International Journal of Electrical and Computer Engineering (IJECE)
Vol. 7, No. 4, August 2017, pp. 1760~1769
ISSN: 2088-8708, DOI: 10.11591/ijece.v7i4.pp1760-1769

A Simplified Speed Control Of Induction Motor based on a Low Cost FPGA

Lotfi Charaabi, Ibtihel Jaziri
Departement of Electrical Engineering, L.S.E-ENIT, université Tunis El Manar BP 37 EL Belvédère, 1002 Tunis, Tunisia

Article Info

Article history:

Received Jan 1, 2017
Revised Mar 16, 2017
Accepted Mar 30, 2017

Keyword:

Design efficiency
FOC
FPGA
Induction machine
PI controller

ABSTRACT

This paper investigates the development of a simplified speed control of induction motor based on indirect field oriented control (FOC). An original PI-P controller is designed to obtain good performances for speed tracking. Controller coefficients are carried out with analytic approach. The algorithm is implemented using a low cost Field Programmable Gate Array (FPGA). The implementation is followed by an efficient design methodology that offers considerable design advantages. The main advantage is the design of reusable and reconfigurable hardware modules for the control of electrical systems. Experimental results carried on a prototyping platform are given to illustrate the efficiency and the benefits of the proposed approach.

Corresponding Author:

Lotfi Charaabi,
Departement of Electrical Engineering,
L.S.E-ENIT, université Tunis El Manar BP 37 EL Belvédère,
1002 Tunis, Tunisia.
Email:lotfi.charaabi@enit.rnu.tn

1. INTRODUCTION

The Field Oriented Control (FOC) or vector control has seen rapid expansion in recent years. The FOC can be used to vary the speed of an induction motor over a wide range. It was initially developed by Blaschke in the beginning of 1970s [1]. The FOC can be implemented in two ways Indirect and Direct control scheme. The technique described in this work is based on indirect FOC because there is no direct access to the rotor currents. Indirect vector control of the rotor currents is accomplished using the following data:

a. Rotor mechanical velocity
b. Instantaneous stator phase currents
c. Rotor electrical time constant

The motor must be equipped with stator currents sensors and a rotor velocity feedback device. Traditional indirect vector control consists of the ten blocks [2], [3]:

a. Clarke forward transform block
b. Park forward and inverse transform block
c. Rotor flux angle estimator block
d. Three PI controller blocks
e. Field weakening block
f. SVM block

This paper presents a simplified speed control of induction Motor which consists of only six blocks:

a. Clarke inverse transform block
b. Park inverse transform block

c. Rotor flux angle estimator block
d. One PI-P controller block
e. Field weakening block
f. Hysteresis block

Microprocessors and Digital signal Processors (DSP) based solutions are available for digital AC motor control applications like DSPIC family from Microchip [4], TMS320C24x family from Texas Instruments [5] and STM32 family from STMicroelectronics [6]. Nevertheless, hardware solutions such as FPGAs have already been used with success in AC motor control and drive applications such as pulse width modulation (PWM) [7], [8], Direct Torque Control of induction motor [9-11] and FOC [12], [13] drives.

In this paper, a simplified speed control of an induction motor based a low cost FPGA is proposed. The FPGA implementation is outlined by an efficient design methodology which is based on modularity and reusability concepts [14], [15].

The major benefit for using FPGA is the achievement of the digital control algorithm within a few µ second [16]. The calculation time, including the A/D conversion time of 2.4 µs, for the proposed FPGA based controller is only equal to 3.135 µs. So, for a 20 KHz power converter, the digital control feedback can be approximated quite closely to an analogue one because the effects of sampling and delay in the feedback loop are small compared to the process time scale improving therefore the performances of the control.

The first section will detail the simplified speed control strategy of an induction motor. This strategy is based on indirect FOC. The second section will present the hardware architecture design based FPGA. Finally, in a third section, some experimental results carried on a prototyping platform will be shown for the validation of the developed control system.

