Journal of Engineering Research **Vol. 6 No. (2) June 2018 pp.** *138-154*

Using system generator to design a hardware implementation of a fault-tolerant control of induction motor for electrical vehicle

B. Souha*, G. Soufien and A. Mtibaa

Laboratory of Electronics and Microelectronics, Faculty of Sciences, University of Monastir, Tunisia

**Correspondent Author: boukadidasouha@gmail.com*

ABSTRACT

In this paper, a design method of Active Fault Tolerant Control (AFTC) of an Induction Machine (IM) is proposed. AFTC ensures continuous operation of IM in case of faulty current sensor. The proposed system is illustrated using two controls. Direct Torque Control based Space Vector Modulated (DTC-SVM) is used when the system is operating in healthy mode; the adopted DTC-SVM requires the use of three current sensors. Speed Control with Slip Regulation (SCSR) is used when the system is operating in failed mode; the adopted SCSR does not require any current sensors. Using a microcontroller or a Digital Signal Processor, the implemented algorithm suffers from a calculating delay. Therefore, to solve this difficulty, Field Programmable Gate Array (FPGA) is required, owing to its fast processing speed. Several new design flows and tools are developed to implement an algorithm on FPGA, such as Xilinx System Generator (XSG). XSG is a high-level block-based design tool that offers bit and cycle accurate simulation. This tool can automatically generate the Very High-Density Logic (VHDL) code without resorting to a tough programming, without being obliged to do approximations, and, furthermore, we can visualize the behavior of the machine before implementation.

Keywords: FPGA; fault tolerant control; induction motor; XSG.

INTRODUCTION

Over the past few years, research interest in implementation on FPGA has grown considerably. The use of FPGA reduces the execution time, through the adoption of a parallel processing. The FPGA cost decreases. For this reason, the substitution of microcontrollers with FPGA is a new tendency. Therefore, the use of FPGA means a gain in matters of cost and time. For example, the cost of Spartan-3A DSP is $30; it can deliver 20 billion multiply-accumulates per second. In comparison, with $30, we can buy a 600 MHz C64x DSP that delivers 2.4 billion multiply-accumulates per second.

Over the past few years, several researchers use FPGA to control electrical system (Hasanien et al., 2001; Dagbagi et al., 2012; Monmasson et al., 2012; Shahbazi et al., 2013; Jezernik et al., 2013). VHDL hardware description language is used to develop the algorithm. In this paper, to implement the AFTC of an IM on the FPGA, we use XSG toolbox developed by Xilinx and added to Matlab/Simulink. The XSG advantages are the real-time, rapid time to market, and portability. Once the design and simulation of the proposed algorithm are completed, we can automatically generate the VHDL code in Xilinx ISE.

FTC of electrical drives was a very active research field for many research groups (Tabbache et al., 2013; Bruccoleri et al., 2003; Isermann et al., 2002; Dumont et al., 2006; Ebrahimi Bavil et

al., 2015; Ghachem et al., 2011; Boussaid et al., 2013, Hamdi et al., 2013). The FTC aims to ensure the continuous system functionality, even after faults occurrence. This allows increasing system availability and reliability. Different types of failures can occur in controlled electrical drives: IM, power converters, connectors, and sensors. The electric motor failures can have various origins such as failures related to the exploitation leading to faults or premature degradation, and failures related to wrong weak dimensioning and design causing premature degradation. Current sensors are widely used in controls of electrical drives. Faults in the current measurement chains have been treated for various electrical drives. Researches in this field initially focused on the effect of current sensor fault and the development of Fault Detection and Isolation (FDI) methods (Bibhrajit et al., 2013; Halder et al., 2006). In general, FDI methods utilize the concept of redundancy, which can be either a hardware redundancy or analytical redundancy. Sensor fault is identified and detected using analytical redundancy as described in Ghachem et al. (2011); Boussaid et al. (2013); Hamdi et al. (2013). In Ghachem et al. (2011), a robust nonlinear analytical redundancy technique is presented to detect and isolate sensor/actuator faults in a mobile robot. The most used FDI methods are based model methods, like parity space approach (Berriri et al., 2011) or observatory method (Saravanan et al., 2013; Balasubramanian et al., 2013). Easy and fast sensor FDI algorithm for IM is presented in Berriri et al. (2011); the proposed FDI algorithm is based on parity space approach. However, the analytical model could be imprecise, so the residual could indicate a false alarm. For this reason, analytical models are restricted for deterministic models. To solve this problem, intelligent techniques (Artificial Neural Network (ANN) and Fuzzy Logic (FL)) are used (Li et al., 2012; Cheol Cho et al., 2010; Adouni et al., 2013). This category has many advantages, such as uncertainty, complexity, and disturbance system modeling.

The detection and location of faults are essential but not sufficient to ensure the safety and operation in degraded mode. Researchers now are more concerned with FTC strategies based on reconfigurable controls in order to guarantee continuous operation of the system, even with a sensor failure. FTC system design techniques can be classified into two types: Passive Fault Tolerant Control (PFTC) such as robust fault accommodation approach (Ghachem et al., 2011) and AFTC approach such as Projection Based Off-line and Hardware redundancy (Boussaid et al., 2013; Hamdi et al., 2013). In PFTC, the controller structure can tolerate only a limited number of faults. These faults are assumed to be known. To overcome this problem, AFTC should be employed. The AFTC aims to ensure the continuous system operation, even after the fault occurrence.

In this paper, the proposed AFTC system is illustrated using two controls. DTC-SVM is used when the system is operating in healthy mode. SCSR is used when the system is operating in failed mode. The contribution of this paper is to develop a new method to implement an easy and simple AFTC based intelligent technique.

DESIGN METHODOLOGY FOR IMPLEMENTATION ON FPGA

The design developed in this paper was performed according to an appropriate methodology. XSG uses the Xilinx ISE design suite. This tool can generate the VHDL code without resorting to a difficult and tough programming. The VHDL code can then be used as a stand-alone design or integrated with other designs. Therefore, the implementation of any design becomes easier. Unlike the VHDL languages, XSG provides a design interface based on a model using an extended

Matlab\Simulink library. The XSG library consists of a set of hardware blocks that can carry out complex operates, such as Interpolation filter, Cordic (divider, tan, sin, cos, and Log), FFT, and FIR filter design. The new methodology XSG consists of a set of steps and roles that offer considerable hardware design advantages as illustrated in Figure 1.

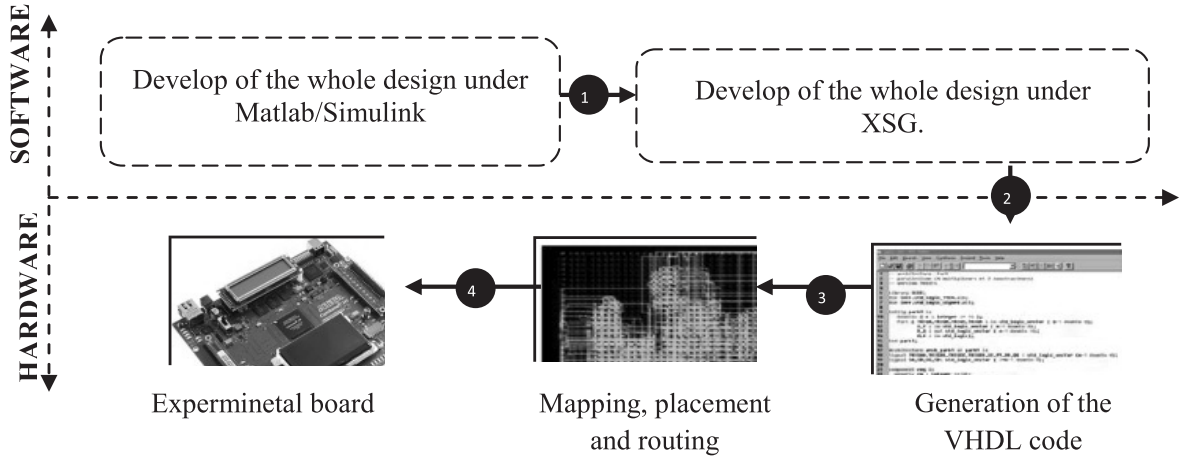


Figure 1. Step for implementation on FPGA.

