A Design of Fuzzy Sliding Mode Controller for DC Motor Drive Base on FPGA

Bui ThiHaiLinh
Department of Electrical Engineering
Thai NguyenUniversity of Technology, Thai Nguyen City, Vietnam

Abstract

This paper focus on the realization of Fuzzy Sliding Mode Controller (FSMC) for DC motor drive based on Field Programmable Gate Arrays (FPGAs). Firstly, a mathematic model of the DC motor is defined. Then to increase the performance of the DC motor drive system, an FC constructed by a fuzzy basis function and its parameter adjustable mechanism using FSMC is applied to the speed control loop of the DC motor drive system. Secondly, FPGA by using finite state machine (FSM) method is presented to realize the aforementioned controllers, and VHSIC hardware description language (VHDL) is adopted to describe the circuit of the FSMC. Finally, an experimental system is established to verify the effectiveness of the proposed FPGA-based fuzzy sliding control system for DC motor, and some experimental results are confirmed theoretically.

Keywords- *FPGA;fuzzy controller;sliding mode controller; DC motor; Finite state machine ; VHDL; Nios II;*

I. INTRODUCTION

A field programmable gate array (FPGA)incorporates the architecture of a gate arrays and the programmability of a programmable logic device. It consists of thousands of logic gates, some of which are combined together to form a configurable logic block (CLB) thereby simplifying high level circuit design. Interconnections between logic gates using software are externally defined through SRAM and ROM, which will provide flexibility in modifying the designed circuit without altering the hardware. Moreover, concurrent operation, simplicity, programmability, a comparatively low cost and rapid prototyping make it the favorite choice for prototyping an application specific integrated circuit (ASIC). Furthermore, all the internallogic elements and therefore all the control procedures of the FPGA are executed continuously and simultaneously. The circuits and algorithms can be developed in the VHSIChardware description language (VHDL) [1-2]. This methodis as flexible as any software solution. Another importantadvantage of VHDL is that it is technology independent. The samealgorithm can be synthesized into any FPGA and even has a direct path to an ASIC, which opens interesting possibilities in industrial applications in terms of performance and cost. The major disadvantage of a FPGA-based system for hardware implementation is the limited capacity of available cells. Therefore, the research on FPGA-based sliding mode or fuzzy controllers can be found in the high performance control application literature. Chen and Tang [3] proposed a FPGA-based sliding-mode control scheme that used an improved equivalent control method with out complicated computations for pulse-width modulation(PWM) brushless DC motor drives. A fixed-frequency quasi-sliding control algorithm based on switching-surface zero-averaged dynamics is reported in [4]. This FPGA based system is applied to the control of a buck-based inverter. Kim [5] proposed the implementation of a fuzzy logic controller to a FPGA system. The fuzzy logic controller is partitioned into many temporally independent function modules. Then, the FPGA chip is subsequently reconfigured one module at a time by using the run-time reconfiguration method. FPGA-based applications of various motor drives can be found in. A digital wheel-chair controller is presented [6]. The control process consists in command decoding, speed estimation, and speed serving. Through proper partitioning to concurrent blocks, the design complexity is reduced significantly for FPGAs. The concepts of car maneuvers, fuzzy logic control (FLC), and sensor-based behaviors are merged to implement human-like driving skills by an autonomous carlike mobile robot in [7]. Four kinds of FLCs are synthesizedto accomplish the autonomous fuzzy behavior control. The implementation of the proposed control on a FPGA chip is addressed. In [8], a motion control IC, which is realized on FPGAs, for X-Y table is presented. The motivation of this study is to design a suitable control scheme to confront the uncertainties that exist in a DC motor drive including the frictional force using a FPGA chip to allow possible low-cost and high-performance industrial applications. Due to its robustness and case of implementation, a fuzzy sliding-mode speed controller is adopted in this study to control the speed of DC motor. The designation and implementation of the FPGA-based control IC will be described in detail. Compared with a DSP or a PC-based fuzzy controller, the merits of the FPGA-based fuzzy controller are a parallel processing and small size in addition to a low cost. Moreover, the developed VHDL code can be easily modified and implemented to control any type of DC or AC motors as well [9].

