FPGA-Based Implementation Sliding Mode Control and nonlinear Adaptive backstepping control of a Permanent Magnet Synchronous Machine Drive

BADRE BOSSOUFI 1), MOHammed karim 1), Ahmed lagrioui1),
1) Laboratory of Electrical Engineering and Maintenance.
Higher School of Technology-Oujda, University of Mohammed I, Morocco.
2) STIC Team, Faculty of Sciences Dhar El Mahraz Fez, Morocco
Badre_isai@hotmail.com

Abstract: In this paper, the adaptive and non-adaptive nonlinear Backstepping control approach for a permanent magnet synchronous motor drive is discussed and analyzed. We present a Matlab&Simulink simulation and experimental results from a benchmark based on FPGA. The design of a controller for a nonlinear system where the state vector dimension is high, can often be a difficult task, if not impossible. The Backstepping technique provides a systematic method to address this type of problem. It combines the notion of Lyapunov function and a controller procedure recursively. First, the adaptive and no adaptive Backstepping control approach is utilized to obtain the robustness for mismatched parameter uncertainties. The overall stability of the system is shown using Lyapunov technique. The simulation results clearly show that the proposed scheme can track the speed reference. Secondly, some experimental results are demonstrated to validate the proposed controllers. The experimental results carried from a prototyping platform are given to illustrate the efficiency and the benefits of the proposed approach and the various stages of implementation of this structure in FPGA.

Keywords: Not Adaptive Backstepping control; Backstepping Design Technique; FPGAs; Permanent Magnet Synchronous Machine (PMSM); Lyapunov Stability; Robust Adaptive Control; Systems Generator; Reusability.

1. Introduction

Three-phase Permanent Magnet Synchronous Motor (PMSMs) is strongly used in industry and consumes more than 70% of industrial electricity. This is why considerable efforts and different searches are being done to improve their performances and their efficiency. The efficiency of electrical machine drives is greatly reduced at light loads, where the flux magnitude reference is held on its initial value. The loss minimization is realized using high-quality materials and excellent design procedures in the manufacturing process. Moreover, expert control algorithms are employed in order to improve machine performance. In this paper we are interested in two mode controls for PMSM drive, the not adaptive backstepping and indirect sliding mode control (ISMO).

The not adaptive backstepping approach offers a choice of design tools for accommodation of uncertainties nonlinearities. And can avoid wasteful cancellations. However, the not adaptive backstepping approach is capable of keeping almost all the robustness properties of the mismatched uncertainties. The not adaptive backstepping is a rigorous and procedure design methodology for nonlinear feedback control. The principal idea of this approach is to recursively design controllers for machine torque constant uncertainty subsystems in the structure and “step back” the feedback signals towards the control input. This approach is different from the approach of the conventional feedback linearization in that it can avoid cancellation of useful nonlinearities in pursuing the objectives of stabilization and tracking. A nonlinear backstepping control design scheme is developed for the speed tracking control of PMSM that has exact model knowledge. The asymptotic stability of the resulting closed loop system is guaranteed according to Lyapunov stability theorem.

The speed variation of the PMSM is widely used in high-performance applications. The PMSM has very large power density, high power factor and high efficiency. In a high-performance control of PMSM, the information of rotor position and speed is very important. In the speed control loop, for the field oriented control, the coordinate transformation has needs precise rotor position. Rotor position and speed can be measured by a shaft encoder or other type of sensors, in other case the speed is measured with an Encoder resolver connected to the PMSM machine drive. However, the presence of such sensors is not acceptable for cost, maintenance and reliability reasons. The concept of sensorless control
was proposed in the 1970s and has been continually developed for PMSM rotor position and speed estimation. The basic principle of sensorless control is to deduce the rotor speed and position using various information and means, including direct calculation, parameter identification, condition estimation, indirect measuring and so on. The stator currents and voltages are generally used to calculate the information of speed and rotor position.

The FPGA technology is now used by an increasing number of designers in various fields of application such as signal processing, telecommunication, video, embedded control systems, and electrical control systems. This last domain, i.e. the studies of control of electrical machines, will be presented in this paper [1]. Indeed, these components have already been used with success in many different applications such as Pulse Width Modulation (PWM), control of induction machine drives and multimachine system control. This is because the FPGA-based implementation of controllers can efficiently answer current and future challenges of this field.

