Lazhar Manai[1,2],Mohamed Alaa Eddine Hasni[1], Faouzi Armi[1,2],Mongi Besbes[1,2]

Laboratory of Robotics, computing and Complex Systems (RISC), National Engineering School of Tunis, University Tunis El Manar (1),
Higher Institute of information's and communication Technologies of Borj Cedria, Carthage University (2),

doi:10.15199/48.2017.04.32

FPGA-Based 2D Addressing Mode in Programmed Matrix PWM for Multilevel Inverter Control

***Abstrac**t. Conventionally, control signals fed to the PWM inverters are produced by electronic hardware or microprocessor. The cost and complexity of hardware implementation are considerable and control algorithms differ for PWM for different levels and topologies is proposed. On-line computation in the microprocessor approach is laborious and time-consuming. These two approaches seem impractical when they are applied to PWM multilevel inverter control. In this paper, programmed matrix PWM based on two dimensional addressing modes for FPGA memories can solve the problems mentioned above and provides easy, fast and steady control. Experimental results are carried out to confirm the high performance of the proposed embedded PWM.*

Streszczenie.** W artykule zaproponowano programowalny macierzowy przekształtnik PWM bazujący na dwuwymiarowym adresowaniu pamięci FPGA. Otrzymano wbudowany moduł PWM znacznie prostszy od typowych rozwiazań. **Dwuwymiarowo programowany macierzowy przekształtnik PWM wykorzystujacy układy FPGA

Keywords: FPGA, 2D addressing mode, programmed matrix pulse width modulation (PMPWM), multilevel inverter
Słowa kluczowe: FPGA, układ PWM, wielopoziomowy przekształtnik

Introduction

Since the concept of multilevel PWM converter was introduced, various modulation strategies have been developed and studied in detail, such as multilevel sinusoidal PWM, multilevel selective harmonic elimination, and space vector modulation. Among these strategies, the most widely used are the space vector PWM (SVPWM) [1], and the carrier-based PWM method. SVPWM and the carrier-based PWM method offer significant flexibility to optimize switching waveforms, but complexity and computational cost of traditional SVPWM and the carrier-based methods techniques, increase with the number of levels of the converter, and most of all use trigonometric functions or pre-computed tables [2,3].

Many researches chose to implement PWM by digital signal processor (DSP) or microcontroller (MCU). This approach has the advantages of simple circuitry, software realization and flexibility. However, there are also several disadvantages [4]. As the levels of the inverter increase and the inverter structure becomes more complex, the programming of the corresponding PWM in the DSP or MCU becomes one of the most time-consuming tasks [5].

If the DSP or MCU is not able to provide enough on-chip peripherals, such as comparators and dead-time controllers to support the PWM outputs, extra hardware circuits need to be designed to cooperate with the controller [6].

An attractive idea is to implement the PWM via an application-specific integrated circuit (ASIC). The field programmable gate array (FPGA) is a sub-class of ASIC controllers which provides characteristics such as fast prototyping, simple hardware and software design and higher switching frequency [7].FPGAs development reached a level of maturity that made them the choice of implementation in many fields [7,8].

In this paper, a new method based on 2D addressing mode in a programmed matrix pulse width modulation (PMPWM) is proposed to control four-level inverter given by figure1. Matrix PWM algorithm is analyzed and implemented into FPGA memories based on a two dimensional addressing mode. This algorithm allows not only to control more and more switches in parallel of a three phase's multilevel inverter, but it can solves the problems mentioned above and provides easy, fast and steady control

The main objective of the FPGA- based 2D addressing mode in programmed matrix PWM is to produce the multilevel inverter control signals. Matrix PWM programming into addressable FPGA memories is developed as follows: each pulse is composed of several consecutive of '1' to represent its width. Times delays between pulses are represented by a concatenation of '0' , in this way digitized PWM patterns takes a matrix form, programmed and implemented into FPGA memories.

As FPGA memories cells comprise as many rows as columns, in this case, a two dimensional addressing mode can be adopted for matrix PWM in FPGA memories, its advantage is that the same address rows can be used to identify successively the row then the column and if we want to reach the information stored in the same row, we need to send only the address of each column. This addressing mode allows to reduce the number of the logical gates allocated for the FPGA memories cells, consequently the congestion and time-consuming [9, 10].

