PUBLICATION OF Little Lion Scientific R&D, Islamabad PAKISTAN

Journal of Theoretical and Applied Information Technology

30th June 2011. Vol. 28 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

THE OPTIMAL DIRECT TORQUE CONTROL OF A PMSM DRIVE: FPGA-BASED IMPLEMENTATION WITH MATLAB & SIMULINK SIMULATION.

[12]BADRE BOSSOUFI, [1]MOHAMMED KARIM, [2]SILVIU IONITA, [1]AHMED LAGRIOUI

[1)] Laboratory of Data processing, Imagery and Analysis Numerical, University of Sidi Mohammed Ben Abdellah, Faculty of Sciences Dhar El Mahraz Fez, Morocco

[2)] Center of Modeling and simulation of the systems, Faculty of Electronics, Communications University of PITEŞTI, Romania

ABSTRACT

In this paper, we present a new contribution of FPGAs (Field-Programmable Gate Array) for control of electrical machines. A detailed description of the structure of direct torque control for PMSM drive, 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.

Keywords: *FPGA, Direct Torque Control (DTC), Permanent Magnet Synchronous Machine (PMSM), reusability, DSP, ADC.*

1. INTRODUCTION

The speed performance of new components and the flexibility inherent of all programmable solutions give today many opportunities in the field of digital implementation for control systems. This is true for software solutions as microprocessor or *DSP* (Digital Signal Processor). However, specific programmable hardware technology such as Field Programmable Gate Array (*FPGA*) can also be considered as an especially appropriate solution in order to boost performances of controllers [1-3]. Indeed, these generic components combine low cost development, thanks to their re-configurability, use of convenient software tools and more and more significant integration density [4, 5].

The *FPGA* technology is now used by an increasing number of designers in various fields of application such as signal processing [6], telecommunication [7], video [8], embedded control systems [9], and electrical control systems [10]. This last domain, i.e. the studies of control of electrical machines, will be presented in this paper. Indeed, these components have already been used with success in many different applications such as Pulse Width Modulation (*PWM*) [11], control of induction machine drives [12-14] and multimachine system control. This is because the *FPGA-based* implementation of controllers can efficiently answer current and future challenges of this field.

The principal advantages of the digital solutions are as follows:

- High flexibility of changing structures of control;
- Immunity against disturbances;
- No problems of variations of control parameters.

With technological advancement, increased integration of *FPGA* devices is increasing. Nowadays, the density of *FPGA* components can achieve the equivalent of 10 million logic gates with switching frequencies of around 500 MHz. This allows the implementation of complex algorithms control in their entirety with a small period of time to load.

The inherent parallelism of *FPGA* components offers the possibility to run several algorithms in parallel control and configure them according to the defined criteria. Dynamic configuration between the algorithms control has as objective to select the appropriate algorithms depending on your point of operation. It may be useful also to ensure continuous operation in case of faults (sensors, switches ...).

In this paper, a new contribution for the *FPGA*-Based implementation of controls electrical. This approach is based on concept modularity and reusability.

This paper presents the realization of a platform for *DTC* control of *PMSM* using *FPGA* based controller. This realization is especially aimed for future high

PUBLICATION OF Little Lion Scientific R&D, Islamabad PAKISTAN
Journal of Theoretical and Applied Information Technology
30th June 2011. Vol. 28 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

performance applications. In this approach, not only the architecture corresponding to the control algorithm is studied, but also architecture and the *ADC* interface and *RS232 UART* architecture.

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:

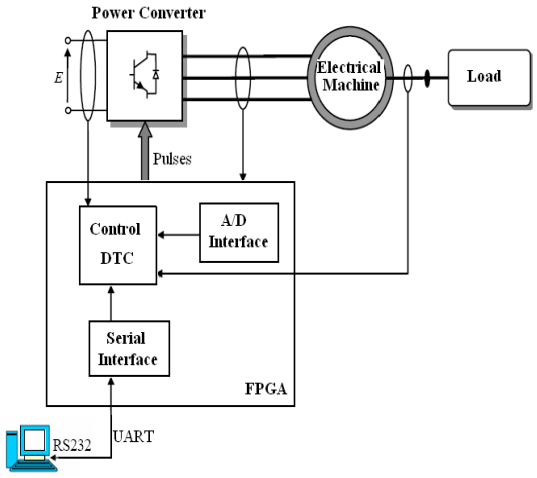

Fig.1. Architecture DTC Control

The concept of implementation is defined as being the introduction of a functionality given on a physical support. Within the framework of control of electric machines, the functionality to be introduced constitutes the algorithm of control, whose objective is to control the development state of mechanical or electric variables of the electric machine (flux, power, torque, speed...). As for the physical support, it constitutes the target of implementation. The latter can be of analogical or numerical nature.

According to the application, consider several algorithms of control can be used such as the current control algorithms, of active or reactive power, speed, position...etc. the structure of the these control algorithms generally comprises an internal loop of regulation of the current. Finally it is often the most difficult to implement because it generally constitutes the most complex part of the control algorithm.

The other control loops are relatively much simpler to implement. This is why, within the framework of this work, one will particularly be interested in the current control technique implementation of the electric machines "Direct Torque Control (DTC)".

