FPGA BASED ON ADAPTIVE FUZZY AND SPACE VECTOR PULSE WIDTH MODULATION TO CONTROL SPEED OF PMSM

Nguyen Vu Quynh[1], Tran Hanh[2], Trinh Tran Thanh Tam[3]

[1] Electrical – Mechanical Department, Lac Hong University

[2,3] Informatics Technology, Lac Hong University

Abstract: This paper presented speed control algorithm of Permanent Magnet Synchronous Motor (PMSM) basing on adaptive fuzzy combined with space vector pulse width modulation (SVPWM). PMSMs were applied so much in manufacturing industry such as metal cutting machines, packaging machines, precision machining; so the controller of motor plays a very important role. The simulation architecture of system was implemented on Simulink of Matlab. The article uses very high speed integrated circuit hardware description language (VHDL) as the hardware simulation language, and embeds programs written in VHDL code to Simulink/Matlab to calculate. Based on parameters of PMSM, the article was calculated with three different load cases: normal-load, light-load, heavy-load as below:

- Jm = 0.000108 Kgm2, F = 0.0013 Nms
- Jm = 0.000108/4 Kgm2, F = 0.0013/4 Nms
- Jm = 0.000108*3 Kgm2, F = 0.0013*3 Nms

(Jm: Combined inertia of rotor and load; F: Combined viscous friction of rotor and load)

After successful simulation, the system was tested again by experiment on FPGA kit. Through simulation and experiment results shown that the speed of PMSM still was stable in case of large load changed. Switching frequency of the modulator could be reached from 150 Hz to 50 kHz. Concurrently, high switching frequency helps to reduce noise generated in the system. Controller integrated in an IC saves space and avoids the influence of external factors preferred as noise or temperature

Keyword: FPGA, Fuzzy controller, VHDL, manufacturing, simulate programming

1 Introduction

The use of PMSM in the industry increased strongly because of the advantages such as high speed and precision. The PMSMs were commonly used in systems requiring high precision such as robot, mechanical processing… But in industrial applications, there are many uncertainties, for example noise, external load, fiction force ... They affect to the performance quality of the system. To cope with those problems, many intelligent control techniques such as fuzzy, neural network, genetic algorithm have been developed. They helped to control exactly speed and position of motor and to increase productivity and quality of products. But they used Digital Signal Processor (DSP) to process in almost studied. Unfortunately, DSP suffers from a long period of development and exhausts many resources of the system. The cost of systems was not cheap. Altera FPGA supplied a solution for those issues. Especially, FPGA has programmable hard-wired feature, fast calculation ability, low power consumption and higher density. Moreover, FPGA is better for the implementation of the digital system than DSP. Besides, the application of FPGA technology in adaptive fuzzy speed control had not been studied in our country. The article applied adaptive fuzzy algorithm and SVPWM based on VHDL code to design controller for PMSM motors. The contents of article includes, firstly, the mathematical model of PMSM, SVPWM algorithm, vector control method and adaptive fuzzy controller were derived and designed in speed control of PMSM. Secondly, the VHDL was used to program for all systems, those were current controller; adaptive fuzzy controller; SVPWM; PI controller, Clark, Park, Clark^{-1} and Park^{-1} transform block. Finally, the article presented simulation on Matlab/Simulink and experiment on FPGA kit results.

In Fig.1, the three phase stator currents (i_a, i_b) and speed of the motor (ω_r) were feed-backed to calculate the generated voltage vector for input of SVPWM control module. SVPWM module generated the electric current to control three phase inverter for providing power to the motor. Fuzzy controller was used to control speed of motor. Adjust mechanism (adaptive) helped to keep stable speed for motor when load was changed.

