*Scientific Paper / Artículo Científico*

UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR
INGENIUS
Revista de Ciencia y Tecnología
https://doi.org/10.17163/ings.n23.2020.09
pISSN: 1390-650X / eISSN: 1390-860X

# Speed control of a permanent magnet synchronous motor actuated by a three-phase multi level inverter

# Control de velocidad de un motor síncrono de imanes permanentes accionado por un inversor trifásico multinivel

Rosalino Mayoral Lagunes[1], José Antonio Juárez Abad[2], Beatriz Angélica Aguilar López[1], Jesús Linares Flores[2], Jorge Luis Barahona Ávalos[2]

## Abstract

This paper presents the design and FPGA embedded implementation of robust controller design to speed tracking problem for a Permanent Magnet Synchronous Motor (PMSM). Then, a linear controller based on the exact static error dynamics passive output feedback (ESEDPOF) is proposed, where the uncertainty estimation is taken into account. The technique of passivity requires knowing the load torque, this is estimated with a traditional reduced-order observer. PMSM is driven by a five levels Three-Phase Cascaded Cell Multi-Level Inverter (3Φ-CCMLI). The medium-scale field-programmable gate array (FPGA) Spartan-6 XC6SLX9 is used for implementing the ESEDPOF controller, the reduced-order observer, and the multilevel pulse width modulator. The parallel processing provided by these devices allowed to obtain a sampling time of 10us. Simulation and Experimental validation shows an excellent dynamical performance.

***Keywords***: PMSM, Passivity Based Control, Multi Level Inverter, FPGA.

## Resumen

Este trabajo presenta el diseño e implementación de un controlador robusto para el seguimiento de velocidad de un motor síncrono de imanes permanentes (MSIP). Se propone un controlador lineal basado en la retroalimentación dinámica de la salida pasiva estática del error exacto. El controlador pasivo propuesto requiere del conocimiento del par de carga, por lo que el mismo es estimado con un observador tradicional de orden reducido. El MSIP es impulsado por medio de un inversor multinivel trifásico de celdas en cascada de cinco niveles. Para la implementación del controlador, estimador y modulador multinivel se emplea un arreglo de compuertas programable en campo (FPGA) de la familia Spartan-6 XC6SLX9. El procesamiento en paralelo que provee este dispositivo permite obtener un tiempo de muestreo de 10 us. Los resultados de simulación y experimentales muestran que el controlador propuesto tiene un excelente desempeño.

***Palabras clave***: PMSM, control por pasividad, inversor multinivel, FPGA

[1,*]Postgraduate Studies Department, Universidad Tecnológica de la Mixteca – México.
http://orcid.org/0000-0003-3420-7416, http://orcid.org/0000-0002-2769-4992.
[2]Institute of Electronics and Mechanics, Universidad Tecnológica de la Mixteca – México,
Corresponding author ✉: jbarahona@mixteco.utm.mx. http://orcid.org/0000-0001-7440-0849,
http://orcid.org/0000-0002-5723-4786, http://orcid.org/0000-0002-5502-6692.

Received: 22-11-2019, accepted after review: 13-12-2019

Suggested citation: Mayoral Lagunes, R.; Juárez Abad, J. A.; Aguilar López, B. A.; Linares Flores, J. and Barahona Ávalos, J. L. (2020). «Speed control of a permanent magnet synchronous motor actuated by a three-phase multi level inverter». Ingenius. N.° 23, (january-june). pp. 97-108. doi: https://doi.org/10.17163/ings.n23.2020.09.

## 1. Introduction

Electrical motors are actuators that are used in most of the industrial machinery and household appliances; their position and/or speed are controlled so that they can fulfill specific tasks, such as: Computer Numerical Control (CNC) machines, belt conveyors, robotics and other industrial processes. Among these applications it is common to find permanent magnet synchronous motors (PMSM), which have better precision in terms of speed and position due to their property of synchrony. Recently, controllers that enable modifying those magnitudes are implemented using reprogrammable and reconfigurable devices, such as digital signal processors (DSP) and field-programmable gate arrays (FPGA) [1, 2].

Different control schemes have been proposed for these type of motors, for instance the passivity-based control shown in [3], and a controller based on the Backstepping technique for position tracking, which is shown in [3].

The PMSM must be driven by a three-phase inverter, which must preferably provide a voltage wave with the lowest possible amount of harmonics, because otherwise some well-known damages may occur in both the electrical and mechanical parts of the motor [4]. A three-phase multilevel inverter of five levels is constructed in this work, with which the harmonics content is reduced thus improving the quality of the voltage supplied to the motor [5]. An additional feature of this type of converter is that it handles more power, since such power is distributed in the series arrays that constitute the commutation devices [6–8].

The rest of the work is structured in the following manner. Section 2 briefly describes the system to be controlled, and then section 3 describes the centered methodology employed for the implementation of processing systems in FPGA, taken from [9–13]; such methodology possesses two phases which are covered in sections 4 and 5. Afterwards, section 6 describes the construction of the power converter employed, in this case the three-phase multilevel inverter of five levels, and at last section 7 presents the experimental results obtained.

## 2. Description of the system

A passivity-based control is implemented for soft tracking the speed of a permanent magnet synchronous motor, driven by a three-phase inverter of cascade topology of 5 levels. This is carried out using a device of reconfigurable logic.

Figure 1 shows the scheme of the proposed system. The PMSM is driving an unknown mechanical load, and the motor power is supplied by the three-phase inverter of 5 levels. The passivity-based controller, the signal processing and the PS-PWM modulator are implemented in a FPGA.

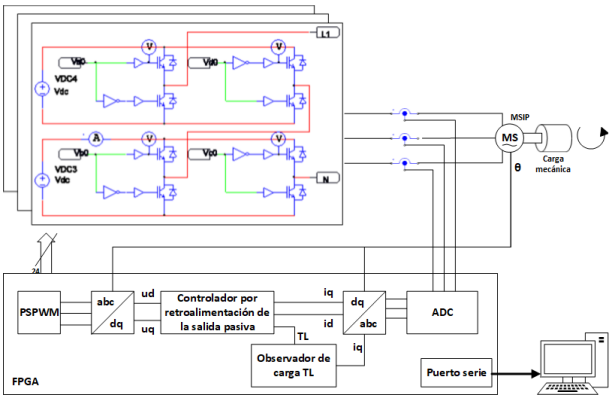


**Figure 1.** General scheme of the system.

## 3. Methodology

According to the methodology employed, taken from [7], this work is divided in 2 phases. In general, the first phase has to do with the modeling of the system, and the analysis of the mathematical properties of this model to formulate the controller of the system. The second phase deals with the implementation of the control algorithm formulated in phase 1, and the complementary modules which will be implemented in the reconfigurable logic device. Each of these phases is described with more detail in the following.

