# MODELLING AND CONTROL OF DC TO DC CONVERTER (BUCK)

MUHAMAD FARHAN BIN UMAR BAKI

This thesis is submitted as partial fulfillment of the requirement for the award of the Bachelor Degree of Electrical Engineering (Power System)

Faculty of Electrical & Electronic Engineering
University Malaysia Pahang
(UMP)

MAY, 2008

## ABSTRACT

This project is done to form a model of switch mode dc-dc converter incorporated with the PID controller. The dc-dc converter (buck) will step down the input dc voltage of 12V to 5V. The PID controller will control the outputs which are voltage and current and those will be display in graph. The PID controller is tuned to get the highest quality of output. The switching frequency of the dc-dc converter (buck) is set to 100khz for faster switching operation. This system is implemented in MATLAB simulink software. This project is also providing some analysis and comparative assessment.

## CHAPTER 1

## INTRODUCTION

### 1.1 OVERVIEW

Main target in power electronics is to convert electrical energy from one form to another. To make electrical energy to reach the load with highest efficiency is the target to be achieved. Power electronics also targets to reduce the size of the device to convert these energy which aims to reduce cost, smaller in size and high availability. In this project the power electronic device that use is dc to dc converter. There are four types of dc to dc converter:

i. Buck

ii. Boost

iii. Buck-boost

iv. Cuk

The dc-dc converter for this project is buck converter. Buck is use to convert unregulated dc input to a controlled dc output with a desired voltage level. The buck will step down the input voltage 12 Vdc to 5 Vdc with the switching frequency 100 kHz and 400 kHz.Together with buck is PID controller that uses to control the behaviors of the system in linear .This system is a close loop system with feedback. The software is use to do simulation is MATLAB SIMULINK. This project consists of modeling, simulation and stability analysis

## 1.2 OBJECTIVE

**i.** To design and form the mathematical model of the dc-dc converter (buck).

**ii.** To implement the dc-dc converter (buck) incorporated with PID controller in MATLAB SIMULINK environment.

**iii.** To analyze the result and form the system stability analysis

## 1.3 SCOPE

i. Design and modelling dc-dc converter (buck) using PWM to generate the pulse

ii. Design of the PID controller and the system will operates in close loop or in other word has feedback to stabilize the system and the system is linear. Implement this system in MATLAB SIMULINK environment.

iii. Form stability analysis of the system.

## 1.4 PROBLEM STATEMENT

The output voltage (Vo) of buck alone usually is unstable. So criteria must concern is rise time, overshoot , settling time and steady state error , to get the desired output and to reduce the undesired output.

Problem statement:

- steady state error

  - The output of buck alone is not reaching the desire value meaning it has error.
- rise time

  - The rise time is too long
- settling time

  - The output oscillating too long, it takes time to reach the stable state.
- Overshoot

  -The over shoot is high.

## 1.5 THESIS ORGANIZATION

This thesis consists of five chapters. This chapter discuss about overview of project, objective research, project scope, problem statement and thesis organization.

Chapter 2 contains a detailed description of continuous conduction mode theory to design the buck converter and PID controller. It will explain about the concept of buck converter and PID controller including tuning method.

Chapter 3 includes the design of the system of buck controller incorporated with PID controller. It will explain how the project is organized and the flow of process in completing this project.

Chapter 4 will discuss on result and analysis. The first analysis is comparison between the simulation result of buck without PID controller and buck with PID controller, and the second analysis is comparison between 100 kHz model and 400 kHz model.

Finally, the conclusions for this project are presented in chapter 5. This chapter also discusses about the recommendation for the project and for the future development.

# CHAPTER 2

# LITERATURE REVIEW

## 2.1 BACKGROUND

This chapter will explain about dc-dc converter (buck), pulse width modulation, driver, and proportional derivative integral (PID) that will use as controller.

## 2.2 BASIC OF DC TO DC CONVERTER (BUCK)

Instead of using transformer we also can use switching converter to step down the input voltage, the reason why should use converter because typically the output produced is at a different voltage level than the input. In addition, DC-to-DC converters are used to provide noise isolation, power bus regulation, etc. In this project basically we must know how the converter operates and the operation of the system. A DC-to-DC converter is a device that accepts a DC input voltage and produces a DC output voltage. [1]

