JPE 10-1-10

Implementation of a Fuzzy PI Controller for Speed Control of Induction Motors Using FPGA

R. Arulmozhiyal† and K. Baskaran*

† Dept. of Electrical and Electronics Eng., Sona College of Technology, Salem, Tamil Nadu, India
* Dept. of Computer Science Eng., Government College of Technology, Coimbatore, Tamil Nadu, India

Abstract

This paper presents the design and implementation of voltage source inverter type SVPWM based speed control of an induction motor using a fuzzy PI controller. This scheme enables us to adjust the speed of the motor by controlling the frequency and amplitude of the stator voltage; the ratio of the stator voltage to the frequency should be kept constant. A model of the fuzzy control system is implemented in real time with a Xilinx FPGA XC3S 400E. It is introduced to maintain a constant speed to when the load varies.

Key Words: Field programmable Gate Array (FPGA), Fuzzy PI controller, Induction motor, Space Vector Pulse Width Modulation (SVPWM), Very High speed integrated circuit Hardware Descriptive Language (VHDL)

I. INTRODUCTION

In recent years there has been a great demand in industry for adjustable speed drives. Induction motors are used in many applications such as HVAC (heating, ventilation and air-conditioning), industrial drives (motion control, robotics), automotive control (electric vehicles), etc. [1], [2].

The space vector pulse width modulation (SVPWM) method is an advanced, computation-intensive PWM method and it is possibly the best among all the PWM techniques for variable frequency drive applications [3]. Because of its superior performance characteristics, it has been finding widespread application in recent years. Space vector modulation is based on the representation of three phase voltages as space vectors. It exhibits the features of good dc-bus voltage utilization and a low THD when compared to other PWM methods [4], [5]. Most space vector modulation schemes are carried out using analog circuits or micro-controllers. Compared with sinusoidal pulse width modulation (SPWM), SVPWM is more suitable for digital implementation and can increase the obtainable maximum output voltage with a maximum line voltage approaching 70.7% of the DC link voltage [6].

Classical control systems like PI control have been used for the speed control of induction machines. The main drawbacks of classical PI controllers are their large overshoot and excessive settling time. To face these problems, fuzzy PI logic control has recently been applied to the control of electrical drive systems [7]. Fuzzy logic, deals with problems that have vagueness, uncertainty and use membership functions with values varying between 0 and 1 [8]. This means that if reliable expert knowledge is not available or if the controlled system is too complex to derive the required decision rules, the development of a fuzzy logic controller becomes time consuming and tedious or sometimes impossible. In cases where expert knowledge is available, fine-tuning of the controller might be time consuming as well [9].

However, induction motors can only run at their rated speed when they are connected to the main power supply. This is the reason why variable frequency drives are needed to vary the rotor speed of an induction motor [10]–[12]. The most popular algorithm for the control of a three phase induction motor is the V/F control approach based on the space vector pulse width modulation (SVPWM) technique to drive a voltage source inverter using fuzzy PI control [13].

The field programmable gate array (FPGA) is one of the substitutes to address the major disadvantages of conventional microprocessor/DSP control. This is due to the major advantage that it can be modified to perform any application and is not specific to a particular function. A FPGA has several advantages. These include on field control, reprogrammability, convenient software tools, high efficiency and very high significant integration density [14]. This paper demonstrates that a more efficient and faster solution is the use of field programmable gate arrays [15]–[18]; it investigates the methods to control the speed of SVPWM based induction motor using fuzzy PI control. The proposed design is tested by experiments. Space vector pulse width modulation is discussed in the second section of this paper. Fuzzy logic control design is discussed in the third section of the paper. The fourth section deals with the FPGA controller. The fifth section is about the implementation of a fuzzy PI controller using FPGA and the

Manuscript received Sep. 2, 2009; revised Dec. 2, 2009
† Corresponding Author: arulmozhiyal@gmail.com
Tel : +91 427 4099725, Fax:+91 427 4099888, Sona College of Technology
* Dept. of Computer Science Eng., Government College of Tech., India

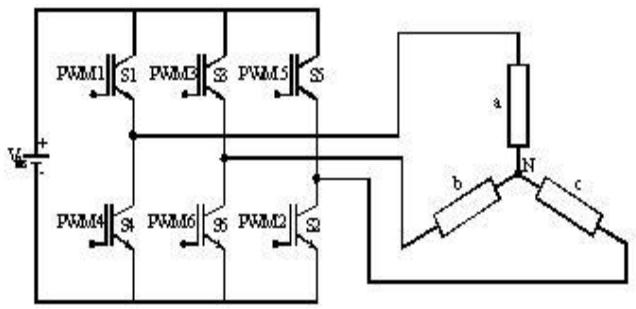


Fig. 1. Power Circuit topology of a three-phase VSI.

results of the implementation. The last section in this paper is the conclusion.