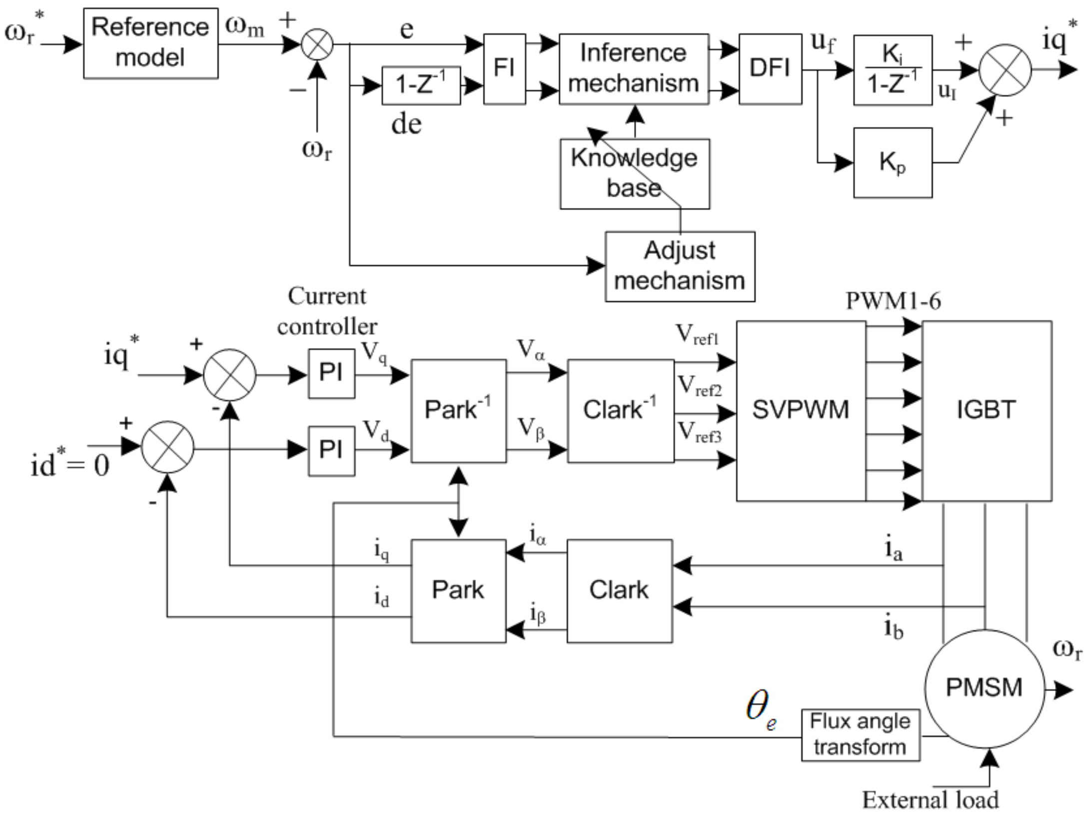


Fig. 1. The structure of whole system

2 PMSM Drive and Speed Controller Design

2.1 Mathematical Model of PMSM[6]

$$\frac{di_d}{dt} = \frac{1}{L_d} v_d - \frac{R}{L_d} i_d + \frac{L_q}{L_d} p\omega_r i_q \qquad \textbf{(1)}$$

$$\frac{di_q}{dt} = \frac{1}{L_q} v_q - \frac{R}{L_q} i_q - \frac{L_d}{L_q} p\omega_r i_d - \frac{\lambda p\omega_r}{L_q} \tag{2}$$

In which: L_q, L_d are the inductance on q and d axis; R is the resistance of the stator windings; i_q, id are the current on q and d axis; V_q, V_d are the voltage on q and d axis; λ is the permanent magnet flux linkage; p is a number of pole pairs; ω_r is the rotational speed of the rotor.

Three phase currents in stator of the motor were being feedback and through Clack and Park transformation, enabling controlling current $i_d \approx 0$, helps controlling three phase motor similar to a one phase motor. Electromagnetic moment of the motor could be calculated using the following formula:

$$T_e = 1.5p[\lambda i_q + (L_d - L_q) i_d i_q] \approx K_t i_q \tag{3}$$

Mathematical equations of the motor when the load bearing:

$$\frac{d\omega_r}{dt} = \frac{1}{J_m (T_e - F\omega_r - T_m)} \tag{4}$$

Where: T_e is the motor torque; J_m and F are the inertia and viscous friction of rotor and load; T_m is the shaft mechanical torque, K_t is the torque constant.

Table 1. Parameters of PMSM

Parameters	value
Stator resistor	1.3Ω
Stator inductance	6.3mH
Pole pairs	4
Inertia	J=0.000108 kg*m^2
Friction factor	F=0.0013 N*m*s
Voltage constant	52.2 V_peak/1000rpm
Torque constant	0.43169 N.m/A_peak

2.2 SVPWM Algorithm [3]

SVPWM was control technique widely used in power electronic equipment control. Idea of the method was to create a continuous shift of the space vector equivalent of the voltage vector of inverter on the circle orbit (Fig.3). With the steady shift of the space vector on a circular orbit, the higher harmonics (noise) was removed and the relationship between control signal and output voltage amplitude became linear. In this article, the SVPWM technique was used in the creation of vector to control the inverter using IGBT power switches.

The combat the short circuit occurs, when the upper switches active, the lower switches must be turned off and must be coordinated delay time (dead-band) between the opening and closing of two switches on the same phase. This was done by software.