XSG provides a hardware co-simulation that enables the incorporation of a design running on an FPGA. Using a co-simulation approach, the simulation of the whole system is simulated using a single simulator. It is an answer to growing simulation needs as systems get increasingly complex.

MODEL AND IMPACT OF CURRENT SENSOR FAULT

The control of the electrical vehicle depends on the quality and the availability of sensor measurements. However, if a sensor is affected by a fault, the measurements become interrupted. Therefore, the controller cannot provide the correct response for the electric vehicle propulsion. AFTC is necessary to overcome this problem. The drive is simulated with offset fault. The system is stable with good performance before the fault occurrence, although it is clear that the controller cannot operate in the presence of faults. Figure 2 presents the temporal evolution of stator current, speed, and electromagnetic torque.

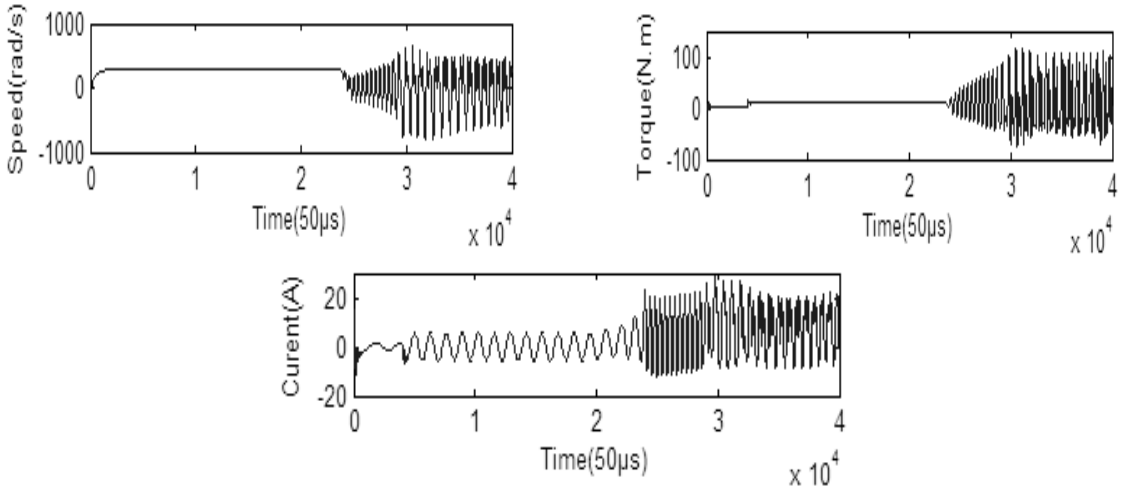


Figure 2. Simulation results with offset fault at t=1s.

FAULT DETECTION AND ISOLATION UNIT DESIGN

In this section, a novel current sensor FDI unit of IM drive systems is presented. The important ideas are presented here. In order to obtain high performance motor drives, modern control strategies like DTC-SVM should be employed. The objective of SVM is to obtain the demanded output voltage, by instantaneously combination of the switching states corresponding to the basic space vectors. Figure 3 provides the global configuration of a DTC-SVM scheme. There are three different loops corresponding to the electromagnetic torque, stator flux, and angle. The error between the estimated stator flux magnitude φ_s and the reference stator flux magnitude φs* and the error between the estimated torque T and the reference torque T* are the inputs of two PI controllers. The outputs of the regulator are used as inputs of direct axis voltage. The (d, q) axis voltages are converted into amplitude of stator voltage. The voltages (V_d, V_q) and angle Θ are used as reference signals in the SVM approach.

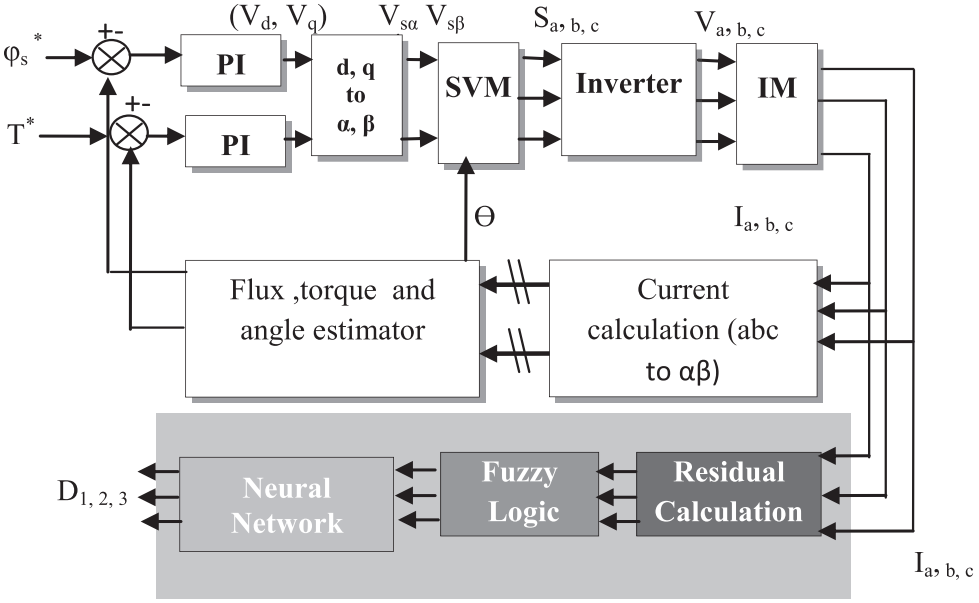


Figure 3. Proposed FDI and DTC-SVM approach.

The stator voltage and stator current are calculated from the state of three phases (S_a, Sb, Sc) and measured currents (i_a, i_b, i_c).

$$V_s(S_a,S_b,S_c)=\sqrt{\frac{2}{3}}E(S_a+S_be^{j\frac{2\pi}{3}}+S_ce^{j\frac{4\pi}{3}})$$
$$i_s(S_a,S_b,S_c)=\frac{2}{3}(i_a+i_be^{j\frac{2\pi}{3}}+i_ce^{j\frac{4\pi}{3}}) \quad (1)$$

Phase angle and stator flux amplitude are calculated using equations (2) and (3), respectively:

$$\theta_s=\tan^{-1}\left(\frac{\phi_{s\alpha}}{\phi_{s\beta}}\right) \quad (2)$$

$$\phi_s=\sqrt{\phi_{s\alpha}^2+\phi_{s\beta}^2} \quad (3)$$

The developed electromagnetic torque T of the IM can be expressed by equation (4):

$$T = \frac{3}{2} p \left(i_{s\beta} \phi_{s\alpha} - i_{s\alpha} \phi_{s\beta} \right) \tag{4}$$

Expressing the voltage vector in the graduation (α, β), we have

$$\overrightarrow{V_s} = V_{s\alpha} + jV_{s\beta} = \frac{T_1}{T_{\text{mod}}} \overrightarrow{V_1} + \frac{T_2}{T_{\text{mod}}} \overrightarrow{V_2} \tag{5}$$

where T_{mod} is the switching period, T_1 is the time of application of V_1, and T_2 is the time of application of V_2.

