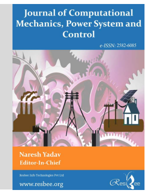
Vol.8 No.1 Jan 2025

Journal of Computational Mechanics, Power System and Control

Received 2 December, Revised 10 December, Accepted 3 January



Performance Comparison of FPGA-Based Controllers for Single Phase Square Wave Inverter

C. Naga Kota Reddy
Department of EEE, UCEK (A),
Jawaharlal Nehru Technological University,
Kakinada, 533003, India.

K. Sandhya Rani
Department of EEE,
Aditya University,
Suralmpalem, 533437, India.

N. Sagar Teja Yadav
Department of EEE, UCEK (A),
Jawaharlal Nehru Technological University,
Kakinada, 533003, India.

Abstract: This paper presents the design and implementation of switching pulse generation for a single-phase square wave inverter using two Field-Programmable Gate Array (FPGA) controllers: the Artix-7-based Basys3 and Zynq-based Wavect controller. The study evaluates the effectiveness of these FPGA platforms in managing the inverter's output voltage and waveform quality. The switching pulses for the inverter are generated by the two FPGA boards, and the effectiveness of each controller is evaluated in terms of implementation complexity and the resulting Total Harmonic Distortion (THD) of the output waveform. The performance comparison highlights the advantages and limitations of each FPGA platform, with the Basys3 controller offering a simple, cost-effective solution for basic switching pulse generation, while the Zynq-based Wavect controller provides enhanced processing capabilities and flexibility for more advanced applications. The focus is on generation switching pulses for a single-phase square wave inverter for driving an RL load with parameters 220 Ω and 150 mH. The results offer valuable insights into the use of FPGA-based controllers for high-performance inverter systems, particularly in applications involving renewable energy and power electronics, where cost-effectiveness, reliability, and efficient waveform control are essential.

Keywords: Single Phase Square Wave Inverter, Artix-7 Board, Basys3 FPGA Controller, Zynq Board, Wavect Controller, THD.

Nomenclature

Abbreviation	Expansion
UPS	Uninterruptible Power Supplies
DSO	Digital Storage Oscilloscope

1. Introduction

Inverters are fundamental components in power electronic systems, responsible for converting DC power into AC power for a wide range of applications, including renewable energy systems, UPS, motor drives, and more. Among the different types of inverters, square wave inverters are simple and cost-effective solutions for low-power applications where the quality of the AC output is not critical. However, the primary challenge with square wave inverters is the high harmonic distortion inherent in their output, which can affect the performance of the connected load and lead to inefficiency in systems where high-quality power is required. To address this, precise control of the switching devices is crucial. Traditionally, microcontrollers have been used to generate the switching signals for such inverters. However, microcontrollers are limited by their processing power and real-time control capabilities, particularly when dealing with high-frequency switching or complex load conditions. In power electronics, both digital and analog controllers are widely used to generate control signals (such as PWM pulses) for driving power devices like MOSFETs, IGBTs, or thyristors. Each type of controller has its unique strengths and weaknesses, making them more suitable for different applications depending on factors like complexity, flexibility, cost, precision, and scalability [1].

Digital controllers are characterized by high precision, flexibility, and the ability to implement complex control algorithms. They leverage digital processing and sampling, which minimizes errors and allows for easy integration with other digital systems like sensors or communication interfaces. However, they require higher initial costs due to the complexity of microcontrollers, DSPs, or FPGAs, and their development process is longer due to the need for coding, software design, and debugging. Additionally, digital systems generally have slower response times due to computational delays but can achieve high-

https://doi.org/10.46253/jcmps.v8i1.a4

speed performance with specialized processors. Maintenance and upgrades are easier since they can be handled through software updates [2-3].

In contrast, analog controllers are simpler and faster, with lower costs and shorter development times. They operate in real-time, leading to quicker response times. However, they are limited by the precision of passive components, and analog drift can affect accuracy over time. While analog systems are more robust to faults like power surges and noise, they are less flexible, harder to upgrade, and challenging to scale or integrate with other digital systems. Analog controllers are most suitable for basic control tasks where speed and cost are critical factors. Ultimately, the choice between digital and analog controllers depends on factors such as application complexity, cost, speed, and scalability [4].

