

STARS


University of Central Florida
STARS

Electronic Theses and Dissertations, 2004-2019

2006

Dsp Implementation Of Dc Voltage Regulation Using Adaptive Control For 200 Kw 62000 Rpm Induction Generator

Othman Elkhomri
University of Central Florida

 Part of the [Electrical and Electronics Commons](#)
Find similar works at: <https://stars.library.ucf.edu/etd>
University of Central Florida Libraries <http://library.ucf.edu>

This Masters Thesis (Open Access) is brought to you for free and open access by STARS. It has been accepted for inclusion in Electronic Theses and Dissertations, 2004-2019 by an authorized administrator of STARS. For more information, please contact STARS@ucf.edu.

STARS Citation

Elkhomri, Othman, "Dsp Implementation Of Dc Voltage Regulation Using Adaptive Control For 200 Kw 62000 Rpm Induction Generator" (2006). *Electronic Theses and Dissertations, 2004-2019*. 859.
<https://stars.library.ucf.edu/etd/859>



**DSP IMPLEMENTATION OF DC VOLTAGE REGULATION USING
ADAPTIVE CONTROL FOR 200 KW 62000 RPM
INDUCTION GENERATOR**

by

OTHMAN ELKHOMRI
B.S. University of Central Florida, 2003

A thesis submitted in partial fulfillment of the requirements
for the degree of Master of Science
in the School of Electrical Engineering and Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Spring Term
2006

ABSTRACT

The thesis discusses the development of closed loop system to control the DC voltage for 200 kW induction generator rated at a speed of 62000 RPM under different load conditions. The voltage regulation has been implemented using PI controller. A gain scheduling control algorithm has been developed to select the appropriate controller gains with respect to the generator load. Further, a relationship between the generator loads and the controller gains has been established. This relationship has been modeled using adaptive control technique to vary the gains automatically at any load condition.

The adaptive control technique has been successfully generalized for real time DSP implementation to regulate the DC voltage for high speed induction generators rated from 5 kW to 200 kW.

ACKNOWLEDGMENTS

I am grateful to my supervisor and co-advisor, Jay Vaidya, President of Electrodynamics Associates, for guiding me in this project over the past three years. He has provided me with the knowledge, the tools, and the motivation that I will need to succeed in the future. I also would like to thank him for the tremendous financial support during my master's degree.

I would like to thank my co-advisor, Prof. Thomas Wu, for his guidance and inspiration throughout my master's degree, and his help putting this work together. Also, I would like to give many thanks to Prof. Wasfy Mikhael for his time reviewing my work.

I also would like to thank Prof. Jeffrey Lang from Massachusetts Institute of Technology for guiding me to resolve issues in the controller. I am grateful to my colleagues and specially, Al Spears, for being patient in long hours of testing in the lab.

I would like to extend special thanks to my friends Dr. Liping Zheng and Jie Chen who provided both friendship and assistance during the course of this study.

This work was supported by the U.S Air Force Research Laboratory under SBIR phase II program (contract number: F33615-00-C-2018).

TABLE OF CONTENTS

LIST OF FIGURES	VI
LIST OF TABLES	VIII
LIST OF SYMBOLS	IX
LIST OF ACRONYMS	XI
CHAPTER ONE: INTRODUCTION.....	1
1.1 Background.....	1
1.2 Problem Statement.....	3
1.3 Thesis Overview	4
CHAPTER TWO: GENERATOR CLOSED LOOP CONTROL	5
2.1 Introduction.....	5
2.2 Induction Machine Dynamic Modeling.....	5
2.3 Field Orientation Condition	7
2.4 Field Oriented Control Scheme	8
2.5 200 kW Induction Generator Closed Loop Algorithm.....	12
2.6 Space Vector Modulation for Three Phase Voltage Source Inverter.....	19
2.6.1 Three-Leg Voltage Source Inverter	19
2.6.2 Voltage Space Vector	20
2.6.1 Space Vector Modulation	21
2.6 Digital Controller Implementation Issues.....	26
2.6.1 Finite Word Length Effects	26
2.6.2 Closed Loop Algorithm Sampling Rate.....	27
2.6.3 Software Development.....	28

CHAPTER THREE: GAIN SCHEDULING METHOD.....	29
3.1 Gain Scheduling For Specific Load Conditions	29
3.2 Search of Proportional & Integral Gain Using Ziegler-Nichols Tuning Method	30
CHAPTER FOUR: ADAPTIVE METHOD	35
4.1 Adaptive Method For Any Load Condition.....	35
4.2 First Order Polynomial	39
4.3 Second Order Polynomial	40
4.4 Results.....	41
CHAPTER FIVE: CONCLUSION.....	43
APPENDIX A: SYSTEM CLOSED LOOP SOFTWARE	44
System Reference.....	44
System Interrupt Configuration	46
System Initialization	46
System Infinite Loop.....	46
System Interrupt Save & Restore.....	46
System Algorithm	47
LIST OF REFERENCES.....	53

LIST OF FIGURES

Figure 1: Induction machine dynamic model	7
Figure 2: Simplified induction machine dynamic model.....	8
Figure 3: Relationship between $\alpha\beta$ axis and dq axis.....	9
Figure 4: Relationship Between $\alpha\beta$ and abc currents.....	10
Figure 5: I_{DQ} from the current loop.....	13
Figure 6: Phase a current.....	13
Figure 7: Phase c current.....	14
Figure 8: α current.....	14
Figure 9: β current.....	15
Figure 10: Filtered I_{DQ}	16
Figure 11: DC voltage control loop	16
Figure 12: I_{DQ} generated from the voltage loop.....	17
Figure 13: Speed loop.....	17
Figure 14: Block diagram for closed loop control algorithm.	18
Figure 15: Three-leg voltage source inverter.....	20
Figure 16: Space vectors.....	21
Figure 17: Block diagram of space vector	23
Figure 18: Control signal	24
Figure 19: Generation of the PWM signals	24
Figure 20: Three phase PWM signals.....	25

Figure 21: Output space vector trajectory.....	26
Figure 22: Closed algorithm sampling rate.....	27
Figure 23: Classical proportional and integral controller	29
Figure 24: Generator closed loop algorithm.....	32
Figure 25: DC voltage, DC current and space vector command versus time	33
Figure 26: K_p versus power.....	36
Figure 27: First and second order polynomial	38
Figure 28: First order fit.....	39
Figure 29: Second order fit	40
Figure 30: DC current, DC voltage Versus Time	42

LIST OF TABLES

Table 1: SV PWM switching states 22

Table 2: Kp & Ki under different loads condition..... 31

Table 3: Four load condition from 0 to 5.1 kW 41

LIST OF SYMBOLS

ω_e	electrically synchronous angular velocity
i_{ds}^e	d-axis stator current in the synchronously rotating reference frame
λ_{dr}^e	d-axis rotor flux in the synchronously rotating reference frame
λ_{qr}^e	q-axis rotor flux in the synchronously rotating reference frame
v_{qs}^e	q-axis stator voltage in the synchronously rotating reference frame
i_{qs}^e	q-axis stator current in the synchronously rotating reference frame
v_{ds}^e	d-axis stator voltage in the synchronously rotating reference frame
ω_{sl}	electrically slip angular velocity
L_m	magnetizing inductance
T_e	electromagnetic torque
λ_{qs}^e	q-axis stator current in the synchronously rotating reference frame
λ_{ds}^e	d-axis rotor flux in the synchronously rotating reference frame
ω_{sl}	electrically slip angular velocity
ω_e	electrically synchronous angular velocity
ω_r	electrically rotor angular velocity
f	supply frequency
K_i	integral gain
K_p	proportional gain
K_T	torque constant

L_m	mutual inductance
L_r	rotor inductance
P	number of poles
R_r	rotor resistance
Θ_r	angular position (phase) of the vector of rotor flux in the stator reference frame
τ_r	rotor time constant
ω	supply radian frequency
ω_r	slip

LIST OF ACRONYMS

AC	Alternating Current
ADC	Analog to Digital Converter
PI	Proportional and Integral (controller)
DC	Direct Current
DSP	Digital Signal Processor
EMF	Electro Motive Force
FOC	Field Oriented Control
PI	Proportional-Plus-Integral (controller)
SVM	Space Vector Modulation
TI	Texas Instruments
RPM	Revolutions per Minute

CHAPTER ONE: INTRODUCTION

1.1 Background

When an induction machine is driven from external prime mover at a speed greater than synchronous speed, it works as induction generator as long as reactive power is supplied. Basically, the excitation current is supplied to the armature winding from which it is induced into the short circuited squirrel cage secondary winding in the rotor.

At no load the induction generator can be excited provided with enough excitation current. DC voltage can stay constant as long as a constant speed is maintained. However, as the generator load is increased or the speed varies, the DC voltage decreases or become unstable respectively. In order to maintain a constant voltage, the excitation current needs to be adjusted during load variation from no load to full load. Using high power switching IGBT's, this issue can be solved. Induction generator can be controlled by adjusting the excitation current based on the load condition.

DC voltage control of induction generators is a topic that has been investigated by engineers and researchers for the last decade [1-4]. Many numerical methods have been proposed to predict the transient and the steady state performance of induction generators [2, 3, 4], most of them were unpractical for implementation. The complication was that the generator mathematical model is composed of linear differential equations with periodically time-varying inductance coefficients assuming rotor speed constant [4].

In 1965, Krause noted [8] that all known real transformations used in induction machine analysis could be described by one general transformation (Park transformation) which eliminates all time-varying inductance by referring the stator and rotor variables to a frame of

reference which may rotate at any angular velocity or remain stationary. The physical meaning of the Park transformation is to define a new set of stator and rotor variables in (dq) frame in terms of the actual winding variables (abc) in stationary frame. This method has been further developed to Field oriented control (FOC) schemes.

FOC schemes are implemented to operate the generator in the maximum torque conditions available and to decouple the maximum torque from the field under transient and steady state operation. The FOC algorithm transforms the three phase sinusoidal currents from the stationary reference frame to DC quantities in rotating reference frame. The rotating reference frame is composed of the direct current component and the quadrature current component which are to be orthogonal to each other. Typically, these currents are regulated by Classical Proportional and Integral (PI) controllers because of their simple implementation. However, PI controllers do not perform well when controlling high order non-linear dynamic plants - such as the high speed induction generators - due to the overshoot response and the output saturation when generator is loaded.

The focus of this work was the development and implementation of the closed loop control system for a high speed induction generator using LF 2407 Texas Instruments (TI) Digital Signal Processor (DSP). A motor closed loop controller was also developed to drive the generator. A gain scheduling algorithm was developed and applied to the PI DC voltage regulator to enhance the overall closed loop performance at three load conditions. This algorithm was successfully implemented and experimentally verified. Furthermore, an adaptive control algorithm was generalized and implemented in real-time to regulate DC voltage for any load condition.

1.2 Problem Statement

DC voltage regulation using PI controller with FOC algorithm determines the overall system performance for the generator. When a sudden step load is applied, the PI controller became unstable. The transient voltage response was not well controlled due to the characteristics of the PI controller. Two issues have been found: (1) overshoot response caused by a large error between the DC voltage command to the sensed DC voltage. (2) the voltage response between step loads is very slow due to the fixed proportional and integral gain controller. Furthermore, DC voltage regulation becomes unstable and causes the generator to de-excitation. The source of this issue is that PI controller gains cannot be set to satisfy both the overshoot and load variation simultaneously.

To overcome this issue, the proportional gains need adjustments with respect to the load variation as well as the overshoot response in real-time. A gain scheduling control algorithm has been developed. This algorithm represents a set of linear controllers, each of them designed for a specific load condition. Thus, when the generator is under a certain load condition, the algorithm control signal determines which linear controller to activate. The gain scheduling algorithm was implemented successfully to regulate the DC voltage at load conditions where the gains were predetermined. Later a relationship between the proportional, integral gains and the load conditions has been established. The relationship allowed the development of an adaptive control technique to vary the gains automatically at any load condition. The adaptive control technique has been tested from 0 to 5 kW.

1.3 Thesis Overview

Beginning with Chapter 2, the generator closed loop control is discussed in details including induction generator dynamic modeling, principle of the FOC conditions and schemes. Also, closed algorithm implementation including space vector modulation is discussed. Finally, digital control implementation issues using fixed point programming with the TI C2000 platform are mentioned. Chapter 3 discusses the development of the gain scheduling algorithm and how the proportional and integral gains have been obtained. Verification of this method was tested at 44000 RPM with a 48.6 kW load. Chapter 4 addresses the developed adaptive control algorithm including the curve fitting analysis. The adaptive control algorithm has been tested at 12000 RPM with a 5.1 kW load. Finally, Chapter 5 is a conclusion of the work.

CHAPTER TWO: GENERATOR CLOSED LOOP CONTROL

2.1 Introduction

As mentioned previously, in order to maintain a constant DC output voltage, the excitation current need to be adjusted accordingly to the generator load condition. For high performance control, closed loop algorithm using FOC are implemented to control the generator at various loads.