Phase 1 is graphically summarized in Figure 2. This phase is independent of the device to be used in the implementation, and in turn is divided in three stages:

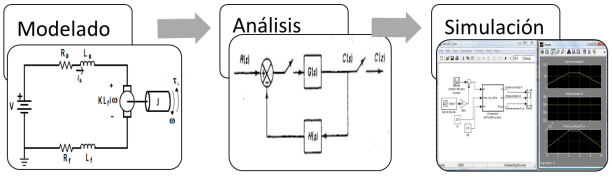


**Figure 2.** Phase 1 of the methodology.

1. **Modeling:** In this stage the PMSM and the three-phase multilevel are mathematically modelled, using the laws and theorems necessary to obtain the differential equations of the system.

2. **Analysis:** The dynamic models of the combination inverter-motor are analyzed, the proposed controller is designed by means of algebraic tools, and the feasibility of the complete system is determined.

3. **Simulation:** In this stage the obtained mathematical models are simulated, to confirm the

approximation to the real system; the MATLAB-Simulink software together with the software PSIM are utilized for this purpose.

Phase 2 of the proposed design methodology is summarized in Figure 3, and comprises 7 stages. This proposal is a balanced solution between two opposing requirements: 1) a friendly method that perfectly adapts to a control engineer who is not an expert in digital design and 2) obtain a good performance of the control system [9]. The aforementioned stages are described in the following:

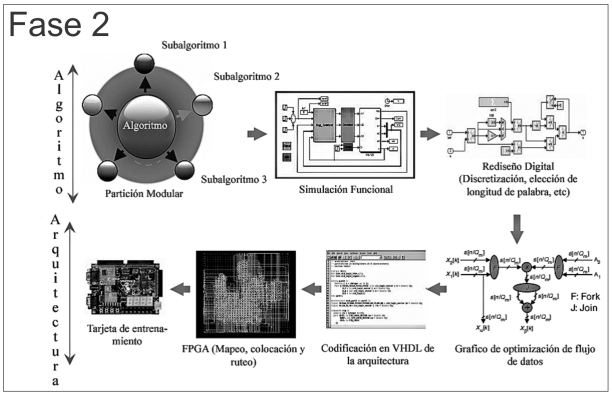


**Figure 3.** Phase 2 of the methodology. [9]

1. **Modular partition:** Consists of dividing the algorithm in reusable blocks that make sense from the functional point of view, in other words, the control algorithm and the modulator of the inverter are divided in modules, and those that may be reusable such as operations, coders, search tables, among others, are identified. It is sought to divide the design in smaller parts that may be manageable.

2. **Functional simulation:** In this stage the controller is validated as a function of continuous-time blocks using, for instance, the MATLAB-Simulink software.

3. **Digital redesign:** This stage is very important, since it includes the realization of the digital filter, the selection of the sampling time, the selection of the format of the coefficients and variables according to the control requirements and constraints, such as: word length, and fixed or floating point, among others.

The final four stages are for the implementation based on FPGA, and consist of the following:

4. **Optimization graph of the data flow:** In this stage, the data flow of the algorithm is modeled to have the best balance in terms of time/area; for this purpose, a graphical representation of every module of the algorithm is obtained. The graphical representation is constituted by lines and nodes, each line corresponding to a data transfer and the nodes representing arithmetical operations, or logic or mathematical functions.

5. **HDL coding:** the graphic of data flow of the previous stage is transcribed to VHDL code; regularly, the data transfers are controlled by finite states machines, synchronized at the clock signal of the FPGA.

6. **Implementation in FPGA:** This is an automatic process, regularly carried out with the tool provided by the manufacturer of the device. In this stage a mapping of the resulting functions, placement and routing is performed.

7. **Experimental validation:** Consists of integrating the whole system, carrying out the performance tests, and adjusting all the necessary parameters until obtaining the desired results.

## 4. Phase 1 of the methodology

### 4.1. Control by DFSPOEE, modeling and analysis

Starting with the $d-q$ model obtained from the equations of the original mathematical model presented in [14], the input voltages $u = [u_d u_q]$ and input currents $i = [i_d i_q]$ in the coordinate axes $d-q$ are calculated. For the design of the controller it is supposed that the angular speed and the load torque are known. To start with the design of the controller by passive output feedback, it is considered the following model in the coordinate axes $d-q$ [15]:

$$\begin{aligned} L\frac{di_d}{dt} &= -r_e i_d + L i_q n_p \omega + u_d \\ L\frac{di_d}{dt} &= -r_e i_q + L i_d n_p \omega - K_m \omega + u_q \\ J\frac{d\omega}{dt} &= \frac{3}{2} K_m i_q - D\omega - T_L \end{aligned} \tag{1}$$

Representing the system of equations (1) in matrix form and rewriting it in passive form, yields [16]:

$$\begin{aligned} A\dot{x} = [J(y) - R]\, x + Bu - N\eta, x \in R^3, u \in R^2 \\ y_1 = i_d \\ y_2 = i_q \end{aligned} \tag{2}$$

where:

$$A = diag\begin{pmatrix} L & L & \frac{2}{3}J \end{pmatrix}; N = diag\begin{pmatrix} 1 & 1 & 1 \end{pmatrix} \tag{3}$$

$$J(y) = \underbrace{\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -k_m \\ 0 & k_m & 0 \end{pmatrix}}_{=J_0} + y \underbrace{\begin{pmatrix} 0 & n_pL & 0 \\ -n_pL & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}}_{=yJ_1} \quad (4)$$

$$R = \begin{pmatrix} r_e & 0 & 0 \\ 0 & r_e & 0 \\ 0 & 0 & \frac{2}{3}B \end{pmatrix}; B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}; \eta = \begin{pmatrix} 0 \\ 0 \\ \frac{2}{3}T_L \end{pmatrix} \quad (5)$$

The state vector is constituted by:

$$x^T(t) = (i_d, i_q, \omega) \quad (6)$$

$J_0$ and $J_1$ are constant and antisymmetric matrices, y is a scalar representing the output y of the system which, in addition, is one of the variables of the state vector $x$; $R$ is a positive semidefinite symmetric matrix, $B$ is a constant matrix of the control inputs, and finally $\eta$ is a vector of load torques. The definitions of passivity and dissipativity discussed in [15]] are used to verify that the average nonlinear system of the PMSM given in (1) is passive.