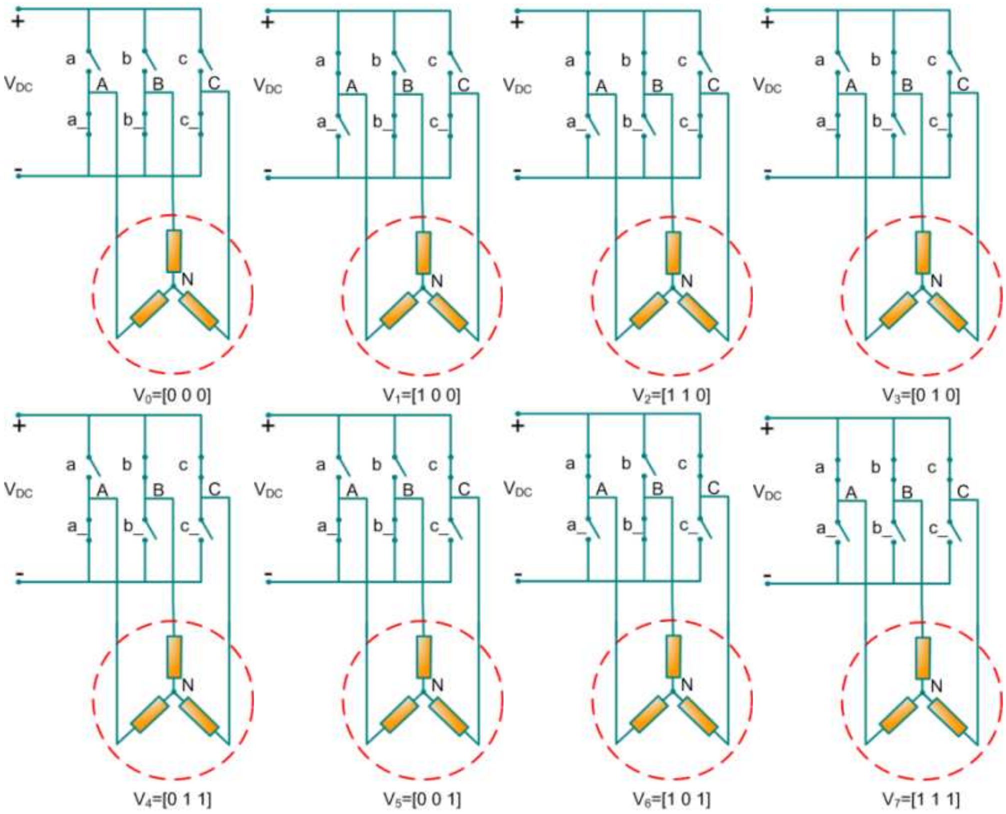


Fig. 2. Switching status of power electronic switches created eight voltage vectors to control the PMSM motor.

From the values of switching vector above, we established voltage phase to phase and phase to neutral equation to supply to the motor:

$$\begin{bmatrix} V_{AB} \\ V_{BC} \\ V_{CA} \end{bmatrix} = V_{DC} \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} \tag{5}$$

$$\begin{bmatrix} V_{AN} \\ V_{BN} \\ V_{CN} \end{bmatrix} = \frac{1}{3} V_{DC} \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} \tag{6}$$

In which: a, b, c are switching vectors of IGBT power electronic switches

To change (6) through the d, q coordinates:

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = \begin{bmatrix} 1 & \frac{-1}{2} & \frac{-1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} V_{AN} \\ V_{BN} \\ V_{CN} \end{bmatrix} \tag{7}$$

From there, we could calculate the lifetime of voltage vectors

$$\begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} = \frac{\sqrt{3}T}{V_{DC}} \begin{bmatrix} 1 & 0 \\ \frac{1}{2} & -\frac{\sqrt{3}}{2} \\ \frac{1}{2} & \frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} V_d \\ V_q \end{bmatrix} \qquad \textbf{(8)}$$

With conditions:

$$T_{1(sat)} = T_1 \frac{T}{T_1 + T_2} \qquad \textbf{(9)}$$

$$T_{2(sat)} = T_2 \frac{T}{T_1 + T_2} \qquad \textbf{(10)}$$

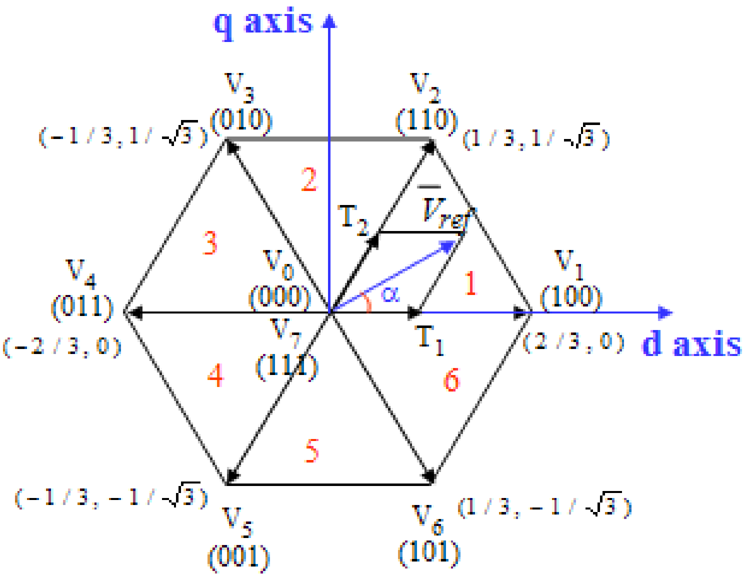


Fig. 3. Basic vector space and witching patterns.

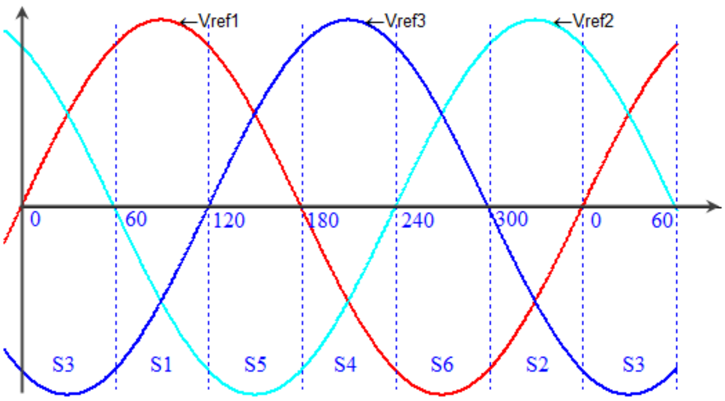


Fig. 4. Voltage values supplying for the SVPWM

Implement SVPWM by VHDL:

From value of Vref1, Vref2, Vref3, we determined the sector as Fig.4, where Vref1, Vref2, Vref3 were the input signals of the SVPWM block.

