

APPLICATION OF FPGA TO CONTROL SPEED OF PERMANENT MAGNET SYNCHRONOUS MOTOR WITHOUT SENSOR

Nguyen Vu Quynh¹, Tran Hanh², Trinh Tran Thanh Tam³, and Le Phuong Truong⁴

^{1,4} Electrical – Mechanical Department, Lac Hong University

^{2,3} Informatics Technology, Lac Hong University

Abstract: Permanent magnet synchronous motor (PMSM) was applied so much in the accurate manufacturing industry. Almost all of automatic systems included optical encoder for velocity measurement. The price of optical encoders is very expensive, so some authors had studied to control speed of motor without sensor. But they used Digital Signal Processor (DSP) to process, which was accuracy; the cost of systems was not cheaper. Besides, there was no paper research studied at Viet Nam. This article was presented on the speed control algorithm of the PMSM without using sensors based on FPGA technology. The contents of paper including: Firstly, the mathematical model of PMSM was derived, that based on vector control. Secondly, Rotor flux position was estimated by sliding mode control and this was feed-backed to the current and speed controller. Thirdly, the Very High speed integrated circuit Hardware Description Language (VHDL) was used to program for all systems, which those were current controller, speed controller, space vector pulse width modulation (SVPWM) and sliding mode observer (SMO). Finally, the article was presented simulation on Matlab/Simulink and experiment on FPGA kit results. The simulation and experiment results were validated the effectiveness of the speed control system without sensor. The system without sensor still had accuracy like as the system with sensor. The success of this algorithm brought in a new system with low price, small size, less noise and more optimal. Based on VHDL code, we can manufacture a dedicated IC for motor speed control without sensor

Keywords: FPGA, PI controller, VHDL, manufacturing machinery, simulate programming, sliding mode observer

1 Introduction

The use of PMSM in the industry increased strongly because of the advantages such as: high speed and precision. The majority of systems used major sensors and optical encoder to measure the speed and feedback to the controller. But in industrial applications, there are many uncertainties, for example noise, external load, friction force ... They affect to the performance quality of the system. To deal with those problems, many intelligent control techniques such as fuzzy, neural network, genetic algorithm have been developed. Besides, new advanced sensorless algorithms used the back-EMF (Electromotive force) generated in the stator winding to determine the rotor position. Based on this algorithm, we can control motor without position and speed sensor. But the back-EMF was suitable to be implemented by the fix-pointed processor, which have been implemented by DSP in most studies [2, 3]. Unfortunately, DSP suffers from a long period of development and exhausts many resources of the system and it is expensive. Altera FPGA supplied a solution for this issue. 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.

The article presented basically the theory of rotor flux estimation via - the electric current to supply to the motor and the program based on VHDL for testing. In Fig.1, three phase stator current of the motor (i_a , i_b) was feed-backed to calculate the creation of parameters for estimating the rotor angle and controlling voltage vector. PI₁, PI₂, PI₃ controller was used to generate speed control signals and current on the axis d, q. SVPWM block generated signal to control the three phase inverter; Clark, Park, Clark⁻¹ and Park⁻¹ transform block; PI controller blocks; SVPWM block; SMO block were programmed in VHDL.

The VHDL code in the motor drive system was verified by Electronic Design Automation (EDA) simulator. VHDL code was embedded to Matlab/Simulink to analyze the simulation's response (Fig.2). After that, all the HDL design elements were verified again by experiment on FPGA kit. The simulation and experiment results validate the effectiveness of the speed control system of PMSM drive without sensor.