2.2 Induction Machine Dynamic Modeling

The mathematical model of the induction machine (IM) in the synchronously rotating reference frame when $w = w_e$ is shown as follow:

$$\frac{di_{qs}^e}{dt} = -\gamma_{qs}^e - \omega_e i_{ds}^e - \beta \omega_r \lambda_{dr}^e + \beta \alpha \lambda_{qr}^e + \beta_1 v_{qs}^e \quad (2.1)$$

$$\frac{di_{ds}^e}{dt} = -\gamma_{ds}^e + \omega_e i_{qs}^e + \beta \omega_r \lambda_{qr}^e + \beta \alpha \lambda_{dr}^e + \beta_1 v_{ds}^e \quad (2.2)$$

$$\frac{d\lambda_{qr}^e}{dt} = -\alpha \lambda_{qr}^e - \omega_{sl} \lambda_{dr}^e + \alpha L_m i_{qs}^e \quad (2.3)$$

$$\frac{d\lambda_{dr}^e}{dt} = -\alpha \lambda_{dr}^e + \omega_{sl} \lambda_{qr}^e + \alpha L_m i_{ds}^e \quad (2.4)$$

$$T_e = \frac{3}{2} \frac{p}{2} \frac{L_m}{L_r} (\lambda_{dr}^e i_{qs}^e - \lambda_{qr}^e i_{ds}^e) \quad (2.5)$$

where:

$$\lambda_{qs}^e = \left(\frac{L_s L_r - L_m^2}{L_r} \right) i_{qs}^e + \frac{L_m}{L_r} \lambda_{qr}^e \quad (2.6)$$

$$\lambda_{ds}^e = \left(\frac{L_s L_r - L_m^2}{L_r} \right) i_{ds}^e + \frac{L_m}{L_r} \lambda_{dr}^e \quad (2.7)$$

$$\omega_{sl} = \omega_e - \omega_r \quad (2.8)$$

$$\sigma = 1 - \frac{L_m^2}{L_s L_r} \quad (2.9)$$

$$\alpha = \frac{R_r}{L_r} \quad (2.10)$$

$$\beta = \frac{L_m}{\sigma L_s L_r} \quad (2.11)$$

$$\beta_1 = \frac{1}{\sigma L_s} \quad (2.12)$$

$$\gamma = \frac{L_m^2 R_r + L_r^2 R_s}{\sigma L_s L_r^2} \quad (2.13)$$

To visualize the induction machine dynamic model, a block diagram of the dynamic model has been drawn in Figure 1.

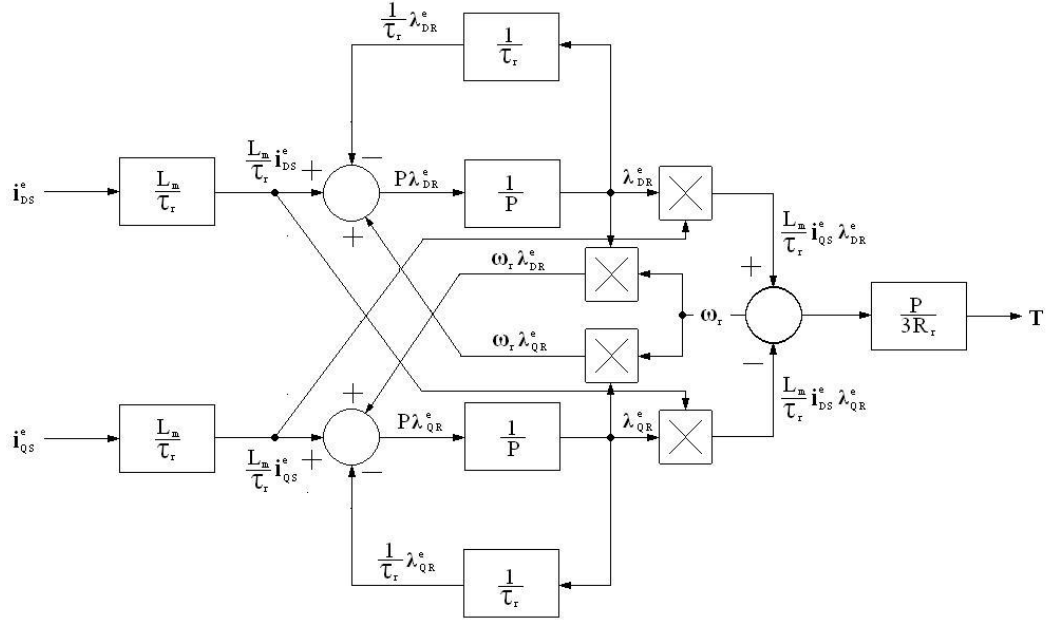


Figure 1: Induction machine dynamic model

The block diagram illustrates the source of difficulties in the control. Four multipliers make the model nonlinear. The FOC scheme represents a special case to simplify the dynamic model.

2.3 Field Orientation Condition

When the synchronously rotating reference frame is aligned with the rotor flux angle, d-axis angle will be defined on the total rotor flux angle. The condition is made when $\omega_e = \omega_{\lambda_r}$.

Then the q-axis rotor flux becomes zero, $\lambda_{qr}^e = 0$. This condition reduces the model to:

$$T = K_T \lambda_{dr}^e i_{qs}^e \quad (2.14)$$

where

$$K_T = \frac{P}{3R_r} \frac{L_m}{\tau_r} \quad (2.15)$$

So now when $\lambda_{QR} = 0$ and $\lambda_{DR} = \text{Constant}$ is a constant, the induction generator can be linearly controlled. Figure 2 illustrates the block diagram of the equation.

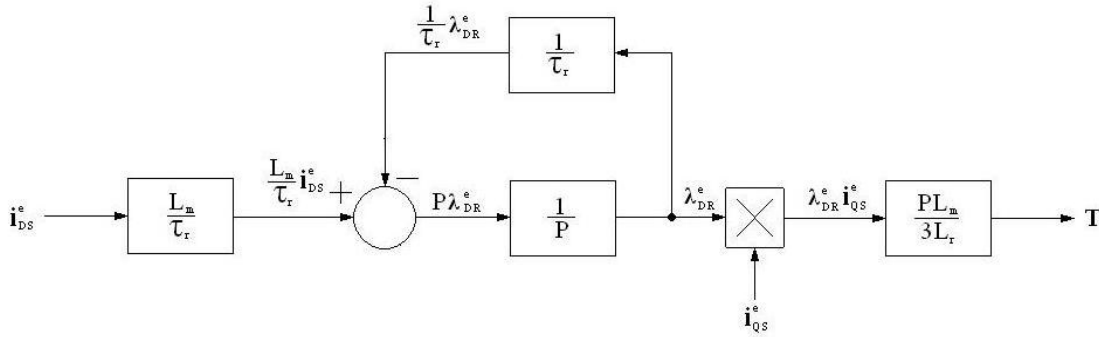


Figure 2: Simplified induction machine dynamic model

2.4 Field Oriented Control Scheme

The main goal of the FOC system is to independently control torque and flux of an AC generator by using the rotor flux angle to transform the sinusoidal variables (i.e., currents, voltages, flux, etc.) to the DC-type quantities. Such transformation is called Park transformation [1]. The Park Transformation converts vectors in balanced two-phase orthogonal stationary system into orthogonal rotating reference frame. The transformation equation is of the form:

$$[f_{dq}] = [T_{dq}(\theta_d)][f_{\alpha\beta}] \quad (2.16)$$

where variable f can be currents, voltages, or fluxes and θ_d is the angle between α -axis and d-axis as shown in Figure 3 **Figure 3**

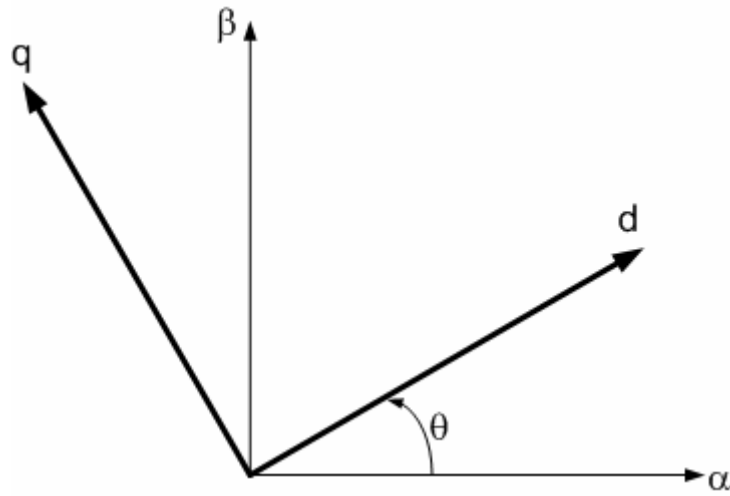


Figure 3: Relationship between $\alpha\beta$ axis and dq axis.

and the dq transformation matrix is defined as

$$[T_{dq}(\theta)] = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \quad (2.17)$$

For the transformation to occur correctly and the dq -axis currents become DC component, It is mandatory that α -axis and β -axis to be 90° phase shifted and no offset in the amplitude and also rotor flux angle to be correct. Otherwise, the vector control scheme fails. To obtain the two-phase orthogonal stationary system, another transformation, called Clarke Transformation [2], which is used as an intermediary between abc -axis stationary reference frame to dq -axis synchronously rotating reference frame. The Clarke Transformation is the method to transfer the three-phase into the two-phase in stationary coordinate system. In Figure 4, the three-phase variables are denoted as a , b and c and the two-phase variables are denoted as α and β .

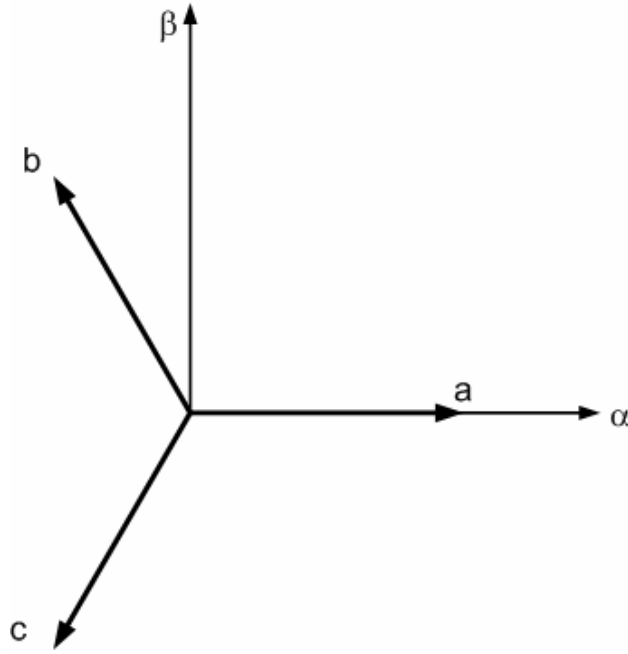


Figure 4: Relationship Between $\alpha\beta$ and abc currents.

A third variable known as the zero-sequence component is added to make the transformation to be bi-directional

$$[f_{\alpha\beta 0}] = [T_{\alpha\beta 0}][f_{abc}] \quad (2.18)$$

where variable f can be currents, voltages, or fluxes, and the transformation matrix is given by

$$T_{\alpha\beta 0} = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \quad (2.19)$$

As mentioned previously, The FOC defines conditions for maximum torque and the decoupling of torque control from the field control for both steady state and transient conditions. To operate in the maximum torque region, the rotor currents and the flux vectors must be maintained orthogonal to each other at all times.

However, to meet orthogonality during transient conditions, FOC special techniques are required. Two approaches are mainly implemented in induction generator control. First, the direct FOC scheme that is based on measuring the magnetic field using a sensor mechanically installed in the generator. Second, the Indirect FOC scheme that does not involve a direct measurement of the flux but the mechanical speed is sensed instead. The speed information is used to align the rotating and excitation reference frame with the rotor flux vector.

The 200 kW induction motor and generator closed loop control has been implemented using the indirect FOC techniques. Also a special case of the indirect FOC scheme [7] has been implemented which operates at maximum torque at all operation. The next section will discuss this scheme.

2.5 200 kW Induction Generator Closed Loop Algorithm

The closed loop control algorithm is based on selecting the D-Q axes so that the direct current and the quadrature current are equal. Equation 2.20 shows the developed torque as a result of $I_D = I_Q$.

$$T = K_T (I_{DQ})^2 \left(\frac{\tau_r \omega_s}{\tau_r^2 \omega_s^2 + 1} \right) \quad (2.20)$$

where

$$K_T = \text{Torque Constant, } \frac{Nm}{A^2}$$

$$I_{DQ} = \sqrt{2} I_D, \text{ A}$$

$$\tau_r = \text{Rotor time constant, s}$$

$$\omega_s = \text{Slip frequency, } \frac{rad}{s}$$

For maximum torque,

$$\tau_r = \frac{1}{\omega_s} \quad (2.21)$$

$$T = \frac{1}{2} K_T I_{DQ}^2 \quad (2.22)$$

Equation (2.22) summarizes the mechanism of the generator closed loop algorithm. The I_{DQ}^2 total current is generated from two loops. The voltage loop generates the reference and the current loop generates the feedback then compared and controlled by PI current regulator. The output of the PI regulator provides the duty ratio information to the three-phase inverter power switches. Figure 5 illustrates the I_{DQ} generated from the current loop.

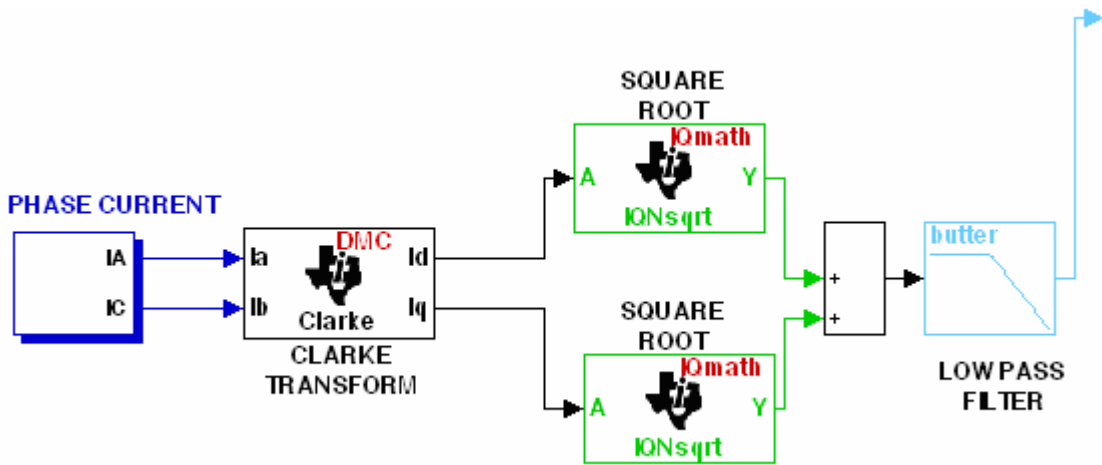


Figure 5: I_{DQ} from the current loop

Phase a and Phase c currents are sensed and Phase b current is calculated. The calculation of Phase b is based on equation 2.23

$$i_a + i_b + i_c = 0 \tag{2.23}$$

Figure 6 and 7 illustrate the actual Phase a and c currents with 240° phase shift respectively.