Calculate value of Tx, Ty, Tz from (8) with conditions (9) and (10)

Calculate time value of T1, T2 at table 2:

Table 2. Time value at each section

Value	S1	S2	S3	S4	S5	S6
T_1	T_z	T_y	$- T_z$	$- T_x$	T_x	$-T_y$
T2	Ty	-Tx	Tx	Tz	-Ty	- Tz

Calculate the duty cycle Taon, Tbon, Tcon by

$$T_{aon} = T_1 \frac{T - T_1 - T_2}{2} \quad \textbf{(11)}$$

$$T_{bon} = T_{aon} + T_1 \quad \textbf{(12)}$$

$$T_{con} = T_{bon} + T_2 \quad \textbf{(13)}$$

(1) Calculate comparative value CMPR1, CMPR2, CMPR3 at the duty cycles from table 3.

Table 3. The value of comparative at each section

Section	0^0~60^0 S3	60^0~120^0 S1	120^0~180^0 S5	180^0~240^0 S4	240^0~300^0 S6	300^0~360^0 S2
CMPR1	Taon	Tbon	Tcon	Tcon	Tbon	Taon
CMPR2	Tbon	Taon	Taon	Tbon	Tcon	Tcon
CMPR3	Tcon	Tcon	Tbon	Taon	Taon	Tbon

2.3 Structure of Adaptive Fuzzy Controller [4-5]

The structure of adaptive fuzzy controller includes Reference Model (RM), Fuzzifer (FI), Knowledge Base, Adjusting Mechanism, Inference Mechanism, Defuzzifier (DFI) and the PI controller (Fig 1).

The speed of the motor (ω_r) was being feedback to adaptive fuzzy controller for comparison with the RM and generated an error signal (e) and the signal changes over the time (de). Based on the e and de signal, Fuzzy controller decided adjusting and inferring the next control state of the motor. This signal was de-fuzzy and then amplified by PI controller. When motor had heavy or light load, difference between rotor speed and output of RM increased, the adaptive feature of system stimulated and changed Knowledge Base of Fuzzy Controller. With this Adjusting Mechanism, motor had a balance speed in case load changed.

Fuzzy controller used singleton fuzzy rule and method of center average solution with input linguist value includes:

$$e(k) = \omega_m(k) - \omega_r(k) \tag{14}$$

$$de(k) = e(k) - e(k-1) \tag{15}$$

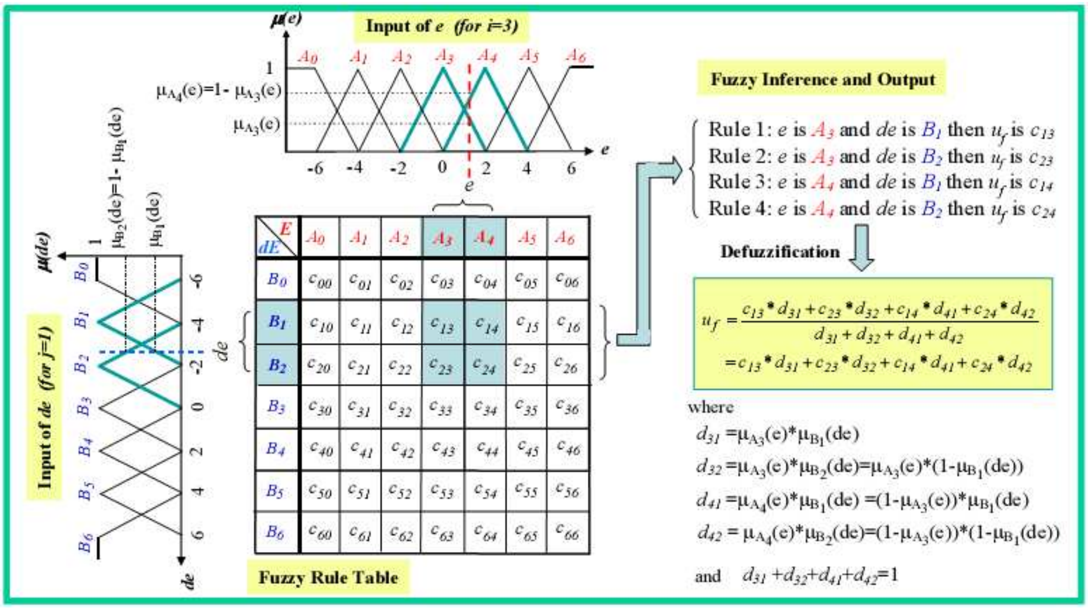
Membership function has shape of symmetrical triangle and controlling rules has form:

$$\text{If e was } A_m \text{ and de was } B_n \text{ then } u_f \text{ was } C_{m,n} \tag{16}$$