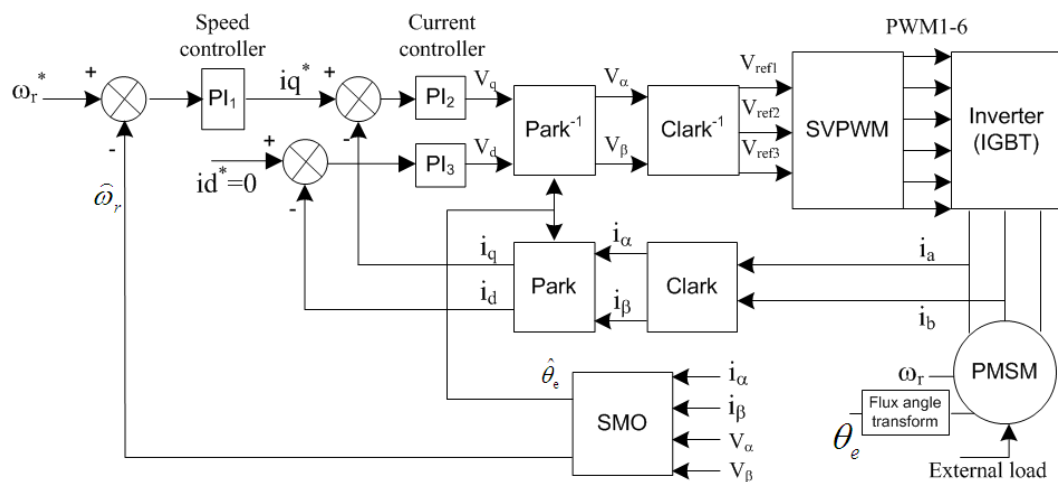


Fig. 1 Structure of system

2 SMO Block Design [10]:

The SMO was used to estimate the rotor flux position. The SMO was modeled like as a PMSM model on VHDL code. The inputs of SMO were currents and voltages on α - β axis, the outputs of SMO were estimated rotor speed and rotor flux angle. SMO block was designed consisting of estimation current, bang-bang controller, low-pass filter, flux angle position calculation and estimation speed (Fig.3). [4 ~8]

Firstly, to design the SMO, we must rely on motor parameters. The mathematical model of the PMSM motor in d-q axis was described as follows [9].

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

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

In which: L_q, L_d are the inductance on q and d axis; R is the resistance of the stator windings; i_q, i_d 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.

Mathematical equation of PMSM on the d-q rotating and α - β fixed coordinate: [2]

$$v_d = (R + sL)i_d - p\omega_r L i_q \quad (3)$$

$$v_q = p\omega_r L i_d + (R + sL)i_q + p\omega_r \lambda \quad (4)$$

$$v_\alpha = (R + sL)i_\alpha - p\omega_r \lambda \sin \theta_e \quad (5)$$

$$v_\beta = (R + sL)i_\beta + p\omega_r \lambda \cos \theta_e \quad (6)$$

In which: $L = L_d = L_q$; $V_\alpha, V_\beta, I_\alpha, I_\beta$ are the voltage and current on fixed coordinate; θ_e is the angular position at magnet flux, s is the differential operator.

Secondly, from equation (3) and (4) the electromotive force was calculated as:

$$e_\alpha = -p\omega_r \lambda \sin \theta_e \quad (7)$$

$$e_\beta = p\omega_r \lambda \cos \theta_e \quad (8)$$

Thirdly, from equation (5) and (6) were rewritten with current variable for easily observed the electromotive force:

$$\frac{di_\alpha}{dt} = -\frac{R}{L} i_\alpha + \frac{1}{L} v_\alpha - \frac{1}{L} e_\alpha \quad (9)$$

$$\frac{di_\beta}{dt} = -\frac{R}{L} i_\beta + \frac{1}{L} v_\beta - \frac{1}{L} e_\beta \quad (10)$$

Therefore, the current estimation block was designed by equation following:

$$\frac{d\hat{i}_\alpha}{dt} = -\frac{R}{L} \hat{i}_\alpha + \frac{1}{L} v_\alpha - \frac{1}{L} z_\alpha \quad (11)$$

$$\frac{d\hat{i}_\beta}{dt} = -\frac{R}{L} \hat{i}_\beta + \frac{1}{L} v_\beta - \frac{1}{L} z_\beta \quad (12)$$