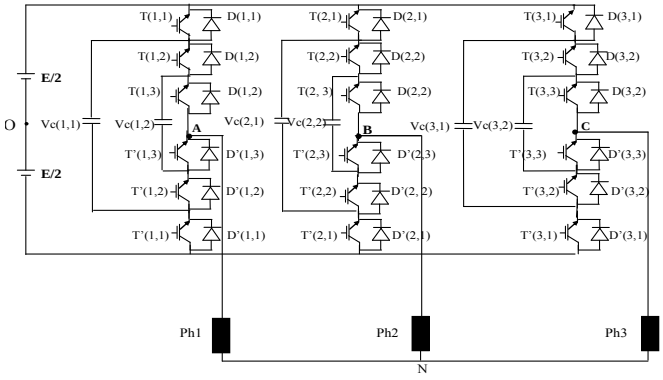

Fig. 1. Three-phase four- level inverter

The remainder of this paper is organized as follow. In Section 2, we present the expression of the multilevel inverter output voltages. In Section 3, we develop the programmed matrix PWM algorithm based on two dimensional addressing mode. Simulation and experimental results are presented in section 4 and 5, respectively with some brief concluding remarks given in Section 6.

Four levels inverter output voltage expression

In our paper we give a particular attention on the implementation of pulses widths modulation strategy control for a three-phase 4 levels inverter. Concerning the topology of this inverter, every phase is constituted of p=3 cells, 2 flying capacitors and 6 power switches as shown in figure 1. Each two switches T(i,j) and T'(i,j) (i=1:3 is the number of the phases; j=1: p is the number of commutations cells) have complementary control signals S(i,j) and S'(i,j) [11].

The line-line output voltages V_{AB}, V_{BC} and V_{CA} are given by:

$$\begin{bmatrix} V_{AB} \\ V_{BC} \\ V_{CA} \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ -1 & 0 & 1 \end{bmatrix} . \begin{bmatrix} V_{AO} \\ V_{BO} \\ V_{CO} \end{bmatrix} \quad (1)$$

Such as:

$$V_{AO}(t) = \frac{E}{2} - \sum_{j=1}^{j=3} V_T(1,j) \quad (2)$$

$$V_{BO}(t) = \frac{E}{2} - \sum_{j=1}^{j=3} V_T(2,j) \quad (3)$$

$$V_{CO}(t) = \frac{E}{2} - \sum_{j=1}^{j=3} V_T(3,j) \quad (4)$$

where $V_T(i,j)$: is the voltage across the switch of j^{th} cell of the i^{th} phase.

The per phase output voltage of the four-level inverter V_{AN},V_{BN} and V_{CN}, are expressed in terms of the DC link voltage E, the number of the commutation cells p and the switching function S(i,j) as:

$$\begin{bmatrix} V_{AN} \\ V_{BN} \\ V_{CN} \end{bmatrix} = \frac{E}{3p} \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix} \begin{bmatrix} \sum_{j=1}^{3} S(1,j) \\ \sum_{j=1}^{3} S(2,j) \\ \sum_{j=1}^{3} S(3,j) \end{bmatrix} \quad (5)$$

The simulation in MATLAB-SIMULINK of the 4 level inverter with PWM modulation control shows in figure 2 that the per phase output voltage waveform is optimized and formed of 4 levels voltages.

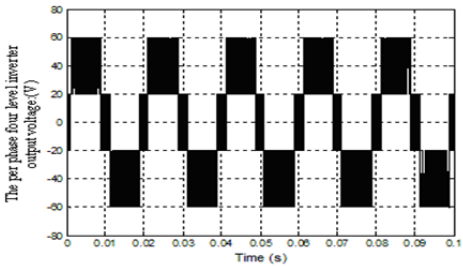

Fig .2. The per phase output voltage of the four levels inverter

Matrix PWM algorithm based on two dimensional addressing mode

In this section we develop the per phase programmed matrix PWM control of the four level inverter. For the digital conception, the programmed matrix PWM patterns are implemented into FPGA memories cells in a matrix form. The PWM patterns control of the 1st commutation cell of the four-level inverter are presented by the matrix [M1] given by equation (6) and implemented into FPGA memory.

$$[M_1] = \begin{bmatrix} 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & a_{1,n} & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & a_{2,n-1} & a_{2,n} & a_{2,n+1} & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & a_{h,n-(h-1)} & \cdots & \cdots & a_{h,n-1} & a_{h,n} & a_{h,n+1} & \cdots & \cdots & a_{h,+n+(h-1)} & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & a_{n-2,3} & \cdots & a_{n-2,n-(h-1)} & \cdots & a_{n-2,n-2} & a_{n-2,n-1} & a_{n-2,n} & a_{n-2,n+1} & a_{n-2,n+2} & \cdots & a_{n-2,n+(h-1)} & \cdots & a_{n-2,\,2n-2} & 0 & 0 \\ 0 & a_{n-1,2} & a_{n-1,3} & \cdots & \cdots & \cdots & a_{n-1,n-2} & a_{n-1,n-1} & a_{n-1,n} & a_{n-1,n+1} & a_{n-1,n+2} & \cdots & a_{n-1,n+(h-1)} & \cdots & a_{n-1,2n-2} & a_{n-1,2n-1} & 0 \\ a_{n,1} & a_{n,2} & a_{n,3} & \cdots & a_{n,n-(h-1)} & \cdots & a_{n,n-2} & a_{n,n-1} & a_{n,n} & a_{n,n+1} & a_{n,n+2} & \cdots & a_{n,n+(h-1)} & \cdots & a_{n,2n-2} & a_{n,2n-1} & a_{n,2n} \end{bmatrix} \quad (6)$$

The PWM matrix memory [M1] is organised with n rows and 2n columns generates n different pulse width represents the different possible control signals of the first commutation cell and stored in the Integrated Programmable FPGA memory.