In contrast, FPGAs have emerged as a powerful alternative due to their high speed, parallel processing capabilities, and flexible configuration. FPGAs can generate high-frequency switching signals with minimal latency, offering a significant advantage in controlling inverters with high precision. The use of FPGAs in power electronics has gained widespread attention in recent years due to the flexibility, parallelism, and real-time control capabilities they offer. FPGA-based systems can efficiently manage high-speed switching, complex modulation schemes, and precise control of power converters, making them ideal candidates for applications such as inverters, motor drives, and renewable energy systems. This section reviews key studies on FPGA-based inverters, focusing on their design and implementation for single-phase inverters, with an emphasis on switching pulse generation, harmonic distortion reduction, and performance comparisons between different FPGA platforms. Inverter designs using FPGAs have been extensively studied for their ability to generate high-quality switching signals with minimal delay. One of the most common types of inverters is the square wave inverter, which is relatively simple but typically produces high harmonic distortion due to the abrupt switching transitions. The work by [5] demonstrated the implementation of an FPGA-based three-phase inverter with PWM control, where the authors showed a significant reduction in THD compared to traditional microcontroller-based designs. The use of FPGA enabled faster response times and better control over the switching frequency, leading to improved output quality. In the case of single-phase square wave inverters, the design challenge is simpler, but the harmonic distortion remains a concern. Several studies have explored the use of FPGA technology to optimize switching control. For example, [6] proposed an FPGA-based square wave inverter design and focused on minimizing the harmonic content in the output waveform through efficient switching techniques. The research highlighted the potential of FPGAs to improve the quality of the inverter output by adjusting the switching scheme in real time and dynamically controlling the transitions between the high and low states. FPGAs come in various families with differing capabilities in terms of logic elements, processing power, and ease of use. The Xilinx Artix series and Zynq series are two popular FPGA platforms commonly used for inverter and power electronics applications.

The Artix FPGA family, especially the Basys3 development board, has been widely adopted in both academic and industrial applications for simple inverter designs. The Basys3 board is known for its low cost, moderate logic density, and ease of integration with external peripherals, making it ideal for educational purposes and basic power electronics projects. Research by [7] explored the implementation of a square wave inverter using the Basys3 board, emphasizing its suitability for educational demonstrations and prototypes. Although Artix FPGAs have limitations in processing power compared to higher-end models, they are capable of meeting the requirements of basic inverters with moderate switching frequencies and less complex control strategies.

The Zynq family, which combines a dual-core ARM processor with programmable logic (FPGA fabric), offers greater computational power and flexibility, making it ideal for more complex inverter designs and applications where real-time processing and higher switching frequencies are required. Zynq-based systems, such as the Wavect controller, have been used

In advanced power conversion systems, including grid-connected solar inverters, as demonstrated in [8-9]. The ability of Zynq FPGAs to implement both control algorithms and signal processing tasks in parallel allows for more sophisticated inverter designs, such as those employing advanced modulation schemes or adaptive switching techniques. Additionally, the integration of ARM processors in the Zynq platform allows for easier system integration and communication with external devices, which can be advantageous in multi-level or multi-phase inverter designs.

Although FPGA-based inverter designs have been studied extensively, there is a lack of direct comparative studies that evaluate the performance of different FPGA platforms for square wave inverter applications. The comparison between Artix and Zynq FPGA controllers has been explored in some studies related to other power conversion systems, but not specifically for square wave inverters. In [10], a study compared the Basys3 and Zynq platforms for motor drive applications, highlighting the trade-offs between cost, complexity, and performance. The authors found that while the Basys3 board was suitable for simple applications, the Zynq platform provided better processing power and flexibility for more complex systems.

This paper fills the gap by providing a detailed comparison of the Basys3 and Wavect controllers for the specific case of a single-phase square wave inverter. The comparison focuses on implementation complexity, switching pulse generation, and THD performance, providing valuable insights into the selection of FPGA platforms for inverter applications.