Let $u^* \in \mathbb{R}^2$ denote the nominal input trajectory corresponding to the trajectory of the nominal state vector $x^*(t) \in \mathbb{R}^3$. Starting from this, the reference dynamics of the system is constructed, which must satisfy the following expression:

$$A\dot{x}^* = [J(y^*) - R]x^* + Bu^* - N\eta^* \quad (7)$$

It is defined the tracking error $e = x - x^*$, the error of the control input $e_u = u - u^*$, and the error of the disturbance input $e_\eta = \eta - \eta^*$, and considering that $J(y) = J_0 + yJ_1$, then:

$$A\dot{e} = J(y)e - Re + Be_u + J_1x^*l_3e - Ne_\eta$$

Defining $M_3 = J_1x^*l_3$ as the 3×3 matrix given

$$M_3 = \begin{pmatrix} 0 & n_pL & 0 \\ -n_pL & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} i_d^* \\ i_q^* \\ \omega \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}$$

$$= n_pL \begin{pmatrix} 0 & 0 & i_d^* \\ 0 & 0 & -i_q^* \\ 0 & 0 & 0 \end{pmatrix}$$

Afterwards, matrix $M_3$ is decomposed in a symmetric and an antisymmetric matrix using matrix algebra, having as result

$$M_3 = \frac{1}{2}n_pL \underbrace{\begin{pmatrix} 0 & 0 & i_d^* \\ 0 & 0 & i_q^* \\ -i_d^* & i_q^* & 0 \end{pmatrix}}_{W_3} + \frac{1}{2}n_pL \underbrace{\begin{pmatrix} 0 & 0 & i_d^* \\ 0 & 0 & i_q^* \\ -i_d^* & i_q^* & 0 \end{pmatrix}}_{S_3}$$

Subsequently $W_3$ is added to the conservative matrix $J(y)$, and $S_3$ is added to the dissipative part $R$. With this operation it is found the dynamic equation of the tracking error, expressed as:

$$A\dot{e} = J^*e - R^*e + Be_u \quad (8)$$

If it is considered that $R^* > 0$, in other words, positive definite, then the following condition is satisfied:

$$R^* = \begin{pmatrix} r_e & 0 & -\frac{1}{2}n_pLi_q^* \\ 0 & r_e & \frac{1}{2}n_pLi_d^* \\ -\frac{1}{2}n_pLi_q^* & \frac{1}{2}n_pLi_d^* & \frac{2}{3}B \end{pmatrix} \quad (9)$$

Using the Sylvester criterion it is verified that $R^* > 0$, namely, that it is a positive definite matrix, it must hold that

$$det(R^*) = \frac{2}{3}r_eB - \frac{1}{4}n_p^2L^2\left[(i_d^*)^2 + (i_q^*)^2\right] > 0 \quad (10)$$

Following with the design methodology of the controller, using Lyapunov, it is proposed that $e_u$ como $e_u = -\delta B^T e$, where:

$$\delta = \begin{pmatrix} \delta_1 & 0 \\ 0 & \delta_2 \end{pmatrix} > 2 \qquad \delta_1, \delta_2 > 0 \quad (11)$$

with $\delta$ a positive matrix, such that

$$\begin{pmatrix} u_d \\ u_q \end{pmatrix} = \begin{pmatrix} u_d^* \\ u_q^* \end{pmatrix} - \begin{pmatrix} \delta_1 & 0 \\ 0 & \delta_2 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} x^* \quad (12)$$

From equation (12) the control laws that will act on the system are obtained, yielding:

$$u_d = \widetilde{u_d} - \delta_1(i_d - \widetilde{i_d}) \qquad u_q = \widetilde{u_q} - \delta_2(i_q - \widetilde{i_q}) \quad (13)$$

#### 4.1.1. Equilibrium points

The equilibrium points will be used to calculate the desired reference variables $i_d^*$, $i_q^*$, $u_d^*$, $u_d^*$, $w^*$. For this case, $w^*$ is the desired value of the speed which will be measured in the motor shaft. To obtain these references, the derivatives of equations (1) are set equal to zero, thus obtaining the following expressions:

$$0 = L\widetilde{i_q}n_p\omega_d + u_d \quad (14)$$

$$0 = -r_e\widetilde{i_q} - K_m\omega_d + u_q \quad (15)$$

$$0 = \frac{3}{2}K_m\widetilde{i_q} - D\omega_d - T_L \quad (16)$$

from which it can be solved:

$$\begin{aligned} \widetilde{i_d} &= 0 \\ \widetilde{i_q} &= \frac{2}{3}\frac{D}{K_m}\omega_d + \frac{2}{3}\frac{T_L}{K_m} \\ \widetilde{u_d} &= -L\widetilde{i_q}n_p\omega_d \\ \widetilde{u_q} &= r_e\widetilde{i_q} + K_m\omega_d \end{aligned} \tag{17}$$

#### 4.1.2. Design of the observer of the load torque parameter

To design and develop and efficient and robust controller, it is necessary to know all the variables that disturb the system. Since the controller by feedback of the error passive output is not robust in the presence of disturbances of constant load torque, a reduced order observer is designed to estimate the load torque parameter together with the unmolded friction terms [17]. The quadrature current $i_q$ is defined as the control input, $u_1 = i_q$, and the estimated angular speed is defined as the measured output, $y_1 = \omega$. The nominal values of the parameters $J$ and $K_m$ are considered known. Substituting variables $u_1$ and $y_1$ in the mechanical equation given in (1), yields the following expression:

$$J\frac{dy_1}{dt} = \frac{3}{2}k_m u_1 - By_1 - T_L \tag{18}$$

The following observer of load torque is proposed as:

$$\frac{\widetilde{T_L}}{dt} = \lambda(T_L - (\widetilde{T_L}) \tag{19}$$

where $\lambda > 0$. The dynamics of the estimation error is given by:

$$e_{T_L} = T_L - \widetilde{T_L} \tag{20}$$

$$\frac{de_{T_L}}{dt} = \frac{dT_L}{dt} - \frac{de_{T_L}}{dt} \tag{21}$$

If $T_L$ is considered as constant, $\frac{dT_L}{dt} = 0$, and substituting (19) in (21) results in the following dynamics of the estimation error:

$$\frac{de_{T_L}}{dt} = -\lambda e_{T_L} \tag{22}$$

Choosing a gain $\lambda > 0$, the observing error $e_{TL}$ converges exponentially to cero as $t \to \infty$. Now, substituting equation (18) in (19) gives:

$$\frac{d\widetilde{T_L}}{dt} = \lambda\left[\frac{3}{2}K_m i_q - \left(By_1 + J\frac{dy_1}{dt}\right)\widetilde{T_L}\right] \tag{23}$$

Solving and rearranging the previous equation yields the following expression:

$$\frac{d\widetilde{T_L}}{dt} + \lambda J\frac{dy_1}{dt} = \frac{3}{2}K_m u_1 - \lambda By_1 - \lambda\widetilde{T_L} \tag{24}$$

Making the change of variable $\Psi = \widetilde{T_L} + \lambda J y_1$, results in the following reduced order observer:

$$\frac{d\Psi}{dt} = -\lambda\Psi + (J\lambda - B)\lambda y_1 + \frac{3}{2}\lambda K_m i_q \tag{25}$$

$$\widetilde{T_L} = \Psi - \lambda J y_1 \tag{26}$$

The estimated value of the load torque parameter together with the unknown friction terms $\widetilde{T_L}$, adapt online to the control law for soft tracking the angular speed. The purpose of this adaptation is to reduce the effects produced by the load applied to the shaft of the PMSM.