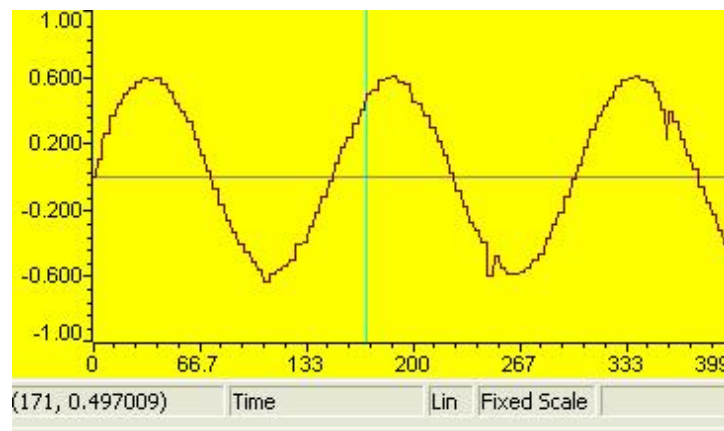


Figure 6: Phase a current

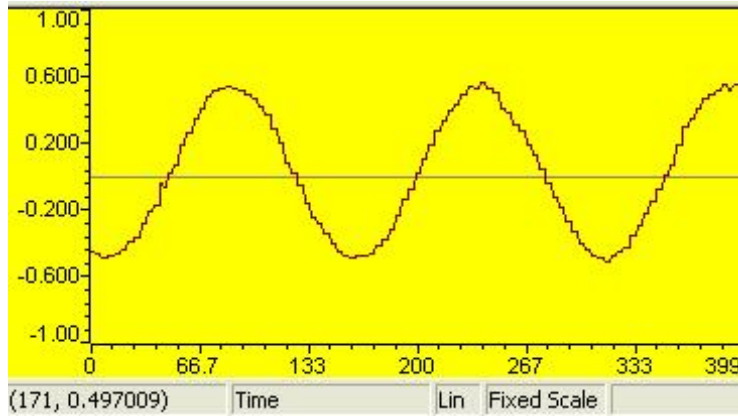


Figure 7: Phase c current

Then the Clarke Transformation is implemented to convert the phase currents into α and β currents which are 90° phase shifted. Figures 8 and 9 illustrate the actual α and β currents respectively.

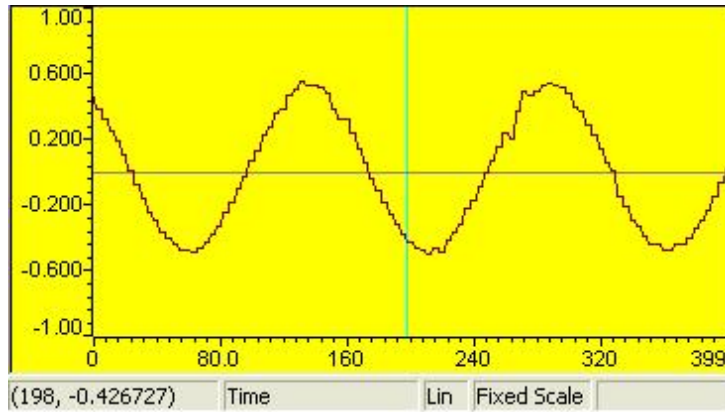


Figure 8: α current

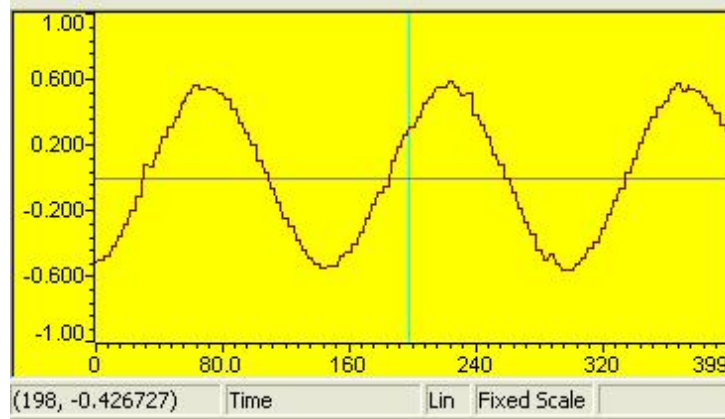


Figure 9: β current

Further, I_{DQ} is calculated by normalizing the output of the Clarke Transformation as:

$$I_{DQ} = \sqrt{(I_{\alpha}^2 + I_{\beta}^2)} \quad (2.24)$$

The square calculation I_{α} and I_{β} contributes into spikes in the I_{DQ} current. Furthermore, I_{DQ} current signal fidelity deteriorated and resulted into instability of the loop. A first order low pass digital filter has been implemented to remove unwanted data. Figure 10 shows the filtered I_{DQ} .

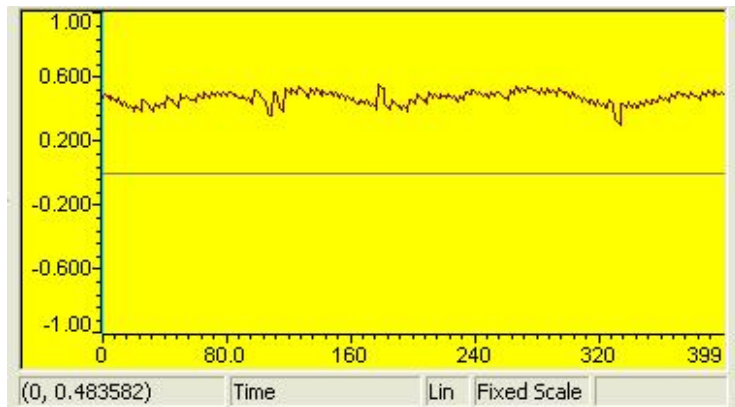


Figure 10: Filtered I_{DQ}

Secondly, the I_{DQ} reference generated by the voltage loop is illustrated in Figure 11.

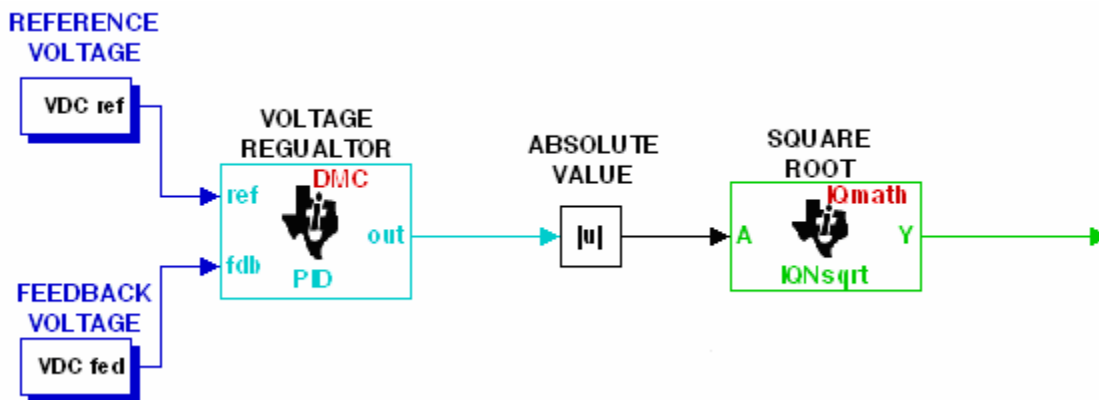


Figure 11: DC voltage control loop

Both the DC voltage command and DC voltage feedback are sensed and controlled by a PI regulator. The output of the PI regulator is the I_{DQ}^2 current that will act as the reference to the PI current regulator. The I_{DQ}^2 current is then taken through an absolute value and square root

function to obtain I_{DQ} . The absolute value function is necessary because when the generator is under load, the I_{DQ}^2 current may become negative in a short time and the square function will not work. Figure 12 shows I_{DQ} generated from the voltage loop.

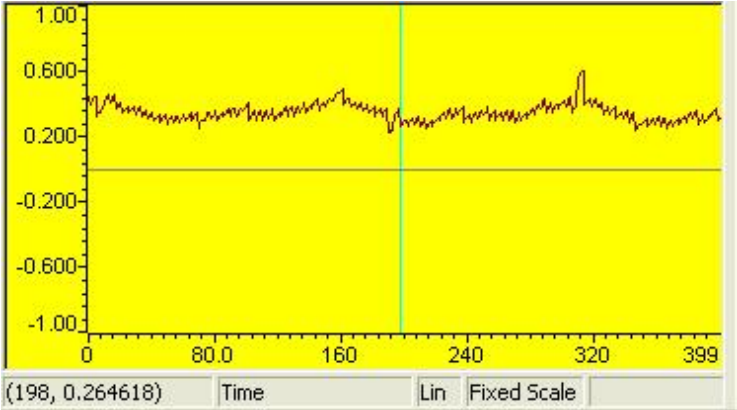


Figure 12: I_{DQ} generated from the voltage loop

The generator speed is sensed and the slip is subtracted from it. The generator slip needs to be always subtracted for the generator to operate. Figure 13 illustrates the speed loop.

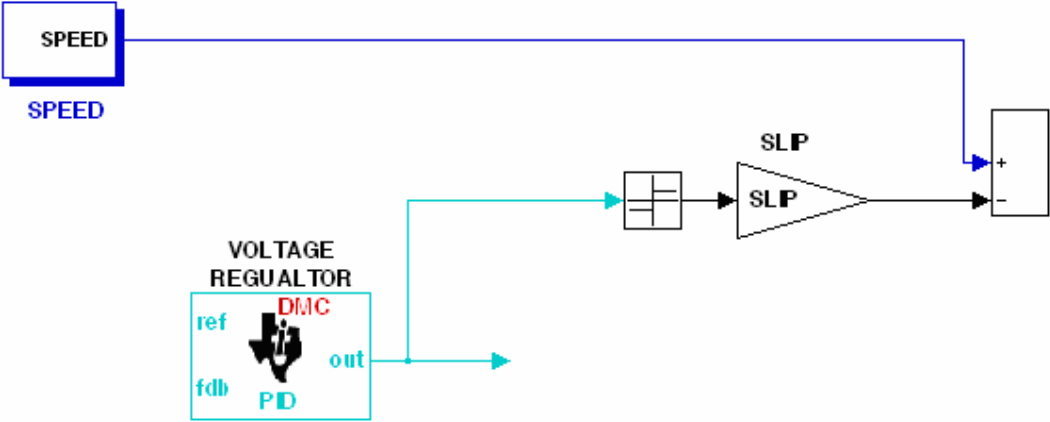


Figure 13: Speed loop

When the generator is under load, the I_{DQ}^2 current may become negative. The sign of the slip need change accordingly. Otherwise the I_{DQ}^2 current will saturate and results to instability of the loop. A sign function is implemented to control the slip sign. The sign function branches to two conditions:

- 1) If $I_{DQ}^2 > 0 \rightarrow$ slip subtracted
- 2) If $I_{DQ}^2 < 0 \rightarrow$ slip added

Figure 14 illustrates the 200 kW induction generator closed loop control algorithm.

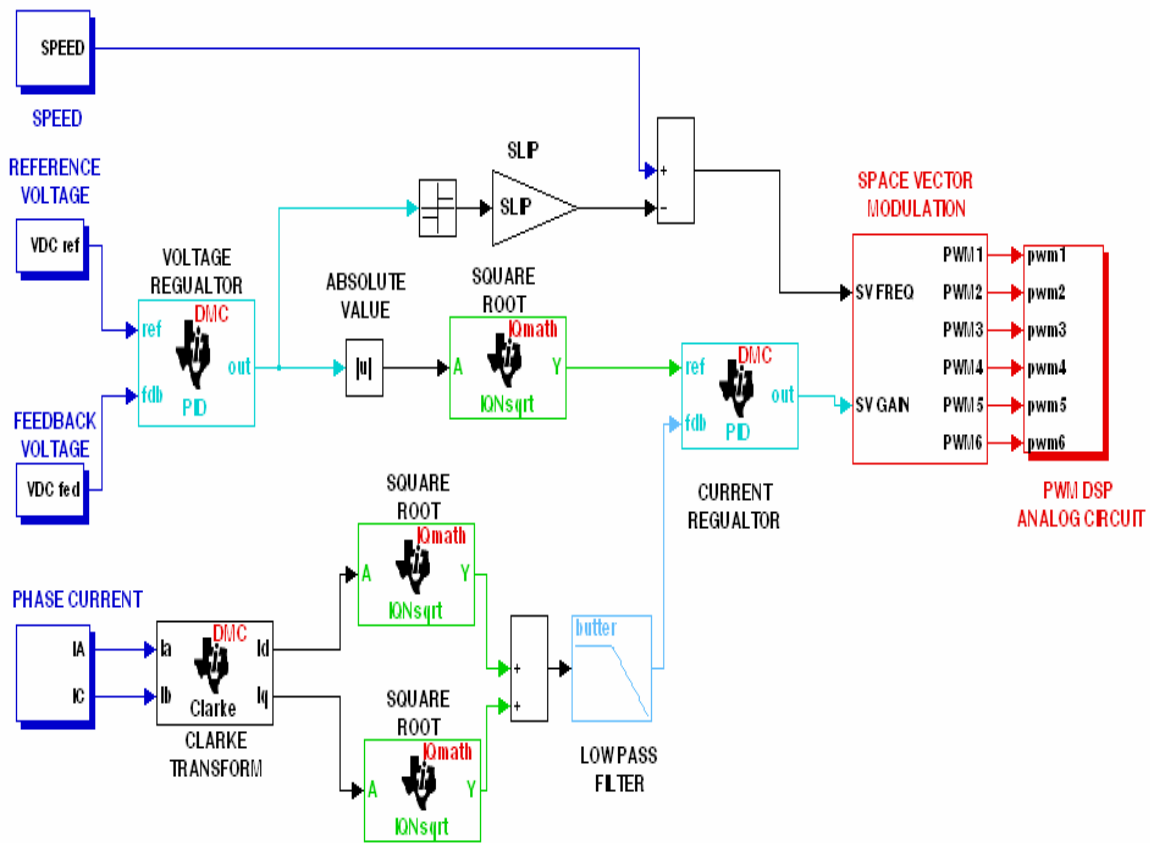


Figure 14: Block diagram for closed loop control algorithm.