In which: $\hat{i}_\alpha, \hat{i}_\beta$ are estimated current on fixed coordinate; z_α, z_β are output gain of the bang – bang controller [6, 8]:

$$z_\alpha = k * \text{sign}(\hat{i}_\alpha - i_\alpha) \quad (13)$$

$$z_\beta = k * \text{sign}(\hat{i}_\beta - i_\beta) \quad (14)$$

The difference value between i_α current and \hat{i}_α estimated current was defined by:

$$i_{\alpha e} = \hat{i}_\alpha - i_\alpha \tag{15}$$

$$i_{\beta e} = \hat{i}_\beta - i_\beta \tag{16}$$

Condition for SMO can enter sliding mode:

$$e_i^T \dot{e}_i < 0 \tag{17}$$

Fourthly, the bang – bang controller was connected at the output of current estimation. With $e_i = [i_{\alpha e} i_{\beta e}]^T$, if we choose the k being large enough, the inequality in (17) can be satisfied. So, it generates the results of $e_i = \dot{e}_i = 0$. The z_α and z_β in in (11) and (12) will approach to electromotive force:

$$z_\alpha = e_\alpha = -p\omega_r \lambda \sin \theta_e \tag{18}$$

$$z_\beta = e_\beta = p\omega_r \lambda \cos \theta_e \tag{19}$$

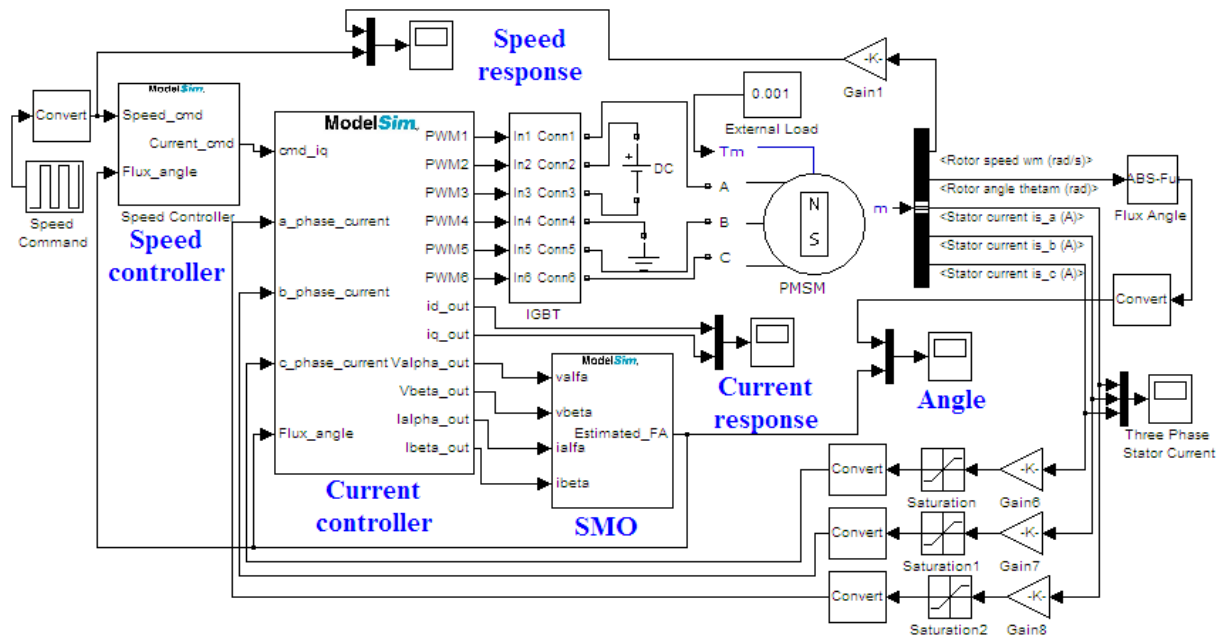


Fig. 2 Model of system on Simulink [1, 9]