### 4.2. Simulations of the controller in MATLAB

With the use of the MATLAB-Simulink software, it is proceeded to design the whole system in block diagram. Figure 4 shows the main block diagram.

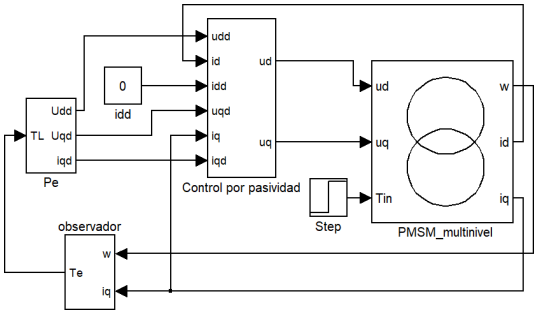


**Figure 4.** Main block diagram.

The passivity-based controller shown in Figure 4 is based on the set of equations (12); Figure 5 shows the implementation in blocks.

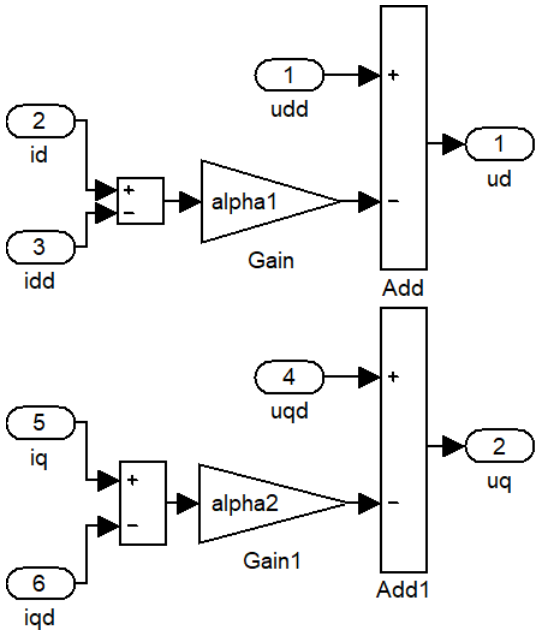


**Figure 5.** Passivity-based controller.

The load observer of equations (25) and (26) is shown in blocks in Figure 6.

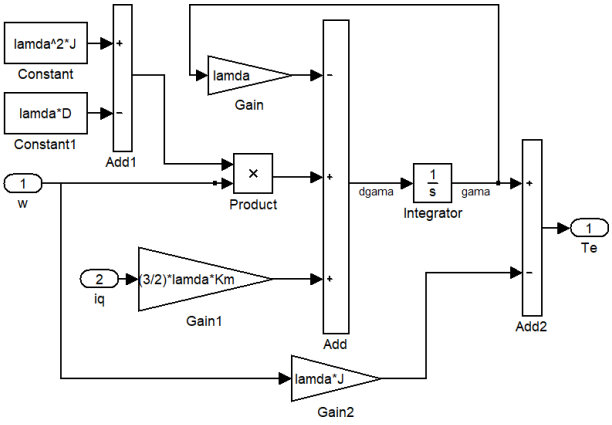


**Figure 6.** Reduced order load observer.

Equations (17) of the equilibrium points of the system, which are utilized to find the references and desired trajectories, are shown in blocks in Figure 7.

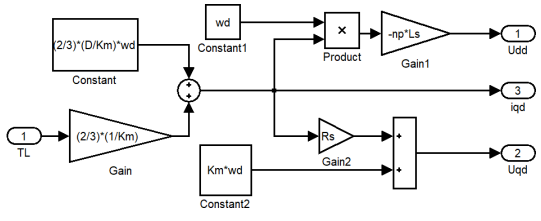


**Figure 7.** Reference signals.

A cosimulation with Simulink and Psim is carried out to validate the operation of the system. The electric diagram in Psim of the multilevel inverter in conjunction with the PMSM, is shown in Figure 8. It can be observed that the multilevel inverter is constituted by three sections, equal to the one shown in the square at the left of Figure 8.

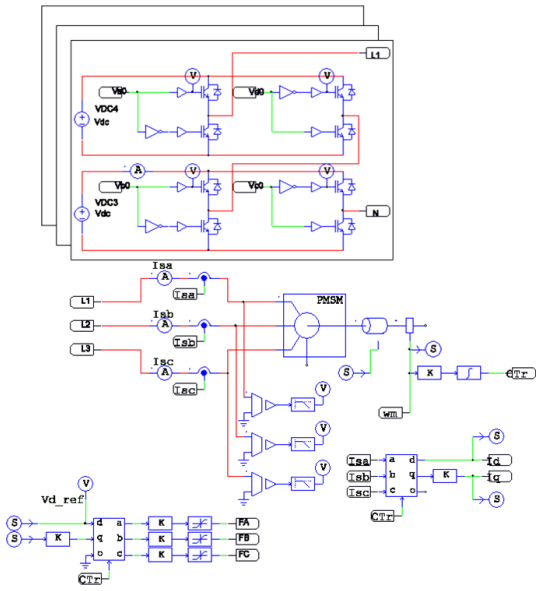


**Figure 8.** Electric diagram in Psim.

### 4.3. Multilevel inverter

For this work, a multilevel inverter with cascaded cells is utilized. This configuration is based on cells connected in series to add up the voltages, and obtain the 5 desired levels as shown in Figure 9.

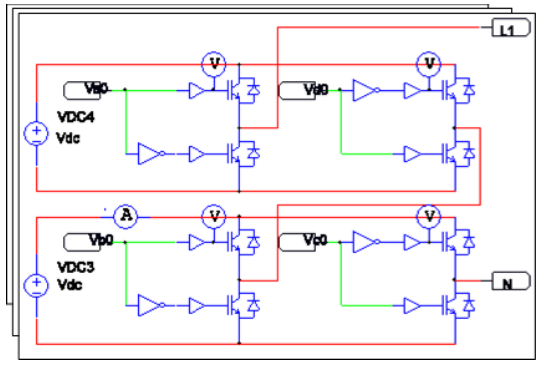


**Figure 9.** Topology of the selected cascade inverter of five levels.

The modulation technique employed is PS-PWM, which for the case of the inverter of 5 levels, consists of four triangular signals called carriers, 90° out of phase between them, i.e., 0°, 90°, 180° and 270°, respectively; in this case the carriers are generated at a frequency of 12 kHz. These signals are compared with a sinusoidal signal called modulating signal; for the case of interest, such modulating signal is generated by the transformation dq-abc. In general, its operation consists in obtaining logic ones and zeros by means of the comparison between the carrier and modulating signals: a

logic one is obtained if the carrier signal is greater than or equal to the modulating signal, otherwise a logic zero arises. In this way, PWM signals that activate the commutation devices are generated. It should be pointed out that each phase contains a modulating signal 120° out of phase with respect to the other phases; therefore, there is a total of 24 PWM signals, 12 main channels with their corresponding 12 complementary signals. Figure 10 shows the diagram to generate the PS-PWM modulation for a single phase, with its four main signals.