Considering the complexity of the diversity of the electric control devices of the machines, it is difficult to define with universal manner a general structure for such systems.

Considering the complexity of the diversity of the electric control devices of the machines, it is difficult to define with universal manner a general structure for such systems. However, by having a reflexion compared to the elements most commonly encountered in these systems, it is possible to define a general structure of an electric control device of machines which is show in Fig.1:

![Image of Control Architecture]

This paper presents the realization of a platform for non adaptive and adaptive Backstepping control of PMSM using FPGA based controller. This realization is especially aimed for future high performance applications. In this approach, not only the architecture corresponding to the control algorithm is studied, but also architecture and the ADC interface, Encoder interface and RS232 UART architecture [2].

2. PMSM model system

In this paper, we apply the different algorithms control on a machine type PMSM (Permanent Magnet Synchronous Motor) [3], which consists of three stator windings and a rotor magnet. This motor is described by the following equation (Voltage, Flux, Torque...),

\[
\begin{align*}
V_{sd} &= r_s i_{sd} + \frac{d\Phi_{sd}}{dt} - \omega \Phi_{sq} \\
V_{sq} &= r_s i_{sq} + \frac{d\Phi_{sq}}{dt} - \omega \Phi_{sd} \\
\Phi_{sd} &= L_{sd} i_{sd} + \Phi_f \\
\Phi_{sq} &= L_{sq} i_{sq} \\
C_e &= J \frac{d\Omega}{dt} + f \Omega - C_r \\
\omega &= p \Omega
\end{align*}
\]

Where \( \Omega \) is the rotation's speed, \( p \) the Number of pairs of poles, \( J \) the moment of inertia, \( f \) the Coefficient of viscous friction, \( C_r \) the resistive torque, \( \Phi_f \) the flux produced by the permanent magnet, \( L_{sd} \) and \( L_{sq} \) the d-q axis stator inductance, \( V_{sd} \) and \( V_{sq} \) the d-q axis stator voltage, \( r_s \) the stator winding resistance and \( C_e \) the electromagnetic torque.

3. Nonlinear not adaptive Backstepping approach

The schematic diagram of the speed control system study with application of the nonlinear Backstepping controller is shown in Fig.1. The parameters of the synchronous machine are given in the Appendix. In this section, we employ the nonlinear Backstepping schemes to design the controllers for PMSM systems with angular velocity measurement.
The Not Adaptive Backstepping approach algorithm is control techniques that can linearizing a nonlinear system such as the PMSM machine drive in the presence of uncertainties. Unlike other feedback linearization techniques, adaptive Backstepping has the flexibility of keeping useful non linearity’s intact during stabilization. The essence of Backstepping is the stabilization of a virtual control state. Hence, it generates a corresponding error variable which can be stabilized by carefully selecting proper control inputs. These inputs can be determined from Lyapunov stability analysis [4].

It is obvious that the dynamic model of PMSM is highly nonlinear because of the coupling between the speed and the stator currents (equation (1)). According to the vector control principle, the direct axis current $i_d$ is always forced to be zero in order to orient all the linkage flux in the d axis and achieve maximum torque per ampere.

$$\frac{di_d}{dt} = -\frac{r_s}{L_{sd}} i_d + \frac{L_{sq}}{L_{sd}} p\Omega i_q + \frac{V_{sd}}{L_{sd}}$$

$$\frac{di_q}{dt} = -\frac{r_s}{L_{sq}} i_q - \frac{L_{sd}}{L_{sq}} p\Omega i_d - \frac{\Phi_f}{L_{sq}} + \frac{V_{sq}}{L_{sq}}$$

$$\frac{d\Omega}{dt} = \frac{3p}{2J} (\Phi_f i_q + (L_{sd} - L_{sq}) i_d i_q) - \frac{f}{J} \Omega + \frac{C_r}{J}$$

The vector $[x] = [i_d \ i_q \ \Omega]$ choice as state vector is justified by the fact that currents and speed are measurable and that the control of the instantaneous torque can be done comfortable via the currents $i_{sd}$ and/or $i_{sq}$. And stator voltages as control variables $u = [V_{sd} \ V_{sq}]$.

The principal objective of the backstepping controller is to regulate the speed of the PMSM drive to its reference value $\Omega_{ref}$ whatever external disturbances. We assume that the engine parameters are known and invariant.