The closed loop algorithm inputs are shown on the left side of Figure 14. The commanded voltage, measured generator bus voltage, measured shaft speed, and measured three phase ac generator currents. Total I_{DQ}^2 current is calculated from the current feedback loop. Then From the difference of the output voltage and the voltage command, the I_{DQ}^2 current command is then generated. Both currents are compared and based on the error; (PI) current regulator generates the necessary switching commands for the six switches in the three-phase full bridge inverter using space vector modulation technique.

2.6 Space Vector Modulation for Three Phase Voltage Source Inverter

One of the most widely used method to control AC output of power electronic converters is the pulse width modulation (PWM) technique, which varies the duty cycle of the converter switches at a high switching frequency to achieve the desired output voltage or current [3]. The combination of both the fast switching IGBT and the advanced DSP technologies, make it possible to improve systems efficiency. Several PWM techniques are available, the 200 kW generator inverter used the Space Vector PWM technique (SVPWM). SVPWM is an advanced, computation-intensive PWM method with superior performance characteristics. It has been shown that SVPWM generates less harmonic distortion in the output voltage or current in comparison with sine wave PWM and hysteresis control [9].

2.6.1 Three-Leg Voltage Source Inverter

The topology of a three-leg voltage source inverter is shown in Figure 15.

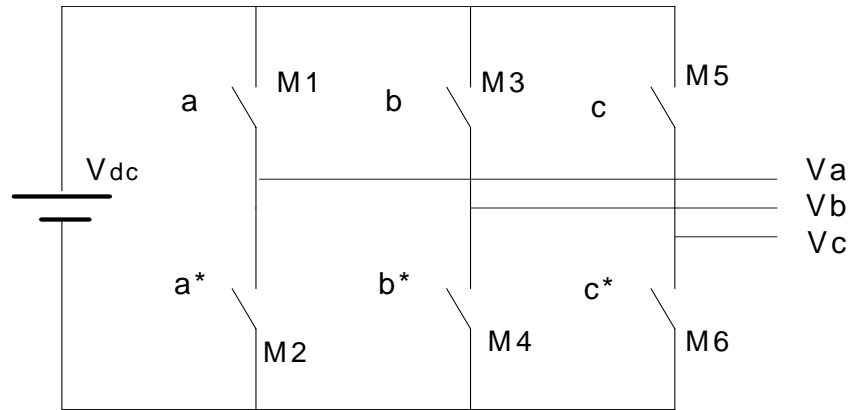


Figure 15: Three-leg voltage source inverter

Because of the constraint that the input lines must never be shorted and the output current must always be continuous, the voltage source inverter can assume only eight distinct topologies.

2.6.2 Voltage Space Vector

For a three-phase voltage source inverter (VSI), shown in Figure 15, V_a , V_b and V_c are the output voltages and M_1, M_2, \dots through M_6 are power IGBT which are controlled by gate signal. When the upper IGBT's are turned on, i.e. a, b or c is 1, the lower sides are turned off, i.e. a^*, b^* or c^* is 0. SV PWM determines the switching sequence of upper sides. The on and off status of the three upper transistors compose eight possible vectors. They are denoted by $U_0, U_{60}, U_{120}, U_{180}, U_{240}, U_{300}, O_{000}$ and O_{111} , as shown in Figure 16.

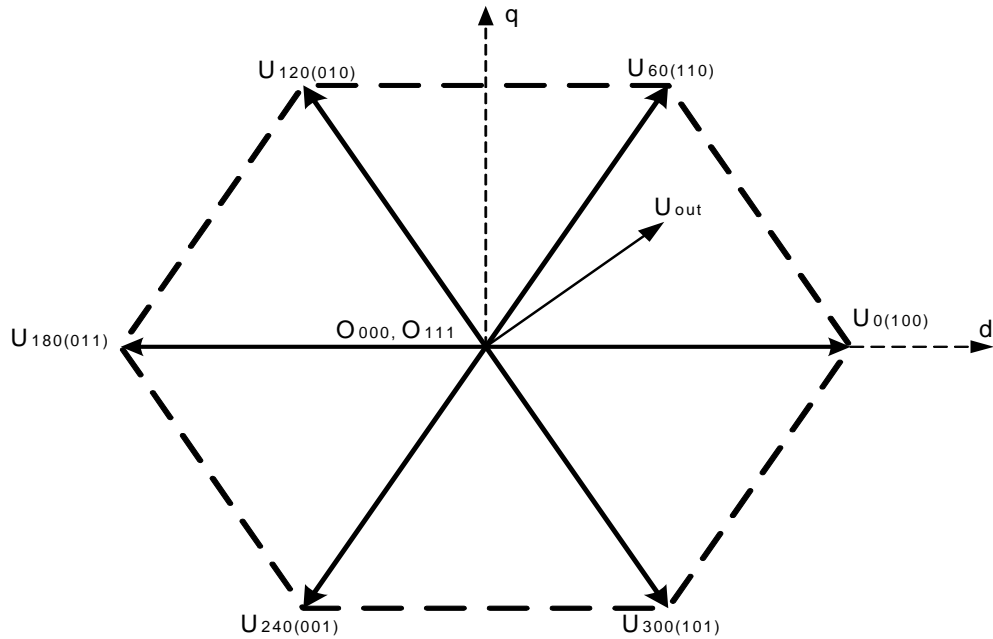


Figure 16: Space vectors

There are six non-zero vectors and two zero vectors. The six nonzero vectors form a hexagon as shown in figure 16. The angle between any two adjacent non-zero vectors is 60 degrees. Six out of these eight cases produce a nonzero output voltage and are known as non-zero switching states and the remaining two topologies produce zero output voltage and are known as zero switching states.

2.6.1 Space Vector Modulation

The desired three phase voltages at the output of the inverter could be represented by an equivalent vector V rotating in the counter clock wise direction. The magnitude of this vector is related to the magnitude of the output voltage. The time this vector takes to complete one revolution is the same as the fundamental time period of the output voltage. Table 1 gives a

summary of the switching states and the corresponding line to line voltages after normalized by DC voltage V_{dc} of the three-phase voltage source inverter.

Table 1

SV PWM switching states [5]

a	b	c	V_{ab}	V_{bc}	V_{ca}
0	0	0	0	0	0
1	0	0	1	0	-1
1	1	0	0	1	-1
0	1	0	-1	1	0
0	1	1	-1	0	1
0	0	1	0	-1	1
1	0	1	1	-1	0
1	1	1	0	0	0

Simulation of the space vector has been conducted to verify the functionality of the space vector modulation. Figure 17 illustrates space vector generation loop.

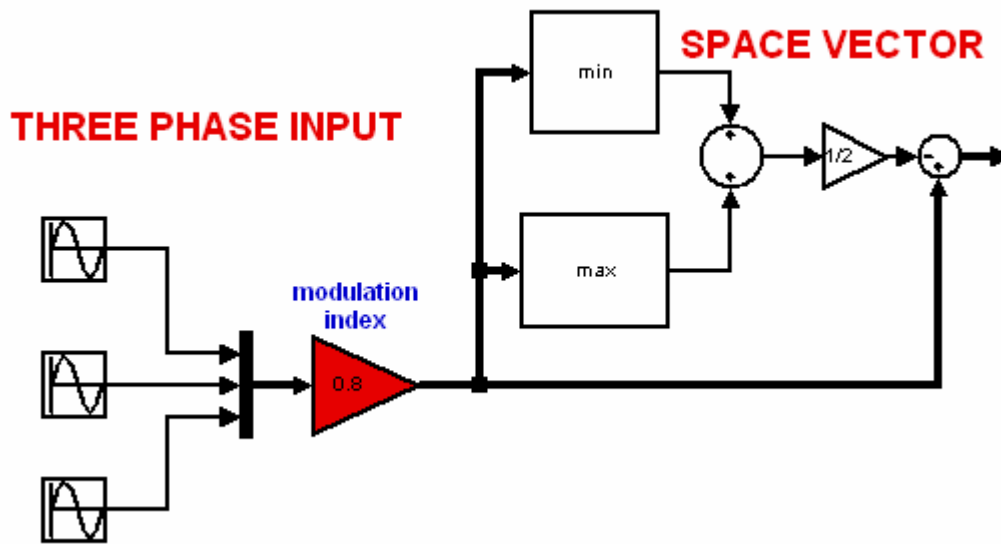


Figure 17: Block diagram of space vector

Three-phase currents are multiplied by the modulation index (M) which is the ratio of the peak to peak AC voltage to the DC voltage. It determines the width of the pulses and therefore the RMS value of the inverter output voltage. M is usually adjusted by varying the amplitude of the reference wave while keeping the carrier wave amplitude fixed. The inverter output frequency is varied by varying the reference wave frequency [6]. This loop generates the control signal that includes the duty ratio information. Figure 18 illustrates the control signal.

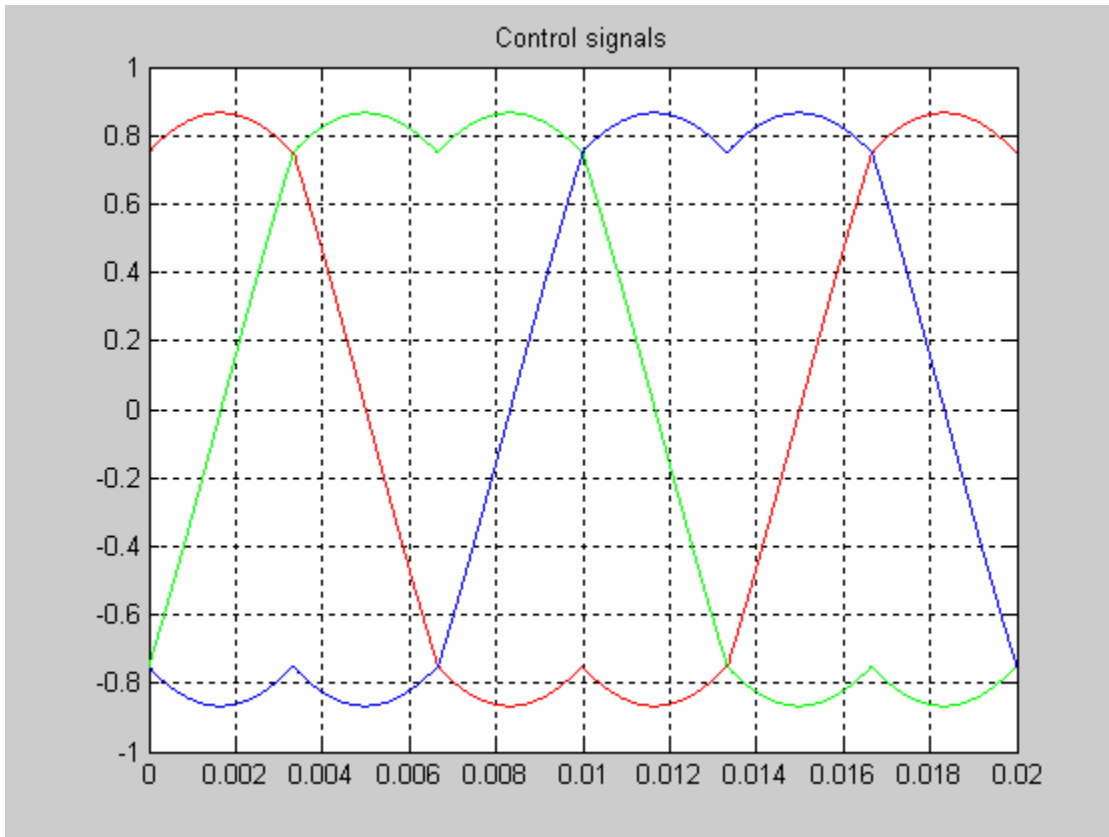


Figure 18: Control signal

Then, Figure 19 illustrates the loop for PWM generation and Figure 20 illustrates the simulated three phase.

