

**DEVELOPMENT OF DC-DC CONVERTER FOR DC MOTOR USING NEURAL  
NETWORK CONTROLLER**

**RASYIDI BIN ABDULLAH**

A project report submitted in partial  
fulfilment of the requirement for the award of the  
Degree of Master of Electrical Engineering

Faculty of Electrical and Electronics Engineering  
Universiti Tun Hussein Onn Malaysia

JULY 2013

## **ABSTRACT**

A neural network of DC-DC converter is designed and presented in this project. In order to control the output speed of the DC-DC converter, the controller is designed to change the duty cycle of the converter. The mathematical model of DC-DC converter and neural network controller are derived to design simulation model. The simulation is developed on Matlab simulation program. To verify the effectiveness of the simulation model, an experimental set up is developed. The neural network controller to generate duty cycle of PWM signal is programmed. The simulation and experimental results will show output speed of the DC-DC converter can be controlled according to the value of duty cycle.

## **ABSTRAK**

Neural Network dari penukar DC-DC direka dan dibentangkan di dalam projek ini. Dalam usaha untuk mengawal kelajuan output penukar DC-DC, pengawal direka untuk menukar kitaran tugas penukar. Model matematik penukar DC-DC dan neural network controller diperolehi untuk mereka bentuk model simulasi. Simulasi dibangunkan atas program simulasi Matlab. Untuk memastikan keberkesanan model simulasi, satu eksperimen dibangunkan. Neural network controller akan menjana kitaran tugas isyarat PWM diprogramkan. Simulasi dan keputusan eksperimen akan menunjukkan kelajuan output DC-DC converter boleh dikawal mengikut nilai kitar tugas.

## CONTENT

	<b>TITLE</b>	I
	<b>DECLARATION</b>	II
	<b>DEDICATION</b>	III
	<b>ACKNOWLEDGMENT</b>	IV
	<b>ABSTRACT</b>	V
	<b>CONTENT</b>	VII
	<b>LIST OF FIGURES</b>	IX
	<b>LIST OF TABLES</b>	XI
	<b>LIST OF SYMBOLS AND ABBREVIATIONS</b>	XII
<b>CHAPTER 1</b>	<b>INTRODUCTION</b>	1
	1.1 Motivation	1
	1.2 Project Background	2
	1.3 Problem Statement	3
	1.4 Project Objectives	4
	1.5 Project Scopes	4
<b>CHAPTER 2</b>	<b>LITERITURE REVIEW</b>	5
	2.1 Technology Development	5
	2.2 Theory Of Operation Buck Converter	6
	2.3 The DC-DC Buck Converter	10
	2.4 Neural Network	11
	2.5 DC Motor	12
	2.6 PID Controller	13

<b>CHAPTER 3</b>	<b>METHODOLOGY</b>	15
3.1	Project Design	15
3.2	Proposed DC Motor	18
3.3	Proposed Proportional Integral Controller	19
3.4	Proposed PID Controller	20
3.5	Architecture Of Neural Network Controller	21
3.6	Proposed Neural Network Controller	21
<b>CHAPTER 4</b>	<b>RESULT AND ANALYSIS</b>	25
4.1	DC Motor Speed Using PI Controller	25
4.2	DC Motor Speed Using PID Controller	27
4.3	DC Motor Speed Using NN Controller	29
<b>CHAPTER 5</b>	<b>CONCLUSION AND RECOMMENDATION</b>	34
5.1	Existing Paper	34
5.2	Future Works	34
	<b>REFERENCES</b>	35

## LIST OF FIGURES

2.1	Buck Converter Circuit	6
2.2	PWM signal to control the switches in the DC-DC converter	6
2.3	Equivalent circuit of the buck converter when the switch is closed	7
2.4	Equivalent circuit of the buck converter when the switch is open	7
2.5	Ideal switch, (a) used to reduce the voltage dc component	8
2.6	Output voltage waveform $v_s(t)$ .	8
2.7	Output voltage dc component by the switching period	9
2.8	Insertion of low-pass filter, to remove switching harmonics and pass only the dc component of $v_s(t)$ to the output.	10
2.9	Buck converter dc output the voltage $V$ vs. duty cycle $D$ .	10
2.10	DC buck converter topology	10
2.11	Buck converter circuit when switch: (a) turns on (b) turns off	11
2.12	Matlab/Simulink model of DC motor	11
2.13	A perceptron network with three layers	12
2.14	A simulation model of PID controller	13
3.1	Block Diagram For Propose DC-DC Buck Converter Using Neural Network Controller	15
3.2	The methodology Flow Chart	17
3.3	Matlab model of DC motor	18
3.4	Block diagram of PI controller	19
3.5	PI controller Matlab Simulink block	20
3.6	The model for PID controller in MATLAB/Simulink	20
3.7	Architecture of Neural Network	21
3.8	NN block	24
3.9	Layer 1 block	24
3.10	Layer 2 block	24
4.1	Simulation model for PI controller	25

4.2	Simulation output speed for PI controller	26
4.3	Maximum overshoot and ripple	26
4.4	Output speed for PI controller	27
4.5	Simulation model with PID controller	27
4.6	Maximum overshoot and ripple with PID controller	28
4.7	Output speed for PID controller	28
4.8	Simulation diagram for NN controller	29
4.9	Simulation output speed for NN controller	29
4.10	Maximum overshoot and ripple with NN controller	30
4.11	Output speed for NN controller	30
4.12	NN training tools	31
4.13	NN training performance	31
4.14	NN training error histogram	32
4.15	Simulation result of error and duty cycle	32

**LIST OF TABLES**

4.1	Comparison between PI controller and NN controller	33
-----	--	----



**LIST OF SYMBOLS AND ABBREVIATIONS**

$D$	-	Duty Cycle
$T$	-	Time
$V_s$	-	Supply voltage
$V_o$	-	Output voltage
$M_v$	-	Ratio
$V_{ref}$	-	Reference voltage
$e$	-	Error
$J$	-	Inertia
$L$	-	Inductance
$C$	-	Capacitor
$R$	-	Resistance
$N_o$	-	Output speed
$N_{ref}$	-	Reference speed
$S_w$	-	Switch

## **CHAPTER 1**

### **INTRODUCTION**

#### **1.1 Motivation**

The switched mode dc-dc converters are some of the simplest power electronic circuits which convert one level of electrical voltage into another level by switching action. These converters have received an increasing deal of interest in many areas. This is due to their wide applications like power supplies for personal computers, office equipment, appliance control, telecommunication equipment, DC motor drives, automotive, aircraft, etc.

The commonly used control methods for dc-dc converters are pulse width modulated (PWM) voltage mode control, PWM current mode control with proportional (P), proportional integral (PI), and proportional integral derivative (PID) controller. These conventional control methods like P, PI, and PID are unable to perform satisfactorily under large parameter or load variation.

Therefore, the motivation of this thesis is to control the output DC motor speed of a dc-dc buck converter through neural network control (NNC). Hence, this thesis focused open loop circuit, open loop circuit using proportional integral (PI) and neural network control (NNC) for dc-dc buck converter circuit. The circuit is design with the equation and the comparison of motor speed is shown in this report.