II. Space vector pulse width modulation

The basic power circuit topology of a three-phase voltage source inverter supplying a star connected three-phase load is given in Fig. 1. The power circuit contains in general six bidirectional semiconductor switches such as IGBTs with antiparallel diodes for protection. The two power switches of one leg are complementary in operation with a small dead band between the switching of two devices. The switching operation of the inverter yields in total 8 output vectors with 6 being active or non zero and two zero vectors. The six active vectors are labeled as V_1, V_2, V_3, V_4, V_5 and V_6 while the two zero vectors are labeled as V_o and V_7.

The three phase sinusoidal and balance voltages are given by the equations as follows:

$$\begin{aligned} V_{an} &= V_m \cos(\omega t) \\ V_{bn} &= V_m \cos(\omega t - \frac{2\pi}{3}) \\ V_{cn} &= V_m \cos(\omega t + \frac{2\pi}{3}) \\ \bar{V} &= \frac{2}{3}\left[V_{an} + aV_{bn} + a^2 V_{cn}\right] \end{aligned} \tag{1}$$

If these 8 voltage vectors are converted to two axes, they can be plotted as shown in Fig. 2. The tips of the 6 non zero vectors, when cornered form a regular hexagon with the two zero vectors lying at the origin.

The binary digits shown in Fig. 2 represent the states of the three legs; the most significant bit represents leg 'c', the middle digit represents leg 'b' and the least significant bit is for leg 'a'. The digit '0' represents the lower switch being 'on' while digit '1' represents the upper switch being 'on'.

The V_{ref} in the $\alpha - \beta$ plane rotates circularly, so that the output voltage will be sinusoidal. Since the voltage source inverter can have 8 states, V_{ref} can only be synthesized by using 8 voltage vectors. There can be infinite ways to synthesize the input reference but the most simple is by using the two neighboring active vectors and a zero vector. In other words it can be said that the inverter is switched in such a way that the two neighboring output voltages are generated. If the direction of the reference vector rotation is assumed as anticlockwise then the vector to the right is designated with the suffix 'a' and the vector lying on the left of the reference

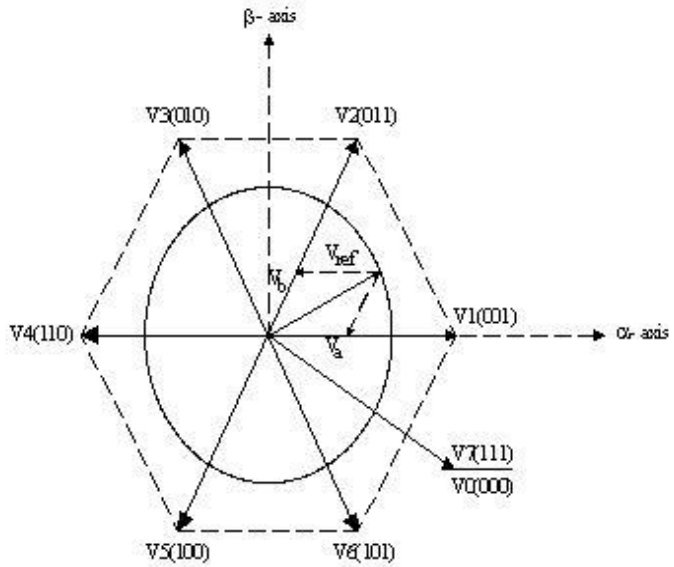


Fig. 2. Space voltage vectors of a three-phase VSI.

is denoted with the suffix 'b'. Inverter switching is done in such a way that one state (say 001) remains for some time (say T_a) and is followed by another state (say 011) for some time (say T_b). The remaining time out of one switching period (T_s) is filled with the application of vector V_o for time $T_0/2$ and vector V_7 for time $T_0/2$. Thus a symmetrical space vector modulation is obtained. If V_{ref} is in the 1^{st} sector, and:

T_a-Switching time for non zero vector (V1) in 1^{st} sector
T_b-Switching time for non zero vector (V2) in 1^{st} sector
T_0-Zero vector switching time (V_o,V_7) in 1^{st} sector
T_s-Total switching time period
Then:

$$\begin{aligned} V_{ref} &= \bar{V}_a + V_b \\ \bar{V}_{ref} \times T_s &= \bar{V}_1 \times T_a + \bar{V}_2 \times T_b + \bar{V}_0(V_7) \times T_0 \\ T_s &= T_a + T_b + T_c \end{aligned} \tag{2}$$

The second equation is termed the equal volt-sec principle. The inverter output voltage can assume only eight discrete locations in the complex plane, including the zero voltage vectors. The zero voltage vector will be included in the switching sequence to minimize the commutations in the inverter (Turn ON and Turn OFF losses will be less). One can generate any voltage vector lying inside the hexagon whose corners are on the six active switching state vectors $(V_1, \ldots, V_6)$ in the space vector modulation strategy. The inverter out net reference vector $\bar{V}ref$ can be approximated during a sampling time $\Delta t = 1/(2f_s)$ by a sequence of 3 space voltage vectors V_a, V_b and V_N, where V_a and V_b are 2 of the adjacent six active vector $V_1, \ldots, V_6$. V_N is a zero vector, V_o and V_7 are chosen in such a way as to minimize the commutations in the inverter and f_s is the switching frequency.

The above time intervals T_a, T_b and T_o during which the switching state vectors V_a, V_b and V_N are applied are derived from the geometry of the hexagon and can be written as the

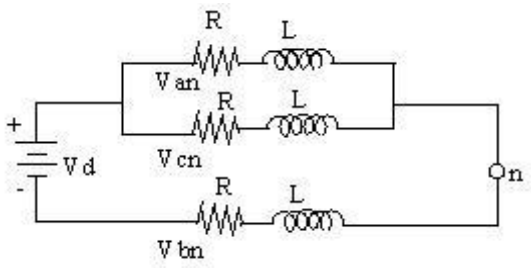


Fig. 3. Equivalent circuit.

follows:

$$T_a = \frac{\sqrt{3}}{\pi} m T_s \left[\sin\frac{k\pi}{3}\cos\theta - \cos\frac{k\pi}{3}\sin\theta \right]$$
$$T_b = \frac{\sqrt{3}}{\pi} m T_s \left[\cos\frac{(k-1)\pi}{3}\sin\theta - \sin\frac{(k-1)\pi}{3}\cos\theta \right] \quad (3)$$
$$T_0 = T_s - (T_a + T_b)$$

where θ is the angle between V_a and $\bar{V}_{ref}$ and m is the modulation index. This is the generic solution for each sector in the plane. θ always varies in the interval (0-60°). By varying the modulation index m and θ, the output voltage amplitude and frequency can be varied. The modulation index controls the output voltage magnitude and θ controls the frequency of the output voltage.

where:

m - Modulation index

θ - Vector Angle

When equations for T_a, T_b and T_o are implemented in a Xilinx FPGA XC3S-400E, the compare register's values corresponding to T_a, T_b and T_o have to be calculated and loaded to generate the SVPWM for a three phase voltage source inverter. This results in 3 phase voltage output from a voltage inverter using space vector pulse width modulation.

The output voltage for various switching patterns can be obtained by drawing the equivalent circuit shown in Fig. 3. As an example consider space voltage V_6 (state 101); the output voltages V_{an}, V_{bn} and V_{cn} are given by an equation as follows:

$$Z_{eq} = \frac{3z}{2}$$
$$V_{an} = V_{cn} = \frac{V_{dc}\frac{z}{2}}{3\frac{z}{2}} = \frac{V_{dc}}{3} \quad (4)$$
$$V_{bn} = \frac{V_{dc} z}{3\frac{z}{2}} = \frac{2V_{dc}}{3}$$

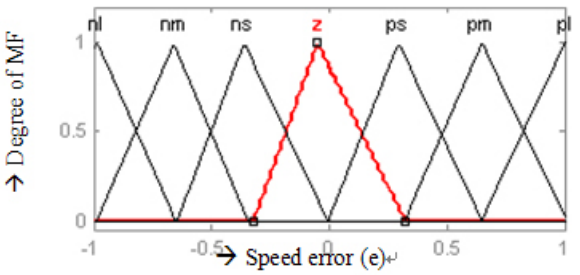
Where V_{dc} is the DC link voltage. Similarly we can calculate the values of V_{an}, V_{bn} and V_{cn} for the other five non-zero states. This is tabulated in table I.