2. SIMPLIFIED SPEED CONTROL STRATEGY

2.1. Principle of the Proposed Control Strategy

The well-known discrete-time model of a squirrel-cage induction motor in the dq reference frame is used for this study. The voltage, the stator flux linkage and the electromagnetic torque Equations expressed in the rotor reference frame (d-q coordinates, with d-axis linked to the inductor) are:

$$u_{sd} = \left(R_s + R_r \frac{L_m^2}{L_r^2}\right) i_{sd} + \sigma L_s \frac{di_{sd}}{dt} - \left(\omega_e \sigma L_s i_{sq} + R_r \frac{L_m}{L_r^2} \psi_{rd}\right) \cdots \quad (1)$$

$$u_{sq} = \left(R_s + R_r \frac{L_m^2}{L_r^2}\right) i_{sq} + \sigma L_s \frac{di_{sq}}{dt} + \left(\omega_e \sigma L_s i_{sd} + \omega_{sl} \frac{L_m}{L_r} \psi_{rd}\right) \cdots \quad (2)$$

$$u_{rd} = 0 = R_r i_{rd} + \frac{d\psi_{rd}}{dt} - (\omega_e - \omega_r)\psi_{rq} \quad (3)$$

$$u_{rq} = 0 = R_r i_{rq} + \frac{d\psi_{rq}}{dt} - (\omega_e - \omega_r)\psi_{rd} \quad (4)$$

$$\psi_{sd} = L_s i_{sd} + L_m i_{rd} \quad (5)$$

$$\psi_{sq} = L_s i_{sq} + L_m i_{rq} \quad (6)$$

$$\psi_{rd} = L_r i_{rd} + L_m i_{sd} \quad (7)$$

$$\psi_{rq} = L_r i_{rq} + L_m i_{sq} \quad (8)$$

$$T_e = p \frac{L_m}{L_r} \left(\psi_{rd} i_{sq} - \psi_{rq} i_{sd}\right) \quad (9)$$

$$(\omega_e - \omega_r) = \omega_{sl} \quad (10)$$

$$J \frac{d\omega_r}{dt} + f\omega_r = T_e - T_l \quad (11)$$

Where Rs is the stator resistance, R_r is the rotor resistance, Lm the stator/rotor mutual inductance, L_s and L_r the stator-rotor inductances, p the number of pole pairs, ω_e the electrical velocity, ω_r is the angular velocity of the rotor, ω_{sl} is the slip velocity, usd and usq the d-q components of the stator voltage, i_{sd} and i_{sq} the d-q components of the stator current, urd and urq the d-q components of the rotor voltage, i_{rd} and i_{rq} the d-

q components of the rotor current, ψ_{sd} and ψ_{sq} the d-q components of the stator flux linkage , ψ_{rd} and ψ_{rq} the d-q components of the rotor flux linkage and Te the electromagnetic torque. The rotor flux is allowed to be aligned with the d-axis so that

$$\psi_{rq} = 0 \tag{12}$$

This constraint can be represented by the vector diagram in Figure 1.

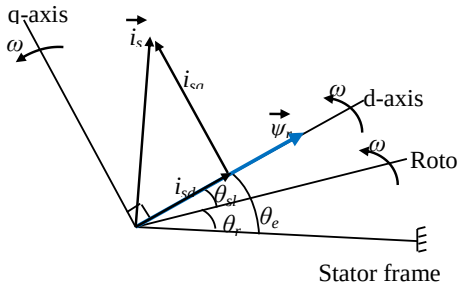

Figure 1. Induction machine vector diagram with ψrq set to zero

Setting ψ_{rq} to zero in Equations (9), the new torque Equation becomes.

$$T_e = p\frac{L_m}{L_r}(\psi_{rq} i_{sq}) \tag{13}$$

If Equations (3) and (7) are combined using the constraint (12), the rotor flux Equation becomes

$$\psi_{rq} = \frac{L_m}{1+\tau_r s} i_{sd} \tag{14}$$

Where *s* denotes the differential operator *d/dt* and τ_r the rotor time constant.
Equation (14) implies that the rotor flux depends only on the stator current.
If Equations (4) and (8) are combined using the constraint (12), the slip velocity becomes

$$\omega_{sl} = -\frac{R_r i_{rq}}{\psi_{rd}} \tag{15}$$

Using Equations (6) and (10) a new slip velocity Equation can be defined

$$\omega_{sl} = \frac{L_m}{\tau_r}\frac{i_{sq}}{\psi_{rd}} \tag{16}$$

So, the slip angle is estimated by the following relation:

$$\theta_{sl} = \int \frac{L_m}{\tau_r \psi_{rd}} i_{sq} dt + \theta_{sl0} \tag{17}$$

Where θ_{sl0} is the initial slip angle which can be set to zero
Equation (10) gives

$$\theta_e = \theta_r + \theta_{sl} = \int \frac{L_m}{\tau_r \psi_{rd}} i_{sq} dt + \theta_r \tag{18}$$

Using Equations (2), (6) and (15), the q-component stator voltage usq can be expressed with i_{sd}, i_{sq} and ψ_{rd}

$$u_{sq} = \left(R_s + 2R_r\frac{L_m^2}{L_r^2}\right) i_{sq} + \sigma L_s \frac{di_{sq}}{dt} + (\sigma L_s \omega_e i_{sd}) \tag{19}$$

For this control strategy, the d-component stator current isd is imposed to obtain the nominal torque. Subsequently, by Equation (14), the d-component of the rotor flux linkage ψ_{rd} becomes constant in steady-state. Then, controlling isd implies controlling the torque. The related control scheme is shown in Figure 2.