2. Implementation Procedure in Basys3 Controller

The Basys 3 FPGA Board, designed by Digilent, is an affordable and beginner-friendly development platform featuring a Xilinx Artix-7 FPGA (XC7A35T) with 33,280 logic cells. It includes 16 switches, 16 LEDs, 4 push buttons, 4 analog inputs, 4 7-segment displays, and

2 PMOD connectors for expansion, making it ideal for learning digital circuit design, embedded systems, and FPGA programming. The photograph of the BASYS 3 FPGA Board is depicted in Fig.1. The board supports programming via Xilinx Vivado and Digilent Adept using a micro-USB port for both power and programming. While its resources are limited compared to higher-end FPGAs, the Basys 3 offers a comprehensive set of tools, documentation, and example projects, making it a great choice for education, signal processing, and small-scale hardware prototyping. However, it lacks an embedded ARM processor, limiting its ability for complex embedded software applications [11]. The BASYS 3 board includes a single 100 MHz oscillator connected to pin W5.

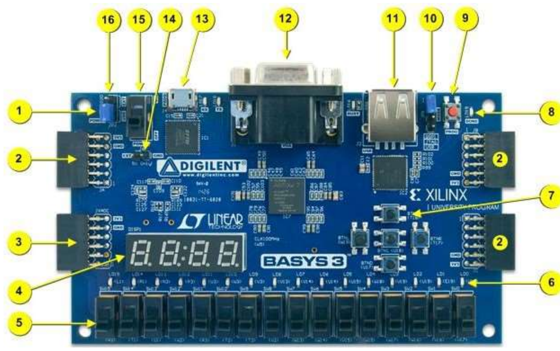


Fig.1. *Photograph of BASYS 3 FPGA Board*

1. Power goof LED, 2.PMOD connectors, 3.XADC,4.7-segment display,5.Slide switches,6.LEDs,7. Pushbuttons,8.Done LED,9.reset,10.jumper,11.USD connector,12.VGA connector 13.JTAG USB connector,14.External power connector,15. Power switch, 16→Power select jumper.

The MATLAB software also provides a tool for FPGA using the Xilinx System Generator. To Open MATLAB, click on Xilinx System Generator. A new MATLAB window will open. In the MATLAB window, click on Simulink in the toolbar. New windows will pop up. In the new window, click on the blank model. A new blank window will open. In the toolbar, click on the Simulink library browser. In the library browser, select Xilinx block set then system generator. Copy the System Generator and then paste it on the Simulink window. Then click on System Generator and select the following parameters shown in Fig. 2.

The Simulink model for a generation of switching pulses is designed in the MATLAB/Simulink environment with Xilinx Blockset. The switching pulses are generated by comparing a sawtooth waveform with a constant value. The counter is used to generate the sawtooth waveform. The frequency of the sawtooth waveform dictates the frequency of the switching pulse. The aim is to generate a 50 Hz signal from the Basys 3 Artix-7 FPGA Controller Board having a 100 MHz clock frequency. First, calculate the value of the period register for the desired frequency.

$$Periodregistervalue = \frac{Clockfrequency}{Desiredfrequency} \tag{1}$$

For generation of 50 Hz signal,

$$Periodregistervalue = \frac{Clockfrequency}{Desiredfrequency} = \frac{100X10^6}{50} = 200000 \tag{2}$$

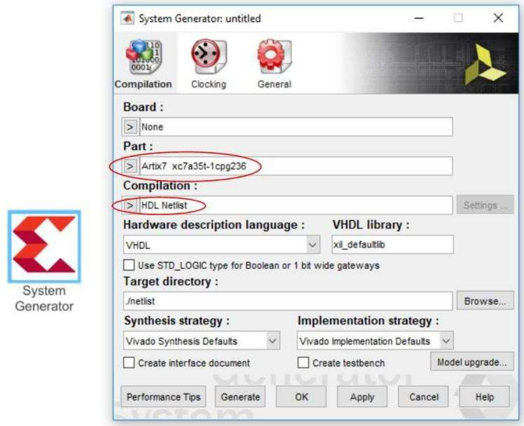


Fig. 2. *Parameters to be entered into the System Generator*

As most of the registers in FPGA are 16-bit, it is better to reduce the value of the period register less than 65536 (i.e., 216). The period register value is written as

$$Periodregistervalue = 20000X100 \tag{3}$$

A counter is used to generate 20000 samples (i.e., 0 to 19999) with an explicit period of 100. Select counter block for Simulink library browser-> Xilinx Blockset -> Basic Elements. The up counter with the required settings is shown in Fig. 3. For a generation of pulse for the switch use a constant (constant) block having a value of 10000 (i.e., 50% duty cycle) and shown in Fig.4. The complete Simulink model for a generation of switching pulses for single-phase square wave inverter with counter, constant, relational operator and logical operator is shown in Fig. 5. Now connect gateway out blocks at the output of each pulse. Enter the output port address and I/O standard in a single quote from the reference document for each gateway out of the block.