III. Fuzzy Logic Control

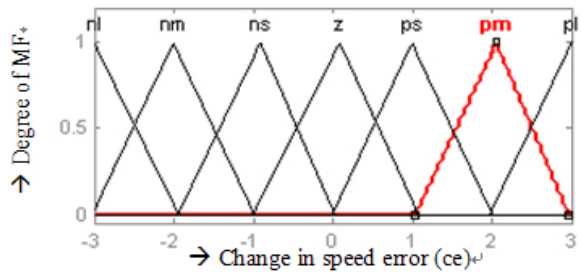
The process of fuzzy logic controller design includes the following steps. (i) Fuzzification: the process of representing inputs as suitable linguistic variables. (ii) Decision Making: the appropriate control action to be carried out needs to be based on knowledge. (iii) Defuzzification: the process of converting fuzzified outputs into crisp values.

TABLE I
Summary of Inverter Switching States

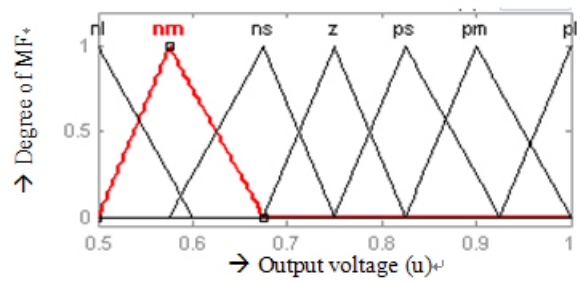
Name	C	B	A	V_{An}	V_{Bn}	V_{Cn}
V_0	0	0	0	0	0	0
V_1	0	0	1	$2Vdc/3$	$-Vdc/3$	$-Vdc/3$
V_2	0	1	1	$Vdc/3$	$Vdc/3$	$-2Vdc/3$
V_3	0	1	0	$-Vdc/3$	$2Vdc/3$	$-Vdc/3$
V_4	1	1	0	$-2Vdc/3$	$Vdc/3$	$Vdc/3$
V_5	1	0	0	$-Vdc/3$	$-Vdc/3$	$2Vdc/3$
V_6	1	0	1	$Vdc/3$	$-2Vdc/3$	$Vdc/3$
V_7	1	1	1	0	0	0



(a) MF for speed error



(b) MF for change in speed error



(c) MF for output voltage

Fig. 4. Membership functions of input and output variables.

A fuzzy logic controller initially converts the crisp errors and changes in error variables into fuzzy variables. Then they are mapped into linguistic labels. Membership functions are associated with each label as shown in the Fig. 4(a)-(c).

The linguistic labels are divided into seven groups: nl-negative large, nm-negative medium, ns-negative small, z-zero, ps-positive small, pm-positive medium, pl-positive large. Each of the inputs and output contains membership functions with all seven linguistics.

The mapping of the fuzzy inputs into the required output is derived with the help of a rule base as given in Table 2.

IV. FPGA Controller

A field programmable gate array is made up of digital integrated circuits that can be programmed to do any type of

TABLE II
RULE BASED SPEED CONTROL

ce \ e	nl	nm	ns	z	ps	pm	pl
nl	nl	nl	nl	nl	nm	ns	z
nm	nl	nl	nl	nm	ns	z	ps
ns	nl	nl	nm	ns	z	ps	pm
z	nl	nm	ns	z	ps	pm	pl
ps	nm	ns	z	ps	pm	pl	pl
pm	ns	z	ps	pm	pl	pl	pl
pl	z	ps	pm	pl	pl	pl	pl

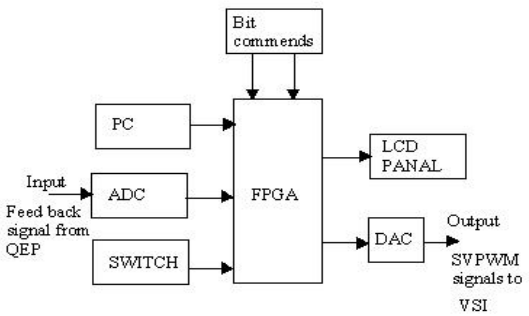


Fig. 5. Functional Block Diagram of the Controller.

digital function. There are two main advantages of an FPGA over a microprocessor chip for a controller:

1) An FPGA has the ability to operate faster than a microprocessor chip.
2) The new FPGAs that are on the market will support hardware that is upwards of one million gates. A FPGA consists of three major configurable elements. These are:
   - Configurable logic blocks (CLBs) arranged in an array that provides the functional elements and implements most of the logic in an FPGA.
   - Input-output blocks (IOBs) that provide the interface between the package pins and the internal signal lines.
   - Programmable interconnect resources that provide the routing path to connect the inputs and outputs of the CLBs and IOBs onto the appropriate network.