II. MATHEMATICAL MODEL OF A DC

DC motors are widely used in industrial and domesticequipment. The control of the position of a motor with highaccuracy is required. The electric circuit of the armature andthe free body diagram of the rotor are shown inFig. 1 [10].The dynamics of a DC motor may be expressed as:

$$V_t = R_a I_a + L_a \frac{dI_a}{dt} + E_a \quad (1)$$

$$T = J\frac{d\omega}{dt} + B\omega - T_k \quad (2)$$

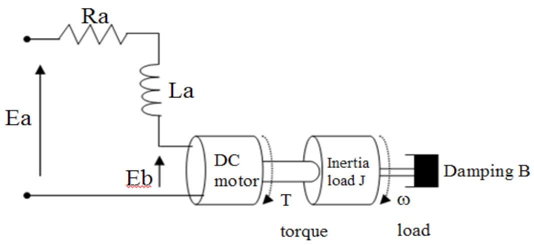

Fig1: Model of DC Motor [9]

$$T_m = K_i I_a \quad (3)$$

$$E_a = K_a \omega \quad (4)$$

$$\frac{d\omega}{dt} = \phi \quad (5)$$

With the following physical parameters:

E_a : The input terminal voltage (source), (v);

E_b : The back emf, (v);

R_a : The armature resistance, (ohm);

I_a : The armature current (Amp);

L_a : The armature inductance, (H);

J : The moment inertial of the motor rotor and load,(Kg.m^2);

T : The motor torque, (Nm)

ω : The speed of the shaft and the load (angular velocity),(Rad/s);

ϕ : The shaft position, (rad);

B : The damping ratio of the mechanical system, (Nms);

T_k : The torque factor constant, (Nm/Amp);

$T_L = 0$, DC motor runs without load.

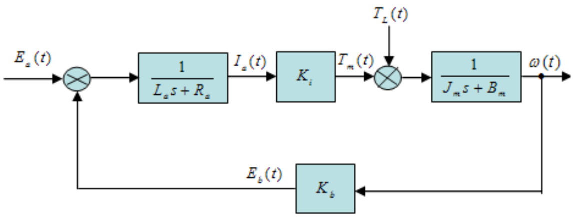

Fig 2: The Model of a DC Motor

III. SLIDING MODE CONTROL

The sliding mode control schemes have been widely developed over several decades of years. Essentially, the SMC uses discontinuous control action to drive state trajectories toward a specific hyper plane in the state space, and to maintain the armature or the control of the field. Easy controlling and cheapness of the circuit drive of the DC motors comparing to AC motors has lead to be chosen by the consumers and industries. DC motors are done mainly controls through the control of the state trajectories sliding on specific hyper plane until the origin of the state space is reached. In an SMC system [11], the control commands are adequately designed such that the states will move towards the desired sliding plane. Once the states reach the sliding surface, the system is said to be in sliding mode. During the sliding mode, the system possesses some invariance properties, such as robustness, order reduction and disturbance rejection. The first step to design a sliding mode control is to determine the sliding hyper plane with desired dynamics of the corresponding sliding motion. And the next step is to design the control input so that the state trajectories are driven and attracted toward the sliding hyper plane and then remained sliding on it for all subsequent time. In the following, the sliding mode control for continuous and discrete time system is reviewed. A Sliding Mode Controller is a Variable Structure Controller(VSC). Basically, a VSC includes several different continuous functions that can map plant state to a control surface, and the switching among different functions is determined by plant state that is represented by a switching function. Without lost of generality, consider the design of a sliding mode controller for the following second order system ;$u(t)$ is the input to the system. The following is a possible choice of the structure of a sliding mode controller u is control law :