### 3.1. Backstepping Speed Controller

The first step is defined the tracking errors:

$$e_\Omega = \Omega_{ref} - \Omega$$

The derivative of (3) is:

$$\dot{e_\Omega} = \frac{de_\Omega}{dt} = \dot{\Omega}_{ref} - \dot{\Omega}$$

$$= \dot{\Omega}_{ref} - \frac{1}{J} \left[ \frac{3p}{2} (\Phi_f i_q + (L_{sd} - L_{sq}) i_d i_q) - f \Omega + C_r \right]$$

We define the following quadratic function:

$$V_1 = \frac{1}{2} e_\Omega^2$$

Its derivative along the solution of (5), is given by:

$$\dot{V_1} = e_\Omega \dot{e_\Omega}$$

$$= e_\Omega \left( \dot{\Omega}_{ref} - \frac{1}{J} \left[ \frac{3p}{2} (\Phi_f i_q + (L_{sd} - L_{sq}) i_d i_q) - f \Omega + C_r \right] \right)$$

Using the Backstepping design method, we consider the d-q axes currents components $i_{sd}$ and $i_{sq}$ as our virtual control elements and specify its
desired behavior, which are called stabilizing function in the backstepping design terminology as follows:

\[
\begin{align*}
\dot{i}_{s\text{ref}} &= 0 \\
i_{s\text{ref}} &= \frac{2}{3p\Phi_f} (f\Omega + C_r + J_k\Omega e_{\Omega})
\end{align*}
\] (7)

With \(k_\Omega\) is a positive constant

Substituting (7) in (6) the derivative of \(V_1\):

\[\dot{V}_1 = -k_\Omega e_{\Omega}^2 \leq 0\] (8)

3.2. Backstepping Current Controller

We have the asymptotic stability of the origin of the system (1). We defined current following errors:

\[
\begin{align*}
e_d &= i_{s\text{ref}} - i_{sd} \quad \text{with} \quad i_{s\text{ref}} = 0 \\
e_q &= i_{s\text{ref}} - i_{sq}
\end{align*}
\] (9)

Their dynamics can be written:

\[
\begin{align*}
\dot{e}_d &= \dot{i}_{s\text{ref}} - \dot{i}_{sd} = \frac{r_s}{L_{sd}} i_{sd} - \frac{L_{sq}}{L_{sd}} p\Omega i_{sq} - \frac{V_{sd}}{L_{sd}} e_d \\
\dot{e}_q &= \dot{i}_{s\text{ref}} - \dot{i}_{sq} = \frac{2}{3p\Phi_f} (f\Omega + C_r + J_k\Omega e_{\Omega}) \\
&\quad + \frac{r_s}{L_{sq}} i_{sq} + \frac{L_{sd}}{L_{sq}} p\Omega i_{sd} + \frac{p\Phi_f}{L_{sq}} \Omega - \frac{V_{sq}}{L_{sd}}
\end{align*}
\] (10)

To analyze the stability of this system we propose the following Lyapunov function:

\[V_2 = \frac{1}{2} (e_d^2 + e_q^2 + e_q^2)\] (11)

Its derivative along the trajectories (8), (9) and (10) is:

\[\dot{V}_2 = e_d \dot{e}_d + e_q \dot{e}_q + e_q \dot{e}_q = -k_\Omega e_{\Omega}^2 - k_\Omega e_{\Omega}^2 + \] (12)

\[e_d \left[ k_d \dot{e}_d = \frac{V_{sd}}{L_{sd}} + \frac{r_s}{L_{sd}} \frac{L_{sq}}{L_{sd}} p\Omega i_{sq} + \frac{3pL_f}{2J} (L_{sd} - L_{sq}) e_d i_{sq} \right] + e_q \left[ k_e \dot{e}_q + \left( \frac{2k_e f J - f}{3p\Phi_f} \right) \frac{3p\Phi_f}{2J} e_q + \frac{3pL_f}{2J} (L_{sd} - L_{sq}) e_d i_{sq} - k_\Omega e_{\Omega} \right]
\]

\[+ \frac{3pL_f}{2J} e_q \right] + \frac{r_s}{L_{sq}} i_{sq} + \frac{L_{sd}}{L_{sq}} p\Omega i_{sd} + \frac{\Phi_f}{L_{sq}} \]