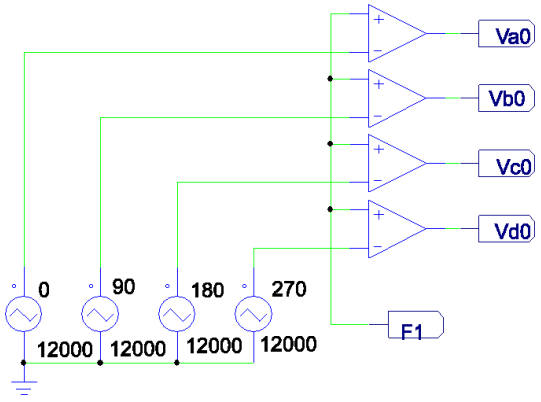


**Figure 10.** PS-PWM modulator for one phase.

### 4.4. Simulation results

Hereunder, the results obtained from the cosimulation of the system in different scenarios are shown, with the purpose of performing a comparison. These scenarios are: the Psim model of the PMSM driven by a conventional three-phase inverter, and finally driven by the three-phase multilevel inverter of five levels. The designed passivity-based controller is applied in both cases. As a result of the simulation, the plot in Figure 11 shows the angular speed of the motor driven by both types of inverters. It can be observed in such plot that when a change of the load torque is applied on the motor shaft, at time t = 1 s, the disturbance caused by such change is effectively counteracted by the controller. There is no significant difference between one type of inverter and the other.

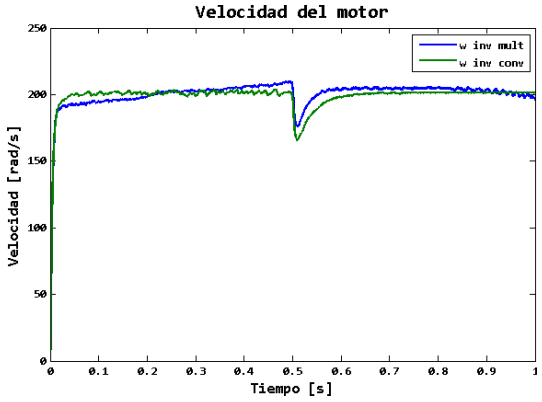


**Figure 11.** Motor speed.

Figures 12 and 13 show plots of the currents $i_d$ and $i_q$, respectively, for the aforementioned conditions. It is important to point out that with the use of the multilevel inverter the ripple in the currents decreases considerably, due to the combined use of such inverter and the designed controller.

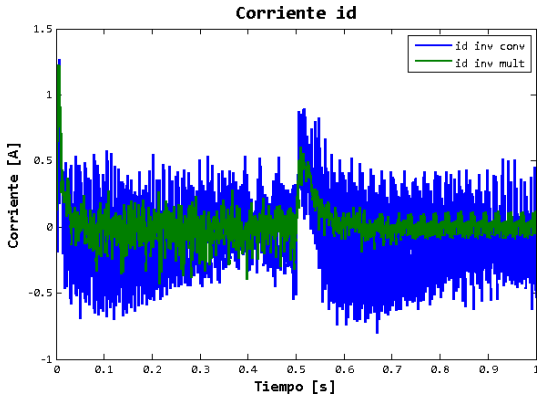


**Figure 12.** $i_d$ current.

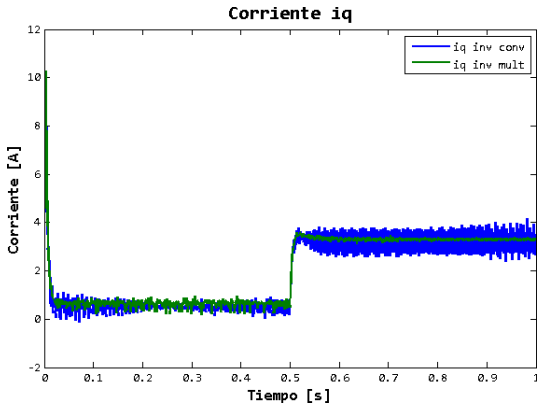


**Figure 13.** $i_q$ current.

At last, Figure 14 shows the estimated load torque. It can be seen that the designed observer equally works better in the case of the multilevel inverter, which verifies its correct operation.

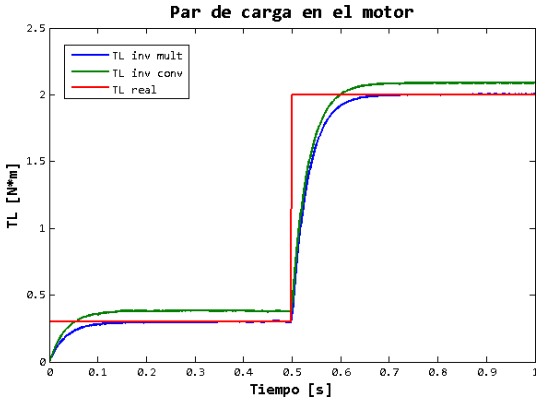


**Figure 14.** Estimated load torque of the motor.

## 5. Phase 2

The programming of the FPGA is carried out in this phase. It is worth mentioning that in this work, the coding is directly performed in the VHDL language, without employing any code generator; this approach enables making optimal use of the device resources.

### 5.1. Implementation of the PS-PWM modulator module

The modulator was programmed in VHDL language in the ISE DESIGN software of XILINX. For simplicity, Figure 15 shows the design by means of a schematic block diagram. The main programmed blocks are: block of carrier signals, in which the triangular signals are generated; block of modulating signal, which generates the modulating signal; at last, the comparators generate the PWM signals to perform the comparison between the carrier signals and the modulating signal.

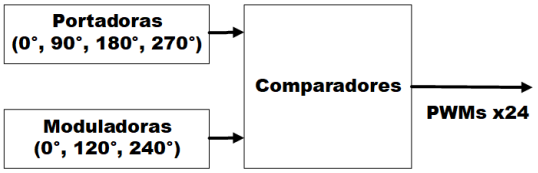


**Figure 15.** Block diagram of the modulator.

### 5.2. Implementation of the controller module

The controller is divided in various sub-modules: abc-dq transformation, dq-abc transformation, load observer, equilibrium points, passivity-based speed control, reading of current sensors, reading of position/speed sensor. These modules work with the IEEE 754 32-bit simple precision standard, which is the numerical format mostly used in hardware [18].