## 1.2 Project Background

General idea of DC-DC converter to convert a fixed voltage dc source into a variable voltage dc source. The output voltage of the DC-DC converter can be higher or lower than the input. DC-DC Converter widely used for traction motor in electric automobiles, trolley cars, marine hoists, and forklift trucks. They provide smooth acceleration control, high efficiency, and fast dynamic response. Dc converter can be used in regenerative braking of dc motor to return energy back into the supply, and this feature results in energy saving for transportation system with frequent stop; and also are used, in dc voltage regulation. There are many types of dc-dc converter which is buck (step down converter), boost (step-up) converter, and buck-boost (step up- step-down) convertor. (Muhammad H. Rashid, 2004).

Classical PID controllers are commonly used in industries due to their simplicity and ease of implementation (Rubaii, A. et al., 2008). In linear system model, controller parameters of the PID controller are easy to determine and resulting good control performances. However, for nonlinear system model applications such as BLDC motor drive, control performance of the PID controller becomes poor and difficult to determine the controller parameters (Hong, W. et al., 2007 and Tipsuwanporn, V, et al., 2002).

In order to improve control performance of the BLDC motor drive, several intelligence controllers such as fuzzy logic control, neural network control and hybrid neuro-fuzzy control methods for BLDC motor have been reported in (Rubaii, A. et al., 2008, Tipsuwanporn, V. et al., 2002, Mahdavi, J. et al., 2011, Lee, B. K. et al., 2003, Cunkas, M. et al., 2010, Gokbulut, M. et al., 2007, Ji, H. et al., 2008 and Ji, H. et al., 1997).

In addition, the NNC also have been applied for several others power electronic and motor drive application (El-Balluq, T. NN. et al., 2004 and Shanmugasundram, R. et al., 2009). In order to improve performance of the NNC some researchers have been done to develop online learning scheme of the NNC.

In this project, a complete simulation model with neural network control (NNC) method for DC motor drive is proposed using MATLAB/Simulink. The developed NNC has the ability to learn instantaneously and adapt its own controller parameters based on external disturbance and internal of the converter with minimum steady state error, overshoot and rise time of the output voltage.

### **1.3 Problem Statement**

DC-DC converter consists of power semiconductor devices which are operated as electronic switches. Operation of the switching devices causes the inherently nonlinear characteristic of the DC-DC converters including one known as the Buck Converter. The switching technique of the Buck converter causes the converter system to be nonlinear system. Nonlinear system requires a controller with higher degree of dynamic response. Proportional-Integral (PI) is one of the controllers used as a switching device for the converter. However the PI controller is known to exhibit sluggish disturbance rejection properties [5].

A study by Zulkiflie Ibrahim and Emil Levi (2002) shows that the PI speed control offers high speed dip and large recovery time when the load is connected. Therefore the implementation of Neural Network that will deal the issue must be investigated. Since the Buck converter is a nonlinear system, the Neural Network controller method will be developed to improve overshoot speed at starting of the motor and settling time. The developed Neural Network controller has the ability to learn instantaneously and adapt its own controller parameters based on external disturbances and internal variation of the converter. Thus this neural network can overcome the problem stated to obtain better performances in terms of speed control.

## **1.4 Project Objectives**

The objectives of this project are:-

- i. To develop DC-DC buck converter using Proportional Integrated (PI) Controller
- ii. To develop DC-DC buck converter using Neural Network Controller (NNC)
- iii. To compare the NNC with PI performance in terms of speed overshoot, settling time and ripple factor

## **1.5 Project Scopes**

The scopes of the project are:

- i) Modelling the DC-DC Buck converter with DC Motor
- ii) Modelling the Proportional-Integrated (PI) controller for speed control
- iii) Modelling the Neural Network controller for speed control
- iv) Compare the output speed of the DC motor for both PI and NN controller in terms of starting overshoot, ripple factor and settling time.

## **CHAPTER 2**

### **LITERATURE REVIEW**

#### **2.1 Technology Development**

Switch mode DC-DC converters efficiently convert an unregulated DC input voltage into a regulated DC output voltage. Compared to linear power supplies, switching power supplies provide much more efficiency and power density. Switching power supplies employ solid-state devices such as transistors and diodes to operate as a switch either completely on or completely off [4].

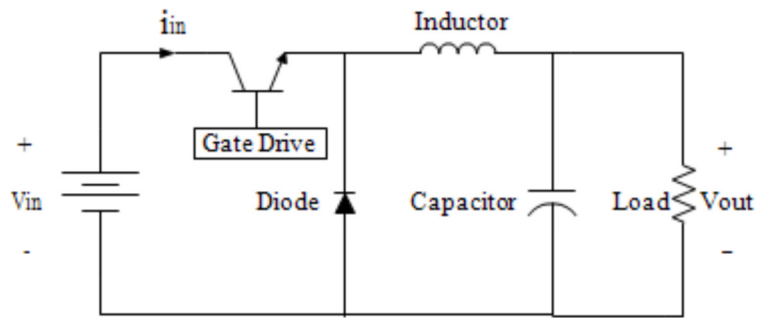
Energy storage elements including capacitors and inductors are used for energy transfer and work as a low-pass filter. The buck converter and the boost converter are the two fundamental topologies of switch mode DC-DC converters. Most of the other topologies are either buck-derived or boost-derived converters, because their topologies are equivalent to the buck or the boost converters [2].

Traditionally, the control methodology for DC-DC converters has been analog control. In the recent years, technology advances in very-large-scale integration (VLSI) have made digital control of DC-DC converters with microcontrollers and digital signal processors (DSP) possible. The major advantages of digital control over analog control are higher immunity to environmental changes such as temperature and changing of components, increased flexibility by changing the software, more advanced control techniques and shorter design cycles.

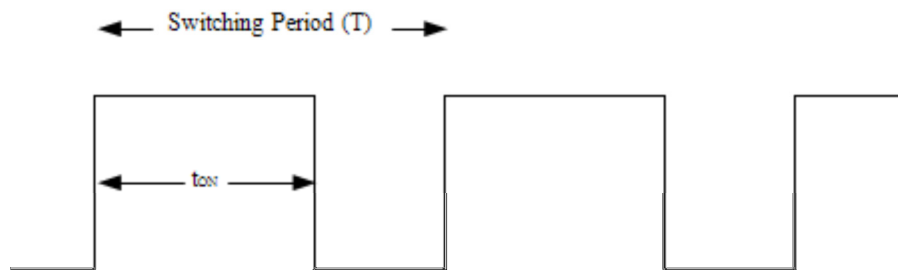
## 2.2 Theory of Operation Buck Converter

The operation of the buck converter is fairly simple, with an inductor and two switches (usually a transistor and a diode) that control the inductor. It alternates between connecting the inductor to source voltage to store energy in the inductor and discharging the inductor into the load.

The buck converter, shown in Figure 2.1, converts the unregulated source voltage  $V_{in}$  into a lower output voltage  $V_{out}$ . The NPN transistor shown in Figure 1 works as a switch. The ratio of the ON time ( $t_{ON}$ ) when the switch is closed to the entire switching period ( $T$ ) is defined as the duty cycle  $D = t_o/T$ . The corresponding PWM signal is shown in Figure 2.2 [10].



