BASKENT UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

ANALYSIS AND CONTROL OF SYNCHRONOUS BUCK CONVERTER

OKKES TOLGA ALTINOZ

MASTER OF SCIENCE, THESIS

2009

SENKRON BUCK DÖNÜŞTÜRÜCÜSÜNÜN ANALİZ VE KONTROLÜ

ANALYSIS AND CONTROL OF SYNCHRONOUS BUCK CONVERTER

ÖKKEŞ TOLGA ALTINÖZ

In partial fulfillment of the requirements of the
Degree of Master of Science in
Electrical Electronic Engineering
2009

Graduate School of Natural and Applied Sciences;

This thesis has been approved in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN ELECTRICAL AND ELECTRONIC ENGINEERING** by the committee members.

Chairman : Yrd. Doç. Dr. İbrahim SEFA

Member(Supervisor) : Yrd. Doç. Dr. Hamit ERDEM

Member : Yrd. Doç. Dr. Mustafa Doğan

APPROVAL

This thesis is approved by the committee members on...../...../.....

..../..../.....

Prof.Dr. Emin AKATA

DIRECTOR, GRADUATE SCHOOL OF

NATURAL AND APPLIED SCIENCES

ACKNOWLEDGMENTS

I wish to express my deepest gratitude to my supervisor Yrd. Doç. Dr. Hamit Erdem for his guidance, advice, critics, encouragement, insight through the research and motivating me to achieve my goals.

I am greatful to my parents for their patience and being there for me at time of downfalls.

Dedicated to…

Prof. Dr. Turhan Çiftçibaşı

(1948-2008)

ABSTRACT

ANALYSIS AND CONROL OF SYNCHRONOUS BUCK CONVERTER

Okkes Tolga ALTINOZ

Baskent University Graduate School of Natural and Applied Sciences

Department of Electrical Electronic Engineering

In this study, analysis and control of synchronous buck converter is investigated. Modeling of the converter is achieved and optimized control methods are applied as simulation and hardware implementation. Matlab/Simulink is used as simulation environment. Hardware implementation is realized by using a microcontroller based circuit, which is dsPICDEM SMPS development board.

Prior studies show that PID, fuzzy logic and sliding mode controllers have been frequently applied to buck converter. In this thesis, to improve the efficiency of these control methods, some parameters of these controllers are optimized. Particle Swarm Optimization method is selected for obtaining optimum controllers because of the method's easy programming and fast convergence features. The optimized controllers are implemented on the Synchronous Buck converter and their performances are compared under load and line variations.

KEYWORDS: synchronous buck converter, PID, sliding mode control, fuzzy logic controller, particle swarm optimization

Supervisor: Yrd.Doç.Dr. Hamit ERDEM, Baskent University, Department of Electrical and Electronic Engineering

ÖZ

SENKRON BUCK DÖNÜŞTÜRÜCÜSÜNÜN ANALİZ VE KONTROLÜ

Ökkeş Tolga ALTINÖZ

Başkent Üniversitesi Fen Bilimleri Enstitüsü

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Bu çalışmada, senkron buck dönüştürücüsünün analiz ve kontrolü incelenmiştir. Dönüştürücü modeli elde edilmiş ve optimize edilmiş kontrol metotları benzetim ve donanım ortamında uygulanmıştır. Matlab/Simulink programı benzetim uygulaması için kullanılmıştır. Donanım uygulamasında mikroişlemci tabanlı dsPICDEM SMPS geliştirme kartı kullanılmıştır.

Önceki çalışmalardan PID, bulanık mantık ve kayan kipli kontrol yöntemlerinin sıklıkla kullanılmakta olduğu görülmüştür. Bu tezde, bu kontrol yöntemlerinin etkinliğini arttırmak için kontrol yöntemlerinin bazı parametreleri optimize edilmiştir. Parçacık Sürü Optimizasyon yöntemi kolay programlanabilir olması ve hızlı yakınsaması nedenleri ile optimum kontrol yöntemlerinin bulunması için seçilmiştir. Optimize edilmiş kontrol yöntemleri senkron buck dönüştürücüsüne uygulanmış ve yük ve hat değişimleri altındaki performansları karşılaştırılmıştır.

ANAHTAR SÖZCÜKLER: senkron buck dönüştürücüsü, PID, kayan kip kontrolü, bulanık mantık, parçacık sürüsü optimizasyonu

Danışman: Yrd.Doç.Dr. Hamit ERDEM, Başkent Üniversitesi, Elektrik ve Elektronik Mühendisliği Bölümü.

TABLE OF CONTENTS

LIST OF FIGURES

Page

LIST OF TABLES

Page

LIST OF SYMBOLS

SMC : Sliding Mode Controller

FLC : Fuzzy Logic Controller

PSO : Particle Swarm Optimization

GA : Genetic Algorithm

SA : Simulating Annealing

DSP : Digital Signal Processor

ZN : Ziegler/Nichols

MF : Membership Function

V_{ref} : Reference Voltage

V_{out} : Output Voltage

C : Capacitor

L : Inductor

R : Load Resistor

u : Control Signal

S : Sliding Surface

1. INTRODUCTION

As a part of the power electronics, power converters are used to convert electrical power to desired form [1]. The power converters are the key area for various disciplines such as; computers, communication, medical, robotic, autonomous structures and many other systems. For example, for any unmanned vehicle, the main problem should be localization, mapping and mission execution but the operation time is also critical [2]. Thus, the power converter must be designed to increase the run time of the system. In general, application areas of the power electronic are STATCOM FACTS devices (multi-pulse and multi-level inverters in a range between 1MW and 100MW), medium voltage drivers (multi-level and current source inverter in a range between 100kW and 10MW), low voltage drivers (matrix and V-source inverter in a range between 100W and 10kW), renewable energy supplies, power supplies (flyback and forward dc/dc converters in a range between 10W and 1kW) and DC/DC converters, which are in a range between 1W and 100W. In this thesis, the development board dsPICDEM SMPS is used, which is composed two 5W DC/DC synchronous buck converter. These power level converters are used in compact devices like PDA and media players. This development board which is produced for experimental purposes has input voltage level minimum 7V and maximum 15V. In this thesis, various control methods are implemented on that development board. In general, the power converters can be classified based on their input-output relation as: AC-DC converters, DC-AC converters and DC-DC converters [3].

DC/DC converters use to convert the DC input voltage into a different DC output voltage level. There are five topologies are commonly used for DC/DC converters: Buck, Boost, Buck-Boost, Cuk and SEPIC converters. The buck and boost converters are the fundamental topologies. The other topologies are the combinations of these converters [1].

The most basic form of the DC/DC converter is the Buck converter. Buck converter is used to reduce the input voltage into a lower desired level. It composed from two switching circuit elements like a diode and/or MOSFET, and two storage elements as inductor and capacitor. The Buck converters are produced for real

time problems into two different classes: Buck Converter and Synchronous Buck Converter. This thesis will focus on Synchronous Buck Converter.

1.1 Problem Statement

The objective of the synchronous buck converter is to change the input voltage into the desired level and maintained that level in spite of fluctuations on input voltage (line variation) and output load change (load variation). Thus, control algorithms are needed to get the desired performance against load and line variations.

The control algorithms must be designed properly to get the desired performance. Therefore, the model of the converter is needed because most controllers are designed based on the mathematical model of the system and all of these controller performances are compared and investigated in the simulation environment by using system and controller models.

The performance of the controller is investigated in terms of the transient response in start up and under load & line variations [4]. The performance criteria are overshooting, settling time and steady state error. In summarized, the problem of the power converters is to obtain the mathematical model of the converter and design the controller with optimized closed-loop performance.

The problem in this thesis is the analysis, design and implementation of linear and nonlinear control methods for synchronous buck converter on a DSP.

1.2. Literature Review

The idea of the design of a control algorithm is to keep the output voltage in the desired level in response to load and line variations. Various control actions have been defining for converters. As control perspective, control actions that used power converters are classified as: linear controller and nonlinear controller.

Linear controller design is depended on frequency response and root locus analysis [5]. To design the controller, first the mathematical model of the converter is obtained then, by using frequency response and root locus methods, the linear controller is designed based on classical control system approach [6]. The linear controllers are designed based on the small signal model of the converter. Thus,

the linear controllers do not have the satisfactory response under varying operation points such as load and line variations [4]. To achieve good performance under change in the operation points, more accurate model is necessary. But, model becomes too complex. Another solution is to use an intelligent or nonlinear controller.

To control of the buck converters, various nonlinear methods are applied: Fuzzy Logic Controller, Sliding Mode Controller and PID Controller. Fuzzy logic controller (FLC) as an intelligent controller is the popular controller which is used in power converters [3]. FLCs are designed by using trail-error method and the mathematical model of the converter is not necessary. Thus, unlike linear controllers, FLC can give satisfactory performance under load and line change. But, the analysis of the FLC is very complex. To solve that problem, a scaling factor is defined to avoid the oscillation. FLC has complex nature when in comparison to the linear controller. For simplification design step and improve control performance, tuning of fuzzy sets and rule reductions has been applied.

Due to variable structure of the power converter, sliding mode controllers have been applied to control of these converters. Sliding mode controller with variable frequency and fixed frequency has been applied for control of DC/DC converters. In order to eliminate chattering problem and use PWM technique in the control of the converter, a PWM based fixed frequency SMC was preferred in recent studies. Sliding mode controllers (SMC) is a nonlinear control technique which is frequently used for control of the power converters [7]. SMC is very robust against parameter variation, external disturbance and unmodaled dynamics of the model of the converter. SMC has very basic control function that can be implemented in analog and digital devices [7-9]. However, the control signal of SMC should be infinite frequency in theory. Therefore, it is impossible to apply in real world problems. Thus, the problem called “chattering” occurs. The other problem of the SMC, variable frequency of the signal for power electronics that it makes the design of the input and/or output filter complex.

1.3. Motivation for the Research

In this thesis, the various control methodologies are reviewed and three significant control systems are selected, which are SMC, FLC and PID controllers. These controllers are discussed in literature before. In this thesis, the parameters of these controllers are determined by using the evolutionary algorithm which is called Particle Swarm Optimization. Then, the optimized controllers are implemented on the simulation and hardware environment.

Designing of the controllers (compensator, linear controller) for synchronous buck converter is based on frequency domain techniques. These techniques can be difficult to apply in practice. Fuzzy logic controller (FLC) is a nonlinear controller which isn't needed the mathematical model of the system. However, parameters of the FLC are determined by the trial-error approach which is complex and difficult to implement. Sliding Mode Controller (SMC) is widely used in DC-DC converter because converters' variable structure and time varying nature. SMC is robust against fluctuations and doesn't need an accurate model of the system. However, the sliding coefficient affects the performance of the controller and this coefficient is generally determined by geometric approaches or by solving Lyapunov equation.

The implementation of the controllers is the key point for the controller design. In general, the analog controllers (by using opamp and flip-flops) were used to implement the controllers. Nevertheless, complex controllers aren't implemented because they are unable to execute by analog devices. In the present, low cost microcontrollers are become available for control of converters. Some researchers use the 8-bit micro controllers to implement control algorithms. These microcontrollers are cheap in price but couldn't be achieved the exact controller performance. Other researchers decide to use 32-bit DSP devices. However, these devices are expensive to use in power converters.

In a general manner, two steps are taken to obtain the controller:
First: Designing of the linear, intelligent and nonlinear control systems are depended on different techniques such as frequency analysis, trail-error and geometric approaches. However, these approaches are generally complex and may not give the optimal performance.

Second: Implementation of the controller is chosen between analog or digital devices (8-bits, 16-bits, 32-bits).

1.4. Thesis Objectives

Linear controllers can be designed for control of converters by frequency techniques. However, linear controllers have disadvantages that a mathematical model (small signal model) is necessary to design the controller. Parameter tuning methods like Ziegler Nichols and genetic algorithm are used for parameter determination. Tuning of proper coefficients is important in design step. In implementation, linear controllers are implemented on analog and digital devices.

Fuzzy logic controller has been applied for power converter control by using expert knowledge. To simplify the design of FLC various learning methods have been applied. These studies provided tuning of fuzzy set, rule reduction or rule extract from data. All FLC for power converters used PWM technique. Output of fuzzy as change of the duty cycle is integrated and applied to the converter. Generally, a coefficient was used before integration, which affects converter output. Tuning of this coefficient will improve FLC performance.

The parameters of the controllers for power converters are determined by frequency analysis (for linear control, PID); geometrical approach (for nonlinear controller, SMC) and trail-error analysis (for intelligent control, FLC). These parameters are critical to obtain for the optimal performance of the controller.

The optimal methods for determinate the controller parameters are generally complex and/or not straightforward. Therefore, the optimization algorithm that has a basic structure, easy to implement and efficient against local optimum, is needed to a controller parameter determination. Thus, for the contribution to power electronic area, reduced the complexity of the parameter determination and provide step-by-step design approach, the control structures, which are PID, SMC and FLC parameters will be determined by the optimization algorithm which is called Particle Swarm Optimization (PSO), are selected as a test bed.

The objective of this thesis is;

- to analysis the synchronous buck converter and obtained the most popular model called state space averaging method.

- design PID, SMC and FLC controllers by using PSO and conventional ways
- provide a generalized controller tuning algorithm step-by-step approach
- implement the optimized controllers on DSP (dsPIC is selected 16-bits)

Optimized digital control for synchronous buck converter is an interesting multi-disciplinary research because the theory of this thesis is in the areas of power electronics, control systems and optimization.

This thesis analyzes modeling of the converter and aim to implementation of PID, FLC and SMC to the converter. Unknown controller coefficients are optimized by using PSO. All methods are simulated on computer and implemented on dsPIC board. The performance of the proposed controller is discussed and compared regarding to load and line variations.

The thesis is organized as follows; chapter two provides the analysis of the synchronous buck converter based on well known state space averaging method and provide the difference between synchronous buck converter and buck converter. Chapter three presents the general control methodologies which are used for control of power converters as extended literature review. Chapter four gives the simulation of PID, SMC and FLC based on both conventional methods and PSO. Chapter five provides the experiment results of the controllers and last chapter is the discussion of the results obtained in this thesis.

2. ANALYSIS OF SYNCHRONOUS BUCK CONVERTER

Analysis means reaching an outcome by dividing a subject into pieces and identifying the relationship between them. In this thesis, the subject is synchronous buck converter and outcome is the model of the converter because, designing the controller for the converter; the mathematical model is necessary.

The synchronous buck converter is a variable structure system (VSS) which means a system has the different structure based on different switching conditions. Thus, synchronous buck converter is divided into pieces called states of the converter. At the end of this chapter, the mathematical model will be obtained by analyzing the converter.

2.1. Synchronous Buck vs. Nonsynchronous Buck

DC/DC converters will frequently use in the next generation devices to supply lower or higher voltage level. The next generation of devices (cell phones, PDA, digital cameras and players) will have to provide the long operation period. Thus, microprocessors and memory devices need low power voltages. That kind of voltage is supplied by optimum DC/DC converters (synchronous switch).

Designing an optimum DC/DC converter involves issues such as; size, efficiency, temperature, accuracy and transient response. For next generation systems efficiency is the important property because of thermal constraints. Designers want to improve the efficiency without an increase the cost.

As one of the DC/DC converters, Buck converter is used for step down the input DC voltage. In previous decades, buck converter topology is composed from a single switch (generally FET, MOSFET) and a diode (generally Schottky) for freewheeling. The reason for not using the synchronous buck converter was the cost of the FET and FET driver circuits.

As FET technology has been improving FETs become lower prices and chosen against diode. The semiconductor industry has been developing synchronous buck converters that are thought more efficient than nonsynchronous buck converter. The synchronous buck converter can achieve low voltage and high current because the diode is replaced with a MOSFET. As an example [10], assume both converters are under full load and assume the duty cycle is about

10%, because of the dissipation in diode the nonsynchronous buck converter is not efficient. For 25% duty cycle the efficiencies are nearly same. Finally, for the lower duty cycle at switch 2 makes the nonsynchronous buck converter more efficient because the diode is conducting less often. As a result, the power dissipation is reduced significantly. On the other hand, use of a MOSFET instead of the diode prohibits the converter to enter the discontinuous conduction mode (DCM) at light loads.

In synchronous buck converter, the conduction loss is reduced and allows the bidirectional inductor current flow. Thus, the synchronous buck converter always maintains in continuous conduction mode (CCM). The synchronous buck converter has a higher efficiency at full load because of forward voltage drop of the diode in nonsynchronous buck converter. Both converters are in CCM in full load but in light load, nonsynchronous buck converter goes to DCM because the diode blocks the negative inductor current.

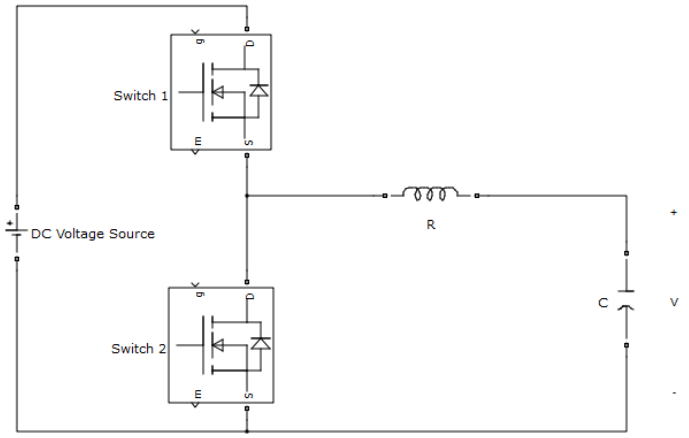


Figure 2.1 Synchronous Buck Converter

Figure 2.1 shows the synchronous buck converter. The main switch which is same as nonsynchronous buck converter is *Switch 1* and for synchronous buck converter topology the synchronous switch is *Switch 2*, *L* is inductance and *C* is the capacitor. The main advantage of the synchronous buck converter is the voltage drop against *Switch 2* is lower than the voltage drop across the diode in nonsynchronous buck converter topology. In synchronous buck converter, that means less power dissipations and higher efficiency.

The MOSFET saves space but complexity of the control is increased because both switches don't conduct simultaneously. Any simultaneously conduction could cause to overload and damage the system. This phenomenon is called "shoot-through". To get around this problem a suitable delay which is called "dead-time" is introduced.

As the conclusion; following results can be obtained from previous discussions about synchronous vs. nonsynchronous buck converter:

- Synchronous buck converter is smaller than nonsynchronous buck converter in size.
- Because of the shoot-through phenomenon, the dead-time must be defined for synchronous buck converter.
- Synchronous buck converter always in CCM. That makes easier to obtain the mathematical model and to design the controller because there isn't DCM to occur.
- The efficiency of synchronous buck converter at light load is higher than nonsynchronous buck converter. However, under higher load level, the efficiency depends on the duty cycle. If the duty cycle of the *Switch 1* is in lower level, that means the duty cycle of *Switch 2* is in higher level, makes the nonsynchronous buck converter is efficient that synchronous buck converter; vice versa.

2.2 Modeling of Power Converters

Buck converter is the simplest power converter that is widely used in power supplies. The switching process in buck converter makes the converter components connect with each other in a changing manner. Moreover, at each change the separate set of equations are obtained. Therefore, the transient analysis and controller design become more complex to solve each equation in a sequence [11-16]. Thus, the single equation is needed to overcome these problems. In literature, couples of methods for obtaining that equation exist. They are namely as averaging and linearization method [11-12, 16] and discrete modeling method [13-15].

State space averaging method is the most common modeling technique. In general, two different equations are obtained from two different configurations for two different switch positions. Then linearly weighted averages of these equations are become the overall converter model.

In spite of its easy structure, this averaging method is removed the all information related to switching frequency ripples of each variable. This averaging model is valid only if the switching frequency ripple of each component is smaller than average variables. Fortunately, these conditions are usually satisfied for buck converter.

Actually, after the average process, the duty appears in the *b* matrix (equation 2.16). Thus, the average model became time varying. To get rid of this problem the linearization is applied to the time varying model so, AC and DC components are separated. As linearization it is assumed that AC components are small-signal and neglected from the model. In literature, it can be found that other linearization methods are applied to state space method namely as an extended state space averaging or large signal model. (*The small signal model assumes the perturbations are very small and converter is always in the CCM but large signal model is operated in both CCM and DCM.)*

Another interesting research for modeling of converters is obtaining the average circuit. But, these structures do not have any advantage over designing the control system.

Another modeling method is called discrete modeling. This method is used same steps as averaging method but in results it constructed complex equations. In [13], Lee's study is presented generalized discrete modeling of any type of switching regulators using duty-cycle controller. Like state space averaging method, first for each time interval, discrete time state space equations are obtained. Then the general model is obtained. Then, the new equation is linearized at the equilibrium state of system. First equilibrium state is found and then the system is linearized. In [14], Lee's study presents the efficiency of the previously given the generalized discrete time-domain modeling without linearization. Thus, the produced model is become nonlinear. Then, this method is analyzed; simulation results and implementation results are compared for the multi-loop control system which can

be called peak current mode control. However, the model gives very close to real switching system; the complexity is higher than state space averaging method. In [15], the previous study is extended. In previous study the author is proposed a new modeling technique. (*but after Lee proposed that structure than Lee did not use it on his studies*) This method is called Discrete-time modeling. This study is aim to join average method and discrete method. But for constant frequency model of Buck converter is exactly the same as average model. The author first applied averaging method for state equation (dynamic equation) and discrete method to output (static equation) equation.

From previous studies, the discrete-time method is same with averaging method for buck converter. Thus, in this thesis, for obtaining the mathematical model of buck converter, the state space averaging method is used.

2.3 State Space Averaging Method

The state space averaging method is firstly presented by Cuk & Middlebrook [11-12] in 1976, 1977. In this study, the same steps are taken based on [11-12], but some missing derivations and conclusions are driven based on linear theory.

State space averaging is the modeling structure which is given by [11]. This modeling is based on taking averaging of the two different mathematical models and produced only one model that has the same properties of each model. These two models are based on two different switch positions in the converter. In figure 2.2 switching positions are given *'ON'* and *'OFF'* and position times as *Td* (or *dT*) and *Td'* (or *d'T*), where *T* is the period, *d* is the duty and *d'* is equal to *1-d*.

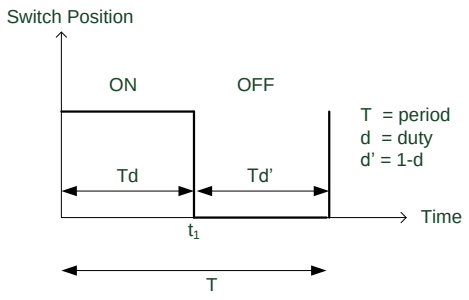


Figure 2.2 Switch Positions

Generally, each switching converter is presented as two linear equations which are belong to corresponding switching position; presented in the following table.

For the interval *Td*	For the interval *Td'*
$\dot{x}_1 = A_1 x_1 + b_1 vg$	$\dot{x}_2 = A_2 x_2 + b_2 vg$
$y_1 = C_1^T x_1$	$y_2 = C_2^T x_2$

The state equations are referred as dynamic (first row) and static (second row) equations respectively. To convert two equations into one, first the solutions of each differential equation should be obtained. Therefore, the response function is the solution of these differential equations (static equations are only the multiplication of a constant with the state vector. It is the reason why it is called static equation). Without taking the constant multiplier ($C_i, i = 1,2$) into account, state transition function, which is equal to response function can be found.

(Note that, for most converter matrix C is taken as equal to the identity matrix in theoretically. However, because of the implementation problems like output voltage interval is larger than an analog-to-digital converter voltage interval. Thus, there are the voltage dividers must be used so C will not be equal to the identity matrix)

The solutions of each interval are given below. *(Assume that the initial time instant is zero, so when the converter is in the ON position and OFF position means that the time is in [0,t_1] and [t_1,T] respectively)*

$$x_1(t_1) = e^{A_1 t_1} x_1(0) + \int_0^{t_1} e^{A_1 (t_1 - \tau)} b_1 v_g d\tau \tag{2.1}$$

$$x_2(T) = e^{A_2 (T - t_1)} x_2(t_1) + \int_{t1}^{T} e^{A_2 (T - \tau)} b_2 v_g d\tau \tag{2.2}$$

In figure 2.3, the exact and proposed averaged state space state solutions are given. The system composed from two different subsystem and these systems are used sequentially, which means after the first switch position the other one take

action and so on. But unlike these steps, the average system is wanted to act as single movement because of its overall characteristics.

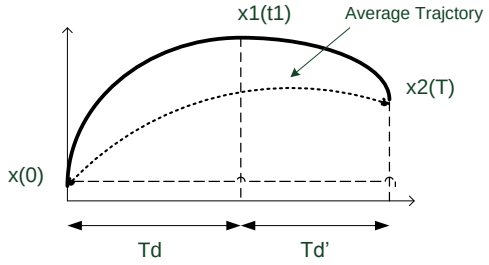


Figure 2.3 Solutions of State Variables and Approximate Solution for One Cycle

(Note that, in general state variables like speed, voltage, current, pressure, cannot change instantaneously) From figure 2.3 clearly at the time instant 't_1', $x_1(t_1) = x_2(t_1)$. By using this fact, equation 2.1 is replaced into equation 2.2 which results following equations.

$$e^{A_2(T-t_1)}x_1(t_1) = e^{A_2(T-t_1)}e^{A_1 t_1}x_1(0) + \int_0^{t_1} e^{A_2(T-t_1)}e^{A_1(t_1-\tau)}b_1 v_g d\tau \tag{2.3}$$

$$x_2(T) = e^{A_2(T-t_1)}e^{A_1 t_1}x_1(0) + \int_0^{t_1} e^{A_2(T-t_1)}e^{A_1(t_1-\tau)}b_1 v_g d\tau + \int_{t1}^{T} e^{A_2(T-\tau)}b_2 v_g d\tau \tag{2.4}$$

$$x_2(T) = e^{A_2(T-t_1)}e^{A_1 t_1}x_1(0) + b_1 v_g \int_0^{t_1} e^{A_2(T-t_1)}e^{A_1(t_1-\tau)}d\tau + b_2 v_g \int_{t1}^{T} e^{A_2(T-\tau)}d\tau \tag{2.5}$$

Known that $dT = (t_1 - 0)$ and $d'T = (T - t_1)$, replace these results into equation (2.5).

$$x_2(T) = e^{d'A_2 T}e^{dA_1 T}x_1(0) + b_1 v_g e^{d'A_2 T}\int_0^{t_1} e^{A_1(t_1-\tau)}d\tau + b_2 v_g \int_{t1}^{T} e^{A_2(T-\tau)}d\tau \tag{2.6}$$

$$x_2(T) = e^{(d'A_2 + dA_1)T}x_1(0) + b_1 v_g e^{d'A_2 T}\int_0^{t_1} e^{A_1(t_1-\tau)}d\tau + b_2 v_g \int_{t1}^{T} e^{A_2(T-\tau)}d\tau \tag{2.7}$$

In equation 2.7, there are nonlinear components existing and that makes the equation harder to solve. As the next step, some operations must be applied.

$$e^{AT} = I + AT + \frac{A^2T}{2!} + \frac{A^3T}{3!} + ... \tag{2.8}$$

Equation 2.8 is the power serial expansion of the exponential function. To reach an approximate value of the equation (2.7); only the first two expression of the power serial expansion is sufficient (because these are only one power expressions; the matrix *A* is composed from components like capacitor & inductor and frequency of the switching pulse is high enough like 100kHz). This process is called *linearization*. *(It is assumed that, the charge and discharge times for storage elements are smaller than the switching frequency)*

(Note that, all exponential functions are replaced with the first two parts of the right side of equation 2.8 accept $e^{d'A_2T}$ *expression. As mentioned before the first assumption is the switch is "ON" position initially, but if its position in "OFF" than that expression should be* $e^{d'A_1T}$ *. So that expression is taken as identity matrix)*

$$\begin{aligned} x_2(T) \cong\ & (I + dA_1T + d'A_2T)x_1(0) + b_1vg(I + d'A_2T)\int_0^{t_1}(I + (t_1 - \tau)A_1)d\tau \\ & + b_2vg\int_{t_1}^{T}(I + (T - \tau)A_2)d\tau \end{aligned} \tag{2.9}$$

$$\begin{aligned} x_2(T) \cong\ & (I + dA_1T + d'A_2T)x_1(0) + b_1vg(I + d'A_2T)(d'T + A_1d'^2T^2 - A_1d'T) \\ & + b_2vg(dT + A_2d^2T^2 - A_2dT) \end{aligned} \tag{2.10}$$

Equation 2.10 has second power expressions so if these equations are taken *0* the following equation is constructed (linearization, equation 2.8)

$$x_2(T) \cong (I + dA_1T + d'A_2T)x_1(0) + b_1vg(d'T) + b_2vg(dT) \tag{2.11}$$

$$x_2(T) \cong (I + dA_1T + d'A_2T)x_1(0) + vg\left[b_1(d'T) + b_2(dT)\right] \tag{2.12}$$

Note that, in spite of in figure 2.3 shows; the initial value and the final value of the state are different. This means that at each cycle, the initial state will be different. To get rid of this difficulty, we assume that overall system must repeat itself for every cycle, which means $x_1(0) = x_2(nT)$ and $x_1((n+1)T) = x_2(nT)$ where *n* is the number of cycle.