#### 5.2.1. dq – abc transformation module

The architecture shown in Figure 16 is implemented for the development of this module, which enables online solving equation (27) and optimizing resources. A multiplier, an adder and a register are utilized in the case of this module; although all these elements are inside the FPGA, their use should be optimized because they are limited. As observed in Figure 16, it requires 8 inputs: d, q and the 6 trigonometric functions. The outputs are the values a, b and c, which correspond to the value of voltage of the phases. Afterwards, Figure 17 shows the hardware implementation of equations (27); this methodology was taken from [19]. Since the complexity of the system makes difficult to exemplify the development, only this module is shown; however, the construction of the remaining modules is carried out in a similar manner. Table 1 shows the consumption of logic resources in the FPGA after performing the implementation of all modules involved in the design, and Figure 18 presents the highest hierarchy diagram of the system.

$$\begin{aligned} V_a &= V_d cos(\theta) + V_q sin(\theta) \\ V_b &= V_d cos(\theta - \frac{2}{3}\pi) + V_q sin(\theta - \frac{2}{3}\pi) \\ V_c &= V_d cos(\theta + \frac{2}{3}\pi) + V_q sin(\theta + \frac{2}{3}\pi) \end{aligned} \tag{27}$$

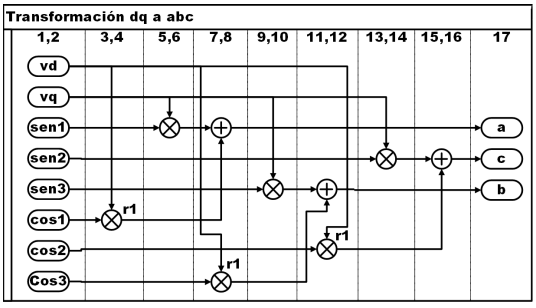


**Figure 16.** Sequence diagram of the transformation dq to abc.

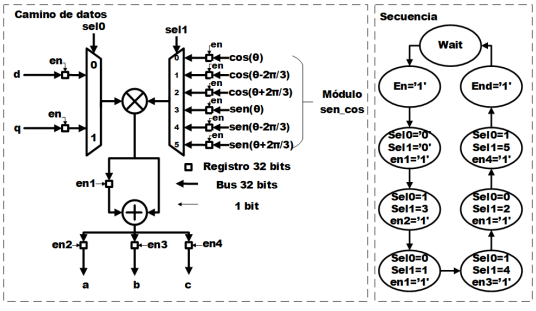


**Figure 17.** Flow diagram of data and machine of states.

**Table 1.** Level of utilization of the FPGA

| Device Utilization Summary | | | |
|---|---|---|---|
| **Slice Logic Utilization** | **Used** | **Available** | **Utilization** |
| Number of Slice Registers | 4,038 | 11,440 | 35% |
| Number of Slice LUTs | 4,834 | 5,720 | 84% |
| Number used as logic | 4,599 | 5,720 | 80% |
| Number used as Memory | 179 | 1,440 | 12% |
| Number of occupied Slices | 1,408 | 1,430 | 98% |
| Number of MUXCYs used | 1,716 | 2,860 | 60% |
| Number with an unused Flip Flop | 1,808 | 5,107 | 35% |
| Number with an unused LUT | 273 | 5,107 | 5% |
| Number of fully used LUT-FF pairs | 3,026 | 5,107 | 59% |
| Number of unique control sets | 84 | | |
| Number of slice register sites lost to control set restrictions | 245 | 11,440 | 2% |
| Number of bonded IOBs | 56 | 186 | 30% |
| Number of LOCed IOBs | 55 | 56 | 98% |
| Number of RAMB16BWERs | 17 | 32 | 53% |
| Number of RAMB8BWERs | 10 | 64 | 15% |
| Number of BUFIO2/BUFIO2_2CLKs | 1 | 32 | 3% |
| Number of BUFIO2FB/BUFIO2FB_2CLKs | 1 | 32 | 3% |
| Number of BUFG/BUFGMUXs | 5 | 16 | 31% |
| Number of DCM/DCM_CLKGENs | 1 | 4 | 25% |
| Number of DSP48A1s | 16 | 16 | 100% |
| Average Fanout of Non-Clock Nets | 3.65 | | |

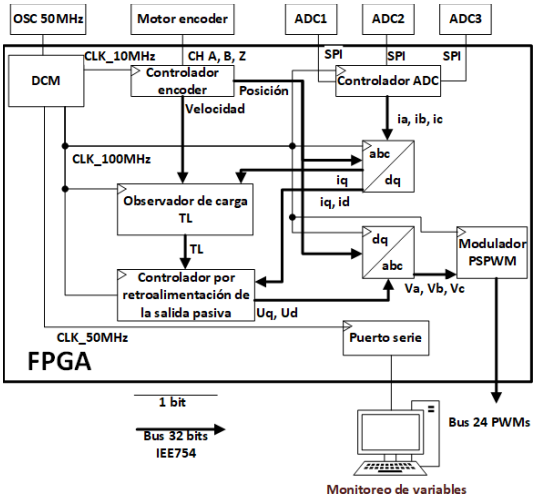


**Figure 18.** General schematic diagram of the system in the FPGA.

## 6. Experimental platform

The electrical configuration of the platform of the multilevel inverter is shown through the one-line diagram in Figure 19.

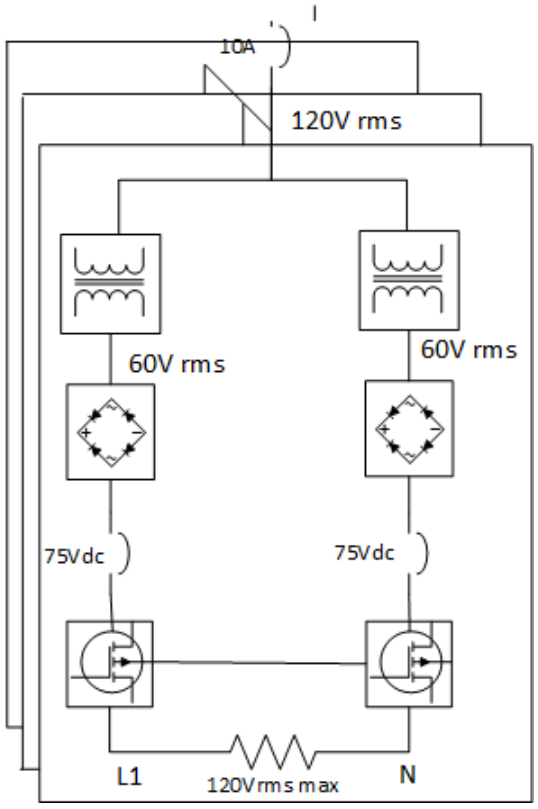


**Figure 19.** One-line diagram of the multilevel inverter.

Figure 20 shows the experimental platform constructed and used for the laboratory tests. The control signals are generated by the FPGA, which are used to switch the multilevel inverter and generate the power necessary to drive the PMSM according to the desired reference trajectory. The experimental platform is constituted by the following elements:

- Isolated DC sources
- H-bridges
- Main charge center
- Transformer of the DC source
- Diode bridge
- Capacitor of the DC source
- DIN rail
- Connection screw terminals
- Protection breakers

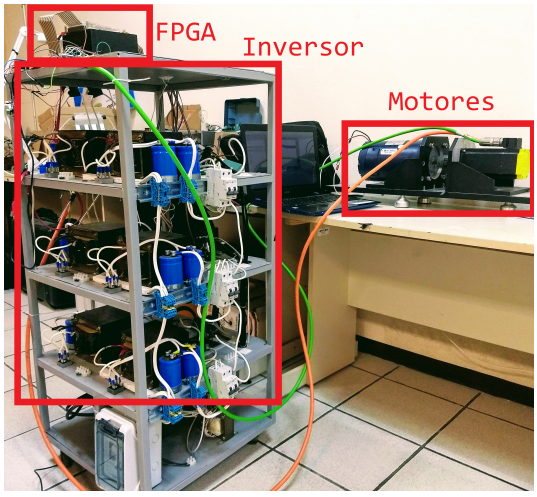


**Figure 20.** Experimental platform.

### 6.1. ALINX development kit

The development kit utilized is ALINX brand, AX309 model, shown in Figure 21. It comprises a FPGA SPARTAN-6 XC6SLX9, which is a low-cost development platform, with the following characteristics:


**Figure 21.** Development kit.

- FPGA model: XC6SLX9
- Size: 90x130 mm
- Crystal oscillator: 50 MHz
- Number of expansion I/O: 68
- SDRAM: 256 Mbit
- FLASH: 16 Mbit
- IO port level: 3.3 V
- Serial port: onboard U transfer serial port

### 6.2. Permanent magnet synchronous motor

The motor utilized is BALDOR Brand, whose main characteristics are 1.23 kW at 4000 rpm; Figure 22 shows the motor, while Table 2 shows its parameters.

**Table 2.** Motor parameters

| Parameters | Value |
|---|---|
| CAT.NO. | BSM80N-275AF |
| SPEC. | S2P141W042G1 |
| TORQ CONT STALL | 3.2 NM |
| CUR CONT STALL/A RMS | 4 |
| POWER | 1.23KW |
| RATED SPEED/RPM | 4000 |
| RATED BUS VOLTS | 300 |
| PEAK CUR/A RMS | 14.4 |
| MAX SPEED/RPM | 7000 |
| CLASS | F |
| AMB. | 25 |

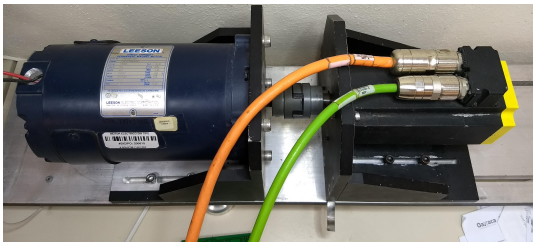

**Figure 22.** PMSM employed (left) and DC motor used as load (right).

## 7. Experimental results

Since the processing inside the FPGA device is numerical, it is necessary to have a mean to read the different signals that are processed. For this purpose, readings of the signals are taken and sent through a serial port/UART at a speed of 921600 bps; once received in the computer, they are processed and presented by means of a GUI designed in LabView. The measured parameters include the speed ($\omega$), the currents iq, id and the estimated load torque (LT); each plot has a total of 70000 samples in 5 seconds.

On the other hand, the shaft of the PMSM is coupled to a direct current (DC) motor that operates as an external mechanical load, as can be seen in Figure 22.

Then, Figure 23 shows the angular speed $\omega$ measured by the motor encoder. The desired speed $\omega_d$ is determined by tracking a curve obtained from a 6th-order Bezier polynomial, which is used as reference. The duration of the starting curve is 5 s, after which it can be observed that the motor speed settles at 200

rad/s, as desired. Nevertheless, note that there is a ripple in the real speed, which is mainly due to the numerical differentiation algorithm used to obtain the speed from the readings of the encoder.

During the same period of time, Figure 24 shows the tracking of the $i_q$ current at the start of the motor, which settles at 0.5 amperes; this is the active current, which is transformed by the motor into mechanical torque. Similarly, Figure 25 shows the regulation of the $i_d$ current, which should remain equal to 0 amperes because is it the reactive current that cannot be utilized. Both currents exhibit ripple, due to the resolution of the employed sensor currents. At last, Figure 26 shows the estimation of the load torque at the motor shaft during the same period of time, which settles at 0.4 Nm when the desired speed is reached.

