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

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.

Copyright © 2017 Institute of Advanced Engineering and Science. All rights reserved.

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
A Simplified speed Control Of Induction motor based on a low cost FPGA (Lotfi Charaabi)

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 d-q 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:

\[ u_{sd} = \left( R_s + R_r \frac{L_s}{L_m} \right) i_{sd} + \sigma L_s \frac{di_{sd}}{dt} - \left( \omega_e L_s i_{sq} + R_r \frac{L_m}{L_r} \psi_{rd} \right) \]  

(1)

\[ u_{sq} = \left( R_s + R_r \frac{L_s}{L_m} \right) i_{sq} + \sigma L_s \frac{di_{sq}}{dt} + \left( \omega_e L_s i_{sd} + \omega_s L_m i_{eq} \frac{L_m}{L_r} \psi_{rd} \right) \]  

(2)

\[ u_{rd} = 0 = R_r i_{rd} + \frac{d\psi_{rd}}{dt} - \left( \omega_e - \omega_r \right) \psi_{rq} \]  

(3)

\[ u_{rq} = 0 = R_r i_{rq} + \frac{d\psi_{rq}}{dt} - \left( \omega_e - \omega_r \right) \psi_{rd} \]  

(4)

\[ \psi_{sd} = L_s i_{sd} + L_m i_{rd} \]  

(5)

\[ \psi_{sq} = L_s i_{sq} + L_m i_{rq} \]  

(6)

\[ \psi_{rd} = L_r i_{rd} + L_m i_{sd} \]  

(7)

\[ \psi_{rq} = L_r i_{rq} + L_m i_{sq} \]  

(8)

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

(9)

\[ \left( \omega_e - \omega_r \right) = \omega_{sl} \]  

(10)

\[ J \frac{d\omega_v}{dt} + f \omega_r = T_e - T_i \]  

(11)

Where Rs is the stator resistance, Rr is the rotor resistance, Lm the stator/rotor mutual inductance, Ls and Lr the stator-rotor inductances, p the number of pole pairs, \( \omega_e \) the electrical velocity, \( \omega_r \) is the angular velocity of the rotor, \( \omega_{sl} \) is the slip velocity, usd and usq the d-q components of the stator voltage, isd and isq the d-q components of the stator current, urd and urq the d-q components of the rotor voltage, iqr and iqr the d-
q components of the rotor current, \( \psi_{sd} \) and \( \psi_{sq} \) the d-q components of the stator flux linkage, \( \psi_{rd} \) and \( \psi_{rq} \) the d-q components of the rotor flux linkage and \( T_e \) 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.

![Induction machine vector diagram with \( \psi_{rq} \) set to zero](image)

Figure 1. Induction machine vector diagram with \( \psi_{rq} \) set to zero

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

\[
T_e = \frac{p}{2} \frac{\lambda}{\lambda_r} (\psi_{rq} i_{sq}) \tag{13}
\]

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

\[
\psi_{rq} = \frac{\lambda_m}{1 + \sigma_L} i_{sd} \tag{14}
\]

Where \( s \) denotes the differential operator \( d/dt \) and \( \tau \), 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

\[
\omega_{sl} = \frac{\psi_{rq}}{\psi_{rd}} \tag{15}
\]

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

\[
\omega_{sl} = \frac{\lambda_m}{\tau_r} \frac{i_{sq}}{\psi_{rd}} \tag{16}
\]

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

\[
\theta_{sl} = \int \frac{\lambda_m}{\tau_r \psi_{rd}} i_{sq} dt + \theta_{sl0} \tag{17}
\]

Where \( \theta_{sl0} \) is the initial slip angle which can be set to zero

Equation (10) gives

\[
\theta_e = \theta_r + \theta_{sl} = \int \frac{\lambda_m}{\tau_r \psi_{rd}} i_{sq} dt + \theta_r \tag{18}
\]

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

\[
u_{sq} = \left( R_e + 2R_r \frac{\lambda_m}{\tau_r} \right) i_{sq} + \sigma L_s \frac{di_{sq}}{dt} + (\sigma L_s \omega_e i_{sd}) \tag{19}
\]
For this control strategy, the d-component stator current $i_{sd}$ is imposed to obtain the nominal torque. Subsequently, by Equation (14), the d-component of the rotor flux linkage $\psi_{rd}$ becomes constant in steady-state. Then, controlling $i_{sd}$ implies controlling the torque. The related control scheme is shown in Figure 2.