As final step, the forward difference is defined as differentiation operation:

$$\dot{x}(nT) = \frac{x((n+1)T) - x(nT)}{T} = \frac{x_2(T) - x_1(0)}{T} = \frac{(I + dA_1T + d'A_2T)x_1(0) + vg\left[b_1(d'T) + b_2(dT)\right] - x_1(0)}{T} \tag{2.13}$$

$$\dot{x}(nT) = (dA_1 + d'A_2)x(0) + vg(db_1 + d'b_2) \tag{2.14}$$

$$\dot{x}(nT) = (dA_1 + d'A_2)x(nT) + vg(db_1 + d'b_2) \tag{2.15}$$

The equation 2.15 is the basic state space averaging model of the switching converters. The averages of both state equation forms *basic state space average model* of the converters. As It can be seen from equations, the *d* and *Vg* is assumed to be constant. Now, general expressions for state space average model for changing line voltage and duty value is produced. *(Also note that the duty value is between [0, 1]).*

Now the following equations give the basic state space averaging model of the converter.

$$\begin{aligned} \dot{x} &= A\,x + b\,vg \\ y &= C^T x \end{aligned}, \quad \text{where} \quad \begin{aligned} A &= DA_1 + D'A_2 \\ b &= Db_1 + D'b_2 \\ C^T &= DC_1^T + D'C_2^T \end{aligned} \tag{2.16}$$

The duty value cannot stay in a constant value, it can change cycle-to-cycle by $d(t) = D + \hat{d}$ (Note that $D + D' = 1$ so $D' = D' - \hat{d}$) which makes the system time varying. Furthermore, the change in the line voltage result is depended on the duty, so the change in the duty will be resulted in the change of the voltage perturbation. By using the main equation which is the basic state space averaging (is given in equation 2.16) following sequence of equations are obtained.

$$(\dot{X} + \dot{\hat{x}}) = \left[(D + \hat{d})A_1 + (D' - \hat{d})A_2\right](X + \hat{x}) + \left[(D + \hat{d})b_1 + (D - \hat{d})b_2\right](Vg + \hat{v}g)$$

$$= \left(DA_1 + \hat{d}A_1 + D'A_2 - \hat{d}A_2\right)(X + \hat{x}) + \left(Db_1 + \hat{d}b_1 + D'b_2 - \hat{d}b_2\right)(Vg + \hat{v}g)$$

$$= \left((DA_1 + D'A_2) + \left(\hat{d}A_1 - \hat{d}A_2\right)\right)(X + \hat{x}) + \left((Db_1 + D'b_2) + \left(\hat{d}b_1 - \hat{d}b_2\right)\right)(Vg + \hat{v}g)$$
$$= (DA_1 + D'A_2)X + (DA_1 + D'A_2)\hat{x} + (Db_1 + D'b_2)Vg + (Db_1 + D'b_2)\hat{v}g + OT$$
$$= AX + bVg + A\hat{x} + b\hat{v}g + OT$$

where

$$OT = \left(\hat{d}A_1 - \hat{d}A_2\right)X + \left(\hat{d}A_1 - \hat{d}A_2\right)\hat{x} + \left(\hat{d}b_1 - \hat{d}b_2\right)Vg + \left(\hat{d}b_1 - \hat{d}b_2\right)\hat{v}g$$
$$= \left[(A1 - A2)X + (b1 - b2)Vg\right]\hat{d} + \left[(A1 - A2)\hat{x} + (b1 - b2)\hat{v}g\right]\hat{d}$$

$$\dot{\hat{x}} = AX + bVg + A\hat{x} + b\hat{v}g + \left[(A1 - A2)X + (b1 - b2)Vg\right]\hat{d} + \left[(A1 - A2)\hat{x} + (b1 - b2)\hat{v}g\right]\hat{d} \quad (2.17)$$

Similarly same process applied to static equation and the next equation is become as follows.

$$Y + \hat{y} = C^T X + C^T \hat{x} + (C_1^{\ T} - C_2^{\ T})X\hat{d} + (C_1^{\ T} - C_2^{\ T})\hat{x}\hat{d} \quad (2.18)$$

Equations 2.17 & 2.18 have high order terms, to get rid of these terms the fallowing assumptions are applied into that equation (*linearization*).

$$\frac{\hat{v}g}{Vg} << 1, \frac{\hat{d}}{D} << 1, \frac{\hat{x}}{X} << 1$$

which means that the perturbations in the system are very small also means that the following equation can be linearized. (It also means that the harmonic source in the system is vanished)

$$\dot{\hat{x}} = AX + bVg + A\hat{x} + b\hat{v}g + \left[(A1 - A2)X + (b1 - b2)Vg\right]\hat{d} \quad (2.19)$$

Equation 2.19 is constructed. By decomposition of equations 2.19 and 2.18 into AC and DC parts make the final state space averaging model of converters.

Steady State (DC) model

$$X = -A^{-1}bVg$$

$$Y = C^T X = -C^T A^{-1}bVg$$

Dynamic (AC) model

$$\dot{\hat{x}} = AX + bVg + A\hat{x} + b\hat{v}g + \left[(A1 - A2)X + (b1 - b2)Vg\right]\hat{d} \quad (2.20)$$

$$\hat{y} = C^T \hat{x} + (C_1^{\ T} - C_2^{\ T})X\hat{d} \quad (2.21)$$

Equations 2.20 and 2.21 are the state space averaging model of the switching converters.

2.4 Mathematical Model of Synchronous Buck Converter

In figure 2.4 there is the diagram of the basic synchronous buck converter.

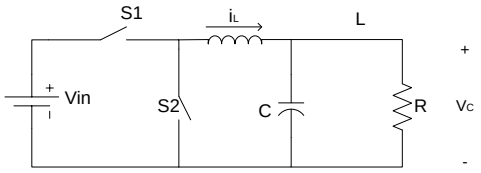
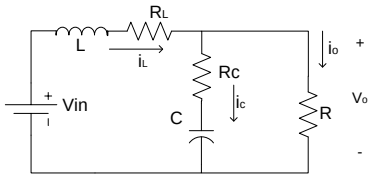
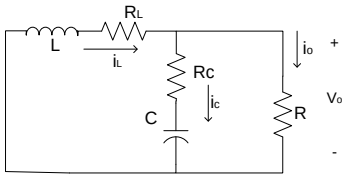


Figure 2.4 The Synchronous Buck Converter Diagram

To get the proper dynamic equation for buck converter, first define the two phase of switches and model each passed as discussed previous chapter. The states are defined as inductor current and capacitor voltage respectively.

S1 is “ON” and S2 is “OFF”	S1 is “OFF” and S2 is “ON”
	
$i_C = i_L - i_0$ $i_C = i_L - \frac{V_C + i_C R_C}{R}$ $i_C = i_L - \frac{V_C}{R} - \frac{i_C R_C}{R}$ $\left(1 + \frac{R_C}{R}\right) i_C = i_L - \frac{V_C}{R}$ $\left(\frac{R + R_C}{R}\right) C \frac{dV_C}{dt} = i_L - \frac{V_C}{R}$ $\frac{dV_C}{dt} = \left(\frac{R}{C(R + R_C)}\right) i_L - \left(\frac{1}{C(R + R_C)}\right) V_C$	$i_C = i_L - i_0$ $i_C = i_L - \frac{V_C + i_C R_C}{R}$ $i_C = i_L - \frac{V_C}{R} - \frac{i_C R_C}{R}$ $\left(1 + \frac{R_C}{R}\right) i_C = i_L - \frac{V_C}{R}$ $\left(\frac{R + R_C}{R}\right) C \frac{dV_C}{dt} = i_L - \frac{V_C}{R}$ $\frac{dV_C}{dt} = \left(\frac{R}{C(R + R_C)}\right) i_L - \left(\frac{1}{C(R + R_C)}\right) V_C$
and	and

$-V_{IN}+L\frac{di_L}{dt}+i_LR_L+i_CR_C+V_C=0$ $L\frac{di_L}{dt}=V_{IN}-i_LR_L-i_CR_C-V_C$ $L\frac{di_L}{dt}=V_{IN}-i_LR_L-C\frac{dV_C}{dt}R_C-V_C$ $\frac{di_L}{dt}=\frac{V_{IN}}{L}-i_L\frac{R_L}{L}-\frac{V_C}{L}-i_L\frac{R_CR}{L(R+R_C)}+V_C\frac{R_C}{L(R+R_C)}$ $\frac{di_L}{dt}=\frac{V_{IN}}{L}-i_L\left(\frac{R_L}{L}+\frac{R_CR}{L(R+R_C)}\right)-V_C\left(\frac{1}{L}-\frac{R_C}{L(R+R_C)}\right)$ State equation for this phase is given below $\dot{x}=\begin{bmatrix}-\left(\frac{R_L}{L}+\frac{R_CR}{L(R+R_C)}\right) & -\left(\frac{1}{L}-\frac{R_C}{L(R+R_C)}\right)\\ \left(\frac{R}{C(R+R_C)}\right) & -\left(\frac{1}{C(R+R_C)}\right)\end{bmatrix}x+\begin{bmatrix}\frac{1}{L}\\0\end{bmatrix}Vin$	$L\frac{di_L}{dt}+i_LR_L+i_CR_C+V_C=0$ $L\frac{di_L}{dt}=-i_LR_L-i_CR_C-V_C$ $L\frac{di_L}{dt}=-i_LR_L-C\frac{dV_C}{dt}R_C-V_C$ $\frac{di_L}{dt}=-i_L\frac{R_L}{L}-\frac{V_C}{L}-i_L\frac{R_CR}{L(R+R_C)}+V_C\frac{R_C}{L(R+R_C)}$ $\frac{di_L}{dt}=-i_L\left(\frac{R_L}{L}+\frac{R_CR}{L(R+R_C)}\right)-V_C\left(\frac{1}{L}-\frac{R_C}{L(R+R_C)}\right)$ State equation for this phase is given below $\dot{x}=\begin{bmatrix}-\left(\frac{R_L}{L}+\frac{R_CR}{L(R+R_C)}\right) & -\left(\frac{1}{L}-\frac{R_C}{L(R+R_C)}\right)\\ \left(\frac{R}{C(R+R_C)}\right) & -\left(\frac{1}{C(R+R_C)}\right)\end{bmatrix}x$

From previous table the fallowing matrix values are extracted and then the general state space equation will be composed depend on equations 2.20 and 2.21.

$$A_1=A_2=\begin{bmatrix}-\left(\frac{R_L}{L}+\frac{R_CR}{L(R+R_C)}\right) & -\left(\frac{1}{L}-\frac{R_C}{L(R+R_C)}\right)\\ \left(\frac{R}{C(R+R_C)}\right) & -\left(\frac{1}{C(R+R_C)}\right)\end{bmatrix},b_1=\begin{bmatrix}\frac{1}{L}\\0\end{bmatrix},b_2=0,C_1=C_2=I$$

From equation 2.20

$$\dot{\hat{x}}=(DA+D'A)X+b_1Vg+(DA+D'A)\hat{x}+b_1\hat{v}g+[(A-A)X+(b_1-0)Vg]\hat{d} \qquad (2.22)$$

Into equation 2.27 assume that the line voltage ($vg=Vin$) is rectified there is no AC components which means that $\hat{v}g=0$. (Note that $D+D'=1$)

$$\dot{\hat{x}}=AX+b_1Vg+A\hat{x}+b_1Vg\hat{d}$$

$$\dot{\hat{x}}=A(X+\hat{x})+b_1Vg(1-d)$$

$$\dot{x}=Ax+b_1Vgd(t)$$

and

$$\dot{x}=\begin{bmatrix} -\left(\frac{R_L}{L}+\frac{R_C R}{L(R+R_C)}\right) & -\left(\frac{1}{L}-\frac{R_C}{L(R+R_C)}\right) \\ \left(\frac{R}{C(R+R_C)}\right) & -\left(\frac{1}{C(R+R_C)}\right) \end{bmatrix} x+\begin{bmatrix} \frac{Vin}{L} \\ 0 \end{bmatrix} d(t) \tag{2.23}$$

So the state space average model for buck converter is constructed. Equation 2.23 gives the relation between duty and capacitor voltage, but the same relations is needed between duty and output voltage. Thus, the following equation 2.24 which gives the relation between output voltage and capacitor voltage should be used to obtain necessary equations.

$$\begin{aligned} V_0 &= V_C + i_C R_C \\ V_0 &= V_C + R_C C \frac{dV_C}{dt} \end{aligned} \tag{2.24}$$

2.4.1 Transfer Function of Synchronous Buck Converter

The state-space equation of the synchronous buck converter was obtained. In this subsection, the transfer function will be obtained by taking Laplace transform or the average equations which are given in equation 2.23.

$$\dot{i}_L = -i_L\left(\frac{R_L(R+R_C)+R_C R}{L(R+R_C)}\right)-V_C\left(\frac{R+R_C-R_C}{L(R+R_C)}\right)+d(t)\frac{V_{IN}(R+R_C)}{L(R+R_C)} \tag{2.25}$$

$$\dot{V}_C = i_L\left(\frac{R}{C(R+R_C)}\right)-V_C\left(\frac{1}{C(R+R_C)}\right) \tag{2.26}$$

$$\begin{aligned} sV_C(s)[C(R+R_C)] &= I_L(s)R - V_C(s) \\ V_C(s)[sC(R+R_C)+1] &= I_L(s)R \end{aligned}$$

Laplace transform of the equation 2.26 is calculated as equation 2.27.

$$V_C(s)\frac{[sC(R+R_C)+1]}{R} = I_L(s) \tag{2.27}$$

Following equations are obtained from taking Laplace transform of the equation 2.25.

$$[sL(R+R_C)]I_L(s) = -I_L(s)[R_L(R+R_C)+R_C R]-V_C(s)R+[V_{IN}(R+R_C)]d(s)$$

$$\left[sL(R+R_C)+R_L(R+R_C)+R_CR\right]I_L(s)=-V_C(s)R+\left[V_{iN}(R+R_C)\right]d(s) \tag{2.28}$$

Equation 2.27 is put into equation 2.28;

$$\left[sL(R+R_C)+R_L(R+R_C)+R_CR\right]\frac{\left[sC(R+R_C)+1\right]}{R}V_C(s)=-V_C(s)R+\left[V_{iN}(R+R_C)\right]d(s)$$

$$\left(\frac{\left[sL(R+R_C)+R_L(R+R_C)+R_CR\right]\left[sC(R+R_C)+1\right]}{R}+R\right)V_C(s)=\left[V_{iN}(R+R_C)\right]d(s)$$

$$\left(\frac{s^2L(R+R_C)^2+sL(R+R_C)+sR_LC(R+R_C)^2+R_L(R+R_C)+sR_CRC(R+R_C)+R(R+R_C)}{R}\right)V_C(s)$$
$$=\left[V_{iN}(R+R_C)\right]d(s)$$

$$\frac{V_C(s)}{d(s)}=\frac{RV_{iN}}{s^2L(R+R_C)+sL+sR_LC(R+R_C)+R_L+sR_CRC+R}$$

$$\frac{V_C(s)}{d(s)}=\frac{RV_{iN}}{s^2L(R+R_C)+s\left(L+R_LC(R+R_C)+R_CRC\right)+R_L+R}$$

$$\frac{V_C(s)}{d(s)}=\frac{RV_{iN}}{s^2L(R+R_C)+s\left(L+C\left(R_C(R+R_L)+R_LR\right)\right)+\left(R_L+R\right)}$$

The transfer function of $\frac{V_C(s)}{d(s)}$ is obtained in equation 2.29. However, $V_C(s)$ is the capacitor voltage. Thus, it must be defined transfer function for output voltage and inductor current, which is given in equation 2.30 and 2.31 respectively.

$$\frac{V_C(s)}{d(s)}=\frac{RV_{iN}}{(R+R_L)}\left(\frac{1}{s^2L\frac{(R+R_C)}{(R+R_L)}+s\left(\frac{L}{(R+R_L)}+C\left(R_C+\frac{R_LR}{(R+R_L)}\right)\right)+1}\right) \tag{2.29}$$

From equation 2.24

$V_0(s)=(1+sR_CC)V_C(s)$ is obtained and applied to equation 2.29 and equation 2.30 is obtained.

$$\frac{V_0(s)}{d(s)} = \frac{RV_{IN}}{(R+R_L)}\left(\frac{(1+sR_CC)}{s^2L\frac{(R+R_C)}{(R+R_L)}+s\left(\frac{L}{(R+R_L)}+C\left(R_C+\frac{R_LR}{(R+R_L)}\right)\right)+1}\right) \quad (2.30)$$

From equation 2.27, $\frac{I_L(s)}{d(s)}$ is obtained in equation 2.31.

$$\frac{I_L(s)}{d(s)} = \frac{V_{IN}}{(R+R_L)}\left(\frac{1+sC(R+R_C)}{s^2L\frac{(R+R_C)}{(R+R_L)}+s\left(\frac{L}{(R+R_L)}+C\left(R_C+\frac{R_LR}{(R+R_L)}\right)\right)+1}\right) \quad (2.31)$$

3. CONTROL OF SYNCHRONOUS BUCK CONVERTER

In this chapter, the control methodologies that used to control the buck converter and other power converters are reviewed. In general there are two types of the controller:: Hysteretic Controller and Pulse width modulation (PWM) controller. However, in the present, PWM is using frequently. Thus, the following layout which is shown in figure 3.1 is constructed based on PWM controllers. However, some Hysteretic controllers are also introduced.

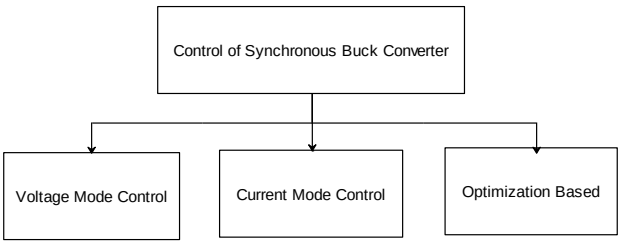


Figure 3.1 Overall Layouts of the Control Schemes for Power Converters

3.1 Voltage Mode Control

Voltage mode controllers (also known as direct duty ratio control) are the most common control structures which are used in power converters. General schematic of the voltage mode control is given in figure 3.2.

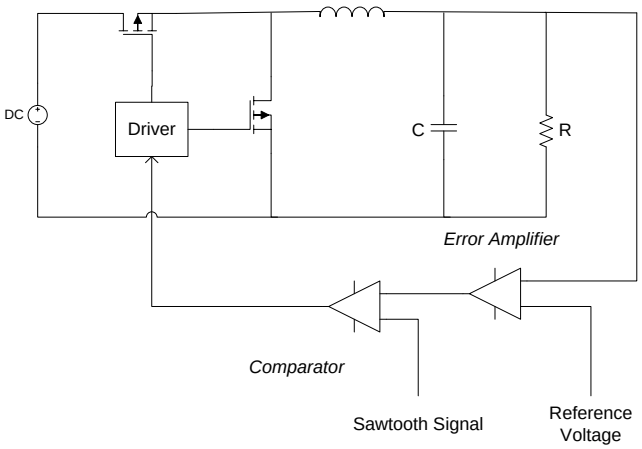


Figure 3.2 General Voltage Mode Control Scheme

In voltage mode control, the error amplifier compares the output voltage with the reference voltage and generates the error voltage. This error fed into comparator that compares the error voltage with saw tooth signal and the PWM is constructed with the same frequency with saw tooth signal. The advantage of this controller is easy to analyze. But, it is hard to design the compensator (phase-lead or phase-lag) for desired performance. Figure 3.3 gives the different voltage mode control methods for power converters.

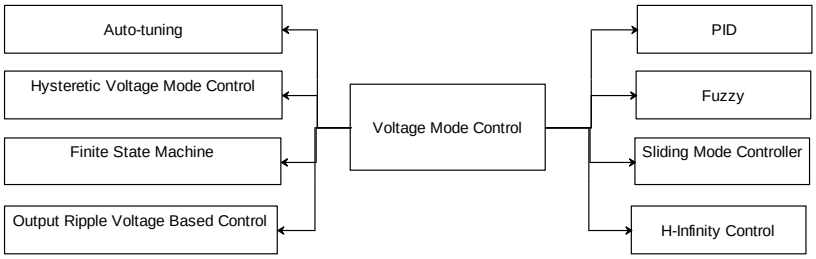


Figure 3.3 Voltage Mode Control Methods

Auto-tuning:

Auto-tuning controllers improve dynamic response efficiency and reliability [17]. However, they are rarely used because of their complex structure. The main idea of auto-tuning is presented as: first system identification is executed, then the controller parameters are tuned. In [17], limit cycle oscillation is introduced and used to decide the PID parameters. By using the controller which is proposed [17], the oscillations on the output voltage are removed when the load variation happens. Furthermore, in [18], adaptive tuning system is applied to power converter. In that study PID controller is also selected as the controller.

PID:

In general form, the PID controller has the following transfer function

$$G_C(s) = K_P + \frac{K_I}{s} + K_D s$$

The terms K_P, K_I and K_D are called proportional, integral and derivative terms respectively. The PID controller is a form of the phase lead-leg compensator with one pole at the origin and other at infinity [1]. The aim of this controller is stabilized

the system by adjusting the controller terms by using root locus and frequency response methods. There are various methods are introduced to adjust the controller terms. In this thesis, a new adjustment rule applied by using PSO algorithm.

Hysteretic Voltage Mode Control:

Two different types of hysteretic voltage mode controller based on implementation are defined: Analog Hysteretic Voltage Mode Controller and Digital Hysteretic Voltage mode controller. The digital Hysteretic voltage mode is discussed in [19] and compared with traditional PID controller. However, the determination of the PID coefficients is not discussed. Although, the PID controller results from the steady state error in load variation, the digital hysteretic voltage mode controller gives more noisy output.

Fuzzy Logic Controller:

During the last decade, there has been an increase in the application of fuzzy logic controllers (FLC) for control of static power converters. FLC as an intelligent nonlinear controller has been applied successfully for control of power converters. Power converters can be controlled in ’voltage mode control’ or ’current mode control’ options. Most of the applications of FLC for power converters are ’voltage mode control’. In [4, 20], 7 membership functions and 49 rules are defined to control the power converter. FLC is implemented on DSP [20] and 8-bit microcontrollers [4]. Figure 3.4 shows the block diagram of the power converter system with FLC controller.

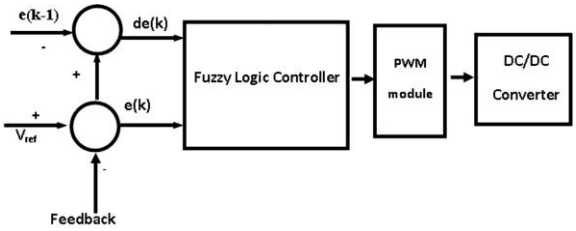


Figure 3.4 General Fuzzy Logic Controller Scheme

Finite State Machine:

This control structure is generally used for FPGA implementations. Like other discrete event systems (DES) the output voltage is divided into the different levels and state diagram is defined. In [21] 4-level controller is investigated. There isn't any transient response discussion found in literature but in [21] the controller is the response against load and line variations is given. However, the controller produces the steady state error for load change.

Output Ripple Voltage Based Control:

In literature, three different controllers are proposed for "output ripple voltage based controller" as given in figure 3.5: V^2 control, Derivative-output ripple voltage (DOR) control and end-point prediction control (EPP).

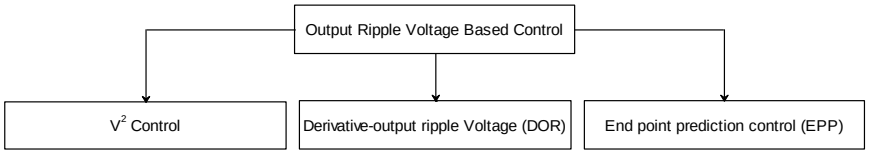


Figure 3.5 Output Ripple Voltage Based Control Methods

In V^2 control (shown in figure 3.6), the output voltage is used as both generate ramp signal to use for obtain PWM signal and error voltage. The compensated error voltage and ramp signal is compared and then PWM signal is constructed.

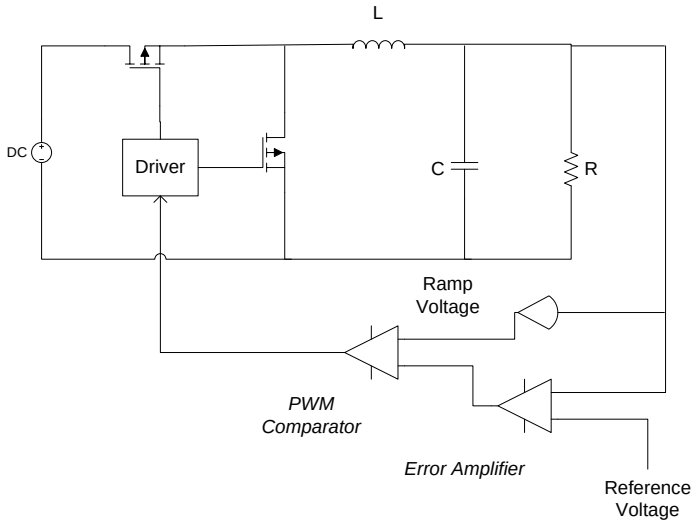


Figure 3.6 General V^2 Control Scheme

The idea behind the controller is summarized as; when the load variation happens, the output current will affect the output voltage thus the dc level of the ramp signal is changed and duty is also changed. This process improved the load transient response of the system. Actually, the V^2 control is same as the peak current mode controller and the same disadvantages is observed (sub harmonic oscillations). Also, slow transient response has been observed [22].

The DOR control scheme is defined to improve fast transient response when the load variation happened and EPP defined to improve the reference tracking speed of the converter.

These controller performances are depended on the ESR of the capacitor. Thus, the controller design is based on the power converter parameters. Also, the transient response of the controller isn't given in [22].