The real time control system is implemented using a FPGA XC3 400E from Xilinx, Inc.

A functional block diagram of the controller is shown in Fig. 5.

The controller contains:

- 16×2 LCD display
- PLL oscillator
- Clock source selector
- Xilinx SPARTAN 3 FPGA1
- JTAG connecter for the FPGA1
- XCF02S Flash PROM
- 26 Pin I/O Termination
- 16 Nos of LED
- 34 Pin I/O Termination
- Level translators
- Xilinx SPARTAN 3 FPGA2

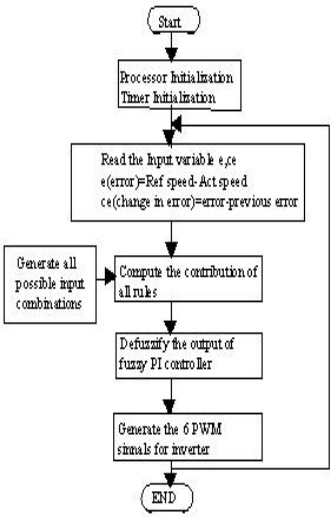


Fig. 6. Flow chart.

- 4Nos of ADC (AD 7266)
- JTAG connecter for the FPGA2
- 4×2 micro switches
- AD 5328 (DAC)

A flow-chart of the software is shown in Fig.6. The set of SVPWM signals is stored in a file with an acceptable format for the development system of the FPGA.

A full description of the fuzzy controller is fed to the program. The description include the number of input variables, the dimensions of the universe of discourse of the input variables, the dimensions of the universe of discourse of the output variables, definitions of the fuzzy linguistic terms used, the number of rules and the rules. To facilitate the use of the software, a skeleton of the input files is formed by the software program. Once the skeleton of the input file is formed, the data can be entered. Changes, such as modifications in the definition of the fuzzy terms, can be made with relative ease and the program can be re-executed to get better performance, if necessary.

The Xilinx ISE Foundation computer-aided-design tool is used for the design and development of the FPGA. The FPGA design flow for the system is given as follows: first, the system is implemented using the Xilinx ISE Foundation tools and simulated at the register transfer level to verify the correctness of the design. Then by using the Xilinx ISE Foundation tools, the logic synthesis is carried out to optimize the design, and the placement and routing are carried out automatically to generate the FPGA implementation file. Finally, the generated implementation file is downloaded to the FPGA development board for testing.

V. IMPLEMENTATION AND RESULTS

This paper intensifies three phase induction motor drive based FPGA controllers. The constant flux technique (V/F) is used for induction motor speed control. As a test of the proposed FPGA controller performance a driver is implemented.

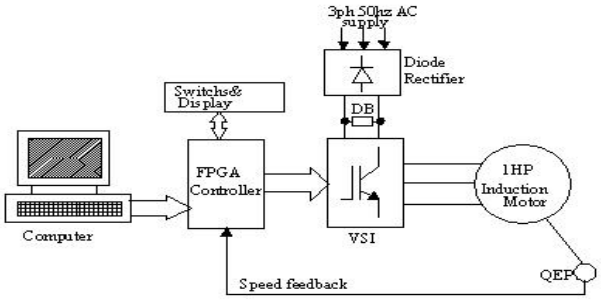


Fig. 7. Block Diagram of Implemented Drive.

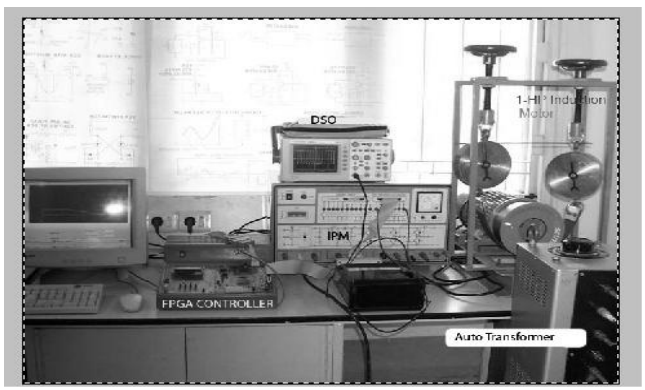


Fig. 8. Experimental setup.