DTC-SVM is inherently dependent on current sensors that should operate properly. To support the proper operation of the system in the case of a current sensor fault occurrence, we need to detect, to locate, and to isolate the failure. FDI system is designed as a hybrid system, in which Fuzzy Logic (FL) system and Artificial Neural network (ANN) may cooperate and interact to implement efficiently the required FDI tasks. The FDI scheme is divided into three steps. The first step is based on generating the residual signal. The residual signal represents the difference between the current value estimated and the current value calculated. If the residual is equal to zero, the system operates in healthy case; otherwise, if the residual is different than zero, then the system is affected by a fault. The second step consists of specifying the current sensors attached by the fault; it is based on the FL system. The role of the residual fuzzification step is to convert analog inputs into fuzzy variables. The FL is designed to have three inputs and three outputs. The inputs are the residues of each current sensor (Ra, Rb, Rc). The outputs are the fuzzy variables (μ_{Ra}, μ_{Rb}, μ_{Rc}). The fuzzification step is realized using three fuzzified units. The linguistic variables are defined by the following Membership Functions (MFs): ‘‘N: negative residual with trapezoidal MF”, ‘‘Z: zero residual with triangular MF”, and ‘‘P: positive residual with trapezoidal MF”. Each residual (R_a, R_b, R_c) could be described with three memberships (two trapezoidal and one triangular). The fuzzy set comprises a MF, which could be defined by parameters. For the choice of parameters, many tests are affected. The trapezoidal and triangular MF are calculated using the following equations, respectively:

$$f(x,a,b,c,d) = \begin{cases} 0 & si \quad x \le a \\ \dfrac{x-a}{b-a} & si \quad a \le x \le b \\ 1 & si \quad b \le x \le c \\ \dfrac{d-x}{d-c} & si \quad a \le x \le b \\ 1 & si \quad d \le x \end{cases} \tag{6}$$

$$f(x,a,b,c,d) = \begin{cases} 0 & si \quad x \le a \\ \dfrac{x-a}{b-a} & si \quad a \le x \le b \\ \dfrac{c-x}{c-b} & si \quad b \le x \le c \\ 0 & si \quad c \le x \end{cases} \tag{7}$$

where a, b, c, d are the parameters, f is the output, and x is the input.

Using VHDL language, to find a membership value from an MF, there are two solutions: computation oriented approaches and memory oriented approaches. Using these situations, it is difficult to quickly change the MF characteristics, and you should require a significant amount of memory resources. According to the advantages of XSG, these equations are implemented in the hardware by only using comparators, substractors, and multipliers. The trapezoidal MF of the residue Ra is illustrated in Figure 4.

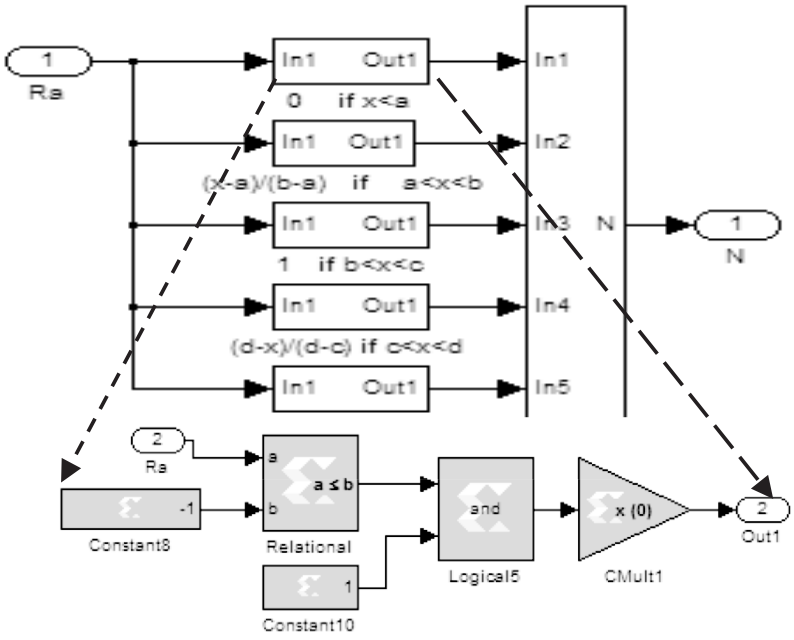


Figure 4. Implementation of trapezoidal MF.

The third step is based on ANN. The ANN, shown in Figure 5, is trained by a learning algorithm, which carries out the adaptation of the weight and the bias. The inputs of the network are the fuzzy variables (μ_{Ra}, μ_{Rb}, μ_{Rc}). As outputs, we have the decisions (D_1, D_2, and D_3).

The mathematical model of a neuron is given in equation (8).

$$D_i = \varphi(\sum_i w_i * \mu_{Ri} + b) \tag{8}$$

where φ is the activation operate (tangent sigmoid operate), (w1, w2… wp) are the corresponding weights, (μRa, μRb, μRc) are the inputs of the neuron, b is the bias of the neuron, and Di is the output signal. Many tests and simulations are used to choose the number of layers and neurons per layer. The neural network that has developed for controlling the IM has three layers. The output of each layer is the input to the next layer. The different input will be spread through the different layers of neurons to the output layer. The ANN consists of nine neurons in the input and three neurons in the output layer.

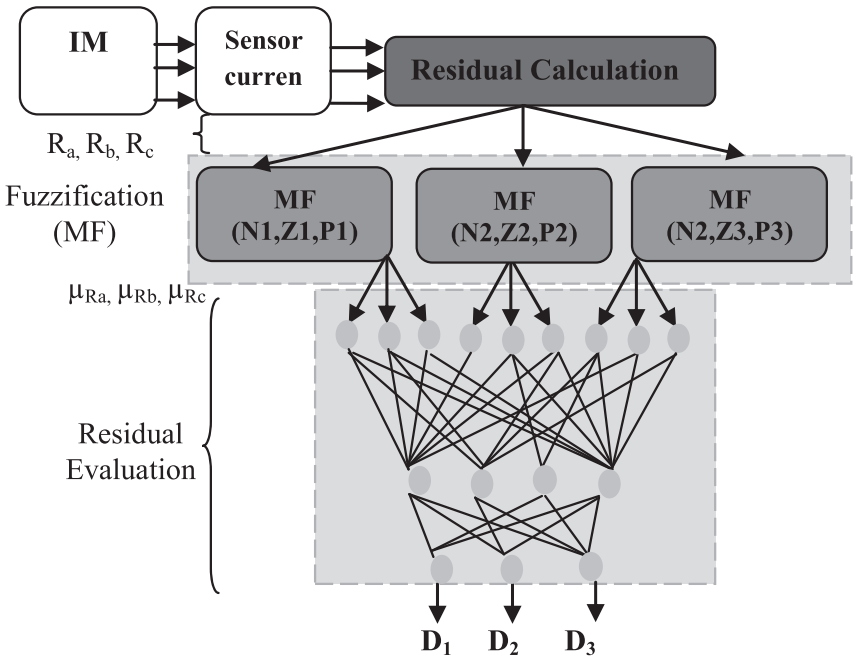


Figure 5. Current sensor FDI unit.

