ISSN: 2277-9655
Impact Factor: 1.852

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

# Comparative Analysis between Digital PWM and PI with Fuzzy Logic Controller for the Speed Control of BLDC Motor

**Ruchita Patel[*1],Hemant Amhia[2]**
Department of Electrical Engineering, Jabalpur Engineering College, Jabalpur, India
patelruchi.12ex@gmail.com

## Abstract

Residential and commercial appliances such as refrigerators and air conditioning systems have been using conventional motor drive technology. The machines in these applications typically have low efficiencies and high maintenance. Development of advanced motor drives has yielded increases in efficiency and reliability. A Brushless DC (BLDC) drives are known for higher efficiency, lower maintenance and higher cost. In this paper presents a proposed method for speed control of BLDC motor using Digital PWM and PI with Fuzzy logic controller. This digital control treats BLDC motor as a digital system and regulates speed with the help of two predefined state variables techniques, which makes the concept of controller extremely simple for design and implementation. The main disadvantage of DC motor is sparking problem in brush but if using BLDC motor solve this problem. PI with Fuzzy is very nice and good concept for speed control of motor because this concept is combination of convention and modern technique. The key advantage of Fuzzy logic controller is switching is possible at different stage, and key advantage is change the value of P and I and regulate the speed of motor. The main advantage of Digital PWM is control the speed without change the voltage and current that's why only change the width of plush and reduce the losses related to current. If we apply the conventional methods generally some resistance are connected in series of supply that case change the value of resistance and regulate the speed of motor. But using this methods losses is increase but if we used the Digital PWM and PI with Fuzzy logic controller and find out which one is best for speed controller of BLDC motor.

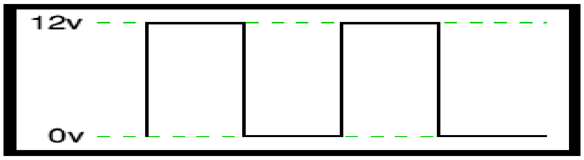
**Keywords**: Bldcm, digital pwm, pi controller, fuzzy logic controller.

## Introduction

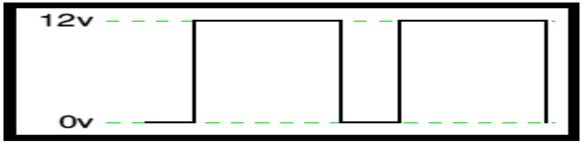
Typically, machines found in these appliances are single-phase induction motors or brushed dc machines which are characterized by low efficiency and high maintenance, respectively [6]. Replacing these inefficient motors with more efficient brushless dc (BLDC) motors will result in substantial energy savings. Proportional-integral (PI) control with hysteresis or pulse width modulation (PWM) switching is the most widely used speed control technique for BLDC motors with trapezoidal back EMF. It can be easily implemented on analog or digital components because it is well understood, simple, and in practice for a fairly long period of time. To enhance the performance or to reduce the cost has been focus of development work for a long time. Cost and implementation complexity are often the most important factors for design trade-offs between techniques, implementation, and strategy of motor control hardware. More than often, microcontrollers, microprocessors, and digital signal processors (DSPs) are chosen to digitally implement the techniques/controls. Digitizing analog controllers serves to add complexity to the overall design procedure. Moreover, digital implementation of a continuous control technique does not produce a digital controller.

## Digital PWM Control Scheme

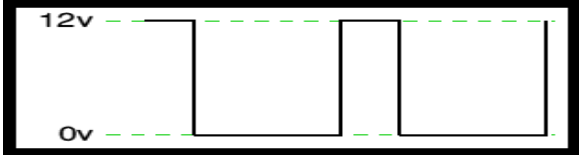
**Pulse Width Modulation** (PWM) is the most effective means to achieve constant voltage battery charging by switching the solar system controller's power devices. When in PWM regulation, the current from the solar array tapers according to the battery's condition and recharging needs Consider a waveform such as this: it is a voltage switching between 0v and 12v. It is fairly obvious that, since the voltage is at 12v for exactly as long as it is at 0v, then a 'suitable device' connected to its output will see the average voltage and think it is being fed 6v - exactly half of 12v. So by varying the width of the positive pulse - we can vary the 'average' voltage.



