI_qOut, w0
                mov.w w0,w2
                mov.w #DutyConversionFactor, w3
                repeat #17
                div.u w2,w3
                mov.w w0,IPI_High
                mov.w #MD_PFC, w1
                cp w1,w0
                bra LE, saturatePfc
                mov.w IPI_High, w0
                mov.w w0, Pfc_Duty
                bra goback
saturatePfc:    mov.w #MD_PFC, w1
                mov.w w1, Pfc_Duty
goback:
```

```
pop.d w10
pop.d w8
pop.d w6
pop.d w4
pop.d w2
pop.d w0
return
//********************** propotional_integral.s file*****************//
.include "inc/ Header01.inc"
.include "inc/ propotional_integral.inc"
        .equ BaseW0,   w0
        .equ OutW1,    w1
        .equ SumLW2,   w2
        .equ SumHW3,   w3
        .equ ErrW4,    w4
        .equ WorkW5,   w5
        .equ UnlimitW6,w6
        .equ WorkW7,   w7
        .section  .text
        .global   _initialize_PI
        .global   initialize_PI
_initialize_PI:
initialize_PI:
        mov.w     w1,[BaseW0+PI_qOut]
        return
        .global   _Calculation_PI
        .global   Calculation_PI
_Calculation_PI:
Calculation_PI:
                mov.w     [BaseW0+PI_qInMeas],WorkW5
                mov.w     [BaseW0+PI_qInRef],WorkW7
```

```
                sub.w           WorkW7,WorkW5,ErrW4
        lac     [++BaseW0],B
        mov.w   [--BaseW0],WorkW5
        mov.w   WorkW5,ACCBLL
        mov.w   [BaseW0+PI_qKp],WorkW5
        mpy     ErrW4*WorkW5,A
        sftac   A,#-NKo
        add     A
        sac     A,UnlimitW6
        mov.w   [BaseW0+PI_qOutMax],OutW1
        cp      UnlimitW6,OutW1
        bra     GT,jPI5
        mov.w   [BaseW0+PI_qOutMin],OutW1
        cp      UnlimitW6,OutW1
        bra     LE,jPI5
        mov.w   UnlimitW6,OutW1
jPI5:
        mov.w   OutW1,[BaseW0+PI_qOut]
        mov.w   [BaseW0+PI_qKi],WorkW5
        mpy     ErrW4*WorkW5,A
        sub.w   UnlimitW6,OutW1,UnlimitW6
        mov.w   [BaseW0+PI_qKc],WorkW5
        msc     WorkW5*UnlimitW6,A
        add     A
        sac     A,[++BaseW0]
        mov.w   ACCALL,WorkW5
        mov.w   WorkW5,[--BaseW0]
        return
        .end

//****************** Header01.h file***************//
```

```
#include "p33fj12mc202.h"
#define PWM_PERIOD        499
#define SupplyFrequency       50
#define SamplingFreq20000
void Delay(void); void initialisePortsPorts(void);
void initialisePortsADC(void);
void initialisePortsPWM2(void);
void Voltage_PI_control(void);
void Iac_Ref(void);
void Current_PI_control(void);
void VsumAndFreq(void);
void __attribute__((__interrupt__ , no_auto_psv)) _ADC1Interrupt(void);
void __attribute__((__interrupt__ , no_auto_psv)) _MPWM2Interrupt(void);
void __attribute__((__interrupt__ , no_auto_psv)) _MathError(void);
void __attribute__((__interrupt__ , no_auto_psv)) _AddressError(void);
void __attribute__((__interrupt__ , no_auto_psv)) _StackError(void);
//****************** propotional_integral.inc file***************//
      .equ PI_qdSumL,0
      .equ PI_qdSumH,2
      .equ PI_qKp,4
      .equ PI_qKi,6
      .equ PI_qKc,8
      .equ PI_qOutMax,10
      .equ PI_qOutMin,12
      .equ PI_qInRef,14
      .equ PI_qInMeas,16
      .equ PI_qOut,18
      .equ NKo,4
//****************** Header01.inc file***************//
.include "p33fj12mc202.inc"
```

END