Using the Toolbox in Matlab/Simulink, and after many simulation tests, Table 1 presents the different rules of the ANN. Each control rule from Table 1 can be described using the input variables (μ_{Ra}, μ_{Rb}, μ_{Rc}) and the outputs variables (D_1, D_2, D_3). For example, rule 3 is expressed as follows:

If {μRa is Positive and μ_{Rb} is Positive and μRc is Zero}, then current sensors 1 and 2 are faulty.

Table 1. Inference table.

	SENSOR 1			SENSOR 2			SENSOR 3			DECISION		
N	N_1	Z_1	P_1	N_2	Z_2	P_2	N_3	Z_3	P_3	D_1	D_2	D_3
1	0	0	1	0	0	1	0	0	1	1	1	1
2	0	0	1	0	0	0	0	0	0	1	0	0
3	0	0	0	0	0	1	0	0	0	0	1	0
4	0	0	0	0	0	0	0	0	1	0	0	1
5	0	0	1	0	0	1	0	0	0	1	1	0
6	0	0	0	0	0	1	0	0	1	0	1	1
7	0	0	1	0	0	0	0	0	1	1	0	1
8	0	0	0	0	0	0	0	0	0	0	0	0

ACTIVE FAULT TOLERANT CONTROL

In this paper, an AFTC has been treated. The adopted controller is composed of two parts: FDI and reconfiguration strategy. Figure 6 shows the proposed AFTC system, which is composed of

two control techniques. DTC-SVM is used when the system is operating in healthy mode. SCSR is used when the system is operating in failed mode. We should resort to use SCSR, because the adopted SCSR does not require any current sensors. The proposed flexible architecture for AFTC maintains maximum performance, as well as the overall system failure rate at an acceptable level. With this approach, the reliability of the drive system is greatly improved.

The proposed AFTC approach could be summarized as follows:

- Before sensor failures, the two controllers are used. The DTC-SVM is used as the primary controller, and the SCSR is kept in a standby mode.
- If a current sensor fails, the SCSR becomes the primary controller, and the DTC-SVM is shifted to the standby mode.

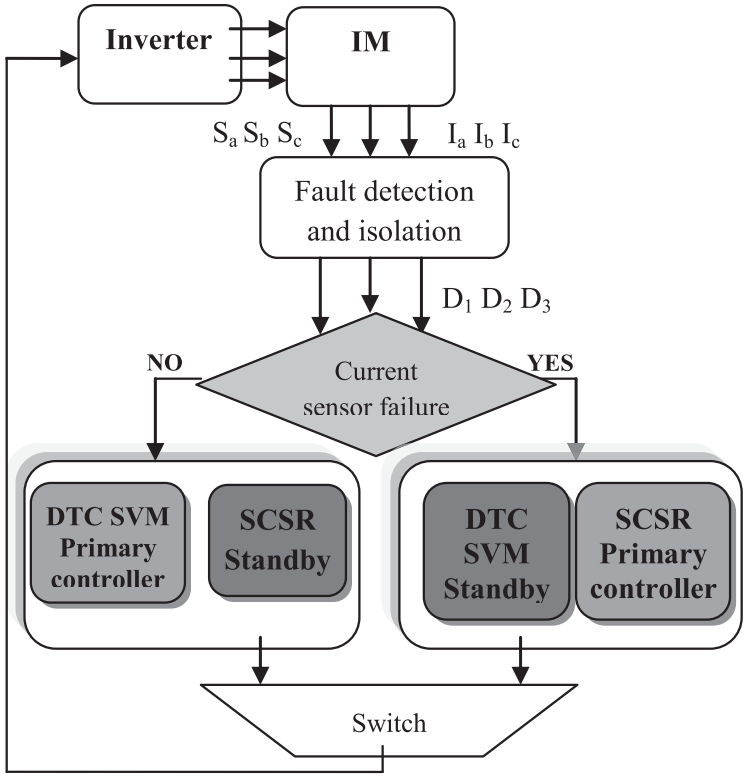
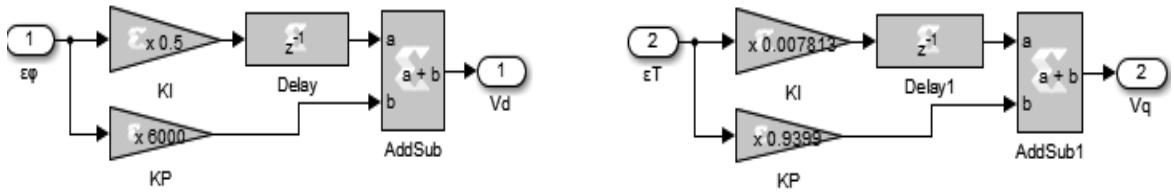


Figure 6. The proposed active FTC system control.

Implementation of DTC-SVM using XSG

The design of the DTC-SVM using the XSG is based on a mathematical model. The design stages of logical operations and required arithmetic are applied in a modular fashion and in hierarchical order. The outputs of the regulator are used as inputs of direct axis voltage. The (d, q) axis voltages are converted into amplitude of stator voltage as shown in Figures 7 and 8. The XSG design of torque and flux estimator is shown in Figure 9.



(a) Calcul of V_d (b) calcul of V_q

Figure 7. Design of the components V_d and V_q.

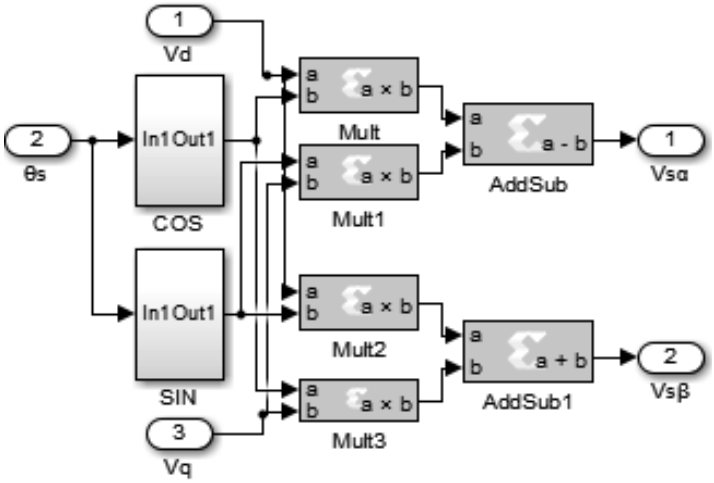


Figure 8. Design of the components $V^{s\alpha}$ and $V^{s\beta}$.

