Abdelmonoem Nayli[(1,3)], Sami Guizani[(2,3)], Faouzi Ben Ammar[(3)]

J. Electrical Systems 11-4 (2015): 420-432

Regular paper

Journal of Electrical Systems

Implantation of the scalar control for the open-end winding induction machine On FPGA Spartan 3E

In this paper, the authors propose the implantation of the scalar control for controlling the open-end winding induction machine on FPGA. The scalar control is simulated in matlab Simulink environment using a Xilinx System Generator. This control is implemented in a Xilinx Spartan 3E FPGA board. It is experimentally verified by the visualization of command signals for the switches of the inverters.

Keywords: Open-end winding induction machine; Scalar control; Field Programmable Gate Arrays; Xilinx System Generator.

Article history: Received 9 January 2014, Received in revised form 18 September 2015, Accepted 14 October 2015

1. Introduction

The power segmentation is now a major interest in electrical engineering researches. the considerable interests are given for multiphase machines [1,2], multi star machines [3,4], and open-end stator windings induction machines "OEWIM" [5-8], and the dual open-end stator windings induction machines [9,10].

The implementation of digital control of electric machines on the Field Programmable Gate Arrays FPGA occupies a large part of industrial drive systems. Indeed, the "FPGA" are reconfigurable components. They present the advantage that they can be reconfigured as many times as needed which allows their quickly reuse with different algorithms. In addition, they present time very impressive calculation capabilities without forgetting the flexibility that they offer for the designer since it is alone to design algorithms [11] and to implement them [12-17].

The authors treat in the paper three parts:

In the first part, the different modules made which allowed to have the scalar control are presented. The each module is validated in the Matlab-Simulink environment using Xilinx System Generator.

The second part is presented the digitization validation of the scalar control for an open-end winding induction machine using Matlab-simulink with Xinlinx System Generator.

Finally, the composed algorithm of the different modules of the scalar control is implemented on Spartan 3E FPGA board. The implementation is verified by the visualization of command signals for the switches of the inverters.

2. Digitization of the scalar control

Before commencing the digitization of the scalar control "V / f law ", the figure 1 shows the supply of the open-end winding induction machine by two PWM voltage source inverters based on V/f law.

(1) University of Tunis, ESSTT, Tunisia. Email: n.ayli@hotmail.fr
(2) University of El Manar, IPEIEM, Tunisia. Email: Guizani_sami@yahoo.fr,
(3) University of Carthage, MMA Laboratory, INSAT, Tunisia. Email : Faouzi.Benamar@insat.rnu.tn

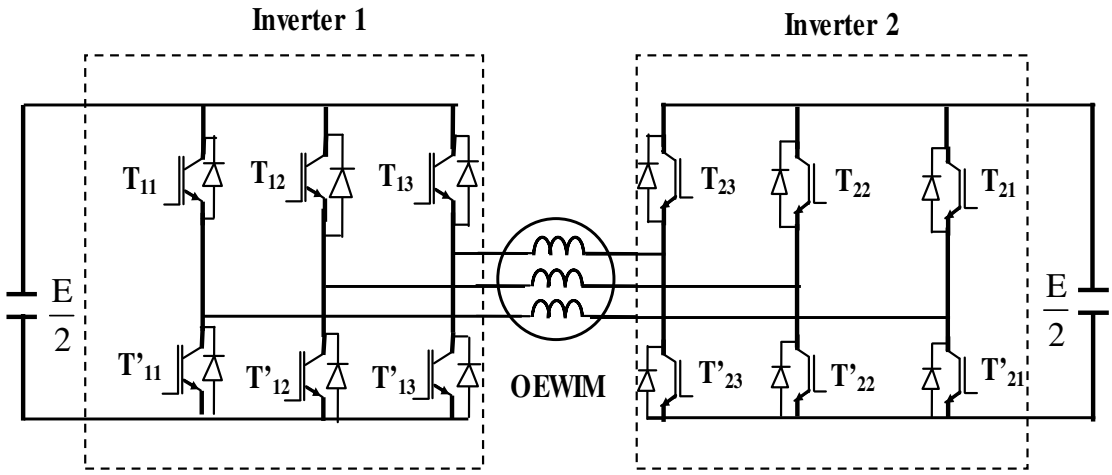


Fig. 1. Open-end winding induction machine supplied by two voltage source inverters.

The figure 2 shows the principle of PWM to supply the open-end winding induction machine by two 2- level inverters.

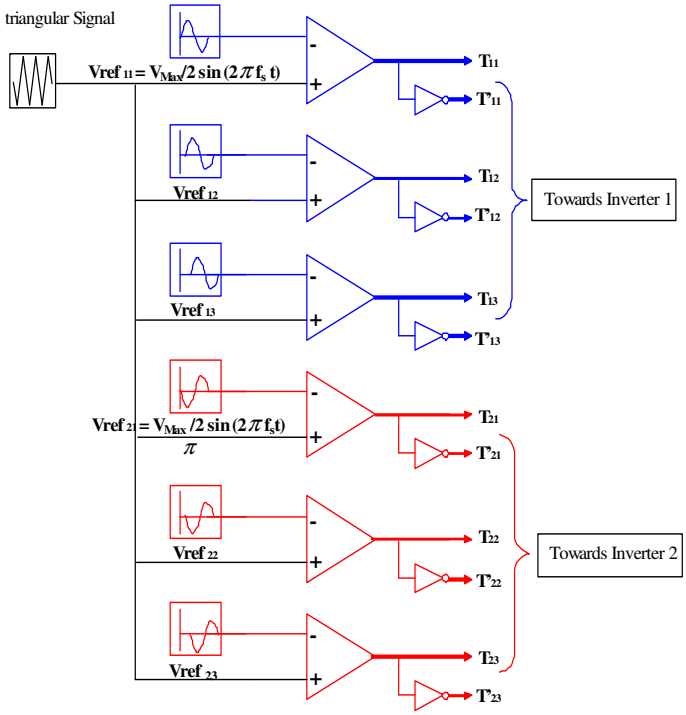


Fig. 2. Principle of the PWM sine triangle.