To simplify [M1] calculation, n is selected such as:

$$n = 2^q ; \quad q \geq 7 \quad (7)$$

Where q is the number of the address inputs of the FPGA memory to be used.

Each row of this matrix represents the commutation cell control signal and having T_{CLK_C} as duration. f_{clk_c} is the obtained control signal frequency system by dividing the FPGA frequency clock based on programmed frequency divider. Thus, [M1] represents n PWM control signals. [M1] components are binaries digits. Each component is equal to ''1'' or '' 0'' . Consequently the [M1] components given by equation (8) formed a triangle of 1 in the matrix [M1] and represents n different duty cycles.

$$a_{h,\,n\pm(h-1)} = 1 \;\; \forall \;\; h \in [1 \;\; n] \quad (8)$$

In the same time interval, to create the matrixes [M2] and [M3] to control the 2cd and the 3rd commutation cell respectively, it is just enough to shift the matrix [M1] a third of row to the right to obtain [M2] and to the left to obtain [M3]. [M2] and [M3] components are given by equation 9 and 10 respectively:

$$a_{h,\,n\pm(h-1)\bmod\left[\frac{2n}{3}\right]} = 1; \;\; \forall \; h \in [1 \;\; n] \quad (9)$$

$$a_{h,\,n\pm(h-1)\bmod\left[\frac{4n}{3}\right]} = 1 \;\; \forall \;\; h \in [1 \;\; n] \quad (10)$$

In this case, dimension of [M1] is given such as:

$$\text{Dim}\,[M1] = \text{Dim}\,[M2] = \text{Dim}\,[M3] = 2n * n = 2^{q+1} * 2^q = 2n^2 = 2^{2q+1} \quad (11)$$

The matrix PWM algorithm development is summarized by the flow chart in figure 3 presented below.

