Xilinx implementation of a serial-parallel digital converter

Xilinx implementation of a serial-parallel digital converter

C D Cunțan, I Baciu and C M Diniș

Politehnica University of Timisoara, Department of Electrical Engineering and Industrial Informatics, 5 Revolution Street, Hunedoara, 331128, Romania

E-mail: corina.cuntan@upt.ro

Abstract. In the paper it is presented a serial parallel converter on 8 bit using a movement register with parallel output. The scheme created in Xilinx and implemented on a development board Nexis4DDr. The scheme contains two 4 bit counters and two registers schematically made and a frequency divider created in VHDL. The aimed sequence tact is obtained by dividing the tact frequency of 100MHz from the development board with the help of a divider circuit that allows the soft setup of the necessary frequency. The entering signal is brought to the serial input of a movement register, starting with the less significant bit. From the movement register the byte is transferred to the output register. The transfer between the two blocks is made in the moment when the 4 bits counter reaches 7 (moment which represents the closure of taking a bite) sequence which commands through a gate AND-NOT with three entrances, the command signal of the output data transfer. The output sequence can be memorized through a memorizing circuit or can be used for the command of a physical circuit. The H→L transition of the synchronizing signal, signal which is obtained also from the tact signal with the help of another 4 bit counter, determines the beginning of the conversion process. The circuit can function in a continuous way (without any reset circuit) or depending on the application it can be connected to a circuit which provides a command signal to obtain the reset signal of the implemented scheme.

1. Introduction

The most widespread shift registers application is the data conversion from parallel format in serial format and also the restoration through conversion of series data in parallel format to process or display them. A shift register is a register of n bits with the possibility of moving the stored data in it with one position of 1 bit at each clock impulse.[1] The series parallel convertor was designed using the developing software package Xilinx ISE, it's implementation being made on the reprogrammable structure FPGA of Nexis4DDR type.[2]

The probable logic circuits FPGA are probable logic gate arrays in the electric field. The circuit contains a big number of probable logic blocks (CLB) arranged in a probable connection web, surrounded by I/O blocks which are also probable.

With these types of circuits logic functions of great complexity can be obtained using both electronic schemes and codes which are written in hardware description language VHDL, Verilog or ABEL.

The software is essential for the efficient usage of the probable logic devices. The series of programs Integrated Software Environment of the Xilinx firm represents an integrated development environment for the designing of the digital systems with the help of FPGA.

The usage of programmable circuits brings a series of advantages in comparison to the usage of classic integrated circuits, among which a few are reminded:

- Reducing the size and power dissipation;
- Improving speed performance;
- Ease of debugging faulty systems;
- Decreasing the failure rate and the price of the printed circuit;[3]
- Updating and improving the implemented function by reprogramming, without any hardware change.

2. Problem Formulation

In this paper it is presented a series - parallel convertor on 8 bits using a shift register with parallel output.

The series-parallel conversion implies a data transfer from the source to destination, transfer which means using three command signals (Figure 1):

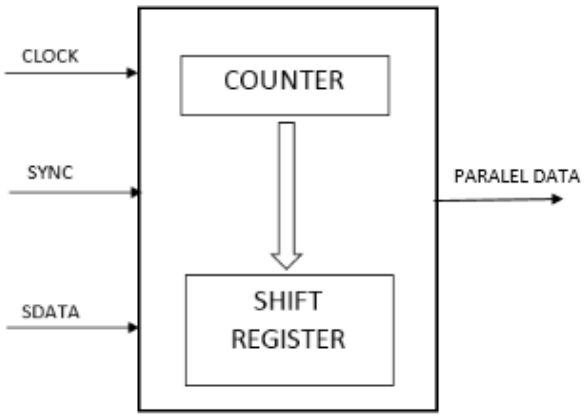

Figure 1. Block simplified scheme of the convertor

The clock signal (clock), the data series signal (SDATA) and the timing signal (SYNC). The clock signal gives the time reference needed for the transfer, defining the time period in which 1 bit is transferred. The desired frequency is obtained by dividing the frequency on the main circuit (100MHz) through a dividing circuit which allows the soft setting of the output frequency.

The data signal represents the input size for the serial - parallel convertor. (Figure 2)

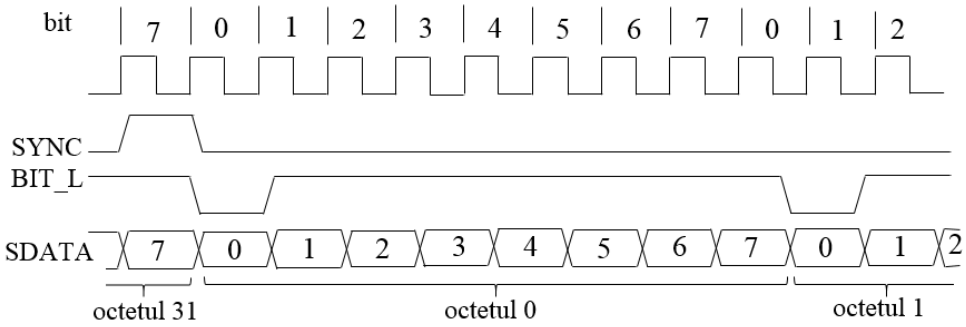

Figure 2. Temporal diagram for the parallel-series conversion

The beginning and ending of a byte is ordered by BIT_L which is obtained at the output of the 4 bits' number and which orders the data transfer from the shifting register to the parallel register. The data is transmitted starting with the most significant bit. The 32 temporal divisions are determined with the help of the synchronizing input.