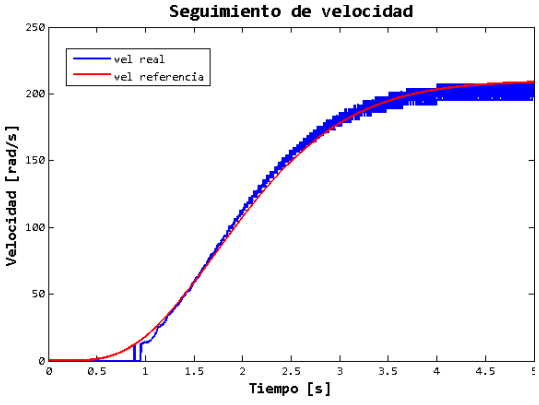


**Figure 23.** Soft tracking of velocity at start.

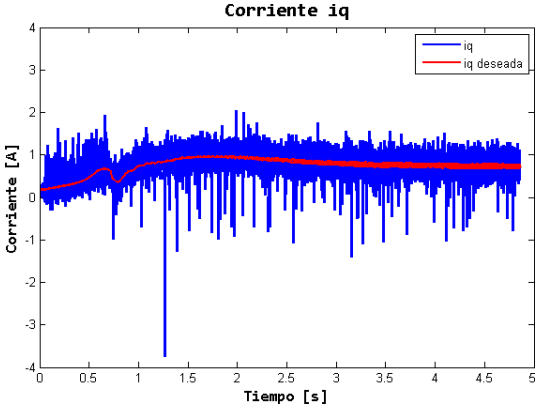


**Figure 24.** Tracking of the iq current at start.

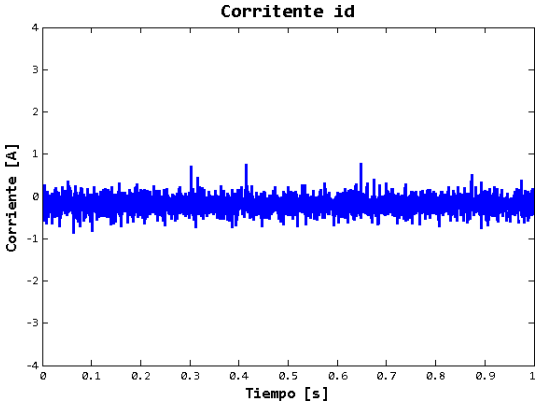


**Figure 25.** Tracking of the $i_q$ current at start.

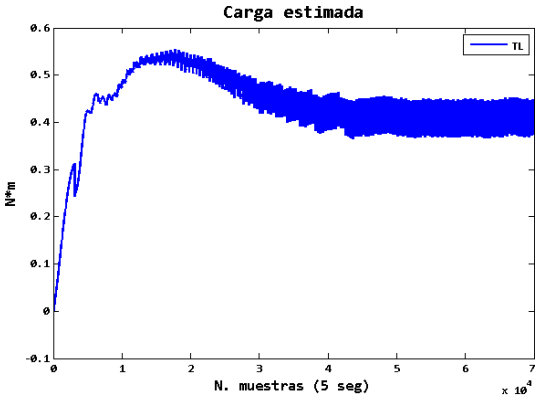


**Figure 26.** Estimated load at the motor shaft.

## 8. Conclusions

Based on the simulation and experimental results, it can be argued that the designed and implemented passive output feedback controller exhibits an adequate performance in the tracking and regulation of the angular speed of the PMSM. On the other hand, it is verified that the reduced-order load torque estimator, correctly estimates this parameter. In addition, in terms of hardware, it is observed that the multilevel inverter considerably reduces the harmonic distortion of the voltage, which in general results in less noise in the system and avoids the need of using filters. Due to the particularities of the digital device (FPGA) where the implementation was carried out, a sampling time of 10 µs is achieved, which results in the benefit of having larger speed of response in the presence of sudden changes in the load.

As a future work, it is intended to implement a more complex algorithm, such as an algebraic estimator, to accomplish a better estimation of the mechanical load torque. The above would be possible taking advantage of the parallelism, which is the main characteristic of the FPGA.

## References


[1] T. Wildi, *Máquinas eléctricas y sistemas de potencia*, 2007. [Online]. Available: https://bit.ly/35aPtZm

[2] L. Blanco Rubio, "Diseño electromagnético de un motor síncrono de imanes permanentes para el accionamiento directo de la hélice de un barco," 2017. [Online]. Available: https://bit.ly/2PAJZAj

[3] J. Linares-Flores, C. Garcia-Rodriguez, H. Sira-Ramirez, and O. D. Ramirez-Cärdenas, "Robust backstepping tracking controller for low speed pmsm positioning system: Design, analysis, and implementation," in *2015 IEEE International Conference on Industrial Technology (ICIT)*, March 2015, pp. 2131–2138. [Online]. Available: https://doi.org/10.1109/ICIT.2015.7125411

[4] IEEE, "Ieee recommended practices and requirements for harmonic control in electrical power systems," *IEEE Std 519-1992*, pp. 1–112, April 1993. [Online]. Available: https://doi.org/10.1109/IEEESTD.1993.114370

[5] M. H. Rashid, *Electrónica de potencia: circuitos, dispositivos y aplicaciones*, 2004. [Online]. Available: https://bit.ly/2t9mZ42

[6] J. A. Juárez-Abad, J. Linares-Flores, E. Guzmán-Ramírez, and H. Sira-Ramírez, "Generalized proportional integral tracking controller for a single-phase multilevel cascade inverter: An fpga implementation," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 256–266, Feb 2014. [Online]. Available: https://doi.org/10.1109/TII.2013.2242085

[7] F. Chauca Llusca, F. Llerena Rengel, and P. Chico Hidalgo, "Diseño y construcción de un inversor multinivel," *Revista Politécnica*, vol. 33, no. 1, 2014. [Online]. Available: https://bit.ly/36rAxpI

[8] L. G. Franquelo, J. Rodriguez, J. I. Leon, S. Kouro, R. Portillo, and M. A. M. Prats, "The age of multilevel converters arrives," *IEEE Industrial Electronics Magazine*, vol. 2, no. 2, pp. 28–39, June 2008. [Online]. Available: https://doi.org/10.1109/MIE.2008.923519

[9] E. Monmasson, L. Idkhajine, and M. W. Naouar, "Fpga-based controllers," *IEEE Industrial Electronics Magazine*, vol. 5, no. 1, pp. 14–26, March 2011. [Online]. Available: https://doi.org/10.1109/MIE.2011.940250

[10] D. G. Maxinez and J. Alcalá Jara, *VHDL: el arte de programar sistemas digitales*, 2002. [Online]. Available: https://bit.ly/2PCE8dL

[11] J. J. Rodríguez-Andina, M. D. Valdés-Peña, and M. J. Moure, "Advanced features and industrial applications of FPGAs–a review," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 4, pp. 853–864, Aug 2015. [Online]. Available: https://doi.org/10.1109/TII.2015.2431223

[12] W. Zhu, "Fpga logic devices for precision control: An application to large friction actuators with payloads," *IEEE Control Systems Magazine*, vol. 34, no. 3, pp. 54–75, June 2014. [Online]. Available: https://doi.org/10.1109/MCS.2014.2308691

[13] E. Mandado, L. J. Álvarez, and M. D. Valdés, *Dispositivos Lógicos Programables*, 2002. [Online]. Available: https://bit.ly/38uCiEq

[14] R. Krishnan, *Permanent Magnet Synchronous and Brushless DC Motor Drives*, 2017.

[15] J. Linares-Flores, C. García-Rodríguez, O. D. Ramírez-Cárdenas, C. Escobar-Noriega, and M. A. Contreras-Ordaz, "Control robusto de seguimiento suave de posición angular del motor síncrono de imanes permanentes," in *Memorias del XVI Congreso Latinoamericano de Control Automático, Octubre 14-17, 2014. Cancún, Quintana Roo, México*, 2014, pp. 1113–1118. [Online]. Available: http://doi.org/10.13140/2.1.2760.9607

[16] H. J. Marquez, *Nonlinear control systems: analysis and design*. Wiley-Interscience Hoboken, 2003, vol. 1. [Online]. Available: https://bit.ly/2YMdMKM

[17] Y. E. Gliklikh, "Necessary and sufficient conditions for global-in-time existence of solutions of ordinary, stochastic, and parabolic differential equations," *Abstract and Applied Analysis*, vol. 2006, Special Issue, p. 17, 2006. [Online]. Available: https://doi.org/10.1155/AAA/2006/39786

[18] IEEE, *IEEE 754-2019 - IEEE Standard for Floating-Point Arithmetic*, 2019. [Online]. Available: https://bit.ly/2E5tnvo

[19] M. Naouar, E. Monmasson, A. A. Naassani, I. Slama-Belkhodja, and N. Patin, "FPGA-based current controllers for AC machine drives–a review," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 4, pp. 1907–1925, Aug 2007. [Online]. Available: https://doi.org/10.1109/TIE.2007.898302