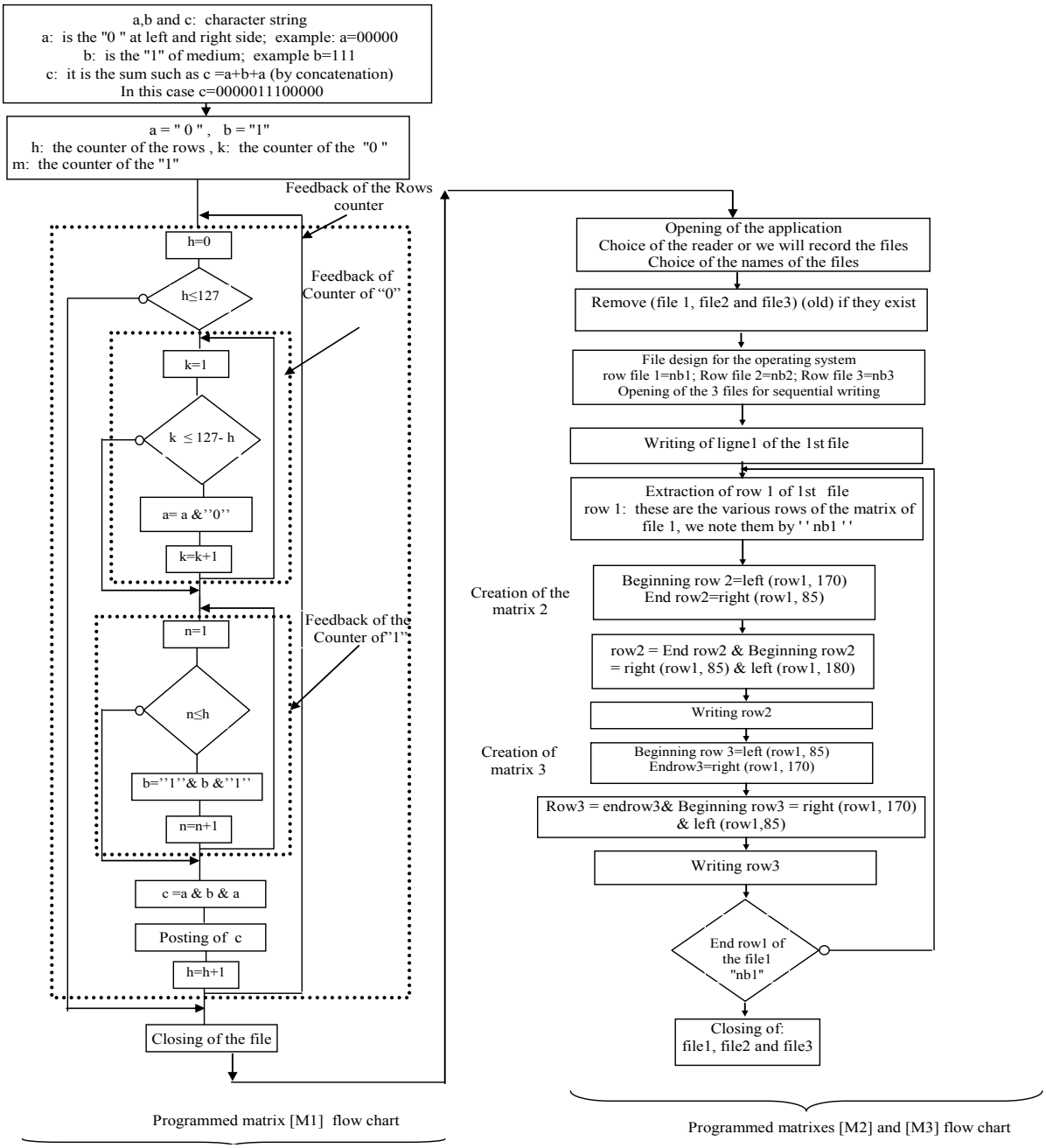

Fig.3. Flow chart of the programmed matrix PWM

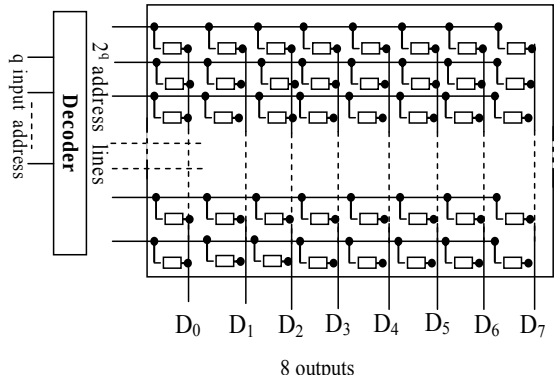

Fig.4. FPGA memory matrix addressing mode

In matrix addressing mode, for q address input we obtain 2^q row address at the decoder output. According to the address, the decoder activates one of the 2^q rows. Thus only the cells corresponds to the required address are selected and the memorised information is then available at the output [12].In this case, since in matrix structure we have n rows and 2n columns, the total number of the logical gates (Nb_lg) is calculated as follow:

$$Nb\_lg = 2n * n = 2^{q+1} * 2^q = 2n^2 = 2^{2q+1} \quad (12)$$

Figure 4, represents the logical gates organization in FPGA memory cell in a matrix addressing mode

This simple architecture is not the most economic in term of number of logical gate, reason for which we passed to another mode called two dimensional matrix addressing mode, noted by 2D matrix addressing mode, presented by Figure 5

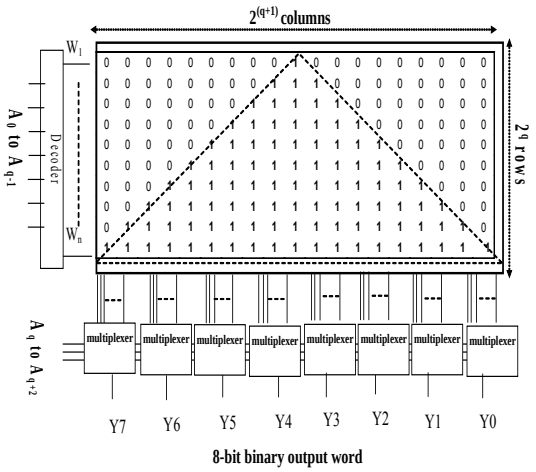

Fig.5. Programmed matrix PWM implemented into FPGA memories based on 2D addressing mode

Based on this architecture which is called also X-Y mode, a significant economy can be achieved by organizing the FPGA memory cell in a matrix form, composed by n rows and 2n columns and of dimension 2^{2q+1} in which each row of the 2^q rows can be selected by q-bit address [13]. As we want only at the output, words with 2^3 bits (8 bits), in this case, it is necessary to use 2^3 multiplexers. As result, multiplexer selects only one column from the $\frac{2^{q+1}}{2^3} = 2^{q-2}$ columns.In this case, the logical gates (Nb_lg) to be used in 2D addressing mode are calculated as follow:

$$\text{Nb\_lg} = \text{nb\_ rows} + \text{nb\_columns} + \frac{\text{nb\_column}}{\text{nb\_multiplixer}} \quad (13)$$

$$= 2^q(1+2+2^{-2})$$

where nb_rows, nb_columns and nb_multiplixer are the number of the rows, the columns and the multiplexer respectively to be used.