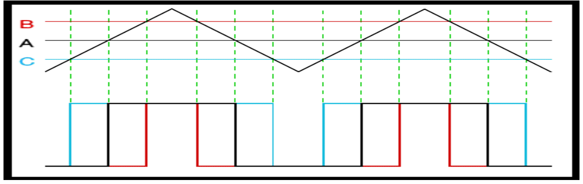
Similarly, if the switches keep the voltage at 12 for 3 times as long as at 0v, the average will be 3/4 of 12v - or 9v, as shown below.



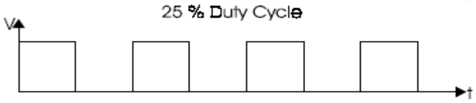
And if the output pulse of 12v lasts only 25% of the overall time, then the average is



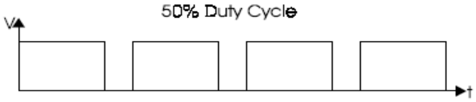
By varying - or 'modulating' - the time that the output is at 12v (i.e. the width of the positive pulse) we can alter the average voltage. So we are doing 'pulse width modulation'. I said earlier that the output had to feed 'a suitable device'. A radio would not work from this: the radio would see 12v then 0v, and would probably not work properly. However a device such as a motor will respond to the average, so PWM is a natural for motor control. A more efficient technique employs **pulse width modulation** (PWM) to produce the constant current through the coil. A PWM signal is not constant. Rather, the signal is on for part of its period, and off for the rest. The **duty cycle**, D, refers to the percentage of the period for which the signal is on. The duty cycle can be anywhere from 0, the signal is always off, to 1, where the signal is constantly on. A 50% D results in a perfect square wave.



25 % Duty Cycle

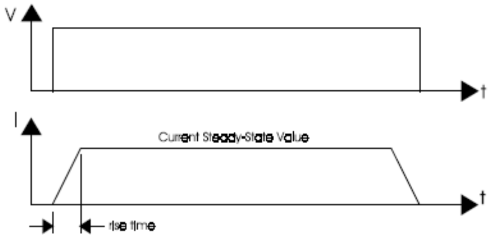


50% Duty Cycle



75% Duty Cycle

A solenoid is a length of wire wound in a coil. Because of this configuration, the solenoid has, in addition to its resistance, R, a certain **inductance**, L. When a voltage, V, is applied across an inductive element, the current, I, produced in that element does not jump up to its constant value, but gradually rises to its maximum over a period of time called the **rise time** (Figure 2). Conversely, I does not disappear instantaneously, even if V is removed abruptly, but decreases back to zero in the same amount of time as the rise time.



Therefore, when a low frequency PWM voltage is applied across a solenoid, the current through it will be increasing and decreasing as V turns on and off. If D is shorter than the rise time, I will never achieve its maximum value, and will be discontinuous since it will go back to zero during V's off period (Figure 3).* In contrast, if D is larger than the rise time, I will never fall back to zero, so it will be continuous, and have a DC average value. The current will not be constant, however, but will have a ripple (Figure 4).