Sliding Mode Controller (SMC):

Sliding mode controller is a nonlinear control structure which is used for variable structure systems. SMC is robust against non-modeled dynamics, load and line variations and don't necessary to actual mathematical model. However, SMC is

suffered from infinitely frequency switching control input, which is caused “chattering”. To get rid of this phenomena fixed frequency SMC is proposed [23].

H∞ Control:

Robust tracking is applied to power converters. The control objective is to regulate a high frequency ripple signal to robustly track a constant reference signal [24]. The simulation results show promising results but this controller hasn’t been implemented yet.

3.2 Current Mode Control

Generally, current mode control is a two-loop control structure which used the inductor, capacitor or switches current as the inner current control and voltage controller defined as feedback voltage signals. This kind controllers are generally applied for boost, buck-boost controller. In current mode control, the voltage loop is generally simplified and system stability (phase margin) and control dynamics are improved. Figure 3.7 shows the general diagram of the current mode controller.

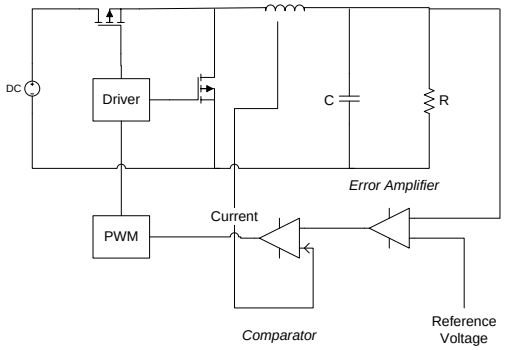


Figure 3.7 General Current Mode Control Scheme

Generally, there are two main current mode controller is used: Peak current mode controller and average current mode controller. In the present, average current mode controller is frequently chosen because of its stability against change in the duty and immune to current noise. Figure 3.8 presents various current mode methods applied for power converters.

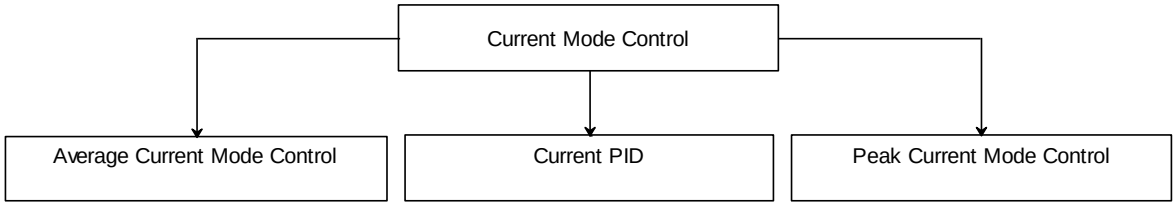


Figure 3.8 Output Ripple Voltage Based Control Methods

Average Current Mode Control:

In average current mode control [25], the output voltage is compared and compensated to construct the reference current signal. This signal is compared with the current (inductor, capacitor or switch) and current error signal is obtained. This signal is compensated and PWM signal is obtained. This method stable against duty values larger than 0.5 and because of the average of the current is taken the controller has good noise immunity. However, this method is more complex than peak current mode controller and two compensators must be designed.

Peak Current Mode Control:

The aim of the peak current mode control is to force the current (inductor, capacitor or switch) to track the reference current signal which is constructed the outer voltage loop. When the current peak value reaches to the reference current signal, the switch position is the change to ‘OFF’. This current controller isn’t used in present because of two reasons: 1) The sense of current is influenced from the noise. 2) The instability because of the duty values which is larger than 0.5. 3) Unwanted response is obtained because of the noisy peak value of the current.

Current PID:

The current PID is the conventional PID controller but for current mode control as explained before, two controllers are needed. For current PID, the inner and outer controllers are both defined as PID. In [26], the current PID controller is compared with the digital two switching cycle compensation algorithm. This new controller is also used the current PID controller but some innovations are constructed as logical expressions. The new algorithm is the response much better than PID

controller when the line variation exists. However, there is no load variation reported. However, it is expected to the response same with current PID controller. The current PID controller scheme is given in figure 3.9.

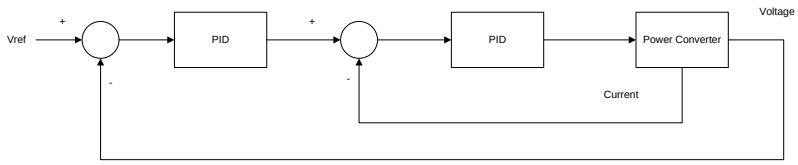


Figure 3.9 Current PID Controller Block Diagram

3.3 Optimization Based

The main idea of the optimization based controllers are to obtain optimum power converter and compensator parameters with same time or distinctively. There is only one study is made to determine optimum converter and controller parameters at the same time. This constrained optimization problem is called control-structure optimization [27]. Furthermore, only the controller coefficients are obtained. This parameter optimization is based on covariance control theory (CCT). The results from [27] shows that, the transient response of the optimized controller is not sufficient, there are very high overshoot is observed.

3.4 Final Comments on Control of Converters

In literature, more than 300 papers (*in Scopus with key words 'control' and 'buck converter'*) are scanned and previous control methods are classified. In the present, researches are continued for both current mode control and voltage mode control. The main problem of the current mode control is the noise in the current and the very complex control structure. However, also, in current mode control the voltage controller (outer loop) is not focused enough. Furthermore, current mode control can only be important only for fixed current applications like charging circuits. On the other side, the voltage mode controller is easy to implement but also using voltage is cheaper than using current, because for current sense circuit must be used. In this thesis, the voltage mode controllers are investigated because of the following reasons.

In both voltage and current mode controllers, one important controller is used frequently, this is PID controller. PID controller not only used for comparison of proposed controllers, but also they use as a part of the novel control techniques. Another important part is other controllers like sliding mode or fuzzy logic controllers the implemented like PI, PD or PID controller structure.

As the conclusion, in this thesis the voltage mode controller is used because of their basic nature, noise immunity, no current sense circuit necessary and there is still discussed the voltage mode controllers existed. From voltage mode controllers, only sliding mode controller, fuzzy logic controller is different than others. The other controllers are implementable only in FPGA, or they are actually a variation of the PID controllers. However, the PID parameter estimations are not discussed enough. Thus in this thesis, PID controller, sliding mode controller, fuzzy logic controller and proposed neural controller are investigated detailed by simulation. In literature, determination of the unknown parameters of PID is obtained by frequency analysis, determination of the unknown parameters of sliding mode is obtained by geometrical approaches and determination of the unknown parameters of the fuzzy controller is obtained from trial and error experiments. However, in this thesis, an evolutionary optimization algorithm (Particle swarm optimization) is used to the determination of the unknown parameter for all controllers.

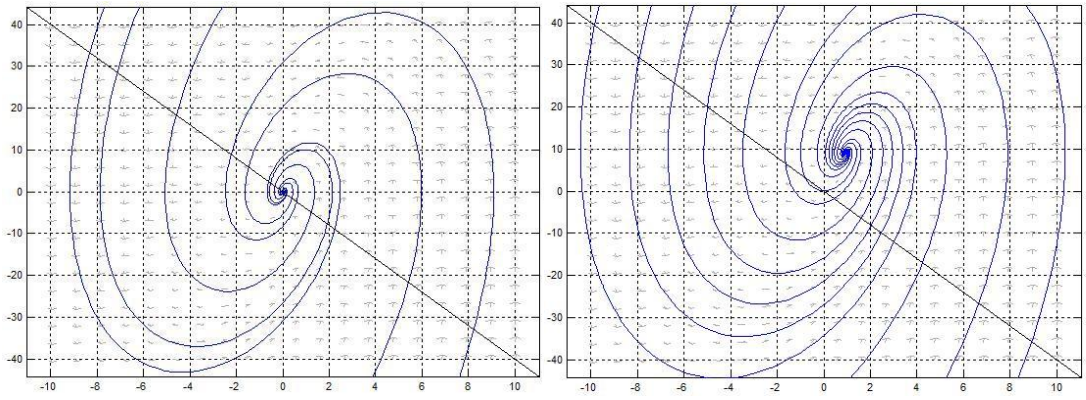
4. SIMULATION OF SYNCHRONOUS BUCK CONVERTER

In this chapter SMC, FLC and PID techniques are applied to synchronous buck converter. The results are investigated in detailed. The aim of this chapter is present the simulation results of various control techniques. The unknown parameters are obtained from conventional ways and also by converting and solving as an optimization problem.

4.1 Sliding Mode Controller

DC/DC converters have been successfully controlled for many years using analog integrated circuit and digital controllers. Especially the utilization of digital control techniques to control of power converters has more and more increased. SMC offers several advantages in comparison to traditional control methods: stability, even for large line and load variations, robustness, good dynamic response and simple implementation. Application of SMC is well found in many areas of power electronics. Power electronic converters are inherently bound in variable structure systems for the fact that the structure of the converter system changes in every control action. Theory of SMC to DC/DC converters have been widely investigated in literature. Although, the SMC has advantages over external disturbance and un-modeled dynamics, the controller is suffered from several disadvantages. First, for SMC, it is assumed that the control input sends signals infinite frequency. However, in practice, it is impossible to change states in that fast, thus ‘chattering’ always exists. Second, the frequency of the input doesn’t fix so this phenomena makes the input and output filter design harder.

The design of SMC for synchronous buck converter is begun with the phase plane of the converter model which is defined in chapter 2. Figure 4.1 shows the phase portrait of the synchronous buck converter. The sliding surface (the line on the graphic) is defined as given in figure 4.1. But, for *Switch 2* is *ON* and *Switch 1* is *OFF*, it could be impossible to hit the state trajectory to the line. Thus, that makes the design of the SMC based on states that are given in chapter 2 is improper (which are inductor current and output voltage).



(a) (b)

Figure 4.1 Phase Portrait of the Synchronous Buck Converter Modal (a) *Switch 1* is *ON* and *Switch 2* is *OFF* (b) *Switch 2* is *ON* and *Switch 1* is *OFF*

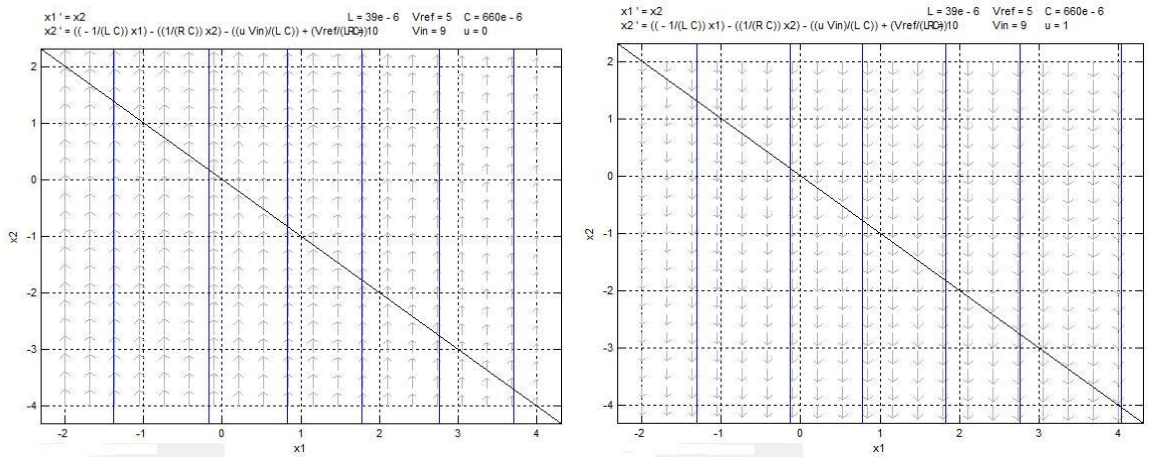
The new mathematical model is obtained based on state space averaging as described in chapter 2. But, this model is proposed with two state inputs defined as $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, where x_1 is the error between reference voltage and output voltage and x_2 is the differentiation of the error defined as $x_2 = \dot{x}_1$.

$$x_1 = V_{ref} - V_{out}$$
$$x_2 = \dot{x}_1 = -\frac{dV_{out}}{t}$$

where, V_{ref} is the reference voltage and V_{out} is the output load resistor voltage. One of the important points of this approach is the state definition. These states are defined as error and change of error. The same approach is defined in fuzzy systems either. The state space equation of the synchronous buck converter is given below. (*Note that, ESR resistors is taken zero and derivation of this equation is not given because, it is straightforward*)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{1}{LC} & -\frac{1}{RC} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{V_{in}}{LC} \end{bmatrix} u + \begin{bmatrix} 0 \\ \frac{V_{ref}}{LC} \end{bmatrix} \tag{4.1}$$

Now, the new phase plane is investigated for new system as given in equation 4.1.



(a) (b)

Figure 4.2 Phase Portrait of the Synchronous Buck Converter Modal as given in Equation 4.1 (a) *Switch 2* is *ON* and *Switch 1* is *OFF* (b) *Switch 1* is *ON* and *Switch 2* is *OFF*

Figure 4.2 presents the phase plane of the error state equation of the synchronous buck converter. From figure 4.2, the states are hit and by switching the circuit, the states will stay in the line. The equation 4.1 gives the error space equation of synchronous buck converter. In general, the proposed SMC rule is defined as follows.

$$S = \alpha x_1 + x_2$$
$$and \qquad (4.2)$$
$$u = \begin{cases} 1, & 'ON', \quad when \quad S > 0 \\ 0, & 'OFF', \quad when \quad S < 0 \end{cases}$$

where u is the control signal, S is sliding surface, x_1 & x_2 are states of plant. The equation 4.2 presents the variable frequency SMC for buck converter. But, it is hard to be implemented on DSP because of “chattering” phenomena. The “chattering” is caused by unmodaled dynamics or discrete implementation, and it results from low control accuracy and hard wear of hardware of the synchronous buck converter [28]. In general, there are coupled of methods are used to chattering suppression. First, observer-based chattering suppression is used to prevent chattering. The idea of the observer-based chattering is based on the estimate the state instead of measure. Second, state-dependent gain method is

used to change the switching gain by this way; the amplitude of the chattering is reduced. Third, equivalent-control-dependent gain method is designed with the idea of reduce the switching gain by using the average value of the sign of the state. Last method, chattering frequency control using the hysteresis loop is the common method which is used in power converters [9, 23, 29]. In this thesis, the fixed frequency SMC is proposed based on the idea of Lee & Utkin [28] “any method would be helpful to reduce chattering if it can decrease the switching gain effectively”. The proposed control action is defined as;

$$S = \alpha x_1 + x_2$$
$$S = C^T x$$
$$and$$
$$u = d(t) = \begin{cases} d(t-1) + u_{\max}, & when \quad S > 0 \\ d(t-1) - u_{\max}, & when \quad S < 0 \end{cases}$$

To guarantee the sliding existence condition the following equation can be satisfied.

$$\lim_{s \to 0} s\dot{s} < 0$$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & \frac{1}{} \\ -\frac{1}{LC} & -\frac{1}{RC} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{V_{in}}{LC} \end{bmatrix} u + \begin{bmatrix} 0 \\ \frac{V_{ref}}{LC} \end{bmatrix}$$
$$\dot{x} = Ax + Bu + D$$

$$u = d(t) = \begin{cases} d(t-1) + u_{\max}, & when \quad S > 0 \\ d(t-1) - u_{\max}, & when \quad S < 0 \end{cases}$$
$$d(t) = d(t-1) + u_{\max} Sign(S)$$
$$u = d(t) = d(0) + u_{\max} Sign(S) + u_{\max} \sum_{k=1}^{t-1} Sign(S(k))$$
$$average\left(u_{\max} \sum_{k=1}^{t-1} Sign(S(k)) \right) = 0$$
$$u = d(0) + u_{\max} Sign(S)$$

$$s = C^T x$$
$$\dot{s} = C^T x$$
$$\dot{s} = C^T \left(Ax + Bu + D \right)$$
$$s\dot{s} = s\left[C^T \left(Ax + Bu + D \right) \right] < 0$$
$$s\dot{s} = s\left[C^T \left(Ax + B\left(d(0) + u_{\max} Sign(S) \right) + D \right) \right] < 0$$
$$s\left(C^T Ax + C^T Bd(0) + C^T D \right) + C^T Bu_{\max} |s| < 0$$
$$Ax + Bd(0) + D < -Bu_{\max}$$
$$Ax + Bd(0) + D > Bu_{\max}$$
$$Bu_{\max} < Ax + Bd(0) + D < -Bu_{\max}$$

$$\begin{bmatrix} 0 \\ -\frac{u_{\max} V_{in}}{LC} \end{bmatrix} < \begin{bmatrix} x_2 \\ -\frac{x_1}{LC} - \frac{x_2}{RC} \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{d(0) V_{in}}{LC} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{V_{ref}}{LC} \end{bmatrix} < \begin{bmatrix} 0 \\ \frac{u_{\max} V_{in}}{LC} \end{bmatrix}$$

$$\begin{bmatrix} 0 \\ -\frac{u_{\max} V_{in}}{LC} \end{bmatrix} < \begin{bmatrix} x_2 \\ -\frac{x_1}{LC} - \frac{x_2}{RC} - \frac{d(0) V_{in}}{LC} + \frac{V_{ref}}{LC} \end{bmatrix} < \begin{bmatrix} 0 \\ \frac{u_{\max} V_{in}}{LC} \end{bmatrix}$$

$$-\frac{u_{\max} V_{in}}{LC} < -\frac{x_1}{LC} - \frac{x_2}{RC} - \frac{d(0) V_{in}}{LC} + \frac{V_{ref}}{LC} < \frac{u_{\max} V_{in}}{LC}$$

$$-u_{\max} V_{in} < -x_1 - \frac{x_2 L}{R} - d(0) V_{in} + V_{ref} < u_{\max} V_{in}$$

$$-u_{\max} V_{in} < -V_{ref} + V_{out} - \frac{x_2 L}{R} - d(0) V_{in} + V_{ref} < u_{\max} V_{in}$$

$$-u_{\max} V_{in} < V_{out} - \frac{x_2 L}{R} - d(0) V_{in} < u_{\max} V_{in}$$

$$-u_{\max} < \frac{V_{out}}{V_{in}} - \frac{x_2 L}{R V_{in}} - d(0) < u_{\max}$$

$\frac{x_2 L}{R V_{in}}$ is a very small quantity.

$$-u_{\max} < \frac{V_{out}}{V_{in}} - d(0) < u_{\max}$$

For reducing the chattering $u_{\max}$ is selected very small and the aim to make $V_{out} = V_{ref}$ *where* $\mathrm{d}(0)$ is the initial duty value which is equal to steady state duty $= \frac{V_{ref}}{V_{in}} = 0.56$

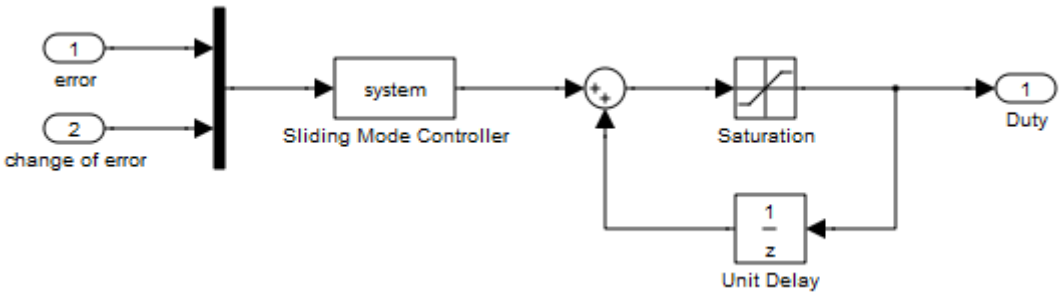


Figure 4.3 Sliding Mode Controller for the Synchronous Buck Converter

where α is sliding coefficient, *u* is the control input and *S* is the sliding surface, d(t) is the duty of the PWM signal and $u_{\max}$ is the small constant duty value which is selected as 0.01. Previous equation gives the SM control rule for buck converter. (*And, the sliding coefficient* $\alpha = \frac{1}{RC}$ *is found from authors in [9]. However, the simulation results aren't given in [9]. And also the sliding coefficient is obtained from the phase plane with the geometrical approach. But, this coefficient is not optimal.)* Thus, as innovation the optimal coefficient is found by using the optimization algorithm. First, the output voltage graphic is given from the various SMC sliding parameters. Then the result is investigated and the optimal sliding coefficient is obtained and new coefficient value and new output voltage graphic will be given and discussed.

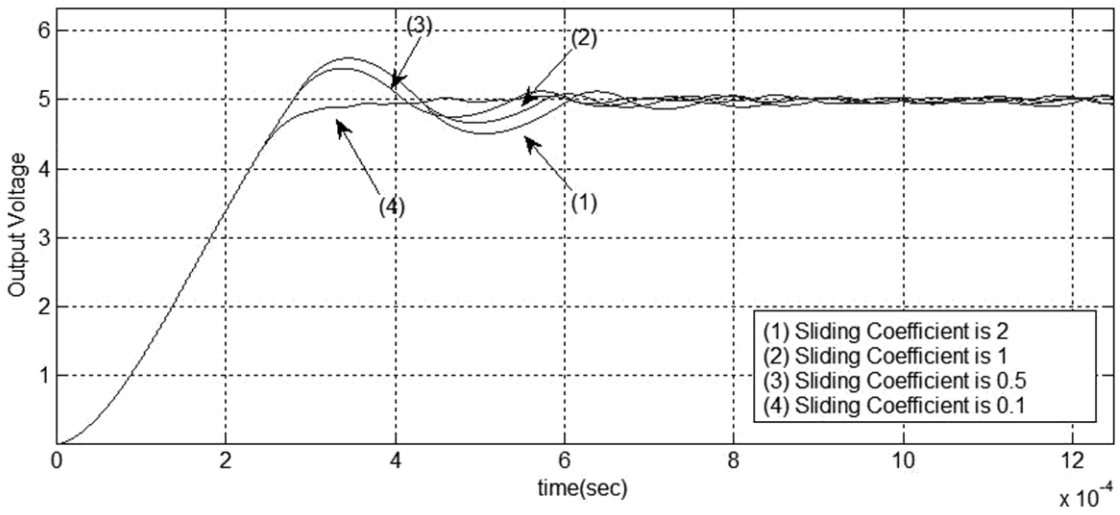


Figure 4.4 The Output Voltage of the Synchronous Buck Converter for Various Sliding Coefficient

Figure 4.4 presents the effect of the sliding coefficient into output voltage of the system. From figure 4.4, it is observed that, not only the overshoot and undershoot, but also the oscillation at the steady state is effected from sliding coefficient. Thus, the optimum coefficient is critical for system performance.

By using the Particle Swarm Optimization algorithm the optimal sliding coefficient is obtained by using evolutionary function, which is given in equation 4.3. In this thesis same function is used to find the optimum values.

$$J = t\sum_{k} e^2(k) \tag{4.3}$$

where, $e(k) = x_1(k) = V_{ref} - V_{out}(k)$ is error at time instant *k* and *t* is the coefficients which is selected large integer as 100.

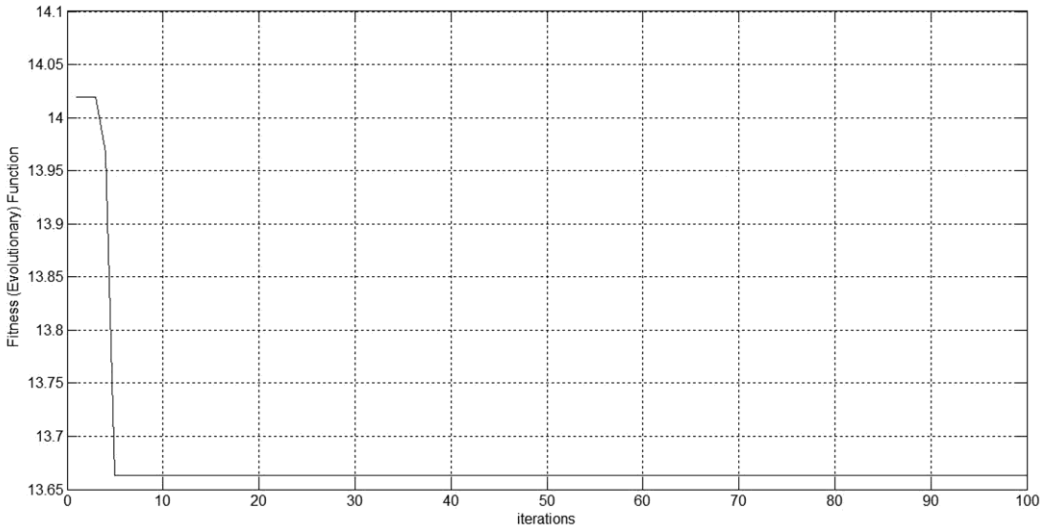


Figure 4.5 Error Convergence of the Optimization Algorithm

Figure 4.5 presents the error convergence of the optimization algorithm. As shown in figure the algorithm is detected the optimum only in 5 iterations.

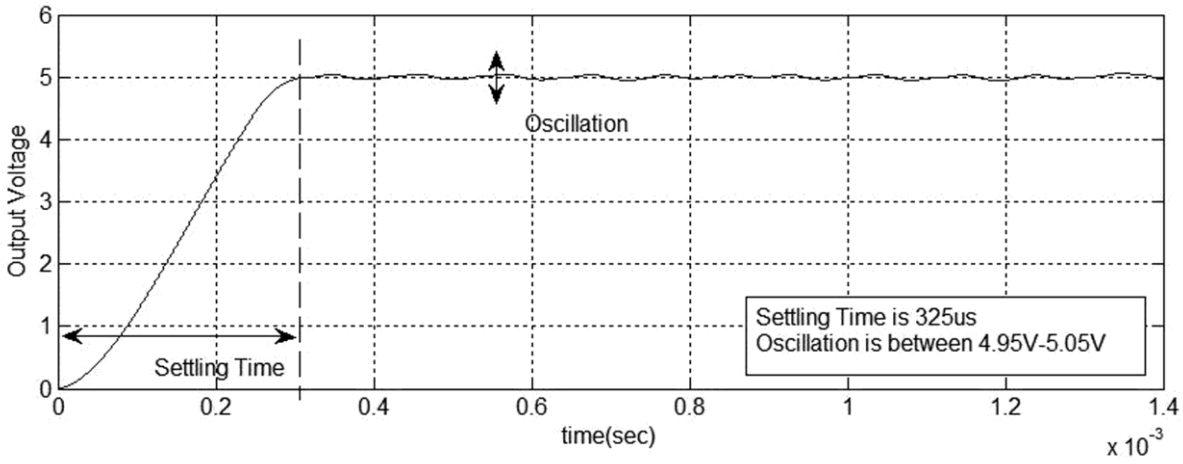


Figure 4.6 The Output Voltage Graphic after PSO is applied, where $\alpha = 0.1527$

Figure 4.6 shows the output voltage graphic of the Particle Swarm Optimized SMC. Although the overshoot is removed from the graphic, the oscillation is observed between 4.95V and 5.05V which is an acceptable performance. The settling time of the system is 325μs.