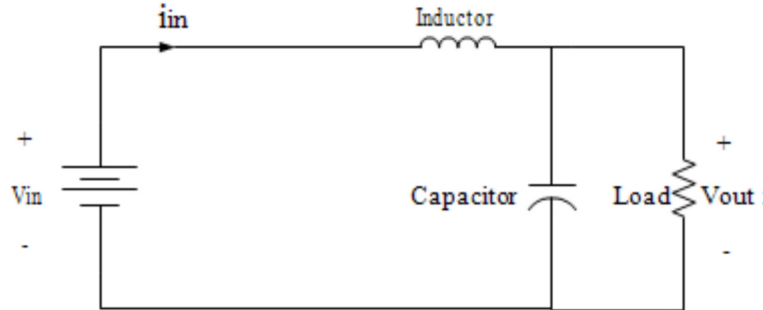
**FIGURE 2.1:** Buck Converter Circuit



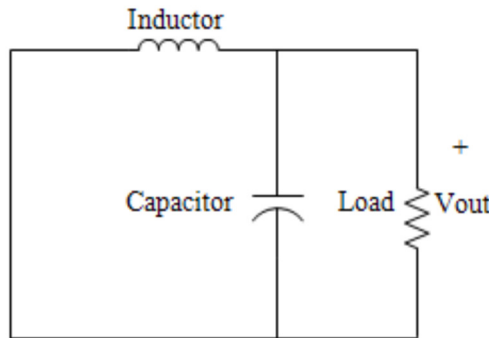
**FIGURE 2.2:** PWM signal to control the switches in the DC-DC converter

The equivalent circuit in Figure 2.3 is valid when the switch is closed. The diode is reverse biased, and the input voltage supplies energy to the inductor, capacitor and the load. When the switch is open as shown in Figure 2.4, the diode conducts, the capacitor supplies energy to the load, and the inductor current flows

through the capacitor and the diode [2]. The output voltage is controlled by varying the duty cycle. On steady state, the ratio of output voltage over input voltage is  $D$ , given by  $V_{out}/V_{in}$ .



**FIGURE 2.3:** Equivalent circuit of the buck converter when the switch is closed

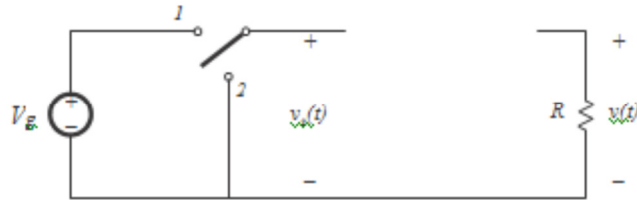


**FIGURE 2.4:** Equivalent circuit of the buck converter when the switch is open

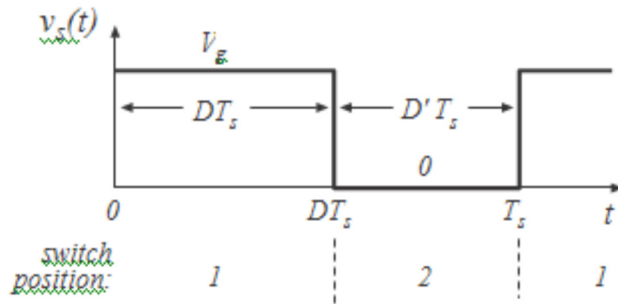
A buck converter is a step-down DC to DC converter. Its design is similar to the step-up boost converter, and like the boost converter it is a switched-mode power supply that uses two switches (a transistor and a diode), an inductor and a capacitor. The buck converter reducing the dc voltage, using only non-dissipative switches, inductors, and capacitors. The switch produces a rectangular waveform  $v_s(t)$  as illustrated in Figure 2.5. The voltage  $v_s(t)$  is equal to the dc input voltage  $V_g$  when the switch is in position 1, and is equal to zero when the switch is in position 2. In practice, the switch is realized using power semiconductor devices, such as transistors and diodes, which are controlled to turn on and off as required to perform the function of the ideal equal to the inverse of the switching period  $T_s$ , generally lies in the range of switching speed of the semiconductor devices.



The duty ratio  $D$  is the fraction of time which the switch spends in position 1, and is a number between zero and one. The complement of the duty ratio,  $D'$ , is defined as  $(1-D)$  [2].



**FIGURE 2.5:** Ideal switch, (a) used to reduce the voltage dc component



**FIGURE 2.6:** Output voltage waveform  $v_s(t)$ .

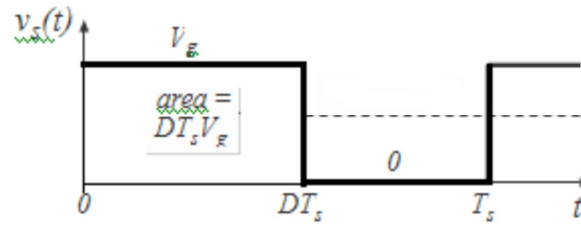
The switch reduces the dc component of the voltage: the switch output voltage  $v_s(t)$  has a dc component which is less than the converter dc input voltage  $V_g$ . From Fourier analysis, we know that the dc component of  $v_s(t)$  is given by its average value  $\langle v_s \rangle$ , or

$$\langle V_s \rangle = \frac{1}{T_s} \int_0^{T_s} v_s(t) \cdot dt \quad (2.1)$$

As illustrated in Figure 2.7, the integral is given by the area under the curve, or  $DT_s V_g$ . The average value is therefore

$$\langle V_s \rangle = \frac{1}{T_s} DT_s V_g = DV_g \quad (2.2)$$

So the average value, or dc component, of  $v_s(t)$  is equal to the duty cycle times the dc input voltage  $V_g$ . The switch reduces the dc voltage by a factor of  $D$ .



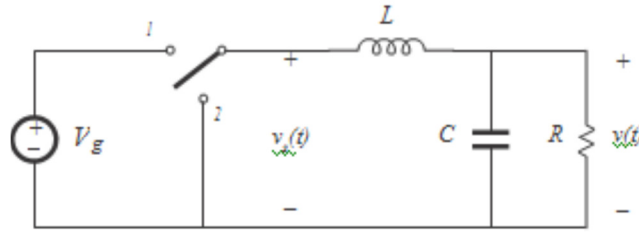
**FIGURE 2.7:** Output voltage dc component by the switching period

What remains is to insert a low-pass filter as shown in Figure 2.7. The filter is designed to pass the dc component of  $v_s(t)$ , but to reject the components of  $v_s(t)$  at the switching frequency and its harmonics. The output voltage  $v(t)$  is then essentially equal to the dc component of  $v_s(t)$ :

$$V < V_s > = DV_g \quad (2.2)$$

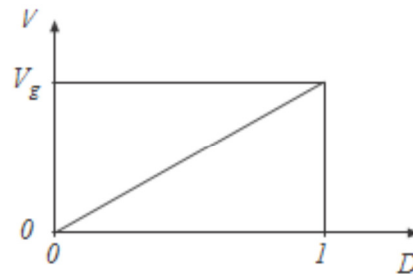
The converter of Figure 2.8 has been realized using lossless elements. To the extent that they are ideal, the inductor, capacitor, and switch do not dissipate power. For example, when the switch is closed, its voltage drop is zero, and the current is zero when the switch is open. In either case, the power dissipated by the switch is zero. Hence, efficiencies approaching 100% can be obtained. So to the extent that the components are ideal, we can realize our objective of changing dc voltage levels using a lossless network.

The network of Figure 2.8 also allows control of the output. Figure 2.9 is the control characteristic of the converter. The output voltage, given by equation (2.3), is plotted vs. duty cycle. The buck converter has a linear control characteristic. Also, the output voltage is less than or equal to the input voltage, since  $0 < D < 1$ . Feedback systems are often constructed which adjust the duty cycle  $D$  to regulate the converter output voltage. Inverters or power amplifiers can also be built, in which the duty cycle varies slowly with time and the output voltage follows [3].