## 2.3 PWM AND TRANSISTORS [2]

PWM is the main part in designing a buck converter. By using pulse-width modulation (PWM) control, regulation of output voltage is achieved by varying the duty cycle of the switch. Duty cycle refers to ratio of the period where power semiconductor is kept ON to the cycle period. Pulse width modulation (PWM) is a powerful technique for controlling analog circuits with a processor's digital outputs. PWM is employed in a wide variety of applications, ranging from measurement and communications to power control and conversion .Control of PWM is usually effected by an IC is necessary for regulating the output. The transistor switch is the most important thing of the switched supply and controls the power supplied to the load. It is also stated that Power MOSFET’s are more suitable than BJT at power output of the order of 50 W. Choosing of transistor also must consider its fast switching times and able to withstand the voltage spikes produced by the inductor [2]

### 2.3.1 Digital control

By controlling analog circuits digitally, system costs and power consumption can be drastically reduced. What's more, many microcontrollers and DSPs already include on-chip PWM controllers, making implementation easy.In a nutshell, PWM is a way of digitally encoding analog signal levels. Through the use of high-resolution counters, the duty cycle of a square wave is modulated to encode a specific analog signal level. The PWM signal is still digital because, at any given instant of time, the full DC supply is either fully on or fully off. The voltage or current source is supplied to the analog load by means of a repeating series of on and off pulses. The on-time is the time during which the DC supply is applied to the load, and the off-time is the periods during which that supply is switched off. Given a sufficient bandwidth, any analog value can be encoded with PWM. [2]

## 2.4 PROPORTIONAL INTEGRAL DERIVATIVES [3]

A proportional-integral-derivative controller (PID controller) is a generic control loop feedback mechanism widely used in industrial control systems. A PID controller attempts to correct the error between a measured process variable and a desired set point by calculating and then outputting a corrective action that can adjust the process accordingly.

The PID controller calculation (algorithm) involves three separate parameters; the Proportional, the Integral and Derivative values. The Proportional value determines the reaction to the current error, the Integral determines the reaction based on the sum of recent errors and the Derivative determines the reaction to the rate at which the error has been changing. The weighted sum of these three actions is used to adjust the process via a control element such as the position of a control valve or the power supply of a heating element.By "tuning" the three constants in the PID controller algorithm the PID can provide control action designed for specific process requirements. The response of the controller can be described in terms of the responsiveness of the controller to an error, the degree to which the controller overshoots the setpoint and the degree of system oscillation. Note that the use of the PID algorithm for control does not guarantee optimal control of the system. [4]

### 2.4.1 Proportional term

The proportional term makes a change to the output that is proportional to the current error value. The proportional response can be adjusted by multiplying the error by a constant $K_p$, called the proportional gain.

The proportional term is given by:

$$P_{OUT} = K_P e(t)$$

Where

- $P_{out}$: Proportional output
- $K_p$: Proportional Gain, a tuning parameter
- e: Error = SP − PV
- t: Time or instantaneous time (the present)

A high proportional gain results in a large change in the output for a given change in the error. If the proportional gain is too high, the system can become unstable (See the section on Loop Tuning). In contrast, a small gain results in a small output response to a large input error, and a less responsive (or sensitive) controller. If the proportional gain is too low, the control action may be too small when responding to system disturbances.

In the absence of disturbances pure proportional control will not settle at its target value, but will retain a steady state error that is a function of the proportional gain and the process gain. Despite the steady-state offset, both tuning theory and industrial practice indicate that it is the proportional term that should contribute the bulk of the output change.

### 2.4.2 Integral term

The contribution from the integral term is proportional to both the magnitude of the error and the duration of the error. Summing the instantaneous error over time (integrating the error) gives the accumulated offset that should have been corrected previously. The accumulated error is then multiplied by the integral gain and added to the controller output. The magnitude of the contribution of the integral term to the overall control action is determined by the integral gain, $K_i$.

The integral term is given by:

$$I_{OUT} = K_i \int_0^t e(\tau)d\tau$$

Where

- $I_{out}$: Integral output
- $K_i$: Integral Gain, a tuning parameter
- e: Error = SP − PV
- τ: Time in the past contributing to the integral response

The integral term (when added to the proportional term) accelerates the movement of the process towards set point and eliminates the residual steady-state error that occurs with a proportional only controller. However, since the integral term is responding to accumulated errors from the past, it can cause the present value to overshoot the setpoint value (cross over the set point and then create a deviation in the other direction). For further notes regarding integral gain tuning and controller stability, see the section on Loop Tuning.

### 2.4.3 Derivative term

The rate of change of the process error is calculated by determining the slope of the error over time (i.e. its first derivative with respect to time) and multiplying this rate of change by the derivative gain $K_d$. The magnitude of the contribution of the derivative term to the overall control action is determined the derivative gain, $K_d$.