The principle of the scalar control "V / f law" is to vary the speed of the machine while holding constant the ratio of the voltage about the stator frequency.

In the following, we will interest in the numerical algorithm of the "V / f law", for this we must generate three voltages of variable references in amplitude and frequency during the acceleration phase which corresponds to the transitional regime.

The three reference signals are compared with a triangular carrier for the command signals. Then just create the dead time in order to avoid any short circuit between the different switches.

The different blocks necessary to obtain the control signals for each entry of the stator windings, are shown in figure 3.

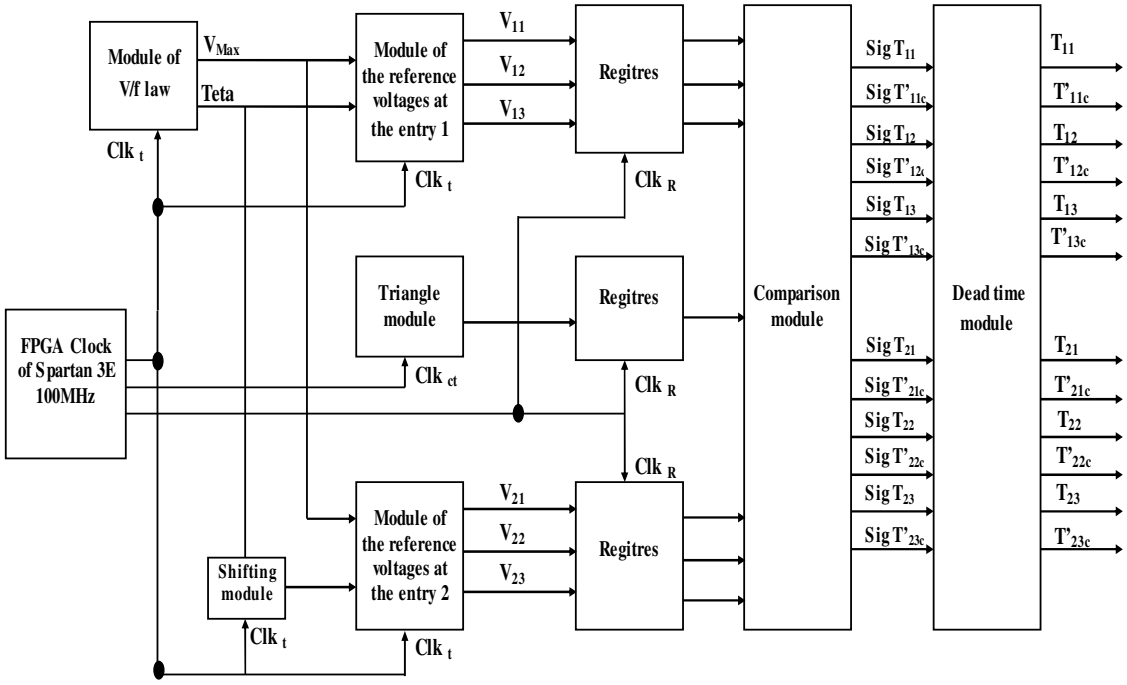


Fig. 3. Diagram of the scalar control "V / f law".

By the figure 3, the diagram of the digitization of the scalar control "V / f law" groups the following modules:

- Triangle module
- Generation module of the variation law
- Generation module of the reference voltages
- Comparison module
- Module of the dead time
- Module of the shift 180 °.

2.1. Triangle module

This module generates a triangular carrier of the frequency and fixed amplitude. The triangular design of a signal is based on the block counter / down counter which counts from 0 to (N-1) / 2 and count of (N-1) / 2 to 0, is shown in the figure 4.

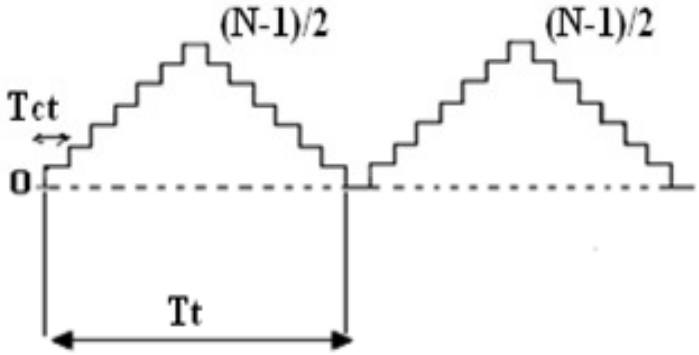


Fig. 4. Principle of a digital triangular carrier.

With :

- T_{ct} : the period of the counter and down counter
- T_t : the period of the triangular carrier = chopping period of the inverter
- f_t : the frequency of the triangular carrier
- f_{ct} : the frequency of the counter and down counter
- n : number of bits of the counter / down counter
- $N-1 = 2^n-1$: modulo counter / down counter

We will use an operating frequency identical to that used for the simulation in the Matlab Simulink environment, is a frequency triangular signal f_t equal to 3150Hz. This value will be adjusted to 3258Hz order to have all periods of the operation of the integer values.

$$T_t = \frac{1}{f_t} = 306900\text{ns} \qquad (1)$$

The clock period of the counter can be determined according to the period of the triangular carrier by the following relationship:

$$T_t = (N-1).T_{ct} \qquad (2)$$

So we must determine the period counter and down counter as equal:

$$T_{ct} = \frac{306900 \ .10^{-9}}{1023} = 300\text{ns} \qquad (3)$$

$$f_{ct} = \frac{1}{T_{ct}} = 3.333\text{MHz} \qquad (4)$$

We use a RS flip-flop to control the counter / down counter block. The output of this flip-flop is set to 1 if the counter reaches the maximum amplitude $\frac{2^{10}-1}{2} = 512$.

The block counter fonctions as a down counter to the minimum value which is equal to 0.

The detection of the maximum count value (512) and the minimum value of counting (0) is provided by the two comparison blocks C1 and C2 of the figure 5. According to the state of the two comparators, we must give orders (sel) to the multiplexer across the RS flip-flop to indicate the operating state of the up / down counter.

The module of the triangular signal is mainly constituted by an counter, an RS flip-flop, a multiplexer and two comparators. In the figure 5, this module is presented in System Generator.