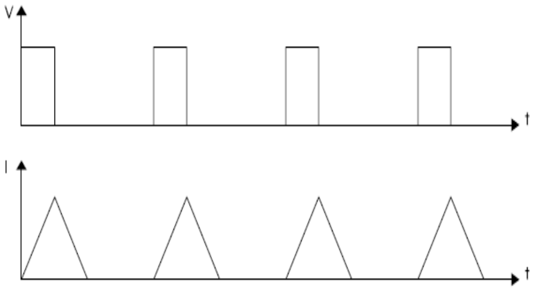
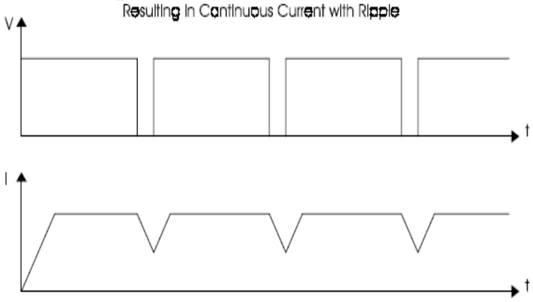
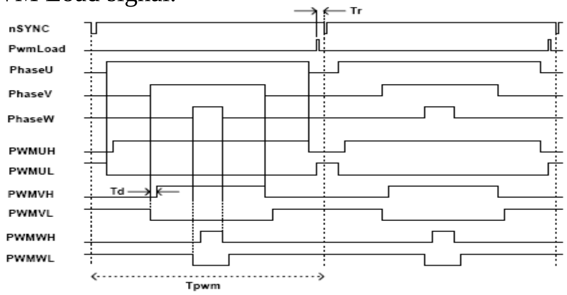


Figure 4 - Low Frequency PWM with D>rise time
Resulting in Continuous Current with Ripple



### PWM Controller Features

This controller offers a basic “Hi Speed” and “Low Speed” setting and has the option to use a “Progressive” increase between Low and Hi speed. Low Speed is set with a trim pot inside the controller box. Normally when installing the controller, this speed will be set depending on the minimum speed/load needed for the motor. Normally the controller keeps the motor at this Lo Speed except when Progressive is used and when Hi Speed is commanded (see below). Low Speed can vary anywhere from 0% PWM to 100%. **PWM Operation.** Upon receiving the modulation index commands (UAlpha and UBeta) the sub-module SVPW M_Tm starts its calculations at the rising edge of the PWM Load signal.



### Controller Design for DPWM

The value of the duty ratio D can be obtained from the electrical and mechanical equations [1]. The value of D can be expressed as a function of the motor parameters.

$$T_{em} = J\frac{dw}{dt} + Bw + Tl \quad \ldots\ldots\ldots\ldots..1$$

$$T_{em} = \mathrm{Ktl} \quad \ldots\ldots\ldots\ldots\ldots..2$$

Where *Te*, w(t), *B* , *J* and *TL* denote developed electromagnetic torque, rotor angular velocity, viscous friction rotor moment of inertia and load torque respectively. Equate (1) and (2), we get J

$$\mathrm{Ktl} = J\frac{dw}{dt} + Bw + Tl \ldots\ldots\ldots.3$$

where *Kt* = torque constant and *I* = average current. At steady state, (3) can be written in terms of steady-state angular velocity Wss as

$$I_{wss} = \frac{1}{kt} + (Bwss + Tl) \ldots\ldots\ldots\ldots\ldots\ldots4$$

At steady state angular velocity *Wss* , phase voltage *Van* can be expressed in terms of phase current I, winding resistance and velocity constant *ke*, ie given by

Van = Ira+KeWss …………………..5

The phase voltage in terms of dc-link voltage *V*dc and duty ratio *D* is

Van = DVdc ………………………...6

Substituting the value of the steady-state current from (4) and phase voltage from (6) in (5),we get the value of duty ratio

$$D = \frac{1}{V_{dc}}\left[\frac{(Tl + Bwss)}{Kt} + Kewss\right]\ldots\ldots\ldots\ldots7$$

By considering *WL* and *WH* we can get the *DL* and *DH* respectively. The maximum deviation from the reference speed (*W**) due to the application of high duty *DH* is denoted by *ΔωH*, and the maximum deviation from the reference speed due to the application of a low duty *DL* is denoted by *ΔωL*. The speed response can be expressed as

$$w(t) = \frac{Tem - Tl}{B} + \left[W - \left(\frac{Tem - Tl}{B}\right)\right]e^{\left(\frac{-1}{B}\right)t} \ldots\ldots8$$

### Controller Design for PI with Fuzzy Controller

The PI control problem has to be converted form a theoretical continuous process into a real "discrete" system running on a microcontroller. What this mean in practice is that the measuring of the set point and motor speed and the calculation of the output is only performed a regular interval. In the context of a microcontroller, this is might correspond to some code run from a timer interrupt.