Crisp value in the output of fuzzy controller:

$$u_f(e,de) = \frac{\sum_{n=i}^{i+1}\sum_{m=j}^{j+1} c_{m,n}[\mu_{A_n}(e)*\mu_{B_n}(de)]}{\sum_{n=i}^{i+1}\sum_{m=j}^{j+1}[\mu_{A_n}(e)*\mu_{B_n}(de)]} \triangleq \sum_{n=i}^{i+1}\sum_{m=j}^{j+1} c_{m,n}d_{n,m} \tag{17}$$

In which $c_{m,n}$ $d_{n,m}$ is adjusting parameters for fuzzy controller



E \ dE	A_0	A_1	A_2	A_3	A_4	A_5	A_6
B_0	c_{00}	c_{01}	c_{02}	c_{03}	c_{04}	c_{05}	c_{06}
B_1	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
B_2	c_{20}	c_{21}	c_{22}	c_{23}	c_{24}	c_{25}	c_{26}
B_3	c_{30}	c_{31}	c_{32}	c_{33}	c_{34}	c_{35}	c_{36}
B_4	c_{40}	c_{41}	c_{42}	c_{43}	c_{44}	c_{45}	c_{46}
B_5	c_{50}	c_{51}	c_{52}	c_{53}	c_{54}	c_{55}	c_{56}
B_6	c_{60}	c_{61}	c_{62}	c_{63}	c_{64}	c_{65}	c_{66}

Fig. 5. Membership function of fuzzy controller

Reference Model. Based on the value setting in the input, RM was responsible for increasing the value after the period of time preset, the output of the RM was tangential to the set value. To build the RM to meet the above requirements, we used typical second order system:

$$\frac{\omega_m(s)}{\omega_r^*(s)} = \frac{\omega_n^2}{s^2 + 2\omega_n(s) + \omega_n^2} \tag{18}$$

Using the bilinear transformation, we had discrete function in Z domain and difference equation:

$$\frac{\omega_m(z^{-1})}{\omega_r^*(z^{-1})} = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 + b_1 z^{-1} + b_2 z^{-2}} \tag{19}$$

$$\omega_m(k) = -b_1\omega_m(k-1) - b_2\omega_m(k-2) + a_0\omega_r^*(k) + a_1\omega_r^*(k-1) + a_2\omega_r^*(k-2) \quad \textbf{(20)}$$

With coefficients: a_0=0.0077, a_1=0.0153, a_2=0.0077, b_1=-1.6496, b_2=0.6803

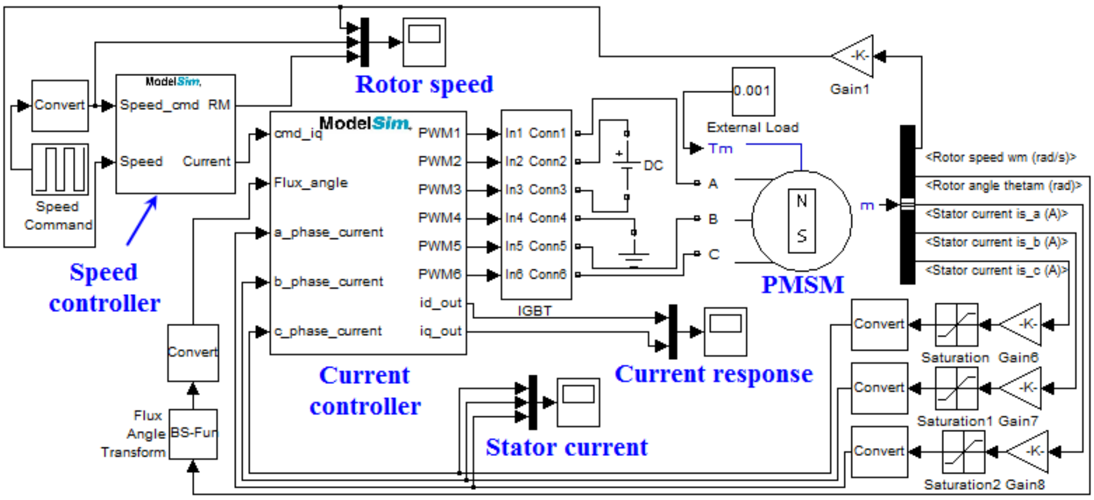


Fig. 6. Simulation model on Simulink/Matlab

Adaptive Adjustment Mechanism. The purpose of the mechanism was created the signal adjustment of the fuzzy controller so that the error between the speed of the rotor and the output of the RM was the smallest. Using the method of gradient descent to determine the adaptive control law for the system:

Definition of instantaneous value function:

$$J(k+1) = \frac{1}{2}[e(k+1)]^2 = \frac{1}{2}[\omega_m(k+1) - \omega_r(k+1)]^2 \quad \textbf{(21)}$$

Parameters $C_{m,n}$ was determined by variation of the instantaneous value function

$$\Delta C_{m,n}(k+1) = -\alpha\frac{\partial J(k+1)}{\partial C_{m,n}(k)} \quad \textbf{(22)}$$