2. DIRECT TORQUE CONTROL STRUCTURE

In this paper, we apply the command on a machine type *PMSM* (Permanent Magnet Synchronous Motor), which consists of three stator windings and a rotor magnet. This motor is described by the following equation (Voltage, Flux, Torque…) [2],

$$u_{sd} = r_s.i_{sd} + \frac{d\Phi_{sd}}{dt} - \omega.\Phi_{sq} \quad (1)$$

$$u_{sq} = r_s.i_{sq} + \frac{d\Phi_{sq}}{dt} - \omega.\Phi_{sd} \quad (2)$$

$$\Phi_{sd} = L_d.i_{sd} + \Phi_f \quad (3)$$

$$\Phi_{sq} = L_q.i_{sq} \quad (4)$$

$$Ce = \frac{3}{2} p\left[\phi_f.I_q + (L_d - L_q).I_d.I_q\right] \quad (5)$$

$$Ce - C_r = J.\frac{d\Omega}{dt} + f.\Omega \quad (6)$$

Where Ω is the rotation's speed, p the Number of pairs of poles, J the moment of inertia, f the Coefficient of viscous friction, Cr the resistive torque, Φf the flux produced by the permanent magnet, Lsd and Lsq the d-q axis stator inductance, Vsd and Vsq the d-q axis stator voltage, rs the stator winding resistance and Ce the electromagnetic torque,

In the middle of 1980's Depenbrock and Isao Takahashi proposed Direct Torque Control for electrical machines [9, 11], more than one decade later. Idea the DTC basic for electrical motor is slip control [2]. In the 1990's, DTC for PMSM was developed [7, 12, 13]. The DTC control a magnet synchronous motor Standing is based on the direct determination of sequence of commands applied to switches a voltage inverter. This strategy is based generally on the use of comparators hysteresis whose role is to control the amplitudes stator flux and electromagnetic torque, as in [8, 14]. The Fig.2 is a typical DTC system. It includes estimators flux and torque electromagnetic, flux and torque hysteresis controllers and a switching table.

PUBLICATION OF Little Lion Scientific R&D, Islamabad PAKISTAN
Journal of Theoretical and Applied Information Technology
30th June 2011. Vol. 28 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

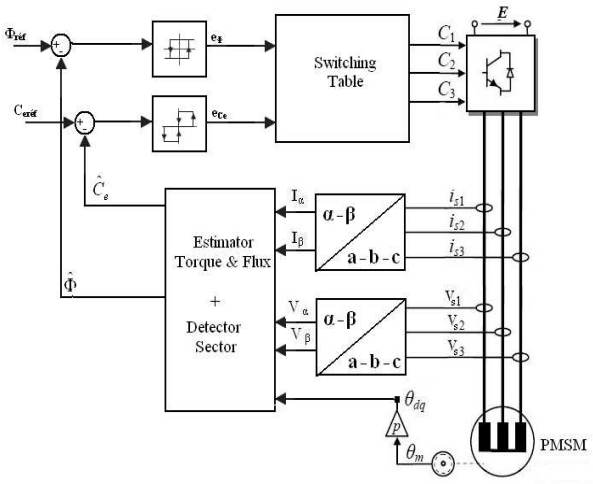

Fig.2. DTC Control Principle

The Fig.2 presented the chosen control. The characteristics memorized for an optimal torque control strategy generates the d-q component of the vector stator current. Then, an abc-to-αβ transformation generates the flux and torque electromagnetic. Finally, these generated parameters are compared to the references and hysteresis controllers to allow the determination of the switching table status C1, C2 and C3 of the voltage inverter

3. THE DIFFERENT STEPS OF FPGA-BASED IMPLEMENATION

3.1. FPGA devises

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.

The Fig.3 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.