The expression (12) found above requires the following control laws:

\[
\begin{align*}
V_{sd} &= k_d L_{sd} e_d + r_s i_{sd} - L_{sq} \Omega i_{sq} + \frac{3pL_f}{2J} (L_{sd} - L_{sq}) e_d i_{sq} \\
V_{sq} &= \frac{2L_{sd}(k_d f J - f)}{3p\Phi_f} e_q + \frac{3pL_f}{2J} (L_{sd} - L_{sq}) e_d i_{sq} - k_\Omega e_{\Omega}
\end{align*}\] (13)

With this choice the derivatives of (13) become:

\[\dot{V}_2 = -k_\Omega e_{\Omega} - k_d e_d - k_e e_q \leq 0\] (14)

3.3. Simulation and test performance

In this section, we show the great improvement of performance PMSM machine, at first the results of not adaptive Backstepping control is analyzed. After, the performance of the indirect sliding mode control is discussed and compared to that before. The characteristics for the PMSM motor simulated in this experiment are follows:

\[r_s = 0.412 \Omega, \quad p = 4, \quad L_{sd} = 3.24 mH, \quad L_{sq} = 3.28 mH, \quad J = 0.0001473 Kg.m^2\]

For a trajectory \(\Omega_{\text{ref}} = 150 \text{ rad/s} at 0s, i_{sd\text{ref}} = 4.5A\) and \(C_r = 7N.m\) the following (Fig.3) shown the performance of the input output linearization control.
Follow of the trajectory

For testing the further trajectory, the speed reference was made variable. The reference of the direct component of current was set to zero. For a trajectory $\Omega_{ref}$ (=200 rad/s at 0s, =100 rad/s at t=0.3s, =250 rad/s at t=0.6s), the following (Fig.4) shown the performance of the PMSM.
Fig. 3: (a) Speed response trajectory, (b) Error Speed response, (c) abc Axis Stator Current, (d) dq Axis Stator Current, (e) d-q Axis Stator Flux, (f) Electromagnetic Torque $C_e$.

**Disturbance rejection torque load**

For a trajectory $\Omega_{\text{ref}} = \{ -250 \text{rad/s at } 0 \text{s}, 130 \text{rad/s at } t=0.4 \text{s} \}$ and $C_r=7 \text{ N.m at } t=0.4 \text{s}$, the following (Fig. 5) shown the performance of the PMSM.
The Dynamic errors and direct currents quadratic write:

\[
\frac{de_k}{dt} = \frac{di_{sd}}{dt} = -\frac{V_{sd}}{L_{sd}} + \frac{R_s}{L_{sd}} i_{sd} - \Omega \frac{L_{sd}}{L_{sd}} i_{sq} \tag{17}
\]

\[
\frac{de_q}{dt} = \frac{di_{qd}}{dt} - \frac{di_{sq}}{dt} = 2(\xi_\Omega f - \frac{3p\Phi_f}{2J}) \left( \frac{3p\Phi_f}{2} e_q + \frac{3p}{2J} (L_{sd} - L_{sq}) e_d i_{sq} - k_{\Omega e_q} e_q \right) \tag{18}
\]

\[
= \frac{2(\xi_\Omega f - \frac{3p\Phi_f}{2J})}{3p\Phi_f} \left[ \frac{3p\Phi_f}{2} e_q + \frac{3p}{2J} (L_{sd} - L_{sq}) e_d i_{sq} - k_{\Omega e_q} e_q \right]
\]

To analyze the stability of this system we propose the following Lyapunov function:

\[
V_2 = \frac{1}{2} \left( e_{\Omega}^2 + e_d^2 + e_q^2 + \frac{\tilde{C}_q^2}{\gamma_1} + \frac{\tilde{R}_q^2}{\gamma_2} + \frac{\tilde{\Phi}_q^2}{\gamma_3} \right) \tag{19}
\]