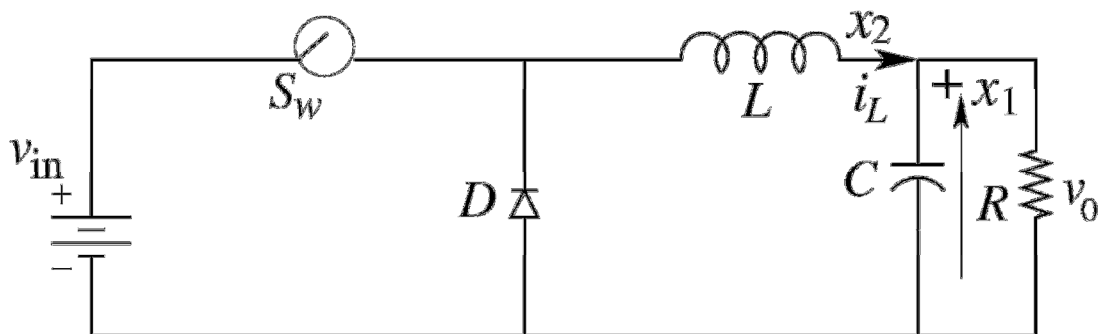
**FIGURE 2.8:** Insertion of low-pass filter, to remove switching harmonics and pass only the dc component of  $v_s(t)$  to the output.

### 2.3 The DC-DC buck converter

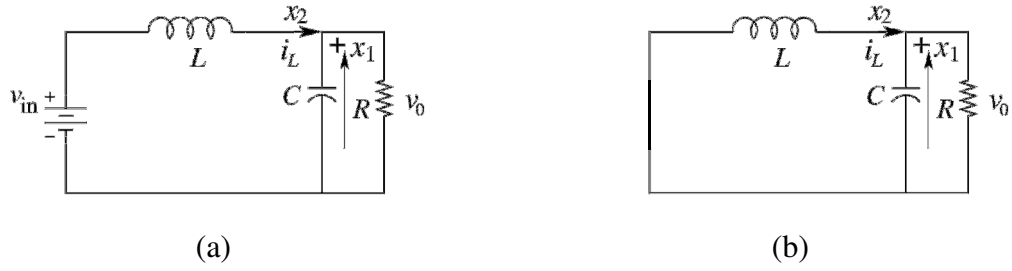


**FIGURE 2.9:** Buck converter dc output the voltage  $V$  vs. duty cycle  $D$ .

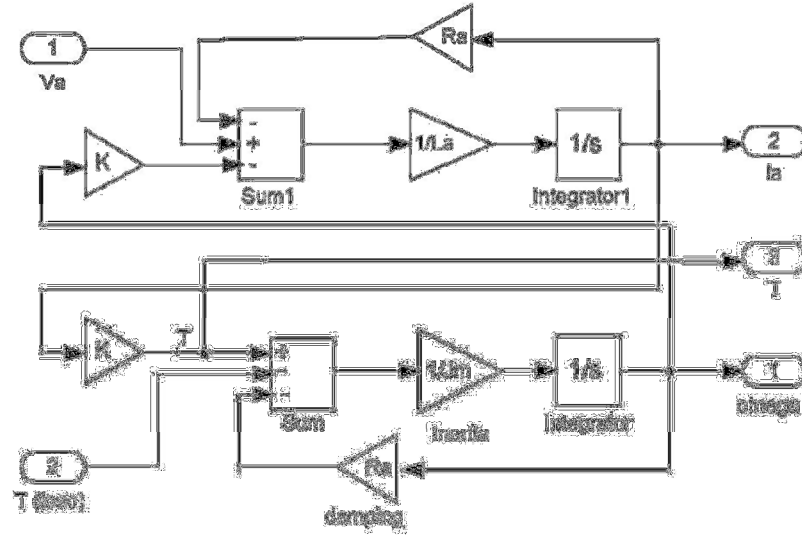
Figure 2.9: Buck converter dc output the voltage  $V$  vs. duty cycle  $D$ . The buck converter circuit converts a higher dc input voltage to lower dc output voltage. The basic buck dc-dc converter topology is shown in figure. 2.10. It consists of a controlled switch  $S_w$ , an uncontrolled switch  $D$  (diode), an inductor  $L$ , a capacitor  $C$ , and a load resistance  $R$



**FIGURE 2.10:** DC buck converter topology



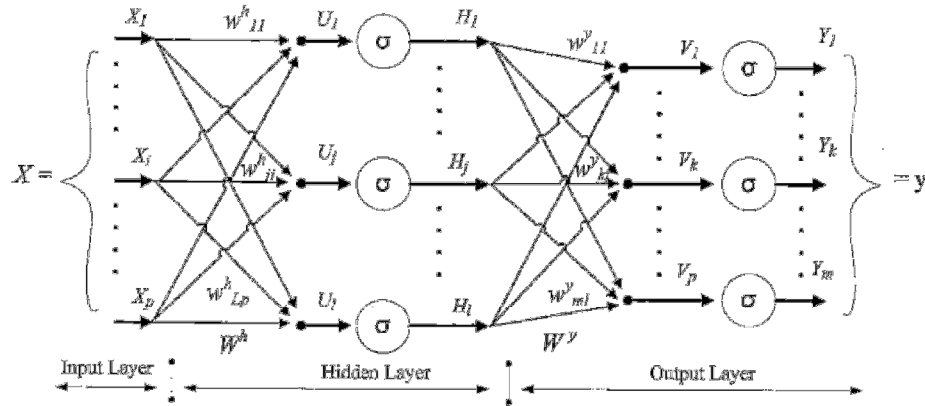
**FIGURE 2.11:** Buck converter circuit when switch: (a) turns on (b) turns off



**FIGURE 2.12:** Matlab/Simulink model of DC motor

## 2.4 Neural network

Figure 2.1 illustrates a Multilayer Perceptron Neural Network model. This network consists of an input layer (on the left) with three neurons, one hidden layer (in the middle) with three neurons and an output layer (on the right) with three neurons. Each layer has some neurons that are connected to the next layer through the link. The input layer of the neural network serves as an interface that takes information from the outside world and transmits it to the internal processing units of the network. Similarly, the output layer sends information from neural network's internal unit to the external world. The nodes in hidden layers are the neural network's processing units.



**FIGURE 2.13:** A perceptron network with three layers

Input layer is a vector of variable ( $x_1, \dots, x_p$ ) that represented to the input. The input layer (or processing before the input layer) standardizes these values so that the range of each variable is -1 to +1. The input layer distributes the values to each of the neurons in the hidden layer. In addition to the predictor variables, there is constant input of 1.0, called the bias that fed to each of the hidden layers. The bias is multiplied by a weight and added to the sum going into the neuron. During hidden layer, the value from each input neuron is multiplied by a weight ( $w_{ji}$ ), and the resulting weighted values are added together producing a combined value  $u_j$ . The weighted sum ( $u_j$ ) is fed into a transfer function,  $\sigma$  which outputs a value  $h_j$ . The outputs from hidden layer are distributed to the output layer.