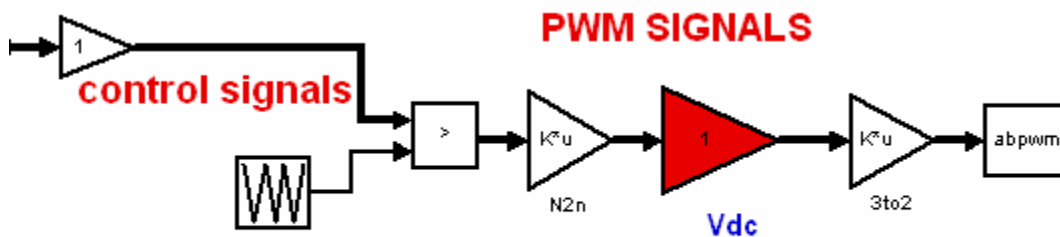


Figure 19: Generation of the PWM signals

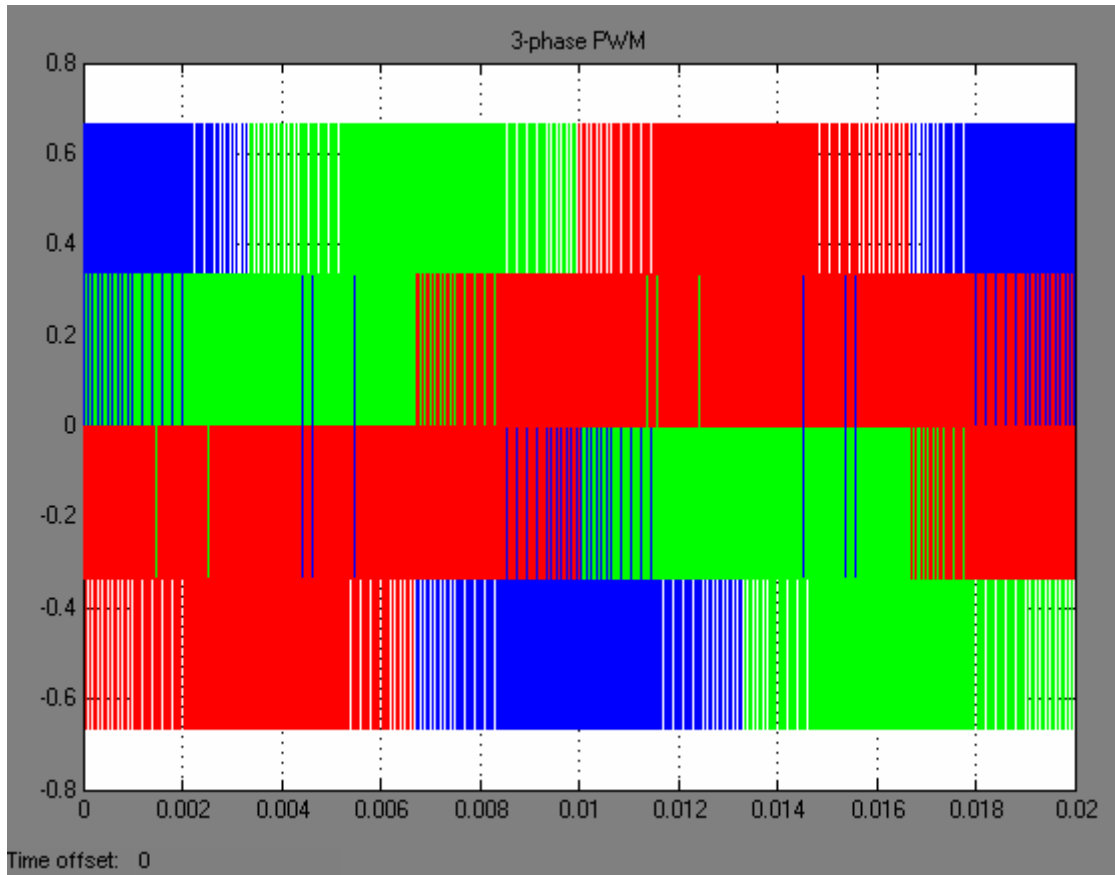


Figure 20: Three phase PWM signals

The output space vector trajectory has been modeled to verify the magnitude and frequency of the output voltage. Figure 21 illustrates the output space vector trajectory.

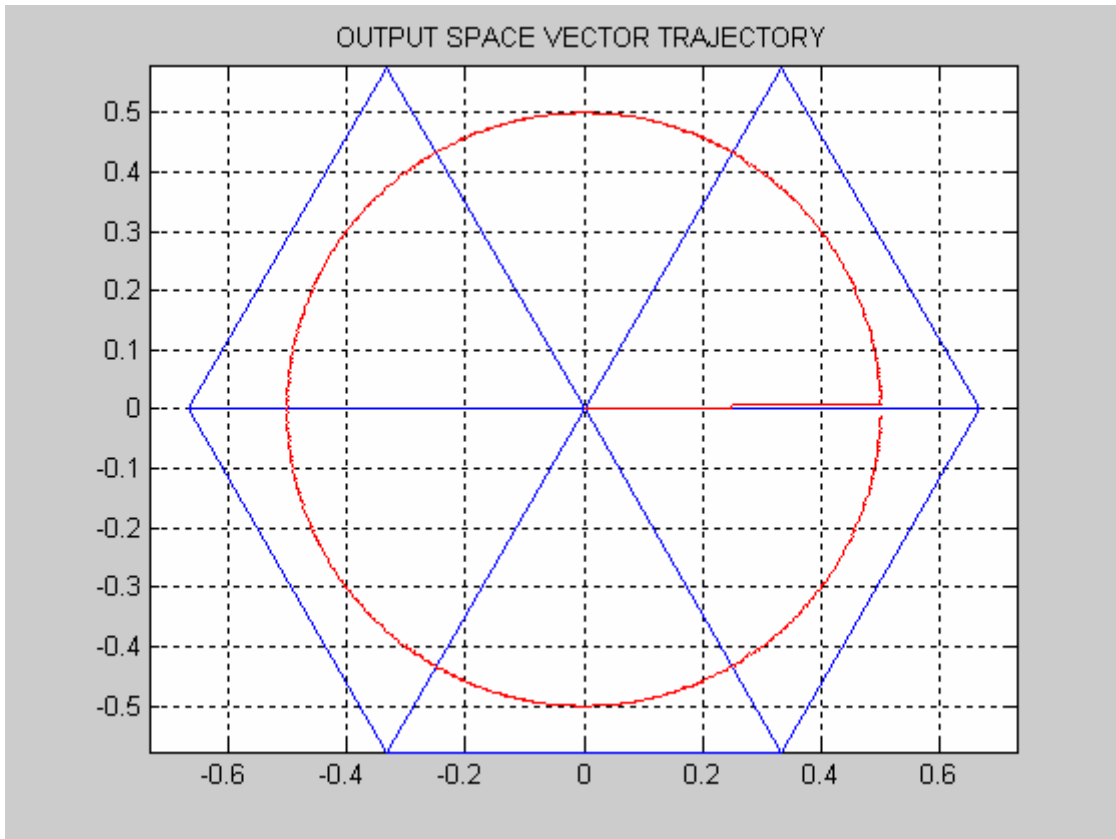


Figure 21: Output space vector trajectory

2.6 Digital Controller Implementation Issues

This section discusses some of the issues involved in implementing digital controllers including finite word length effect, closed loop algorithm sampling rate and code development.

2.6.1 Finite Word Length Effects

The Texas Instruments C2000 platform uses a fixed arithmetic processor. For the case of the 2407 DSP, it has 16 bits available for representing the magnitude of a signal. Signals have to be scaled to fit into the dynamic range and word length of the processor. Otherwise noise is

created and results in program instability. Word length can also be an issue when multiplying two 16 bit variables which require 32 bit storage. The higher 16 bits are stored and the lower are thrown away. This situation results in loss of precision due to round off error. This error grows as the closed loop algorithm iterates. Therefore, the selection of proper scaling is very critical in minimizing the round off error caused by a finite word length.

2.6.2 Closed Loop Algorithm Sampling Rate

One of most important design parameters in control system is the algorithm sampling rate. The 200 kW induction generator closed loop algorithm is chosen to be ten times the system bandwidth. Figure 22 illustrates the closed loop algorithm sampling rate.

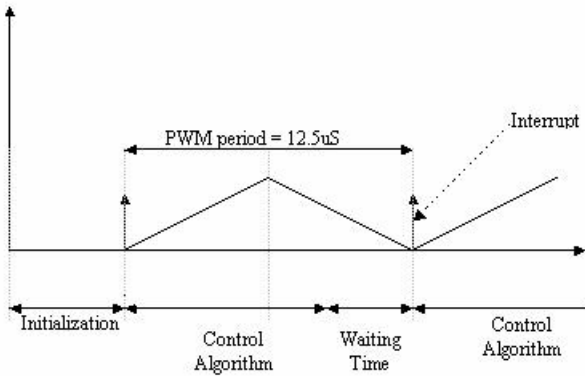


Figure 22: Closed algorithm sampling rate

The interrupt service routine period is 28 kHz and the closed loop algorithm sampling rate is 12.5 *us* .

2.6.3 Software Development

Software for the 200 kW was developed using TI assembly language. Assembly language was used to save on the code overhead. The software loops were verified in real-time using kernel routines. Debugging in real-time implementation was necessary in order to access memory and register without stopping the processor. All software routines are available in the Appendix.

CHAPTER THREE: GAIN SCHEDULING METHOD

3.1 Gain Scheduling For Specific Load Conditions

Voltage regulation of the 200 kW induction generator has been implemented using a (PI) controller. For a nonlinear dynamic plant such as the 200 kW generator, PI controller's are characterized by an overshoot response caused by a large error between the DC voltage command to the sensed DC voltage and also the voltage response between step loads is very slow due to the fixed proportional and integral gain controller Figure 23 shows a block diagram of a PI controller.

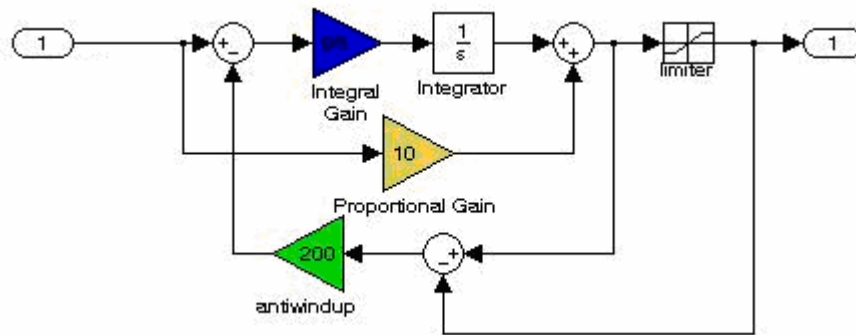


Figure 23: Classical proportional and integral controller

The proportional path adjusts the I_{Qref} in direct proportion to the V_{fdb} . The larger the proportional gain, the larger the I_{Qref} will change for a given error. The proportional path reduces the error however, does not eliminate it; there is always an offset between V_{ref} and V_{fdb} . The integral path corrects for any offset that may occur between V_{ref} and V_{fdb} automatically over time.

As the 200 kW generator load varied, the DC voltage regulation became unstable in the transient region (between loads) when the PI gains were fixed to a constant value. Furthermore, this instability contributed to the 200 kW generator de-Excitation. To eliminate this issue, the proportional gains are required adjustments with respect to the load conditions. Finding the gains for each load condition was the key to appropriately regulate the voltage. A closed loop algorithm was developed with capability to change K_p and K_i as the generator load varies in real-time. Transient tests have been conducted to find the gains at 44000 RPM at no load then at 32.8 kW and 48.6 kW.

3.2 Search of Proportional & Integral Gain Using Ziegler-Nichols Tuning Method

The search for K_p and K_i in the unstable regions has been conducted through transient tests. The gains were obtained by using Ziegler-Nichols tuning method [10], since the generator is not been modeled mathematically. The procedure of this method is conducted by setting K_p to a low value and K_i to zero. Then a voltage step is applied at no-load then at 32.8 kW and 48.6 kW to investigate the DC voltage transient. When the PI output oscillations decay, K_p has to be increased. Then, when the oscillations increase in amplitude (unstable system), K_p has to be reduced. At the point of stability, K_i is set to a value to eliminate the steady state error. This method is iterated until the range over which K_p and K_i varied is obtained.

A constant value of K_i is found to be adequate under both load conditions. However, K_p has to be increased by 36 % when the step load was applied from 32.8 kW to 48.6 kW. Table 2 shows the values for K_p and K_i under 0, 32.8kW, and 48.6 kW load conditions.

Table 2

K_p and K_i under different load condition

POWER, kW	REGULATED VOLTAGE, Volt DC	K_p	K_i
0	270	0	9000
32.8	270	1100	9000
48.6	270	1500	9000

After gathering K_p and K_i with respect to their load conditions, these values have been programmed into a scheduling algorithm. Figure 24 shows block diagram of the generator closed loop algorithm.

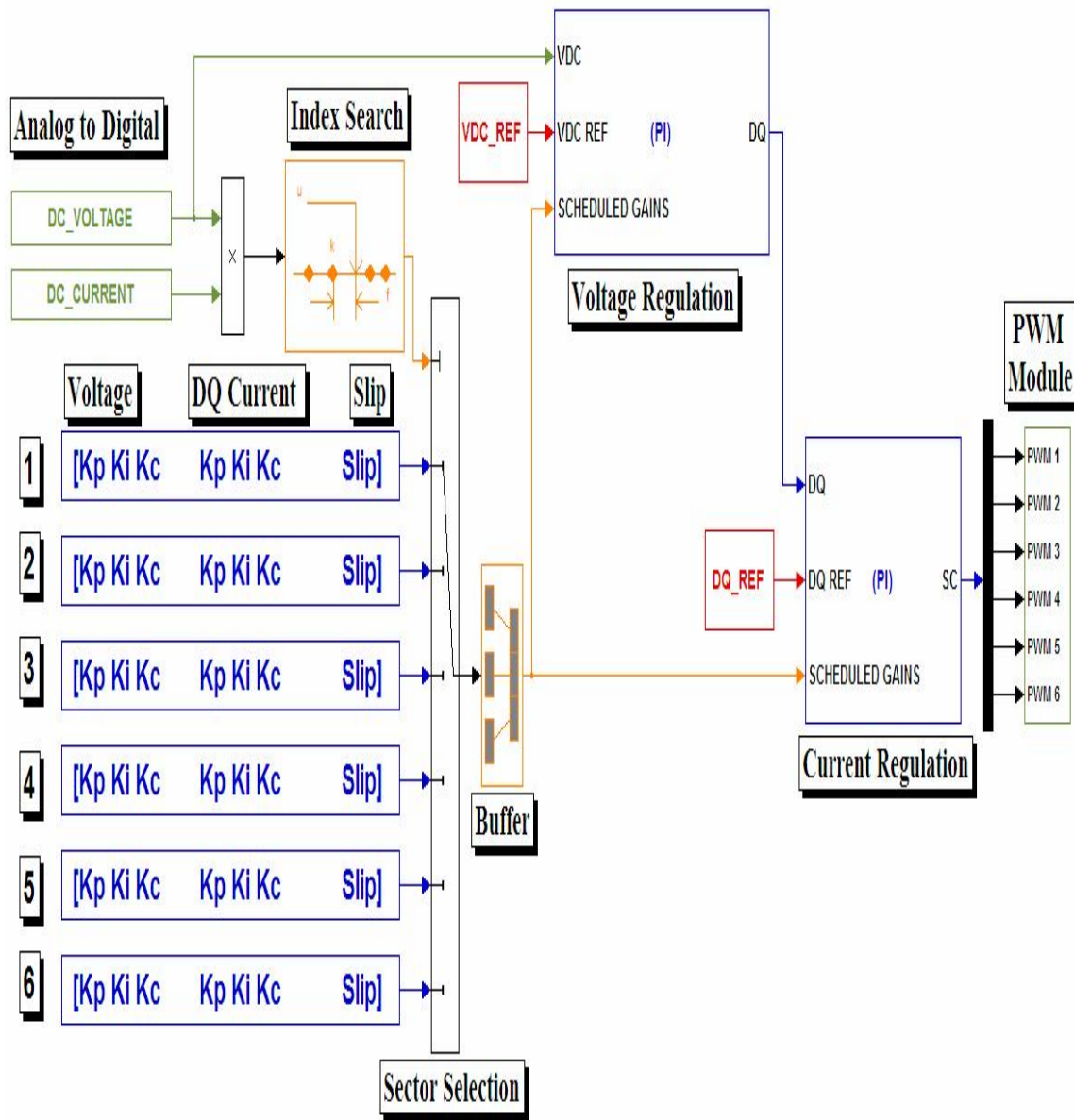


Figure 24: Generator closed loop algorithm

The product between the DC voltage and the DC current signals has been established to obtain the power. The power value is processed in a look up table module to search for the pre-selected load conditions to use. Then, each set of K_p and K_i values is reserved in an array and

stored until triggered by the power signal. This method works such as there is a linear controller for each condition. Based on the load, the PI regulator receives the corresponding gains for that condition. Figure 25 illustrates the responses of the DC voltage, DC current and space vector command respectively under the tested load conditions.