A. Block diagram of the implemented circuit

A block diagram of the implemented drive is shown in Fig.7.

- The block diagram consists of a FPGA Controller, a quadrature encoder pulse (QEP), an A/D Card, a voltage source inverter (VSI) and a 1HP induction motor.
- A voltage source inverter with the specification of 1200V and 25A IGBT is being used as the switching devices in the experiment.
- Current and voltage are sensed by using the Hall Effect principle. The speed of the induction motor is sensed by a quadrature encoder pulse (QEP).
- The output of the speed sensor is fed to the frequency and the voltage converter circuit. In this module set the frequency to the voltage converter with a maximum output of 2.5V at a rated motor speed of 1500rpm.
- Speed error is calculated with a comparison between the set speed and the actual speed. Speed error is given as input to the FPGA XC3S-400E Controller.
- The FPGA controller gives the gate drive signal to the voltage sources inverter there by stabilizing the speed of the induction motor using the constant flux principle.

B. Evaluation and experimental results

Fig.8 shows a experimental setup diagram.
Working principle:

1) A three phase AC supply is given to the experimental setup diagram.

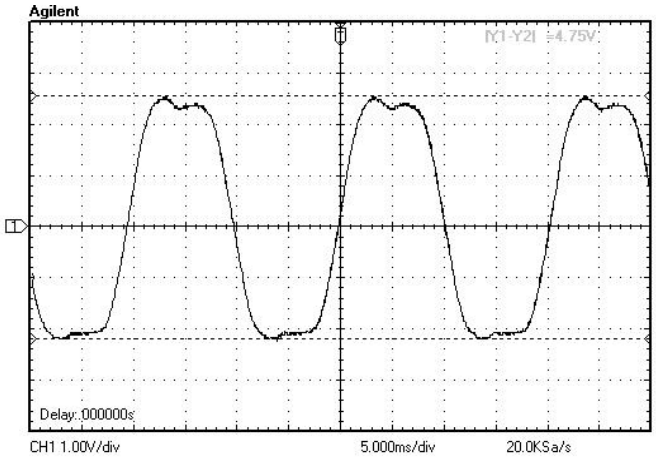


Fig. 9. SVPWM Waveform.

2) Download the program with the Fuzzy PI algorithm and space vector pulse width modulation technique from a PC to the FPGA controller through a data cable.
3) The actual speed is sensed by the QEP sensor and the analog speed is converted to a digital pulse.
4) The set speed is set in the FPGA controller.
5) The converted digital pulse is given to the FPGA controller.
6) The actual speed is compared with the set speed and in accordance with the error the Fuzz PI controller is tuned and then FPGA produces the space vector pulses.
7) SVPWM is given to voltage source inverter and to the induction motor through the cable.
8) For various load conditioned speeds the Vs time graph is monitored and plotted using visual basic software.

The testing is divided into two steps; in the first step, the basic functionality is tested and the SVPWM waveform is generated. In the second step, the speed control of the induction motor is verified to test the performance of the system. The FPGA generated SVPWM waveform is shown in Fig. 9. The switching frequency is set to 10 kHz, the gating on-time $T_1 = T_2 = 0.75$ (pu), To = 0.5 (pu) and the modulation index ma = 0.907 (pu).

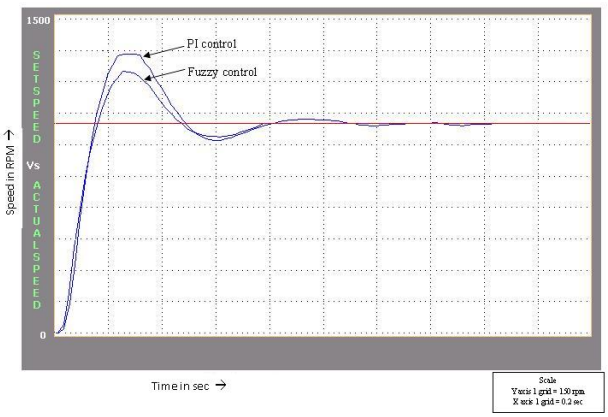
Driver responses for 1000rpm and 1200rpm reference speeds with no load condition have been represented respectively in Fig. 10(a) and 10(b). The results show that a classical PI controller reaches settling time in 1.6 sec, but that a fuzzy PI controller reaches the settling time in 0.7 sec. The steady state error is %5 for each response.