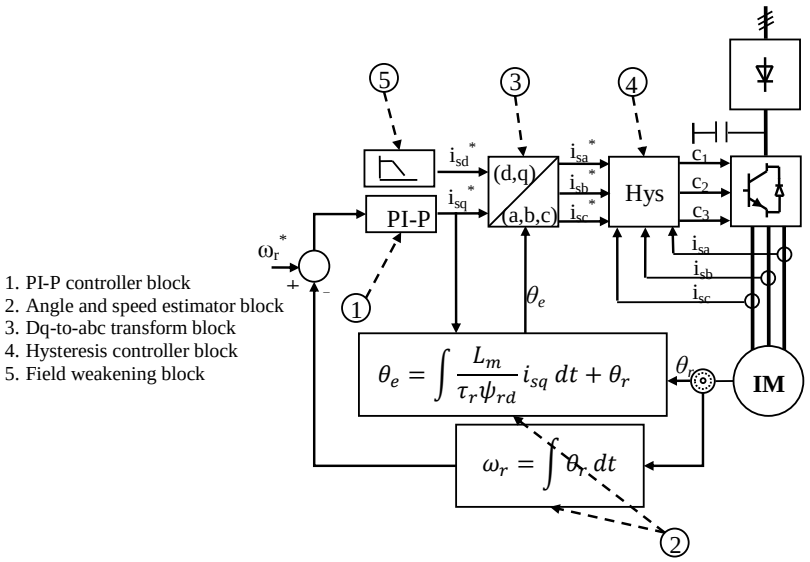

Figure 2. Principle of the control strategy

2.2. PI-P Controller Design

The PI-P regulator is introduced into the control scheme in order to achieve a second order system with a damping coefficient ζ=0.7.

The dynamic model of the speed induction motor drive is significantly simplified, and can be reasonably represented by the block diagram shown in Figure 3. τ represents the time constant for the desired current i_{sq}

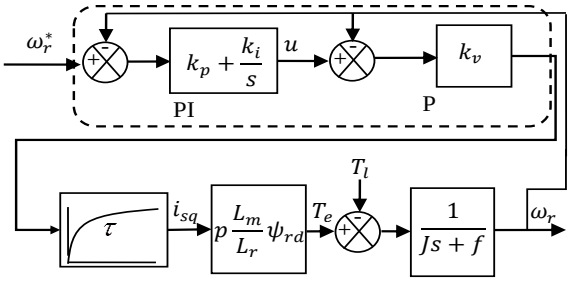

Figure 3. The block diagram of the PI-P regulator with the process

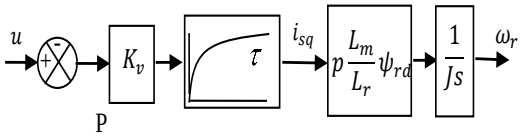

Figure 4. P regulator associated with the process

The goal of the P regulator is to obtain a second order system with real poles in closed loop. In order to simplify the calculation we neglect the viscous friction coefficient f. Figure 4 shows the simplified block diagram of the P regulator with the process.

The closed loop transfer function

$$H(s) = \frac{\omega_r(s)}{u(s)} = \frac{\frac{k_v p L_m \psi_{rd}}{J\tau L_r}}{s^2 + \frac{1}{\tau}s + \frac{k_v p L_m \psi_{rd}}{J\tau L_r}} \quad (20)$$

The P coefficient k_v is selected to obtain a double real pole called ω_n

$$k_v = \frac{JL_r}{4\tau p L_m \psi_{rd}} \quad (21)$$

Then, the transfer function becomes

$$H(s) = \frac{\frac{k_v p L_m \psi_{rd}}{J\tau L_r}}{(s+\omega_n)(s+\omega_n)} \quad (22)$$

Where $\omega_n = 1/2\tau$

The PI regulator is introduced before the P regulator in order to compensate the real pole ω_n and to obtain a second order system in closed loop