The derivative term is given by:

$$D_{OUT} = K_d \frac{de}{dt}$$

Where

- $D_{out}$: Derivative output
- $K_d$: Derivative Gain, a tuning parameter

- e: Error = SP − PV
- t: Time or instantaneous time (the present)

The derivative term slows the rate of change of the controller output and this effect is most noticeable close to the controller setpoint. Hence, derivative control is used to reduce the magnitude of the overshoot produced by the integral component and improve the combined controller-process stability. However, differentiation of a signal amplifies noise in the signal and thus this term in the controller is highly sensitive to noise in the error term, and can cause a process to become unstable if the noise and the derivative gain are sufficiently large.

The output from the three terms, the proportional, the integral and the derivative terms are summed to calculate the output of the PID controller.

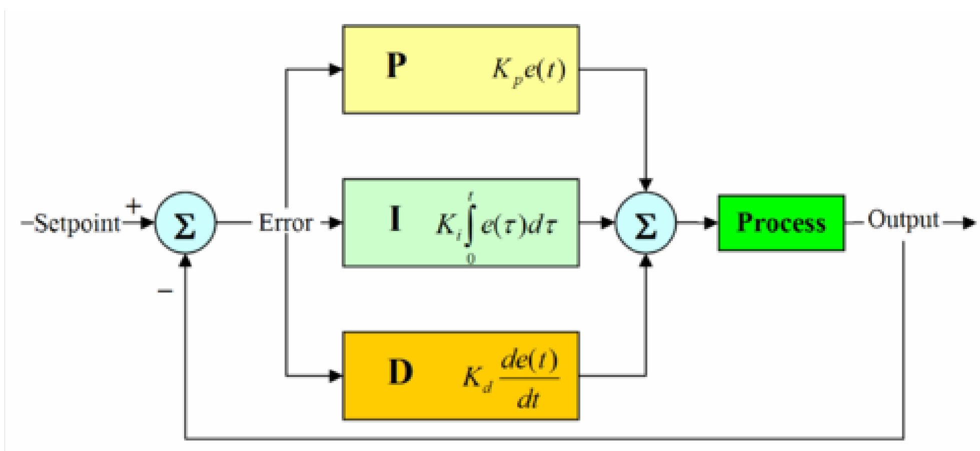


**Figure 2.1: Block diagram of a PID controller**

First estimation is the equivalent of the proportional action of a PID controller. The integral action of a PID controller can be thought of as gradually adjusting the output when it is almost right. Derivative action can be thought of as making smaller and smaller changes as one gets close to the right level and stopping when it is just right, rather than going too far. Making a change that is too large when the error is small is equivalent to a high gain controller and will lead to overshoot. If the controller were to repeatedly make changes

That were too large and repeatedly overshoot the target, this control loop would be termed unstable and the output would oscillate around the setpoint in a either a constant, a growing or a decaying sinusoid. A human would not do this because we are adaptive controllers, learning from the process history, but PID controllers do not have the ability to learn and must be set up correctly. Selecting the correct gains for effective control is known as tuning the controller.

If a controller starts from a stable state at zero error (PV = SP), then further changes by the controller will be in response to changes in other measured or unmeasured inputs to the process that impact on the process, and hence on the PV. Variables that impact on the process other than the MV are known as disturbances and generally controllers are used to reject disturbances and/or implement set point changes.

In theory, a controller can be used to control any process which has a measurable output (PV), a known ideal value for that output (SP) and an input to the process (MV) that will affect the relevant PV. Controllers are used in industry to regulate temperature, pressure, flow rate, chemical composition, level in a tank containing fluid, speed and practically every other variable for which a measurement exists. Automobile cruise control is an example of a process outside of industry which utilizes automated control. $K_p$: Proportional Gain - Larger $K_p$ typically means faster response since the larger the error, the larger the feedback to compensate. An excessively large proportional gain will lead to process instability. $K_i$: Integral Gain - Larger $K_i$ implies steady state errors are eliminated quicker. The trade-off is larger overshoot: any negative error integrated during transient response must be integrated away by positive error before we reach steady state. $K_d$: Derivative Gain - Larger $K_d$ decreases overshoot, but slows down transient response and may lead to instability.

### 2.4.4 Loop tuning

If the PID controller parameters (the gains of the proportional, integral and derivative terms) are chosen incorrectly, the controlled process input can be unstable, i.e. its output diverges, with or without oscillation, and is limited only by saturation or mechanical breakage. Tuning a control loop is the adjustment of its control parameters (gain/proportional band, integral gain/reset, derivative gain/rate) to the optimum values for the desired control response.