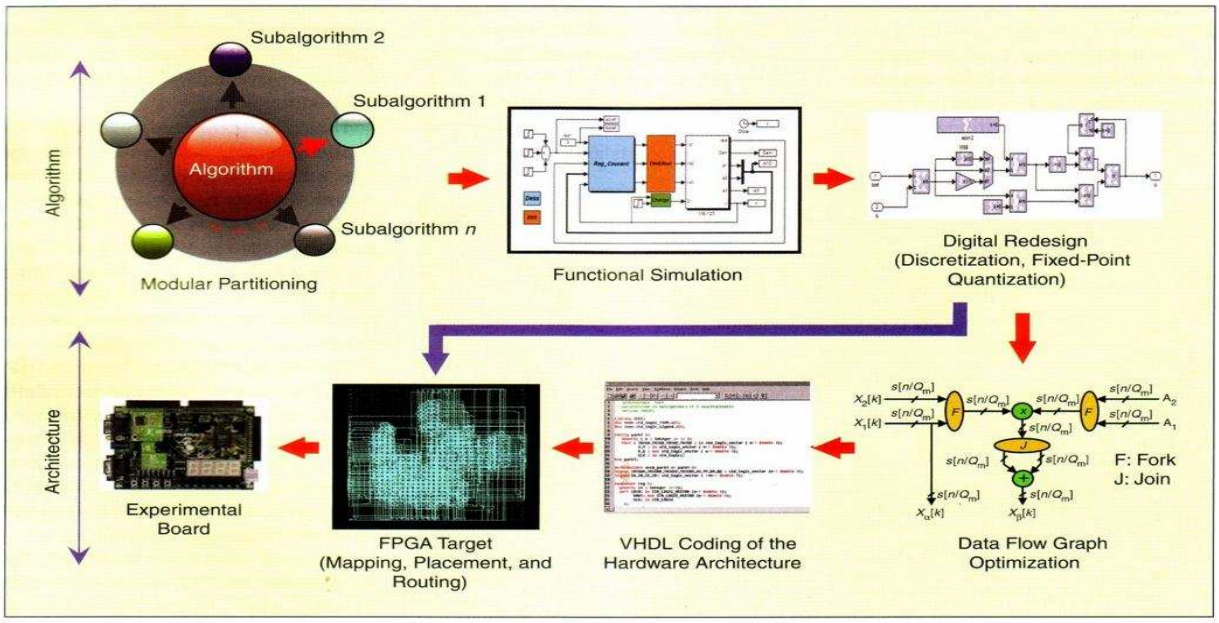

Fig.3. Programming FPGA devisees

PUBLICATION OF Little Lion Scientific R&D, Islamabad PAKISTAN
Journal of Theoretical and Applied Information Technology
30th June 2011. Vol. 28 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

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 (Fig.4).

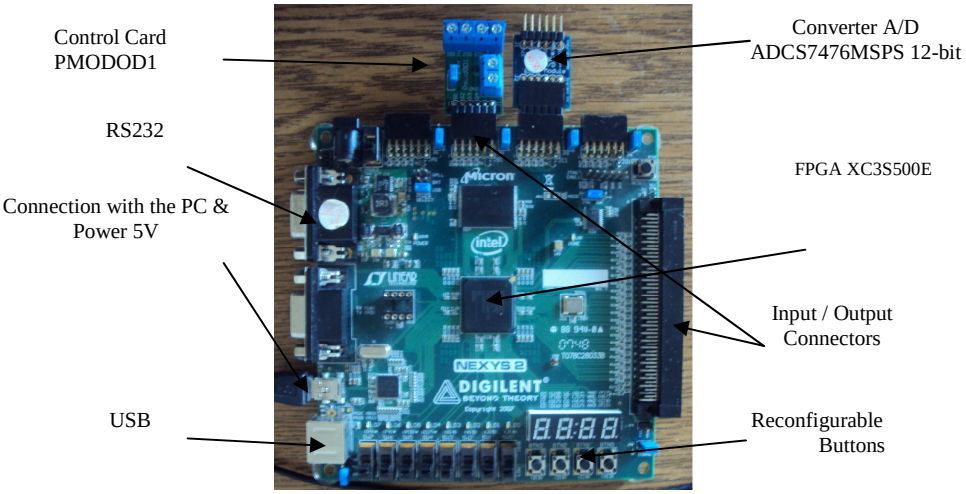

Fig.4. The FPGA XC3S500E Spartan

3.2. Simulation Procedure

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 time's discredited model of the DTC algorithm associated with a voltage inverter and PMSM model. The Fig.5 gives a global view of the functional model.

The Fig.5 shows in detail the programming of the control shown in Fig.2 (DTC Control) in the SYSTEM GENERATOR environment from Xilinx; we will implement it later in the memory of the FPGA for the simulation of PMSM.