Output layer occurred when the value from each hidden layer neuron is multiplied by a weight ( $w_{kj}$ ), and lastly the weighted values are added together producing a combined value  $v_j$ . The weighted sum ( $v_j$ ) is converted into a transfer function,  $\sigma$  which output a value  $y_k$ . The  $y$  values are the outputs of the network

## 2.5 DC motor

The resistance of the field winding and its inductance of the motor used in this study are represented by  $R_f$  and  $L_a$  respectively in dynamic model. Armature reactions effects are ignored in the description of the motor. This negligence is justifiable to minimize the effects of armature reaction since the motor used has

either interpoles or compensating winding. The fixed voltage  $V_f$  is applied to the field and the field current settles down to a constant value.

A linear model of a simple DC motor consists of a mechanical equation and electrical equation as determined in the following equations 3.1 and 3.2

$$J_m \frac{d\omega_m}{dt} = K_m \Phi I_a - b\omega_m - M_{load} \quad (2.1)$$

$$L_a \frac{dI_a}{dt} = V_a - R_a I_a - K_b \Phi \omega_m \quad (2.2)$$

Where

$R_a$  = Armature Resistance (Q).

$L_a$  = Armature Inductance (H).

$J_m$  = Motor of inertia ( $\text{kg.m}^2/\text{S}^2$ ).

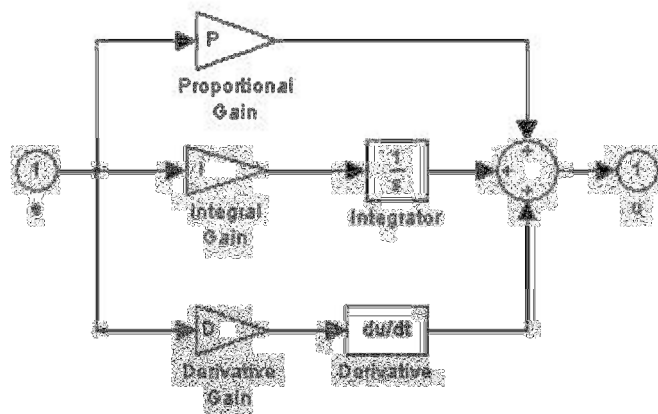
$K = K_b \Phi$  = Motor Constant (Nm / Amp).

$K = K_m \Phi$  = Motor Constant (Nm I Amp).

$b$  = Damping ration of mechanical system (Nms).

## 2.6 PID Controller

Consider the characteristics parameters – proportional (P), integral (I), and derivative (D) controls, as applied to the diagram below in Fig.2, the system,



**FIGURE 2.14:** A simulation model of PID controller

A PID controller is simple three-term controller. The letter P,I and D stand for P- Proportional, I- Integral, DDerivative. The transfer function of the most basic form of PID controller,is

$$C(S) = K_P + \frac{K_I}{S} + K_D S \quad (2.3)$$

$$C(S) = \frac{K_D S^2 + K_P S + K_I}{S} \quad (2.4)$$

Where

KP = Proportional gain,

KI = Integral gain

KD= Derivative gain.

The control  $u$  from the controller to the plant is equal to the Proportional gain (KP) times the magnitude of the error plus the Integral gain (Ki) times the integral of the error plus the Derivative gain (Kd) times the derivative of the error.

$$u = K_p e + K_i \int e dt + K_d \frac{de}{dt} \quad (2.5)$$

Due to its simplicity and excellent if not optimal performance in many applications, PID controllers are used in more than 95% of closed-loop industrial processes. We are most interested in four major characteristics of the closed-loop step response.

They are

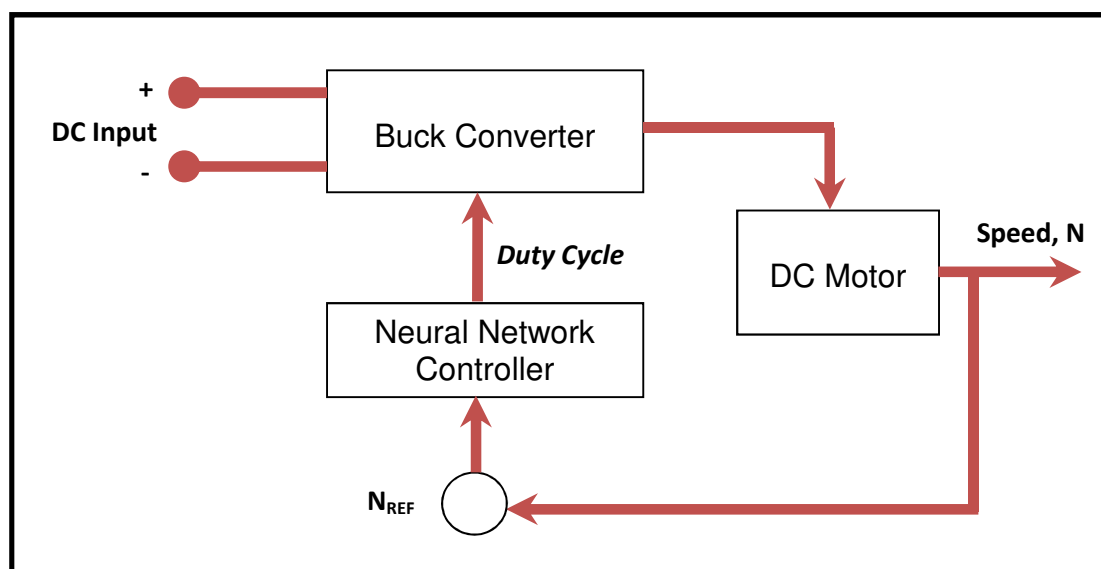
- a. Rise Time : the time it takes for the plant output  $Y$  to rise beyond 90% of the desired level for the first time.
- b. Overshoot : how much the peak level is higher than the steady state, normalized against the steady state.
- c. Settling Time : the time it takes for the system to converge to its steady state.
- d. Steady-state Error : the difference between the steady-state output and the desired output.