System response time in matrix and in 2D addressing mode are denoted by t_{matrix} and t_{2D} respectively, and calculated as follow:

$$t_{matrix} = (\text{nb\_ rows} * \text{nb\_columns})\, T_{CLK\_C} \quad (14)$$

In this case:

$$t_{matrix} = 2^{2q+1}\, T_{CLK\_C} \quad (15)$$

and:

$$t_{2D} = \frac{(\text{nb\_ rows x nb\_columns})\, T_{CLK\_C}}{\text{nombre of binaries digit by word}} \quad (16)$$

Whether:

$$t_{2D} = \frac{2^{2q+1}}{2^3} T_{CLC\_C} = 2^{2(q-1)} T_{CLC\_C} \quad (17)$$

As result, the logical gates to be used in the matrix PWM based on 2D addressing mode represents $\frac{2^{q+1}}{3}$ times less than it is used in the matrix one and time –consuming in 2D addressing mode represents 8 times less than matrix addressing mode

As we fixed in the experimental realization n= 128 and q= 7. In 2D mode the number of logical gates to be used is $\frac{32768}{392} = 83$ times less than that is used in the matrix one.

The functional blocks of the PMPWM patterns generation circuit is shown in Figure 6. This schematic shows how to implement these programmed matrixes into FPGA memories based on two dimensional addressing mode to generate the single phase four-level inverter control signals.

Matrix PWM control are clocked by the same 8-bit counter. Per phase, the PWM pulse patterns having n=128 different duties cycles ranging from 0% to 100% are implemented in FPGA memory cells. Each pattern is having a T_{CLK_C} duration that's corresponding to 2n=256 samples [14,15].

Matrixes PWM, Shift register, dead time block and D latch are programmed by using VHDL language to generate six control signals. Control signals are shifted a third of row of the programmed matrix. Complementary commutation cells are controlled by complementary control signals generated by the programmed D latch by taking into account of the dead time. The Dead Time logic block introduces a user defined dead time for the PWM signals in order to avoid the short circuit of the input voltage. This Block represents the last stage to generate the PWM based on 2D addressing mode for the four level inverter power switches control [16].

The only input of the FPGA application is the onboard 50MHz clock signal from the FPGA circuit. Frequency divider with VHDL is used to synchronize the internal computations of the logic blocks [17].

Binary counter, Multiplexers, decoders and frequency divider are also programmed by using VHDL language. The frequency divider is programmed to generate control signal with f_{CLK_c}=6.5kHz,($f_{CLK\_C} = \frac{1}{T_{CLK\_C}}$) as a commutation cells switching frequency.

Simulation results

Based on Model Sim program, we present in figure 7, the simulation results of the PMPWM based on 2D addressing mode to control the six commutations cells of the single phase four-level inverter.

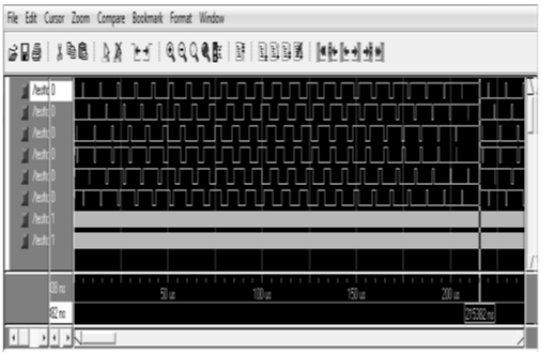

Fig.7: Six control signals generated by PMPWM based on 2D mode

Simulation results show that this algorithm generates all the necessary control signals for the four-level inverter commutation cells. Complementary commutation cells are controlled by complementary signals and dead time is taking into account in the complementary control signals