We show the operation of the RS flip-flop and block counter / down counter in the following truth table.

Table 1: Truth table of the RS flip-flop.

S	R	sel	Operation
0	1	0	Counter
0	0	0	
1	0	1	Down counter
0	0	1	

- If the selection entry "sel" equals 0 then the multiplexer passes in its output, input $d_0 = 1$ that forces the input up / down counter to 1, this latter it starts counting to the max value.
- If the input "sel" is equal to 1 then the multiplexer passes in its output, input d1 = 0 that forces the input up/ down counter to 0, to get this up/down counter in phase down counting up to 0.

From the truth table of the RS flip-flop, we can extract the following equation:

$$Q_n = \overline{R}.S + \overline{R}.Q_{n-1} \qquad (5)$$

After the output of the counter, we added a block 'scale' and a block of subtraction that allows to have a triangle whose amplitude varies between (-1) and 1 to center the triangle obtained on time axis, thereafter the amplitude of the triangle is regulated by adding a multiplication block whose the constant is equal to Vmax. The simulation result of the triangular signal is shown in the figure 5.

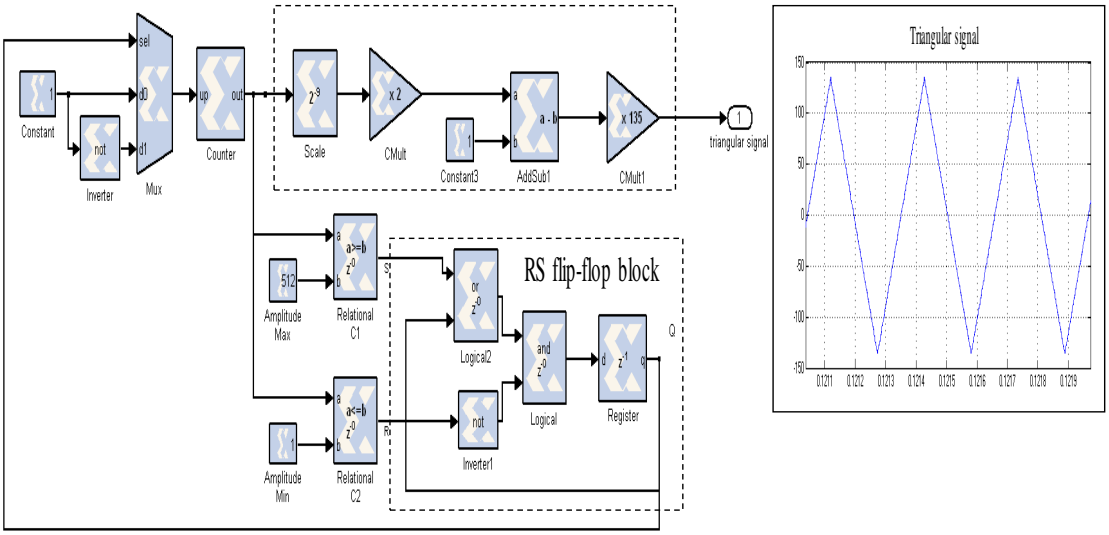


Fig. 5. Carrier triangular block and simulation result.

2.2. Generation module of the variation law and of the angle θ

For the realization of this module, we desire have the possibility to vary at starting, the value of the frequency and the acceleration time. In the figure 6, we represent the general generation module of the variation law of the acceleration time and the angle θ.

The design of this module groups the following blocks:

- Generation Block of the variation law of the frequency and of the acceleration time

With the aid of the library "system generator", we carry a slope "f / tacc" with saturation. To do this we use a 12-bit adder block and a delay block that replaces the ramp under "Simulink". Then we have a comparator that will block its 4087 counter maximum value for the saturation which fixes the value of the frequency f. Initially, the output of the adder is equal to 1, from where the input 'en' of the adder is forced to 1; it starts to increment up to maximum value 4087. Thereafter for pass the blocking phase, the comparison block 4087 detects the maximum value and the input 'en' goes to 0.

To fix the value of the desired frequency, we multiplied by a constant K_2 ($K_2 = \frac{1}{2^{12}-1}$) which allows us to have a unit amplitude. Then we inserted a multiplication block to specify the value of the frequency f. For the voltage, we have multiplied by a constant equal to V_{Max}.

- Generation block of the angle θ

The angle θ is always varying between 0 and 2π , then to determine the angle θ, we have the following relationship:

$$\theta = \int \omega \, dt = \int 2\pi \, f \, dt \qquad (6)$$

ω : angular speed (rad.s^{-1})

f : frequency (Hz)

In digital, the integration requires the multiplication of the entries of integrator block by a sampling period Te, as shown in the equation below:

$$I(z) = T_e . \frac{z}{z-1} = \frac{S(z)}{E(z)} \qquad (7)$$

His recursion equation is:

$$s(k) = s(k-1) + T_e . e(k) \qquad (8)$$

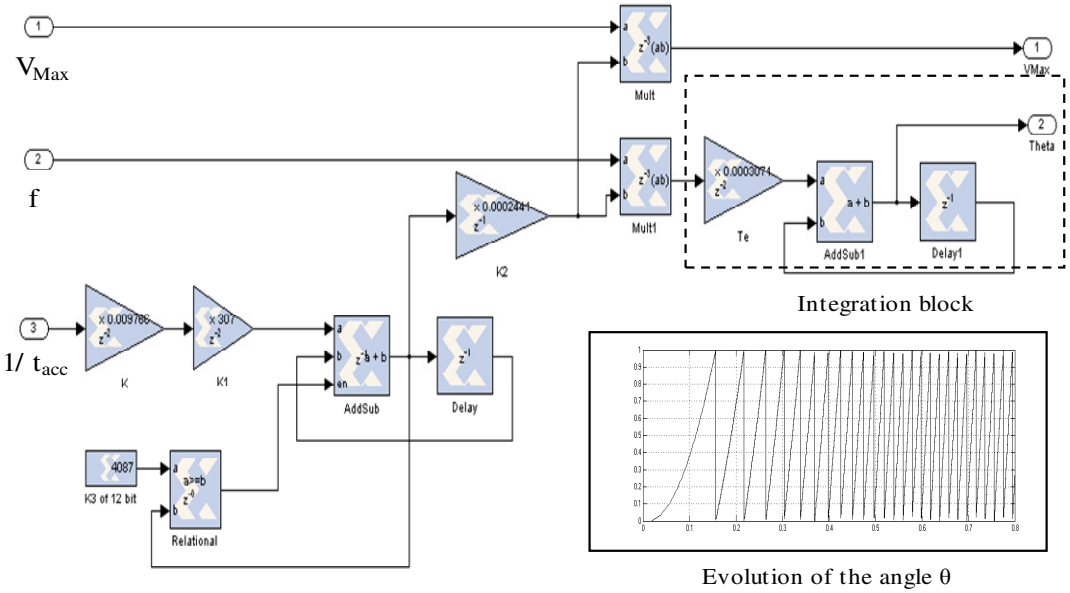


Fig. 6. Generation block of the angle θ.

2.3. Generation module of the reference voltages

To obtain the reference voltages (Vs_{11}, Vs_{12}, Vs_{13}), we used the 2/3 transformation from the two voltages (Vsα, Vsβ).

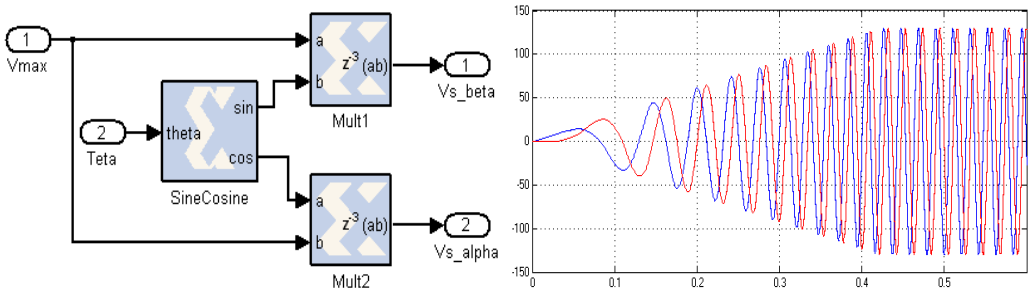
2.3.1. Generation module of two voltages Vsα and Vsβ

This module generates two sinusoidal signals varying in amplitude and frequency with the following equations:

$V_{s\alpha}=V_{max}\cos(\theta)$ (9)

$V_{s\beta}=V_{max}\sin(\theta)$ (10)

For this module, we have a block 'SineCosine' having such as entered the angle θ, the block generate in outputs of the values of sine and cosine between (-1) and 1. The general structure of the generation module voltages Vsα , Vsβ under "Generator System" is represented in the figure 7(a) and the evolution of the two voltages by figure 7 (b).



(a) (b)

Fig. 7. (a) Generation module of the voltages Vsα, Vsβ; (b) Evolution of the two voltages.

2.3.2. Generation module of the reference voltages

The 2/3 transformation is based on the following mathematical relationship:

$$V_{s11} = [V_{s\alpha}\cos(\theta) - V_{s\beta}\sin(\theta)]. \quad (11)$$

$$V_{s12} = \frac{1}{2}V_{s\alpha}.((-\cos\theta) + \sqrt{3}\sin\theta) + \frac{1}{2}V_{s\beta}.(\sin\theta + \sqrt{3}\cos\theta) \quad (12)$$

$$V_{s13} = -(V_{s11} + V_{s12}) \quad (13)$$

With the aid of "System Generator" toolbox, we will implant the mathematical relationships of the 2/3 transformation. We have a block 'SineCosine' which generate in the output the values of sin (θ) and cos (θ). Thereafter, we apply the mathematical relationships using blocks of multiplication, addition ... the representation of the block is shown by the figure 8. The clock period T_h used for these blocks is equal to the sampling period where T_h = 306900 ns.

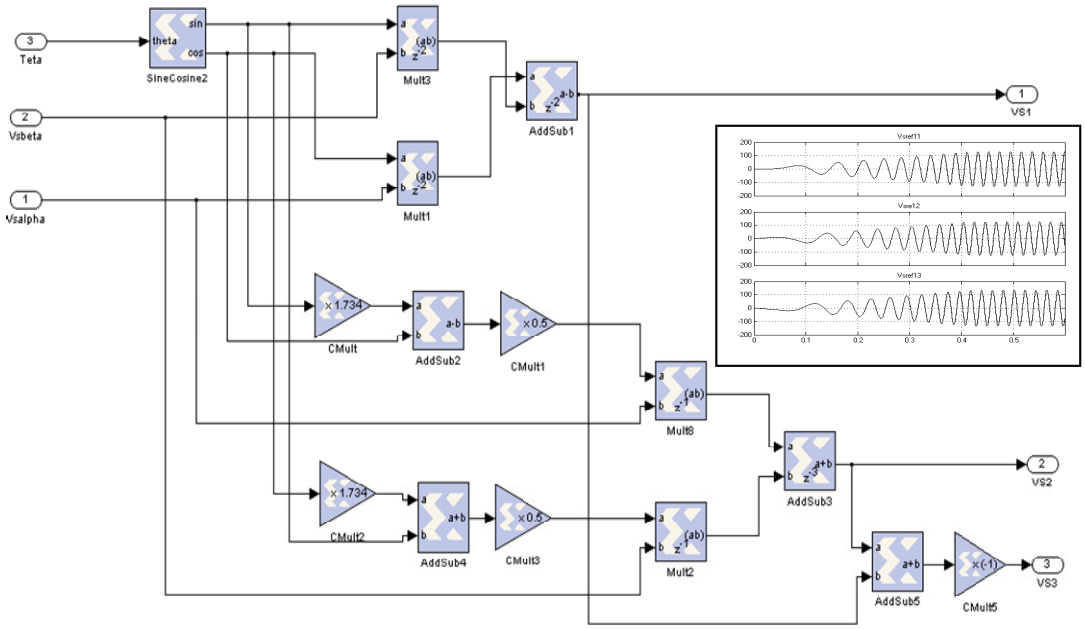


Fig. 8. Generation block of the reference voltages.