The timing signal gives the time reference for defining the data format, for example the beginning of a byte or of a word from the series data row. [4]

The series - parallel conversion is realized through two shift registers. The data transfer between the registers is ordered by a counter of 4 bits which provides the transfer impulse through an identification circuit of number 7. [5] The convertor's structure contains a second counter which allows the determination of the temporal division number of the 8 bits section. The convertor's scheme realized in Xilinx is presented in Figure 3.[6]

The byte from the convertor's output is used for the transistors' command from the bridge inverter of a frequency convertor.[7]

To obtain a signal with an output frequency of 1KHz it is necessary a clock signal with a frequency of 8 KHz. This signal is obtained at the DIV1 divisor's output which divides the clock frequency of the Nexis 4 DDR module to 6250. The DIV1 divisor's output commands both the counters as well as the shift registers from the convertor's structure.

The data sequence is brought with a frequency of 4 KHz which is obtained at the DIV 2 divisor's output; this frequency results from the division of the clock frequency to 12500.

The timing signal is brought at the counter's input and determines the beginning of the counting process.

The command signal for data transfer between the registers is obtained by the first 4 bit counter. The least important 3 bits of the counter are brought at the entrance of a NAND gate with three inputs; its exit is brought through an inverter of the command pin of the data transfer in the output register.

The RCO output of the number commands the functioning validity of the next number. The most significant bit from the first counter together with the second one are used for the elaboration of the command sequence of the frequency phase convertor's transistors, as it can be observed in Figure 3.

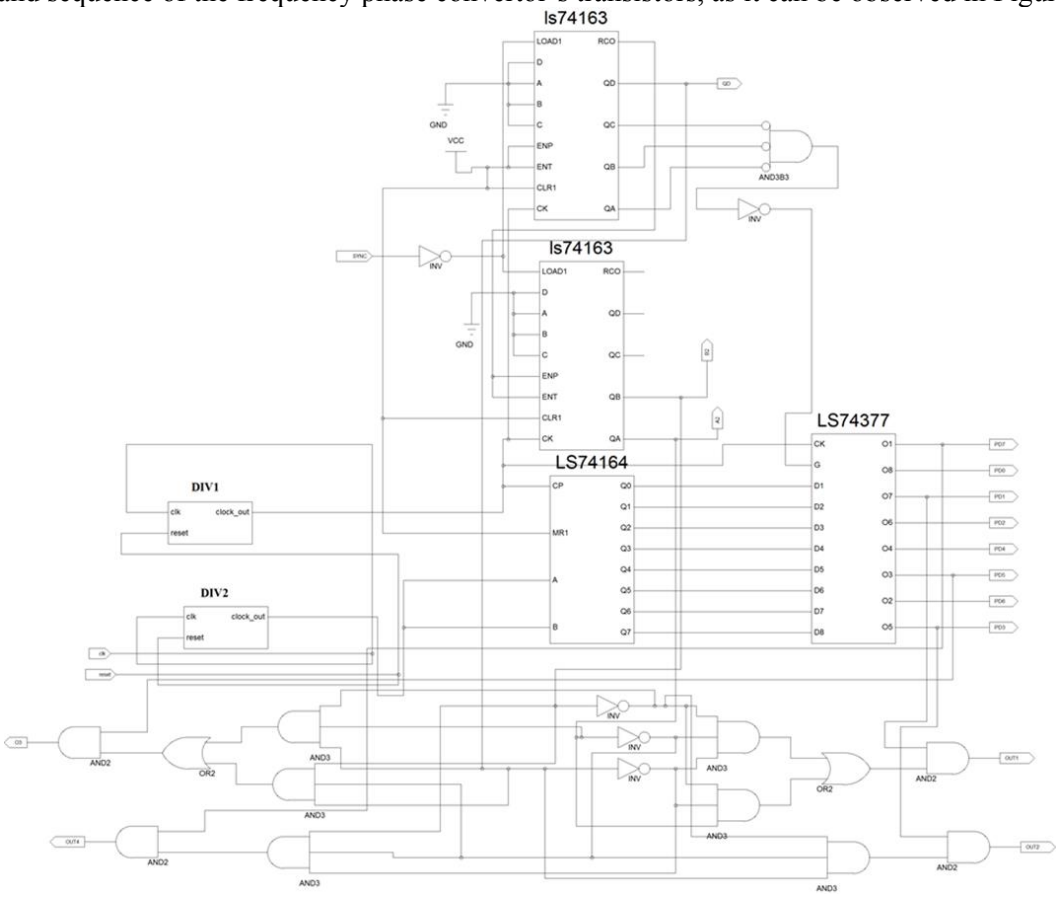

Figure 3. The series - parallel convertor's scheme realized in Xilinx

The structure of a counter (Figure 4) was realized under the form of a schematic file in Xilinx starting from the structure of a 74163 counter.