$$u = -k\,sgn(s) + u_{eq} \quad (6)$$

Where u_{eq} is called equivalent control which is used when the system state is in the sliding mode; $u_{eq} = k_{eq}(\dot{e} + \lambda e)$ and $k_{eq} = u_f * N$; N is the constant, $N>0$. The k is a constant and it is the maximal value of the controller output. The s is called switching function because the control action switches its sign on the two sides of the switching surfaces = 0. The s is defined as:

$$s = \dot{e} + \lambda e \quad (7)$$

Where $e(k) = \omega_p^*(k) - \omega_p(k)$ and ω_p^* is the desired state. $\lambda$is a constant. $sgn(s)$ is a sign function, which is defined as:

$$sgn(s) = \begin{cases} -1 & \text{if } s < 0 \\ 1 & \text{if } s > 0 \end{cases} \quad (8)$$

The control strategy adopted here will

guarantee the systemtrajectories move toward and stay on the sliding surfaces = 0 from any initial condition if the following conditionmeets:

$$s\dot{s} < \eta|s| \quad (9)$$

Where η is a positive constant that guarantees the system trajectories hit the sliding surface in finite time. Using a sign function often causes chattering in practice. One solution is to introduce a boundary layer around the switch surface (7):

$$u = u_c + u_{eq} \quad (10)$$

Where $u_c = -k*sat(s/\varphi)$ and constant factor φ defines the thickness of the boundary layer. The $sat(s/\varphi)$ is a saturation function which is defined as:

$$sat\left(\frac{s}{\phi}\right) = \begin{cases} \frac{s}{\phi} & \text{if } \left|\frac{s}{\phi}\right| \le 1 \\ \text{sign}\left(\frac{s}{\phi}\right) & \text{if } \left|\frac{s}{\phi}\right| > 1 \end{cases} \quad (11)$$

with

$$sign\left(\frac{s}{\phi}\right) = \begin{cases} -1 \text{ if } \frac{s}{\phi} < 0 \\ 0 \quad \text{if} \quad \frac{s}{\phi} = 0 \\ 1 \text{ if } \frac{s}{\phi} > 0 \end{cases} \quad (12)$$

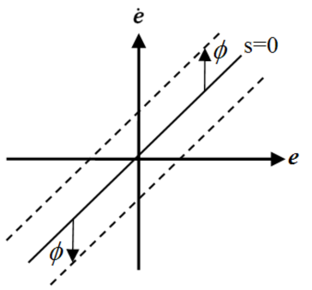

Fig3: The Sliding Surface and the Boundary

IV. FUZZY CONTROLLER

The tracking error and the change of the errorare defined as S and $\dot{S}$ (or ds). And $s$is sliding surface in (7) by $s = \dot{e} + \lambda e$ and $\dot{s}$ $=s(k) - s(k-1)$; $s(k) = de(k)+\lambda e(k)$. Therefore,

$$s(k-1)=de(k-1)-\lambda e(k-1) \quad (13)$$

and $de(k-1)=e(k-1)-e(k-2)$;(14)

$s(k-1)$ and $de(k-1)$ for calculating in the FSM in Fig.6

and$s$, ds and u_f are input and output variables of FC, respectively. The design procedure of the FC is as follows:

1. Take the s and ds as the input variables of the FC, and define their linguist variables as S and dS. The linguist value of S and dS are $\{A_0, A_1, A_2, A_3, A_4, A_5, A_6\}$ and $\{B_0, B_1, B_2, B_3, B_4, B_5, B_6\}$, respectively. Each linguist value of S and anddS. The linguist value of S and dS are $\{A_0, A_1, A_2, A_3, A_4, A_5, A_6\}$ and $\{B_0, B_1, B_2, B_3, B_4, B_5, B_6\}$, respectively. Each linguist value of S and dS is based on the symmetrical triangular membership function which is shown in Fig.4.
2. Compute the membership degree of s and ds. Fig.4 shows that the only two linguistic values are excited (resulting in a non-zero membership) in any input value, and the membership degree $\mu_{A_i(s)}$ can be derived by
   $\mu_{A_i}(s) = \frac{s_{i+1} - s}{2}$ and $\mu_{A_{i+1}}(s) = 1 - \mu_{A_i}(s)$ (15)where $s_{i+1} \triangleq -6 + 2*(i+1)$. Similar results can be obtained in computing the membership degree $\mu_{B_j(ds)}$.
3. Select the initial fuzzy control rules by referring to the dynamic response characteristics, such as,
   IF s is A_i and Δs is B_j THEN u_f is $c_{j,i}$ (16)
   where i and j = 0~6, A_i and B_j are fuzzy number, and $c_{j,i}$ is real number. The graph of fuzzification and fuzzy rule table is shown in Fig. 4.
4. Construct the fuzzy system $u_f(s,ds)$ by using the singleton fuzzifier, product-inference rule, and central average defuzzifier method. Although there are total 49 fuzzy rules in Fig. 4will be inferred, actually only 4 fuzzy rules can be effectively excited to generate a non-zero output. Therefore, the (11) can be replaced by the following expression:

$$u_f(s,ds) = \frac{\sum_{n=i}^{i+1}\sum_{m=j}^{j+1} c_{m,n}[\mu_{A_n}(s)*\mu_{B_m}(ds)]}{\sum_{n=i}^{i+1}\sum_{m=j}^{j+1} \mu_{A_n}(s)*\mu_{B_m}(ds)} \triangleq \sum_{n=i}^{i+1}\sum_{m=j}^{j+1} c_{m,n} * d_{n,m} \quad (17)$$

where $d_{n,m} \triangleq \mu_{A_n}(s) * \mu_{B_m}(ds)$. And those $c_{m,n}$ are consequent parameters. In addition, by using (15), it is straightforward to obtain $\sum_{n=i}^{i+1}\sum_{m=j}^{j+1} d_{n,m} = 1$ in (17).

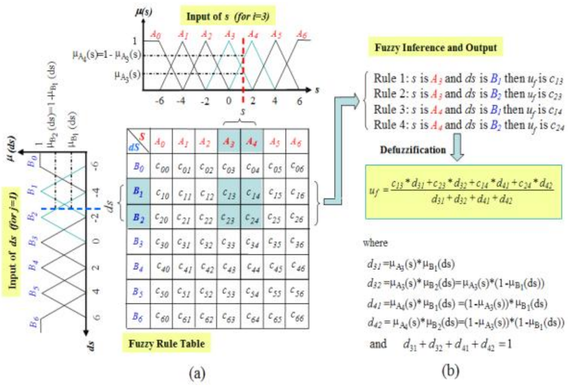

Fig4:The Symmetrical Triangularmembership Function of *S* and *Ds*, Fuzzy Rule Table, Fuzzy Inference and Fuzzification

V. DESIGN OF FUZZY SLIDING MODE CONTROLLER

In this section, a fuzzy sliding surface is introduced to develop a sliding mode controller which the expression$-k$*sat(s / ϕ) is replaced by an inference fuzzy system for eliminate the chattering phenomenon. The designed fuzzy logic controller has two inputs and an output. The inputs are sliding surface (s)and the change of the sliding surface $\dot{s}$ (or ds) in a sample time, and output is the u_f. The control law $u = -k * sat(s) + u_{eq}$ in*(10)*is computed with explanation in section III. The configuration of the overall control system is shown in Fig.5