To compensate the real pole ω_n, PI coefficients must obeys this rule

$$\frac{k_i}{k_p} = \omega_n \quad (23)$$

Using Equation (23), the global closed loop transfer function becomes

$$G(s) = \frac{\omega_r(s)}{\omega_r^*(s)} = \frac{\frac{k_v k_p p L_m \psi_{rd}}{J\tau L_r}}{s^2 + \omega_n s + \frac{k_v k_p p L_m \psi_{rd}}{J\tau L_r}} \quad (24)$$

Then, for a desired damping coefficient ζ, k_p is expressed by the following Equation

$$k_p = \frac{\tau J L_r \omega_n^2}{4\zeta^2 p L_m \psi_{rd}} \quad (25)$$

3. ARCHITECTURE DESIGN

The purpose of this section is to develop a discrete-time and an optimized architecture based FPGA for the control algorithm. The most used discretization method is based on Forward shift approximation [17]. The shift form approximation is given by

$$\frac{d}{dt} = \frac{z-1}{T} \quad (26)$$

Where T is the sampling period.

As shown in Figure 2, the control algorithm is divided into four modules. The description of the different modules is detailed below.

PI-P controller block: This module generates the digital values of the stator component references i_{sq}* through the rotor angular velocity error. The discretization of the PI-P algorithm using the Forward shift approximation gives

$$\begin{cases} e(k) = \omega_r^*(k) - \omega_r(k) \\ u(k) = K_p e(k) + i(k) \\ i(k) = i(k-1) + K_i e(k) \\ i_{sq}(k) = K_v\big(u(k) - \omega_r(k)\big) \end{cases} \quad (27)$$

Where K_p, K_i and K_v depends on k_p, k_i, k_v and T

The data flow graph (DFG) corresponding to the PI-P algorithm is presented in Figure 5

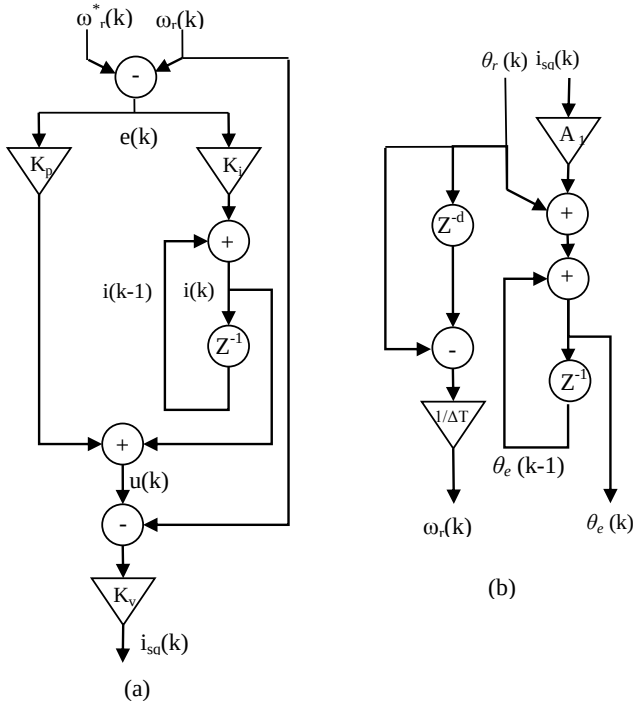

Figure 5. (a) DFG of the PI-P controller, (b) DFG of the speed and angle estimator

Angle and speed estimator block: This module generates the rotor velocity ω_r and the electrical angle θ_e through the rotor angle θ_r. The rotor angle is provided by an absolute coder.

The discretization of Equation (18) provides the discrete-time Equation of the electrical angle estimator:

$$\theta_e(k) = \theta_e(k-1) + A_1 i_{sq}(k) + \theta_r(k) \tag{28}$$

Where $A_1 = T\frac{L_m}{\tau_r \psi_{rd}}$

The rotor velocity is obtained from the rotor angle using the following Equation:

$$\omega_r(k) = \frac{\theta_r(k) - \theta_r(k-d)}{\Delta T} \tag{29}$$

Where $\theta_r(k-d)$ is the rotor angle at instant $k + \Delta T$. Figure 5 shows the DFG of the estimator *dq-to-abc transform block*: This module contains the dq-to-abc transformation. It generates the digital values of the stator current references i_{sa}*, i_{sb}* and i_{sc}*. Equation (30) shows a matrix representation of this module

$$i_{s(a,b,c)} = \begin{bmatrix} \sqrt{\frac{2}{3}} & 0 \\ -\frac{1}{\sqrt{6}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \cos(\theta_s) & -\sin(\theta_s) \\ \sin(\theta_s) & \cos(\theta_s) \end{bmatrix} \begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix} \tag{30}$$