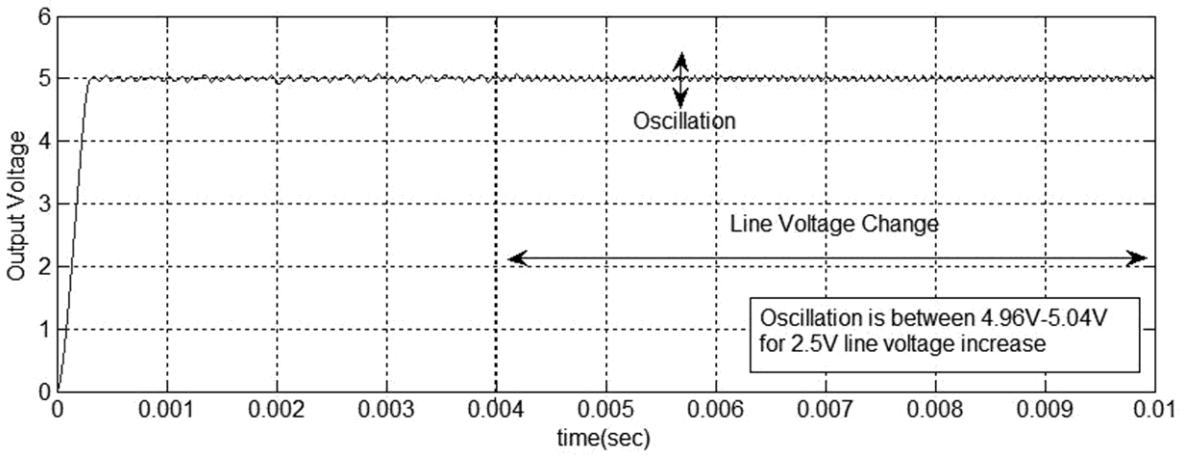


Figure 4.7 Transient Response of the System under Line Voltage Change

Figure 4.7 shows the response of the system for line fluctuation from 9V to 11.5V. The larger oscillation is observed from the simulation that voltage is change as oscillation from 4.96V to 5.04V.

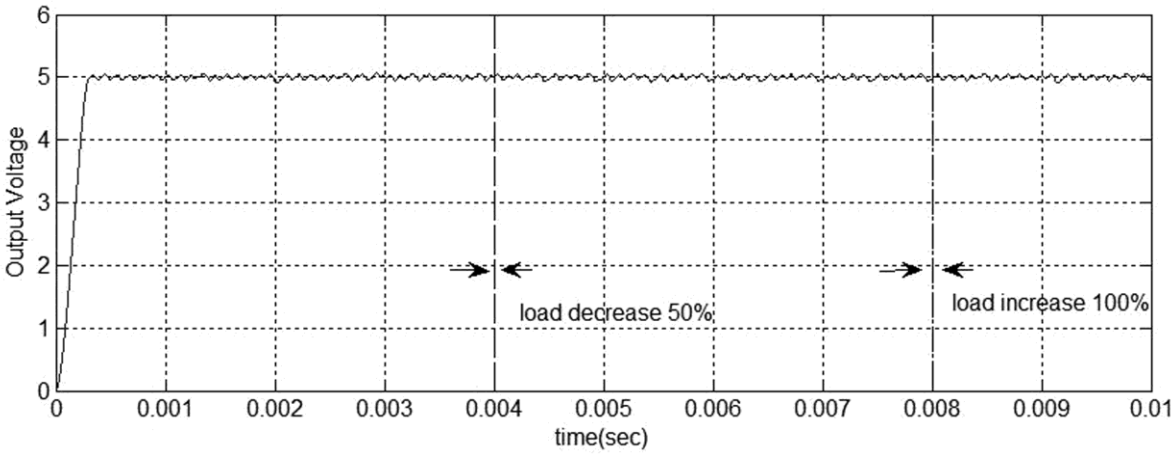


Figure 4.8 Transient Response of the System under Line Variation to Increase and Decrease 50% and 100% respectively

Figure 4.8 shows the system performance under %50 load decrease (10Ω to 5Ω) and %100 load increase (5Ω to 10Ω) (Same change is applied to all simulation and experimental studies). Simulation results show that the system is not changed under load variation. Table 4.1 represents the general properties of the SMC based on the simulation results.

Table 4.1 Summary for SMC Simulation Results; System Performance under Various Events and Performance Criteria

Event	**Performance Criteria**	**Sliding Mode Voltage Controller Performance**
Transient Response	Overshoot	0V
	Settling Time	325μs
Line Variation	Voltage Change	Oscillation
	Recovery Time	-
Load Variation	Voltage Change	-
	Recovery Time	-

4.2. Fuzzy Logic Controller

As a traditional nonlinear controller, Fuzzy Logic Controller (FLC) contains four main sub-block: 1) The fuzzification step which converts input real values into fuzzy information; 2) The fuzzy if-then rule table that contains the expert's control knowledge; 3) The inference mechanism that evaluates which control rules are activated based on current input variables; and 4) The defuzzification interface that converts the fuzzy inference result into a real value for application to controlled plant.

Generally, for control of the power converter, mamdani fuzzy inference system with seven triangular fuzzy sets with 49 (table 4.2) rules are selected for each variable regarded to 2 input variable as provided in (which are shown in figure 4.9 and figure 4.10). The range of input and output values are defined in [-1,1].

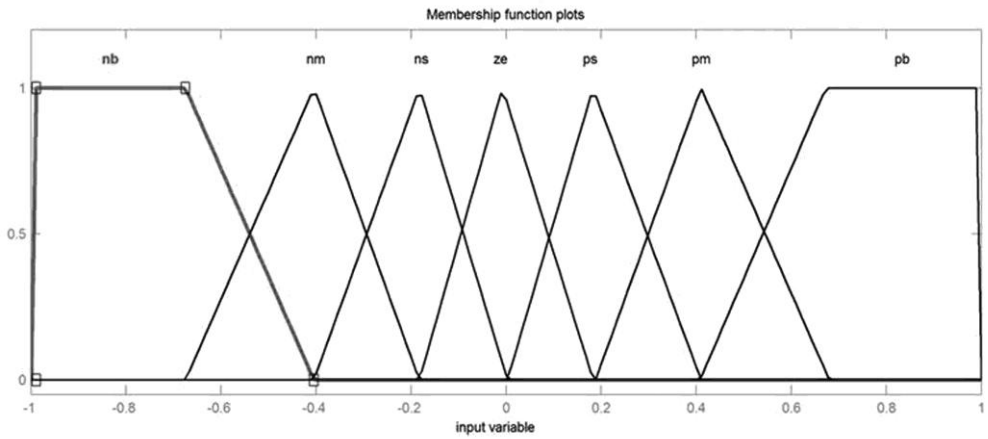


Figure 4.9 Membership Functions for Input Variables

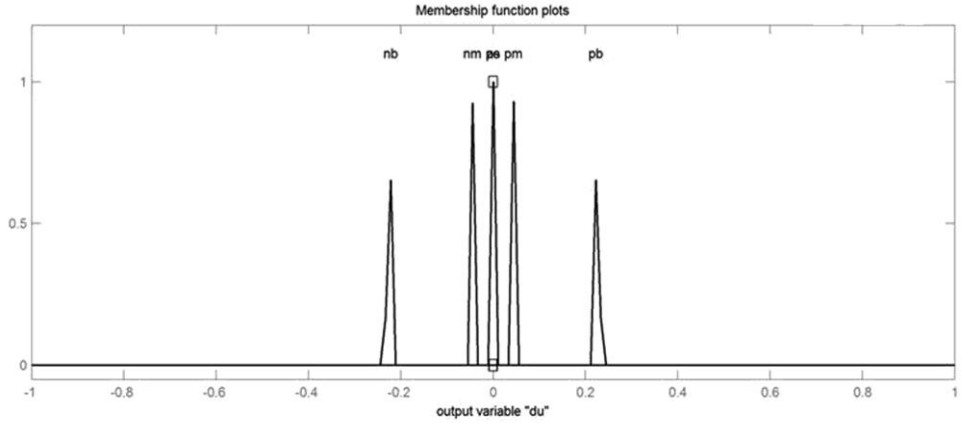


Figure 4.10 Membership Functions for Output Variable

Table 4.2 Rule Table for Fuzzy System

***de*/e**	**NB**	**NM**	**NS**	**ZE**	**PS**	**PM**	**PB**
NB	NB	NB	NB	NB	NM	NS	ZE
NM	NB	NB	NB	NM	NS	ZE	PS
NS	NB	NB	NM	NS	ZE	PS	PM
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PM	PB	PB
PM	NS	ZE	PS	PM	PB	PB	PB
PB	ZE	PS	PM	PB	PB	PB	PB

In previous studies FLC has been applied for control of the power converter in voltage mode control. FLC regulates output voltage by a controller with two inputs and one output. These variables are: output voltage error; change of this error and change of the duty cycle as output. The value of the duty cycle determines output voltage. Like other control systems output voltage errors are the main input of FLC. As shown in figure 3.4 errors is calculated with comparison of sampled Vo[k] and reference voltage value. The second input is the difference between successive errors. These relations are given in equation 4.4. The second input (change of error) can be determined by discrete-time derivative approximation. *k* index implies current error and *k-1* denotes the previous error.

$$\begin{aligned} e[k] &= V_{ref} - V_0[k] \\ de[k] &= e[k] - e[k-1] \end{aligned} \tag{4.4}$$

As the given, duty cycle for control of switching element is changed with respect to input variables and fuzzy inference algorithm. As shown in figure 4.9, input variables are multiplied by the scaling factors (scaling factors are used to normalized input vector between [-1,1]. These variables can be optimized but in this thesis only *h* parameter is optimized.) g_0 and g_1 respectively (which are selected as 0.5 and 1), and then fed into the fuzzy controller. The selection of these scaling factors directly depends on defined fuzzy set ranges. In many applications, the range of universal fuzzy sets has been selected from -1 to +1. Thus, the g_i coefficients (where i=0, 1) are the linear coefficients for scaling or normalization. At the end of the FLC, the output (Δd[k] =change of duty) is differentiated. The output of the fuzzy controller is the change in the duty cycle Δd[k], which is multiplied by a parameter *h*. The selection of *h* has a considerable effect on performance of the controller. In many studies, this value is selected based on trial and error. (The variation of V_0 regard to various values of *h* is given in figure 4.12).

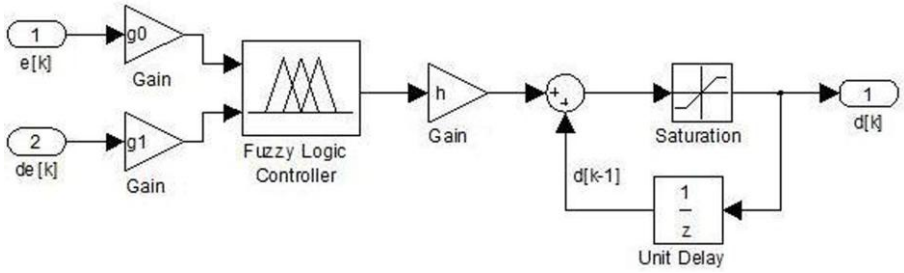


Figure 4.11 Calculation of PWM Duty Value after FLC

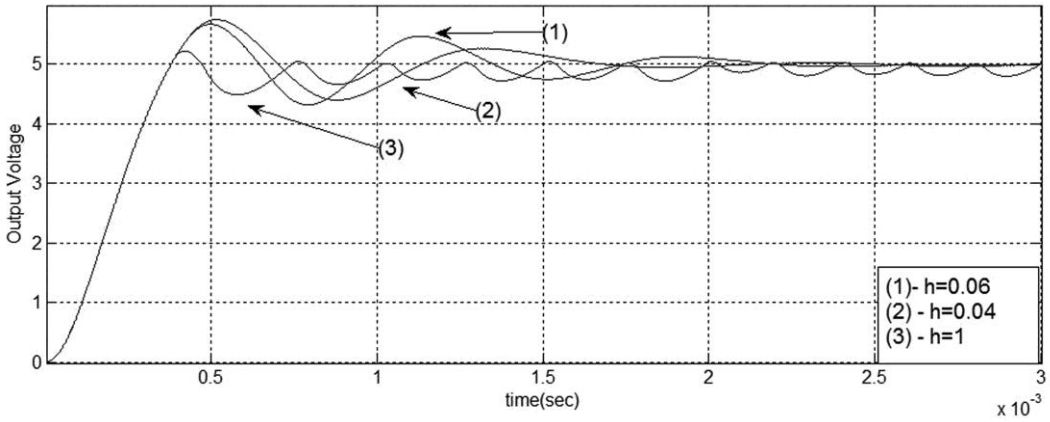


Figure 4.12 Output Voltage for Different *h* Values

As given in figure 4.11, the new duty cycle can be calculated from Δd[k] at the output of FLC. The fuzzy controller output Δd[k] is multiplied by the output gain h, and then added to the previous duty cycle d[k-1]. As shown in equation 4.5, this is discrete time integration of the fuzzy controller output and the integration block is serial to FLC block.

$$d[k] = d[k-1] + h\Delta d[k] \tag{4.5}$$

Table 4.3 summarizes the effect of gain parameter *h*. The performance of controller in term of steady state error and transient response time can be affected by parameter *h*.

Table 4.3 Effects of Gain Parameter h

Effects of gain parameter h
• Serial with FLC • Reduced steady state error • High values tends to oscillation • Very low values slow down the transient response

The aim of the PSO is the minimization of the steady state error and gets a sufficient transient response. Thus, the *h* parameter is calculated by PSO algorithm and the value of h=0.0338915 was obtained. In figure 4.14, transient response of the system is given. From the transient response graphic, it is observed that 1.85ms settling time with 0.8V overshoot and 0.6V undershoot is observed. Figure 4.15 shows the response of the system under 100 percent load increase and 50 percent load decrease. The system is recovered in 1ms and the voltage raises 0.15V and 0.1V respectively.

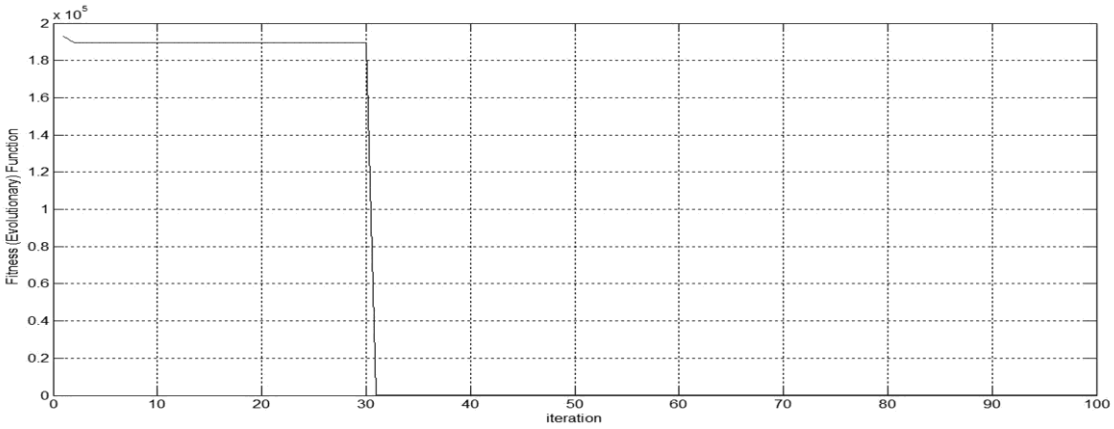


Figure 4.13 Error Convergence of the Optimization Algorithm

Figure 4.13 presents the error convergence of the optimization algorithm. As shown in figure the algorithm is detected the optimum only in 30 iterations.

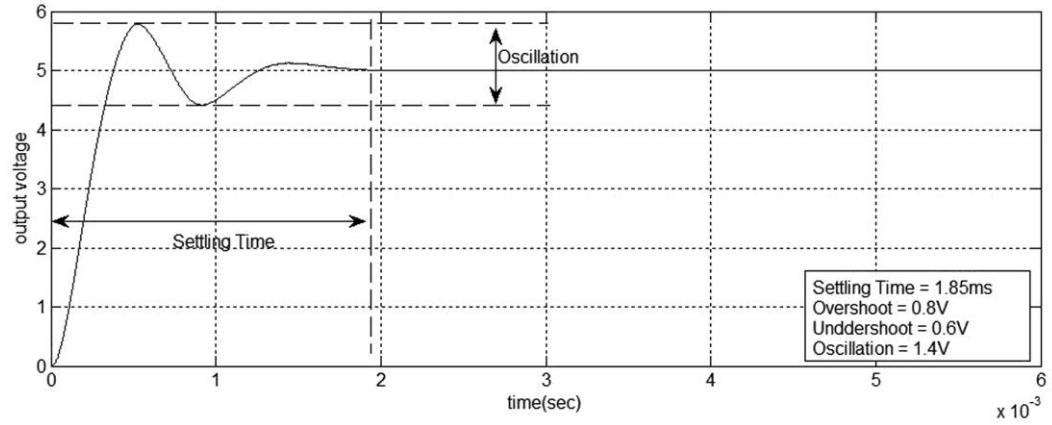


Figure 4.14 Transient Response of Optimized Controller

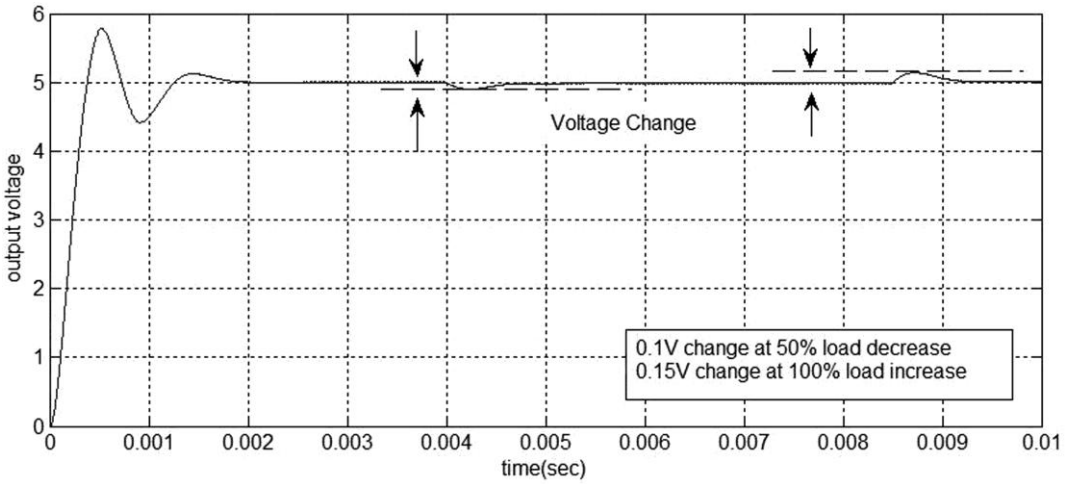


Figure 4.15 Response of Controller under Load Variation

From figure 4.16, the system response under 2.5V increased in the line voltage is seen. The system is recovered in less than 1ms with 0.55V raised without any steady state error and Table 4.4 presents the summary of the FLC based on the simulation results.

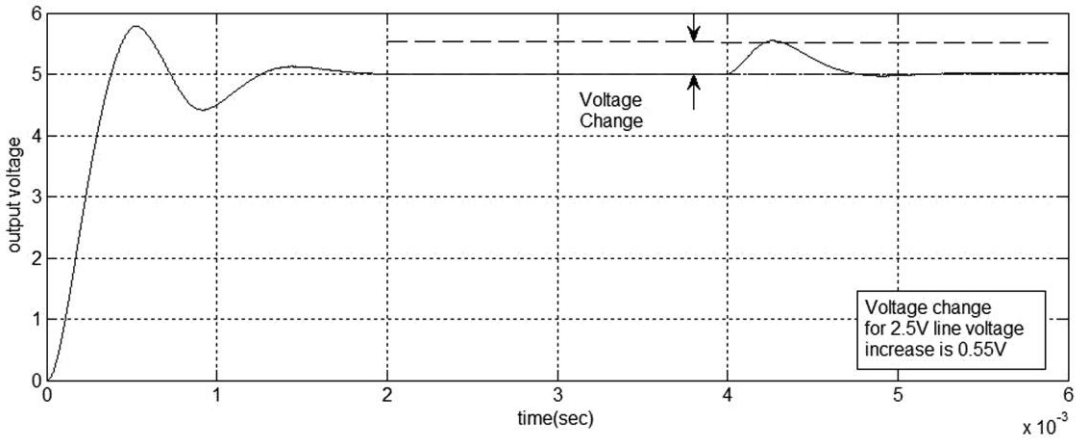


Figure 4.16 Response of Controller under 2.5V Line Variation

Table 4.4 Summary for FLC Simulation Results; System Performance under Various Events and Performance Criteria

Event	**Performance Criteria**	**Fuzzy Logic Voltage Controller Performance**
Transient Response	Overshoot	0.8V
	Settling Time	1.85ms
Line Variation	Voltage Change	0.55V
	Recovery Time	<1ms
Load Variation	Voltage Change	0.1V & 0.15V
	Recovery Time	<700μs

4.3 PID Controller

In general, for synchronous buck converter, PID controller (is sometimes referred as a lead-leg compensator) is designed by using frequency response and root locus methods. The system is controlled with sufficient phase margin. The PID controller is a two pole and one left plane zero. Figure 4.17 shows the pole/zero map of the transfer function (equation 2.30) of the synchronous buck converter. One zero is at *-2.02e+3* and complex conjugate poles are located at $-8.91 \pm 157i$.

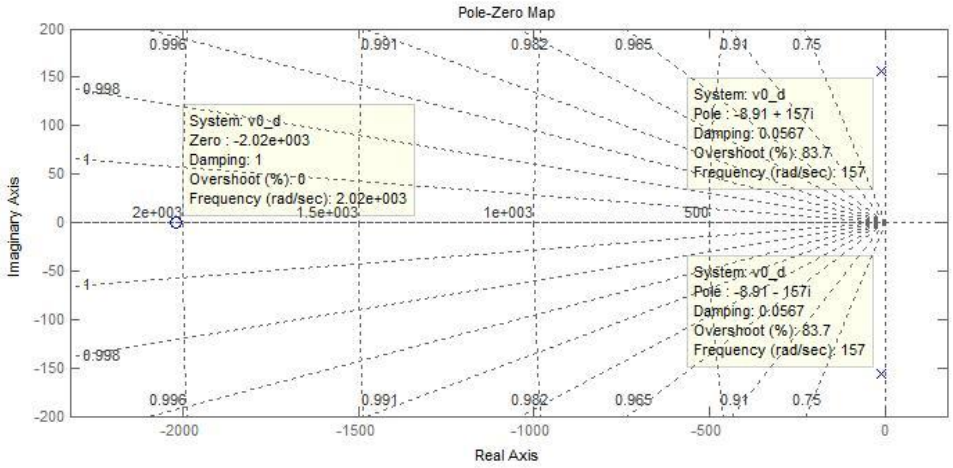


Figure 4.17 Pole/Zero Map of the Open-loop Transfer Function of the Synchronous Buck Converter

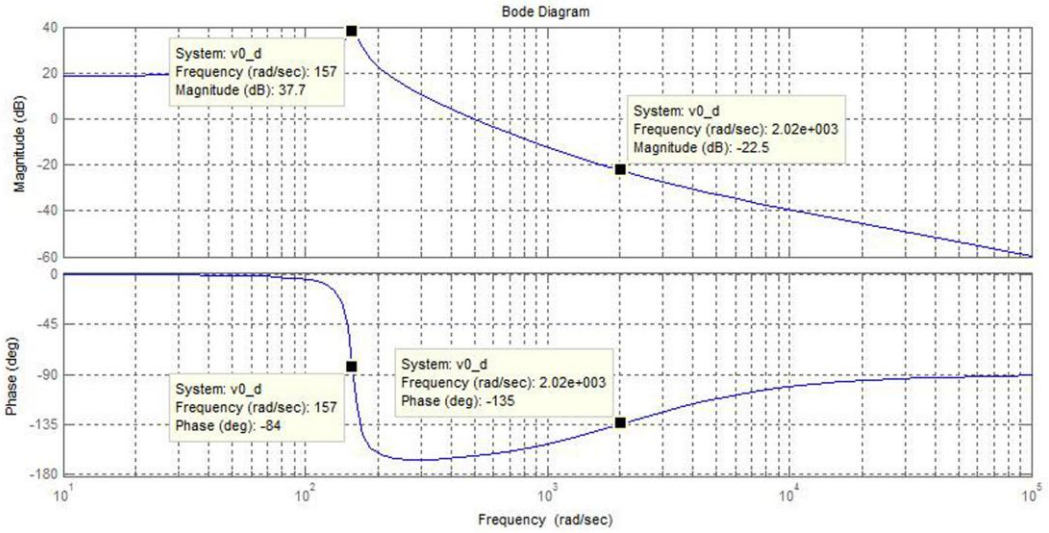


Figure 4.18 Bode Diagram of the Open-loop Transfer Function of the Synchronous Buck Converter

Figure 4.18 shows the bode diagram of the transfer function of the synchronous buck converter. System poles are located at frequency 157 rad/sec, magnitude 37.7 dB and phase -84 deg. The system zero is located at frequency 2.02e3 rad/sec, magnitude -22.5 dB and phase -135 deg.

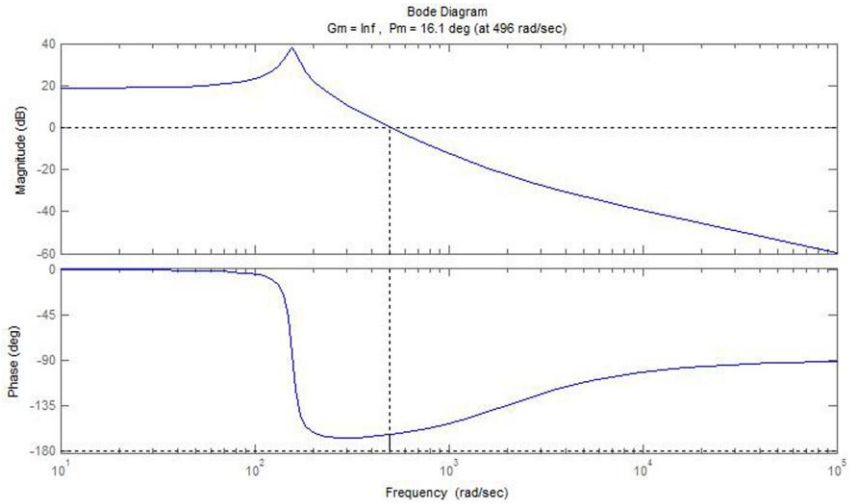


Figure 4.19 Bode Diagram of the Open-loop Transfer Function of the Synchronous Buck Converter shows Phase and Gain Margins

There are two criteria are used to defined stability of the linear time invariant systems: Gain Margin (GM) and Phase Margin (PM). Figure 4.19 presents the PM and GM of the transfer function of the synchronous buck converter. If the system $|GM|<1$ or $|GM|<0dB$ is represents the unstable system. The system with $GM=1(0dB)$ is called neutrally stable. The system to be stable positive PM is necessary. For second order system, it is common that the GM is equaled to infinity because the system does not cross -180º as shown in figure 4.19. In general, the PM is more commonly used to specify the control system.

In general form, the PID controller has the following transfer function

$$G_C(s)=K_P+\frac{K_I}{s}+K_D s$$

In time domain the equation is defined as follows

$$u(t)=K_P e(t)+K_I\int e(t)dt+K_D\frac{de(t)}{dt}$$

K_D, K_I, K_P are called proportional, derivative and integral terms respectively. When $K_D=0$ the controller is called PI controller.

$$G_C(s) = K_P + \frac{K_I}{s}$$

When $K_I = 0$ the controller is called PD controller.

$$G_C(s) = K_P + K_D s$$

The popularity of PID controllers is because of their good performance in a wide range of operating conditions and their functional simplicity and straightforward manner. The PID controller can be consider as follows

$$G_C(s) = K_P + \frac{K_I}{s} + K_D s = \frac{K_D s^2 + K_P s + K_I}{s} = \frac{K_D\left(s^2 + as + b\right)}{s} = \frac{K_D\left(s + a_1\right)\left(s + a_2\right)}{s}$$

The PID controller has the transfer function with one pole at the origin and two zeros can be located anywhere. The root locus and frequency response methods are used to select PID controller gains. Furthermore, some PID tuning schemes are proposed in literature.