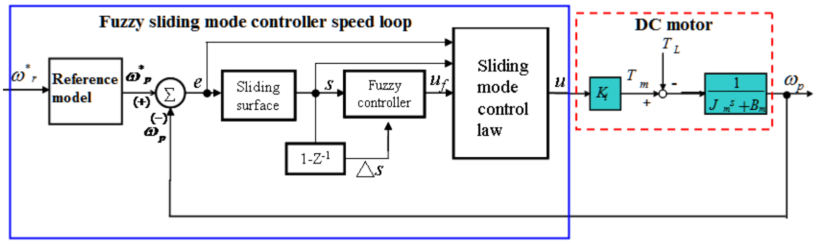

Fig5: Diagram of FSMC for DC Motor Drive System

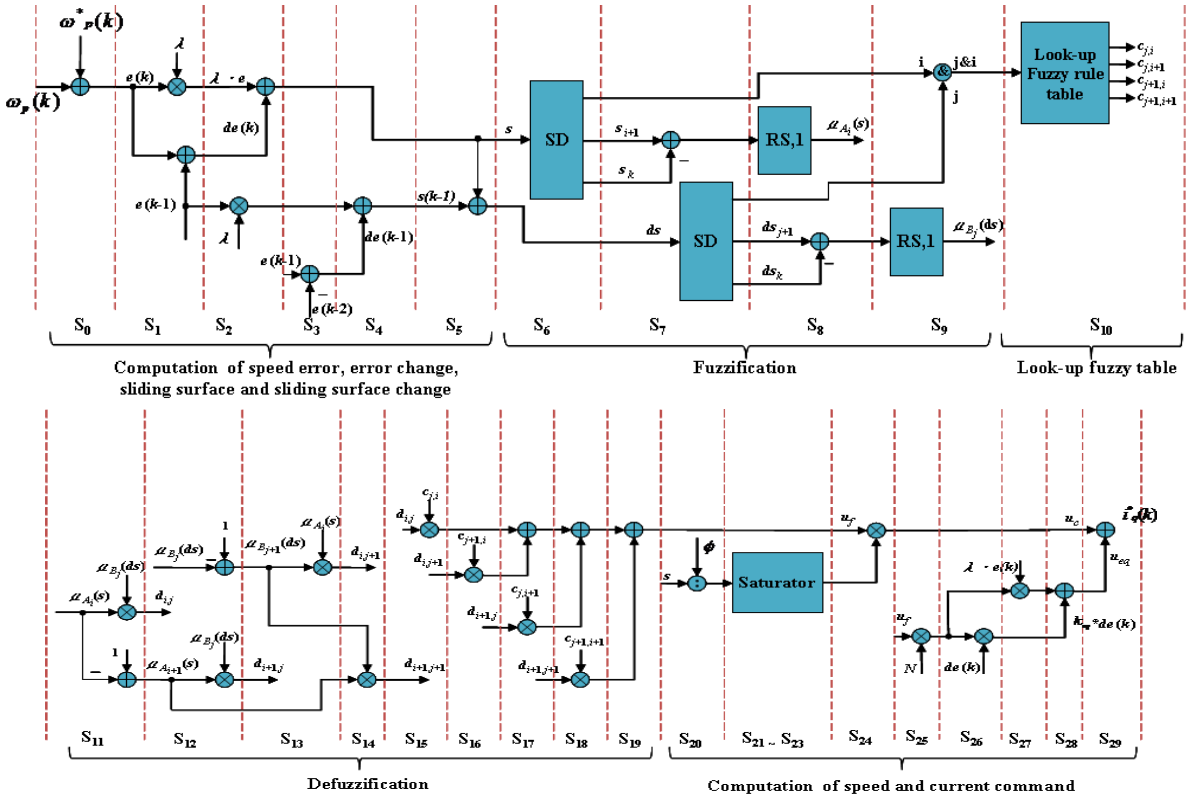

Fig6: State Diagram of a FSM for Describing the Fuzzy Sliding Mode Controller in Speed Loop