The parallel inputs are connected to the mass and the validation and reset inputs are connected to1 logic. The tact impulse for counting is received from the DIV1 divider with an output frequency of 8 KHz and the charging input is brought from the synchronizing signal through an inverting device. The outputs of the number allow obtaining the transfer command given in the parallel register and at the output.

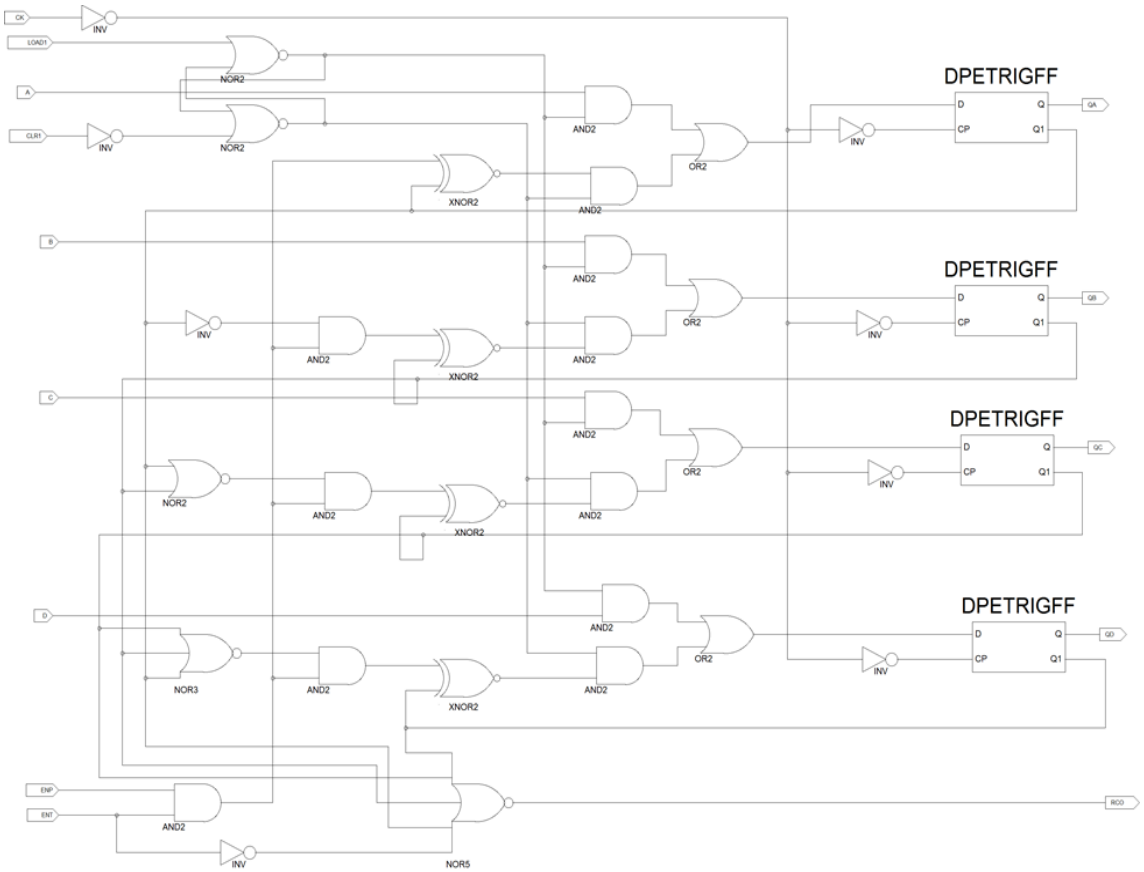

Figure 4. The structure of the counter designed in Xilinx

Also the structures of the shift register and of the parallel register were realized starting from the structure IC74164 (Figure 5) and 74377 (Figure 6).

The shifting register (Figure 5) contains 8 bi-stables of Delay type. The input data brought in a series are transferred in the parallel register at the end of 8 tact impulses. Once with the data transfer in the parallel register the serial acquisition of a new bite can begin.

The parallel register (Figure 6) has in its structure mainly 8 bi-stables of Delay type. This block receives the transfer command from BIT-L in the moment when the counter reaches number 7, transferring the output data in parallel format.

At each made block the independent functioning was checked using Nexis 4 DDR.

Finally the parallel series convertor scheme was implemented and it was verified it's functioning.