Using trigonometric formula, Equation (30) leads to the Equation (31)

$$\begin{cases} i_{sa} = \sqrt{\frac{2}{3}}\left(\sin\left(\theta_s + \frac{\pi}{2}\right) i_{sd} + \sin(\theta_s + \pi)\, i_{sq}\right) \\ i_{sb} = \sqrt{\frac{2}{3}}\left(\sin\left(\theta_s + \frac{11\pi}{6}\right) i_{sd} + \sin\left(\theta_s + \frac{\pi}{3}\right) i_{sq}\right) \\ i_{sc} = -i_{sa} - i_{sb} \end{cases} \tag{31}$$

Figure 6 shows the DFG corresponding to the dq-to-abc transformation module.

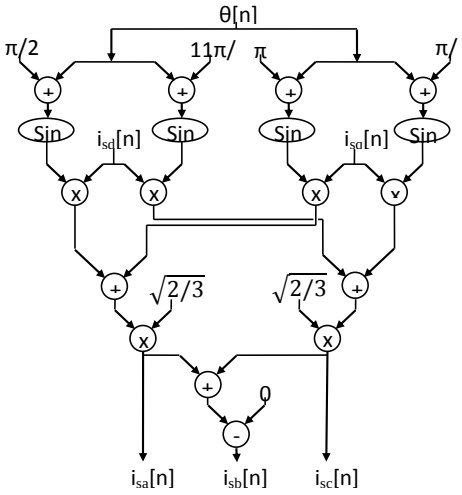

Figure 6. DFG of the dq-to-abc transformation estimator

Hysteresis controller block: This module contains three identical hysteresis controllers. It generates the switching states c_1, c_2 and c_3 via the comparison of the stator current references to the measured stator currents.

4. Experimental Set-up

For this project, the used FPGA target is a XC2s100 from Xilinx Inc. The FPGA based hardware control system includes the speed controller, an AD converter interface and a serial interface in one FPGA chip. Figure 7 shows the corresponding implemented architecture.

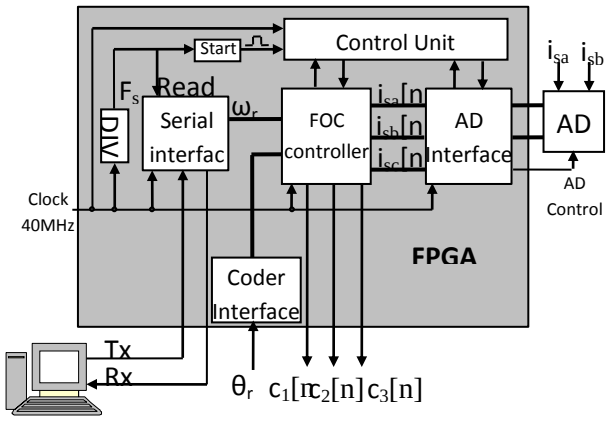

Figure 7. FPGA based hardware control system

The serial interface module provides a serial communication between the host PC and FPGA. The control unit is started at each rising edge of the sampling frequency Fs. It activates firstly the AD and coder interface which starts the AD conversion process. AD conversion of the stator currents takes 2.4µs. When the conversion process is finished, the AD interface module read converted data and treats them to generate the digital values of the measured stator currents $i_{sa}[n]$, $i_{sb}[n]$ and $i_{sc}[n]$. Then, the control unit activates the speed controller module. This module allows the generation of the switching states $c_1[n]$, $c_2[n]$ and $c_3[n]$of the VSI. The computation time, including the AD conversion time, from the AD converter stator currents acquisition

to the switching states generation is equal to 3.135 µs. By comparison, the used dead time of the used VSI is about 3.3 µs. So the computation time is almost negligible and the digital control feedback can be approximated to an analog one.

To test the FPGA based controller, a test bed for the control of an induction machine was assembled. Figure 8 shows the prototyping platform.