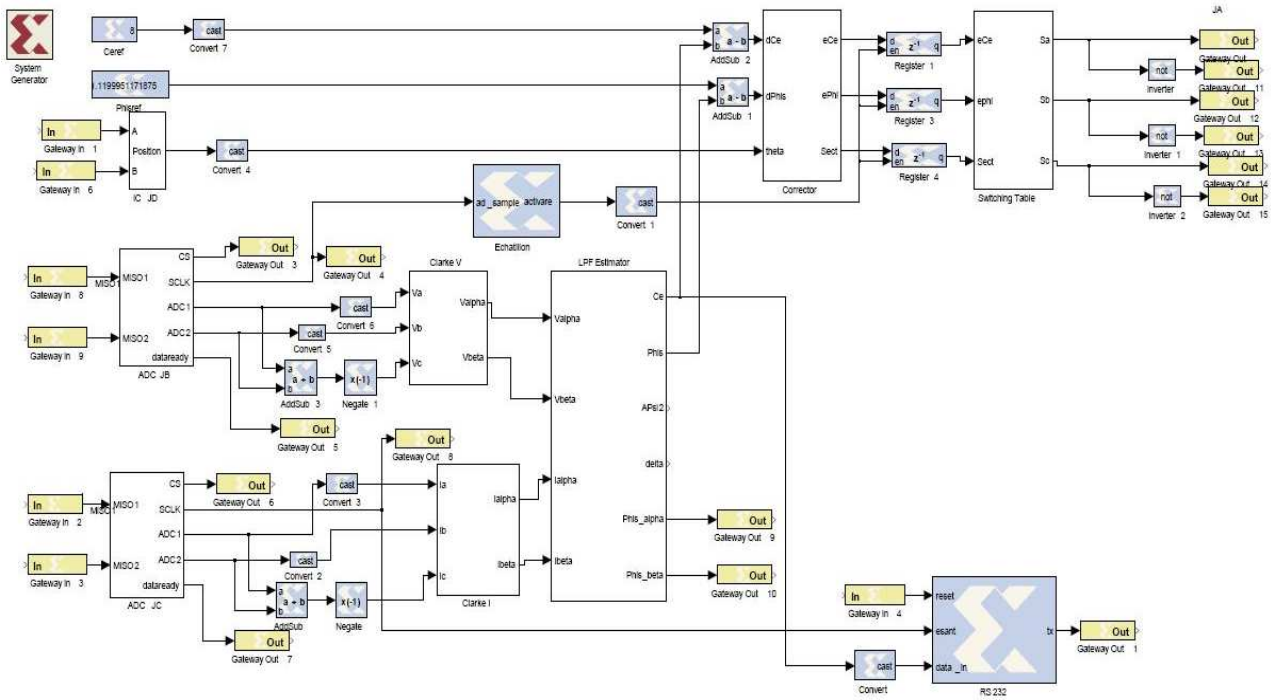

Fig.5. Functional Model DTC from SYSTEM GENERATOR

PUBLICATION OF Little Lion Scientific R&D, Islamabad PAKISTAN
Journal of Theoretical and Applied Information Technology
30th June 2011. Vol. 28 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The description of the different modules is detailed below:

- Two blocks of coordinates transformation: the Clark transformation (abc-to-αβ);
- The block switching table is the most important, because it can provide control pulses to the IGBT voltage inverter in the power section from well-regulated voltages.
- The block estimator torque and flux, corrector hysteresis and detector sector.
- The block encoder interface IC allows the adaptation between the FPGA and the acquisition board to iniquity the rotor position of the PMSM;
- The ADC interface allows the connection between the FPGA and the analog-digital converter (ADCS7476MSPS 12-bit A / D) that will be bound by the following two Hall Effect transducers for the acquisition of the stator currents machine;
- Block "Timing" which controls the beginning and the end of each block, which allows the refresh in the voltages reference V10, V20 and V30 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.6 gives the specification model of the abc-to-αβ (Clark) 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 (7):

$$\begin{bmatrix} V_\alpha \\ V_\beta \end{bmatrix} = \frac{2}{3}\begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} \quad (7)$$

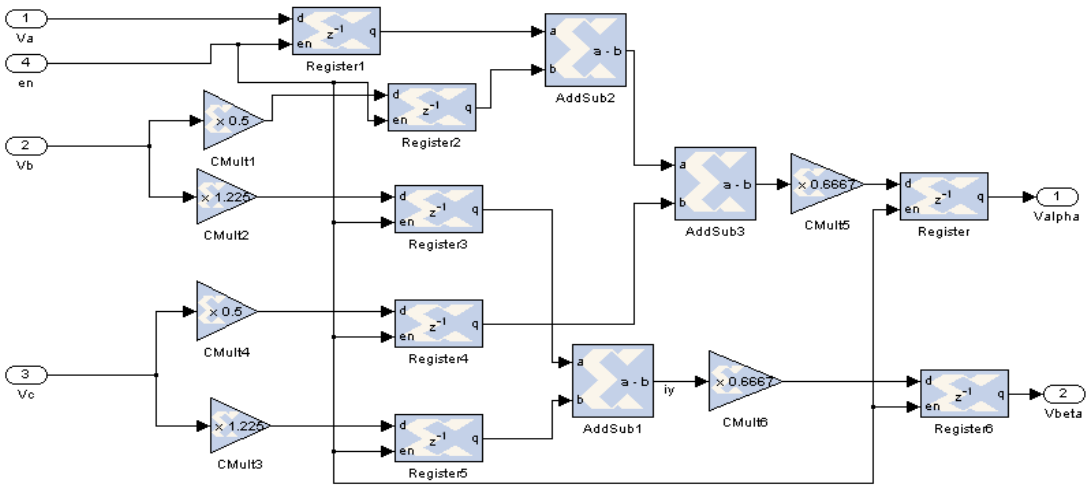

Fig.6. Clark Transformation Model

The specification model is then used for the definition of the corresponding Data Flow Graph DFG. Fig.7 showed the DFG corresponding to the Clark transformation module.

3.3. Optimization of the consumed resources

The Algorithm Adequation Architecture (A^3) ensures this application, in order to generate a new graph called Factorized FDFG (Data Flow Graph). This method is used to generate optimized hardware architecture for each module. If an operator is used several recovery, its factorization via the A3 methodology consists in using it only one time. Fig.8 showed the FDFG of the Clark transformation module.

PUBLICATION OF Little Lion Scientific R&D, Islamabad PAKISTAN
Journal of Theoretical and Applied Information Technology
30th June 2011. Vol. 28 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