2.4. Generation module of control signals of entry 1

Before comparing the voltages of the sinusoidal reference with triangular carrier, it is essential to synchronize the period of the reference voltages with that of the carrier with using a register block. Also the registers of the reference voltages and carrier must have the smallest sampling period which is the carrier Tct = 300e-9ns.

After the synchronization of the different signals, we move to the comparison module, to generate the control signals with complementary signals, is represented by the figure 9.

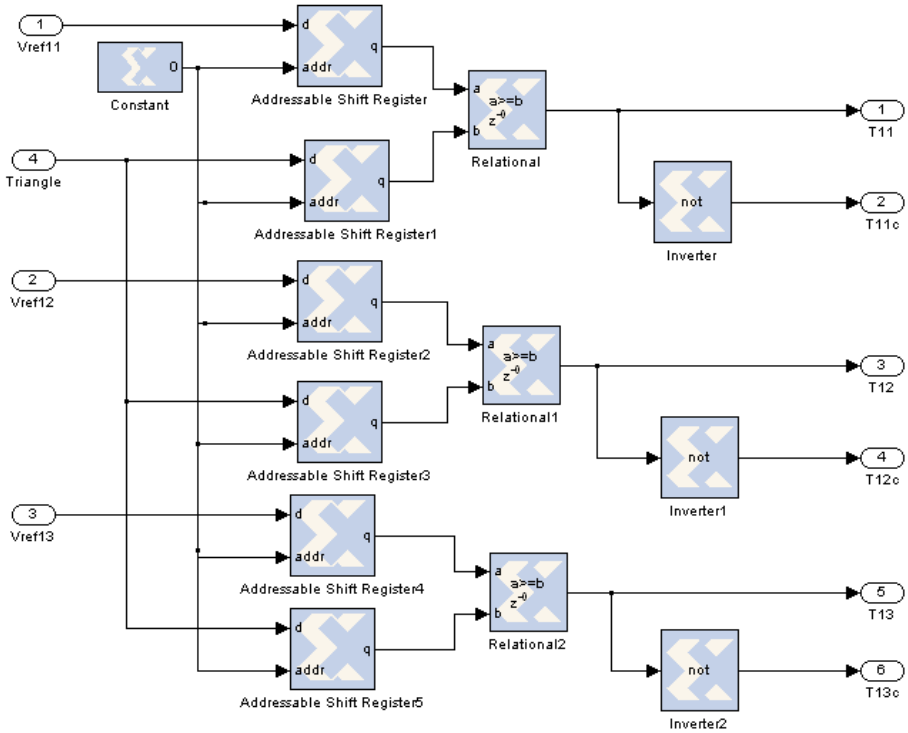


Fig. 9. Generation module of the control signals.

2.5. Module of the dead time

The switches of each arm of the inverter being complementary, but their opening time and the closing time are not zero. To avoid short circuit when commutation of the switches we have introduced a dead time module.

The dead time block consists of a block 'Delay' and a port logic 'AND'. This module insert a dead time at the beginning of each command at the switch close. The principle of this module is given in the figure 10.

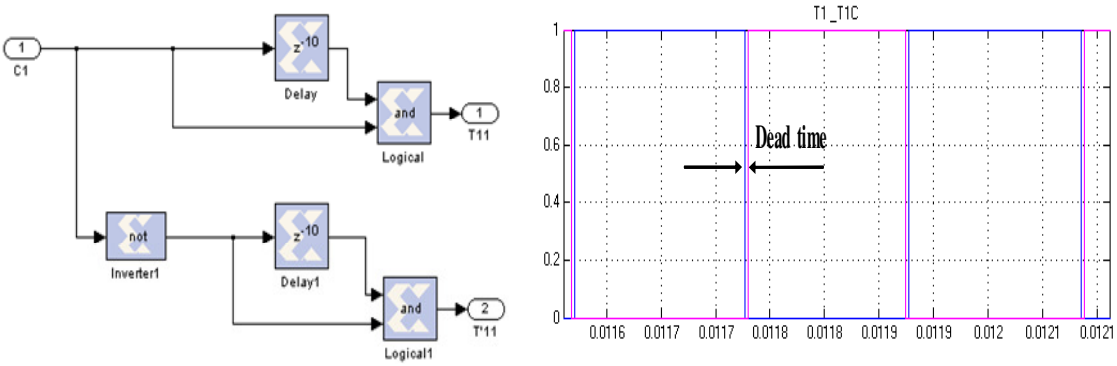


Fig. 10. Generated control signals by the dead time module.

2.6 Shift module of 180 °

To ensure the supply of entry 2 of the machine, we have to introduce a shift of 180° to the generation module of the reference voltages (V_{S11}, V_{S12}, V_{S13}).

Then to introduce a shift of 180 °of the angle θ, we have shift the $T/2$ period , we used it for a subtraction block which calculates the difference between the generated angle θ from 0 at 2π for different times and the constant 0.5 to generate new shifting by 180 °.

For the subtraction block must keep the same parameters as the addition block and of integration in the generation module of the angle θ is shown in the figure 6. For the transitional regime (t < 0.5s), we have a variable period which results the shifting variable but proportional at the period with a proportionality coefficient equal to 0.5. Once the three reference signals shifted by 180 ° are obtained, we do the same thing for the command signals of the inverter 2. The figure 11 shows the simulation results of the command signals of arm 1 of the inverter 2 shifted by 180°.