VI. DESIGN OF FPGA-BASEDSLIDING MODE CONTROLLER FOR DC MOTOR DRIVE

The internal architecture of the proposed FPGA-based motion control IC for DC drive is shown in Fig.7. The FPGA uses Altera DE2 EP2C35 and a Nios II embedded processor which is downloaded into FPGA to construct a SoPC environment. The motion control IC which comprises a Nios II embedded processor IP and a speed control IP, is designed under the SoPC environment. The speed control IP implemented by hardware is adopted to realize the function of a speed fuzzy sliding modecontroller, acurrent controller, PWM generation, QEP detection and transformation, etc.[12]

An FSM is employed to model the fuzzy sliding controller in speed loop which is shown in Fig. 6. In this Figure, it uses adders, multipliers and registers, divider, saturator, etc. and manipulates 30 steps machine to carry out the overall computation. With exception of the data type in reference model are 16-bits, others data type are designed with 32-bits length.

VII.EXPERIMENTAL SYSTEM AND RESULTS

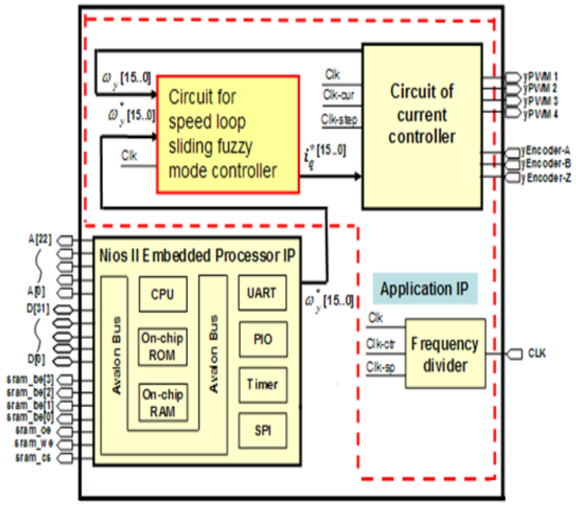

Fig7:Internal Circuit of the Proposed FPGA-Based Motion Control IC for DC Motor Drive

The experimental system is shown in Fig.8. Firstly, the power supplies AC voltage 110V to converter; Then the AC to DC converter converts the voltage from AC signal to DC signal includes: 5V DC supplies to ICs; and also transformer which transforms to 15V DC supplies to IGBTs. DC motor driver board specialized circuits (motor drivers) have been developed to supply motors with power and to isolate the other ICs from electrical problems (12V). The encoder using the 24V DC is supplied by power supply; DC motor uses the 24V DC.The FPGA uses Alter a DE2 EP2C35 which is a core component in this system and used to develop the proposed controller.

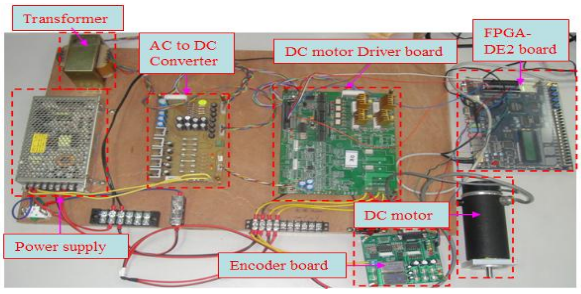

Fig8: Experimental System

Three external loads are added to DC motor which is considered to evaluate the robustness of applied controller as shown in Fig.9.

To investigate the effectiveness of the proposed control algorithm, three experimental cases with different conditions are carried out.

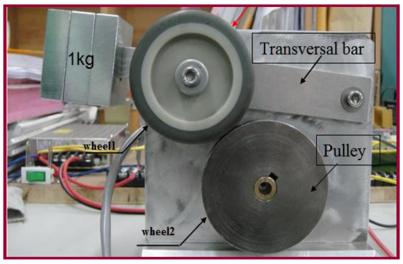

Fig9: Experimental system

- Case 1: Under external load T= 0 Kg
- Case 2: Under external load T= 0.6 Kg
- Case 3: Under external load T= 1 Kg