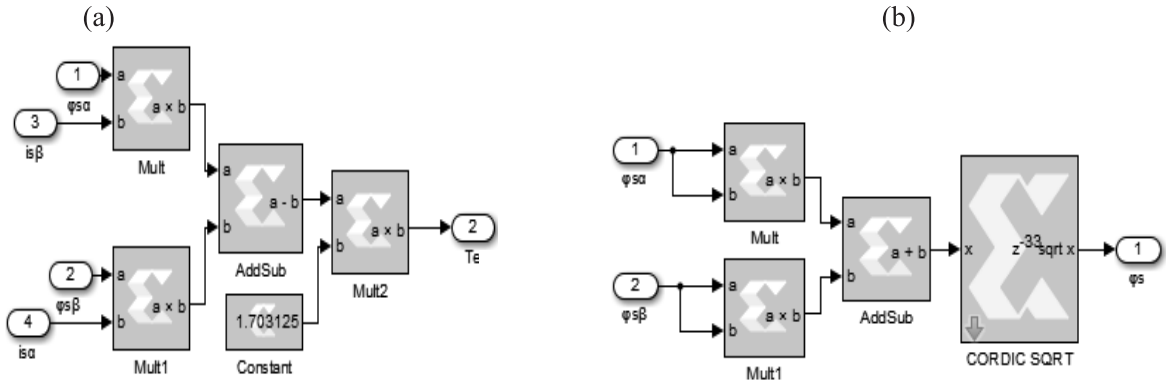


Figure 9. Flux and torque estimator: (a) Calcul of electromagnetic torque T; (b) Calcul of stator flux φ_s.

Implementation of SCSR using XSG

The principle of SCSR is based on the modeling in the permanent regime of IM, as shown in Figure 10. In order to maximize the torque capacities, the flux must be preserved in a range equal to its nominal value.

The SCSR was implemented using comparator, divider, substractors, and multipliers, as shown in Figure 11.

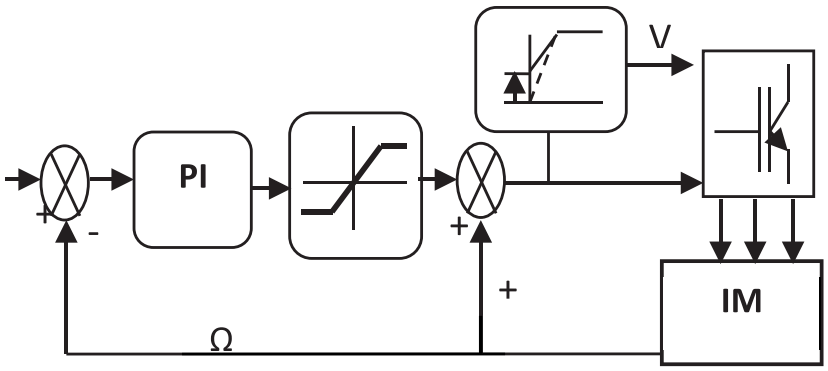


Figure 10. Speed control with slip regulation.

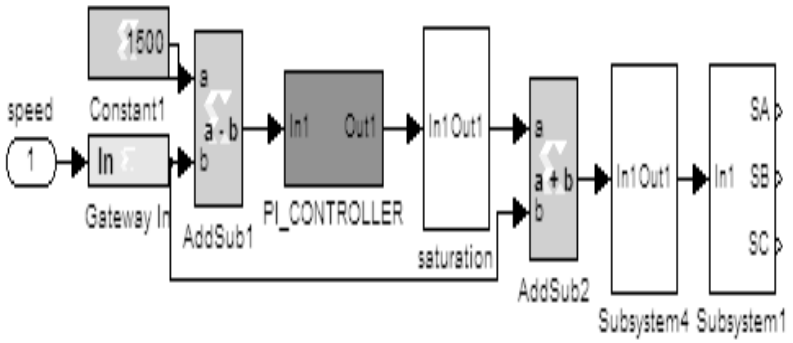


Figure 11. Implementation of SCSR using XSG.

IMPLEMENTATION AND EXPERIMENTAL RESULTS

The experimental setup of the AFTC is represented in Figure 12. It consists of a three-phase inverter, a three-phase IM. Xilinx Virtex-V XC5VFX70T, which includes a 100 MHz oscillator, was used as a target component for the implementation of the AFTC. An Analogical/digital conversion was used for the conversion of the measured currents. The FPGA based hardware control includes a serial interface, the AFTC, an incremental coder interface, and an ADC interface. The results were obtained using the RS232 and plotted within Matlab. Figure 13 presents a photograph of the experimental setup.

Control system's algorithm must be functionally validated before implementation. Results using hardware co-simulation are presented to evaluate the ability of this diagnostic and to detect, isolate, and reconfigure sensor faults in an IM. In order to respect technical constraints of the power inverter, the sampling period is 50 μs. The machine is running at 300 rad/sec. The flux and torque references used are 0.91 Wb and 10 N.m, respectively.

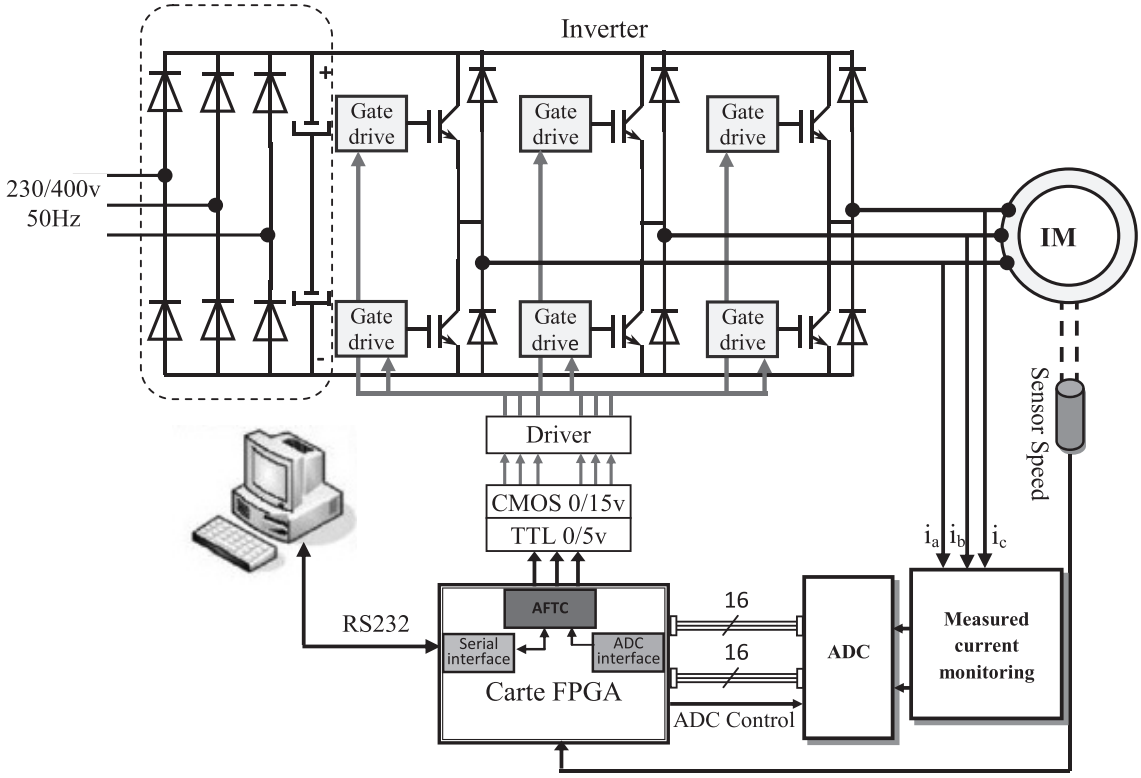


Figure 12. Schematic diagram of the experimental setup.