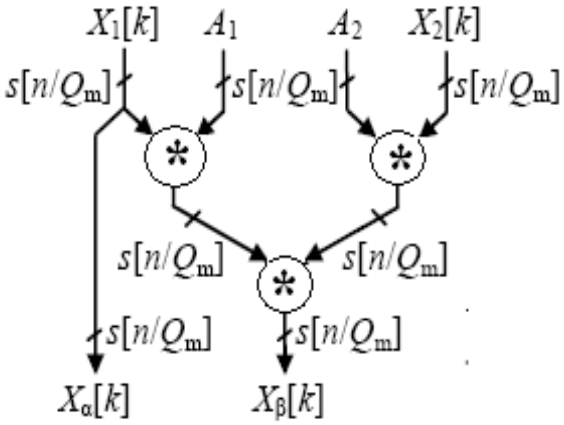

Fig.7. DFG of the Clark Transformation

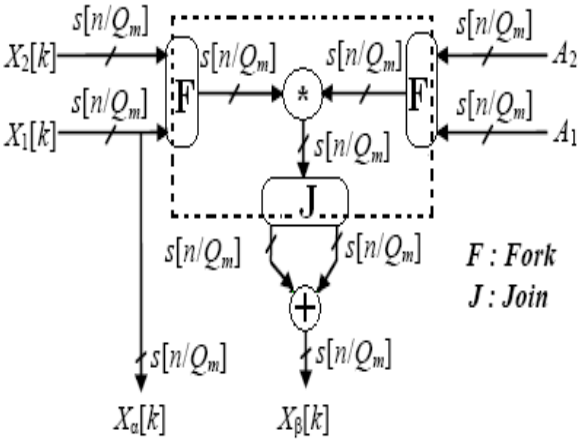

Fig.8. FDFG of the Clark Transformation

3.4. Modular design of the control architecture

To design the control architecture to implement a data path and control unit is defined for each module as shown in Fig.9. The data path is mainly made up of operators and registers. The registers can provide the technical Pipeline which consists on decomposing the performance of an algorithm in different sequences according to the data dependency. The communication between the registers and operators can be achieved through data buses, multiplexers and demultiplexers. Concerning the control unit, its role is to ensure the flow of data according to very specific sequences within the data path. The control unit is other than a finite state machine (*FSM*: Finate State Machine). Fig.9 and Fig.10 present two different architectures *DFG* and *FDFG* processing module Clark.

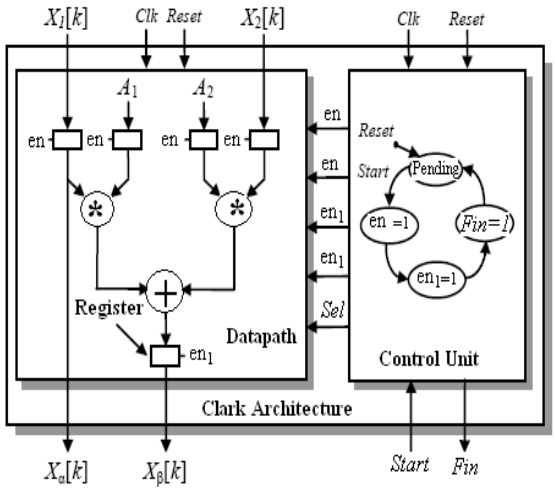

Fig.9. DFG Clark transformation module architecture

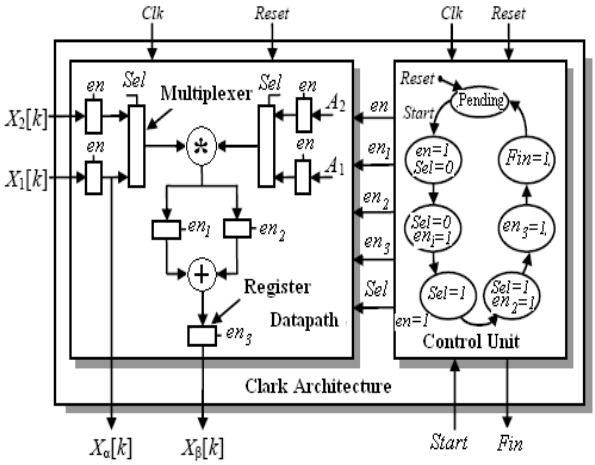

Fig.10. FDFG Clark transformation module architecture

4. MEASUREMENT RESULTS

For this work, the used *FPGA* target is a *XC3S500E* Spartan3E the firm Xilinx, the *FPGA* based hardware control system includes the *DTC*, an *ADC* interface and *IC* interface in one *FPGA* chip. Fig.11 presents the corresponding implemented architecture.

PUBLICATION OF Little Lion Scientific R&D, Islamabad PAKISTAN
Journal of Theoretical and Applied Information Technology
30th June 2011. Vol. 28 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