Fifthly, to remove the high frequency switching in bang-bang control, a low-pass filter was added at output of bang-bang controller (Fig.3) [1]:

$$\hat{e}_\alpha = \frac{\omega_c}{s + \omega_c} z_\alpha \tag{20}$$

$$\hat{e}_\beta = \frac{\omega_c}{s + \omega_c} z_\beta \tag{21}$$

In which ω_c is cut-off frequency of low-pass filter. Finally, the rotor position $\hat{\theta}_e$ can be calculated by:

$$\hat{\theta}_e = \tan^{-1}\left(-\frac{\hat{e}_\alpha}{\hat{e}_\beta}\right) \quad (22)$$

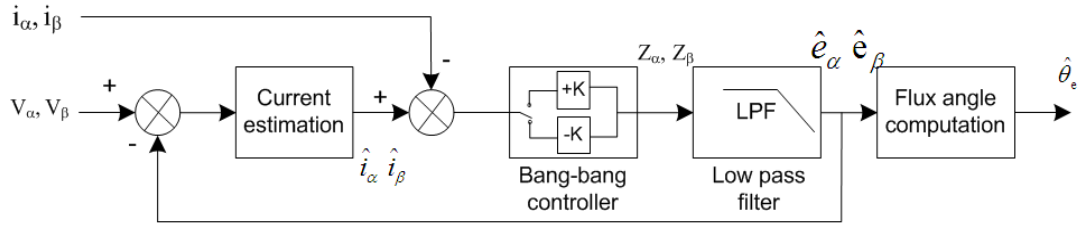


Fig.3 Structure of SMO block

To use z_α and z_β as the feedback value in sliding mode observer and to easily program, we must transfer the continuous system to the discrete system, the equations (11), (12), (20) and (21) will be:

$$\hat{i}_\alpha(n+1) = A\hat{i}_\alpha(n) + B(v_\alpha(n) - \hat{e}_\alpha(n)) \quad (23)$$

$$\hat{i}_\beta(n+1) = A\hat{i}_\beta(n) + B(v_\beta(n) - \hat{e}_\beta(n)) \quad (24)$$

$$\hat{e}_\alpha(n+1) = \hat{e}_\alpha(n) + \omega_c(z_\alpha(n) - \hat{e}_\alpha(n)) \quad (25)$$

$$\hat{e}_\beta(n+1) = \hat{e}_\beta(n) + \omega_c(z_\beta(n) - \hat{e}_\beta(n)) \quad (26)$$

In which: $A = 1 - T_s \frac{R}{L}$, $B = \frac{T_s}{L}$, T_s is sampling time of sliding mode observer. Therefore, the rotor position $\hat{\theta}_e$ can be directly calculated by

$$\hat{\theta}_e(n) = \tan^{-1}\left(-\frac{\hat{e}_\alpha(n)}{\hat{e}_\beta(n)}\right) \quad (27)$$

3 Simulation and Experiment Results

In Fig.2, it was showed the model of sensorless speed control block diagram for PMSM drive on Simulink. The VHDL code of speed control (PI1) was embedded to Speed Controller block; VHDL codes of Clark, Park, Clark⁻¹ and Park⁻¹ transform; PI₂ and PI₃, SVPWM were embedded to Current Controller block; VHDL code of SMO (it included estimation current, bang-bang controller, low-pass filter, flux angle computation, speed estimation) was embedded to SMO block. The sampling frequency of the current controller and SMO was designed with 16 kHz. The speed controller was designed with 2kHz. The clocks of 50MHz and 25MHz were supplied to all module of ModelSim. The FPGA (Altera) resource usages of Speed Controller, Current Controller, SMO block in Fig.2 are 253 LEs (Logic Elements) and 0RAM bits, 1,903 LEs and 196,608 RAM bits; 2,191 LEs and 49,152 RAM bits, respectively. The designed PMSM parameters used in simulation of Fig.2 were listed in table 1.