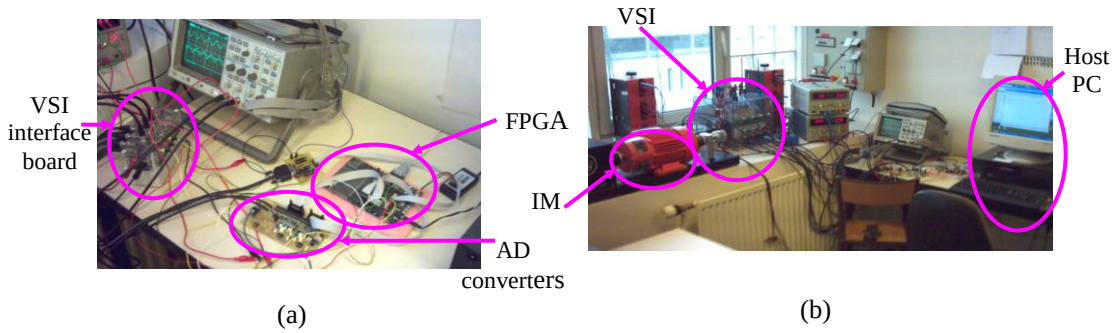

Figure 8. Prototyping platform (a) Control system (b) General view

The test bed is composed of a 0.75 Kw induction motor provided with a 1024 points encoder, current sensors and a controlled load for load torque generation. The VSI module includes a three phase IGBT based inverter, a 2200 µF capacitance and a three phase diode rectifier. An AD conversion circuit board is used to convert the measured currents and an inverter interface circuit board allows the voltage level adaptation of the switching states for the control of the inverter.

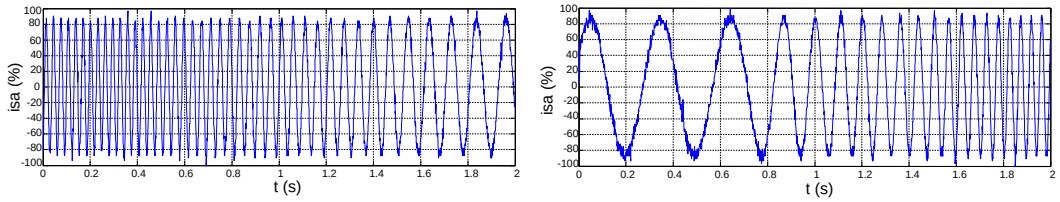

Figure 9. (a)Stator current i_{sa} for speed step input from 200 rad/s to 50 rad/s(Band width=0%Isn , and Fs=10KHz) (b)Stator current isa for speed step input from 20 rad/s to 200 rad/s(Band width=0%Isn , and Fs=10K Hz)

During experimentation, the DC voltage source E of the three phase inverter is set to 400V. Figure 9 presents the experimental results of the stator current isa for an hysteresis controller band width equal to 0% of the rated line current, a sampling frequency F_s equal to 10KHz and different values of the speed input.

Figure 10 presents the speed response of the system for ramp input. It shows the speed response after a torque load perturbation. The speed response is provided by the serial interface to a host PC.

Experimental results shown in Figure 9 and Figure 10 give proof that the control system satisfy the basic requirements of the control strategy and validate therefore the good functionality of the system. The same experiment has been done in the literature [20] with the indirect FOC which gave similar performances. Table 1 shows the variation of the time response for the indirect FOC and the simplified indirect FOC.

Table 1. Variation of settling time, maximum overshoot with indirect FOC

Controller	Indirect FOC	Simplified indirect FOC
Rise time	1.9 s	2 s
Maximum overshoot in (%)	12.6	2.3