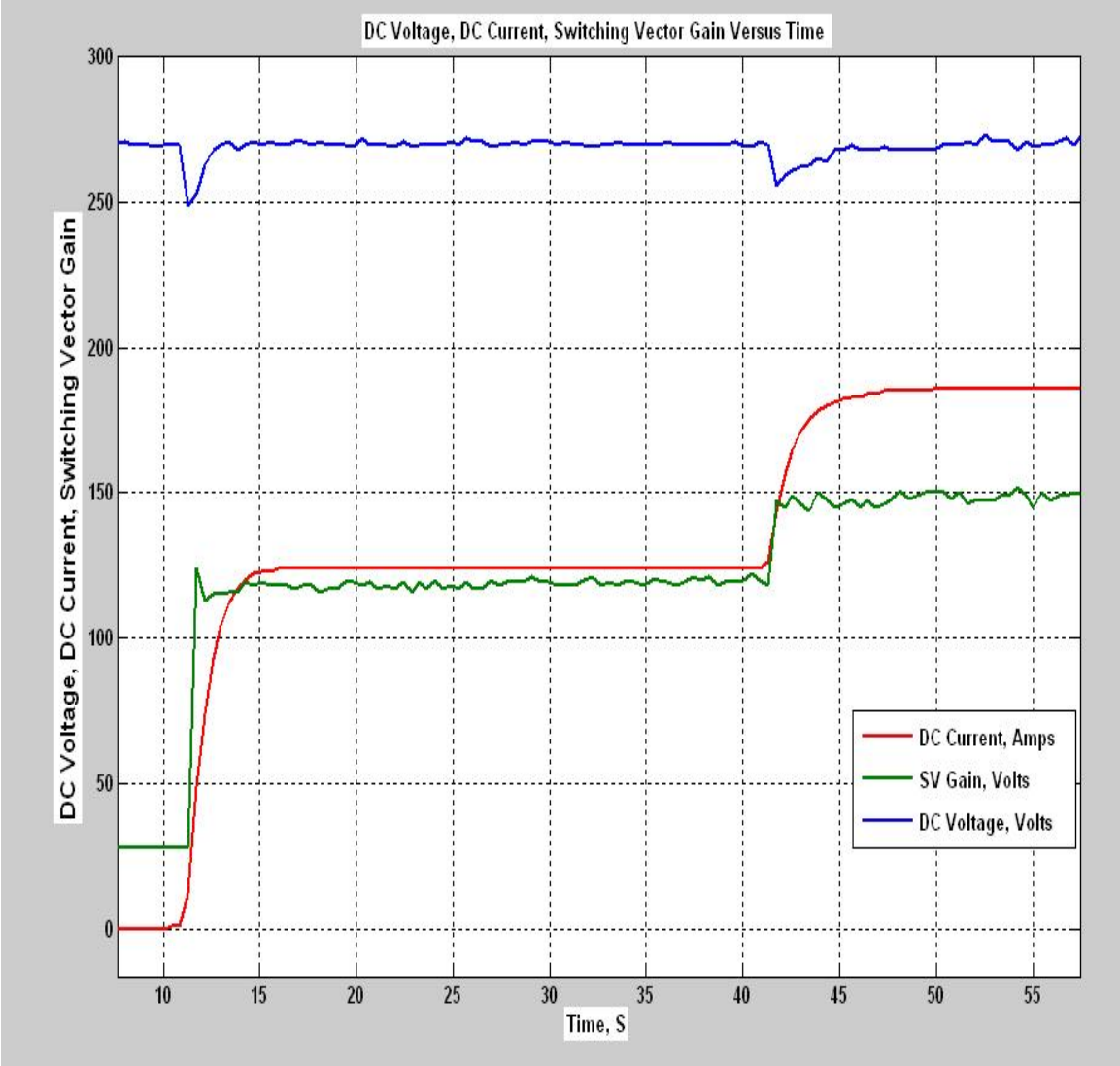


Figure 25: DC voltage, DC current and space vector command versus time

The gain scheduling method has been successfully implemented to regulate DC voltage of the 200 kW generator under 32.8 kW and 48.6 kW. However, in order to regulate at all load conditions, this method has been generalized and was able to tune the PI controller gains over all load condition. The next chapter discusses a generalized adaptive method.

CHAPTER FOUR: ADAPTIVE METHOD

4.1 Adaptive Method For Any Load Condition

The gain scheduling method discussed in chapter 3 is further developed to maintain a constant DC voltage of the generator to any load condition. The idea is to implement an algorithm that automatically regulates K_p according to any load condition. To develop this kind of algorithm, a relationship between K_p , K_i and power needs to be established. However, in the case of the 200 kW generator, a constant K_i is found to be adequate. Only K_p and power relationship needs to be established. Based on the transient tests data, proportional gains are fitted through polynomial curves to represent this relationship. Figure 26 illustrates K_p versus power.

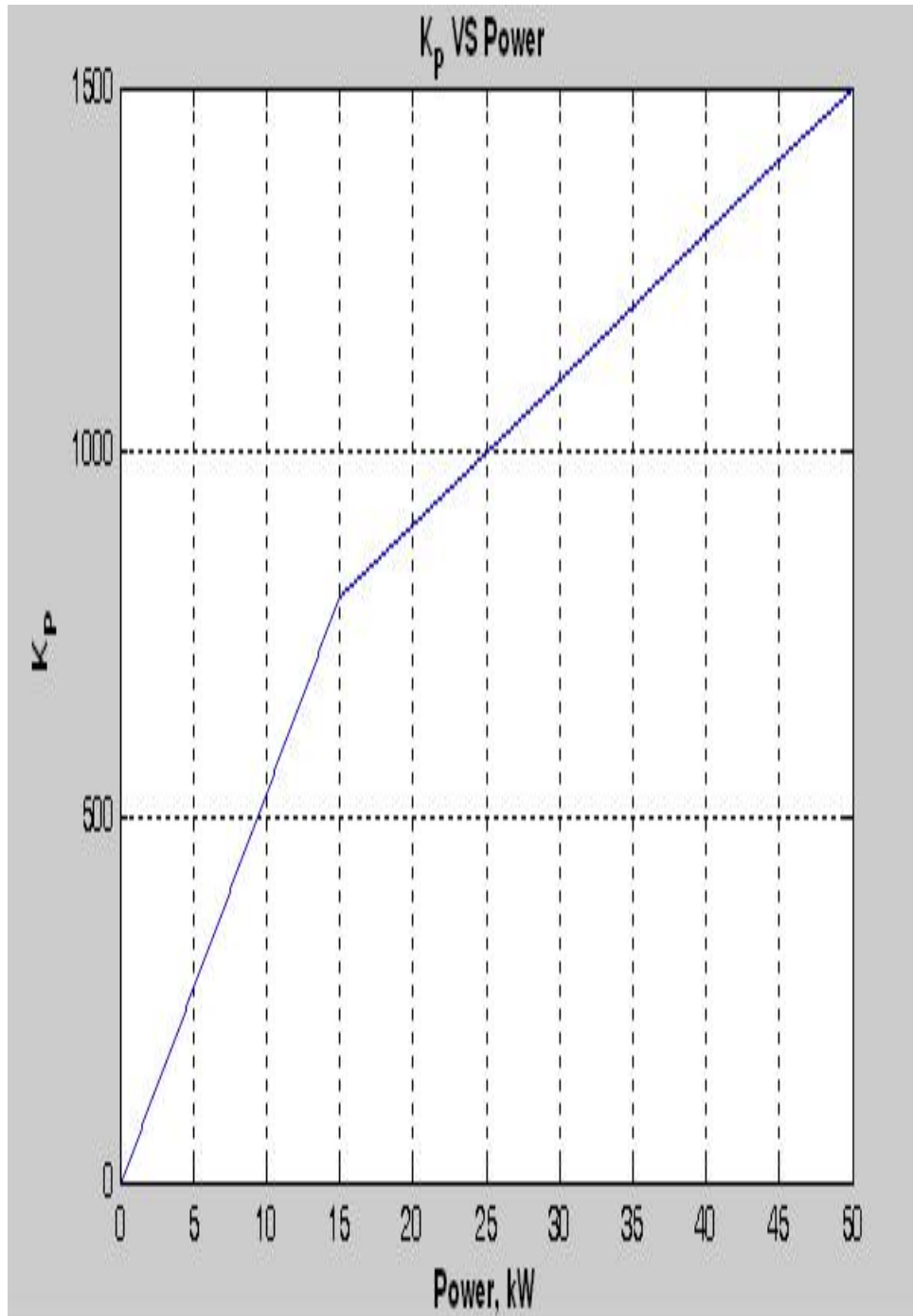


Figure 26: K_p versus power

Analysis of curve fitting has been conducted for first and second order polynomial to find the best fit. The reason to sequentially implement from a lower order polynomial is related to the algorithm computation speed. The lower the order of the polynomial the faster the closed loop algorithm will compute. Figure 27 illustrates the curve fit for the first and second order polynomial.

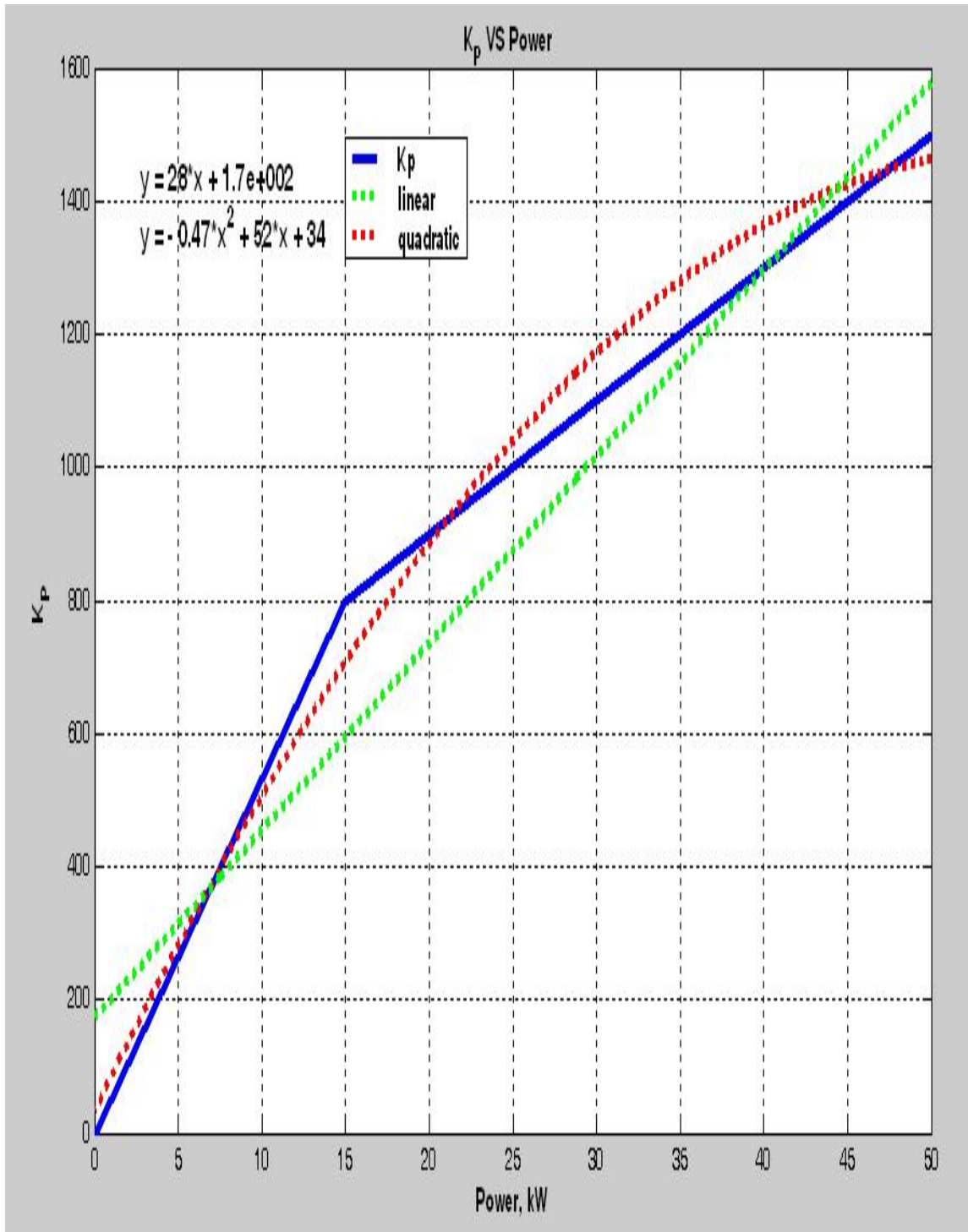


Figure 27: First and second order polynomial

4.2 First Order Polynomial

The first order polynomial has been implemented and tested initially. Figure 28 illustrates the curve fitting and offsets.