In which: α shows the adaption rate of the system

Laplace and bilinear transform of equation (4), we had discrete function in Z domain of PMSM motor

$$\frac{\omega_r(k)}{i_q^*(k)} = \frac{K_t}{F}\frac{(1-e^{\frac{-FT}{J_m}})z^{-1}}{1-e^{\frac{-FT}{J_m}}z^{-1}} \quad \textbf{(23)}$$

In which T is sampling cycle, z-1 is a stage of delay time

Relationship between i_q^* current and output of fuzzy controller u_f was described by the equation:

$$i_q^*(k) = u_i(k-1) + (K_p + K_i)u_f(k) \quad \textbf{(24)}$$

In which K_p, K_i are the coefficients of PI controller; u_f is output function of adaptive fuzzy controller

From equation (23) and (24), we obtained the relationship between motor speed and output functions of the fuzzy controller u_f:

$$\omega_r(k)=\Phi\omega_r(k-1)+u_i(k-2)\gamma+(K_p+K_i)\gamma u_f(k-1) \tag{25}$$

In which: $\Phi = e^{\frac{-FT}{J_m}}, \gamma=\frac{K_t(1-\Phi)}{F}$

Variation of the instantaneous value function J(k+1) from equation number (22) was:

$$\frac{\partial J(k+1)}{\partial C_{m,n}(k)} = -\frac{\alpha e(k-1)(\partial\omega_r(k+1))}{\partial u_f(k)}\frac{\partial u_f(k)}{\partial C_{m,n}(k)} \tag{26}$$

From (24), (25) the parameters of the fuzzy controller could be adjusted through the function

$$\Delta C_{m,n}(k)=\alpha(K_p+K_i)\gamma e(k)d_{m,n}\approx\alpha(K_p+K_i)Sign(\gamma)e(k)d_{m,n} \tag{27}$$

3 Simulation and Experiment Results

In Fig.6, it showed the model of adaptive fuzzy controller and current control diagram for PMSM drive on Simulink. VHDL code of adaptive fuzzy speed control (RM, inference mechanism, knowledge base, adjust mechanism, DI and DFI) was embedded to Speed Controller block; VHDL codes of Clark, Park, Clark-1 and Park-1 transform; PI and SVPWM were embedded to Current Controller block; the inverter based on IGBT, PMSM were designed on Simulink/Matlab. Rotor speed, stator current, current response scopes displayed rotor speed, output of RM, speed command; Id/Iq current and three phase stator current of PMSM respectively. The membership function and the fuzzy rule table were designed as Fig.5. The learning rate was set to 0.5.

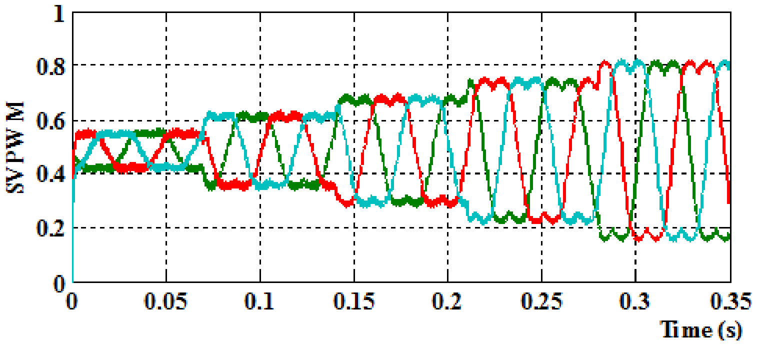


Fig. 7. The output of SVPWM after transfer to RC circuit

The sampling frequency of current control was designed with 16 kHz. Adaptive fuzzy speed control was designed with 2kHz. The clocks of 50MHz and 12.5MHz will supply all module of ModelSim. The FPGA (Altera) resource usaged of adaptive fuzzy, current control block in Fig.6 were 5115 LEs (Logic Elements) and 0RAM bits, 1,847 LEs and 196,608 RAM bits; respectively. The designed PMSM parameters used in simulation of Fig.6 are listed in table 1.

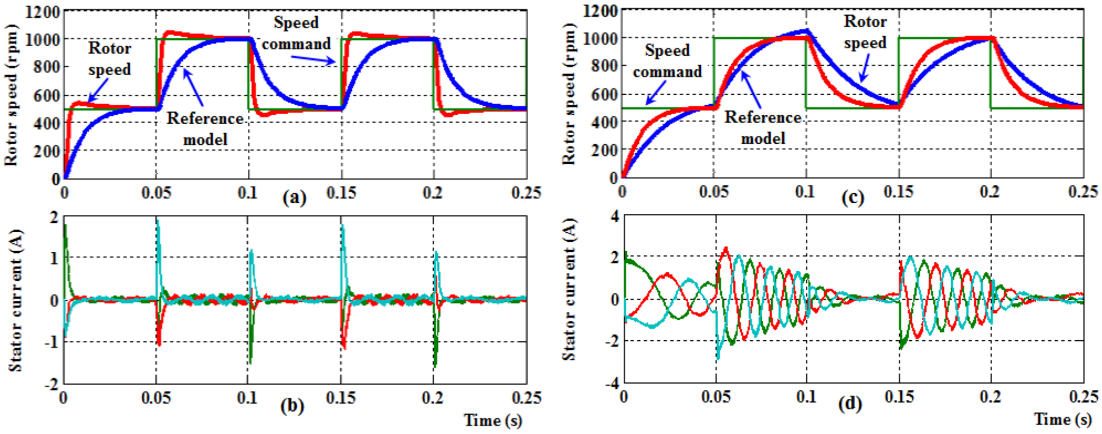


Fig. 8. The case of light load (a) and and heavy load (b) when adaptive fuzzy controller being replaced by PI controller