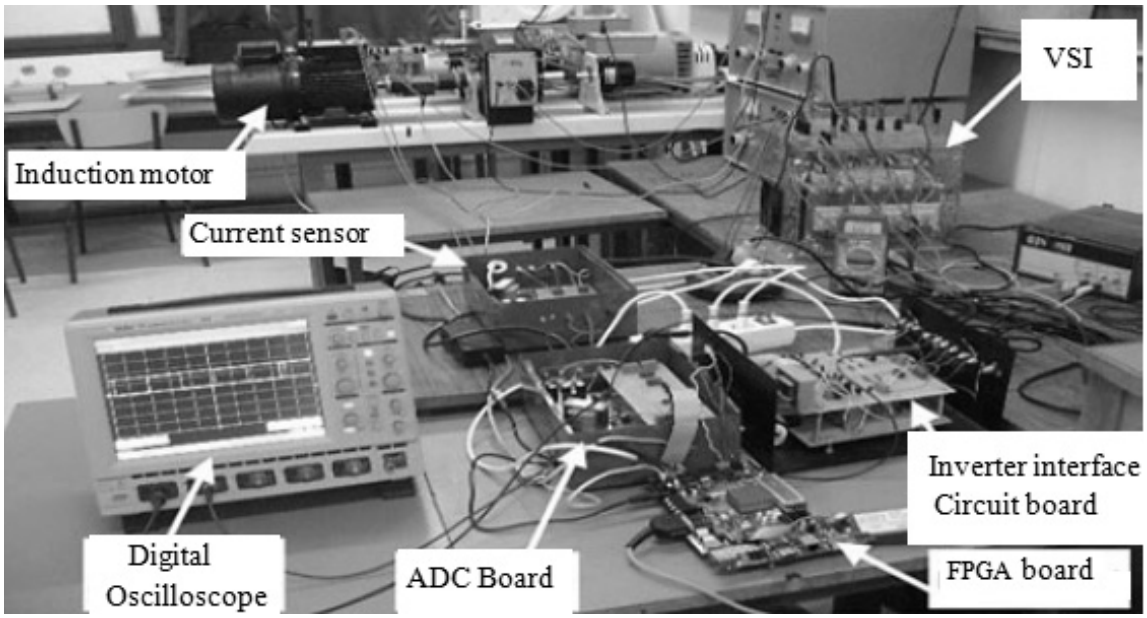


Figure 13. Photograph of the experimental setup.

A serial interface provides serial communication between the FPGA board and the PC through RS232 port. The values of electromagnetic torque, stator flux, rotor speed, and residual are sent to the PC. Graphical User Interface is developed under Visual-Basic. This interface is shown in Figure 14.

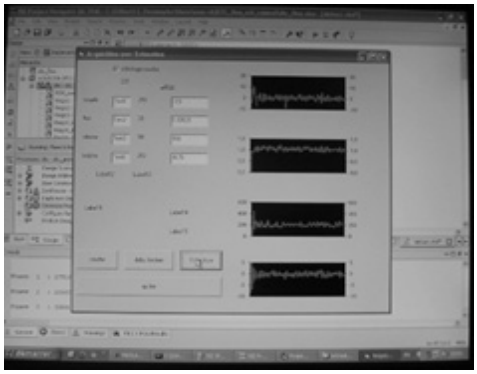


Figure 14. Graphical User Interface of data.

To illustrate the AFTC, two cases are presented:

CASE 1: At t=1.1 s, a fault is injected on sensor Ia. The corresponding speed, torque, flux, decision, and residual are shown in Figure 15. FDI ensures successful detection and isolation of the fault as described in Figure 15. According to the rule summarized in Table 1, the AFTC has been trained to recognize the faults from residual patterns. The SCSR becomes the activated controller, but the DTC SVM is preserved to the standby mode.

CASE 2: A bias type fault is injected on sensors 1 and 2. After the fault occurrence, the SCSR operate, the electromagnetic torque, and speed quickly reach their reference as shown in Figure 16. Smooth and short transients in terms of torque and speed are noticed. Therefore, the effectiveness of the proposed controller is provided. After a fault occurrence, the AFTC is very rapidly reacting to prevent any undesirable event. Therefore, the time duration and the quality of the transition between the two controllers are demonstrated as illustrated in Figures 15 and 16.

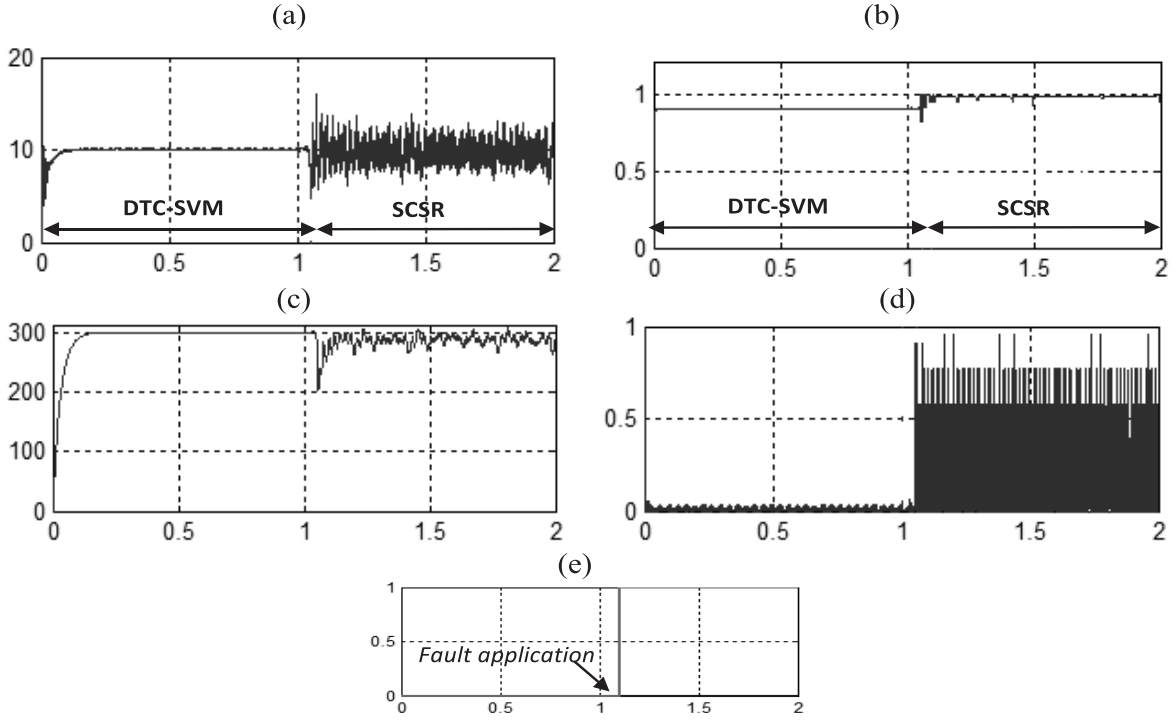


Figure 15. AFTC results with noisy current sensor signal: (a) electromagnetic torque; (b) flux; (c) speed; (d) residual of sensor Ia; (e) decision.