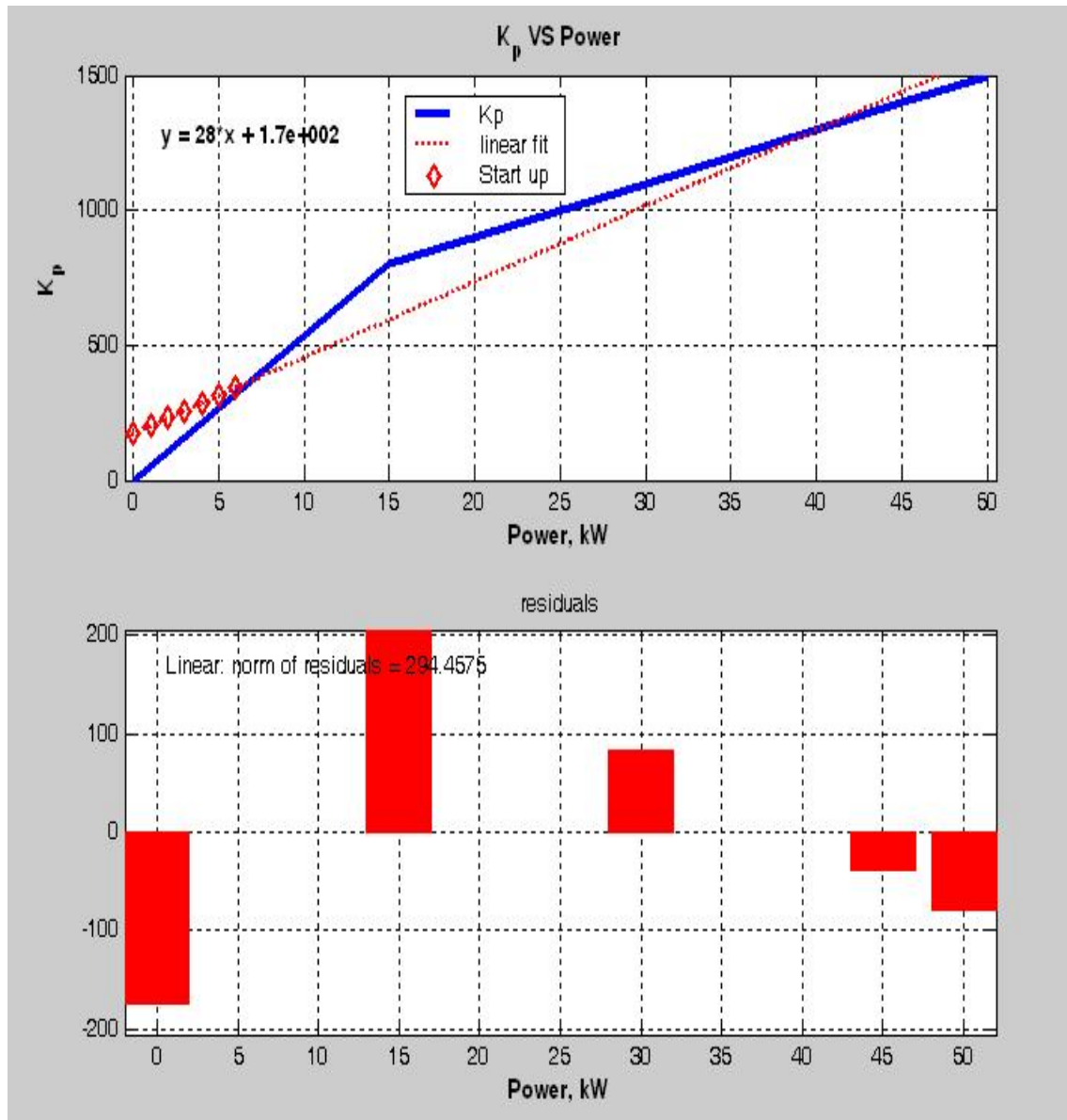


Figure 28: First order fit

The first order polynomial is not capable of regulating K_p because at start up an offset of 173 exists and results in instability of the loop. This results further the investigation into the second order fit.

4.3 Second Order Polynomial

The second order polynomial has been implemented and tested. Figure 29 below illustrates the curve fitting and offsets.

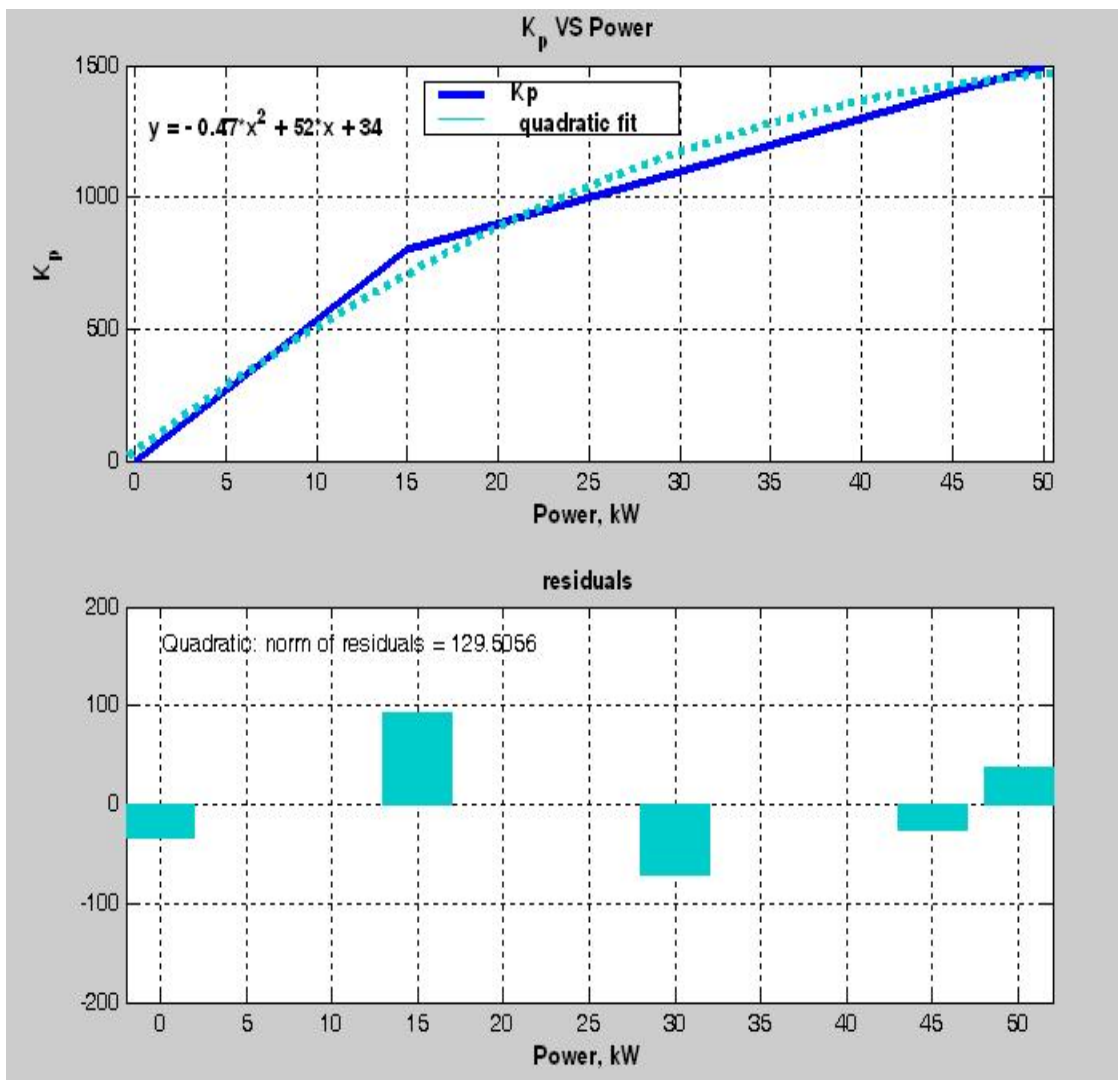


Figure 29: Second order fit

The second order polynomial has been successfully implemented to regulate K_p . The concern is that an offset of 35 existed, however it does not affect the stability of the loop. The second order polynomial has been programmed in the Texas Instrument DSP 2407 and tested at 12000 RPM with four load conditions from 0 to 5.1 kW.

4.4 Results

The relationship between the proportional gains and power has been established with a second order polynomial. The 200 kW generator DC voltage regulation has been successfully tested at 12000 RPM with four load conditions from 0 to 5.1 kW. The proportional gains are automatically computed using the second order polynomial. Table 3 illustrates four load conditions from 0 to 5.1 kW and Figure 30 illustrates the responses of the DC Current, DC Voltage.

Table 3

Four Load conditions from 0 to 5.1 kW

POWER, kW	VOLTAGE REGULATION	DC CURRENT
0	60	0
2.7	60	45
4.02	60	67
4.74	60	79
5.1	60	85

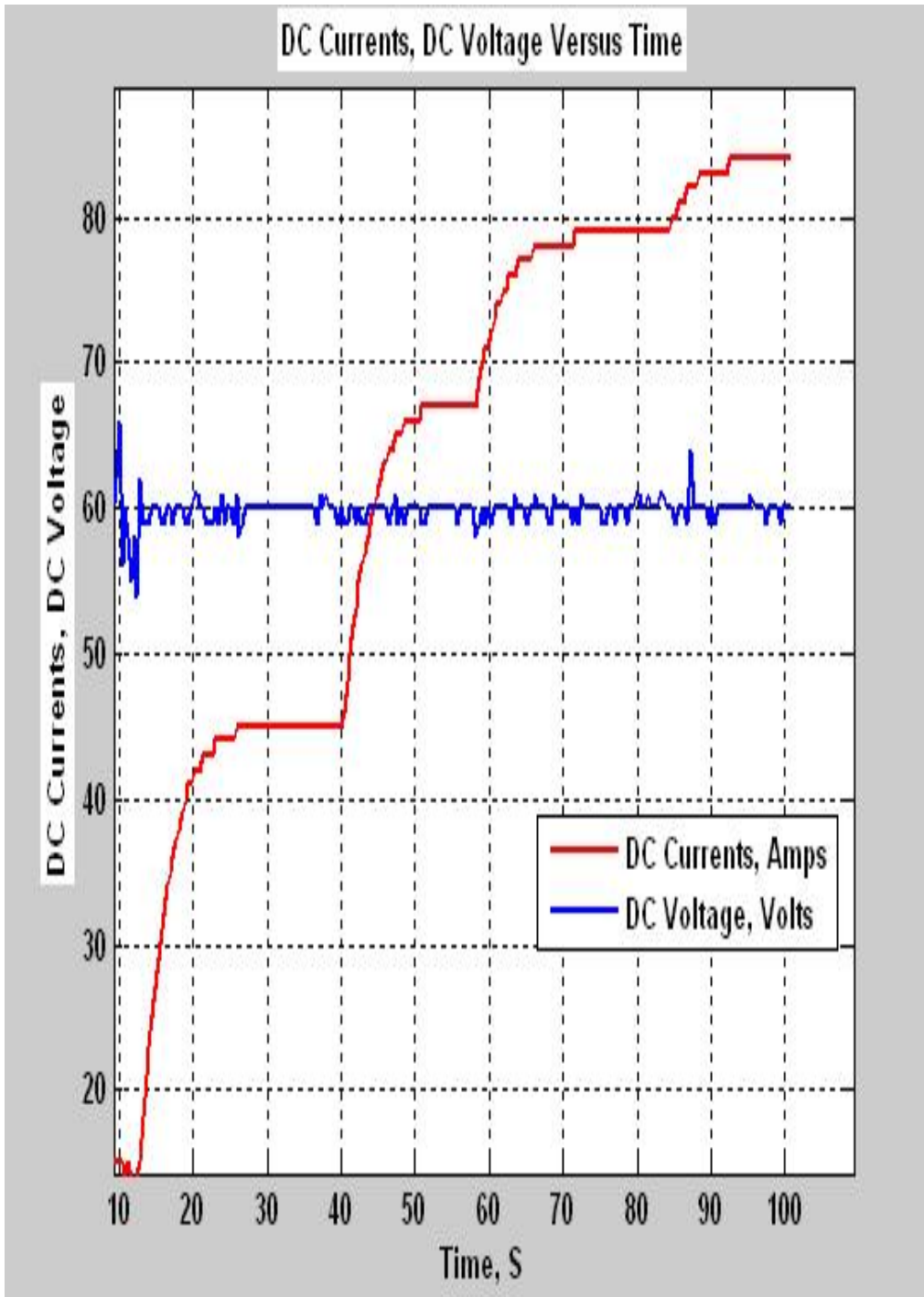


Figure 30: DC current, DC voltage Versus Time

CHAPTER FIVE: CONCLUSION

In this thesis, closed control system has been developed successfully to control the DC voltage output of the 200 kW induction generator using Field Oriented Control. A gain scheduling control algorithm has been developed by selecting the appropriate controller gains with respect to the generator load. The gain scheduling control algorithm facilitates the understanding of the generator behavior which resulted in developing a better approach to control the DC voltage adaptively. Both the gain scheduling and adaptive control methods have been implemented to successfully regulate DC voltage. However, the adaptive method is better approach because it can be tailored to work for high speed induction generators rated from 5 kW to 200 kW.