The most common tuning method is the Ziegler/Nichols (ZN) method. This method is based on system's open-loop step response and later the authors were introduced new ZN method is based on frequency response of the close-loop system. It first assumed the system is under proportional control only. The proportional parameter is increased until the system became critically stable. Then the parameter and oscillation period is recorded. Based on these values, for a given table (can be found any Control Theory book) other parameters are determined. However, this method is not satisfactory for many systems, since it does not give satisfactory phase and gain margins. Additionally, ZN method is not easy to apply processes are the original form on working system. When critical processes are involved, sudden changes in the control signal or operation at the stability limit are not acceptable [30].

PSO is applied for PID tuning [31-38]. In [31], PSO-PID method is used for AVR system and compared with Genetic Algorithm based PID controller (GA-PID). It showed that PSO is more efficient to GA. One of the most popular usages of the PSO-PID is with cooperation of Fuzzy system for an ensured adaptive response of

PID controller [32-35]. In these studies, the optimum PID parameters are determined off-line and these parameters are applied to the system with Fuzzy Logic Controller (generally Mamdami Fuzzy Inference System is used) based on the system changes. However, the researches based on PSO-PID are generally for simulation, or only PC based implementations exist [33, 36], but there aren't any microcontroller implementation and problems are discussed and also there isn't any study to implement PSO-PID into Buck Converter existed.

In this thesis, first the PID controller is obtained by using ZN and by using the optimization algorithm called Particle Swarm Optimization. These two controllers will be compared based on transient responses.

ZN vs. PSO:

The transfer function of the synchronous buck converter is defined in chapter 2

$$\frac{V_0(s)}{d(s)} = \frac{RV_{IN}}{(R+R_L)}\left(\frac{(1+sR_CC)}{s^2L\frac{(R+R_C)}{(R+R_L)}+s\left(\frac{L}{(R+R_L)}+C\left(R_C+\frac{R_LR}{(R+R_L)}\right)\right)+1}\right)$$

The ZN method is applied to synchronous buck converter model which is given in figure 4.20.

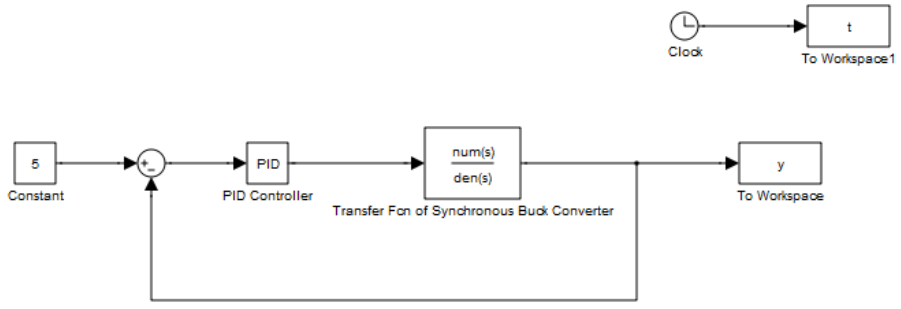


Figure 4.20 The Synchronous Buck Converter Controller

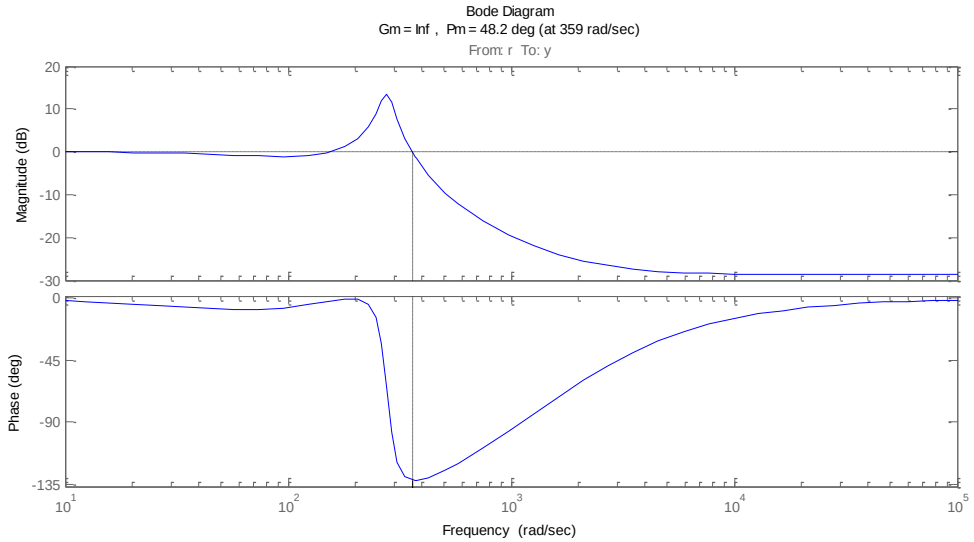


Figure 4.21 Bode Plot of the Closed-loop System with NZ tuned

Figure 4.21 shows the bode diagram of the closed-loop system with NZ tuned PID controller. As expected, the GM of the system is infinity, but the PM is less that 50º which is not desired.

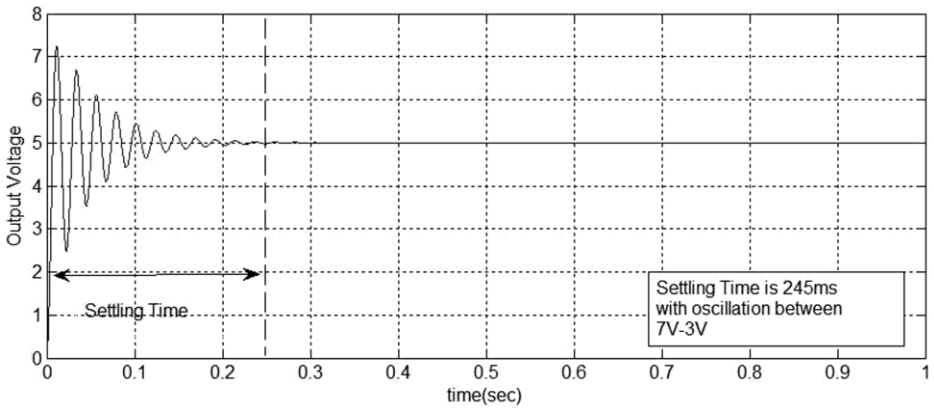


Figure 4.22 Output Voltage of the Synchronous Buck Converter by ZN Parameters

Figure 4.22 shows the output voltage off the synchronous buck converter where the PID parameters are determined from ZN method. As shown in figure 4.22 that the oscillations are existed. Same simulation is executed by using parameters that are determined from PSO algorithm.

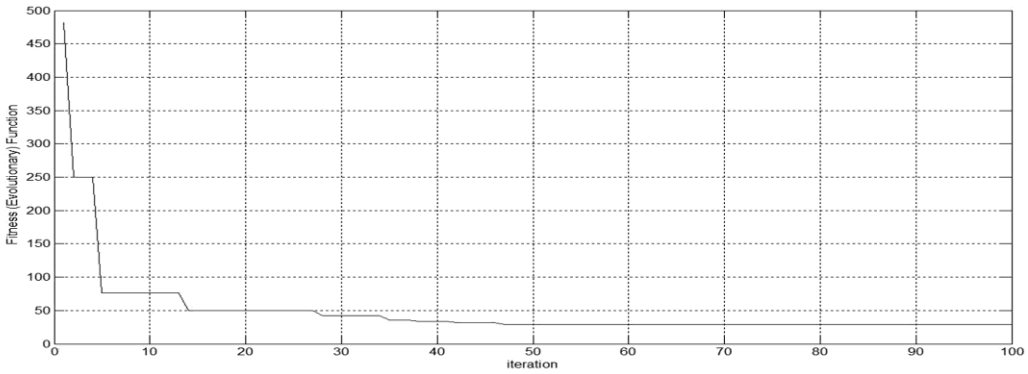


Figure 4.23 Error Convergence of the Optimization Algorithm

Figure 4.23 presents the error convergence of the optimization algorithm. As shown in figure the algorithm is detected the optimum only in 50 iterations.

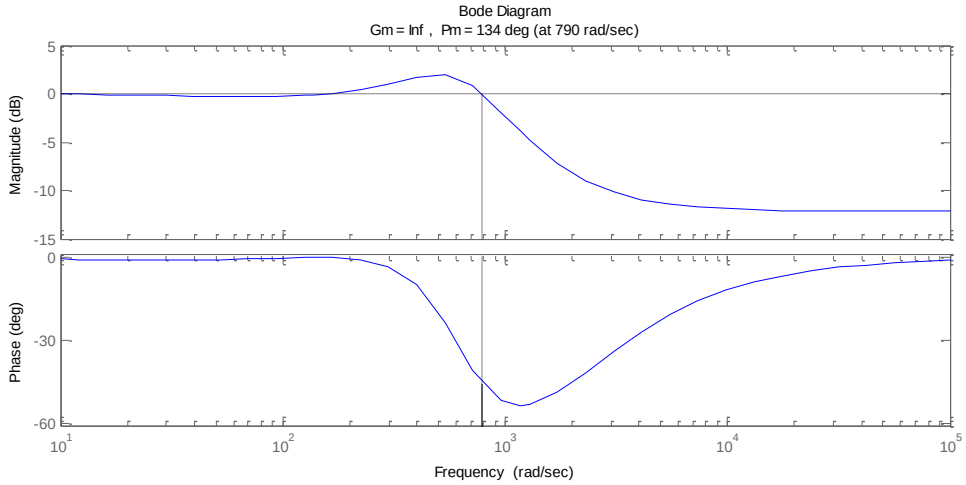


Figure 4.24 Bode Plot of the Closed-loop System with PSO tuned

Figure 4.24 shows the bode diagram of the closed-loop system with PSO tuned PID controller. As expected, the GM of the system is infinity, but the PM is 134º.

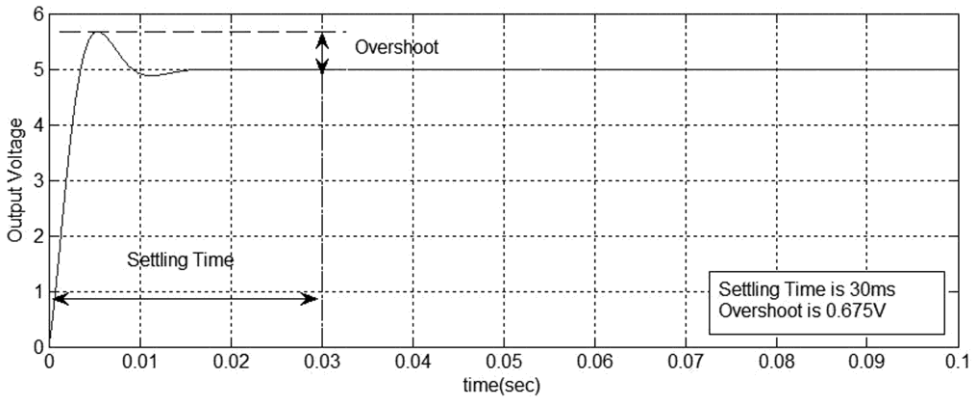


Figure 4.25 Output Voltage of the Synchronous Buck Converter by PSO Parameters

As shown in figure 4.25, it is obvious that the PSO is much better than the NZ method. Thus, it is observed that PSO gives better performance. As second phase, the discrete time PID is applied to synchronous buck converter, which is modified for implementation such as, ADC and PWM modules are added to model (This same model is used for all simulations for SMC and FLC either).

Thus, as second phase the PSO algorithm is used for discrete PID controller to determine the PID terms. When PID control is implemented in digital devices, it can be obtained in the discrete form (Z-Transform) with Trapezoidal approximation (equation 4.5) of continuous-time PID controller.

$$s \to \frac{1-z^{-1}}{T} \tag{4.5}$$

$$U(z) = K_P E(z) + K_I \frac{T}{1-z^{-1}} E(z) + \frac{K_D}{T}(1-z^{-1})E(z)$$

$$U(z) = \left[K_P + K_I \frac{T}{1-z^{-1}} + \frac{K_D}{T}(1-z^{-1}) \right] E(z)$$

$$U(z) = \left[\frac{K_P(1-z^{-1})T + K_I T^2 + K_D(1-z^{-1})^2}{T(1-z^{-1})} \right] E(z)$$

$$U(z) = \frac{(K_P T + K_I T^2 + K_D) + (-K_P T - 2K_D)z^{-1} + K_D z^{-2}}{T(1-z^{-1})}$$

$$K_A = (K_P + K_I T + \frac{K_D}{T})$$

$$K_B = (-K_P + 2\frac{K_D}{T})$$

$$K_C = \frac{K_D}{T}$$

For computer simulation above equations are sufficient enough but for interact in a real-world design following equations are obtained.

$$U(z)(1-z^{-1}) = \left[K_A + K_B z^{-1} + K_C z^{-2} \right] E(z) \tag{4.6}$$

$$u(n) = u(n-1) + K_A e(n) + K_B e(n-1) + K_C e(n-2) \tag{4.7}$$

In the environment, the differential equations for Buck Converter, the PID controller (equation 4.7) and same signal processing blocks like ADC are used. By using PSO method the unknowns of the PID controller (K_A, K_B and K_C) are determined. In figure 4.26, the discrete PID environment is presented.

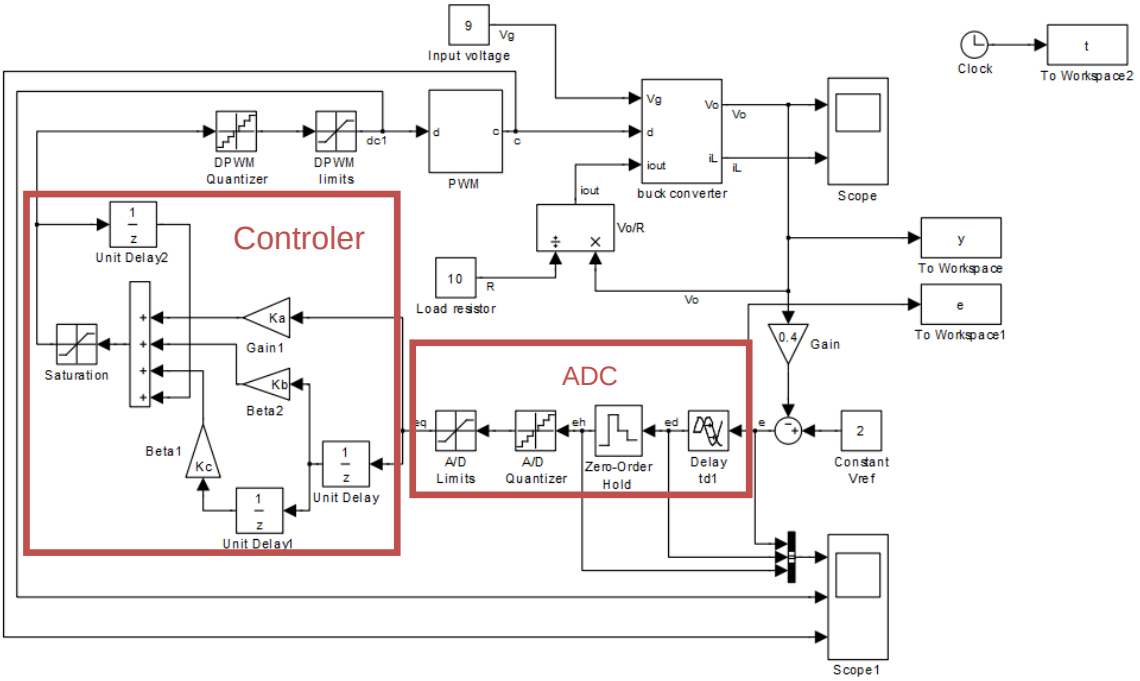


Figure 4.26 The Controller Diagram for Discrete PID Controller applied for Synchronous Buck Converter

PID parameter determination by using PSO is given by step by step (figure 4.27):

STEP 1 Initialization:

First the size of the swarm is determined, which is 25 (from empirical studies the common range of size of the swarm is between 20 and 500). The PSO parameters are assigned as c_1=c_2=2.05 and k=0.7289 [21-23]. Each particle has its own velocity and position. At the beginning, a random number is assigned to each particle between [100, 150] as initial velocity and between [1, 2] as initial position.

STEP 2 Best Position of Particle:

For each particle, the best position so far is recorded in p_{id} based on the evaluation function as shown in equation 19. For each value of each particle is recorded into the temporary variable and then compared at each iteration, if the new value is smaller than previous one the smaller one is hold. The system is executed for each particle. Then the output voltage is used for fitness function.

STEP 3 Best Particle in Swarm:

For all particle, the best particle and its position is recorded in p_{gd} .

STEP 4 Velocities and Position Update:

Velocities and positions of each particle are updated by using equation 13.

STEP 5 Stopping Criteria:

If the maximum number of iteration is reached than break the loop and end the program. In this study the maximum iteration is determined as 100.

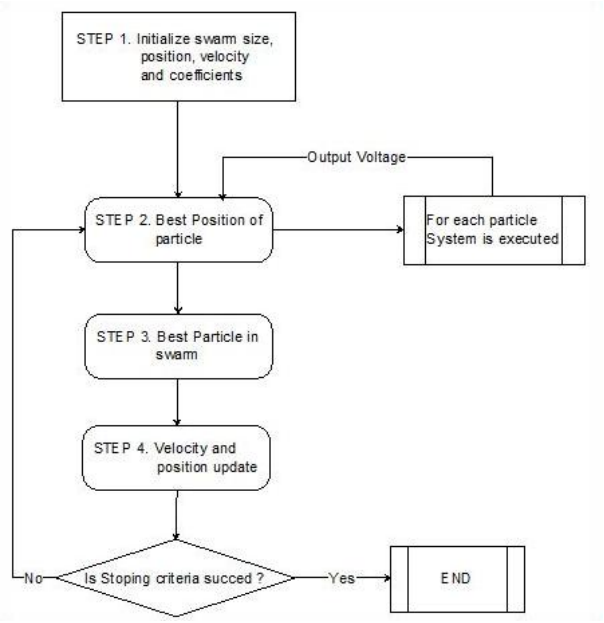


Figure 4.27 Signal Flow Graph of the PSO-PID Algorithm

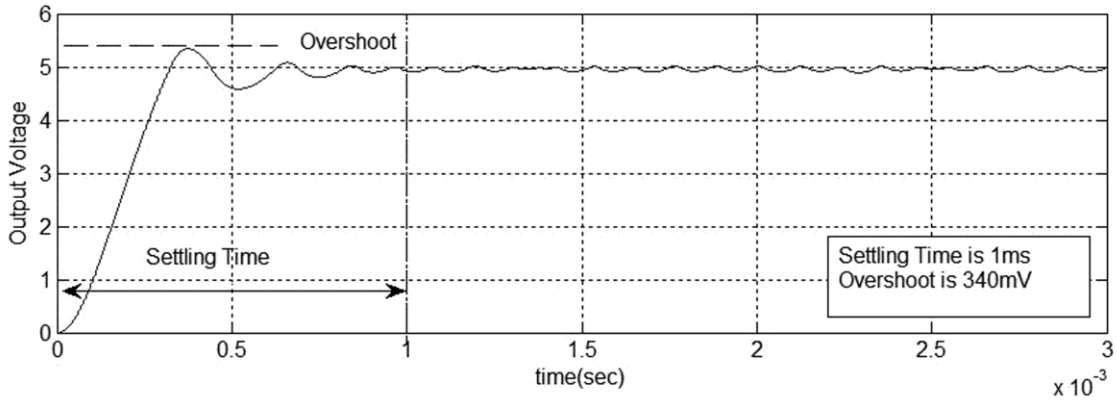


Figure 4.28 Transient Response of the Discrete PID controller

Figure 4.28 presents the transient response of the optimized PSO-PID controller under discrete time simulation environment. The settling time of the transient response is 1ms. The response shows the oscillation in the signal. Thus, settling time is determined based on the oscillation between ±0.02%.

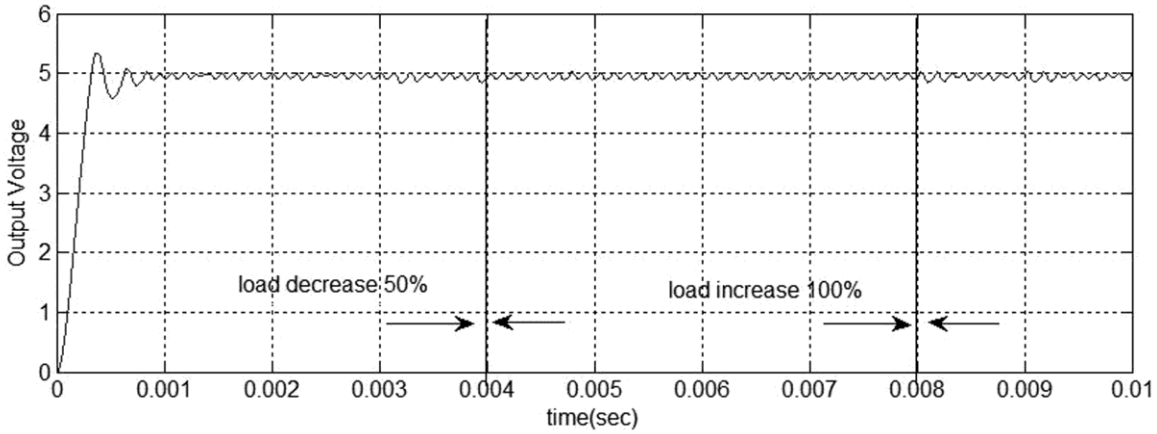


Figure 4.29 Response of Controller under Load Variation

Figure 4.29 shows the response of the controller under load variation at 50% and 100% respectively. The PID controller responds to the change in the load is not detected, which may be discussed as the fast recovery time.

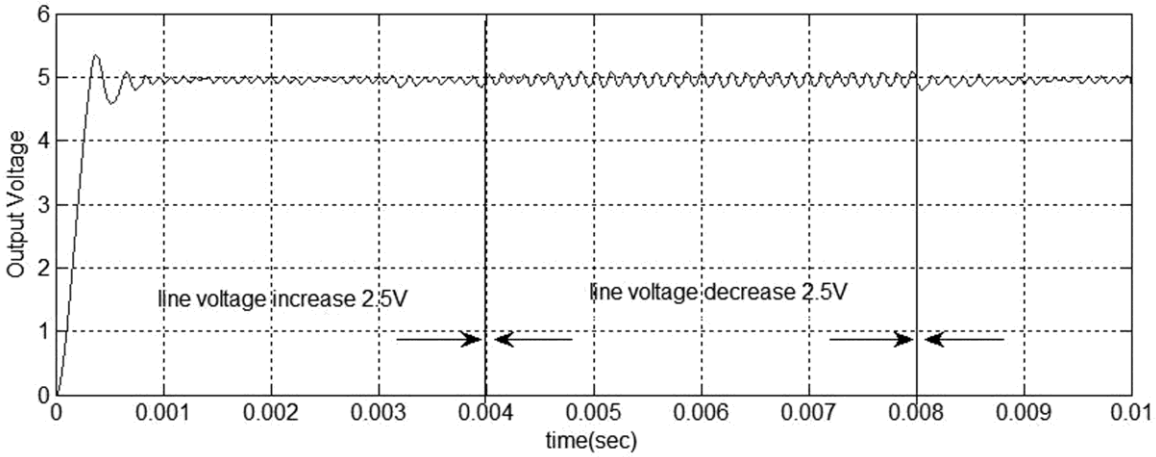


Figure 4.30 Response of Controller under 2.5V Line Variation

Figure 4.30 gives the response of the controller under 2.5V line change. The oscillation of the controller is increased. Table 4.5 presents the simulation results of the optimized PID controller.

Table 4.5 Summary for PID Simulation Results; System Performance under Various Events and Performance Criteria

Event	**Performance Criteria**	**PID Controller Performance**
Transient Response	Overshoot	0.34V
	Settling Time	1ms
Line Variation	Voltage Change	Oscillation
	Recovery Time	-
Load Variation	Voltage Change	-
	Recovery Time	-

4.4 Comments on Simulation Results

Table 4.6 presents the summary of the simulation results as table. The correct judgment about comparison of the controller performances is discussed only for their transient responses because for parameter variations, oscillations are occurred in PID and SMC.

Overshoot is one of the important signs of the controller. Because, larger overshoot is damaged the converter components. In general, for low power converters, only 20% change in the overshoots is seen as good enough. For example, for a system with the 5V reference voltage, only 1V overshoot is acceptable. From table 4.6, SMC has the best overshoot and FLC has the worst one. For settling time, same scores are observed that the SMC is best and the FLC is worst. For PID controller, under load change, the conclusion can be extracted that, the load change can be recovered very fast. Other criterions for load and line variations cannot be compared because off the oscillations occur at SMC and PID controllers (given in table 4.7).

Table 4.6 Summary for Simulation Results for all Controllers; System Performance under Various Events and Performance Criteria

Event	**Performance Criteria**		**Sliding Mode Voltage Controller Performance**	**Fuzzy Logic Controller Performance**	**PID Controller Performance**
Transient Response	Overshoot		0V	800mV	340mV
	Settling Time		325μs	1.85ms	1ms
Line Variation	Voltage Change	Increase	Oscillation increase	0.55V	Oscillation increase
		Decrease	Oscillation increase	-	Oscillation increase
	Recovery Time	Increase	-	<1ms	-
		Decrease	-	-	-
Load Variation	Voltage Change	Increase	-	150mV	-
		Decrease	-	100mV	-
	Recovery Time	Increase	-	<700 μs	-
		Decrease	-	<700 μs	-

Table 4.7 Summary for Simulation Results for all Controllers; System Performance Comparison as Best and Worst Performances

Event	**Performance Criteria**		**Best Performance**	**Worst Performance**
Transient Response	Overshoot		SMC	FLC
	Settling Time		SMC	FLC
Line Variation	Voltage Change	Increase	-	-
		Decrease	-	-
	Recovery Time	Increase	-	-
		Decrease	-	-
Load Variation	Voltage Change	Increase	-	-
		Decrease	-	-
	Recovery Time	Increase	-	-
		Decrease	-	-

5. IMPLEMENTATION FOR SYNCHRONOUS BUCK CONVERTER

In this chapter, the control methodologies that used to control the buck converter and other power converters, which are selected SMC, FLC and PID are applied to the controller. The control algorithms are implemented on the dsPICDEM SMPS development board (Appendix 2) (Figure 5.1). The microcontroller is dsPIC30F2020, which is designed to operate for power converters. The parameters of the Synchronous Buck Converter are given in table 5.1.