## CHAPTER 3

### METHODOLOGY

#### 3.1 Project Design

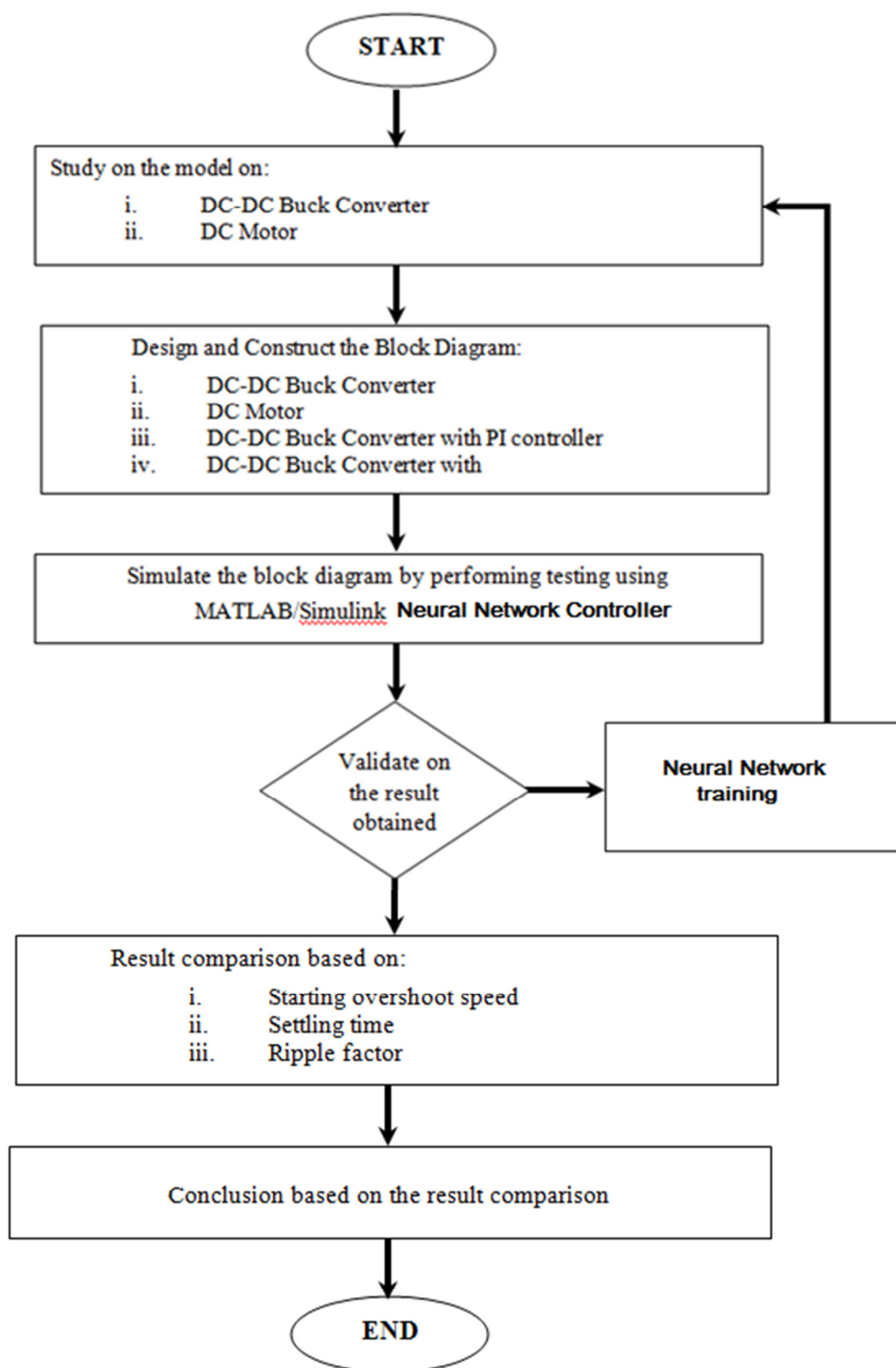
The general proposed block diagram for this is shown in Figure 3.1 and Figure 3.2 shows the overview of the project in flow chart.



**FIGURE 3.1:** Block Diagram For Propose DC-DC Buck Converter Using Neural Network Controller



The proposed design is using Neural Network Controller as a controller to controller speed to the desired speed. The performance of the controller is compared with the designed conventional controller which is Proportional-Integrated controller, PI. The NN controller used to control the duty cycle of the converter to ensure that the motor with run as same speed as the reference speed. The model is design and simulate by MATLAB/Simulink.

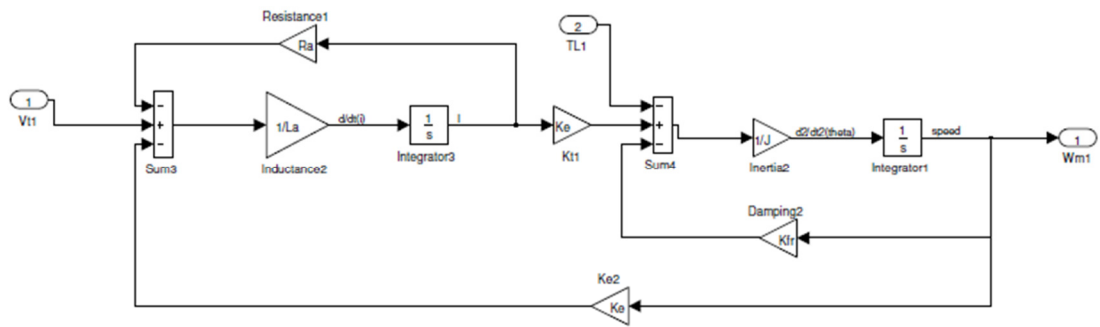


**FIGURE 3.2:** The methodology Flow Chart

### 3.2 Proposed DC Motor

There were several papers such as Zulkifilie Ibrahim and Emil Levi (2002) and Teresa Orłowska-Kowalska (2001) had proposed the DC motor design using MATLAB/Simulink. The design DC motor is used to analyse of the system dynamics. But for this project, I have made the design made by Zulkifilie Ibrahim and Emil Levi as my reference due to the application of the motor which is quite the same.

The dynamic model of the system is formed and Matlab Simulink blocks shown in Figure 3.1



**FIGURE 3.3:** Matlab model of DC motor

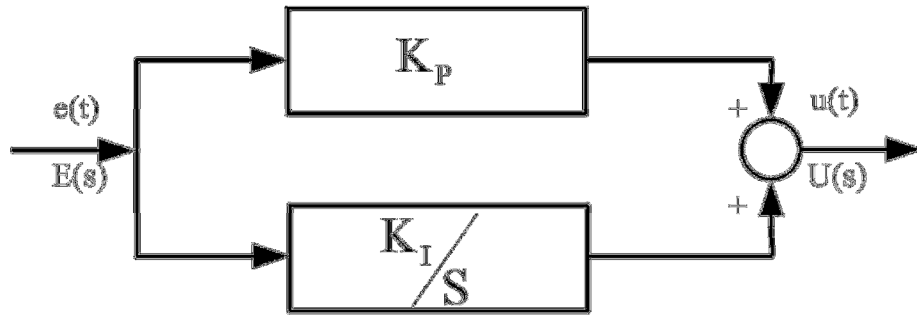
The input voltage of the motor is connected to the Buck converter to ensure that the motor runs at the desired speed control by the controller. The motor is connected to the fixed load. While the output of the motor is the desired speed, in rad/s. Due to the desired speed is in r.p.m, below equation is used to convert the speed in rad/s to speed, N in r.p.m.

$$N = \frac{60T}{2\pi} \quad (3.1)$$

The proposed project is to control the motor speed to 1600 rpm. The value of motor parameter is based on the Muhammad H.Rashid (2004) and the P.Thepsatorn, A.Numsomran (2006) proposed parameter.