Some experimental results are provided to demonstrate the effectiveness of the proposed FPGA-based control system. We obtained some response results using FSMC in controlling speed of DC motor with square commands. By changing the button control, the speed is controlled by hardware system as in Fig.8.The results obtained from Nios II IDE environment by implementing as hardware system and are plotted in Matlab software. Firstly, in case 1 condition without external load, using FSMC, the response performs without overshoots and rabidly track the command at 20ms as shown in Fig.10. Secondly, the case 2 condition under 0.6Kg external load is applied; the FSMC gives good performance with a small overshoots and oscillation phenomenon. Moreover the rising time to reach the speed commands at 60ms as in Fig.11. Finally, the case 3 condition with 1Kg external load is applied. The FSMC is used which gives response with a negligible overshoots as shown in Fig.12 Initially, the mover of the DC motor tracks the commands with overshoot at some changing points, and then the $c_{i,j}$ parameters are tuned to adequate values, and finally the rotor speed can closely follows the commands in all of three cases.

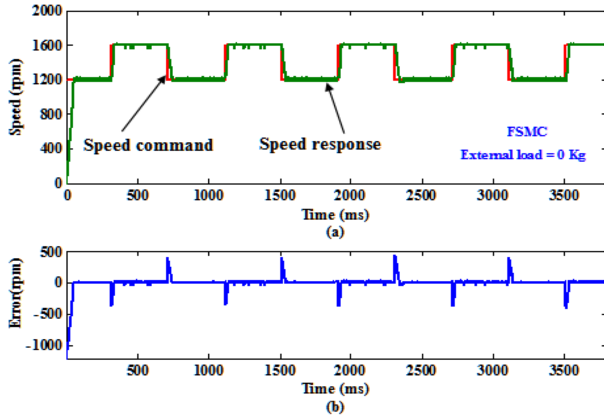

Fig 10: Square Responses using the Fuzzy Sliding Mode Controller under 0 Kg External Load

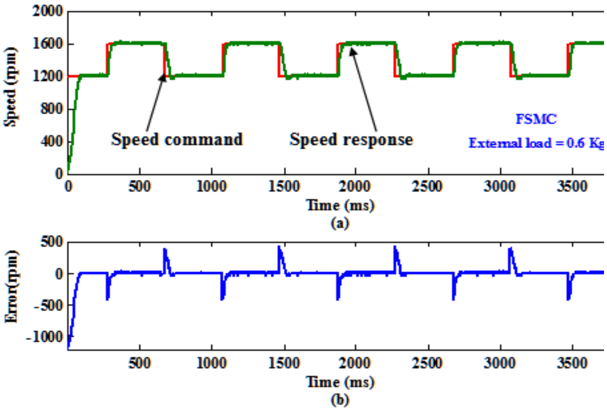

Fig 11: Square Responses using the Fuzzy Sliding Mode Controller under 0.6 Kg External Load

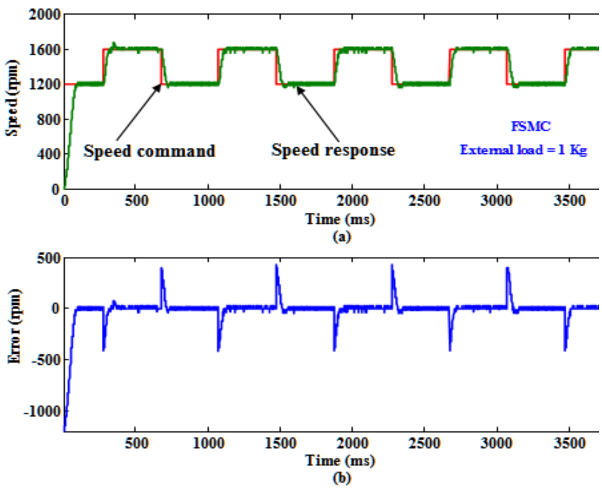

Fig 12: Square responses using the Fuzzy Sliding Mode Controller under 1 Kg External Load

VIII. CONCLUSION

This study has successfully demonstrated the design and implementation of a FPGA-based fuzzy sliding-mode control system for DC motor. First, the dynamic model of DC motor drive was introduced; then, a sliding mode controller technique was designed. The work herein is summarized as follows. The functionalities required to build a fully digital motion controller of DC motor drive have been integrated in one FPGA chip. The behavior of a FSMC has been successfully described by VHDL language. Finally, some experimental results are verified the effectiveness of the proposed controller system with good performance response as well.