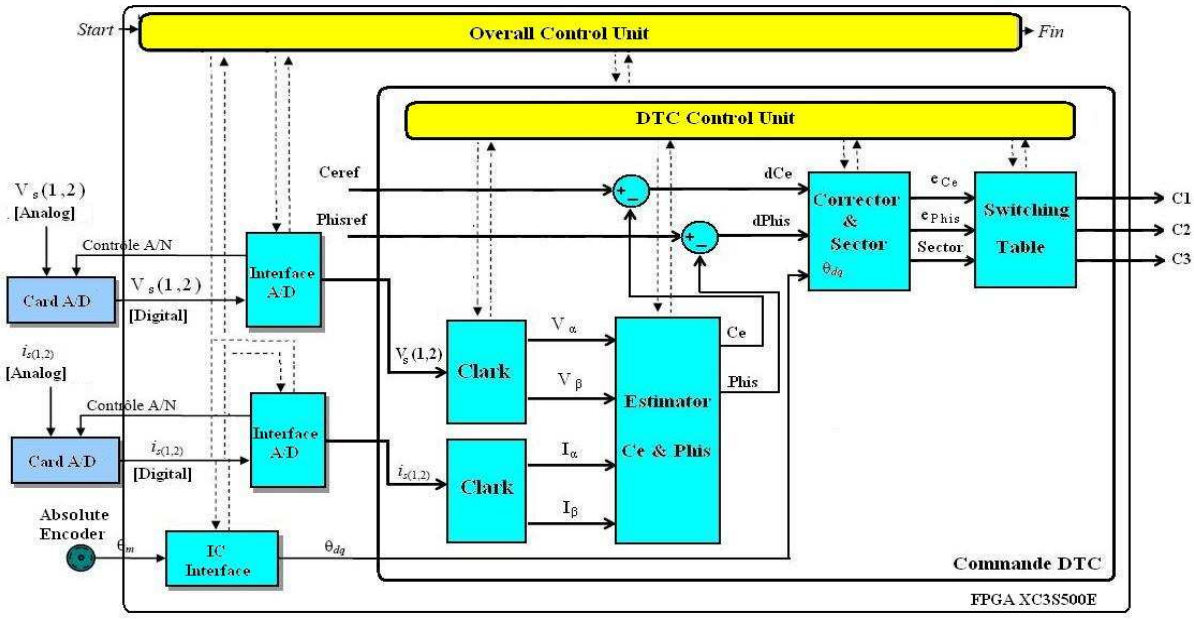

Fig.11. FPGA based hardware DTC

The control unit for architecture ensures implanted control module of the *ADC* interface, the encoder interface and the command encoder *DTC*. The *Fig.12* shows the timing diagram for the mode of operation of these four modules. At the beginning of each sampling period, the *A/D* interface module and encoder interface are activated simultaneously. Then, after a delay conversion from analog to digital $t_{A/D}$, the control module *DTC* is activated. This is driven by its own control unit and allows you to refresh the reference torque and flux after a computation time equal to t_{cr}. Once activated, the control unit control module *DTC* activates the first two modules of the transformation of Clark (*123-αβ*) that will calculate the components $i_{s\alpha}$ and $i_{s\beta}$ stator current vector and the components $V_{s\alpha}$ and $V_{s\beta}$ stator voltage vector. The computation time of this transformation is equal to t_C. When the modules of this transformation indicate the end of the calculation, the estimator module flow and electromagnetic torque is activated. It has the same computation time equal to test and calculates the flux and torque estimated.

Subsequently, when the estimator module indicates the end of the calculation, the correction module of hysteresis and the detector area is activated to calculate the error of the torque and stator flux and the area *N*. This module is characterized by a computation time equal to t_{cr}. Finally, when the errors of flux and torque and sector *N* are calculated, the module of the switching table is activated. The latter has a running time equal to t_{table} and generates the control signals C_1, C_2 and C_3 for controlling the switches of the voltage inverter.

The following table shows the performance of computing time and resource consumption, obtained during the implementation of the control *DTC* architecture. The resources consumed are obtained for a fixed point format *20/Q18*. The total computing time of t_{DTC}, in the *DTC* squeal module is equal to 0.99µs. By adding the analog to digital conversion time $t_{A/D}$, total time T_{ex} architecture equals 3.43µs.

TABLE I
FPGA Performances

Module	Latency	Time Calculation
Interface A/D	120	$t_{A/D}$=2.4 µs
Interface A/D	120	$t_{A/D}$=2.4 µs
IC Interface	2	t_{Cod}=0.04 µs
123-αβ	18	t_C = 0.34 µs
Estimator Flux & Torque	12	t_{Est} = 0.25
Corrector & Sector	14	t_{Cr} = 0.22
Switching Table	8	t_{table} = 0.18µs
$t_{DTC} = t_C + t_{Est} + t_{Cr} + t_{table}$		t_{DTC} = 0.99 µs
Run time $T_{ex}=T_{A/D} + t_{DTC}$		t_{ex} =3.43 µs
Resources Consumed	Number of Slices	1850 de 5376 (34%)
	Wired Multipliers	12 de 16 (65%)
	Memory RAM	6%

PUBLICATION OF Little Lion Scientific R&D, Islamabad PAKISTAN
Journal of Theoretical and Applied Information Technology
30th June 2011. Vol. 28 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