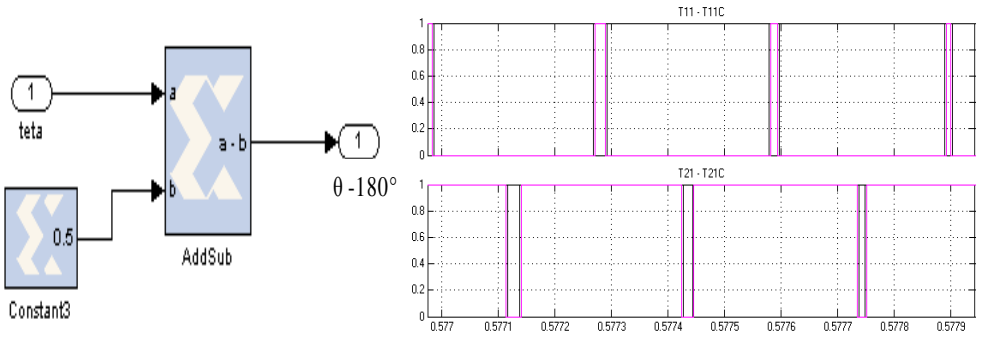


Fig. 11. Command signals shifted of 180°

3. Validation of the scalar control for a open-end winding induction machine

The scalar control controlled the open-end winding induction machine, whose controller is embedded into Xilinx Spartan-3E FPGA by means of the Xilinx System Generator Toolbox; the simulation model of scalar control is validated in the Matlab simulink environment. Design details of the controller developed using Xilinx System generator are provided in the previous sections.

The following cycle of operation, at t = 0.5s, the system has a starting cycle, from t = 0.5s to t = 1s, the machine in working in no-load conditions. At time t = 1s, a load torque Tr = 300mN.

Figure 12 shows the pole voltage inverter 1 (Vs_{11}-Vs_{12}), inverter 2 (Vs_{21}-Vs_{22}) and pole voltage machine U_1= (Vs_{11}-Vs_{12}) - (Vs_{21}-Vs_{22}), the currents stator Isa, Isb, speed and the torque, for load torque Tr = Tn.

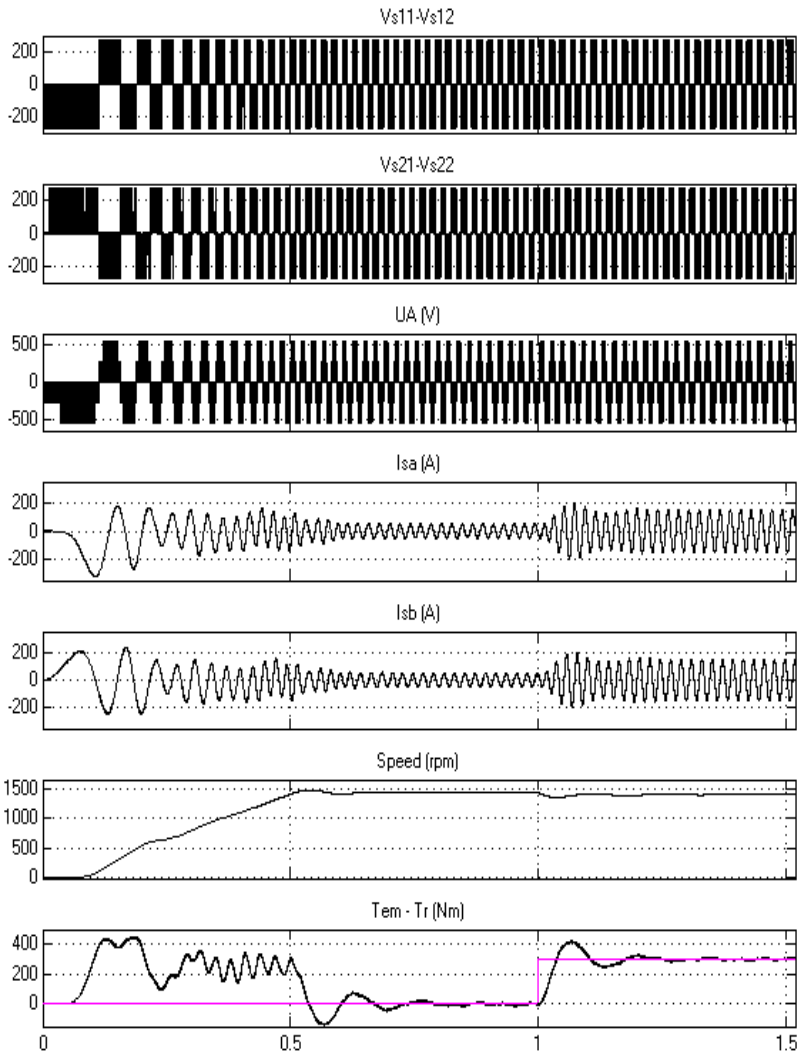


Fig. 12. Evolution of voltage, currents of stator, speed, and torque.

The characteristics of the machine used:
- Nominal power P = 45 KW.
- Speed n = 1450 rpm.
- Resistance of stator Rs = 0.15 Ω.

- Resistance of rotor Rr = 0.046 Ω.
- Inductance of stator Ls = 17.9 mH.
- Inductance of rotor Lr = 18.6 mH.
- Mutual inductance Msr = 17.2 mH.

4. Experimental results of the FPGA implementation of the scalar control

To validate the digitalization of scalar control for the open-end winding induction machine, we represent the implementation of the command using an evaluation board Nexys 2 based on FPGA of the family SPARTAN 3E-XC3S1200e-4fg320 of the society XILINX, figure 13.