Another test for the loaded condition is shown in Fig. 11(a) and 11(b). At the same set speed a classical PI controller reaches settling time in 1.65 sec, but the fuzzy PI controller reaches the settling time in 0.85 sec. With a change in load the motor speed has not changed much and the speed regulation is nearly suitable. Table 3 summarizes the speed response of the motor with different load conditions.

C. Synthesis Report of the Controller

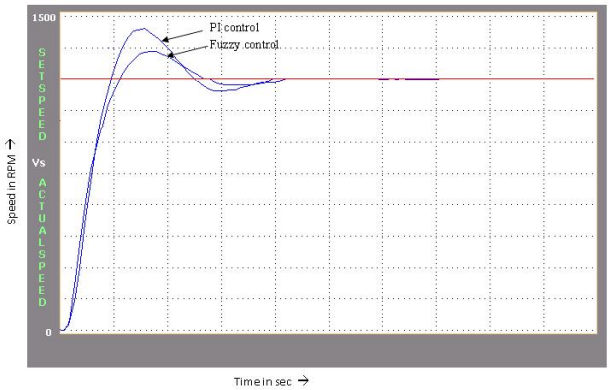
All the modules are integrated and synthesized using the Xilinx project navigator and support tools. The synthesized

(a) Driver Responses for 1000 rpm (No load)

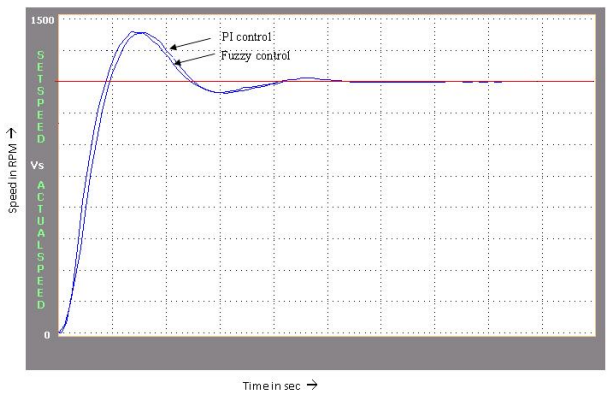


(b) Driver Responses for 1200 rpm (No load)

Fig. 10. Driver responses for No load conditions.



(a) Driver Responses for 1000 rpm (Full load)



(b) Driver Responses for 1200 rpm (Full load)

Fig. 11. Driver responses for Full load conditions.

TABLE III
SUMMARIZES SPEED RESPONSE OF THE MOTOR WITH DIFFERENT LOAD CONDITION

Load condition	Classical PI	Fuzzy PI
	Control (in sec)	Control (in sec)
1000 rpm (No load)	1.6	0.7
1000 rpm (Load)	1.65	0.79
1200 rpm (No load)	1.6	0.79
1200 rpm (Load)	1.65	0.85

VHDL source code is placed and routed. Finally, a bit file is created. This file is fused into the Xilinx XC 3S 400E–4PQ208 FPGA and interfaced with the input and output devices. 68% 4 LUTs, 88% flip-flop slices and 28% bonded IOBs were utilized as shown in Table 4.

TABLE IV
SYNTHESIS REPORT OF THE CONTROLLER

Number of Slices	3185 out of 3584	88%
Number of Slice Flip Flops	1071 out of 7168	14%
Number of 4 input LUTs	4890 out of 7168	68%
Number of Bonded IOBs	40 out of 141	28%
Number of MULTI8X18s	14 out of 16	87%
Number of GCLKs	2 out of 8	25%

VI. CONCLUSIONS

This paper demonstrates that the speed control of induction motor systems can be realized by using a fuzzy PI controller. The control scheme was modeled and designed in VHDL. It was simulated and synthesized using the Xilinx Foundation package and implemented into a Xilinx XC3S400 FPGA. The experiments show that the dynamic response of a system using the proposed controller is better when compared to a classical PI controller. Finally, the proposed method provides induction motors with suitable speed regulation.

ACKNOWLEDGMENT

The authors wish to thank the Management, the Principal and the Department of Electrical and Electronics Engineering at the Sona College of Technology, Salem, Tamil Nadu for providing the required facilities to successfully complete this research.

REFERENCES

[1] Bimal K. Bose, "Modern Power Electronics and AC Drives," Pearson education.
[2] Leonhand.W, "Control of Electrical Drives," Springer Verlag, 1990.
[3] Wei-Feng Zhang and Yue-Hui Yu, "Comparison of Three SVPWM Strategies," *Journal of electronic science and technology of china*, No. 3, pp. 283-287, Sep. 2007.