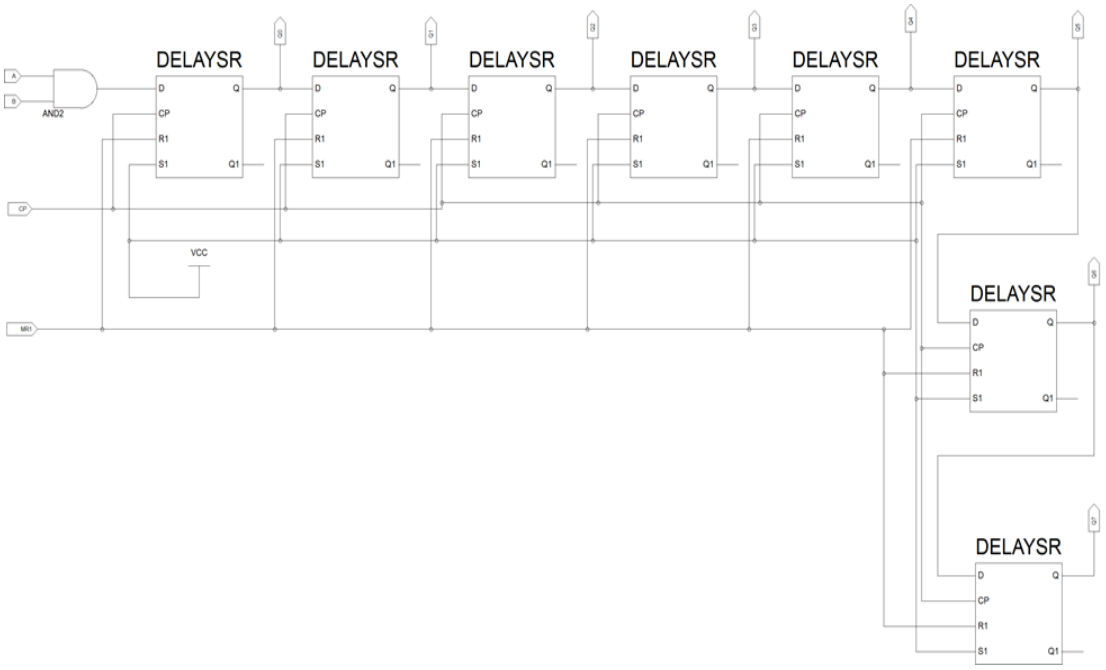

Figure 5. The structure of the shift register designed in Xilinx

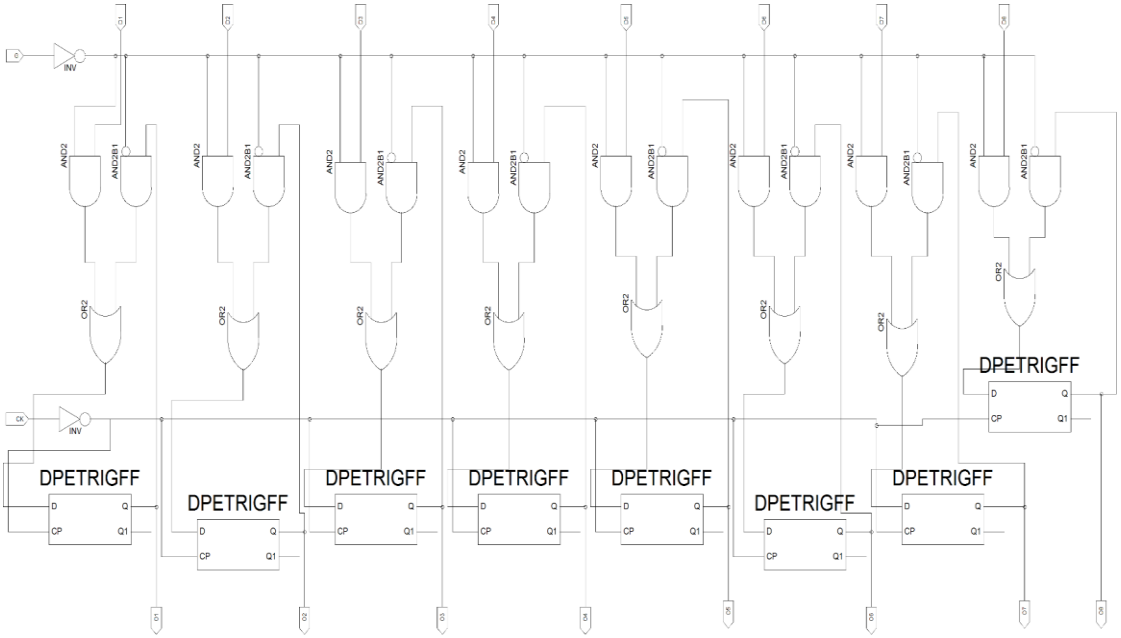

Figure 6. The structure of the parallel register designed in Xilinx

The output obtained signals go to the entrance of an phase inverter block with 4 IGBT transistors (Figure 7).[8]