After completion of the pin allocation, open the System generator and click on generate to generate the HDL netlist file of the Simulink model. The Bitstream file of the above model is needed to dump into the Basys 3 Artix-7 FPGA Controller Board. The program of the controller board has four steps that are run synthesis, run implementation, generate Bitstream, and open hardware manager to program the device. The generated switching pulses can be viewed by connecting DSO at the PMODs. The voltage level of the pulses generated by the Basys 3 Artix-7 FPGA Controller Board is only 3.3 V. A level shifter should be used to drive the switches of the single-phase square wave inverter.

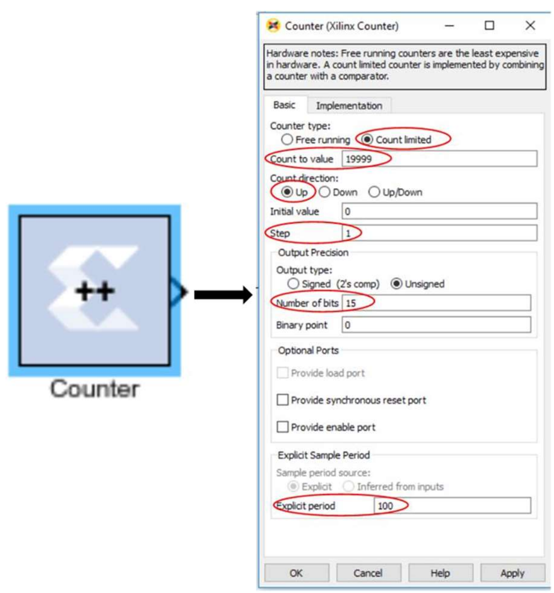


Fig. 3. *Counter with required settings*

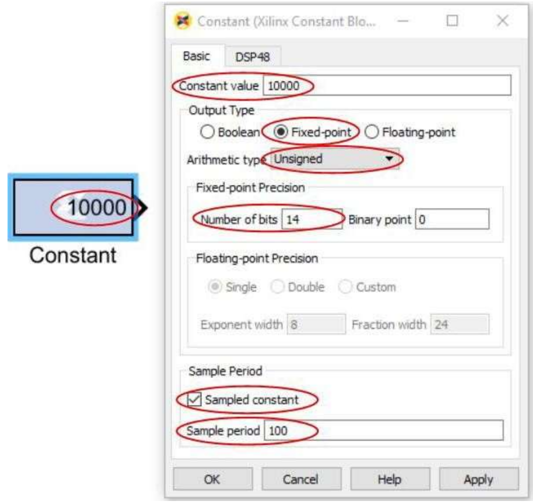


Fig. 4. *Constant block with the required settings and values*

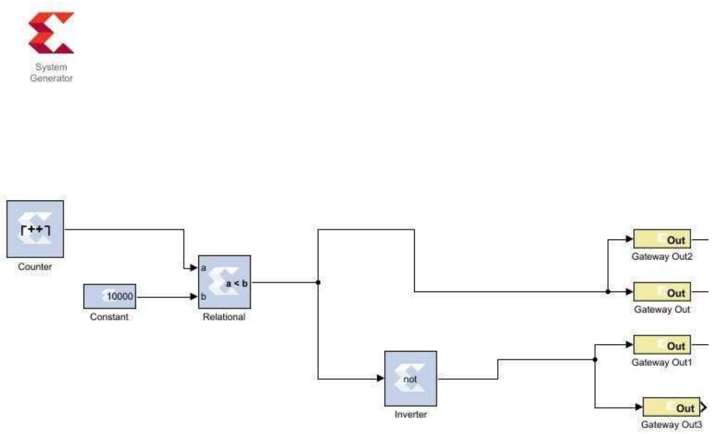


Fig.5. *Complete Simulink model for generation pulses for single-phase square wave inverter*