The estimated rotor flux angle and estimated rotor speed were feed-backed to the current loop for vector control and to the speed loop for speed control. In the simulation of sensorless speed control was running speed of PMSM with 300rpm, 600rpm, 1000rpm and 1500rpm. The simulation results for the real rotor flux position θ_e , the estimated rotor flux position $\hat{\theta}_e$ were shown in Fig. 4. The results for estimated rotor speed and real rotor speed were shown in Fig.5.

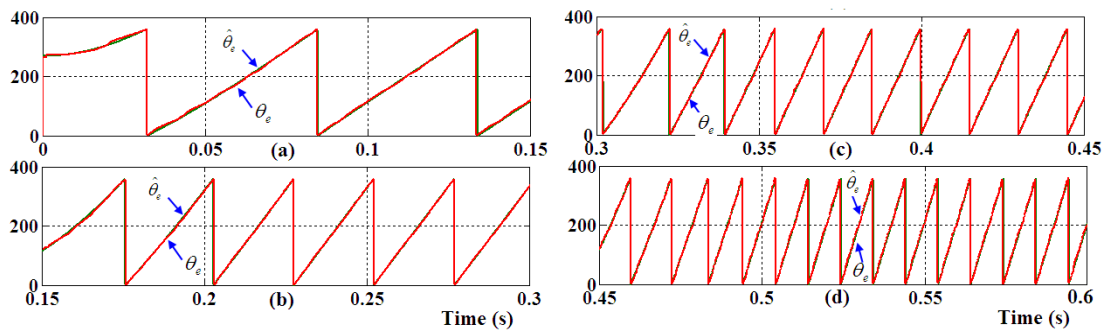


Fig. 4 Real rotor flux angle (θ_e) and estimated rotor flux angle ($\hat{\theta}_e$) under PMSM speed running at (a)300rpm, (b)600rpm, (c)1000rpm and (d)1500rpm

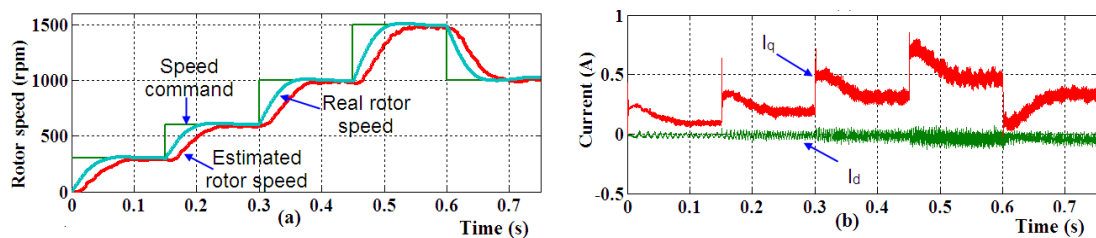


Fig. 5 (a) Step speed command, real rotor speed and estimated rotor speed response (b) current response in simulation

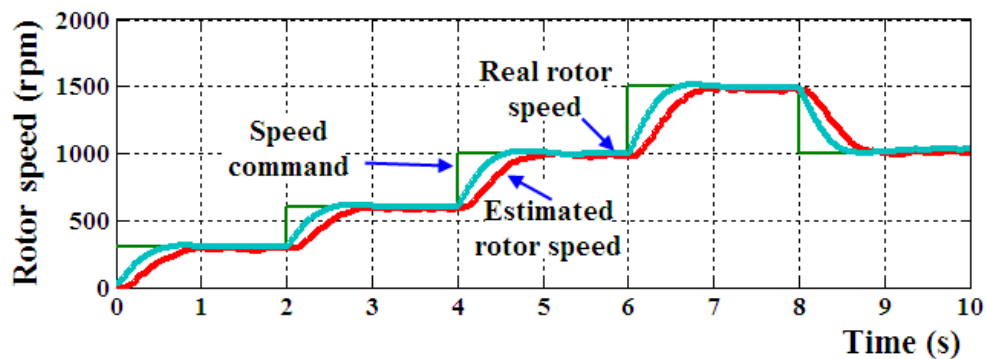


Fig. 6 Step speed command, real rotor speed and estimated rotor speed response in experiment by FPGA kit