### 3.3 Proposed Proportional Integral Controller (PI)

Proportional-Integral (PI) controllers [6, 14] are one of the most applicable controllers in different industries. The main important need in application of these controllers is their parameters tuning in order to gain desired result. So an accessible method with high accuracy and speed has to be used for determination of these control parameters ( $K_p$ ,  $K_i$ ). The control architecture used for PI controller is shown by Figure 3.2



**FIGURE 3.4:** Block diagram of PI controller

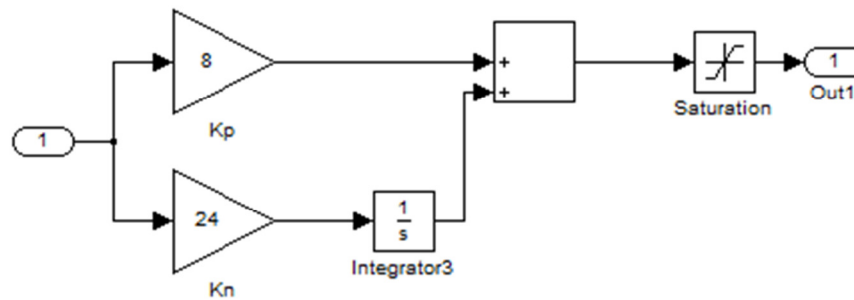
PI-controllers have been applied to control almost any process one could think of, from aerospace to motion control, from slow to fast systems. With changes in system dynamics and variation in operating points PI-controllers should be retuned on a regular basis.

$$U(t) = K_p \left[ e(t) + \frac{1}{T_i} \int_0^t e(\tau) d(\tau) \right] \quad (3.2)$$

Where  $u(t)$  and  $e(t)$  denote the control and the error signals respectively, and  $K_p$  and  $T_i$  are the parameters to be tuned. The corresponding transfer function is given as,

$$K(s) = K_p \left[ 1 + \frac{1}{T_i(s)} \right] \quad (3.3)$$

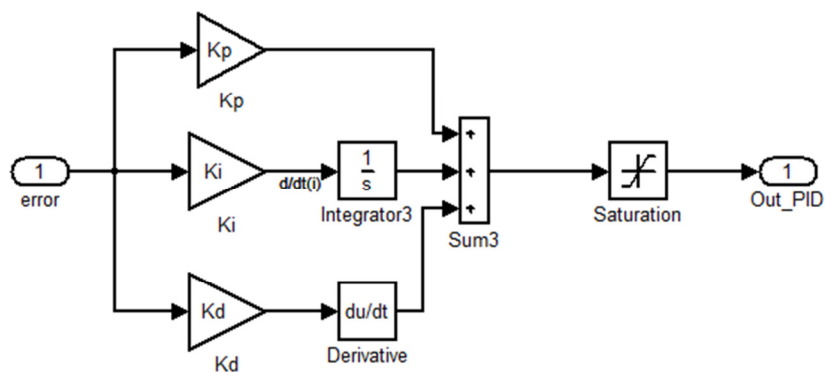
The dynamic model of the system is formed and Matlab Simulink blocks shown in Figure 3.2



**FIGURE 3.5:** Matlab Simulink block

### 3.4 Proposed PID controller

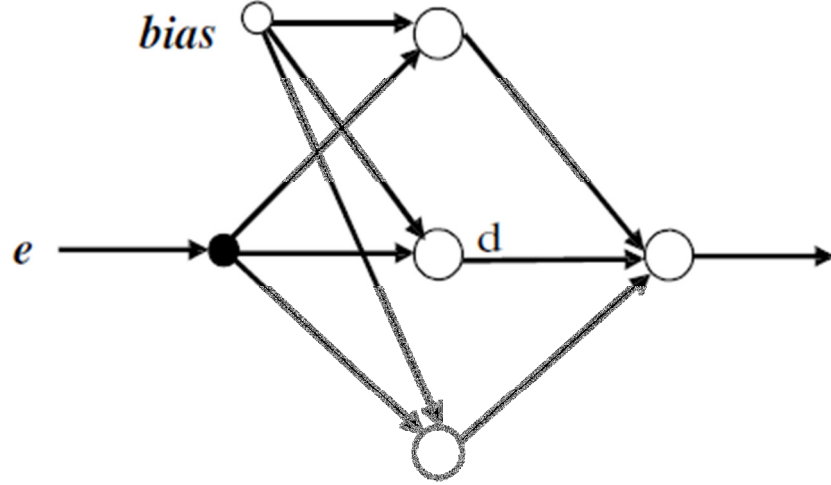
The design of the PI controller is as below where the value for gain was refers to the model proposed Muhammad H. Rashid (2004). The PID will only produce an output with the value range between 0 and 1. This is because the duty cycle range is from 0 to 1



**FIGURE 3.6:** The model for PID controller in MATLAB/Simulink

### 3.5 Architecture of Neural Network Controller

Figure of input, hidden and output layer:



**FIGURE 3.7:** Architecture of Neural Network

### 3.6 Proposed Neural Network Controller

The forward calculations between input and hidden layer is

$$y_i = \sum_{n=1}^k W_{ni} \cdot X_n + b_i \quad (3.4)$$

Where  $y_i$  is the output between input and hidden layer neuron, while  $W_{ni}$  is the connection weight from  $n$ -th neuron to  $i$ -th neuron and  $k$  is the number of previous layer neurons. Then, the activation is transfer to next layer neurons through a nonlinear transfer function as follow

$$Y_i = f(y_i)Y_i \quad (3.5)$$

The nonlinear transfer function used in this project is the following tan-sigmoidal function

$$f(y_i) = \frac{2}{(1+e^{-2y_i})-1} \quad (3.6)$$

Output layer is

$$z_i = \sum_{i=1}^k Y(y_i) \cdot W_i \quad (3.7)$$

Where  $Y(y_i)$  is the output of previous layer neuron, while  $w_i$  is the connection weight from  $i$ -th neuron and  $k$  is the number of previous layer neurons.

The activation function of the output layer is

$$Z_i = f(z_i) \quad (3.8)$$

Where,

$$f(z_i) = \frac{2}{(1+e^{-2z_i})-1} \quad (3.9)$$

The nonlinear transfer functions acts as the important rule in giving ANN nonlinear mapping property. ANNs have capability to obtain such as pattern recognition and any nonlinear function imitation by adjusting the connection weights appropriately.

The process is called the learning and error back propagation algorithm (BP algorithm) is generally use for the learning. BP algorithm is based on gradient descent method, i.e. connection weights are iteratively updated so the mean squared error **EP** to an input pattern **p** can decrease.

The updating weight parameters between input and hidden layer is

$$W_{ni}[n+1] = W_{ni} + \Delta W_{ni} \quad (3.10)$$

$$\Delta W_{ni} = \eta \times \frac{\delta e}{\delta W_{ni}} \quad (3.11)$$

$$\frac{\delta e}{\delta W_{ni}} = \frac{\delta y_i}{\delta W_{ni}} \times \frac{\delta Y y_i}{\delta y_i} \times \frac{\delta z_i}{\delta Y(y_i)} \times \frac{\delta z_i}{\delta z_i} \times \frac{\delta e}{\delta z_i} \quad (3.12)$$

$$\frac{\delta e}{\delta W_{ni}} = X_n \times Y_i [1 - (Y_i)^2] \times W_i \times (Z_i - Z_n) \quad (3.13)$$

$$W_{ni}[n + 1] = W_{ni} + \eta X_n Y_i (1 - Y_i^2) (W_i (Z_i - Z_n)) \quad (3.14)$$