3. Implementation Procedure in Wavect Controller

The WAVECT Controller from Entuple Technologies is a highly advanced and versatile solution based on Xilinx Zynq-7000 XC7Z020-CLG484-1C/I. FPGA, dual-core processor, scalable voltage and current sensors, and high-speed I/Os. It enables efficient development of complex control algorithms using model-based programming and offers a push-button configuration process for quick deployment. The system's FPGA handles real-time processing tasks, while the ARM processor manages control and communication. Suitable for applications like motor control, power electronics, and signal processing, the WAVECT Controller is equipped with PWM outputs, encoder feedback, and dedicated instrumentation software for monitoring and tuning. It leverages Xilinx ISE and System Generator for easy integration and FPGA implementation, making it a powerful solution for rapid prototyping in industries like industrial automation, robotics, and embedded systems [12].

The switching pulses for single-phase square wave inverter are generated in MATLAB/Simulink environment using blocks in HDL coder. The aim is to generate a 50 Hz signal from the WAVECT Controller having a 40 MHz clock frequency. First, calculate the value of the period register for the desired frequency. For generation of 50 Hz signal,

$$Periodregistervalue = \frac{Clockfrequency}{Desiredfrequency} = \frac{40X10^6}{50} = 800000 \tag{4}$$

As most of the registers in FPGA are 16-bit, it is better to reduce the value of the period register less than 65536 (i.e., 216). For this initially, 18 kHz is generated and then a 50 Hz signal is generated. The counter value for the 18 kHz signal is calculated as

$$= \frac{40X10^6}{18X10^3} = 2222 \tag{5}$$

The counter value for 50 Hz from an 18 kHz signal is calculated as

$$= \frac{18X10^3}{50} = 360 \quad (6)$$

The simulink diagram for the generation of sawtooth waveform using HDL counters is depicted in Fig.6 and the counters settings are shown in Fig.7. The main simulink and subsystem simulink diagrams are depicted in Fig.8 and Fig.9 respectively. It shows the voltage and current sensors for measurement. When the model is ready and the simulation results are satisfactory, we will generate the HDL Code using the HDL coder workflow. Right-click on Subsystem then HDL Code and then HDL Workflow Advisor.