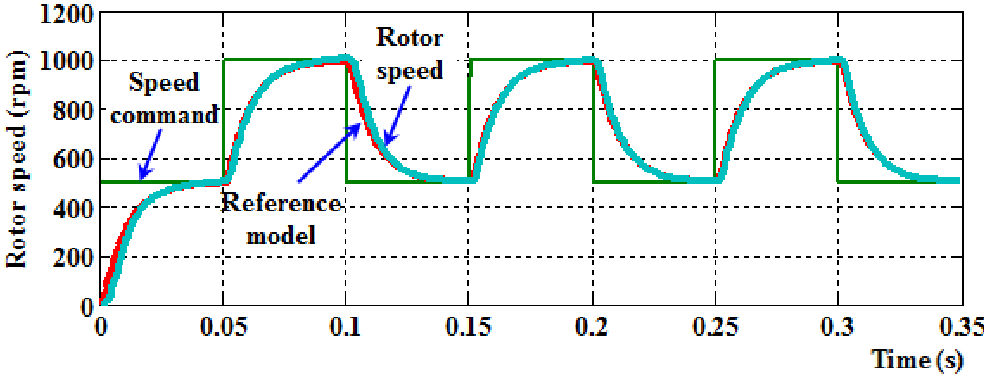


Fig 9. Rotor speed tracked the output of RM in case at light load condition and adaptive fuzzy was used

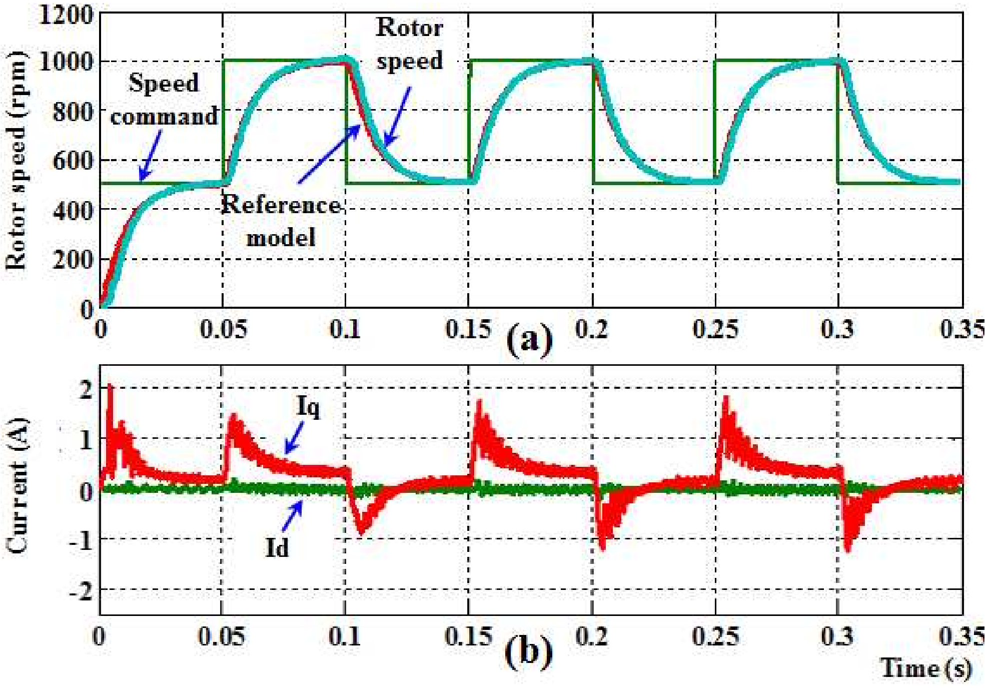


Fig. 10. Speed response (a) and current response (b) in simulation with Adaptive Fuzzy control was used at normal load