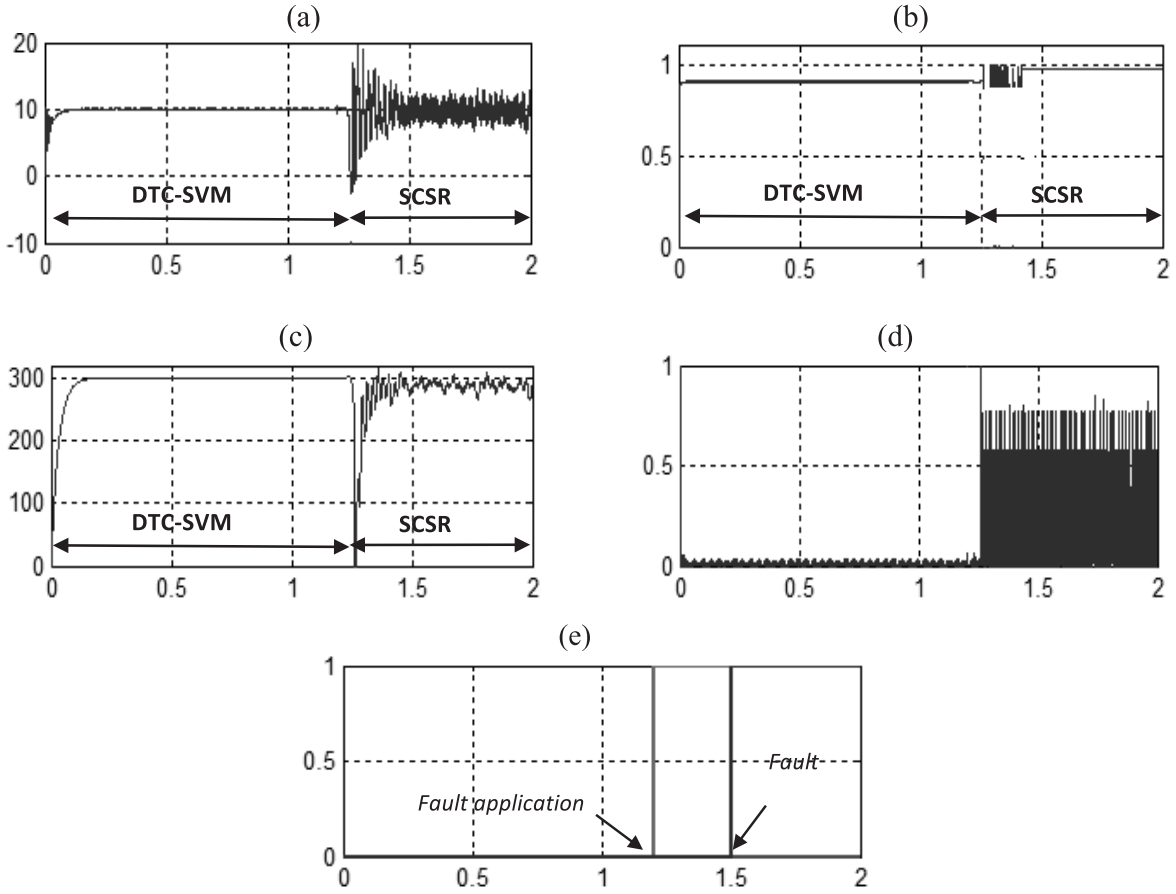


Figure 16. AFTC results with noisy current sensor signal: (a) electromagnetic torque; (b) flux; (c) speed; (d) residual of sensor Ia; (e) decision.

Resource utilization of AFTC implementation on FPGA is shown in Table 2. It presents the information concerning the number of Input and Output blocks, Slices Registers, Slice LUTs, and number of DSP. The performance of the hardware solution based on the FPGA in terms of execution time is shown in Table 3.

Table 2. Used resources.

	Used	Available
Number of bonded Input or Output	341	640
Number of Slices Registers	2269	44800
Number of Slice Look Up Table	27364	44800
Number of DSP48Es	124	128

Table 3. FPGA time performance of the FTC.

Module	(Execution Time (μs
DTC-SVM	0.88
FDI	0.15
SCSR	0.42
AD interface	2.4

A comparison between software and hardware solutions is illustrated in Figure 17.

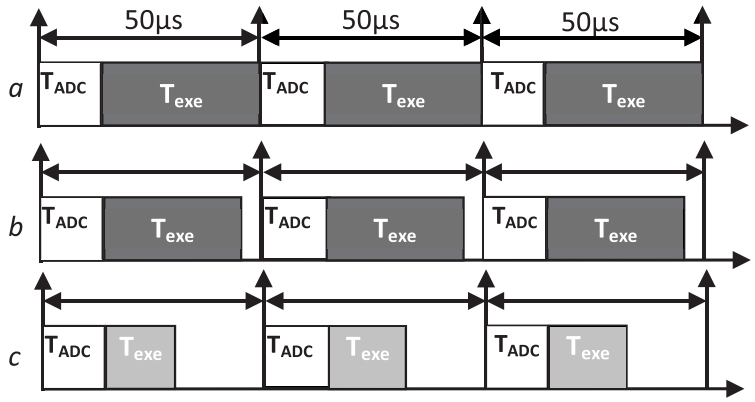


Figure 17. Timing diagram: (a) microcontroller; (b) dSPACE; (c) Xilinx Virtex–V FPGA, where TADC is the Analogue to Digital Conversion time and Texe is the execution time. The whole execution time control of the control algorithm is equal to

$$T_{exe} = T_{DTC\text{-}SVM} + t_{FDI} + t_{SCSR}$$

$$= 0.88 + 0.15 + 0.42 = 1.45\mu s$$

In Tabbache et al. (2013), this paper examines the validation of a current sensor AFTC in IM. The proposed active FTC is implemented using a dSPACE 1103. In Boussak et al. (2006), using the dSPACE 1102, the execution time is 300µs. Therefore, using software solution, the sampling time is 100µs, due to the sequential processing (Bhoopendra et al., 2012; Elbacha et al., 2012; Kostic et al., 2009; Singh et al., 2012). In this paper, using the hardware solution (FPGA), the execution time is of the order of 1 to 2µs. So, the obtained execution time becomes far lower compared to dSPACE.

CONCLUSION

The purpose of this paper was to demonstrate the use of XSG to implement our system. This paper proposed a new method to implement a fast current sensor FTC algorithm for electrical drives. The implemented architecture gives lower complexity and improves efficiency in the area. It is also a good choice for low cost hardware design.

The proposed systems adaptively reorganize themselves in the event of mechanical current sensor loss recovery to sustain the best control performance. The proposed fault-tolerant control strategies have been simulated on IM drive. Simulation results, using Xilinx System Generator, in terms of speed and torque responses, show the global effectiveness of both approaches.

REFERENCES

Adouni, A., Ben Hamed, M., Flah, A. & Sbita, L. 2013. Sensor and actuator Fault Detection and Isolation based on Artificial Neural Networks and Fuzzy logic applicated on induction motor. IEEE Control, Decision and Information Technologies Conference: 917-922.

Balasubramanian, T., Rajesh, T. & Raja Perumal, T.A. 2013. Load Sharing In A Hybrid Power System With Renewable Energy Sources. International Journal of Engineering Trends in Electrical and Electronics 3 (**1**): 35-39.

Berriri, H., Naouar, M. & Slama Belkhodja, I. 2011. Parity space approach for current sensor fault detection and isolation in electrical systems. IEEE Systems Signals and Devices Conference: 1-7.

Bhoopendra, S., Shailendra, J. & Sanjeet, D. 2012. Direct Torque Control Induction Motor Drive with Improved Flux Response. Hindawi Publishing Corporation Advances in Power Electronic: 1-11.

Bibhrajit, H. & Nilanjan, S. 2013. Robust Nonlinear Analytic Redundancy for Fault Detection and Isolation in Mobile Robot. International Journal of Automatic and Computing 4(**2**): 177-182.