The signals are brought at an optocoupler galvanic separation circuit through a MOS inverting. [9]

For the optocouplers command are used transistors (Ta1…Td1) to raise the required current for the input commands of the octocouplers. At the octocouplers' exit another group of transistors is used (Ta2, Ta3…Td2, Td3) for the command both in current as well as in the polarity of the IGBT

(T1…T4) transistors.[10] The IGBT transistors order the primary of a voltage elevated transistor resulting at the output an alternative voltage that has an alternation formed out of two voltage levels. To smooth the voltage and current jumps a L1C1 filter is used.[11]

The presentation of the functioning of the entire circuit is emphasized by the wave obtained in different points of the circuit.

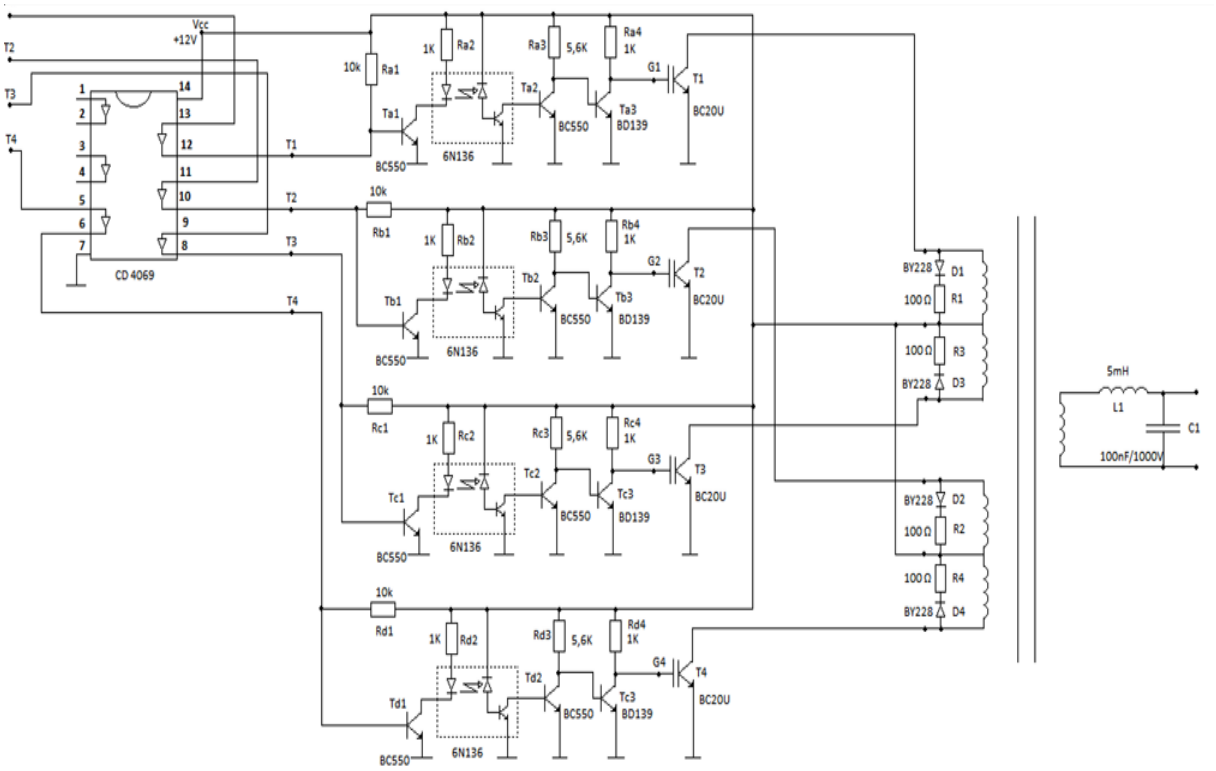

Figure 7. The single - phase inverter scheme

The waveforms were obtained with the help of a two-channel digital oscilloscope which allows the storage of the visualized signals. In this way the waveforms are presented for the command of the Ta1, Tb1 (Figure 8) and Ta1, Tc1(Figure 9).