Its derivative along the trajectories (16), (17) and (18) is:

\[
\dot{V}_2 = e_{\Omega} e_q + e_d \dot{e}_d + e_q \dot{e}_q + \frac{1}{\gamma_1} \frac{\tilde{C}_q}{\gamma_1} + \frac{1}{\gamma_2} \frac{\tilde{R}_q}{\gamma_2} + \frac{1}{\gamma_3} \frac{\tilde{\Phi}_q}{\gamma_3}
\]

\[
- k_{\Omega e_q} e_q - k_f e_q - k_{\Omega e_q} e_q
\]

\[
+ e_q \left[ k_f e_q - \frac{V_{sd}}{L_{sd}} + \frac{R_s}{L_{sd}} i_{sd} - \Omega \frac{L_{sd}}{L_{sd}} i_{sq} + \frac{3p}{2J} (L_{sd} - L_{sq}) e_d i_{sq} \right]
\]

\[
+ e_d \left[ k_f e_q + \frac{2(\xi_\Omega f - \xi_\Omega f)}{3p\Phi_f} \left( \frac{3p\Phi_f}{2} e_q + \frac{3p}{2J} (L_{sd} - L_{sq}) e_d i_{sq} \right) - k_{\Omega e_q} e_q \right]
\]

\[
+ \tilde{C}_q \left[ \frac{1}{\gamma_1} \frac{\tilde{C}_q}{\gamma_1} - \frac{2k_{\Omega f}}{3p\Phi_f} e_d e_q - \frac{2k_{\Omega f}}{3p\Phi_f} e_q \right]
\]

\[
+ \tilde{R}_q \left[ \frac{1}{\gamma_2} \frac{\tilde{R}_q}{\gamma_2} + \frac{1}{L_{sq}} i_{sd} - \frac{1}{L_{sq}} e_d i_{sq} \right]
\]

\[
+ \tilde{\Phi}_q \left[ \frac{1}{\gamma_3} \frac{\tilde{\Phi}_q}{\gamma_3} - \frac{3p}{2J} e_d e_q - \frac{3p}{2J} e_q - \frac{1}{L_{sq}} e_q \right] \tag{20}
\]

The expression (16) found above requires the following control laws:

\[
V_{sd} = k_f L_{sd} e_q + \tilde{R}_q i_{sd} - L_{sq} \Omega i_{sq} + \frac{3pL_{sd}}{2J} (L_{sd} - L_{sq}) e_d i_{sq} \tag{21}
\]

\[
V_{sq} = \frac{2L_{sq}(k_{\Omega f} - f)}{3p\Phi_f} \left( \frac{3p\Phi_f}{2J} e_q + \frac{3p}{2J} (L_{sd} - L_{sq}) e_d i_{sq} - k_{\Omega e_q} e_q \right)
\]

\[
+ \frac{3p\Phi_f}{2J} e_q + \tilde{R}_q i_{sq} + L_{sd} \Omega i_{sd} + \tilde{\Phi}_f + k_f L_{sq} e_q
\]

Therefore the dynamics of the Lyapunov function can be simplified as follows:

4. Nonlinear adaptive Backstepping approach Control

4.1. Principle

In the previous section, the control laws are developed under the assumption that the machine parameters are known and invariants. This assumption is not always true. In fact, the flow created by the magnet varies with increasing temperature and with the fields created by the stator currents. Stator resistance also varies with temperature. Also, the change in operating conditions is implicitly load torque and hence the coefficient of friction and inertia. Adaptive Backstepping version takes into account the variations of these parameters.

In equation (7), we do not know exactly the value of the load torque \(C_r\), it will be replaced by its estimate \(\tilde{C}_r\) :

\[
i_{sqref} = \frac{2}{3p\Phi_f} (f \Omega + \tilde{C}_r + J k_{\Omega e_q} e_q) \tag{15}
\]

From (13) and (15), we deduce the dynamics of the speed error as follows:

\[
\frac{de_q}{dt} = \frac{1}{J} \left[ \tilde{C}_r + \frac{3p\Phi_f}{2} e_q + \frac{3p}{2} (L_{sd} - L_{sq}) e_d i_{sq} - J k_{\Omega e_q} e_q \right] \tag{16}
\]