Some processes must not allow an overshoot of the process variable beyond the setpoint if, for example, this would be unsafe. Other processes must minimize the energy expended in reaching a new setpoint. Generally, stability of response (the reverse of instability) is required and the process must not oscillate for any combination of process conditions and setpoints. Some processes have a degree of non-linearity and so parameters that work well at full-load conditions don't work when the process is starting up from no-load. This section describes some traditional manual methods for loop tuning.

There are several methods for tuning a PID loop. The most effective methods generally involve the development of some form of process model, and then choosing P, I, and D based on the dynamic model parameters. Manual "tune by feel" methods have proven time and again to be inefficient, inaccurate, and often dangerous.]

The choice of method will depend largely on whether or not the loop can be taken "offline" for tuning, and the response time of the system. If the system can be taken offline, the best tuning method often involves subjecting the system to a step change in input, measuring the output as a function of time, and using this response to determine the control parameters.

| Choosing a Tuning Method | | |
|---|---|---|
| Method | Advantages | Disadvantages |
| Ziegler-Nichols | Proven Method. Online method. | Process upset, some trial-and-error, very aggressive tuning |
| Tune By Feel | No math required. Online method. | Erratic, not repeatable |
| Software Tools | Consistent tuning. Online or offline method. May include valve and sensor analysis. Allow simulation before downloading. | Some cost and training involved. |
| Cohen-Coon | Good process models. | Some math. Offline method. Only good for first-order processes. |

**Table 2.1**

If the system must remain online, one tuning method is to first set the I and D values to zero. Increase the P until the output of the loop oscillates, then the P should be left set to be approximately half of that value for a "quarter amplitude decay" type response. Then increase I until any offset is correct in sufficient time for the process. However too much I will cause instability. Finally, increase D, if required, until the loop is acceptably quick to reach its reference after a load disturbance. However too much D will cause excessive response and overshoot. A fast PID loop tuning usually overshoots slightly to reach the setpoint more quickly; however, some systems cannot accept overshoot, in which case a "critically damped" tune is required, which will require a P setting significantly less than half that of the P setting causing oscillation.

| Effects of increasing parameters | | | | |
|---|---|---|---|---|
| Parameter | Rise Time | Overshoot | Settling Time | S.S. Error |
| $K_p$ | Decrease | Increase | Small Change | Decrease |
| $K_i$ | Decrease | Increase | Increase | Eliminate |
| $K_d$ | Small Change | Decrease | Decrease | None |

**Table 2.2**

### 2.4.5 Ziegler-Nichols method

Another tuning method is formally known as the Ziegler-Nichols method, introduced by John G. Ziegler and Nathaniel B. Nichols. As in the method above, the I and D gains are first set to zero. The "P" gain is increased until it reaches the "critical gain" $K_c$ at which the output of the loop starts to oscillate. $K_c$ and the oscillation period $P_c$ are used to set the gains as shown:

| Ziegler-Nichols method | | | |
|---|---|---|---|
| Control Type | $K_p$ | $K_i$ | $K_d$ |
| P | $0.5 \cdot K_c$ | - | - |
| PI | $0.45 \cdot K_c$ | $1.2K_p / P_c$ | - |
| PID | $0.6 \cdot K_c$ | $2K_p / P_c$ | $K_pP_c / 8$ |

**Table 2.3**

### 2.4.6 Limitations of PID control

While PID controllers are applicable to many control problems, they can perform poorly in some applications. PID controllers, when used alone, can give poor performance when the PID loop gains must be reduced so that the control system does not overshoot, oscillate or "hunt" about the control setpoint value. The control system performance can be improved by combining the PID controller functionality with that of a Feed-Forward control output as described in Control Theory. Any information or intelligence derived from the system state can be "fed forward" or combined with the PID output to improve the overall system performance. The Feed-Forward value alone can often provide a major portion of the controller output. The PID controller can then be used to respond to whatever difference or "error" that remains between the controller setpoint and the feedback value. Since the Feed-Forward output is not a function of the process feedback, it can never cause the control system to oscillate, thus improving the system response and stability.

Another problem faced with PID controllers is that they are linear. Thus, performance of PID controllers in non-linear systems (such as HVAC systems) is variable. Often PID controllers are enhanced through methods such as gain scheduling or fuzzy logic. Further practical application issues can arise from instrumentation connected to the controller. A high enough sampling rate and measurement precision and measurement accuracy (more relevant to FF and MPC).

A problem with the differential term is that small amounts of measurement or process noise can cause large amounts of change in the output. Sometimes it is helpful to filter the measurements, with a running average, also known as a low-pass filter. However, low-pass filtering and derivative control cancel each other out, so reducing noise by instrumentation means is a much better choice. Alternatively, the differential band can be turned off in most systems with little loss of control. This is equivalent to using the PID controller as a PI controller.