Where  $\eta$  represent the learning rate. The updating weight parameters between hidden layer and output layer is

$$W_i[n + 1] = W_i + \Delta W_i \quad (3.15)$$

$$\Delta W_i = \eta \left( \frac{\delta e}{\delta W_i} \right) \quad (3.16)$$

$$\frac{\delta e}{\delta W_i} = \frac{\delta z_i}{\delta z_i} \cdot \frac{\delta z_i}{\delta z_i} \cdot \frac{\delta e}{\delta z_i} \quad (3.17)$$

$$\frac{\delta e}{\delta W_i} = Y_i (1 - Y_i) \cdot (Z_i - Z_n) \quad (3.18)$$

$$W_i[n + 1] = W_i + (\eta Y_i (1 - Y_i)) (Z_i - Z_n) \quad (3.19)$$

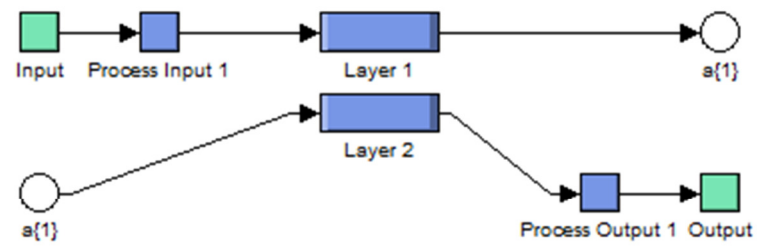
Where  $\eta$  represent the learning rate. The mean square error  $E_p$  is calculated as

$$E_p = \frac{1}{2} \sum_{h=1}^m (Z_i - Z_n) \quad (3.20)$$

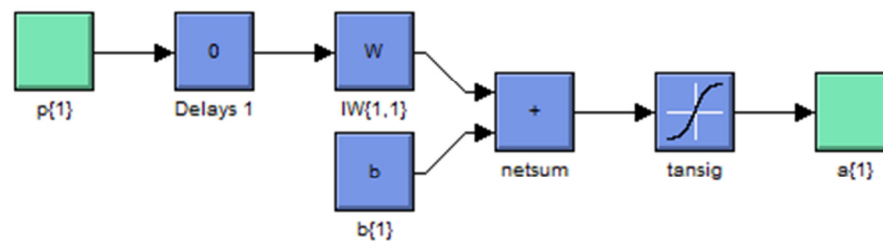
Where  $Z_i$  is the desired output of the h-th neuron in output layer and  $Z_n$  is the corresponding actual output, respectively and m is the number of the output layer neurons.

The neural network of the system is formed and Matlab Simulink blocks shown in Figure 3.6

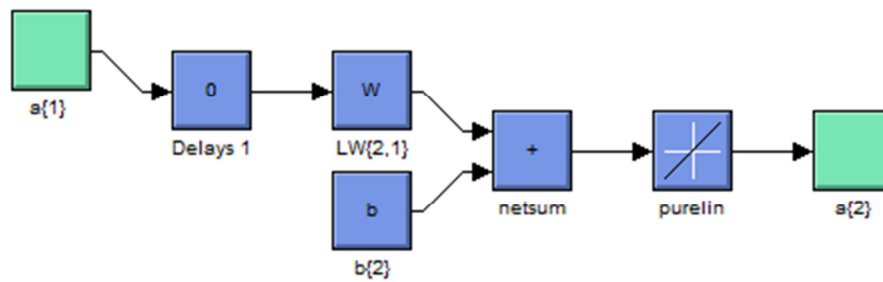




**FIGURE 3.8:** NN block



**FIGURE 3.9:** Layer 1 block



**FIGURE 3.10:** Layer 2 block

## REFERENCES

- [1] A. Giovanni Beccuti, Georgios Papafotiou and Manfred Morari, (2005) "*Optimal control of the Boost dc-dc converter*". IEEE conference on decision and control
- [2] Liu Qingfeng, Leng Zhaoxia, Sun Jinkun, Wang Huamin, (2011) "*A composite PWM control strategy for Boost converter*". 2012 International conference.
- [3] Fathi Shaban Jaber (2011) "*Development of a Buck-Boost Converter Using Fuzzy Logic Control*". Degree of Master Thesis, UTHM
- [4] Siti Hasmah Binti Jamali (2011) "*Voltage Tracking of DC-DC Boost Converter Using Fuzzy Logic Controller*", Master Thesis UTHM
- [5] Zulkiflilie Ibrahim and Emil Levi (2002) "*A Comparative Analysis of Fuzzy Logic and PI Speed Control in High-Performance AC Drives Using Experimental Approach*", IEEE Transaction on Industry Application.
- [6] B. Heber, L.Xu and Y.Tang "*Fuzzy Logic enhanced speed control of an indirect field oriented induction motor drive*", IEEE Trans. Power Electron, vol.12, pp. 772-778, Sept 1997
- [7] P.Thepsatorn, A. Numsomran, V. Tipsuwanporn, T.Teanthong (2006)"*DC Motor Speed Control using Fuzzy Logic based on LabVIEW*". SICE-ICASE International Joint Conference 2006.
- [8] Ramesh Chandra Chourasia, Mukesh Kumar (2013)", *Speed Control of S.E.D.C Motor by Using PI and Fuzzy Logic Controller*. International Journal of Soft Computing and Engineering (IJSCE)
- [9] H. Feshki Farahani (2011)"*Designing and Implementation of a Fuzzy Controller for DC-DC Converters and Comparing With PI Digital Controller*". Islamic Azad University, Iran formerly [4]
- [10] Muhammad H. Rashid (2004):"*Power electronics Circuits, Devices and Applications*" Electrical computer Engineering University of west Florida.
- [11] Sinan Pravadalioglu (2005), "*Single-chip fuzzy logic controller design and an application on a permanent magnet dc motor*". Engineering Application of Artificial Intelligence formerly [14]
- [12] Yigang Shi and P.C Sen (2000) "*A New Defuzzification Method for Fuzzy Control of Power Converters*. IEEE members.
- [13] Jan Jantzen (1998),"*Design of Fuzzy Controllers*". Technical University of Denmark formerly [15]

- [14] Cetin Elmas, Omer Deperlioglu, Hassan Huseyin Sayan (2007),” *Adaptive fuzzy logic controller for DC-DC converters*”. Expert system with application. Formerly [11]
- [15] Wing-Chi So, Chi K.Tse, Yim-Shu Lee (1996), “*Development of a Fuzzy Logic Controller for DC/DC Converter : Design, Computer Simulation and Experimental Evaluation*”. IEEE transaction on Power electronics formerly [12]
- [16] K. Viswanathan, D. Srinivisan, R.Oruganti (2002),”*A Universal Fuzzy Controller for A Non-Linear Power Electronic Converter*”. National University of Singapore. Formerly [13]
- [17] Teresa Orlowska-Kowalska, Krzysztof Szabat, Krzysztof Jaszcak (2001) “*The Influence of parameter and structure of PI-Type fuzzy-logic controller on DC drive system dynamics*”, Fuzzy set and systems
- [18] C.K. Tse and K. M Adams (1992), “*Quasilinear modelling and control of DC/DC Converters*,” IEEE Trans. Power Electron
- [19] V.S. C Raviraj and P.C. Sen, “*Comparative Study of Proportional-Integral, Sliding Mode and Fuzzy Logic Controllers for Power Converters*”. IEEE IAS Proceeding.
- [20] K. Somsai, A. Oonsivilai, A. Srikaew & T. Kulworawanichpong,(2007),”*Optimal PI Controller Design andSimulation of a static VAR Compensator Using MATLAB/SIMULINK*”. International Conference on Power System.
- [21] P.C Sen, (1990) “*Electric Motor Drives and control – Past, Present, and future*”. IEEE Trans on Industrial Electronics, Vol37, No 6, formerly [6]
- [22] Robert Babuska and Stefano Stramigiolo (1999)” *Matlab and Simulink for Modeling and Control*”, Delft University of Technology formerly [8]
- [23] PID Control Tuning: A Short Tutorial, Retrieved June 22, 2013 from [http://wwwdsu.uqac.ca/~rbeguena/Systemes\\_Asservis/PID.pdf](http://wwwdsu.uqac.ca/~rbeguena/Systemes_Asservis/PID.pdf)