After confirming the effectiveness of the sensorless speed control in simulation, the experiment work of the step speed response was tested by FPGA kit. In Fig.7, the dive board included IGBT, protect circuit; the encoder board was used for check PMSM's speed; ADC board converted analog current (three phase current) to digital.

The motor speed command was also designed with step varying from 0rpm→300rpm→600rpm→1000rpm→1500rpm→1000rpm, and the results for actual rotor speed and estimated rotor speed response were shown in Fig.6

4 Conclusions

This study had presented a sensorless speed control for PMSM drive and successfully demonstrated its performance through simulation by using Simulink and ModelSim and experiment on FPGA kit.

In Fig.4, it presented that the estimated rotor flux position could follows the actual rotor flux position with a little delay time. The delay time was about $80\mu s$ while PMSM running speed at

1500 rpm, and only $30\mu\text{s}$ at 300 rpm. In Fig.5a, the rising time and steady-state value are about 110ms and near 0mm, which presents a good speed following response without overshoot occurred. In Fig.6, the real rotor speed and estimated rotor speed were very similar with simulation results. The simulation and experiment results shown in Figs. 4~6 demonstrated the effectiveness and correctness of the sensorless speed control for PMSM. With those results, we can control exact motor speed without sensor. We could reduce price, reduce size, and reduce the impact of noise and make the system achieve more optimal.