# CHAPTER 3

# METHODOLOGY

## 3.1 INTRODUCTION

This chapter explains detail about the design of the whole system developed which are buck converter models with switching frequency 100 kHz and 400 kHz. The analysis will be shows are the comparison between 100 kHz with and without PID controller and the comparison between 100 kHz with PID controller and 400 kHz with PID controller. That analysis will be described in the next chapter. The overall of the system is shown in figure 3.0 below.

## 3.2 The Design of Buck Converter.

For the model consist of RL and Resr. These topologies are nonisolated, that is, the input and output voltages share a common ground. There are, however, isolated derivations of these nonisolated topologies. The power supply topology refers to how the switches, output inductor, and output capacitor are connected. Each topology has unique properties. These properties include the steady-state voltage conversion ratios, the nature of the input and output currents, and the character of the output voltage ripple. Another important property is the frequency response of the duty-cycle-to-output-voltage transfer function. The most common and probably the simplest power stage topology is the buck power stage, sometimes called a step-down power stage.The input current for a buck power stage is discontinuous or pulsating due to the power switch (Q1) current that pulses from zero to IO every switching cycle. The output current for a buck power stage is continuous or nonpulsating because the output current is supplied by the output inductor/capacitor combination; the output capacitor never supplies the entire load current (for continuous inductor current mode operation, one of the two operating modes to be discussed in the next section). [6]

Figure 3.0 shows a simplified schematic of the buck power stage with a drive circuit block included. The power switch, Q1, is an n-channel MOSFET. The diode, CR1, is usually called the catch diode, or freewheeling diode. The inductor, L, and capacitor, C, make up the output filter. The capacitor ESR, $R_C$, (equivalent series resistance) and the inductor DC resistance, $R_L$ , are included in the analysis. The resistor, R, represents the load seen by the power stage output.

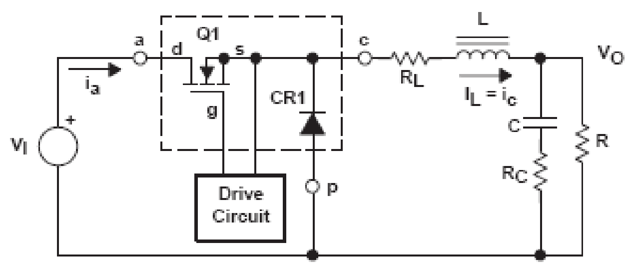


**Figure 3.0: Buck Power Stage Schematic**

During normal operation of the buck power stage, Q1 is repeatedly switched on and off with the on and off times governed by the control circuit. This switching action causes a train of pulses at the junction of Q1, CR1, and L which is filtered by the L/C output filter to produce a dc output voltage, $V_O$. A more detailed quantitative analysis is given in the following sections.

## 3.3 Buck Steady-State Continuous Conduction Mode Analysis

The following is a description of steady-state operation in continuous conduction mode. Steady-state implies that the input voltage, output voltage, output load current, and duty-cycle are fixed and not varying. In continuous conduction mode, the Buck power stage assumes two states per switching cycle. The ON state is when Q1 is ON and CR1 is OFF. The OFF state is when Q1 is OFF and CR1 is ON. A simple linear circuit can represent each of the two states where the switches in the circuit are replaced by their equivalent circuits during each state. The circuit diagram for each of the two states is shown in Figure 3.1.