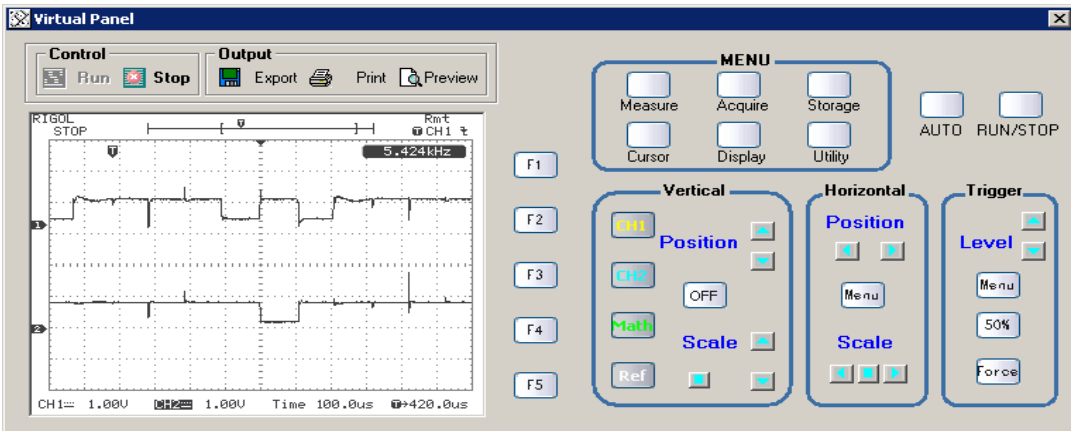

Figure 8. Command impulse for Ta1 and Tb1

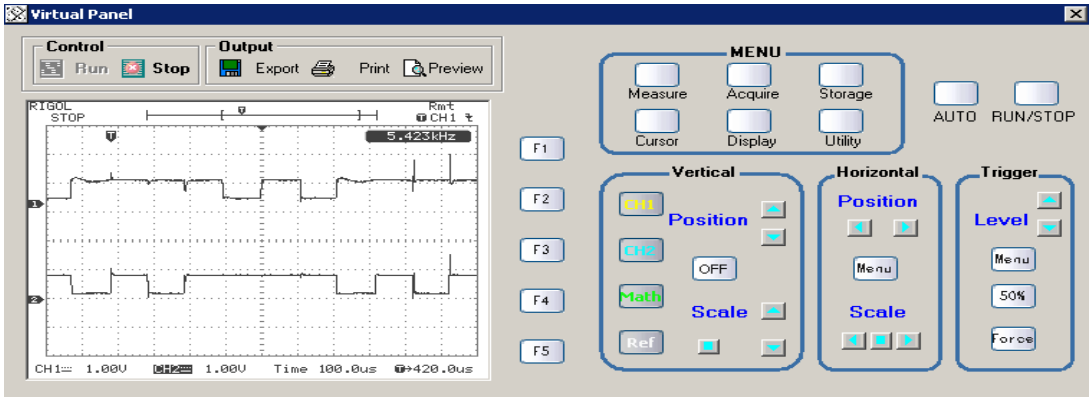

Figure 9. Command impulse for Ta1 and Tc1

These signals command the optocouplers' entrance as it follows: the optocoupler command 1,2 (Figure 10) and the optocoupler command 1,3 (Figure 11).