Table 5.1 Parameters of the Synchronous Buck Converter on the dsPICDEM Development Board

Parameter	Symbol	Value	Parameter	Symbol	Value
Input Voltage	V_{in}	9V	Inductor	I	39μH
Reference Voltage	V_{ref}	5V	Load Resistor	R	10Ω
Capacitor	C	660μF	PWM Frequency	f	100kHz

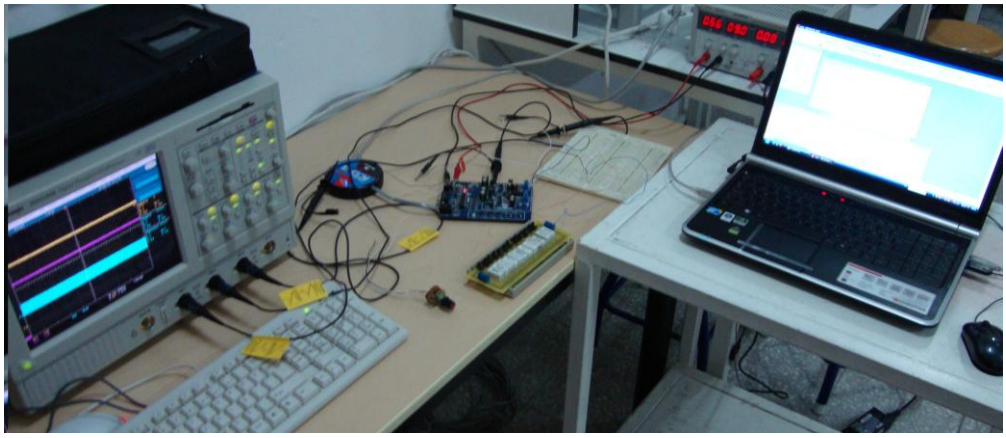


Figure 5.1 The Experiment Environment

5.1 Sliding Mode Implementation

SMC is implemented on digital microcontroller and following results are obtained. The state flow diagram of the SMC implementation is given in figure 5.2. All implementations are based on the PWM interrupt which occurs every cycle of the PWM signal.

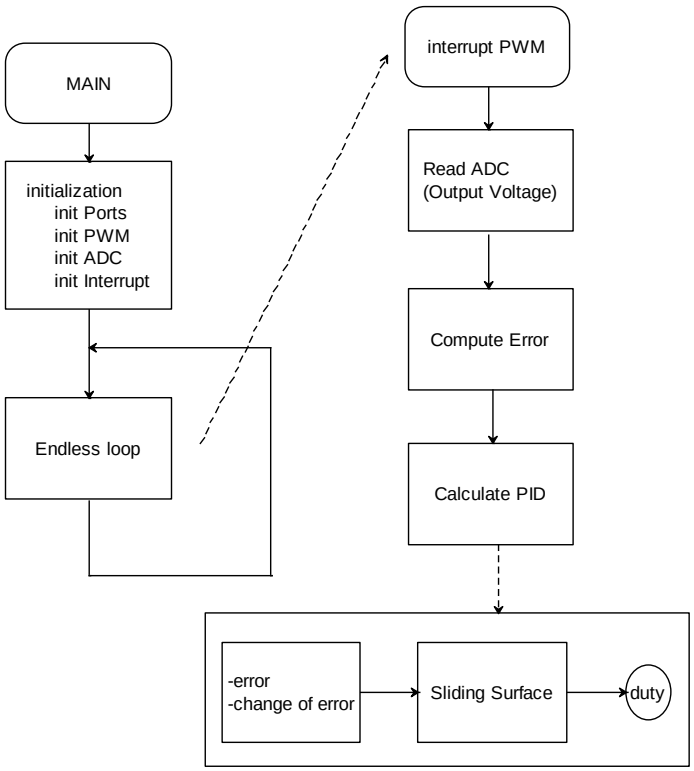
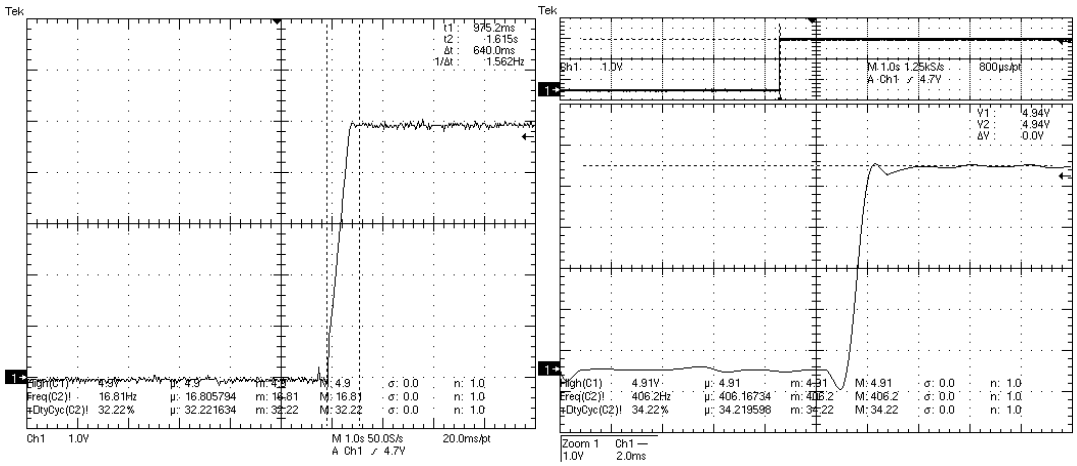
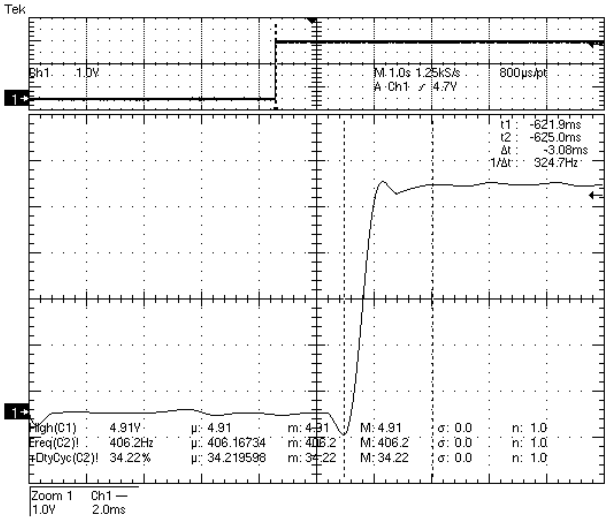


Figure 5.2 State Flow Diagram of the SMC Implementation

First, the fixed frequency SMC is applied with $d(0) = 0$ which is given in figure 5.3(a). The settling time of the transient response of the system is 640ms. The reason of this result is that the controller has to take control action to reach the steady state duty and that process takes extra time. To get rid of this problem, as initial duty, the steady state duty value is given to the controller. The results of the new transient response are given in figure 5.3(b) and 5.3(c).



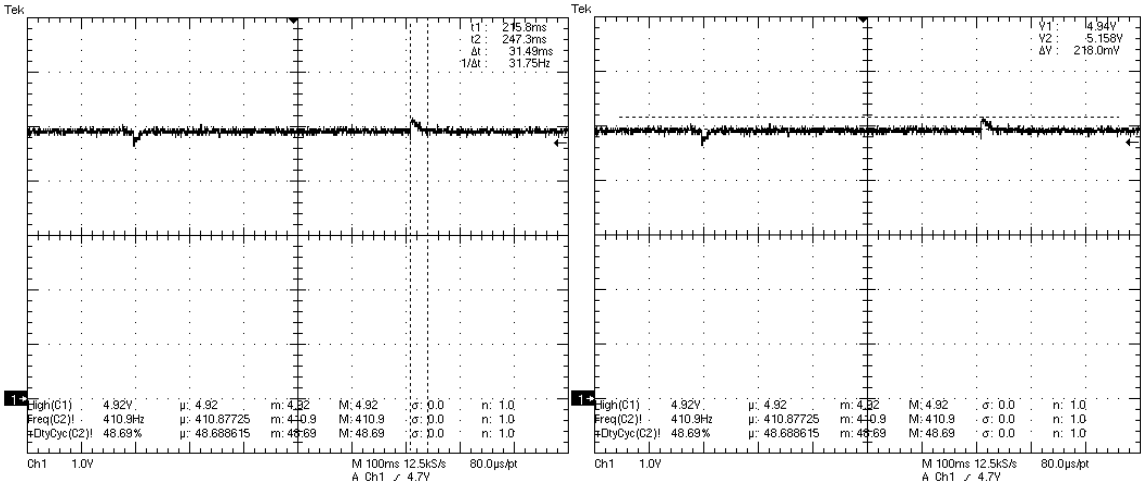
(a) (b)



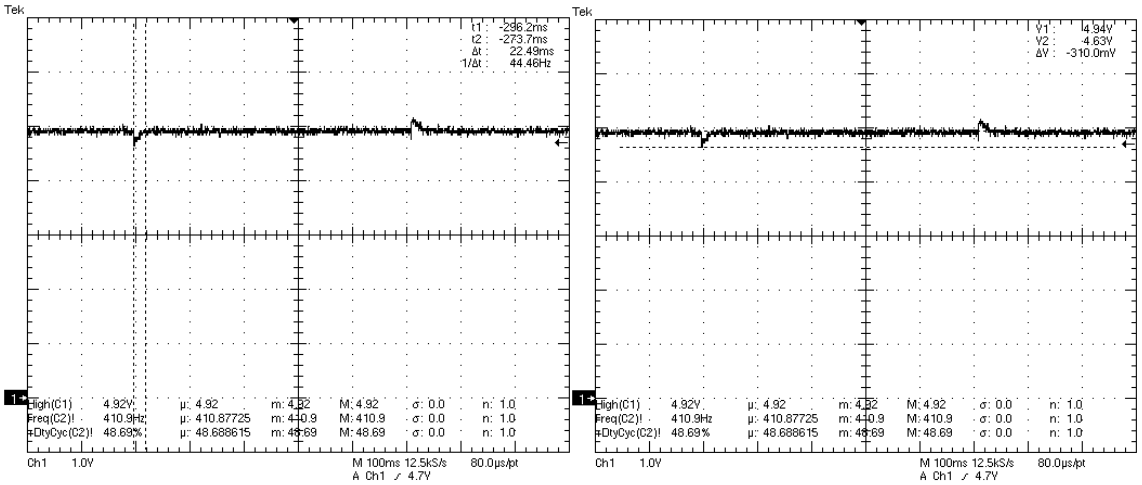
(c)

Figure 5.3 Transient Response of the SM Voltage Controller with (a) Initial Duty is Zero (b) Overshoot of the System that has the Initial Duty is equal to Steady State Duty (c) Settling Time of the System given in (b)

Figure 5.3 shows the transient response of the SM voltage controller. The settling time of the system is obtained as 3.08ms and the overshoot is zero.



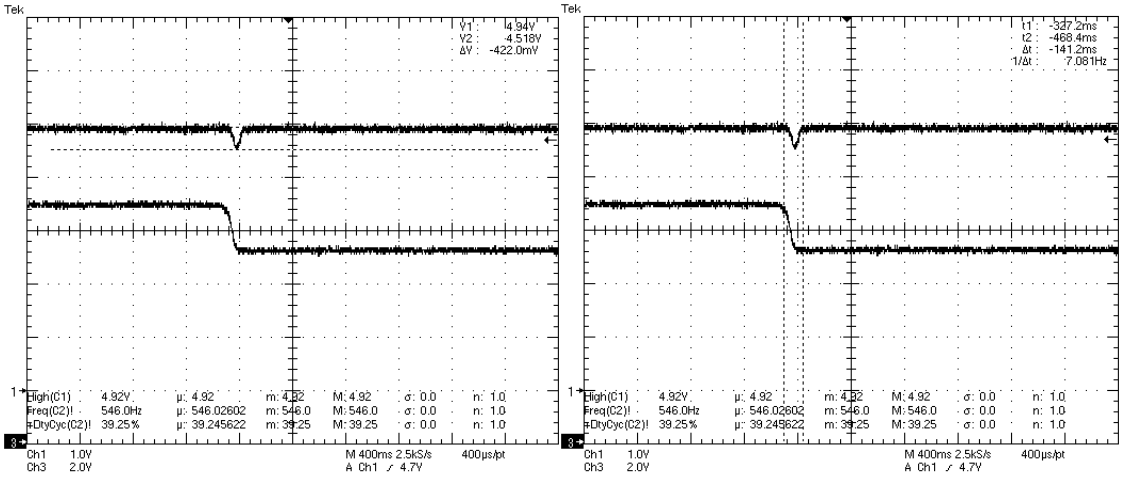
(a) (b)



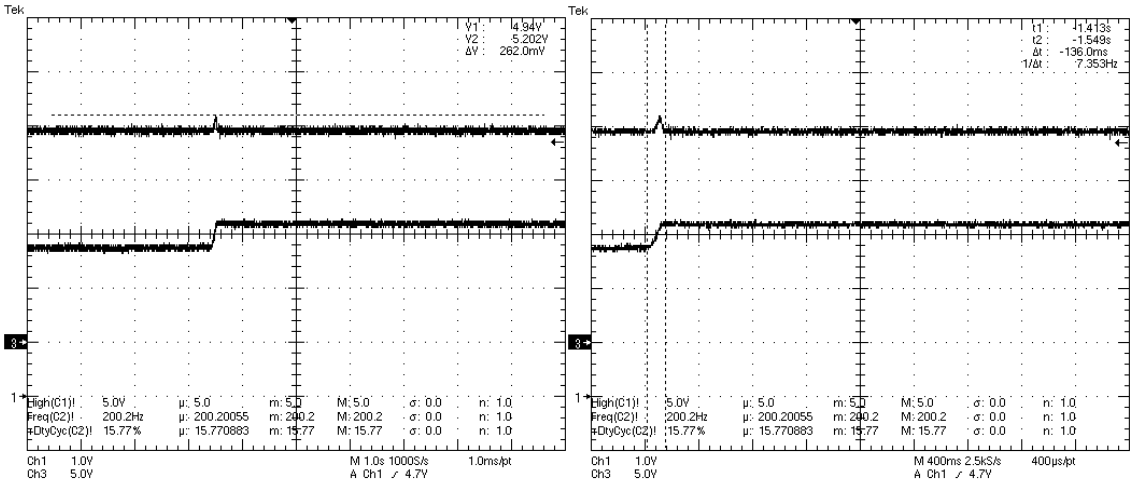
(c) (d)

Figure 5.4 Transient Response of the System with SMC under Load Variation between 10Ω and 5Ω (a) Recover Time of the System for Load Variation from 5Ω to 10Ω (b) Voltage Change of the System for Load Variation from 5Ω to 10Ω (c) Recover Time of the System for Load Variation from 10Ω to 5Ω (d) Voltage Change of the System for Load Variation from 10Ω to 5Ω

Figure 5.4 gives the transient response of the system under load variation between 10Ω and 5Ω. Recover time and voltage change of the system variation for load from 5Ω to 10Ω are 31.49ms and 218mV respectively. Recover time and voltage change of the system variation for load from 10Ω to 5Ω are 22.49ms and 310mV respectively



(a) (b)



(c) (d)

Figure 5.5 Transient Response of the System with SMC under Line Variation between 7.3V and 11.1V (a) Voltage Change of the System for Line Variation from 9V to 7.3V (b) Recover Time of the System for Line Variation from 9V to 7.3V (c) Voltage Change of the System for Line Variation from 9V to 11.1V (d) Recover Time of the System for Line Variation from 9V to 11.1V

Figure 5.5 gives the transient response of the system under line variation between 7.3V and 11.1V. Recover time and voltage change of the system variation for load from 9V to 7.3V are 141ms and 422mV respectively. Recover time and voltage change of the system variation for load from 9V to 11.1V are 136ms and 282mV respectively.

Table 5.2 Summary for SMC Experimental Results; System Performance under Various Events and Performance Criteria

Event	**Performance Criteria**		**Sliding Mode Voltage Controller Performance**
Transient Response	Overshoot		0
	Settling Time		3.08ms
Line Variation	Voltage Change	Line Increase	262mV
		Line Decrease	422mV
	Recovery Time	Line Increase	136ms
		Line Decrease	141.2ms
Load Variation	Voltage Change	Load Increase	218mV
		Load Decrease	310mV
	Recovery Time	Load Increase	31.49ms
		Load Decrease	22.49ms

5.2 Fuzzy Logic Implementation

FLC is an intelligent controller which is frequently used in the control of power converters. The FLC has the parameters of membership functions (MF) to be tuned in general. For the FLC in this thesis has 59 parameters to be determined. These parameters are tuned by the trail-error system or optimized different methods like Nefclass. But, in this thesis, the parameter that applied to the output of the FLC is investigated and optimized by using PSO.

The implementation of FLC can only be made by using digital controllers. As an academic research, only 8-bits and 32-bits microcontrollers are used to implement FLC. In this thesis, 16-bit microcontroller which is cheap in price and gives satisfactory performance is used. The state flow diagram of the FLC is given in figure 5.6.

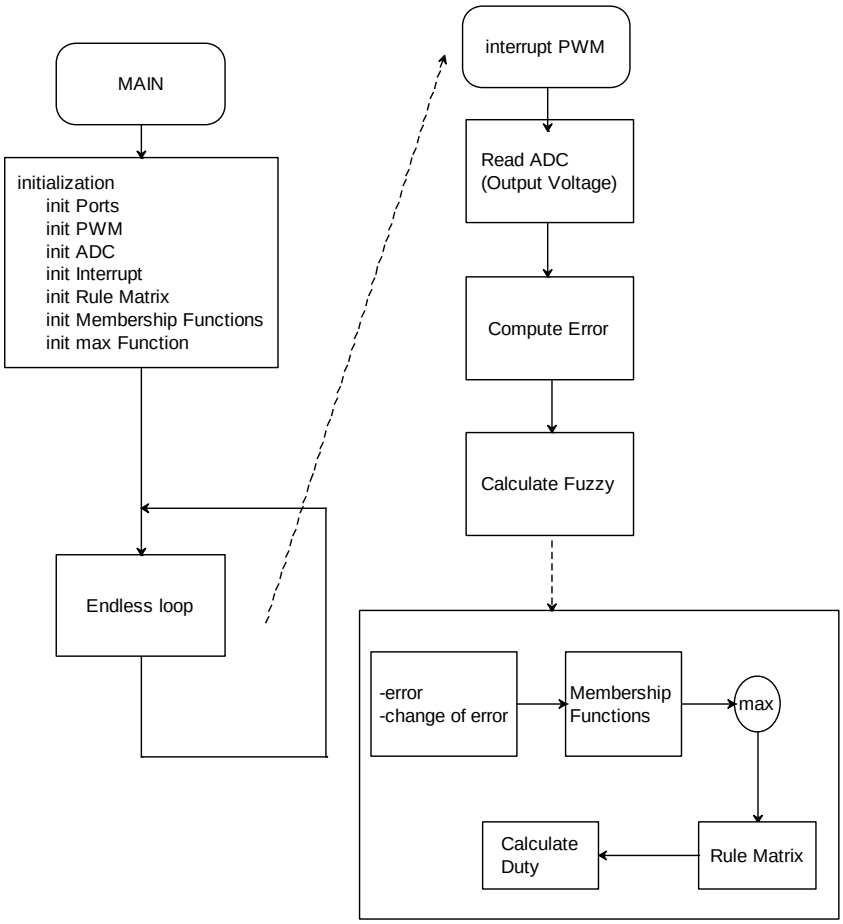


Figure 5.6 State Flow Diagram of the FLC Implementation

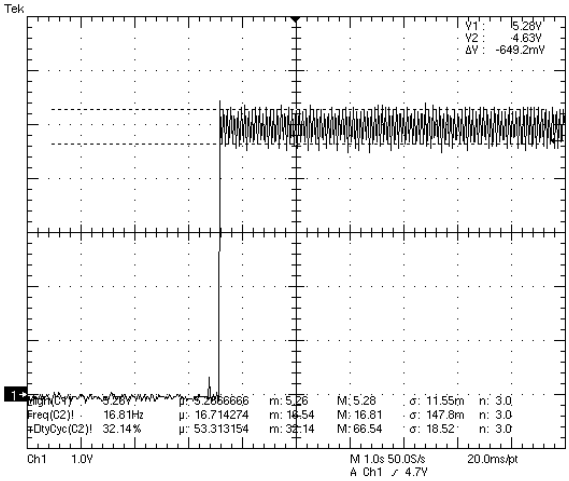


Figure 5.7 Transient Response of the System for h=1

Figure 5.7 presents the transient response of thee Fuzzy Logic Controller for different *h* parameters to show the effect of this parameter. In this thesis, this parameter is optimized by using PSO.

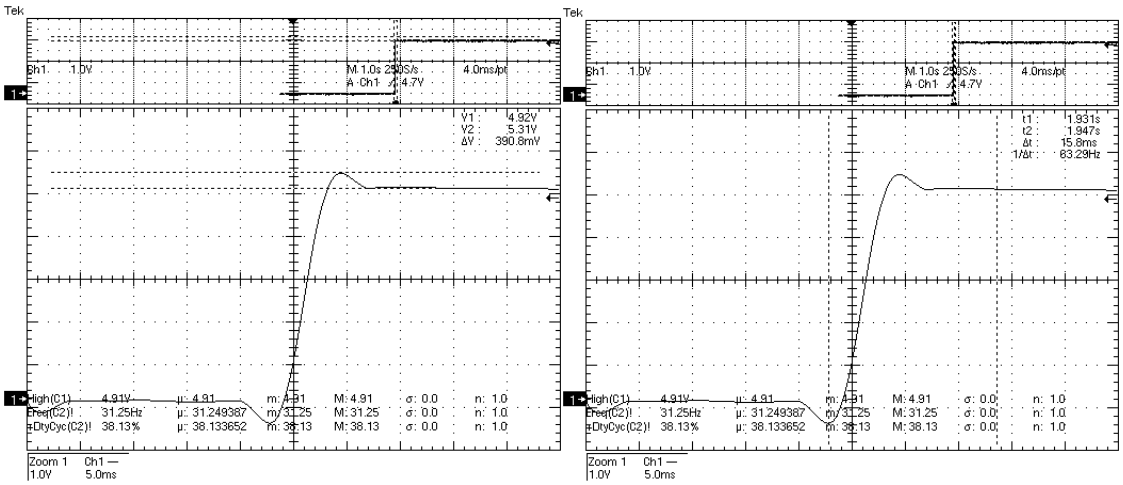
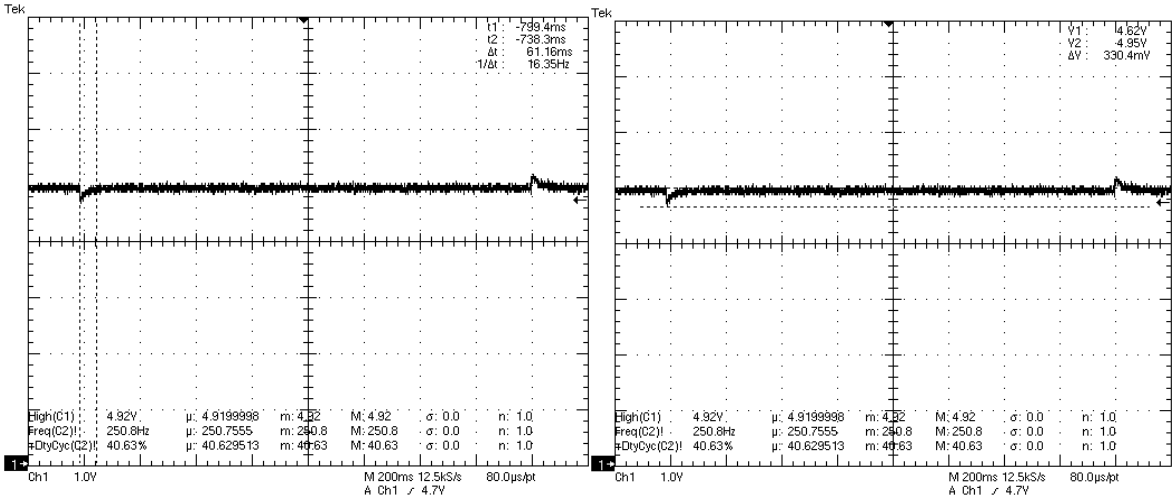


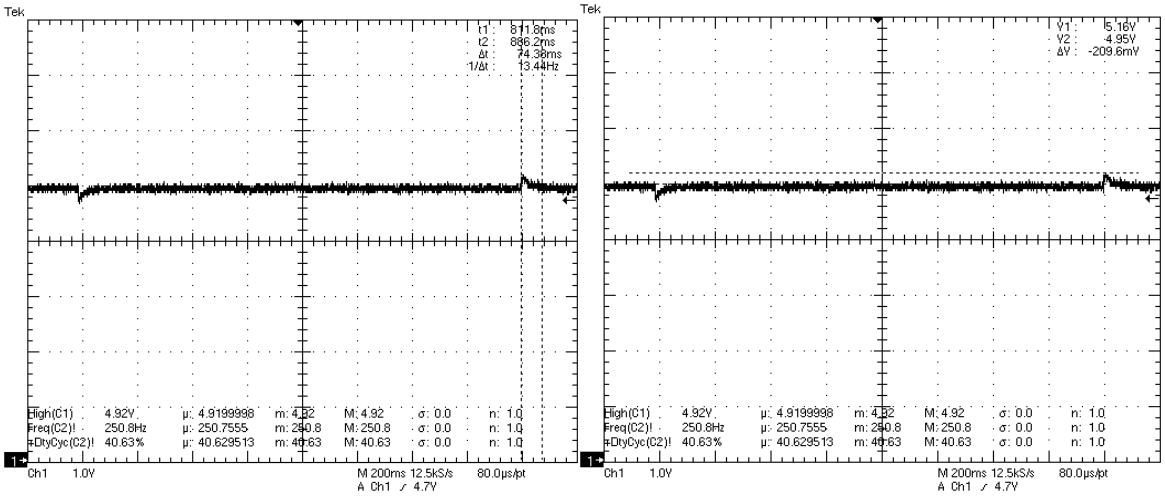
Figure 5.8 Transient Response of the Fuzzy Logic Voltage Controller with Optimized h Parameter (a) Overshoot (b) Settling Time

Figure 5.8 shows the optimized FLC and its transient response characteristic. The overshoot of the system is 390mV and settling time is 15.8ms.

Figure 5.9 gives the transient response of the system under load variation between 10Ω and 5Ω. Recover time and voltage change of the system variation for load from 5Ω to 10Ω are 74ms and 209mV respectively. Recover time and voltage change of the system variation for load from 10Ω to 5Ω are 61ms and 330mV respectively



(a) (b)



(c) (d)

Figure 5.9 Transient Response of the System with FLC under Load Variation between 10Ω and 5Ω (a) Recover Time of the System for Load Variation from 5Ω to 10Ω (b) Voltage Change of the System for Load Variation from 5Ω to 10Ω (c) Recover Time of the System for Load Variation from 10Ω to 5Ω (d) Voltage Change of the System for Load Variation from 10Ω to 5Ω

Figure 5.10 gives the transient response of the system under line variation between 7V and 11V. Recover time and voltage change of the system variation for load from 9V to 7.2V are 203ms and 730mV respectively. Recover time and voltage change of the system variation for load from 9V to 11V are 200ms and

420mV respectively. Table 5.3 summarize the performance of the FLC under load and line variation and for the start-up response of the system.