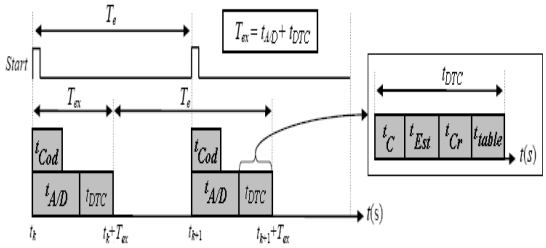

Fig.12. Time diagram of the control structure DTC

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

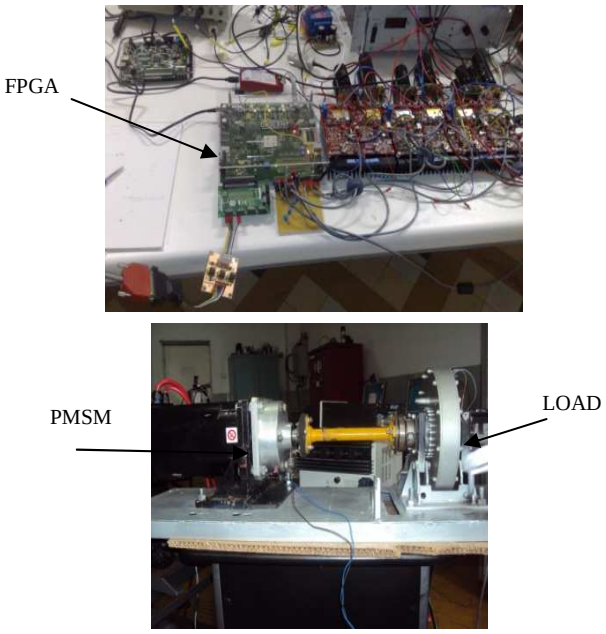

Fig.13. Prototyping platform control

The Fig.14 shows the experimental results obtained during the implementation of the DTC control, it presents the state of control signals for the switches of the inverter voltage in the area in which there is the reference voltage vector. These results are similar to those presented in the theories. Furthermore, the control signals generated from the FPGA board will be filtered before being injected into the voltage inverter.

The above figures show that the phases are balanced and demonstrate the proper functioning of the Switching table.

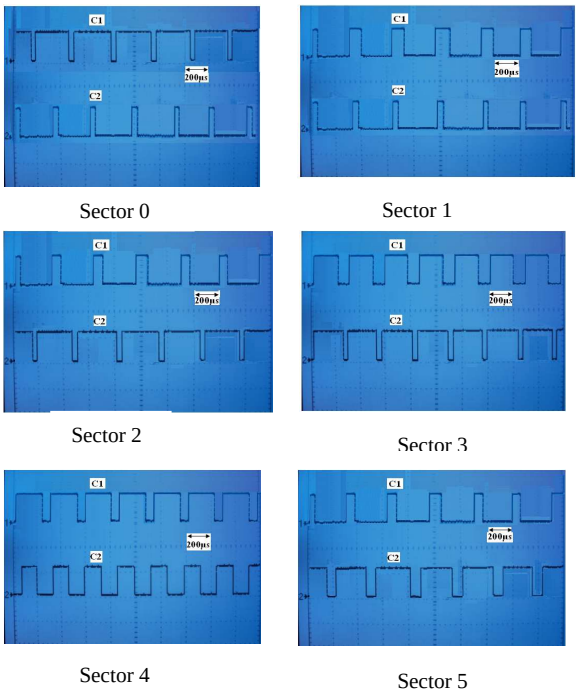

Fig.14. Switching states of control signals C_1 and C_2 for the DTC Control

The *Fig.14* and *Fig.12* show that control system satisfy the basic requirements of the control strategy and validate therefore the good functionality of the system. In fact, It can be noted that:

- The switching frequency is limited to the sampling frequency of the control algorithm to guarantee safe operation of the semiconductor power devices.
- The switching frequency increases weakly when the stator current vector magnitude decreases.

Implementing the *DTC* control in *FPGA* has the drawbacks:

- The switching frequency is variable. It is limited to half the sampling frequency of the control algorithm and maximum at very low speed.
- The zero voltage vectors are not applied.

At the hardware level, the execution time of the control architecture is of the order of several microseconds, which allows a better control of current, including a *THD* lower.

PUBLICATION OF Little Lion Scientific R&D, Islamabad PAKISTAN
Journal of Theoretical and Applied Information Technology
30th June 2011. Vol. 28 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