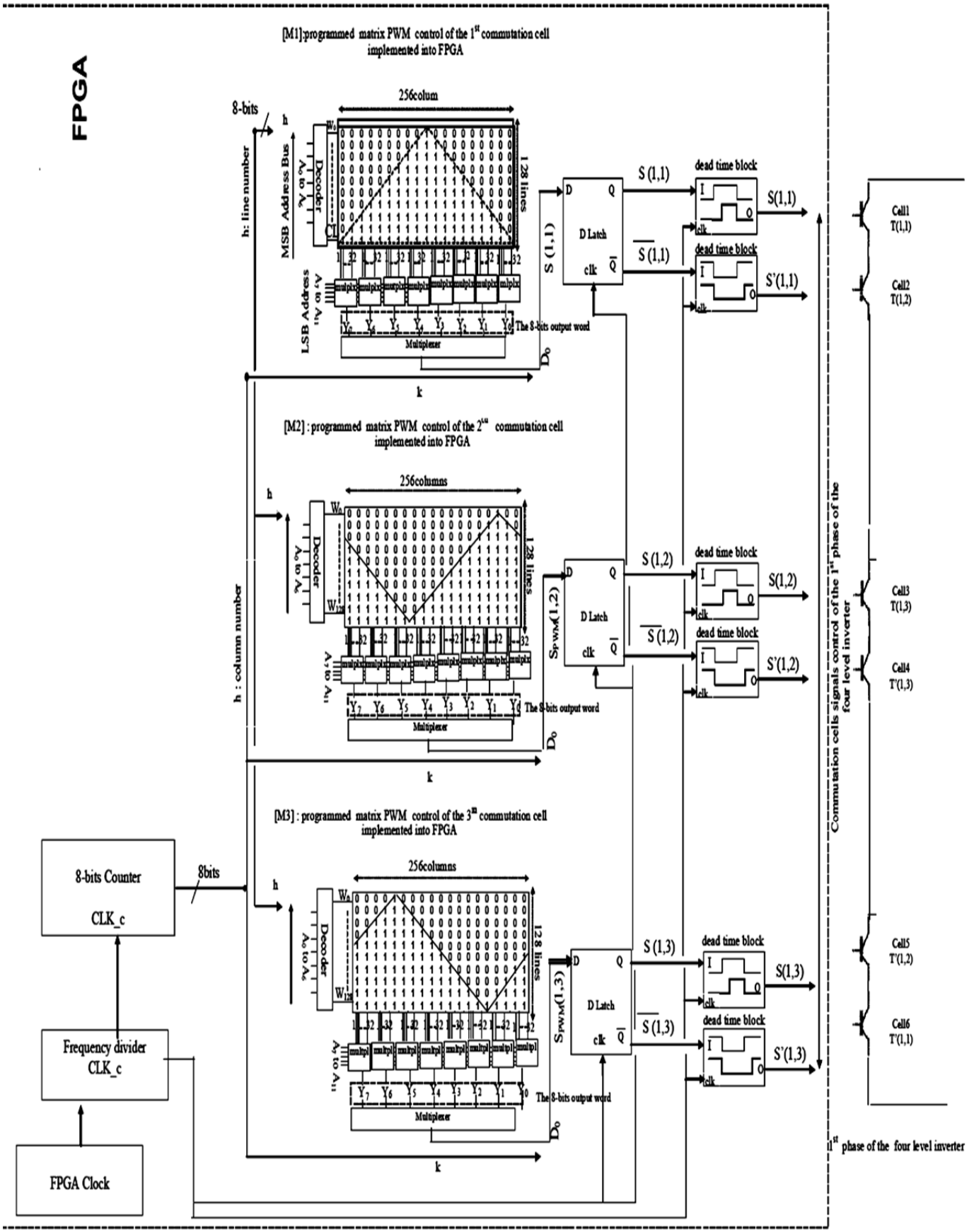

Fig.6. Synoptic of the FPGA-based 2D addressing mode in PMPWM for 4 levels inverter control

Hardware realization and experimental results

The four- level inverter prototype has been realized by using the 800V, 7.8A MOSFET (IRF750) as switching devices. DC link voltage with middle point:E/2=60V, flying capacitors voltages values: Vc(1,1) =40V and Vc(1,2)=80V, output voltage frequency : f=50Hz, 8-binary counter frequency: f_{CLK_C}= 6.5KHz, load resistance : R= 10hm, load inductance: L=0.5mH.

The digital implementation of the control strategy and notably the pulse width modulation is done on a development card around the SPARTAN III XC3S200 FPGA of Xilinx, including a 50MHz oscillator.

The Spartan III card delivers 18 control pulses (6 by phase), sent to the power switch. The control system (FPGA control card) is connected to the power devices via an interfacing card including galvanic isolation (optocouplers) and MOSFET(s) drivers.

Figure 8, given below shows experimental control signals generated by the FPGA based on 2D addressing mode

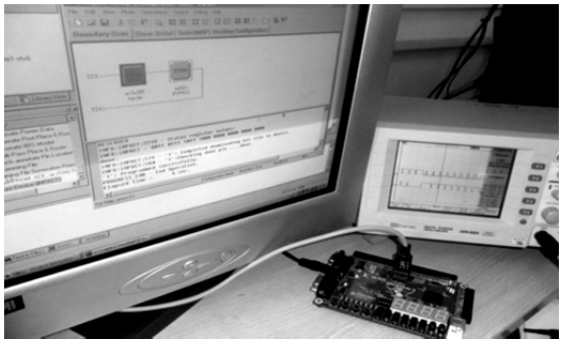

Fig.8. Photograph of the control signals generated by the FPGA based on PMPWM using 2D addressing mode

Figure 9, figure 10 and figure 11 presented below show the experimental control signals generated based on FPGA