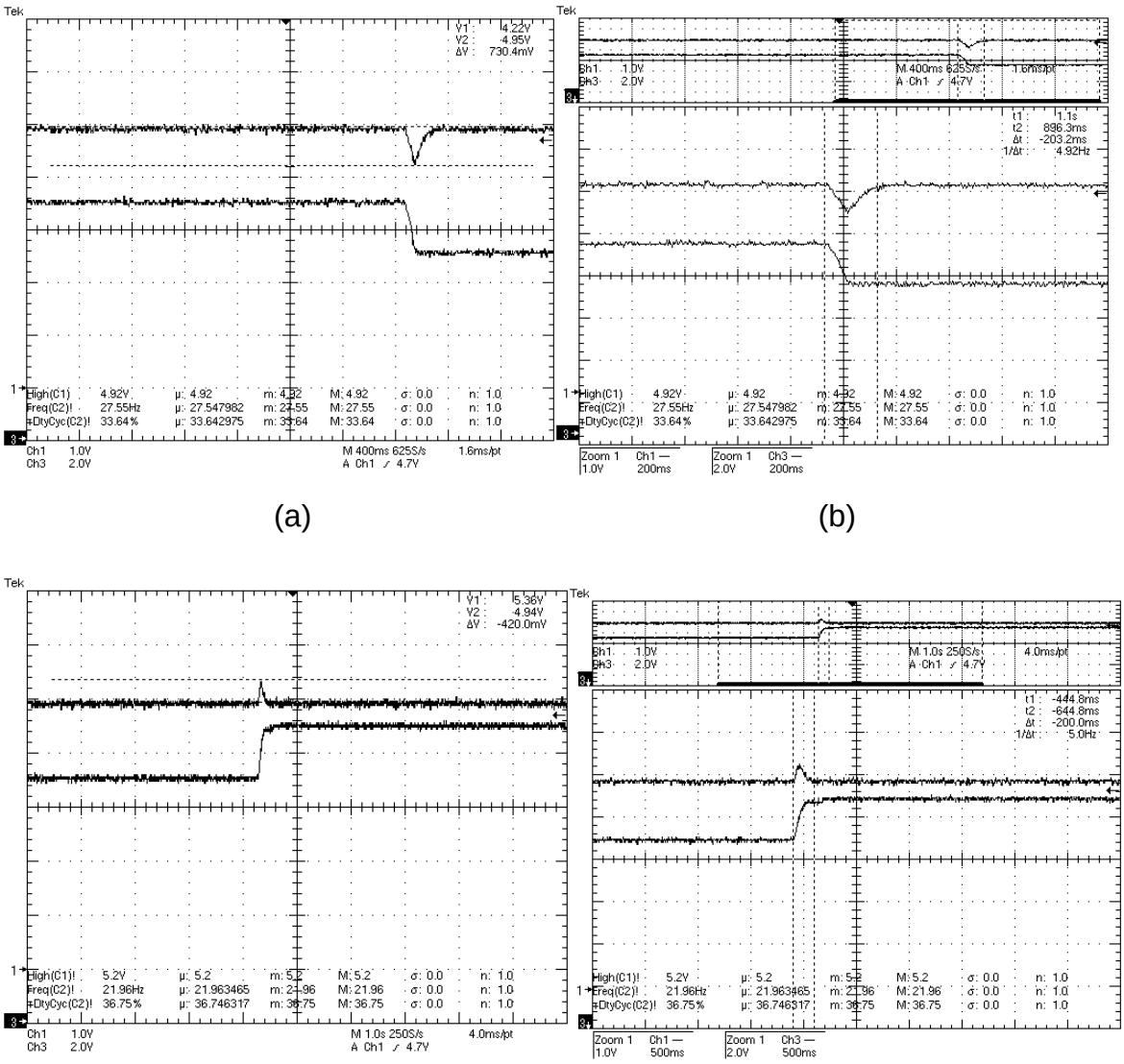


Figure 5.10 Transient Response of the System with FLC under Line Variation between 7.2V and 11V (a) Voltage Change of the System for Line Variation from 9V to 7.2V (b) Recover Time of the System for Line Variation from 9V to 7.2V (c) Voltage Change of the System for Line Variation from 9V to 11V (d) Recover Time of the System for Line Variation from 9V to 11V

Table 5.3 Summary for FLC Experimental results; System Performance under Various Events and Performance Criteria

Event	Performance Criteria		Fuzzy Logic Voltage Controller Performance
Transient Response	Overshoot		390mV
	Settling Time		15.8ms
Line Variation	Voltage Change	Line Increase	420mV
		Line Decrease	730mV
	Recovery Time	Line Increase	200ms
		Line Decrease	203.2ms
Load Variation	Voltage Change	Load Increase	209.6mV
		Load Decrease	330.4mV
	Recovery Time	Load Increase	74.38ms
		Load Decrease	61.16ms

5.3 PID Implementation

The basic controller which is using in academic and industry research is called PID controller. The basic nature of the PID controller comes from four basic mathematical expressions, which are addition, multiplication, differentiation and integration expressions. Although the controller has the basic nature, it provides generic and efficient solution [6]. Thus, that makes the controller easy to understand and applied to various real world problems. The popular usage of the PID controller drifts the researchers on finding the best PID controller. Moreover, the search is focused on the find the methodology to tune the PID to get the best PID controller for any system [39-40]. But, the application and discussion of PID controller are varying between the implementation & application and academic research.

The PID controller is a form of the phase lead-leg compensator with one pole at the origin and other at infinity [1]. Academic research is usually focused on using

the continuous PID formulation in chapter 2. But, for the application point of view, problems are observed in term tuning.

For the optimum performance, all three terms must be tuned together [6]. n general, the PI controller is used frequently because the differentiate term reduces the stability when encountered to time delay system [6]. so, the integral term saturated and leads a low-frequency oscillation and instability.

For the integral term, the automatic reset expression is defined as the mathematical expression by implemented with negative feedback. The differentiation term resulted from the infinite signal when the disturbance or change of the set point is occurred. In academic researches, generally this problem is not focused enough, but in implementations & industry this phenomena is prevented by feedback the output of the system, which makes the PID controller difficult to analyze quantitatively using standard techniques of stability and robustness analysis [6]. Thus, classical methods aren't able to use to tune the terms.

The changed PID controllers with the pre and post processes are tuned by using numerical optimization algorithms to meet the close-loop system with;

- Stability
- Transient response
- Robustness

In this thesis, to use a systematic method for tuning the PID terms, the Particle Swarm Optimization is used and step by step implementation processes are discussed. And obtained PID is implemented on a DSP processor.

In chapter 4, it was observed the PSO-PID gives better results against ZN-PSO in terms of system performance. In this chapter, the PSO-PID controller is implemented and transient response and responses against load and line variation are presented and discussed. Although, the PID controller is the most popular and used controller, the PID controller implementation is not discussed enough. Thus, the PID controller implementation is presented in detailed.

In general form, the PID controller has the following transfer function

$$G_C(s) = K_P + \frac{K_I}{s} + K_D s$$

K_P, K_I and K_D are the PID terms called proportional, integral and derivative terms respectively. By adjusting these terms, the frequency response of the system will be changed.

The proportional term is capable of sensing the error but cannot make it zero. Steady state error always exists. To overcome this problem the integral term is added for removing the steady state error. However, integral term causes the system oscillation. Derivative terms are added to be reactive to sharp changes.
The modified discrete buck converter is defined as follows.

$$u(n) = u(n-1) + K_A e(n) + K_B e(n-1) + K_C e(n-2) \tag{5.1}$$

where $K_A = (K_P + K_I T + \frac{K_D}{T})$ and $K_B = (-K_P + 2\frac{K_D}{T})$ and $K_C = \frac{K_D}{T}$ are parameters of the discrete PID controllers. The equation 5.1 means that, the controller output is equals to summation of previous instant time input, current error times K_A, previous error times K_B and error from two steps ago times K_C.

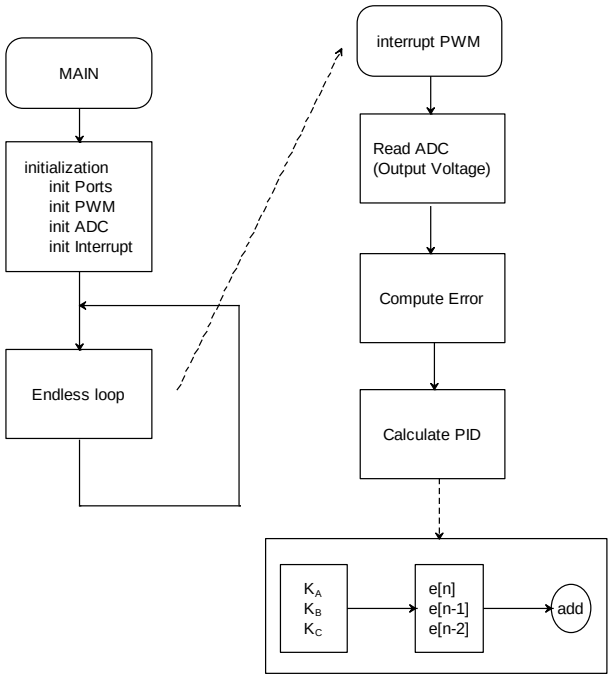
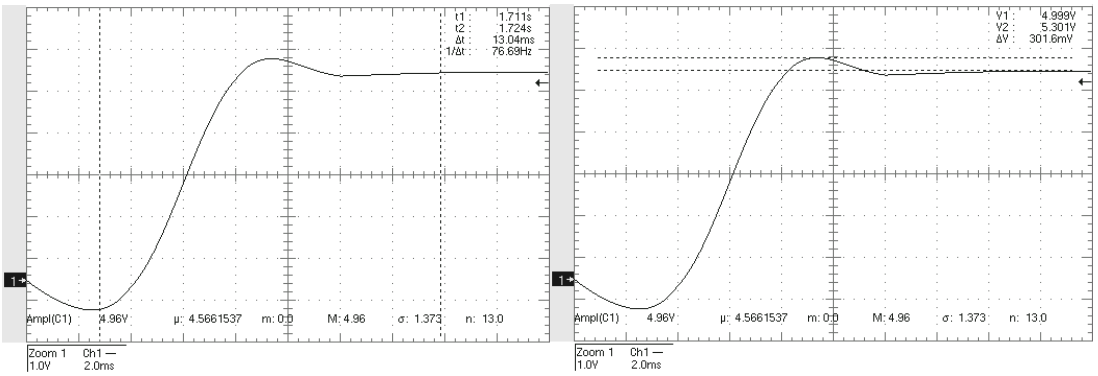


Figure 5.11 Signal Flow Diagram of the PID Implementation

The figure 5.11 shows the PID algorithm for applied any microcontroller. In this thesis, the implementation is summarized as follows:

- It initialized the ADC, PWM, PORTS and Interrupt modules.
- Program is stuck in the endless loop. However, the interrupt occurs every PWM cycle.
- When the interrupt happens, the output voltage is read from ADC module and errors are calculated.
- Based on the error values the PWM duty is calculated as given in equation 5.1.
- Return the main program and wait until the new interrupt happens.



(a) (b)

Figure 5.12 Transient Response of the PSO-PID Controller (a) Settling Time and (b) Overshoot

Figure 5.12 shows the transient response of the system. The overshoot of the system is 300mV and the settling time is 13.4ms obtained. The performance of the PID controller in terms of the load and line variations is expected to be the worst among other controllers. By using PSO, with optimized parameters the performance of the controller is investigated as the next phase of the experiment.

Figure 5.13 gives Transient response of the system under load variation between 10Ω and 5Ω. Recover time and voltage change of the system variation for load from 5Ω to 10Ω are 28ms and 220mV respectively. Recover time and voltage change of the system variation for load from 10Ω to 5Ω are 12.4ms and 720mV respectively

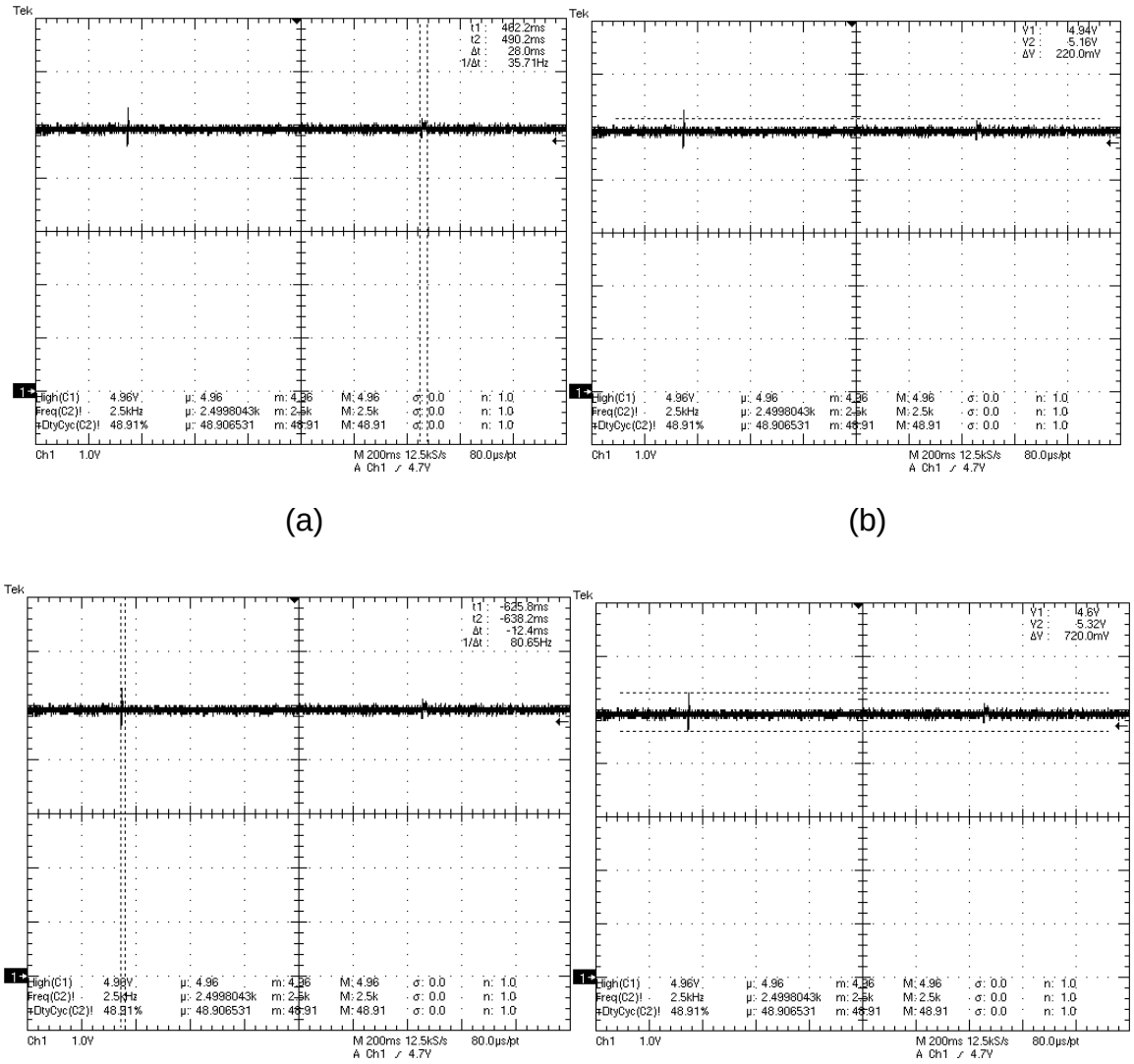
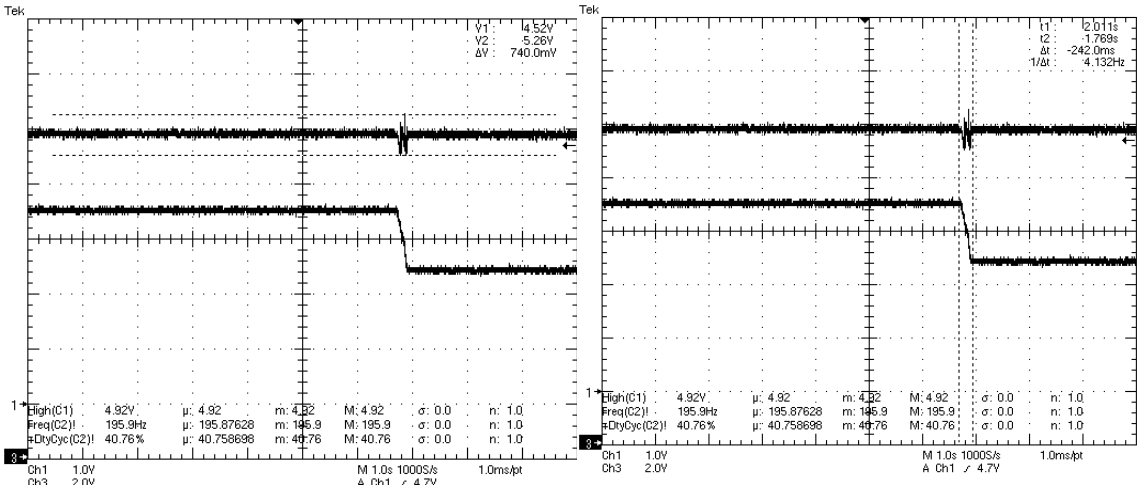
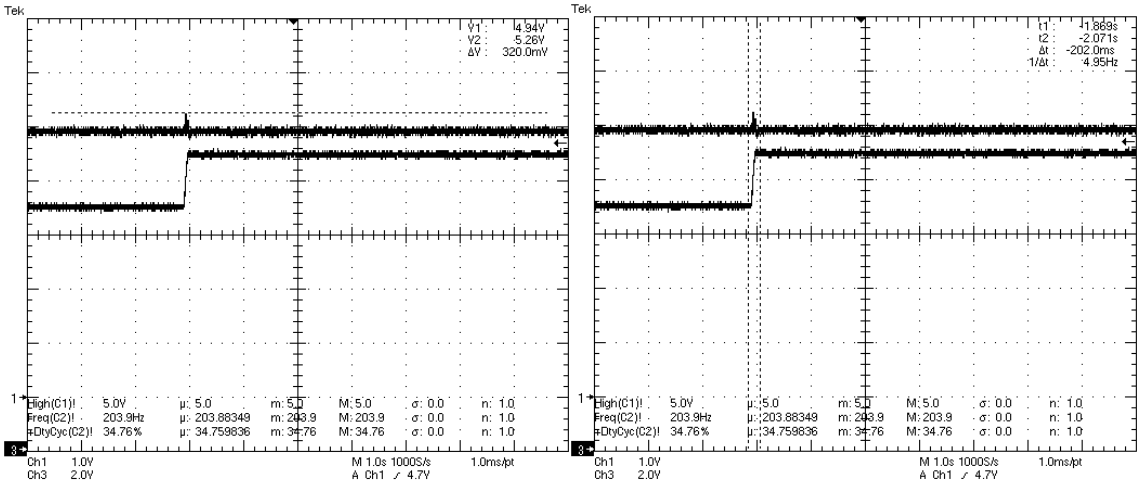


Figure 5.13 Transient Response of the System with PID under Load Variation between 10Ω and 5Ω (a) Recover Time of the System for Load Variation from 5Ω to 10Ω (b) Voltage Change of the System for Load Variation from 5Ω to 10Ω (c) Recover Time of the System for Load Variation from 10Ω to 5Ω (d) Voltage Change of the System for Load Variation from 10Ω to 5Ω



(a) (b)



(c) (d)

Figure 5.14 Transient Response of the System with PID under Line Variation between 7V and 11V (a) Voltage Change of the System for Line Variation from 9V to 7V (b) Recover Time of the System for Line Variation from 9V to 7V (c) Voltage Change of the System for Line Variation from 9V to 11V (d) Recover Time of the System for Line Variation from 9V to 11V

Figure 5.14 gives the transient response of the system under line variation between 7V and 11V. Recover time and voltage change of the system variation for load from 9V to 7V are 242ms and 740mV respectively. Recover time and voltage change of the system variation for load from 9V to 11V are 202ms and 320mV respectively. Table 5.4 present the summary of the PSO-PID control algorithm under load and line variations.

Table 5.4 Summary for PID Experimental Results; System Performance under Various Events and Performance Criteria

Event	Performance Criteria		PID Controller Performance
Transient Response	Overshoot		301.6mV
	Settling Time		13.04ms
Line Variation	Voltage Change	Line Increase	320mV
		Line Decrease	740mV
	Recovery Time	Line Increase	202ms
		Line Decrease	242ms
Load Variation	Voltage Change	Load Increase	220mV
		Load Decrease	720mV
	Recovery Time	Load Increase	28ms
		Load Decrease	12.4ms

5.4 Comments on Experimental Results

Table 5.5 presents the experiment results of the SMC, FLC and PID controllers based on performance of the system under start-up, load and line variation. Also, table 5.6 gives the comparison the controllers based on obtained data which is given in table 5.5.

From experimental results, the SMC is given the best results compared the other controllers based on line variation and transient responses. FLC is given the worst performance under transient response. The most interesting result is taken from PID controller. As expected, PID controller gave the worst results in general. But, optimized PID gives the best recovery time for load change.

Table 5.5 Summary for Experimental Results for all Controllers; System Performance under Various Events and Performance Criteria

Event	**Performance Criteria**		**Sliding Mode Voltage Controller Performance**	**Fuzzy Logic Controller Performance**	**PID Controller Performance**
Transient Response	Overshoot		0	390mV	301.6mV
	Settling Time		3.08ms	15.8ms	13.04ms
Line Variation	Voltage Change	Line Increase	262mV	420mV	320mV
		Line Decrease	422mV	730mV	740mV
	Recovery Time	Line Increase	136ms	200ms	202ms
		Line Decrease	141.2ms	203.2ms	242ms
Load Variation	Voltage Change	Load Increase	218mV	209.6mV	220mV
		Load Decrease	310mV	330.4mV	720mV
	Recovery Time	Load Increase	31.49ms	74.38ms	28ms
		Load Decrease	22.49ms	61.16ms	12.4ms

Table 5.6 Summary for Experimental Results for all Controllers under Worst and Best Performance

Event	Performance Criteria		Best Performance	Worst Performance
Transient Response	Overshoot		SMC	FLC
	Settling Time		SMC	FLC
Line Variation	Voltage Change	Line Increase	SMC	FLC
		Line Decrease	SMC	PID
	Recovery Time	Line Increase	SMC	PID
		Line Decrease	SMC	PID
Load Variation	Voltage Change	Load Increase	FLC	PID
		Load Decrease	SMC	PID
	Recovery Time	Load Increase	PID	FLC
		Load Decrease	PID	FLC

6. CONCLUSION

This thesis has focused on modeling and control of the synchronous buck converter. Modeling of the converter was achieved by using state space averaging method. From the point of the control theory, optimized PID, fuzzy and sliding mode control methods were applied and compared their performance with conventional methods.

In order to control of the proposed controller, voltage mode was selected and the most used control methods: fuzzy, sliding mode and PID was applied. In order to improve controller performance PSO was used to optimize various parameters of controllers. The optimized parameters were as follows.

- PID
- Fuzzy
- Sliding Mode

Simulation and experimental results show that

- Design of PID controller based on PSO is straight forward than frequency and/or root locus methods and results of control action has faster settling time and smaller overshoot
- Optimized fuzzy logic controller is more robust than classical fuzzy under load and line variation
- Sliding mode controller with optimized parameters is presented faster settling time and lower overshoot

From simulations, it was observed that, the SMC is the best performance of the transient response of the system and FLC is the worst performance for the simulation results. The load and line variation comparison cannot be done because of the insufficient mathematical model of the system.

From experimental results, as general performance SMC is suggested to be used in control of the synchronous buck converter. The FLC gave the mild performance, but it can be used with PID to increase the transient response performance.

In hardware application the optimized controller was applied by using dsp digital controller. The performance of the controllers was compared respect to memory usage of program and data is given in table 6.1.

The mathematical model was obtained from analysis of the synchronous buck converter. In spite of the general power electronic concept, the converter was divided into states not to circuit elements. From the relation between these states, the mathematical model of the converter was obtained by using a general application of the state space averaging method.

The voltage mode controller techniques were took the attention of this thesis because of their common usage that they also used in current mode control. From all researches, the controllers were reduced on three different controllers because of the different usage of these controllers. 1) Most of the papers in literature are focused on the compensation and different PID combination. 2) The nature of the power converters is variable structure, so sliding mode controllers are defined for power converters. 3) Fuzzy logic controllers are applied to power converters for their straight forwardness. The PID, sliding mode and fuzzy logic controllers were selected as bench controllers for synchronous buck converter.

In the simulation phase, the optimization algorithm called Particle Swarm Optimization was used for obtain the optimal coefficients of all three controllers. First simulation was applied to sliding mode controller. SMC is divided into variable frequency and fixed frequency approaches. Simulations showed that both fixed and variable sliding mode controllers are acted like PID controllers and didn't give the sufficient response. The fuzzy logic controller was applied to synchronous buck converter. The FLC is defined with seven membership functions for two inputs and one output. The multiplier is defined output of the FLC and the effect of this multiplier is investigated and optimized. Simulations show that FLC is acted like PID. Finally, the PID controller was applied to synchronous buck converter by tuning Ziegler Nichols method and proposed PSO method.

The implementation of controllers was also compared based on program and data memory usage. Table 6.1 presents the storage comparison of the controller algorithms.

Table 6.1 Comparison of the controllers' implementation via program and data memory usage

	Program Memory (x3 Bytes)	Percentage	Data Memory (Bytes)	Percentage
Sliding Mode Controller	961/4096	23%	132/512	%25
Fuzzy Logic Controller	1472/4096	35%	396/512	%77
PID Controller	973/4096	23%	60/512	%11

Table 6.2 Summary for experiment and simulation results for all controllers; system performance under various events and performance criteria

Experimental Results					
Event	**Performance Criteria**		**Sliding Mode Voltage Controller Performance**	**Fuzzy Logic Controller Performance**	**PID Controller Performance**
Transient Response	Overshoot		0	390mV	301.6mV
	Settling Time		3.08ms	15.8ms	13.04ms
Line Variation	Voltage Change	Increase	262mV	420mV	320mV
		Decrease	422mV	730mV	740mV
	Recovery Time	Increase	136ms	200ms	202ms
		Decrease	141.2ms	203.2ms	242ms
Load Variation	Voltage Change	Increase	218mV	209.6mV	220mV
		Decrease	310mV	330.4mV	720mV
	Recovery Time	Increase	31.49ms	74.38ms	28ms
		Decrease	22.49ms	61.16ms	12.4ms
Simulation Results					
Event	**Performance Criteria**		**Sliding Mode Voltage Controller Performance**	**Fuzzy Logic Controller Performance**	**PID Controller Performance**
Transient Response	Overshoot		0V	800mV	340mV
	Settling Time		325µs	1.85ms	1ms

Line Variation	Voltage Change	Increase	Oscillation increase	0.55V	Oscillation increase
		Decrease	Oscillation increase	-	Oscillation increase
	Recovery Time	Increase	-	<1ms	-
		Decrease	-	-	-
Load Variation	Voltage Change	Increase	-	150mV	-
		Decrease	-	100mV	-
	Recovery Time	Increase	-	<700 µs	-
		Decrease	-	<700 µs	-

APPENDIX 1 - PARTICLE SWARM OPTIMIZATION

The behavior of the organisms is one of the main concerns of optimization. The unknowns for all control structures are found by using particle swarm optimization (PSO). PSO is one of the newest optimization structures. In this chapter, general PSO property and types of PSO is given.

A.1 Introduction to Particle Swarm Optimization

The particle swarm optimization (PSO) was introduced by Eberhart and Kennedy in 1995 [41]. This method is based on simplified social behavior of animal swarms like birds and fishes. Generally, bird flocks, like fish and other animals, fly through the sky to search for nesting and food. The way of attitude that group of animals used in order to find their supplies is the main concern of the methodology.

The aim of the PSO is to find the minimum (or maximum) of the function. For connecting the behavior of the swarm to the optimization problem first each bird is defined as weightless particles (and usually called agents). These particles are put into the problem space with random initial positions and velocities. We assume that each bird does not know where the food is but by their instincts they know how far it is. The main idea is to change their velocities and positions based on their instincts.

PSO is investigated into two different topics. The first one assumes that particles have the knowledge about the best value of the all particles. This is called global best or simply ‘gbest’. The second one composed by the concept of that particles have the knowledge about the best value of the nearest neighbor particle.