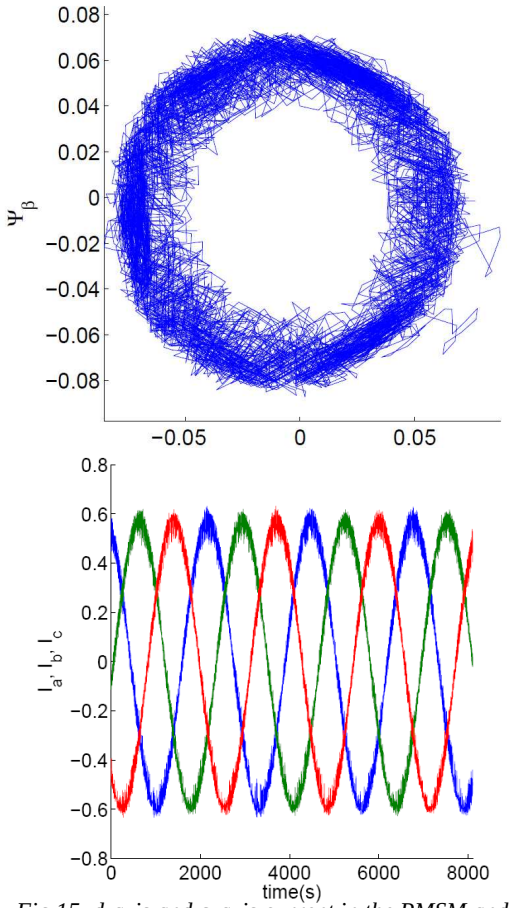

Fig.15. d-axis and q-axis current in the PMSM and stator flux locus for DTC PMSM

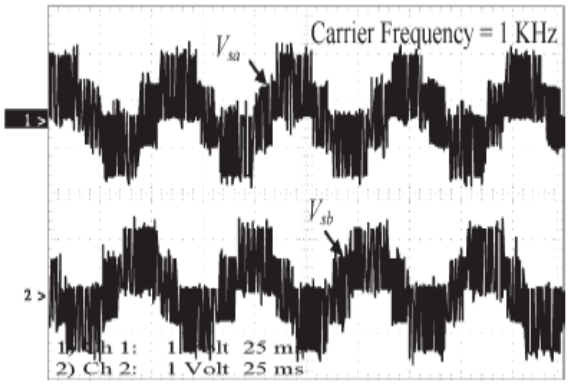

Fig.16. α-axis and β-axis Voltage for DTC PMSM

In *Fig.15and 16 the* experimental results *DTC* of *PMSM* with the *FPGA* platform are shown. Clearly the constant set values for stator flux linkage magnitude and torque result, because of the hysteresis controllers in the *DTC* scheme, in d-axis and q-axis current components that are bounded within hysteresis limits. The hysteresis control is visible in the stator flux locus plot as well. Update frequency for this implementation is 20 kHz. All results were extracted from the *FPGA* by the ChipScope tool of Xilinx.

CONCLUSIONS

In this work, we presented the *FPGA* implementation of Direct Torque Control architecture for permanent magnet synchronous machine (*PMSM*). The results obtained show the benefits of an appropriate methodology that allows the creation of a library of reusable modules optimized. This work provides a new benefit to our laboratory work and a new route for the development and implementation of other control architectures.
In order to demonstrate the benefits of using FPGAs, a set of very demanding industrial control systems in terms of real-time performances has been studied in details. This set consists of several current control techniques applied to ac machine drives.

ACKNOWLEDGMENT

We thank all those who contributed to make this work, including my teachers in the Center of Modeling and simulation of the systems, University of Pitesti from Romania and in my home laboratory *LIIAN* from Morocco, and all my friends for their support.

REFERENCES:

[1] M.W. NAOUAR, « Commande numérique à base de composants FPGA d'une machine synchrone », algorithmes de contrôle du courant, *Thèse de Doctorat, Ecole Nationale d'Ingénieurs de Tunis et l'Université de Cergy Pontoise*, Tunis, 2007.

[2] B.BOSSOUFI, M.KARIM, S.IONITA, A.LAGRIOUI, "Performance Analysis of Direct Torque Control (DTC) for Synchronous Machine Permanent Magnet (PMSM) " *in Proc. IEEE-SIITME'2010*, pp. 275-280, 23-26 Sep 2010, Pitesti, Romania.

[3] NAOUAR M.-W. ; MONMASSON E.; SLANW-BELKHODIA I. "FPGA-based torque controller of a synchronous machine" *in Proc. IEEE-ICIT'04*, pp. 8-10, Hammamet, Tunisia, Dec. 2004.

[4] E. MONMASSON, and M. Cirstea "FPGA Design Methodology for Industrial Control Systems – A Review," *IEEE Trans Ind. Electron..*, vol.54, no. 4, pp.1824-1842, August. 2007.

[5] Y.A. Chapuis, J. P. Blonde, and F. Braun, "FPGA Implementation by Modular Design Reuse mode to Optimize Hardware Architecture and performance of AC Motor Controller Algorithm," *in Proc. EPE-PEMC Conf.*, Sept. 2004.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[6] L. Charaabi, "Conception des architectures matérielles dédiées à la commande de systèmes électriques", *Thèse de Doctorat, Ecole Nationale d'Ingénieurs de Tunis, 2006.*

[7] F. Zhengwei, J. E. Carletta, and R. J. Veillette, "A methodology for FPGA-based control implementation," IEEE Trans. Control Syst. Technol., vol. 13, no. 6, pp. 977–987, Nov. 2005.

[8] X. Lin-Shi, F. Morel, A. M. Llor, B. Allard and J.M. Retif "Implementation of Hybrid Control for Motor Drives," *IEEE Trand. Ind Electron.*, vol.54, no.4, pp.19446-1952, August. 2007.

[9] H. T. moon, H. S. Kim and M. J. Youn, "A Discret Time Predictive Current Control for PMSM" *IEEE Trans. Power Electronic.*, vol.18, no.1, pp. 464-472, Janvier. 2003.

[10] H.J. Lee; S.K. Kim; Y.A. Kwon; S.J. Kim, "ASIC design for DTC based speed control of Induction motor," *in Proc. IEEE ISIE'01 Conf.*, 2001, pp. 956 –961

[11] F. Ricci, H. Le-Huy, "An FPGA-based rapid prototyping platform for variable-speed drives," *in Proc. IEEE IECON'02 Conf.*, 2002, pp. 1156- 1161.

[12] M.W.Naouar, E.Monmasson, I.S.Belkhodja "FPGA-Based Current Controllers for AC Machine Drives-A Review," IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 54, NO. 4, AUGUST 2007.

[13] E. Monmasson; B. Robyns; E. Mendes, B. De Fornel, "Dynamic reconfiguration of control and estimation algorithms for induction motor drives," *in Proc. IEEE ISIE Conf.*, 2002, pp. 828 –833.

[14] Takahashi and T. Noguchi, "A new quick-response and high-efficiency control strategy of an induction machine," *in IEEE Trans. Ind. Applicat.*, vol. 22, pp. 820-827, Sept./Oct. 1986.