![Figure 2. Principle of the control strategy](image)

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 $\zeta=0.7$.

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. $\tau$ represents the time constant for the desired current $i_{sq}$.

![Figure 3. The block diagram of the PI-P regulator with the process](image)

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.

![Figure 4. P regulator associated with the process](image)
The closed loop transfer function

\[
H(s) = \frac{\omega_f(s)}{u(s)} = \frac{kvpL_m\psi_{rd}}{fT_F} \frac{s + \frac{1}{2T}}{s + \frac{kvpL_m\psi_{rd}}{fT_F}}
\]  

(20)

The P coefficient \( k_v \) is selected to obtain a double real pole called \( \omega_n \)

\[
k_v = \frac{J_F}{4\pi pL_m\psi_{rd}}
\]  

(21)

Then, the transfer function becomes

\[
H(s) = \frac{kvpL_m\psi_{rd}}{fT_F} \frac{s + \frac{1}{2T}}{(s + \omega_n)(s + \omega_n)}
\]  

(22)

Where \( \omega_n = 1/2\tau \)

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

To compensate the real pole \( \omega_n \), PI coefficients must obey this rule

\[
\frac{k_i}{k_p} = \omega_n
\]  

(23)

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

\[
G(s) = \frac{\omega_f(s)}{\omega_f(s)} = \frac{kvpk_p\psi_{rd}}{fT_F} \frac{s + \omega_n s + \frac{kvpk_p\psi_{rd}}{fT_F}}{(s + \omega_n)(s + \omega_n)}
\]  

(24)

Then, for a desired damping coefficient \( \zeta \), \( k_p \) is expressed by the following Equation

\[
k_p = \frac{\tau J_F}{4\pi^2 pL_m\psi_{rd}}
\]  

(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

\[
\frac{d}{dt} = \frac{z - 1}{T}
\]  

(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

\[
\begin{align*}
e(k) &= \omega_s^*(k) - \omega_r(k) \\
u(k) &= K_pe(k) + i(k) \\
i(k) &= i(k - 1) + K_i e(k) \\
i_{sq}(k) &= K_v \left( u(k) - \omega_r(k) \right)
\end{align*}
\]  

(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
Angle and speed estimator block: This module generates the rotor velocity $\omega_r$ and the electrical angle $\theta_e$ through the rotor angle $\theta_r$. The rotor angle is provided by an absolute coder.

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

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

Where $A_1 = T \frac{I_m}{\tau_d \Phi_{rd}}$

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

$$\omega_r(k) = \frac{\theta_e(k-k) - \theta_e(k-d)}{\Delta T}$$  \hspace{1cm} (29)

Where $\theta_e(k-d)$ 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

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

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

$$\begin{cases}
i_{sa} = \frac{2}{\sqrt{3}} \left( \sin \left( \theta_s + \frac{\pi}{3} \right) i_{sd} + \sin(\theta_s + \pi) i_{sq} \right) \\
i_{sb} = \frac{2}{\sqrt{3}} \left( \sin \left( \theta_s + \frac{11\pi}{6} \right) i_{sd} + \sin \left( \theta_s + \frac{\pi}{3} \right) i_{sq} \right) \\
i_{sc} = -i_{sa} - i_{sb}
\end{cases}$$  \hspace{1cm} (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 \( F_s \). It activates firstly the AD and coder interface which starts the AD conversion process. AD conversion of the stator currents takes 2.4\( \mu s \). 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.

![Prototyping platform](image)

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.

![Experimental results](image)

Figure 9. (a)Stator current $i_{sa}$ for speed step input from 200 rad/s to 50 rad/s(Band width=0% $I_{sn}$ , and $F_s$=10KHz) (b)Stator current $i_{sa}$ for speed step input from 20 rad/s to 200 rad/s(Band width=0% $I_{sn}$ , and $F_s$=10KHz)

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 $i_{sa}$ 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>
<thead>
<tr>
<th>Controller</th>
<th>Indirect FOC</th>
<th>Simplified indirect FOC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rise time</td>
<td>1.9 s</td>
<td>2 s</td>
</tr>
<tr>
<td>Maximum overshoot in (%)</td>
<td>12.6</td>
<td>2.3</td>
</tr>
</tbody>
</table>

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

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.