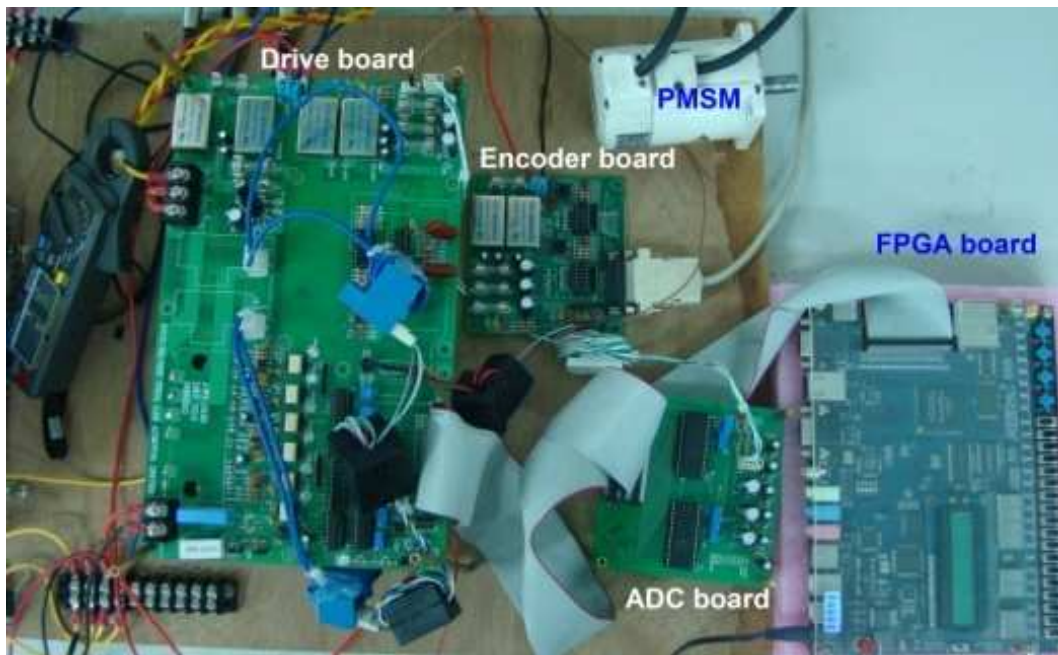


Fig. 7 Experimental system on FPGA kit

Table 1. Parameters of PMSM

Parameters	Value
Stator resistor	1.3Ω
Stator inductance	6.3mH
Pole pairs	4
Inertia	$J=0.000108 \text{ kg}\cdot\text{m}^2$
Friction factor	$F=0.0013 \text{ N}\cdot\text{m}\cdot\text{s}$
Voltage constant	$52.2 \text{ V}_{\text{peak}}/1000\text{rpm}$
Torque constant	$0.43169 \text{ N}\cdot\text{m}/\text{A}_{\text{peak}}$

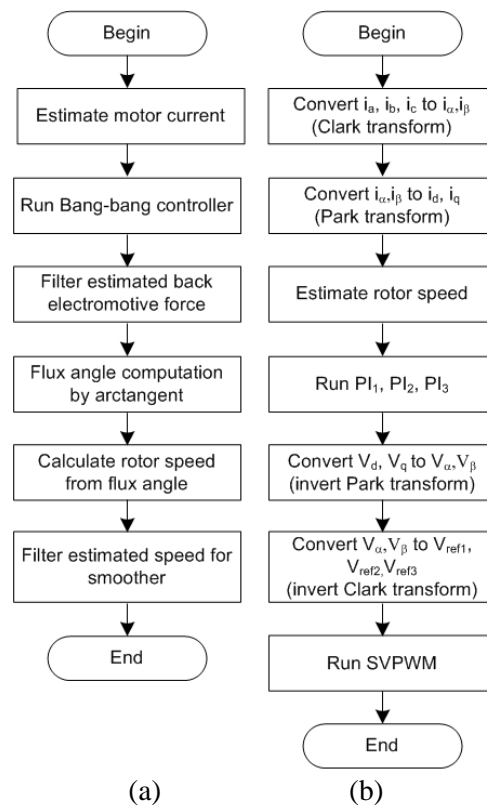


Fig.8 The flow chart of SMO block (a) and main program (b)

Reference

1. B. K. Bose, Power electronics and variable frequency drives – technology and application. New York: IEEE Press (1997)
2. Zhiqian Chen, Mutuwo Tomita, Shinji Ichikawa, Shinji Doki, Shigeru Okuma, An extended electromotive force model for sensorless control of interior permanent magnet synchronous motors, IEEE transactions on industrial electronic, Vol. 50, No. 2, pp: 288 – 295 (April 2003)
3. Ying-Shieh Kung, Ming-Shyan Wang, Chung-Chun Huang, DSP-based Adaptive Fuzzy Control for a Sensorless PMSM Drive, Control and Decision Conference, Chinese, pp: 2379 – 2384 (2009)
4. Vasilios C. ilioudis, Nikolaos I. Margaritis, PMSM sensorless speed estimation based on sliding mode observers, IEEE, pp: 2838 – 2843 (2008)
5. Liu Jun, Wang Gang, Yu JinShou, A study of SMO buffeting Elimination in sensorless control of PMSM, Proceedings of the 8th world congress on intelligent control and automation, Jinan – China (2010)
6. M. Thoma, M. Morari, Nonlinear observers and application, Springer (2007).
7. Wilfrid Perruquetti, Jean Pierre Barbot, Sliding mode control in engineering, Marcel Dekker (2002)
8. Phan Xuan Minh, Nguyen Doan Phuoc, Theory on non-linear control, the Scientific and Technical Publishing, Hanoi (2006).
9. Nguyen Phung Quang, Matlab & Simulink for engineer on automatic control, the Scientific and Technical Publishing, Hanoi (2006).
10. Ying-Shieh Kung, Nguyen Vu Quynh, Chung-Chun Huang, Liang-Chiao Huang, Simulink/ModelSim Co-Simulation of Sensorless PMSM Speed Controller, IEEE, pp: 24 – 29 (2011).