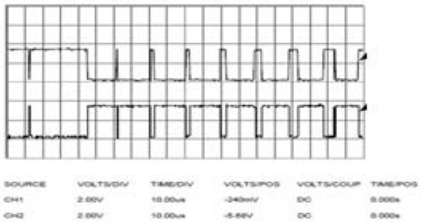

Figure.9. control signals (S(1,1) and S'(1,1)) of the complementary commutation cells (T(1,1)and T'(1,1))

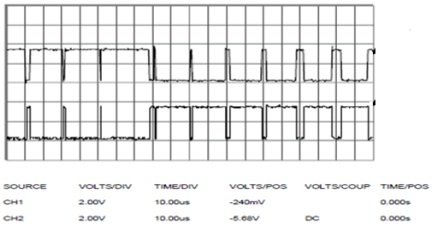

Figure.10. control signals (S(1,2) and S'(1,2)) of the complementary commutation cells (T(1,2) and T'(1,2))

According to figure 9, figure 10 and figure11, we can see clearly that complementary MOSFET(s) are controlled by complementary control signals and dead time is taking into account in experimental results.

Figure 12 shows the realized experimental set-up to test the programmed matrix PWM based on 2D addressing mode, implemented into FPGA to control the four levels inverter.

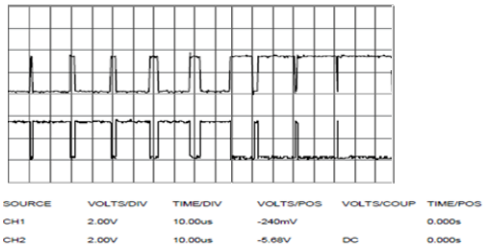

Figure.11. control signals (S(1,3) and S'(1,3)) of the complementary commutation cells (T(1,3) and T'(1,3))

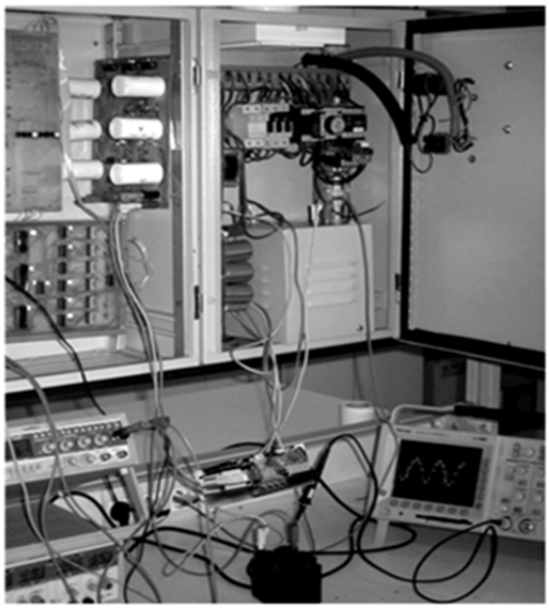

Fig.12. Photograph of the Experimental Prototype of the FPGA based on 2D addressing mode to control the 4 level inverter

Figure 13 shows the experimental flying capacitor voltages Vc(1,1) and Vc(1,2). It is clear that these two flying capacitors voltages are initially uncharged at the start up and they naturally evolve to their target operating voltages. These results confirm that the programmed matrix PWM modulator based on 2D addressing mode guarantees the natural balancing property of this inverter.

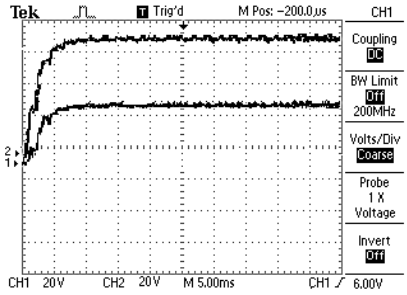

Fig.13. Experimental Flying Capacitor Voltages Vc(1,1) and Vc(1,2)

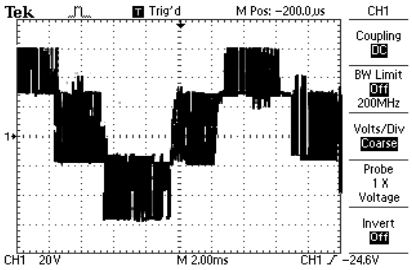

Fig.14.Experimental result of the per phase out put voltage waveform of the four-level inverter

The experimental result of the four-level inverter line to midpoint DC bus output voltage is given by figure 14. We can clearly see that the output voltage is formed by N=p+1=4 distinct voltage levels. In fact the experimental results confirm the high performances of the embedded programmed PWM to control of the 4 levels inverter.

Conclusion

The design of a high performance pulse width modulator is discussed. A prototype is built and tested. Simulation and experimental results are provided. Major features of the proposed configuration are summarized as fallows:

FPGA-based programmed matrix PWM to control multilevel inverter is able to alleviate the burden of the Analogue one which is often used to execute the algorithm of great complexity.