A.1.1 “GBEST” Particle Swarm Optimization

As indicated in [41], the particles are placed into problem space with random initial positions (dimension of the space also called number of the unknowns) and velocities. Each particle observes the distance from the field (or simply evaluates the desired minimized function also called fitness function). Furthermore, each bird knows their best position and if the new position is the best (closes) that it changed it best value with the new one. Thus each bird has its own best value,

then the GBEST structure use, which is from all best values, the global best value and the position of this value is found from all particles (is also called group norm). Finally based on these data, the position and velocities are changed by using also different types of structures.

A.1.2 “LBEST” Particle Swarm Optimization

The idea of LBEST or simply local best was introduced in 1995 [42]. This idea is taken place after the best positions of all particles are observed. From all best values, the only best value and the position of each particle’s neighbor’s is found, which is called local, best or lbest. Finally based on these data, the position and velocities are changed by using also different types of structures. By this reason, it is clear to see that the neighbors are created groups and explore different regions of the problem space. Thus it gives a much flexible approach that gbest model.

The main difference between these two different PSO is the number of iteration and the resistant to the local minimum. From [42], it is clear that the gbest PSO shows lower iterations but has less resistant to the local minimum but lbest is opposite of gbest which is resistant to the local minimum but higher iteration.

A.2 Particle Swarm Optimization Algorithms

In table 1, the general pseudo code of the PSO is given. From previous parts, clearly to find the minimal value around the swarm depends of two methodologies namely as global and local best. In this chapter, PSO algorithms are defined for all cases; to remember that the only difference is the find the best pattern to corresponding particle [43].

Table A1.1 Pseudo code of Particle Swarm Optimization Algorithm

```
Initialize swarm velocity and position randomly
Do
    For i = 1 to swarm size
        Calculate fitness function
        If fit < best pattern
            i^th particle best position ← i^th particle position
            best pattern ← fit
        end if
    end for
    find min best pattern and corresponding particle*
    For i = 1 to swarm size
        Update velocity with equation 2 or 3 or 5
        Update position
    end for
While termination conditions are not met
```

A.2.1 Simplified (Conventional) Particle Swarm Optimization

In this part the first form of the PSO s given as in [41]. As a review from previous chapters, we know that each particle (i is the number of the particle) is put into problem space (which is assumed D dimensional) with random initial position as $xi = (xi1, xi2, ..., xiD)$ and velocity as $vi = (vi1, vi2, ..., viD)$. Each particle evaluates the fitness function (which is their instinct that each particle knows the distance to the field). However, we assume that each particle had the memory that holds the best value so far. This memory also calls autobiographical memory. After evaluation of the fitness function, the best values of each particle so far are obtained as $pi = (pi1, pi2, ..., piD)$. After that point, we know that two different method can be use to obtain the swarm best values. They are called global best and local best values. After that process took into action the index of the best particle among all particles (or in the neighborhood of that particle) is called g . The update process of the conventional PSO is given in equations (A1.1) and (A1.2).

$$v_{id} = v_{id} + c_1 * rand() * (p_{id} - x_{id}) + c_2 * rand() * (p_{gd} - x_{id}) \quad \text{(A1.1)}$$

$$x_{id} = x_{id} + v_{id} \quad \text{(A1.2)}$$

In equations (A1.1) and (A1.2), *rand ()* is the random process between [0 1], c_1 and c_2 are two positive constants. Equation (1) is divided into three parts based on linear adding operation. The first part is the old velocity, that part is added to

following parts to get the new velocity. The second part is called “cognition” part or “simple nostalgia”. This part is the private thinking of each particle. This part is that the bird tends to return to the place that the most satisfied in past [1]. The third part is the “social” part. In that part, particle tends to move to the closest point to the field.

A.2.2 Modified Particle Swarm Optimization

Modified PSO was introduced in 1998 [44]. The idea is based on the behavior of the parts of equation A1.1. Assume that only the old velocity is the new one which means there is no part two and three $v_{id} = v_{id}$. That means that the particle is flying with the same velocity in the same position as shown in figure A1.1(a). It can be seen that the bird can fly through to large search space. This large search space is giving the opportunity to the bird to search globally.

(a) (b)

Figure A1.1 (a) The Constant Velocity for Bird Particle, (b) For not Old Velocity is used

Now, let investigate for the reverse position that when there is no old velocity $v_{id} = c_1 * rand() * (p_{id} - x_{id}) + c_2 * rand() * (p_{gd} - x_{id})$. In that situation, when any particle is in the global best position than that particle is not moved until one of the other particles will be the global best particle. Until that happen, other particles are converged to global one with the narrow searching distance as shown in figure A1.1(b). The narrow searching distance gives the bird to search local area rapidly.

Clearly for each different problem, there is different necessity or local and global search area. Then as mentioned before we investigate the effect of the old velocity, we can multiply the old velocity with a constant to make a balance

between global and local search. So the equation A1.1 is changed into equation A1.3.

$$v_{id} = (w)v_{id} + c_1 * rand() * \left(p_{id} - x_{id}\right) + c_2 * rand() * \left(p_{gd} - x_{id}\right) \quad \text{(A1.3)}$$

From empirical studies of the modified particle swarm optimization is gave the results that if the coefficient (also called inertia weight) (w) change with the iteration, which means linearly decreasing inertia is used, PSO converges quickly. Equation A1.4 gives the linear inertia weight equation.

$$w = w\max - \frac{w\max - w\min}{iter\ \max} iter \quad \text{(A1.4)}$$

From [45] is can be seen that the optimum values of $(w\max)$ and $(w\min)$ is 0.9 and 0.4 respectively. The values of these variables are determined based on empirical studies. It is clear to see that the value of the inertia weigh is the direct effect on to reach to the mean best fitness value.

A.2.3 Constriction Factor Approach (CFA_PSO)

As mentioned previous chapter, the local and global search is dependent on the old velocity. By multiplying the old velocity with the constant (or linearly decrease function or an intelligent and nonlinear structure), namely as inertia weight, the balance between local and global search is constructed. By this way, we try to ensure that the swarm is not stuck in local search area, but there is no process to out from the local search problem (search space is too small).

In [46], the two conditions are defined as NoHope and ReHope. In NoHope, the swarm is in the local search area but the solution is not that field. Then, the Rehope takes place that based on the re-locate the swarm based on the gravity of the swarm namely as Queen. By this way, the swarm is gone out from local search area and the search continuous. By using this idea a general research is applied in [47]. As a conclusion of this work, how a particle search for a complex space is analyzed. First the trajectory of the swarm is investigated based on the simplified

equation of the velocity and position also given in [46]. From this equation, the roots of the equation are found based on the coefficients. From the roots, the trajectories are investigated based on, real or complex root ideas. An as a result it is found that c=c1+c2>4 (choose as c1=c2=2.05) to ensure no cyclic behavior (searching the same area without finding the optimum value) and the swarm velocity and error (difference between optimum point and particle position) distance to the origin is strictly monotonic increasing with time. This means that swarm search will be large.

After that, 5-D perspective is analyzed. This perspective is basically defining five unknowns for the new equations. Then investigate the behavior of the particle. By changing these constants to different values is resulted from the controlling the particles' behavior. And finally the suggested equations are given in equation A1.5 and A1.6.

$$vid = k\left[vid + \phi_1\left(pid - xid\right) + \phi_2\left(pgd - xid\right)\right] \tag{A1.5}$$

$$k = \frac{2}{\left|2 - c - \sqrt{c^2 - 4c}\right|} \tag{A1.6}$$

Also note that, $c_i * rand() = \phi i, i = 1,2$

A.2.4 Compare of the Inertia Weight and Constriction Factor

From previous parts, it is common to ask to which structure should be chosen for PSO. The both Inertia weight (IW) and Constriction Factor (CF) are based on the analysis of the search area of the particle. To answer of this question is given in [48]. As seen from results CF is the special case of IW where the coefficients are adjusted best manner. From [48] in IW the chosen of random number in [0 1.49445] and k as 0.729 is same as CF.

A.3. Comparison of PSO with other Optimization Methods

For optimization, there are two programming methods are existed. First one is called mathematical programming methods. These techniques are based on the searching the solution space and approach the solution by defining a direction to

the optimal such as steepest descent method. The major problem is that the solution is generated usually only at a local optimum [49]. Second method is called heuristic method. This technique is based on the movement that to reach the optimum by making a couple of trial-error or practical try. Modern heuristic methods are based on biological system behaviors. These are simulated annealing (based on thermodynamic), evolutionary algorithms (based on genetic), and swarm algorithms (based on swarm behavior such as Particle Swarm Optimization)

Simulated annealing is the procedure that simulating the process based on slowly cooling a system in order to obtain states with globally minimum energy. By this way, near globally-minimum-cost solutions can be found for large optimization problems [50]. The randomness at the system under thermal energy lets the atoms escape from local optimum and form globally optimal setting [49]. The advantage of the simulated annealing is the ability to guarantee optimal solution and disadvantage is repeatedly annealing is very small, for smooth energy landscape problems simpler and faster local methods will work better [49].

Modern faster methods are Genetic Algorithm and Particle Swarm Optimization. Genetic Algorithm (GA) is similar to the Particle Swarm Optimization (PSO) in the sense of population-based search. The drawback of the GA is the computational cost. In [51], GA and PSO are compared by using static analysis and formal hypothesis tests. As a result, that paper showed that PSO and GA obtain high quality solutions. However, PSO obtained the solution with less effort than GA.

APPENDIX 2 – SYNCHRONOUS BUCK CONVERTER

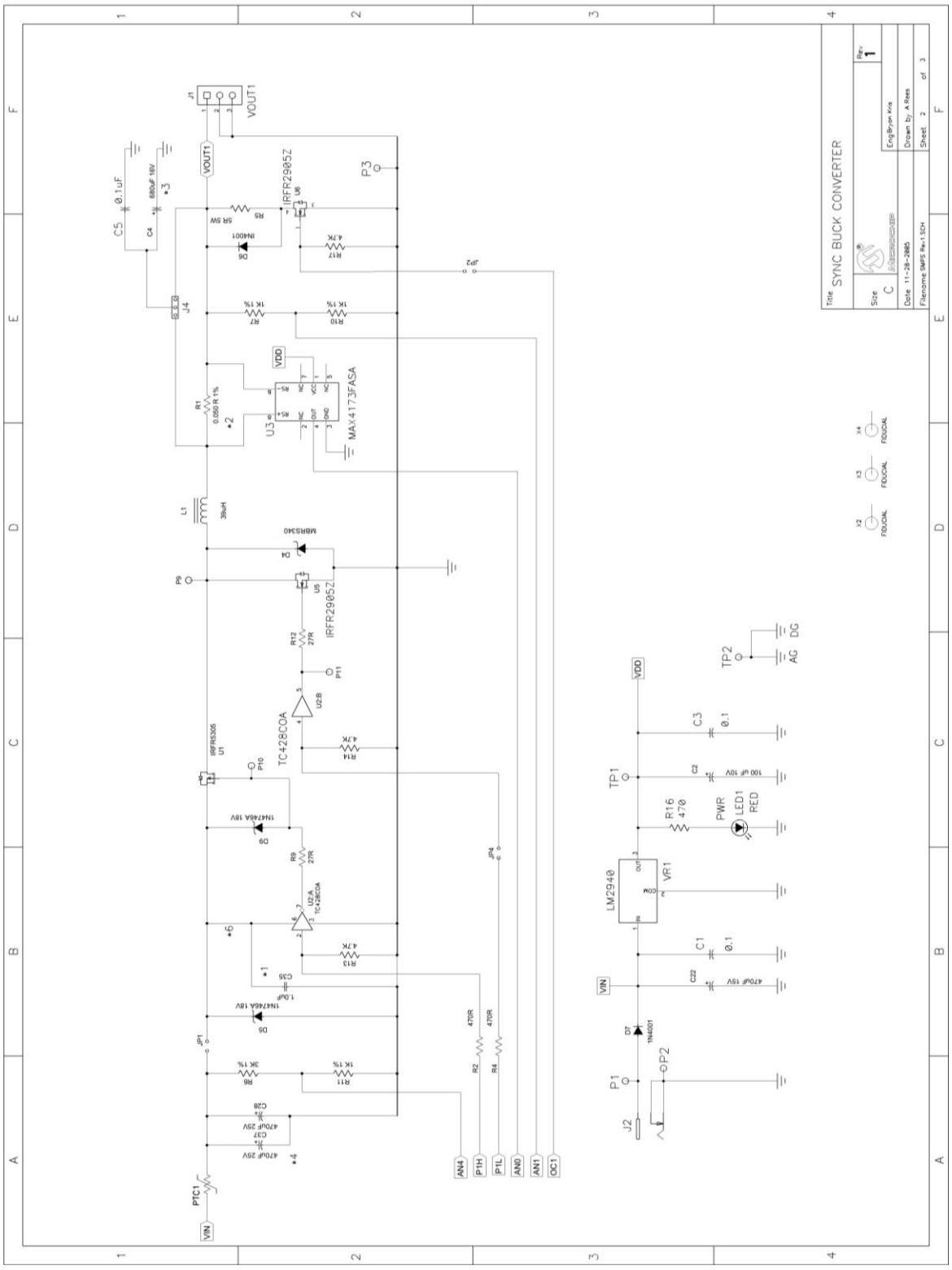


Figure A2.1 Schematic of the Synchronous Buck Converter

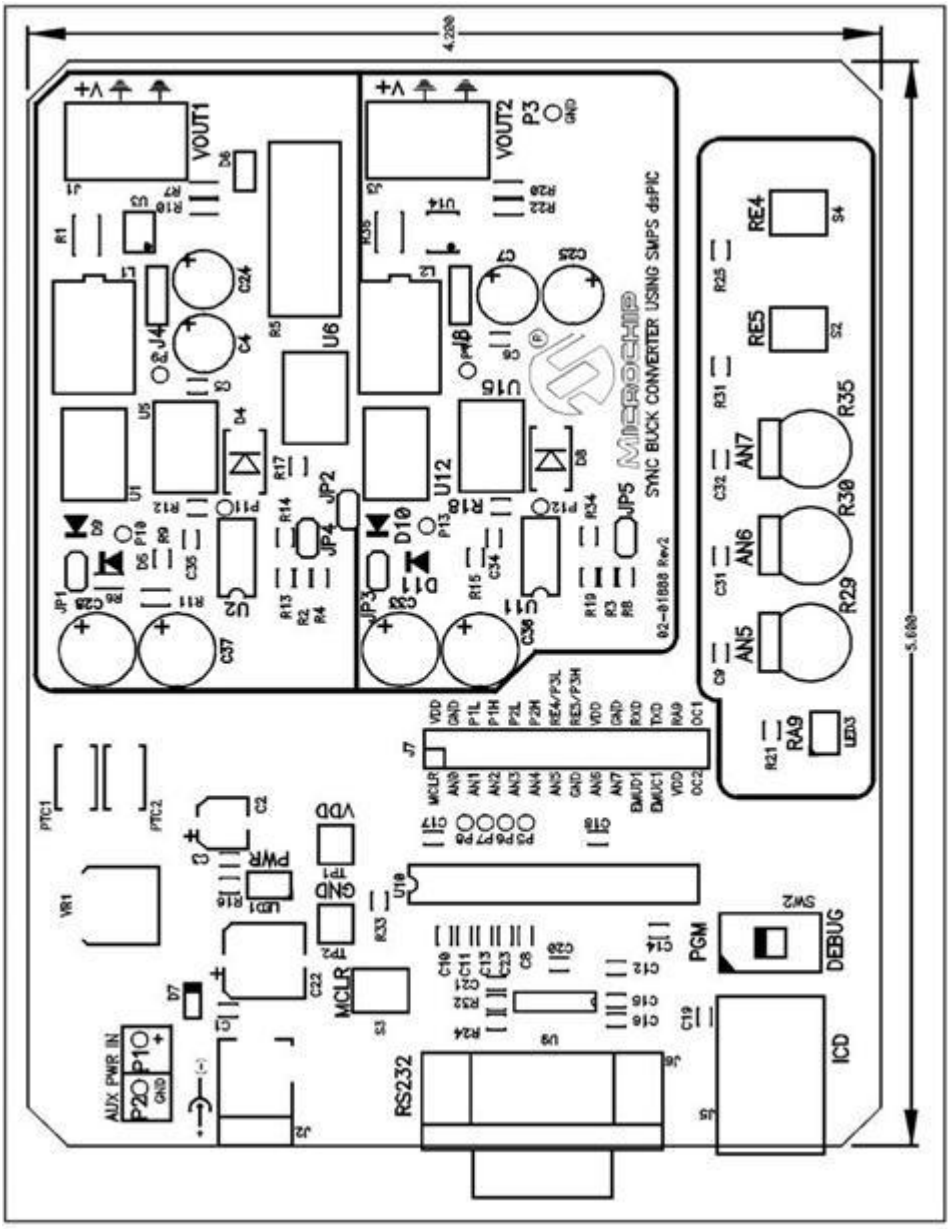


Figure A2.2 The layout of the dsPICDEM SMPS

REFERANCES

[1] N. Mohan, *First Course on Power Electronics and Drivers*, Minneapolis: MNPERE, 2003.

[2] C. H. Rivetta, A. Emadi, G. A. Williamson *et al.*, “Analysis and Control of a Buck DC-DC Converter Operating withh Constant Power Load in Sea and Undersea Vehicles,” *IEEE Transactions on Industry Applications,* vol. 42, no. 2, pp. 559-572, 2006.

[3] A. Perry, “New Methods for the Analysis and Design of Fuzzy Logic Controllers for DC-DC Converters,” Department of Electrical and Computer Engineering, Queen's University, Ontario, 2002.

[4] L. Guo, J. Y. Hung, and R. M. Nelms, “Evaluation of DSP-Based PID and Fuzzy Controllers for DC-DC Converters,” *IEEE Transactions on Industrial Electronics,* vol. 56, no. 6, pp. 2237-2248, 2009.

[5] R. C. Dorf, and R. H. Bishop, *Modern Control Systems*, 10 ed.: Prentice Hall, 2005.

[6] Y. Li, K. H. Ang, and G. C. Y. Chong, “PID Control Systems Analysis and Design,” *IEEE Control Systems Magazine*, pp. 32-41, 2006.

[7] V. Utkin, J. Guldner, and J. Shi, *Sliding Mode Control in Electromechanical Systems*: Taylor & Francis, 1999.

[8] M. Ahmed, M. Kuisma, K. Tolsa *et al.*, “Implementing Sliding Mode Control for Buck Converter,” in Power Electronics Specialist Conference, 2003. PESC '03. 2003 IEEE 34th Annual, 2003, pp. 634- 637.

[9] S.-C. Tan, Y. M. Lai, M. K. H. Cheung *et al.*, “On the Practicle Design of a Sliding Mode Voltage Controlled Buck Converter,” *IEEE Transactions on Power Electronics,* vol. 21, no. 2, pp. 425-437, 2005.

[10] R. Nowakowski, and N. Tang, “Efficiency of Synchronous versus Nonsynchrronous Buck Converter,” *Texas Instrument Incorporated Analog Application Journal Power Management*, pp. 15-18, 2009.

[11] S. M. Cuk, “Modelling, Analysis and Design of Switching Converters,” CALTEC, California, 1977.

[12] R. D. Middlebrook, and S. M. Cuk, “A General Unified Approach to Modeling Switching-Converter Power Stages,” *Proceedings of the IEEE Power Electronics* pp. 73-86, 1976.

[13] F. C. Y. Lee, R. P. Iwens, Y.Yu *et al.*, “Generalized Computer-Aided Discrete Time-Domain Modeling and Analysis of DC-DC Converters,” *IEEE Transactions on Industrial Electronics and Control Instrumentation,* vol. 26, no. 2, 1979.

[14] F. C. Y. Lee, and Y. Yu, “Computer-Aided Analysis and Simulation of Switched DC-DC Converters,” *IEEE Transactions on Industry Applications,* vol. 15, no. 5, pp. 511-520 1979.

[15] D. J. Shortt, and F. C. Lee, “Extension of thee Discrete-Average Model for Converter Power Stages,” *IEEE Transactions on Aerospace and Electronic Systems,* vol. 20, no. 3, 1984.

[16] A. J. Forsyth, and S. V. Mollov, “Modeling and Control of DC-DC Converters,” *Power Engineering Journal*, pp. 229-236, 1998.

[17] Z. Zhao, and A. Prodic, “Limit Cycle Oscillation Based Auto-Tuning System for Digitally Controlled DC-DC Power Supplies,” *IEEE Transactions on Power Electronics,* vol. 22, no. 6, 2007.

[18] J. Morroni, R. Zane, and D. Maksimovic, “Design and Implementation of an Adaptive Tuning System Based on Desired Phase Margin for Digitally Controlled DC-DC Converters,” *IEEE Transactions on Power Electronics,* vol. 24, no. 2, 2009.

[19] L. Corradini, E. Orietti, P.Mattavelli *et al.*, “Digital Hysteretic Voltage-Mode Control for DC-DC Converters Based on Asynchronous Sampling,” *IEEE Transactions on Power Electronics,* vol. 24, no. 1, 2009.

[20] T. Gupta, R. R. Boudreax, R. M. Nelms *et al.*, “Implementation of a Fuzzy Controller for DC-DC Converters using an Inexpensive 8-b Microcontroller,” *IEEE Transactions on Industrial Electronics,* vol. 44, no. 5, 1997.

[21] M. M. Islam, D. R. Allee, S.Konasani *et al.*, “A Low-Cost Digital Controller for a Switching DC Converter with Improved Voltage Regulation,” *IEEE Power Electronics Letters,* vol. 2, no. 4, 2004.

[22] Y. Y. Mai, and P. T. Mok, “A Constant Frequency Output-Ripple-Voltage-Based Buck Converter without using Large ESR Capacitor,” *IEEE Transactions on Circuit And Systems – II: Express Briefs,* vol. 55, no. 8, 2008.

[23] S.-C. Tan, Y. M. Lai, C. K. Tse *et al.*, “A Fixed-Frequency Pulsewidth Modulation Based Quasi-Sliding-Mode Controller for Buck Converters,” *IEEE Transactions on Power Electronics,* vol. 20, no. 6, pp. 1379-1392, 2005.

[24] H. Fujioka, C. Y. Kao, S. Almer *et al.*, “Robust Tracking with H∞ Performance for PWM Systems,” *Elsevier Automatica*, 2009.

[25] M. R. D. Al-Mothafar, “Average Current-mode Control of a two-Module DC-DC converter with mutually coupled Output Filter Inductors,” *International Journal of Electronics,* vol. 83, no. 3, 1997.

[26] G. Feng, E. Meyer, and Y. F. Liu, “A Digital Two-Switching-Cycle Compensation Algorithm for Input-Voltage Transients in DC-DC Converters,” *IEEE Transactions on Power Electronics,* vol. 24, no. 1, 2009.

[27] C. Gezgin, B. S. Heck, and R. M. Bass, “Simultaneous Design of Power Stage and Controller for Switching Power Supplies,” *IEEE Transactions on Power Electronics,* vol. 12, no. 3, 1997.

[28] H. Lee, and V. I. Utkin, “Chattering Suppression Methods in Sliding Mode Control Systems,” *Annual Reviews in Control,* vol. 31, pp. 179-188, 2007.

[29] S.-C. Tan, Y. M. Lai, C. K. Tse *et al.*, “Adaptive Feedforward and Feedback Control Schemes for Sliding Mode Controlled Power Converters,” *IEEE Transactions on Power Electronics,* vol. 21, no. 1, pp. 182-192, 2006.

[30] P. Cominos, and N. Munro, “PID Controllers: Recent Tuning Methods and Design to Specification,” *IEE Proc.-Control Theory Appl.,* vol. 149, no. 1, 2002.

[31] Z. L. Gaing, “A Particle Swarm Optimization Approach for Optimum Design of PID Controller in AVR System,” *IEEE Trans. Energy Conversion,* vol. 19, no. 2, pp. 384-390, 2004.

[32] S. P. Ghoshal, “Optimizations of PID gains by Particle Swarm Optimization in Fuzzy based Automatic Generation Control,” *Electric Power Systems Research*, pp. 203-212, 2004.

[33] C. C. Kao, C. W. Chuang, and R. F. Fung, “The Self-tuning PID control in a Slider-Crank Mechanism System by Applying Particle Swarm Optimization Approach,” *Mechatronics*, pp. 513-522, 2006.
[34] V. Mukherjee, and S. P. Ghoshal, “Intelligent Particle Swarm Optimized Fuzzy PID Controller for AVR System,” *Electric Power Systems Research*, pp. 1689-1698, 2007.
[35] U. S. Banu, and G. Uma, “Fuzzy Gain Scheduled Continuous Stirred Tank Reactor with Particle Swarm Optimization Based ID Control Minimizing Integral Square Error,” *Instrumentation Science and Technology*, pp. 394-409, 2008.
[36] S. M. G. Kumar, R. Sivasankar, T. K. Radhakrishnan *et al.*, “Particle Swarm Optimization Technique Based Design of PI Controller for a Real-Time Nonlinear Process,” *Instrumentation Science and Technology*, pp. 525-542, 2008.
[37] T. H. Kim, I. Maruta, and T. Sugie, “Robust PID Controller Tuning Based on the Constrained Particle Swarm Optimization,” *Automatica*, pp. 1104-1110, 2007.
[38] M. Zamani, N. Sadati, and M. K. Ghartemani, “Design of and H∞ PID Controller using Particle Swarm Optimization,” *International Journal of control, Automation, and Systems*, pp. 273-280, 2009.
[39] L. Wang, T. J. D. Barnes, and W. R. Cluett, “New Frequency-domain Design Method for PID Controllers,” *IEE Proc. Control Theory Appl.,* vol. 142, no. 4, 1995.
[40] J. G. Ziegler, and N. B. Nichols, “Optimum settings for Automatic Controllers,” *ASME,* vol. 64, no. 8, pp. 759-768, 1942.
[41] J. Kennedy, and R. Eberhart, “Particle Swarm Optimization,” *IEEE*, pp. 1942-1948, 1995.
[42] R. Eberhart, and J. Kennedy, "New Optimizer Using Particle Swarm Theory." pp. 39-43.
[43] J. Kennedy, and C. Eberhart, “A Discrete Binary Version of the Particle Swarm Algorithm,” *IEEE*, pp. 4104-4108, 1997.
[44] Y. Shi, and R. Eberhart, “A Modified Particle Swarm Optimi,” *IEEE*, pp. 69-73, 1998.
[45] Y. Shi, and R. Eberhart, “Empirical Study of Particle Swarm Optimization,” *IEEE*, pp. 1945-1950, 1999.
[46] M. Clerc, “The Swarm and The Queen: Towards a Deterministic and Adaptive PSO,” *IEEE*, pp. 1951-1957, 1999.
[47] M. Clerc, and J. Kennedy, “The Particle Swarm – Explosion, Stability, and Convergence in a Multidimensional Complex Space,” *IEEE Transactions on Evolutionary Computation,* vol. 6, no. 1, 2002.
[48] R. Eberhart, and Y. Shi, “Comparing Inertia Weights and Constriction Factors in Particle swarm Optimization,” *IEEE*, pp. 84-88, 2000.
[49] Y.-H. Song, and M. R. Irving, “Optimization Techniques for Electrical Power Sysstems Part 2 Heuristic Optimization Methods,” *Power Engineering Journal*, pp. 151-160, 2001.
[50] B. Hajek, “A Tutorial Survey of Theory and Applications of Simulated Annealing,” in 24th Conference on Decision and Control, 1985, pp. 755-760.

[51] R. Hassan, B. Cohanim, and O. d. Weck, “A Copmarison of Particle swarm Optimization andd Genetic Algorithm,” *American Institute of Aeronautics and Astronautics*, pp. 1-13, 2004.