With \(\tilde{C}_r = \tilde{C}_r - C_r\), is the error of the estimated load torque.
\[ \dot{V}_2 = -k_{\Omega} e_{\Omega}^2 - k_d e_d^2 - k_q e_q^2 \]
\[ + \ddot{C}_r \left[ \frac{1}{\gamma_1} \ddot{C}_r - \frac{2 k_{\Omega} e_{\Omega}}{3 p \Phi_f} + \frac{2 f_e q}{3 p \Phi_f} - \frac{e_\Omega}{J} \right] \]
\[ + \ddot{R}_s \left[ \frac{1}{\gamma_2} \ddot{R}_s + \frac{1}{L_{sd}} e_j - \frac{1}{L_{sq}} e_j \right] \]
\[ + \ddot{\Phi}_f \left[ \frac{1}{\gamma_3} \ddot{\Phi}_f - \frac{3 p}{2 f_j} e_q - \frac{k_{\Omega} J - f}{J \Phi_f} e_j - \frac{1}{L_{sq}} \Omega e_q \right] \]

Hence the adaptation laws as follows:
\[ \dot{\ddot{C}}_r = \gamma_1 \left[ \frac{2 k_{\Omega} e_{\Omega}}{3 p \Phi_f} - \frac{2 f_e q}{3 p \Phi_f} + \frac{e_\Omega}{J} \right] \]
\[ \dot{\ddot{R}}_s = \gamma_2 \left[ \frac{1}{L_{sd}} e_j - \frac{1}{L_{sq}} e_j \right] \]
\[ \dot{\ddot{\Phi}}_f = \gamma_3 \left[ \frac{3 p}{2 f_j} e_q + \frac{k_{\Omega} J - f}{J \Phi_f} e_j + \frac{1}{L_{sq}} \Omega e_q \right] \]

With this choice, the expression (19) becomes:
\[ \dot{V}_2 = -k_{\Omega} e_{\Omega}^2 - k_d e_d^2 - k_q e_q^2 \leq 0 \]

So the system is globally asymptotically stable in the presence of parametric uncertainties.

### 4.2. Simulation and test performance

The following results are obtained by choosing the following values:
- Gains of the control law: \( k_{\Omega} = 0.15 \), \( k_d = 0.01 \), \( k_q = 0.01 \).
- Adaptation gains: \( \gamma_1 = 0.15 \), \( \gamma_2 = 0.01 \), \( \gamma_3 = 0.015 \).
- **Follow of the trajectory**
Fig. 7: Test performance of the adaptive controller for trajectory tracking
(a) Speed response trajectory (b) Error Speed response (c) d-q axis current without uncertainties
(d) abc axis current

Disturbance rejection

Fig. 8: Test performance of the adaptive controller for rejecting disturbance torque load applied at \( t = 0.3 \) s.
(a) Speed response trajectory (b) d-q axis current without uncertainties (c) Electromagnetic Torque
Parametric uncertainties

Fig. 9: Test performance of the adaptive controller following a change in $R_s$
(a) Speed response trajectory (b) d-q axis current without uncertainties (c) Electromagnetic Torque (d) current $i_{sa}$

Fig. 10: Test performance of the adaptive controller following a change in $\Phi_f$
(a) Speed response trajectory (b) d-q axis current without uncertainties
Fig. 11: Test performance of the adaptive controller following a change in L_{sd} and L_{sq} 
Speed response trajectory (b) d-q axis current without uncertainties

5. FPGA-Based Implementation of an Robust Backstepping Control System

5.1. Development of the Implementation

There are several manufacturers of FPGA components such: Actel, Xilinx and Altera...etc. These manufacturers use different technologies for the implementation of FPGAs. These technologies are attractive because they provide reconfigurable structure that is the most interesting because they allow great flexibility in design. Nowadays, FPGAs offer the possibility to use dedicated blocks such as RAMs, multipliers wired interfaces PCI and CPU cores. The architecture designing was done using with CAD tools. The description is made graphically or via a hardware description language high level, also called HDL (Hardware Description Language). Is commonly used language VHDL and Verilog. These two languages are standardized and provide the description with different levels, and especially the advantage of being portable and compatible with all FPGA technologies previously introduced [7].

In this paper an FPGA XC3S500E Spartan3E from Xilinx is used. This FPGA contains 400,000 logic gates and includes an internal oscillator which issuer a 50MHz frequency clock. The map is composed from a matrix of 5376 slices linked together by programmable connections.

The simulation procedure begins by verifying the functionality of the control algorithm by trailding a functional model using Simulink’s System Generator for Xilinx blocks. For this application, the functional model consists in a Simulink times discretiret model of the No adaptive Backstepping algorithm associated with a voltage inverter and PMSM model.