Finally, some suggestions can be provided for future development: (1) use of a more powerful DSP such as the TI 28XX series to implement adaptive based algorithms (sensorless speed sensing, slip control based on rotor temperature) to improve the generator controls, and (2) investigation of a compact packaging using DSP based system for noise-free operation.

APPENDIX : SYSTEM CLOSED LOOP SOFTWARE

System Reference

```
.include      "x24x_app.h"
.global MON_RT_CNFG
.ref  SYS_INIT
.ref  RAMP_CNTL, RAMP_CNTL_INIT
.ref  target_value
.ref  rmp_dly_max, rmp_lo_limit
.ref  rmp_hi_limit
.ref  setpt_value, s_eq_t_flg
.ref  DATA_LOG, DATA_LOG_INIT
.ref  dlog_iptr1, dlog_iptr2
.ref  trig_value
.ref  FC_PWM_DRV, FC_PWM_DRV_INIT
.ref  Mfunc_c1, Mfunc_c2, Mfunc_c3, Mfunc_p
.ref  n_period
.ref  ILEG2DRV, ILEG2DRV_INIT
.ref  Ia_gain, Ib_gain, Ia_offset, Ib_offset
.ref  Ia_out, Ib_out
.ref  CLARKE, CLARKE_INIT
.ref  clark_a, clark_b
.ref  clark_d, clark_q
.ref  SPEED_PRD, SPEED_PRD_INIT
.ref  time_stamp
.ref  rpm_max, speed_scaler, shift
.ref  speed_prd, speed_rpm
.ref  event_period
.ref  CAP_EVENT_DRV, CAP_EVENT_DRV_INIT
.ref  CAP_EVENT_DRV_CLKPS_INIT
.ref  CLK_prescaler_bits
.ref  pid_reg_iq, pid_reg_iq_init
.ref  iq_fdb, iq_ref, Kp_q, Ki_q, Kc_q
.ref  Umax_q, Umin_q
.ref  uq_int
.ref  uq_out
.ref  CURRENT_MODEL, CURRENT_MODEL_INIT;
.ref  spd_cur_mod
.ref  theta_cur_mod
.ref  fs
```



```

.ref    slip_freq
.ref    pid_reg_spd,pid_reg_spd_init ; function call
.ref    spd_fdb,spd_ref,Kp_spd,Ki_spd,Kc_spd
.ref    spd_out
.ref    spd_int
.ref    qsqrt
.ref    sqrt_input,sqrt_output
.ref    qsqrt2
.ref    sqrt_input2,sqrt_output2
.ref    SVGEN_MF, SVGEN_MF_INIT
.ref    sv_gain, sv_offset, sv_freq
.ref    Ta, Tb, Tc
.def    GPR0
.bss   GPR0,1
.bss   GPR1,1
.bss   GPR2,1
.bss   my_iq_ref,1
.bss   my_id_ref,1
.bss   speed_reference,1
.bss   isr_ticker,1
.bss   test_cur_mod1,1
.bss   test_cur_mod2,1 ;
.bss   test_cur_mod3,1
.bss   speed_prdnew,1
.bss   speed_prdlo,1
.bss   speed_prdhi,1
.bss   speedcoeff,1
.bss   speedtemp,1
.bss   speederrorlo,1
.bss   speederrorhi,1
.bss   sqrt_outputnew,1
.bss   sqrt_outputlo,1
.bss   sqrt_outputhi,1
.bss   sqrt_outputcoeff,1
.bss   sqrt_outputtemp,1
.bss   sqrt_outputerrorlo,1
.bss   sqrt_outputerrorhi,1
.bss   sqrt_outputnew2,1
.bss   sqrt_outputlo2,1
.bss   sqrt_outputhi2,1
.bss   sqrt_outputcoeff2,1
.bss   sqrt_outputtemp2,1
.bss   sqrt_outputerrorlo2,1
.bss   sqrt_outputerrorhi2,1
.bss   spd_out1,1

```

```

.bss  sqrt_output1,1
.bss  KT,1
.bss  spd_outh,1
.bss  skip_const,1
.bss  slip_input,1
.bss  speedtemp2,1
.bss  fe,1
.bss  test_cur_mod4,1
.bss  min,1
.bss  max,1
.bss  speed_prdnew1,1

```

System Interrupt Configuration

```

.sect  "vectors"
.def  _c_int0
RESET  B    _c_int0
INT1   B    PHANTOM
INT2   B    T1_PERIOD_ISR
INT3   B    PHANTOM
INT4   B    PHANTOM
INT5   B    PHANTOM
INT6   B    PHANTOM
.include "rtvecs.h"
.text
_c_int0:

```

System Initialization

```

CALL SYS_INIT
CALL FC_PWM_DRV_INIT

```

System Infinite Loop

```

MAIN:
M_1  NOP
NOP
NOP
B    MAIN

```

System Interrupt Save & Restore

```

T1_PERIOD_ISR:
    MAR *,AR1      ;AR1 is stack pointer
    MAR *+         ;skip one position
    SST #1, *+     ;save ST1
    SST #0, *+     ;save ST0
    SACH *+        ;save acc high
    SACL *         ;save acc low
    POINT_EV
    SPLK #0FFFFh,IFRA ; Clear all Group A interrupt flags (T1 ISR)
    SETC SXM       ; set sign extension mode
    CLRC OVM       ; clear overflow mode
    POINT_B0
    LACC isr_ticker
    ADD #1
    SACL isr_ticker

```

System Algorithm

```

CALL    ILEG2DRV
LDP     #clark_a
BLDD   #Ia_out,clark_a
ldp    #Ia_out
lacc   Ia_out,16
add    Ib_out,16
NEG
sach   Ib_out
ldp    #clark_b
BLDD   #Ib_out,clark_b
CALL    CLARKE
LDP     #GPR0
BLDD   #clark_d,GPR0
BLDD   #clark_q,GPR1
spm    0
LT     GPR0
MPY    GPR0
PAC
LT     GPR1
MPY    GPR1
APAC
LDP     #sqrt_input ;
sach
sac1   sqrt_input ;
call   qsqrt
LDP    #spd_ref

```

```

bldd #speed_prd,spd_fdb;non filtered
bldd #speed_prdnew,spd_fdb;filtered
CALL pid_reg_spd
LDP #test_cur_mod4
LACC test_cur_mod4
BCND TestMod4, NEQ
CALL RAMP_CNTL
LDP #target_value
LDP #spd_ref
BLDD #setpt_value,spd_ref
ldp #spd_out1
lacc spd_out1,15
LDP #sqrt_input2
sach sqrt_input2+1
sac1 sqrt_input2
call qsqrt2
setc SXM
clrc OVM
spm #1
ldp #sqrt_output
lacc sqrt_output,16
ldp #sqrt_outputlo
subs sqrt_outputlo
sub sqrt_outputnew,16
sac1 sqrt_outputerrorlo
sach sqrt_outputerrorhi
lt sqrt_outputerrorlo
mpyu sqrt_outputcoeff
pac
sach speedtemp
lt sqrt_outputerrorhi
mpy sqrt_outputcoeff
pac
add speedtemp
adds sqrt_outputlo
add sqrt_outputnew,16
sac1 sqrt_outputlo
sach sqrt_outputnew
LDP #iq_fdb
BLDD #sqrt_outputnew,iq_fdb
LDP #sqrt_output2
setc SXM
clrc OVM
spm #1
ldp #sqrt_output2

```

```

lacc  sqrt_output2,16
ldp   #sqrt_outputlo2
subs  sqrt_outputlo2
sub   sqrt_outputnew2,16
sac1  sqrt_outputerrorlo2
sach  sqrt_outputerrorhi2
lt    sqrt_outputerrorlo2
mpyu  sqrt_outputcoeff2
pac
sach  speedtemp2
lt    sqrt_outputerrorhi2
mpy   sqrt_outputcoeff2
pac
add   speedtemp2
adds  sqrt_outputlo2
add   sqrt_outputnew2,16
sac1  sqrt_outputlo2
sach  sqrt_outputnew2
ldp   #test_cur_mod3
lacc  test_cur_mod3
bcnd  SWITCH_SPEED, NEQ
ldp   #iq_ref
BLDD  #sqrt_outputnew2,iq_ref
b     SWITCH_SPEEDout
SWITCH_SPEED
SWITCH_SPEEDout
CALL  pid_reg_iq

```

;Check for Capture event from Hall sensor (sprocket)

```

POINT_EV
BIT          IFRC, BIT0

```

;Check CAP flag for edge transition on CAP1 pin
;If no edge present skip speed routine

```

BCND SKIP_SPEED, NTC
CALL CAP_EVENT_DRV
LACC FIFO1

```

;Else fetch "Time-stamp" & proceed with Speed meas.

```

LDP          #time_stamp
SACL time_stamp
CALL SPEED_PRD

```

```

setc  SXM
clrc  OVM
spm   #1
ldp   #speed_prd
lacc  speed_prd,16

```

```

    ldp    #speed_prdlo
    subs  speed_prdlo
    sub   speed_prdnew,16
    sacl  speederrorlo
    sach  speederrorhi
    lt    speederrorlo
    mpyu  speedcoeff
    pac
    sach  speedtemp
    lt    speederrorhi
    mpy   speedcoeff
    pac
    add   speedtemp
    adds  speed_prdlo
    add   speed_prdnew,16
    sacl  speed_prdlo
    sach  speed_prdnew1
    lacc  speed_prdnew1
    sub   min
    bcnd  S_gminq,GEQ           ; Continue if tmp_q>=U_min
    lacc  min                   ; otherwise, saturate
    B     NextS
S_gminq
    lacc  speed_prdnew1
    sub   max
    BCND  S_lmaxq,LEQ           ; Continue if tmp_q<=U_max
    lacc  max                   ; otherwise, saturate
    b     NextS
S_lmaxq
    lacc  speed_prdnew1
    NextS
    sacl  speed_prdnew
Int_termq
    POINT_EV
    SPLK  #0FFFFh,IFRC        ;Clear all CAP flags
SKIP_SPEED
    Ldp   #spd_out1
    Lacc  spd_out1
    bcnd  TestMod3, gt
    ldp   #skip_const
    lacc  slip_input
    neg
    sacl  skip_const
    LDP   #slip_freq
    BLDD #skip_const,slip_freq

```

```

        b      TestModEnd33
TestMod3
    LDP  #slip_freq
    bldd #slip_input,slip_freq
TestModEnd33
    LDP  #spd_cur_mod
    BLDD #speed_prdnew,spd_cur_mod;Filtered
    BLDD #speed_prd,spd_cur_mod;non filtered
    CALL CURRENT_MODEL
    ldp  #test_cur_mod2
    lacc test_cur_mod2
    bcnd TestMod55, NEQ
    ldp  #sv_gain
    bldd #uq_out,sv_gain
    b      TestModEnd55
TestMod55
TestModEnd55
    CALL SVGEN_MF
    ldp  #sv_gain
    ldp  #test_cur_mod1
    lacc test_cur_mod1
    bcnd TestMod2, NEQ
    for test = 0
    ldp  #fs
    lacc fs
    ldp  #sv_freq
    BLDD #fs,sv_freq
    B      TestModEnd2
TestMod2
    ldp  #target_value
    ldp  #sv_freq
    bldd #setpt_value,sv_freq
TestModEnd2
    LDP  #Mfunc_c1
    BLDD #Ta,Mfunc_c1
    BLDD #Tb,Mfunc_c2
    BLDD #Tc,Mfunc_c3
    CALL FC_PWM_DRV
    spm  #0
    ldp  #spd_out
    lt   spd_out
    ldp  #KT
    mpy  KT
    pac
    ldp  #spd_out1

```

```

        sac1   spd_out1
        sach   spd_outh
SETC XF
END_ISR:
        POINT_PG0
        MAR   *, AR1      ;make stack pointer active
        LACL  *-          ;Restore Acc low
        ADDH *-          ;Restore Acc high
        LST   #0, *-      ;load ST0
        LST   #1, *-      ;load ST1
        EINT
        RET
PHANTOM B   PHANTOM

```


LIST OF REFERENCES

- [1] R.H. Park, “Two-reaction theory of synchronous machines: generalized method of analysis – Part I,” *AIEE Transaction*, Vol. 48, pp. 716-730, 1929.
- [2] E. Clarke, *Circuit Analysis of A-C Power Systems: Volume I – Symmetrical and Related Components*, John Wiley & Sons, New York, 1943.
- [3] D. Grahame Holmes and Thomas A. Lipo, “Pulse width modulation for power converter,” *IEEE Press Series on Power Engineering*, IEEE Press, Wiley-Interscience, 2003.
- [4] Bose. Bimal, *Modern Power Electronics And AC Drives*, Prentice-Hall, 2002.
- [5] Texas Instruments, SPRA284A, *AC Induction Motor Control Using Constant V/HZ Principle and Space Vector PWM Technique with TMS320C240*.
- [6] Ashfaq Ahmed, *Power Electronics for Technology*, Prentice Hall, 1999.
- [7] Jay Vaidya, Earl Gregory, “High speed induction generator for application in aircraft power systems,” *2004 Power Systems Conference*, Session: Advanced Power Systems Technologies I, Reno, NV, Nov. 2004.
- [8] X. S. Chen, A. J. Fleshj, C. W. Pang, and L. M. Zang, “Digital modeling of an induction generator,” *EE International Conference on Advances in Power System Control, Operation and Management*, Hong Kong, Nov. 1991.
- [9] B.K.Bose, *Power Electronics and Variable Frequency Drive - Technology and Application*, IEEE Press 1997.
- [10] Karl J. Åström and Björn Wittenmark, *Adaptive Control*, Addison-Wesley, 1990.