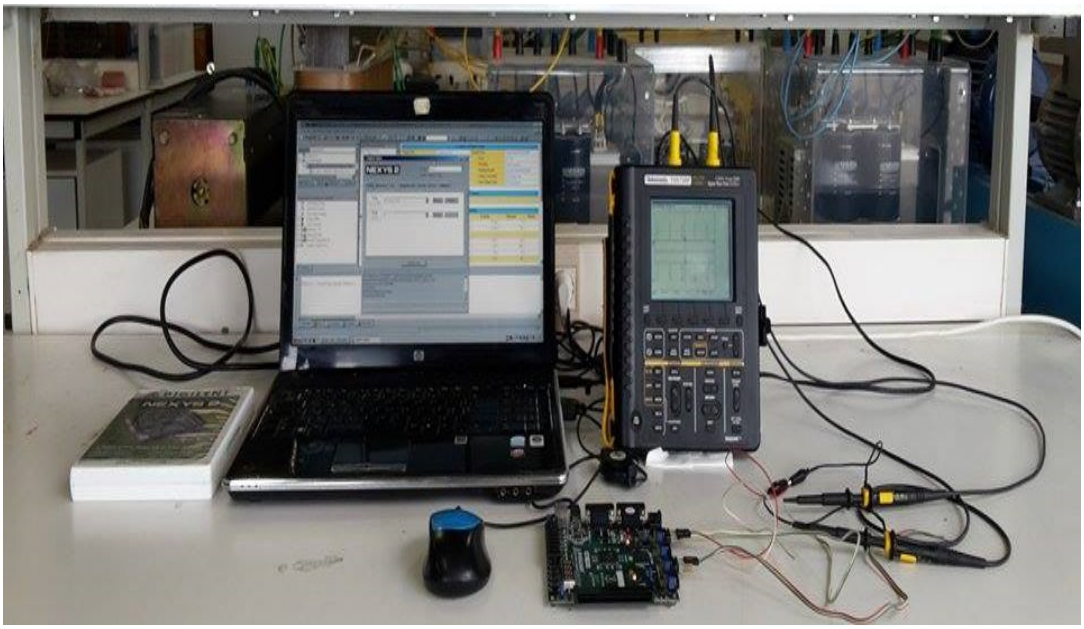


Fig. 13. Implementation of the command algorithm on FPGA.

The report "Map" is shown by the table 2 of the VHDL code of the scalar control that shows the effectiveness of FPGA Spartan3E for this sort of application. This phase allowed to minimize the resources consumed.

Table 2: The report "Map" of the VHDL code

Number of Slice Flip Flops	**2,131 out of 17,344**	**12%**
Number of 4 input LUTs	**3,530 out of 17,344**	**20%**
Number of occupied Slices	**2,041 out of 8,672**	**23%**
Number of Slices containing only related logic	**2,041 out of 2,041**	**100%**
Number of Slices containing unrelated logic	**0 out of 2,041**	**0%**
Total Number of 4 input LUTs	**3,780 out of 17,344**	**21%**
Number used as logic	**3,327**	
Number used as a route-thru	**250**	
Number used as Shift registers	**203**	
Number of bonded IOBs	**13 out of 250**	**5%**
Number of RAMB16s	**3 out of 28**	**10%**
Number of BUFGMUXs	**1 out of 24**	**4%**

The experimental results of the different command signals for the arm 1 of inverters 1 and 2 are shown by figures 14, 15 and 16. These results are obtained at the output of the FPGA Spartan 3E of the board Nexys 2 for the carrier frequency 3258 Hz and dead time equal to 3µs.