The PI controller can thus be expressed as: Output = Proportional Gain*(error_speed) + Integral Gain*S (previous_error_speed_) and **Final output** = [{(Output) or Proportional Gain*(error_speed) + Integral Gain*S (previous_error_speed_)} – (last_error_speed)]

### PI Error Calculation

The PI controller compares the set point (SP) to the process variable (PV) or mean variable (MV) to obtain the error e, as follows:

$$e = SP - PV \qquad 9$$

Then the PI controller calculated the control action, u (t), as follows. In this equation, Kp is the process gain.

$$u = K_p e + \frac{1}{\tau_I}\int e \, d\tau \qquad 10$$

Where $\tau_I$ = "Integration time"

The above following formula represents the proportional gain.

$$Up\ (t) = Kp\ (e) \qquad 11$$

### Implementing the PI Algorithm with the PI Functions

This section describes how the PI control toolbox function implements the PI algorithm. The PI algorithm used in the PI control toolbox

ISSN: 2277-9655
Impact Factor: 1.852

Error Calculation

The following formula represents the current error used in calculating proportional, integral, where PV is the filtered process variable.\

$$e(k) = SP - PV \qquad 12$$

**Proportional Action**

Proportional action is the controller gains times the error, as show the following formula:

$$Up(k) = Kp * e(k) \qquad 13$$

**Trapezoidal Integration**

Trapezoidal Integration is used to avoid sharp changes in integral action when there is a sudden change in the PV or SV. Use nonlinear adjustment of the integral action to counteract overshoot The following formula represents the trapezoidal integration action.

$$Ui(k) = Kp/Ti\Sigma \{[e(i) + e(i-1)]/2\}\ \Delta t \qquad 14$$

Where i = 1, 2, 3 …k

**Controlled Output**

Controller output is the summations of the Proportional, and integral action, as show in following formula:

$$U(k) = Up(k) + Ui(k) \qquad 15$$

**Output Limit**

The actual controlled output is limited to the range specified for control output as follows:

If U (k) ≥ Umax then U (k) = Umax

And

If U (k) ≤ Umin then U (k) = Umin

The following formula shown the practical model of PI controller

$$U(t) = Kp\left[(SP-PV) + \frac{1}{T_i}\int_0^t (SP - PV)\, dt\right]$$

The PI function uses an integral sum correction algorithm that facilitates anti-windup & bumpless manual-to-automatic transfers. Windup occurs at the upper limit of the controller output, for example, 100% when the error (e) decreases the controlled output is decreases, moving out of the windup area. The integral sum correction algorithm prevents abrupt controller parameters. The default range for the SP, PV and output parameter corresponds to percentage value; adjust the corresponding range accordingly.

**Error Calculation**

The current error used in calculating integral action for the precise PI algorithm is shown the following formula:

$$e(k) = (SP-PV_f)(L+(1-L)*\frac{|SP - PV_f|}{SP_{range}}) \qquad 16$$

Where SP range is the range of the SP and L is the linearity factor that produces a nonlinear gain term in which the controller gain increase with the magnitude of the error. If L is 1, the controller is linear. A valve of 0.1 makes the minimum gain of the controller 10% Kp. Use of a nonlinear gain term is referred to as a precise PI algorithm. Results shown in figure no. (2), and (3),

## Design of the Fuzzy Logic Control Scheme

Fuzzy controllers have got a lot of advantages compared to the classical controllers such as the simplicity of control, low cost and the possibility to design without knowing the exact mathematical model of the process. Fuzzy logic is one of the successful applications of fuzzy set in which the variables are linguistic rather than the numeric variables. Linguistic variables, defined as variables whose values are sentences in a natural language (such as large or small), may be represented by fuzzy sets. Fuzzy set is an extension of a 'crisp' set where an element can only belong to a set (full membership) or not belong at all (no membership). Fuzzy sets allow partial membership, which means that an element may partially belong to more than one set. A fuzzy set A of a universe of discourse X is represented by a collection of ordered pairs of generic element x ∈ X and its membership function μ : X → [ 0 1], which associates a number μ A(x): X → [ 0 1], to each element x of X.