REFERENCES

[1] Website: http://www.altera.com

[2] E. Monmasson and M. N. Cirstea, “FPGA design methodology for industrial control systems – a review” IEEE Trans. Ind. Electron., vol. 54, no.4, pp.1824-1842, Aug. 2007.

[3] Chen, J., and Tang, P.C,“A sliding mode current control scheme forPWM brushless DC motor drive”, IEEE Trans. Power Electron.,1999, 14, (3), pp. 541–551.

[4] Ramos, R.R., Biel,D., Fossas, E.,andGuinjoan, F.: ‘Afixedfrequencyquasi-sliding control algorithm: application to power inverters design by means of FPGA implementation’, IEEE Trans.Power Electron., 1999, 18, (1), pp. 344–355.

[5] Kim, D.: ‘An implementation of fuzzy logic controller on the
reconfigurable FPGA system’, IEEE Trans. Ind. Electron., 2000, 47,(3), pp. 703–715.

[6] Y.S. Kung**,** M.S. Wang andC.Y. Yang, “Realization of a Motion Control IC forElectric-Powered Wheelchair,”In Proceeding ofthe 2010 IEEE International Conference on Industrial Informatics (IEEE INDIN 2010),pp.523-528.

[7] T.S. Li, S.J. Chang and Y.X. Chen, “Implementation of human-like driving skills by autonomous fuzzy behavior control on an FPGA-based car-like mobile robot,” IEEE Trans. Ind. Electron.,vol. 50, no. 5, pp. 867-880, 2003.

[8] Y.S. Kung**,** M.S. Wang andC.Y. Yang, “Realization of a Motion Control IC forElectric-Powered Wheelchair,”In Proceeding ofthe 2010 IEEE International Conference on Industrial Informatics (IEEE INDIN 2010),pp.523-528.

[9] F.J.Lin, D.H.Wang and P.K.Huang, “FPGA-based fuzzy sliding mode control for a linear induction motor drive”IEE Proc.- Electr. Power Application, vol. 152, no.5, pp. 1137-1148, 2005.

[10] Mohsen Fallahi and Sassan Azadi, “Fuzzy PID Sliding Mode Controller Design for the position control of a DCMotor”in 2009 International Conference on Education Technology and Computer, pp.978-0-7695-3609-5, 10.1109/ICETC.2009.12 17-20 April 2009.

[11] A Hazzab, I.K. Bousserhane , M. Kamli& M. Rahli “A new fuzzy Sliding Mode Controller for Induction motor Speed Control”, ALGERIA University of Sciences and Technology of Oran, ORAN (31000)ALGERIA.

[12] Ying-Shieh Kung, Nguyen Khanh Quang and Le Thi Van Anh, “FPGA-Based Neural Fuzzy Controller Design for PMLSM Drive”, Power Electronic and Drive system, 2009.PEDS 2009 International Conference.

[13] Y.S. Kung, R.F. Fung and T.Y. Tai, “Realization of a motioncontrol IC for X-Y Table based on novel FPGA technology”, IEEE Trans. Ind. Electron., vol. 56, no. 1, pp. 43-53, Jan. 2009.

[14] Y.S. Kung, “Design and implementation of a high-performance PMLSM drives using DSP chip,” IEEE Trans. on Ind. Electron., vol. 55, no. 3, pp. 1341-1351, March 2008.