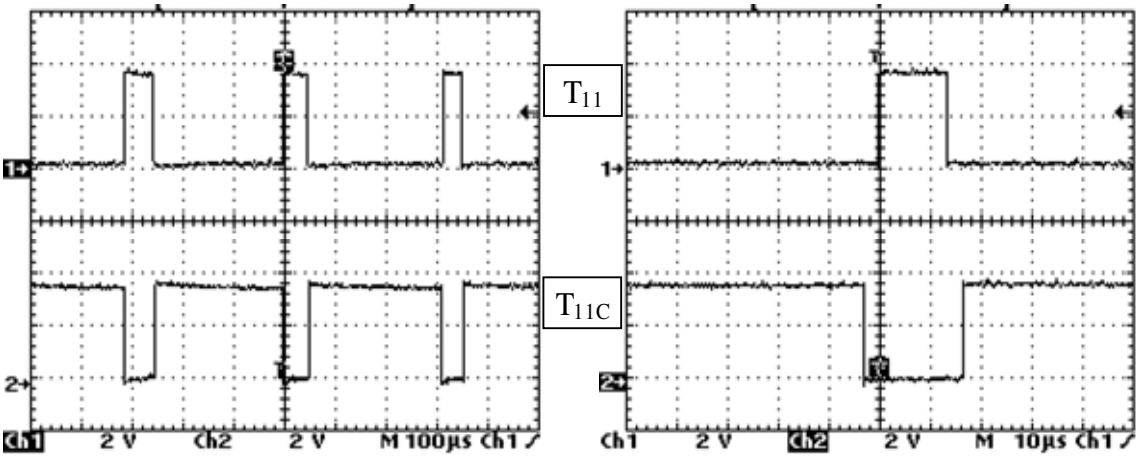


Fig. 14. Command signals of the switches T_{11} and T_{11C}

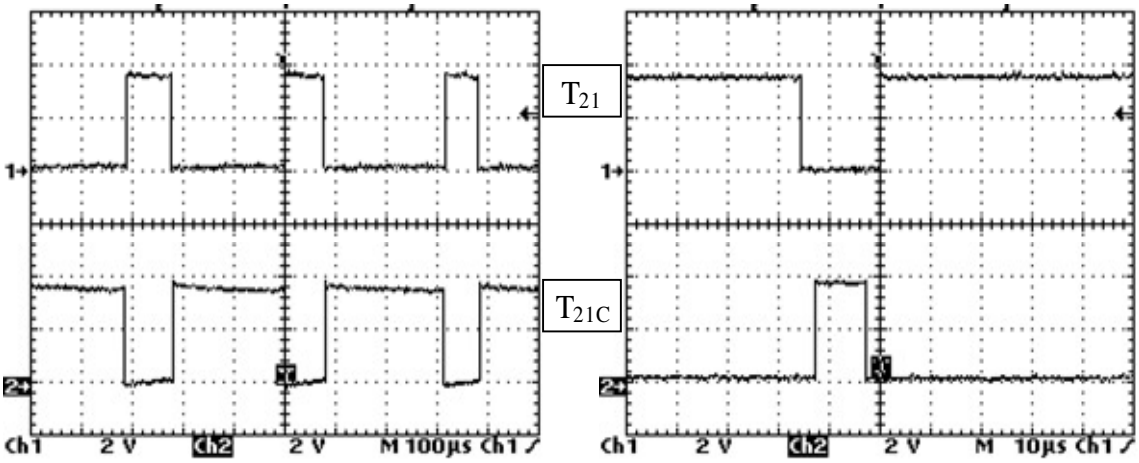


Fig. 15. Command signals of the switches T_{21} and T_{21C}

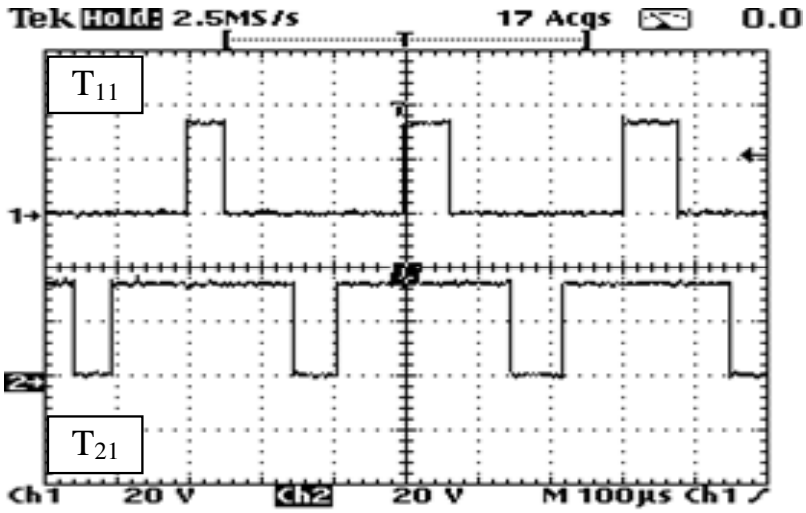


Fig. 16. Command signals of the switches T_{11} and T_{21}.

5. Conclusion

The preparation of an algorithm allowing for the scalar control of the open-end winding induction motor is investigated. Indeed, the realized program is validated in the MATLAB Simulink environment using the Xilinx System Generator.

The simulation model of the open-end winding asynchronous machine supplied by two voltage sources based on V/f law in the environment of « Matlab Simulink » using the Xilinx System Generator is presented.

The program of scalar control is implemented on the Spartan 3E FPGA board while specifying the different steps. The experimental results of the different command signals for the each inverter are visualized and verified the dead time between command signals of the switches.

References

[1] G.K.Singh, V. Pant and Y.P. Singh, Voltage source inverter driven multi-phase induction machine, *Computer and Electrical Engineering Elsevier*, Vol. 29, pp. 813-834, 2003.

[2] A.S. Abdel-Khalik, A.S. Morsy, S. Ahmed and A.M. Massoud, Effect of stator winding connection on performance of five-phase induction machines, *IEEE Transactions on Industrial Electronics*, Vol. 61, pp. 3 - 19, 2014.

[3] F. Ben Ammar and S. Guizani, The Improvement Avalability of a Double Star Asynchronous Machine Supplied by redondant voltage source inverters, *Journal of electrical system JES*, Vol. 4, 2008.

[4] K. Marouani, L. Baghli, D. Hadiouche, A. Kheloui and A. Rezzoug, A New PWM Strategy Based on a 24-Sector Vector Space Decomposition for a Six-Phase VSI-Fed Dual Stator Induction Motor, *IEEE Transactions on Industrial Electronics*, Vol. 55, pp.1910-1920, May 2008.

[5] V. T. Somasekhar, K. Gopakumar, M. R. Baiju, K. K. Mohapatra and L. Umanand, A Multilevel Inverter System for an Induction Motor With Open-End Windings, *IEEE transactions on industrial electronics (2005),* Vol. 52, pp. 824-836, 2005.

[6] A. Nayli, S. Guizani and F. Ben Ammar, Open-end Winding Induction Machine Supplied by Two Flying Capacitor Multilevel Inverters". IEEE Conference on Electrical Engineering and Software Applications ICEESA 2013,Hammamet, Tunisia, March 2013.

[7] S. Guizani, A. Nayli, and F. Ben Ammar, Fault-Tolerant control for Open-end Stator Winding Induction Machine supplied by two three phase cascaded inverters with one failed inverter, *Journal of Electrical Engineering JEE* , Vol. 14, 2014.

[8] J. Kalaiselvi and S. Srinivas, Bearing Currents and Shaft Voltage Reduction in Dual-Inverter-Fed Open-End Winding Induction Motor With Reduced CMV PWM Methods, *IEEE Transactions on Industrial Electronics*, Vol. 62, pp. 144-152, January 2015.

[9] S. Guizani and F. Ben Ammar, the dual open-end winding induction machine fed by quad inverters in degraded mode, *International Journal of Scientific and Engineering Research IJSER*, Vol. 4, 2013.

[10] S. Guizani and F. Ben Ammar, Dual open-end stator winding induction machine fed by redundant voltage source inverters, *Turkish Journal of Electrical Engineering & Computer Sciences,* 2013, accepted.

[11] M. Cirstea, A. Aounis, M. McCormick, P. Urwin and L. Haydock, Induction Motor Drive system Modelled in VHDL", IEEE Conference on Proceedings VHDL International Users Forum Fall Workshop 2000, Orlando, FL, pp. 113-117, October 2000.

[12] F. Ricci and H.L. Luy, An FPGA-Based Rapid Prototyping Plaform for Variable-Speed Drives, IEEE Conference on Industrial Electronic Society IECON 2002, Vol.2, pp. 1156-1161, November 2002.

[13] O. AKIN and A. Irfan, The use of FPGA in field-oriented control of an induction machine, *Turkish Journal of Electrical Engineering & Computer Sciences*, Vol.18, 2010.

[14] B. Badre, K. Mohammed, I. Silviu and L. Ahmed, Low- speed sensorless control of PMSM motor drive using a nonlinear approach backstepping control: FPGA-based implementation, *Journal of Theoretical and Applied Information Technology,* Vol. 36, 2012.

[15] T. Sutikno, N. Idris, A. Jidin, and M. Cirstea, An improved FPGA implementation of direct torque control for induction machines, *IEEE Transactions on Industrial Informatics*, Vol.9, pp. 1280-1290, 2013.

[16] M. A. Zare, R. G. Kavasseri and C. Ababei, FPGA–based design and implementation of direct torque for induction machines, International Conference Reconfigurable Computing and FPGAs (ReConFig) 2014, Cancun, pp. 1-6, December 2014.

[17] Xilinx Corp, «Xilinx System Generator V10.1 Reference Guide for Simulink» www.xilinx.com.