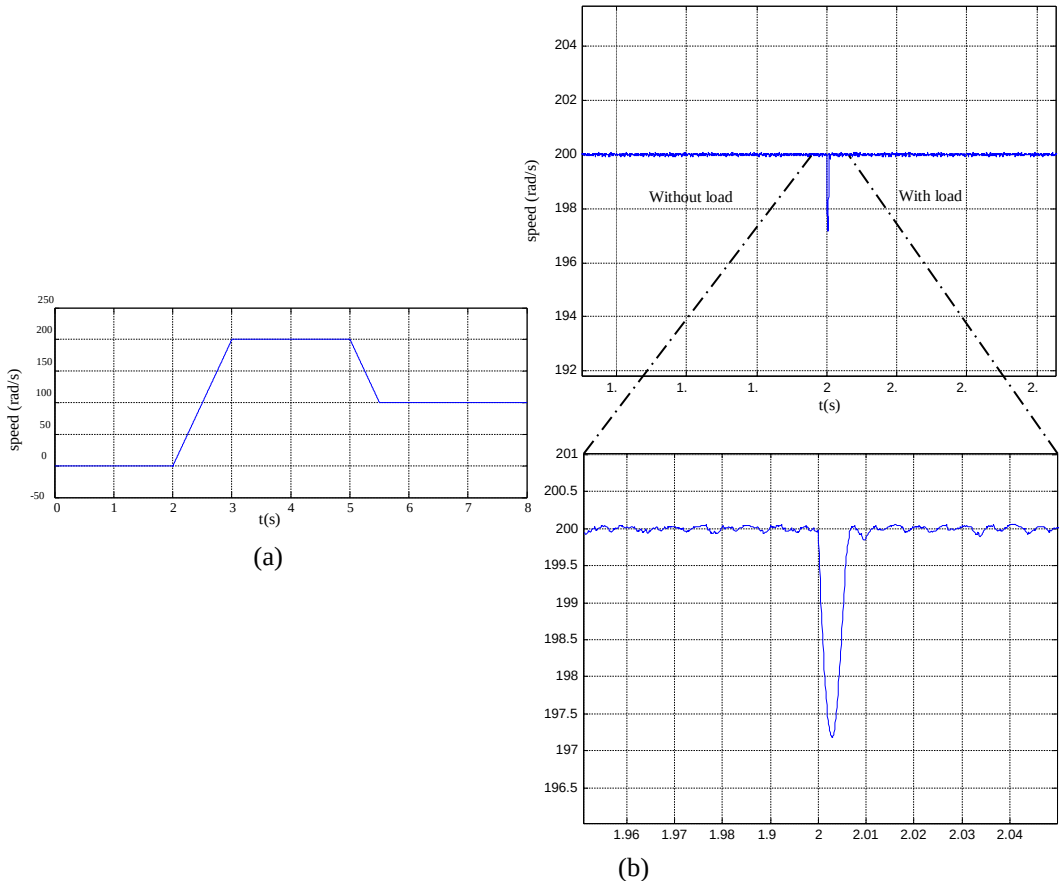

Figure 10. (a) Speed response of ramp as function of time (b) Speed response for 200 rad/s reference input without and with torque load

5. CONCLUSION

This paper presents the implementation on a FPGA of a simplified speed control for induction machine. The control algorithm is based on indirect FOC. It uses only six blocks instead of ten blocks. A PI-P regulator is designed to obtain good performances for speed tracking. The algorithm was implemented on a low cost FPGA. The implementation has rigorously followed an efficient design methodology. This methodology was used with success for the speed control of induction machine using FPGA based controller and it can be considered as a part of a process whose target is the creation of a specific electrical system library of optimized reusable modules which will ensure a great flexibility for the design development.

ACKNOWLEDGEMENTS

This paper was supported by the Tunisian Ministry of High Education and Research: UR-LSE-ENIT-03/UR/ES05

REFERENCES

[1] F. Blaschke, "The Principle of Field Orientation as Applied to the new TRANSVECTOR Closed Loop Control System for Rotating Field Machines", *Siemens Rev*., vol. 34, pp. 217-220, 1972.
[2] N. P. Quan, J.-A. Dittrich, "Vector Control of Three-Phase AC Machines - System Development in the Practice", Springer, 2008.
[3] Hafeezul Haq, Mehedi Hasan Imran, H.Ibrahim Okumus, Mohammad Habibullah, "Speed Control of Induction Motor using FOC Method", *Int. Journal of Engineering Research and Applications*, vol. 5, no. 3, (Part -1) March 2015, pp.154-158.
[4] AN908, "Using the dsPIC30F for Vector Control of an ACIM", (DS00908), Microchip Technology Inc. 2007
[5] Zhenyu Yu, "Space-Vector PWM With TMS320C24x/F24x using Hardware and Software Determined Switching Patterns", Application Report SPRA524, 1999 Texas Instruments Incorporated.