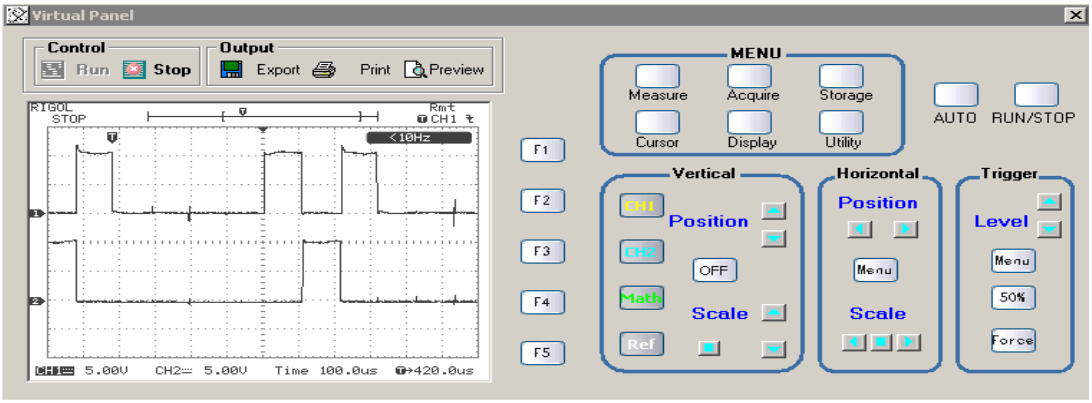

Figure 10. Command impulse for optocouplers 1 and 2

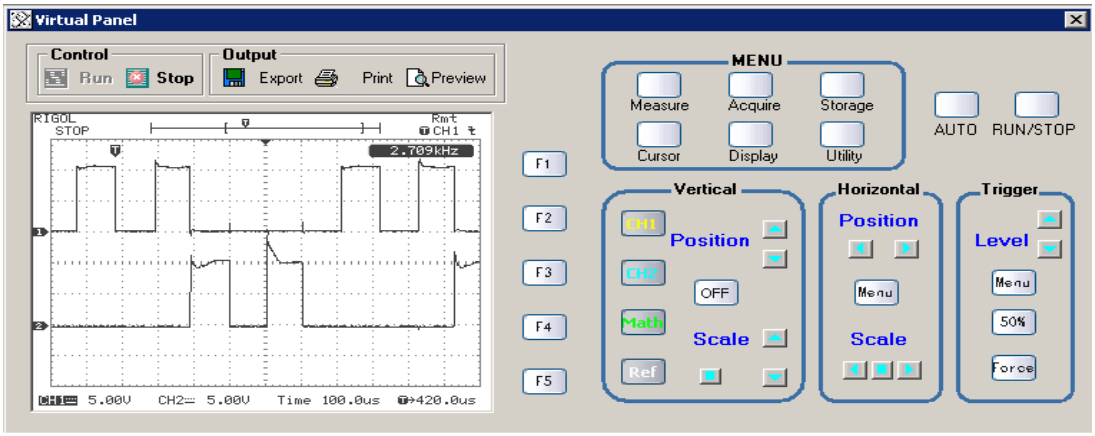

Figure 11. Command impulse for optocouplers 1 and 3

The IGBT transistor's command signals are presented in Figure 11 for T1 and T3 transistors (Figure 12) and for T1 and T2 transistors in Figure 13.

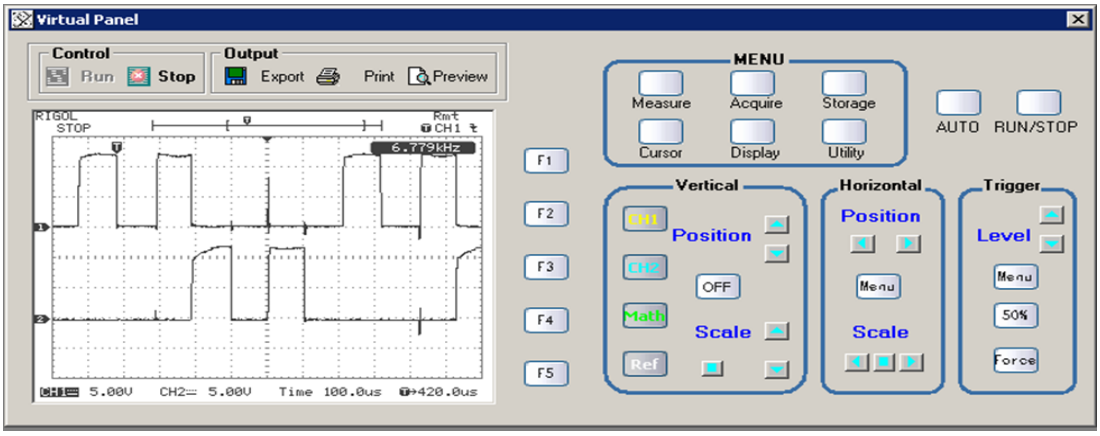

Figure 12. Command voltage for the transistors' command IGBT: T1 and T3

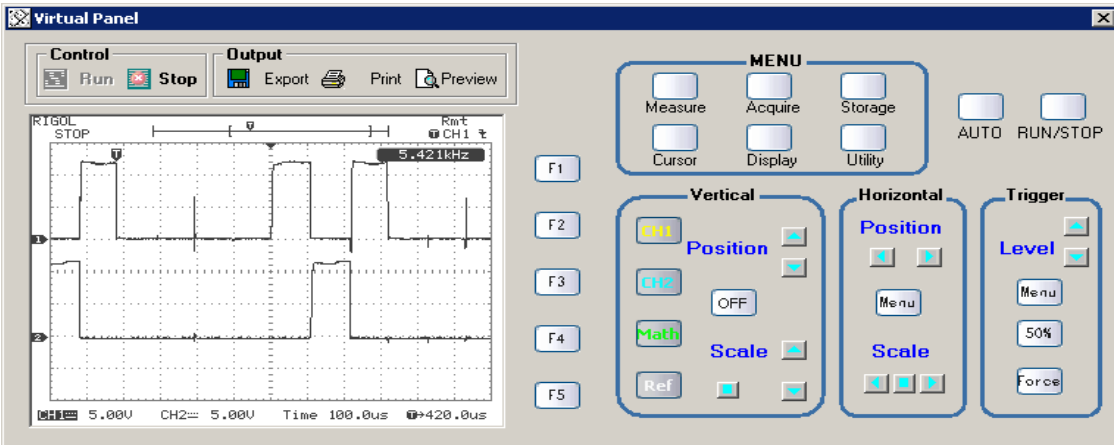

Figure 13. Command voltage for the transistors' command IGBT: T1 and T2

The output voltage (Figure 14) emphasizes the existence of three intervals on each alternation. From the figure it results that the voltage jumps corresponding to levels 1 and 2 are not equal. This fact is due to the transformer's construction.