The Fig. 12 summarizes the different steps of programming an FPGA. The synthesizer generated with CAD tools first one Netlist which describes the connectivity of the architecture. Then the placement-routing optimally place components and performs all the routing between different logic. These two steps are used to generate a configuration file to be downloaded into the memory of the FPGA. This file is called bitstream. It can be directly loaded into FPGA from a host computer.
### 5.2. Simulation Procedure

The simulation procedure begins by verifying the functionality of the control algorithm by trailing a functional model using Simulink’s System Generator for Xilinx blocks. For this application, the functional model consists in a Simulink time discreted model of the No adaptive Backstepping algorithm associated with a voltage inverter and PMSM model. The Fig.12 shows in detail the programming of the control shown in Fig.2 in the SYSTEM GENERATOR environment from Xilinx, we will implement it later in the memory of the FPGA for the simulation of PMSM.

**Fig.13: Functional Model for No adaptive Backstepping Controller from SYSTEM GENERATOR**

The functional model for no adaptive Backstepping Controller from SYSTEM GENERATOR is composed the different blocks:
- The block encoder interface IC allows the adaptation between the FPGA and the acquisition board to inquity the rotor position of the PMSM,
- The ADC interface allows the connection between the FPGA and the analog-digital converter (ACD54746MSPS 12-bit A / D) that will be bound by the following two Hall Effect transducers for the acquisition of the stator currents machine,
- The blocks of coordinate’s transformation: the transformation of Park Inverse (abc-to-dq),
- The blocks of coordinate’s transformation: the transformation of Park (dq-to-abc),
- The SVW block is the most important, because can provide control pulses to the IGBT voltage inverter in the power section from well-regulated voltages,
- The block for the controller no adaptive Backstepping which is designed to regulate the speed and stator currents of the PMSM,
- Block "Timing" which controls the beginning and the end of each block, which allows the refresh in the voltages reference $V_{10}$, $V_{20}$ and $V_{30}$ at the beginning of each sampling period,
- The RS232 block allows signal timing and recovery of signals viewed, created by another program on Matlab & Simulink to visualize the desired output signal.
The second step of the simulation is the determination of the suitable sampling period and fixed point format. The Fig.14 gives the specification model of the abc-to-dq (park) transformation.

For example, we present the construction of Block Clark's Transformation in the system generator environment from Xilinx, which is characterized by the following system (18):

\[
\begin{bmatrix}
V_{dc} \\
V_{aq} \\
V_{dq}
\end{bmatrix} =
\begin{bmatrix}
\cos\theta & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta - \frac{4\pi}{3}) \\
-\sin\theta & -\sin(\theta - \frac{2\pi}{3}) & -\sin(\theta - \frac{4\pi}{3}) \\
\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}
\end{bmatrix}
\begin{bmatrix}
V_{sa} \\
V_{sb} \\
\theta
\end{bmatrix}
\] (26)

The specification model is then used for the definition of the corresponding Data Flow Graph. The Fig.14 shows the DFG corresponding to the Park transformation module.

5.3. Prototyping platform

To test the FPGA based controller, a prototyping platform for the control of a Permanent magnet Synchronous Machine was assembled (Fig.15).

The implementation of the indirect control by sliding mode on FPGA devices is characterized by a reduced operation time.

The Fig.16 shows the experimental results of Indirect Sliding Mode PMSM with the FPGA platform are shown. Update frequency for this implementation is 20 kHz. All results were extracted from the FPGA by the ChipScope tool of Xilinx.
The Experimental results show the performance of PMSM machine, using two approaches control nonlinear. This two control algorithms show the robustness and efficacité of the system.

7. Conclusion

In this paper a robust continuous approaches Nonlinear not Adaptive Backstepping Control and Adaptive Backstepping Control strategy for permanent-magnet synchronous motor (PMSM) drive systems is presented. The FPGA based implementation is detailed, a bench test was realized by a prototyping platform, the experimental results obtained show the effectiveness and the benefit of our contribution and the different steps of implementation for the control FPGA.

Acknowledgment

We thank all those who contributed to make this work, including my teachers in my home laboratory STIC Team and the Center of Modeling and simulation of the systems, my wife and all my friends for their support. This work was performed in the Montefiore Institute, University of Liege (ULG) in Belgium.

References