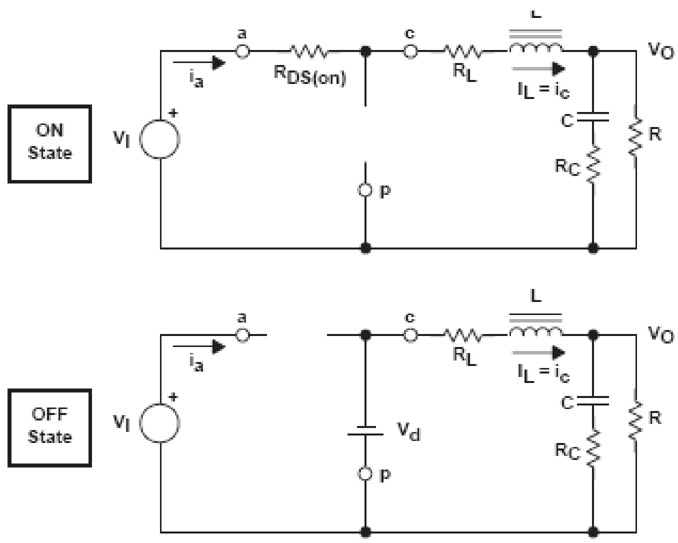


**Figure 3.1: Buck Power Stage States**

The duration of the ON state is D × $T_S = T_{ON}$ where D is the duty cycle, set by the control circuit, expressed as a ratio of the switch ON time to the time of one complete switching cycle, Ts . The duration of the OFF state is called $T_{OFF}$ . Since there are only two states per switching cycle for continuous mode, $T_{OFF}$ is equal to (1–D) × TS . The quantity (1–D) is sometimes called D’. The amount that the inductor current increases can be calculated by using a version of the familiar relationship

$$V_L = L\frac{di_L}{dt} \Rightarrow \Delta I_L = \frac{V_L}{L}\Delta T$$

The inductor current increase during the ON state is given by:

$$\Delta I_L(+) = \frac{(V_{I-}V_{DS} - I_L R_L) - V_o}{L} T_{ON}$$

This quantity, DIL(+), is referred to as the inductor ripple current.

Referring to Figure 3.1, when Q1 is OFF, it presents a high impedance from its drain to source. Therefore, since the current flowing in the inductor L cannot change instantaneously, the current shifts from Q1 to CR1. Due to the decreasing inductor current, the voltage across the inductor reverses polarity until rectifier CR1 becomes forward biased and turns ON. The voltage on the left-hand side of L becomes –($V_D$ + $I_L$ × $R_L$ ) where the quantity, $V_D$ , is the forward voltage drop of CR1. The voltage applied to the right hand side of L is still the output voltage, $V_O$. The inductor current, IL , now flows from ground through CR1 and to the output capacitor and load resistor combination. During the OFF state, the magnitude of the voltage applied across the inductor is constant and equal to ($V_O$ + $V_D$ + $I_L$ × $R_L$ ). Maintaining our same polarity convention, this applied voltage is negative (or opposite in polarity from the applied voltage during the ON time). Hence, the inductor current decreases during the OFF time. Also, since the applied voltage is essentially constant, the inductor current decreases linearly. This decrease in inductor current during $T_{OFF}$ is illustrated in Figure 3.1. The inductor current decrease during the OFF state is given by:

$$\Delta I_L(-) = \frac{V_o + (Vd + I_L R_L)}{L} T_{OFF}$$

This quantity, DIL(–), is also referred to as the inductor ripple current.

In steady state conditions, the current increase, $\Delta I_L$ (+), during the ON time and the current decrease during the OFF time, $\Delta I_L$ (–), must be equal. Otherwise, the inductor current would have a net increase or decrease from cycle to cycle which would not be a steady state condition. Therefore, these two equations can be equated and solved for $V_O$ to obtain the continuous conduction mode buck voltage conversion relationship. In the above equations for $\Delta I_L$ (+) and $\Delta I_L$ (–), the dc output voltage was implicitly assumed to be constant with no AC ripple voltage during the ON time and the OFF time. This is a common simplification and involves two separate effects. First, the output capacitor is assumed to be large enough that its voltage change is negligible. Second, the voltage across the capacitor ESR is also assumed to be negligible. These assumptions are valid because the ac ripple voltage is designed to be much less than the dc part of the output voltage. The above voltage conversion relationship for $V_O$ illustrates the fact that $V_O$ can be adjusted by adjusting the duty cycle, D, and is always less than the input because D is a number between 0 and 1. A common simplification is to assume $V_{DS}$, $V_D$ , and RL are small enough to ignore. Setting $V_{DS}$, $V_D$ , and $R_L$ to zero, the above equation simplifies considerably to:

$$V_o = V_I D$$

To relate the inductor current to the output current, referring to Figures 3.1, note that the inductor delivers current to the output capacitor and load resistor combination during the whole switching cycle. The inductor current averaged over the switching cycle is equal to the output current. This is true because the average current in the output capacitor must be zero. In equation form, we have:

$$I_L(avg) = I_o$$

In switching converter the function of output capacitance is to store energy. The energy is stored in the capacitor’s electric field due to the voltage applied. Thus, qualitatively, the function of a capacitor is to attempt to maintain a constant voltage. The value of output capacitance of a Buck power stage is generally selected to limit output voltage ripple to the level required by the specification. Since the ripple current in the output inductor is usually already determined, the series impedance of the capacitor primarily determines the output voltage ripple. The three elements of the capacitor that contribute to its impedance (and output voltage ripple) are equivalent series resistance (ESR), equivalent series inductance (ESL), and capacitance (C). The following gives guidelines for output capacitor selection. For continuous inductor current mode operation, to determine the amount of capacitance needed as a function of inductor current ripple, $\Delta I_L$, switching frequency, $f_S$, and desired output voltage ripple, $\Delta V_O$, the following equation is used assuming all the output voltage ripple is due to the capacitor's capacitance