[4] Iqbal.A, "Analysis of space vector pulse width modulation for a five phase voltage source inverter," *IE (I) journal-EL*, Vol. 89, Issue 3, pp. 8-15, Sep. 2008.
[5] Mondal, S.K.; Bose, B.K.; Oleschuk, V.; Pinto, J.O.P, "Space vector pulse width modulation of three-level inverter extending operation into overmodulation region," *IEEE Transactions on Power Electronics*, Vol. 18, Issue 2, pp. 604 – 611, Mar. 2003.
[6] Avnesh verma, sunil Dhingra, "Design and synthesis of FPGA for speed control of Induction motor," *International Journal of physical science*, Vol. 4 (11), pp.645-650, Nov. 2009.
[7] Peter Vas, "Artificial –Intelligence based electrical machines and drives," Oxford university, 1999.
[8] Chuen Chien Lee, "Fuzzy Logic in Control Systems: Fuzzy Logic controller – Part1,2," IEEE, 1990.
[9] R.Arulmozhiyal and Baskaran.K, "Space Vector Pulse Width Modulation Based Speed Control of Induction Motor Using Fuzzy PI Controller," *International Journal of Computer and Electrical Engineering*, Vol. 1, No. 1, pp. 98-103.
[10] Alfredo Munoz –Garcia, Thomas .A.Lipo, Donald. W.Novotony, "A New Induction motor V/f control method capable of High performance Regulation at low speeds," *IEEE Transaction on Industry Applications*, Vol. 34, No. 4, pp. 813-821, 1998.
[11] Gupta.A, "Simulation of Variable speed squirrel case Induction motor constant Volts/Hertz operation," *IEEE International on Electric Machines and Drives conference Record*, Pp TB3/8:1-TB3 18.3, 1997.
[12] Tianchai Suksri ans Satean Tunyasrirut "T- DOF PI Controller Design for a speed control of Induction motor," *in Proc. of world Academy of Science, Engineering and Technology*, Volume No. 26, pp 421- 425, 2007.
[13] Mahmoud.A.Manzoal and D.Jayabharathi, "FPGA for fuzzy controllers," *IEEE Transations on systems, Man and cybanetic*, Vol. 25, No. 1, Jan. 1995.
[14] Vranda parkhi, "FPGA Implementation of PWM control Technique for three phase Induction motor Drive," *First International conference on Emerging trends in Engineering and Technology*, DOI 10.11091 ICETET.2008.115, pp 996-1001.
[15] Perry.D , "VHDL," McGraw Hill, 1998.
[16] Navabi.Z "VHDL Analysis And modeling of digital systems", McGraw Hill, 1998.
[17] Muthuramalingam.A, Himavathi.S, "Performance Evaluation of a Neural Network based General propose space vector modulator," *International Journal of Electrical, computer and systems Engineering*, No.1, pp. 19-26, 2007.
[18] Arulmozhiyal.R and Baskaran.K "A Novel Approach to Induction Motor speed control using FPGA," *International Journal of Applied Engineering Research*, ISSN 0973-4562, Vol. 4, No. 1, pp 1-14, 2009.

R. Arulmozhiyal was born in Chennai, Tamil Nadu, India in 1973. She received B.E. and M.E. degrees in Electrical Engineering from the University of the Madras, Anna, India, in 1999 and 2006, respectively. Since 1999, She has been with Department of Electrical and Electronics Engineering, Faculty of Engineering, Sona College of Technology, Tamil Nadu, India where she is currently a Senior Lecturer. Her research interests are in the areas of AC motor control and AI techniques for solid state drives. Mrs. Arulmozhiyal is a member of the IEEE and a life member in ISTE.

K. Baskaran was born in Thanjavur, Tamil Nadu, India, in 1964. He received B.E. and M.E. degrees in Electrical Engineering from the University of the Annamalai, Bharathiyar, Tamil Nadu, India, in 1985 and 1995, respectively, and a Ph.D. in Computer Science Engineering from Anna University, Chennai, India, in 2004. Since 1989, he has been with the Department of Electrical and Electronics Engineering, Faculty of Engineering, Government College of Technology, Coimbatore, India, where he is currently an Assistant Professor. His research interests are in the areas of networking and soft computing. Prof. Baskaran is a member of the IEEE.