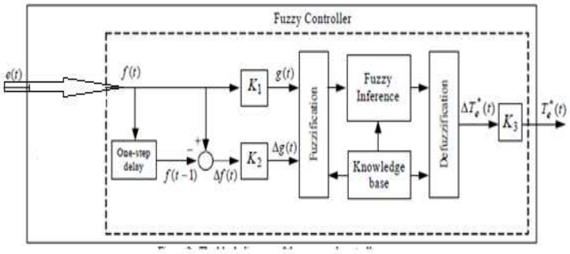


**Fig. Block Diagram of PI with Fuzzy Logic controller**

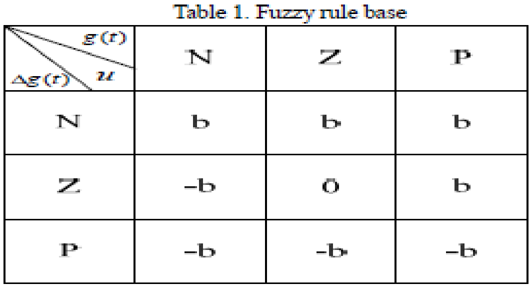


Table 1. Fuzzy rule base

| g(r) / Δg(r) / u | N | Z | P |
|---|---|---|---|
| N | b | b | b |
| Z | -b | 0 | b |
| P | -b | -b | -b |

**Fig. Rule Base for Fuzzy logic controller**

ISSN: 2277-9655
Impact Factor: 1.852

$$\mu_N(x) = \begin{cases} 1 & x \le -b \\ \frac{-x}{b} & -b < x \le 0 \\ 0 & \text{otherwise} \end{cases}$$

$$\mu_Z(x) = \begin{cases} \frac{x+b}{b} & -b < x \le 0 \\ \frac{b-x}{b} & 0 < x \le b \\ 0 & \text{otherwise} \end{cases}$$

$$\mu_P(x) = \begin{cases} 1 & b \le x \\ \frac{x}{b} & 0 < x \le b \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

The fuzzy inference engine, based on the input fuzzy sets in combination with the expert's experience, uses adequate IF-THEN rules in the knowledge base to make decisions and produces an implied output fuzzy set $u$ . For this particular application, the proposed IF-THEN fuzzy rule base is shown in Table 1 and is described as follows:

i. If $\Delta g(t) \in \mathrm{N}$, then $u(g(t), \Delta g(t)) = b$ .

ii. If $\Delta g(t) \in \mathrm{P}$, then $u(g(t), \Delta g(t)) = -b$ .

iii. If $\Delta g(t) \in$ Z and $g(t) \in$ N, then $u(g(t), \Delta g(t)) = -b$ ..

iv. If $\Delta g(t) \in$ Z and $g(t) \in$ P, then $u(g(t), \Delta g(t)) = b$ .

v. If $\Delta g(t) \in$ Z and $g(t) \in$ Z, then $u(g(t), \Delta g(t)) = 0$ .