$$C \geq \frac{\Delta I_L}{8 f_S \Delta V_O}$$

In many practical designs, to get the required ESR, a capacitor with much more capacitance than is needed must be selected. For both continuous or discontinuous inductor current mode operation and assuming there is enough capacitance such that the ripple due to the capacitance can be ignored, the ESR needed to limit the ripple to $\Delta V_o V$ peak-to-peak is:

$$ESR \leq \frac{\Delta V_O}{\Delta I_L}$$

Ripple current flowing through a capacitor's ESR causes power dissipation in the capacitor. This power dissipation causes a temperature increase internal to the capacitor. Excessive temperature can seriously shorten the expected life of a capacitor. Capacitors have ripple current ratings that are dependent on ambient temperature and should not be exceeded. Referring to Figure 3.1, the output capacitor ripple current is the inductor current, $I_L$, minus the output current, $I_O$.

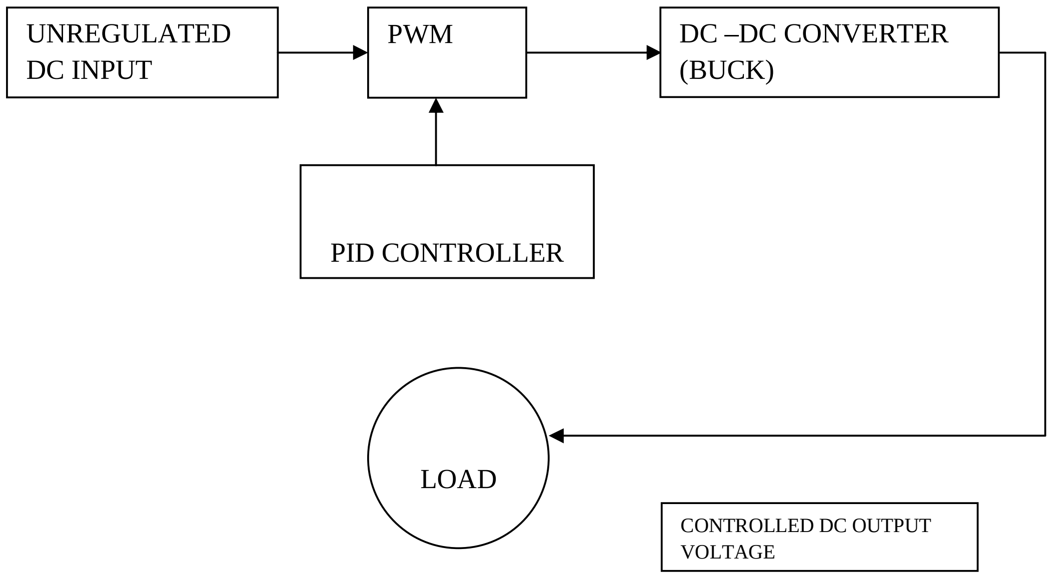


**Figure 3.2: The block diagram of the buck converter incorporated with PID controller**

## 3.4 DC TO DC CONVERTER (BUCK) INCORPORATED WITH PID CONTROLLER DESIGN

The DC to DC converter buck is design base on continues conduction mode (CCM) as briefly explained before. By using CCM theory, all buck circuit parameter is determined. The parameter obtained is stated below:

| | | |
|---|---|---|
| $V_{in}$ =12 V | C=376uF | $R_L$ =80m$\Omega$ |
| $V_{out}$ =5 V | $f_s$ =100 kHz | $R_{esr}$ =5m$\Omega$ |
| L =4.1uH | D=0.42 | $R_{load}$ =1$\Omega$ |

System equation

$$\frac{di_L}{dt} = \frac{1}{L}(V_s D - i_L R_L - V_o) \qquad V_o = V_c + R_{esr}(i_L - i_{out})$$

$$\frac{dv_c}{dt} = \frac{1}{C}(i_L - i_O)$$

The simulink model of open loop buck converter system is shown in figure 3.2 and the close loop model is shown in figure 3.3 as below:

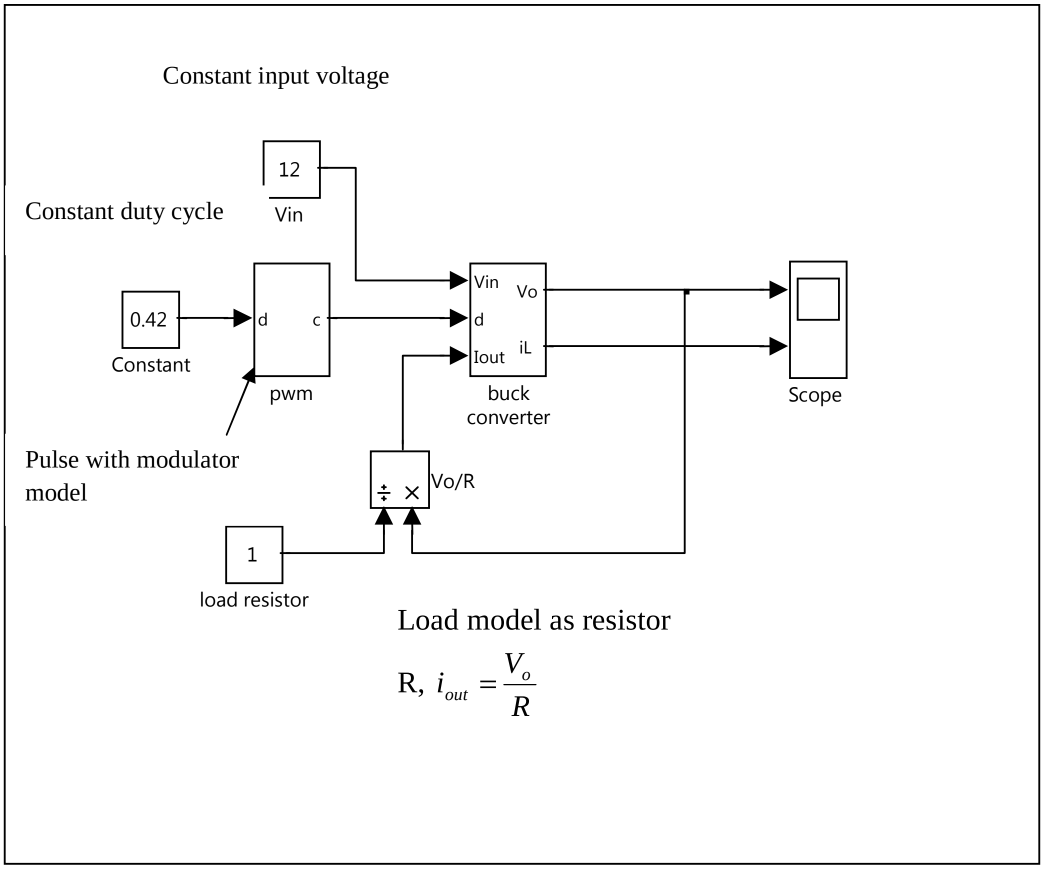


**Figure 3.3**

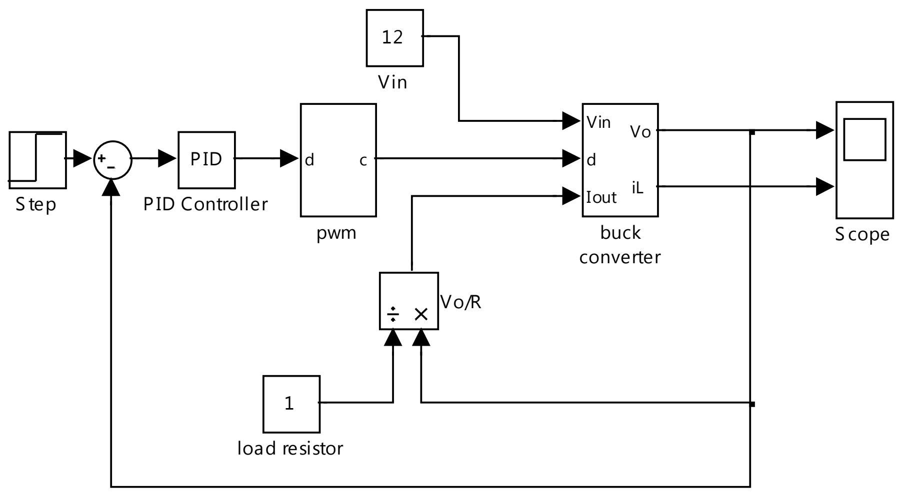
12
Vin
Step
PID
PID Controller
d
c
pwm
Vin
d
Iout
Vo
iL
buck
converter
Scope
Vo/R
÷ ×
1
load resistor

**Figure 3.4**