Boussaid, B., Chouiref, H. & Abdelakrim, M. N. 2013. Integrated active fault tolerant control approach based LMI. International Journal of Sciences and Techniques of Automatic Control and computer engineering 7 (**1**): 1834-1843.

Boussak, M. & Jarray, K. 2006. High-Performance Sensorless Indirect Stator Flux Orientation Control of Induction Motor Drive. IEEE Transaction on Industrial Electronics 53 (**1**): 14-49.

Bruccoleri, M., Amico, M. & Perrone, G. 2003. Distributed intelligent control of exceptions in reconfigurable manufacturing systems. International Journal of Production Research 41 (**7**): 1393–1412.

Cheol Cho, H., Knowles, J., Sami Fadali, M. & Soon Lee, K. 2010. Fault Detection and Isolation of Induction Motors Using Recurrent Neural Networks and Dynamic Bayesian modelling. IEEE Transactions on Control Systems Technology 18 (**2**):430–437.

Dagbagi, M., Idkhajine, L., Monmasson, E. & Slama-Belkhodja, I. 2012. FPGA Implementation of Power Electronic Converter Real-Time Model. International Symposium on Power Electronics, Electrical Drives, Automation and Motion: 658–663.

Dumont, P.E., Aitouche, A., Merzouki, R. & Bayart, M. 2006. Fault Tolerant Control on an Electric Vehicle. IEEE International Conference on Industrial Technology: 2450–2455.

Ebrahimi Bavil, R., Khosrowjerdi, M.J. & Vatankhah, R. 2015. Active Fault Tolerant Controller Design using Model Predictive Control. Journal of control engineering and applied informatics 17 (**3**): 68-76.

Elbacha, A., Boulghasoul, Z. & Elwarraki, E. 2012. A Comparative Study of Rotor Time Constant Online Identification of an Induction Motor Using High Gain Observer and Fuzzy Compensator. WSEAS Transaction on System and Control: 37-53.

Ghachem, S., Benothman, K. & Benrejeb, M. 2011. Nonlinear Approach of Actuator Fault Accommodation. International Journal of Computer Science 8 (**4**): 290-297.

Halder, B. & Sarkar, N. 2006. Impact of the Order of Redundancy Relation in Robust Fault Detection of Robotic Systems. IEEE Decision and Control Conference: 6500–6505.

Hamdi, H., Rodrigues, M., Mechmeche, C. & BenhadjBraiek, N. 2013. Observer based Fault Tolerant Control for Takagi-Sugeno Nonlinear Descriptor systems. International Conference on Control, Engineering and Information Technology: 52-57.

Hasanien, H.M. 2001. FPGA implementation of Adaptive ANN controller for speed regulation of permanent stepper motor drives. Energy Conversion and Management 52(**2**): 1252–1257.

Isermann, R., Schwarz, R. & Stolzl, S. 2002. Fault-tolerant drive-by-wire systems. IEEE control systems magazine 22 (**5**): 64–81.

Jezernik, K., Korelic, J. & Horvat, R. 2013. PMSM Sliding Mode FPGA-Based Control for Torque Ripple Reduction. IEEE Transactions on Power Electronic 28 (**7**):3549 – 3556.

Kostic, V., Petronijevic, M., Mitrovic, N. & Bankovic, B. 2009. Experimental verification of direct torque control methods for electric drive application. Automatic Control and Robotic 8 (1): 111 -126.

Li, H., Monti, A. & Ponci, F. 2012. A fuzzy based sensor validation strategy for AC motor drives. IEEE Transaction on Industrial Informatics 8 (**4**): 490- 499.

Monmasson, E., Bahri, I., Idkhajine, L., Maalouf, A. & Naouar, MW. 2012. Recent advancements in FPGA based controllers for AC drives application. International conference on optimization of electrical and electronic equipment: 8 -15.

Saravanan, K. & Habeebullah Sait, H.2013. Multi Input Converter for Distributed Renewable Energy Sources. International Journal of Emerging Trends in Electrical and Electronics 22 (**1**): 51 -58.

Shahbazi, M., Poure, P., Saadate, S. & Zolghadri, M.R. 2013. FPGA-Based Reconfigurable Control for Fault-Tolerant Back-to Back Converter without Redundancy. IEEE Transactions on Industrial Electronics 60 (**8**): 3360 – 3371.

Singh, B., Jain, S. & Dwivedi, S. 2012. Experimental Direct Torque Control Induction Motor Drive with Modified Flux Estimation and Speed control Algorithm. IOSR Journal of Engineering 2 (**6**): 1296- 1300.

Tabbache, B., Rizoug, N., Benbouzid, M.H. & Kheloui, A. 2013. A Control Reconfiguration Strategy for Post-Sensor FTC in Induction Motor-Based EVs. IEEE Transactions on Vehicular Technology 62 (**3**): 965- 971.

Tabbache, B., Rizoug, N., Benbouzid, M.H. & Kheloui, A. 2013. A Control Reconfiguration Strategy for Post-Sensor FTC in Induction Motor-Based EV. IEEE Transaction on Vehicular Technology 62 (**3**): 965 -971.

Submitted: 28/03/2016
Revised : 22/10/2016
Accepted : 08/01/2017

استخدام طريقة تصميم جهاز ذو نظام تحكم يتحمل الخطأ في محرك حثي للمركبات الكهربائية

سها بو قتيدة ، سفيان قدايم و متبع عبد اللطيف

مختبر الإلكترونيات والإلكترونيات الدقيقة، كلية العلوم، جامعة المنستير، تونس

الخلاصة

في هذا البحث، تم عرض طريقة تصميم لأنظمة تحكم تتسم بالفاعلية والقدرة على تحمل الأخطاء (AFTC) في محرك حثي (IM). ويضمن AFTC التشغيل المستمر لـ IM في حالة حدوث خلل في مجس التيار. وتم توضيح النظام المقترح باستخدام نظامي تحكم. تم استخدام التحكم المباشر في العزم المرتكز على مضمن ناقل فضاء DTC-SVM عند تشغيل النظام في الوضع السليم، ويتطلب DTC-SVM المُعتمد استخدام ثلاثة مجسات للتيار. وتم استخدام التحكم في السرعة مع تنظيم الانزلاق (SCSR) عندما يعمل النظام في وضع الفشل، ولا يتطلب SCSR المُعتمد استخدام أي مجسات للتيار. عند استخدام وحدة تحكم دقيقة أو معالج إشارات رقمية، تعرضت الخوارزمية المستخدمة لتأخر حسابي. لذلك، ومن أجل حل هذه الصعوبة، يتطلب الأمر استخدام مصفوفة البوابات المنطقية القابلة للبرمجة (FPGA) وذلك لسرعتها في المعالجة. تم تطوير عدة تدفقات وأدوات تصميم جديدة لتطبيق خوارزمية على FPGA، مثل مولد نظام Xilinx (XSC) وهو عبارة عن أداة تصميم عالية المستوى ترتكز على الكتلة وتقدم محاكاة دقيقة. ويمكن لهذه الأداة أن تولد رمز منطقي عالي الكثافة جداً تلقائياً (VHDL) دون الاستعانة بعمليات برمجة صعبة، وبدون الحاجة لأن تلتزم بإجراء عمليات التقريب، وأكثر من ذلك يمكننا تصور سلوك الجهاز قبل التنفيذ.