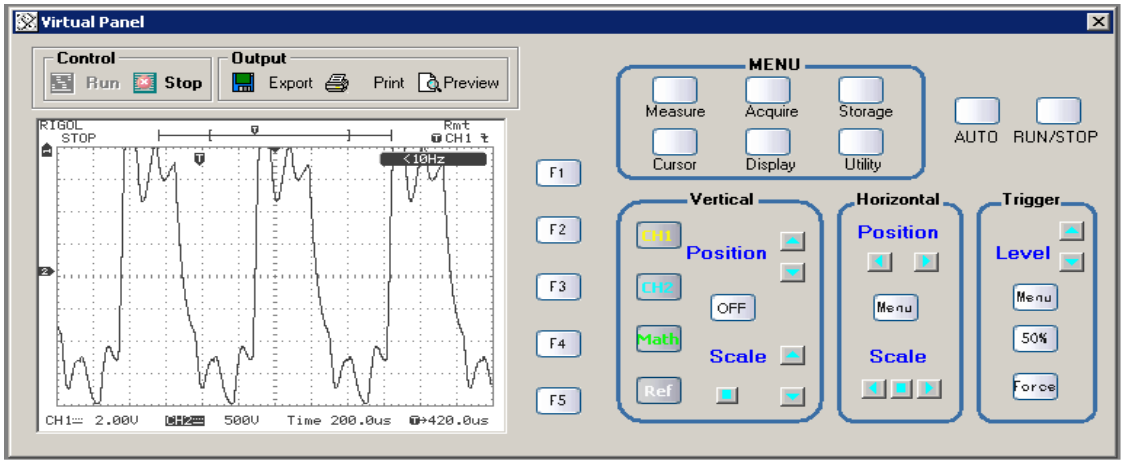

Figure 14. Output voltage

3. Conclusions

The presented waveforms correspond to a clock frequency of 1KHz which corresponds to an output frequency of 125 Hz. Using the output filter allows smoothing the jumps which reduces the number of harmonics in the output voltage.

The final IGBT transistors are protected at high-voltage by connecting an RD group on each primary winding of the transistor. Improving the functioning of the inverter can be done by using a command sequence with pauses between the transistors' commands. The usage of the parallel series convertor allows setting with ease the output frequency by changing the clock impulse. This is realized by soft modifying of the frequency divider without involving other additional circuits.

Also the output frequency can be memorized in a memory cell which can be later used independently for the inverter command.

Implementing the Xilinx circuit allows to easy change the output sequence so that other physical circuits can be commanded.

At the exist of the register from the parallel series convertor's structure it is ordered the passage on 0 in the short time of its exit. This can be realized through a logical scheme which is commanded by another frequency divider.

References

[1] Wakerly J 2002 *Digital Circuits. Principles and practices used in design*, Publishing House Teora, Bucureşti, Romania

[2] Iana G, Şerban G, Ionescu L, Anghelescu P, Popescu M, Bitoleanu A and Dobriceanu M 2010 *Programming with Hardware Description Languages*, Publishing House MatrixRom, Bucuresti, Romania

[3] Nicula D 2000 *Design of digital systems implemented with programmable devices*, Technical Publishing House, Bucuresti, Romania

[4] Büchele M, Fischer H, Gorzellik M, Herrmann F, Königsmann K, Schill C and Schopferer S 2012 128-channel Time-to-Digital Converter (TDC) inside a Virtex-5 FPGA on the GANDALF module, *Journal of Instrumentation* **7** 1-7

[5] Li Y, Huang A Q and Motto K 2002 Series and Parallel Operation of the Emitter Turn-Off (ETO) Thyristor, *IEEE Transactions On Industry Applications* **38**(3) 706-712

[6] Sugimura H, Eid A M, Kown S K, Lee H W, Hiraki E and Nakaoka M 2005 *High frequency cyclo-converter using one-chip reverse blocking IGBT based bidirectional power switches*, Proceedings of the Eighth International Conference on Electrical Machines and Systems, September, Volume 2, pp 1095-1100

[7] Monmasson E and Cirstea M N 2007 FPGA Design Methodology for Industrial Control Systems—A Review, *IEEE Transactions On Industrial Electronics* **54**(4) 1824-1842

[8] Luo D F L, Hong Y and Rashid M H 2005 *Digital Power Electronics and Applications*, Elsevier Academic Press

[9] Sathyan A, Milivojevic N, Lee Y J, Krishnamurthy M and Emadi A 2009 An FPGA-Based Novel Digital PWM Control Scheme for BLDC Motor Drives, *IEEE Transactions on Industrial Electronics* **56**(8) 3040-3049

[10] Popa G N, Popa I, Diniș C M, Deaconu S I and Abrudean I C 2016 *Control of a single-phase inverter with PIC microcontroller*, The XVIth Multidisciplinary International Conference "Professor Dorin Pavel - Founder of Romanian Hydro-energetics" Science and Engineering, Vol 29**,** pp 447-456

[11] Deaconu S I, Babău R, Popa G N and Gherman P L 2018 Hydroelectric power plant with variable flow on drinking water adduction, *IOP Conf. Ser.: Mater. Sci. Eng.* **294** 012023