[6] Application note (BRSTM32MC1211), "Motor control with STM32® 32‑bit ARM®‑based MCU", STMicroelectronics - December 2011.
[7] Y. Tzou, H-J. Hsu, "FPGA Realization of Space-Vector PWM control IC for three-phase PWM Inverters", *IEEE Trans. On Power Electronics*, vol. 12, no. 6, pp. 953-963, November 1997.
[8] E. Monmasson, Y. A. Chapuis, "Contributions of FPGA's to the Control of Electricals Systems, a Review" *IEEE Industrial Electronics Society Newsletter*, vol. 49, no.4, pp.8-15, December 2002.
[9] S. Ferreira, F. Haffner, L. F. Pereira, F. Moraes, *"Design and Prototyping of Direct Torque Control of Induction Motors in FPGAs"*, Proceedings of the 16th Symposium on Integrated Circuits and Systems Design (SBCCI'03), IEEE, 2003.
[10] L. Charaabi, Eric Monmasson, M.A Naassani, Ilhem Slama-Belkhodja, *"FPGA-based DRFC and DTFSC Algorithms"*, Industrial Electronics Society, 2005. IECON 2005. 32nd Annual Conference of IEEE, pp.245-250, Raleigh, North Carolina, USA, Nov 6 to 10, 2005.
[11] F. AubBpart, P. Poure, C. Girerd, Y. A. Chapuis, F. Braun, "Design and Simulation of ASIC-based System Control: Application to Direct Torque Control of Induction Machine", ISIE'99 Bled, Slovenia, IEEE, 1999.
[12] B. Hariram, N. S. Marimuthu, "A VHDL Library of Modules for Vector Control of Induction Motor", *International Journal of Electrical and Power Engineering*, vol. 1, no. 2, pp. 225-259, Medwell Journals, 2007.
[13] Ozkan AKIN, Irfan ALAN "The use of FPGA in Field-Oriented Control of an Induction Machine", *Turk J Elec Eng & Comp Sci*, vol. 18, no. 6, 2010.
[14] W. M. Naouar, L. Charaabi, E. Monmasson, I. Slama-Belkhodja, *"Realization of FPGA Reconfigurable IP_Core Functions for the Control of Electrical Systems"*, EPE-PEMC'2004 Conf. Proc., Riga, Latvia, September 2004.
[15] L. Charaabi, E. Monmasson, I. S. Belkhodja, *"Presentation of an Efficient Design Methodology to Develop IP-Core Functions for Control Systems: Application to the Design of an Antiwindup PI Controller"*, IEEE-IECON'02 Conf.Proc, CD-Rom, Sevilla, Spain, 2002.
[16] T. Yokoyama, M. Horiuchi, S. Shimogata, *"Instantaneous Deadbeat Control for PWM Inverter using FPGA based Hardware Controller",* IEEE-IECON'03 Conf. Proc, CD-Rom, Roanoke, USA, November 2003.
[17] O. Vainio, S. J. Ovaska, and J. J. Pasanen, "A Digital Signal Processing Approach to Real-time AC Motor Modeling," *IEEE Transactions on Industrial Electronics*, vol. 39, no. 1, pp. 36–45, 1992.
[18] T. Grandpierre, C. Lavrenne, Y. Sorel, *"Optimized Rapid Prototyping for Real-Time Embedded Heterogeneous Multiprocessor"*, CODES'99 7th International Workshop on Hardware/Software Co-Design Conf. Proc, CD Rom, Rome, Italy, May1999.
[19] T. Riesgo, Y. Torroja, E. de la Torre, "Design Methodologies Based on Hardware Description Languages", *IEEE. Trans. On Industrial Electronics*, vol. 46, no. 1, pp. 3-12, February 1999.
[20] T.banerjee, S.Choudhuri, K.Das Sharma, *"Speed Tracking Scheme for FOC based Induction Motor by Fuzzy Controller"*, International Conference on Control, Instrumentation, Energy and Communication (CIEC), 2014.

APPENDIX

Induction Machine Parameters
1 KW, 230V, 50 Hz, 3 Phases, Y connection, 2 poles
Stator resistance = 7.2 Ω Rotor resistance = 1.35 Ω
Stator inductance = 0.28 H Rotor inductance = 0.075 H Mutual inductance = 0,118 H

BIOGRAPHIES OF AUTHORS

Lotfi Charaabi received the B.S degree and the M.S degree in Electrical Engineering from the Engineering school of Tunisia, in 2002 and the Ph.D. from the department of Automatic Control systems and Computer Engineering, university of Cachan French in 2006. Since 2009, he has been actively cooperating in industrial project related to motor control algorithm design and implementation. He is member of the Research laboratory LSE of the Tunisian engineering school.

Ibtihel Jaziri received the B.S degree in Industrial electronic Engineering from the Engineering school of Sousse, in 2010. She is currently working toward the Ph.D. degree with the Université de Tunis El Manar, Ecole Nationale d'Ingénieurs de Tunis, Laboratoire des Systèmes Electriques (LSE – LR11ES15), Tunis, Tunisia. Her main research interests include Co-design methodology for motor control, design and implementation of motor control algorithms.