## Simulink Results

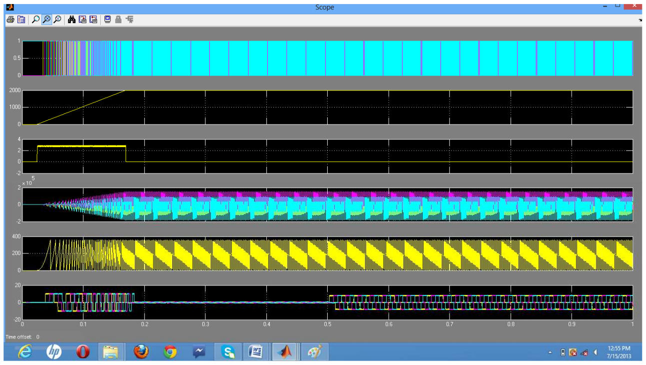


Fig Speed controller of BLDC motor using DPWM

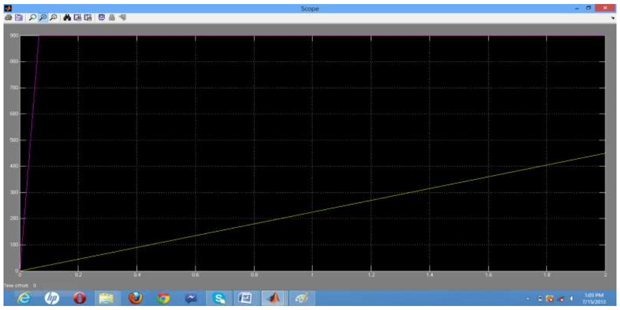


Fig Speed Controller using PI with FLC

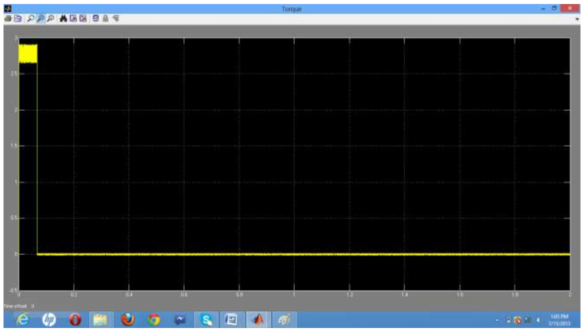


Fig Torque of BLDC Controller using PI with FLC

## Conclusions

In this paper compare the results between DPWM and PI with FLC of BLDC motor. If we apply the DPWM and change the width of freq. And regulate the speed. Another method is PI with FLC in this method Change the parameter and regulate the speed, and FLC for switching purpose. Check out the results and decided PI with FLC is motor better as compare to the DPWM .

## Reference

*[1]* A.Sathyan, N. Milivojevic, Y.-J. Lee, M. Krishnamurthy, and A.Emadi "An FPGA-based novel digital PWM control scheme for BLDC motor drives," *IEEE Trans. Ind. Electron., vol. 56, no. 8, pp. 3040*– Aug. 2009**.**

*[2]* A. A. Aboulnaga, P. C. Desai, F. Rodriguez, T. R. Cooke, and A. Emadi, "A novel low-cost high-performance single-phase adjustable-speed motor drive using PM brushless DC machines: IIT's design for 2003 future energy challenge," in *Proc. 19 th Annu. IEEE Appl. Power Electron. Conf.*, Anaheim, CA, Feb. 2004, pp. 1595–1603.

*[3]* F. Rodriguez and A. Emadi , "A novel digital control technique for brushless DC motor drives, " *IEEE Trans. Ind. Electron., vol. 54, no. 5*, pp. 2365–2373, Oct. 2007.

*[4]* F. Rodriguez, P. Desai, and A. Emadi, "A novel digital control technique for trapezoidal

brushless DC motor drives ," in *Proc. Power Electron. Technol. Conf., Chicago, IL, Nov. 2004.*

*[5]* F. Rodriguez and A. Emadi, "A novel digital control technique for brushless DCmotor drives: Conduction-angle control," in *Proc. IEEE Int.Elect. Mach. Drives Conf., May 2005, pp. 308–314.*

*[6]* A. Emadi, *Handbook of Automotive Power Electronics and Motor Drives*. Boca Raton, FL: CRC Press, 2005.

*[7]* A. A. Aboulnaga, P. C. Desai, F. Rodriguez, T. R. Cooke, and A. Emadi, "A novel low-cost high-performance single-phase adjustable-speed motor drive using PM brushless DC machines: IIT's design for 2003 future energy challenge ," in *Proc. 19th Annu.IEEE Appl. Power Electron. Conf* .Anaheim, CA, Feb. 2004, pp. 1595–1603.