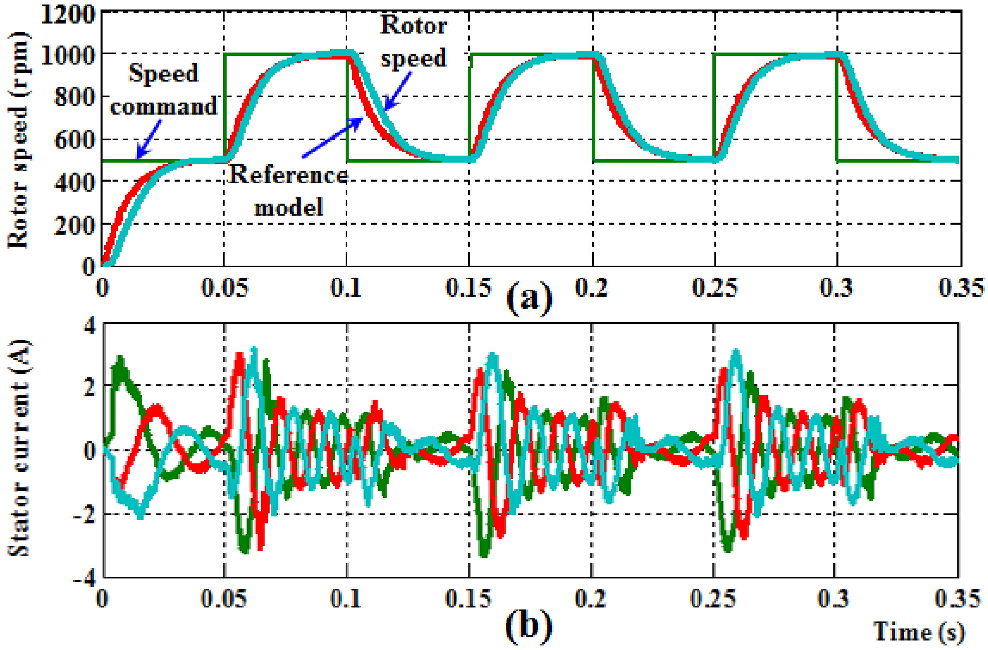


Fig. 11. The rotor speed tracked output of RM (a), the three phase stator current of PMSM (b) in case heavy load condition and adaptive fuzzy was used

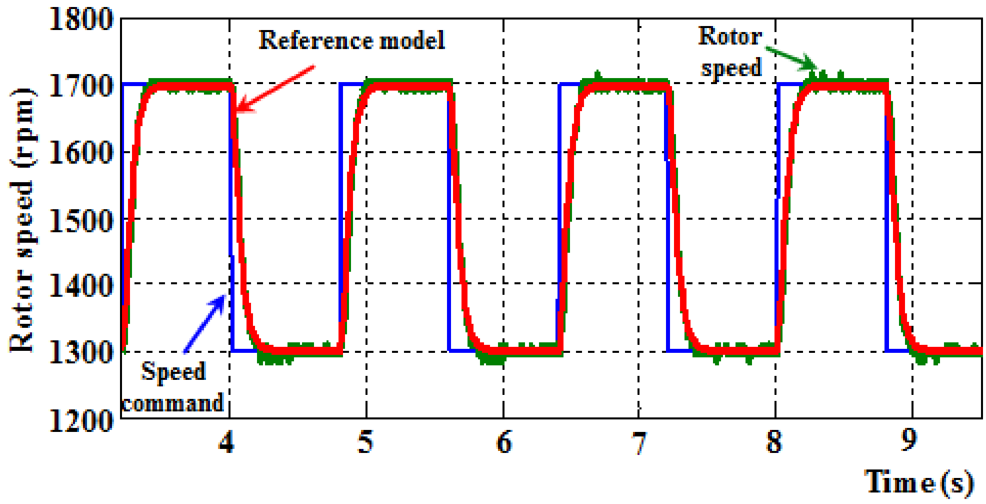


Fig. 12. Experiment result, the rotor speed tracked output of RM in case normal load condition and adaptive fuzzy was used

4 Conclusion

Adaptive fuzzy control technique has been studied for the control of vector controlled PMSM drive. The adjust mechanism produces a compensation signal by using (27) which was added to the knowledge base of the fuzzy controller to force the system to behave like the RM. The simulation results of the article show when using adaptive fuzzy controller, responding speed of the motor obtained in the case of light load, normal load and heavy load (Fig. 9, 10, 11). Although the load changed very large, the rotor speed still tracked the output of the RM, speed of PMSM was balance.

The article also was compared the case that just uses the PI controller for light and heavy load condition. PI control was designed at normal load; the rotor speed tracked the output of RM. After that, the PMSM was changed to light load and heavy load condition for testing. At the light load condition, the response of rotor speed was overshot 11% and ran faster the normal load condition (Fig.8a). At the heavy load condition, the rotor speed ran lower the normal load condition (Fig.8b).

After successful simulation, we realized this code in the experimental on FPGA kit. The experiment result shown on Fig.12, speed command repeated at 1300rpm and 1700rpm, the rotor speed also tracked the output of RM very well.

The simulation and experiment results had confirmed the efficiency of the proposed adaptive fuzzy and SVPWM for PMSM.

References

1. B. K. Bose, Power electronics and variable frequency drives – technology and application. New York: IEEE Press (1997).
2. B. K. Bose, Expert system, fuzzy logic, and neural network application, in power electronic and motion control, proceding IEEE, vol.82 no.8, pp: 1303 – 1323 (1994).
3. R. K Pongiannan, N. Yadaiah, FPGA based Space Vector PWM control IC for three phase induction motor drive conference IEEE, pp: 2061 – 2066 (2006).
4. Kevin M. Passino, Stephen Yurkovich, Fuzzy control, Addison-Wesley 301-390 (1998).
5. Phan Xuan Minh, Nguyen Doan Phuoc, Theory on fuzzy control, the Scientific and Technical Publishing, Hanoi (2006).
6. Nguyen Phung Quang, Matlab & Simulink for engineer on automatic control, the Scientific and Technical Publishing, Hanoi (2006).