Basing on the concurrent behaviours of the programmable FPGA, high-speed input-output response enables a precise switching to reduce the harmonic in the multilevel output voltage content.

FPGA-based 2D addressing mode in PMPWM, minimize the number of the logical gates and it occupies only 4% of the available FPGA ROM memory

The proposed PMPWM controls a single phase four level inverter, but its application can be extended to the multiphase case and the FPGA XC3S200 internal resources in terms of ROM memory are enough for the multiphase control

***Authors**: lazhar manai, Higher institute of information and communication technologies of Borj Cedria, Technopole borj cedria, 2050 Hammam-Lif – Tunis,Tunisia, E-mail: manaii_lazhar@yahoo.fr ; Mohamed Alaa Eddine Hasni, Laboratory of Robotics, Informatics and Complex Systems (RISC) ENIT, Tunis,Tunisia E-mail:* hasni.medalaaeddine@gmail.com; *faouzi Armi, Laboratory of Robotics, Informatics and Complex Systems (RISC) ENIT, Tunis,Tunisia, E-mail:* armifaouzi@gmail.com*; Mongi Besbs, Higher institute of information and communication technologies of Borj Cedria, Technopole borj cedria, 2050 Hammam-Lif – Tunis,Tunisia, E-mail: mongi.besbes@gmail.com*

REFERENCES

[1] Lopez O., Alvarez J.,Doval-Gandoy J.,Freijedo F.D.,Nogueiras A., Lago A., Penalver C.M., Comparison of the FPGA Implementation of Two Multilevel Space Vector PWM Algorithms, IEEE Transactions on industrial electronics, 55(2008),no. 4, 1537 - 1547

[2] Fan S., Zhang K., XiongJ., Xue Y., An Improved Control System for Modular Multilevel Converters with New Modulation Strategy and Voltage Balancing Control, Power Electronics, IEEE Transactions on , 30(2015), no 1, 358 – 371

[3] Charles I.O., Damian B.Ni.,Emeka S.O.,Three-phase,Five-level Multi-level Inverter Topology, Electric Power Components and Systems, 40(2012),no 13,1522-1532

[4] Rao S.N., Kumar D. V.A., Babu C.S.,New multilevel inverter topology with reduced number of switches using advanced modulation strategies, IEEE Conference Publications, (2013), 693 - 699

[5] Knaub V., Model based code generating of VHDL code for DSP applications, Diploma Thesis, Cologne University of Applied Sciences (2010).

[6] Dubey R., P. Agarwal, and M. K. Vasantha, "Programmable logic devices for motion control, IEEE Trans. Ind. Electron., 54(2007), no. 1, 559–566,

[7] Koutroulis E., Dollas A.,Development of an FPGA-Based SPWM Generator for High Switching Frequency DC/AC Inverters, Power Electronics, IEEE Transactions, 29(2014), no1, 356 – 365

[8] Dariusz S.,Przemysław C., Artur M., Rafał M., Marian P. K., DSP - FPGA Based Computing Platform for Control of Power Electronic Converters, Przegląd Elektrotechniczny, 91(2015), no 12, 1-6

[9] Aguirre M.P., Calvino L.,Valla M.I.,Multilevel Current-Source Inverter With FPGA Control, Industrial Electronics, IEEE Transactions on ,60(2012), 13-10

[10] Chandrasekaran S., Rama M., FPGA Based Implementation of Amplitude Modulated Triangular Carrier PWM, Przegląd elektrotechniczny,87 (2013) no.7, 41-43

[11] Tamer H. A., Khaled M.N.,Optimal PWM Control of a New Generalized Family of Multilevel Inverters, Electric Power Components and System, 36(2007), 73-92

[12] Amara A., Amiel F., Ea T., FPGA VS. ASIC for low power application, Microelectronics, Journal 37(2006) 669-677

[13] Monmasson E, Cirstea M. N.,FPGA Design Methodology for Industrial Control Systems-A Review, IEEE Trans. Ind. Electron., 54(2007) no. 4,1824-1842

[14] Spataru L., Floricau. D.,Ionescu. F., FPGA control for three-level static power converters, U.P.B. Sci. Bull., Series C, 73(2011) no 4, 66-72

[15] Georgi K., Stamatis V., and Jos.T.J.E., A 2D addressing mode For multimedia Applications, Embedded Processor Design Challenges 2268(2002), 291-306

[16] Karuppanan P., Mahapatra K.K.,FPGA based cascaded multilevel pulse width modulation for single phase inverter, Environment and Electrical Engineering (EEEIC), (2010) 9th International Conference Publication, (2010), 273 - 276

[17] Jens O. K., Markus H., Andreas R., Rolf R., FPGA-based Control of Three-Level Inverters, PCIM Europe, Nuremberg, Germany (2011), 394-400

[18]