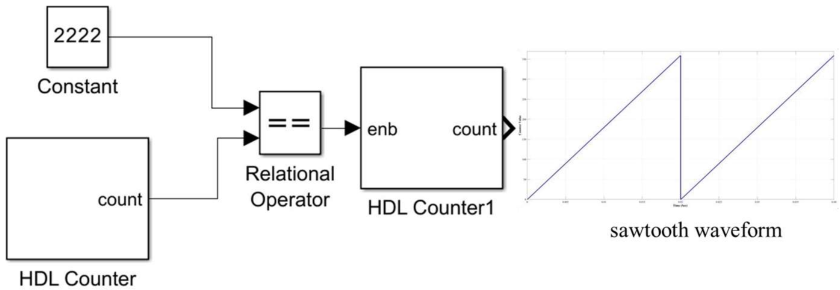


Fig.6. *Simulink diagram for generation of sawtooth waveform using HDL counters*

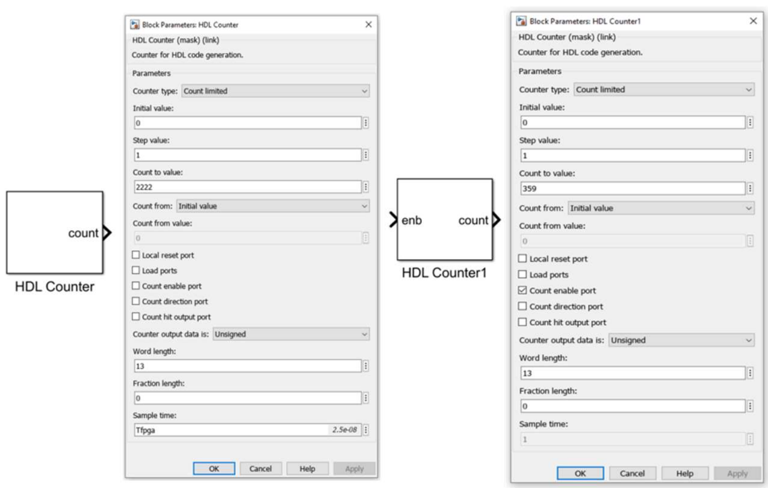


Fig.7. *HDL counters settings for generation of the sawtooth waveform*

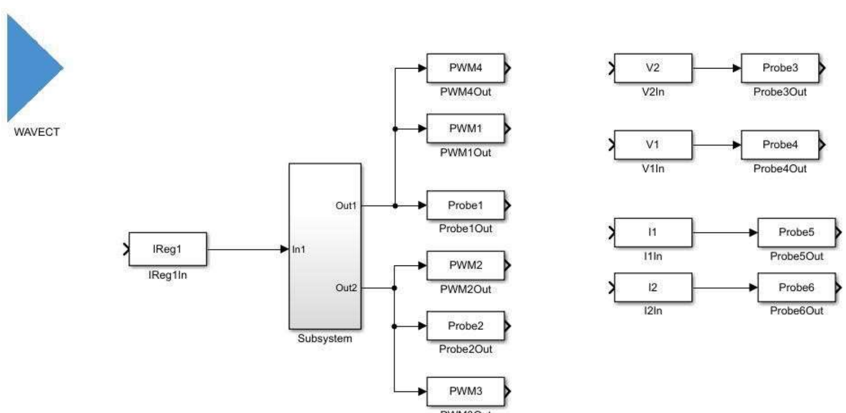


Fig. 8. *The main Simulink diagram*

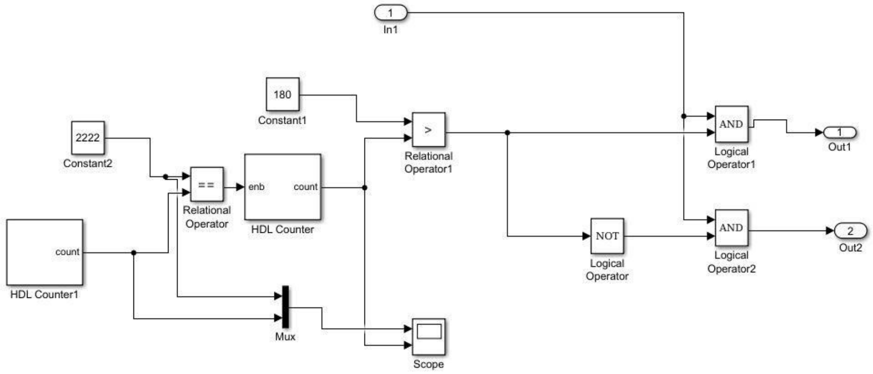


***Fig.9**. The subsystem in the main Simulink diagra*

Run all the workflow is completed and generate HDL code using Wavect block in Simulink. It generates the required binary file to dump in the WAVECT Controller. The voltage level of the pulses generated by the WAVECT Controller Board is only 15 V. So, there is no need for a level shifter to drive the switches of the single-phase square wave inverter.

4. Results and Discussion

The hardware setup of a single-phase inverter with Basys3 FPGA controllers is depicted in Fig. 10. A 100V DC is applied to the single-phase inverter through an uncontrolled rectifier. The switching pulses of the Basys3 FPGA controller are shown in Fig. 11. The voltage level of the switching pulses is 3.3 V.

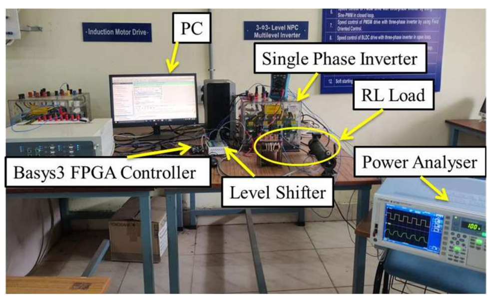


***Fig.10:** Hardware setup of single-phase inverter with Basys3 FPGA controllers*

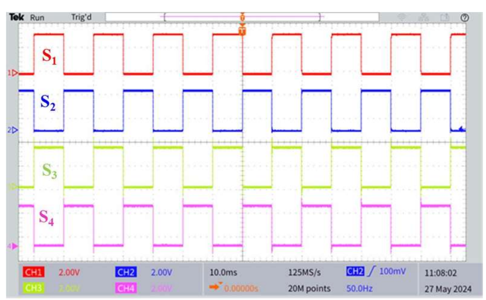


***Fig.11.** Switching pulses from Basys3 FPGA Controller.*

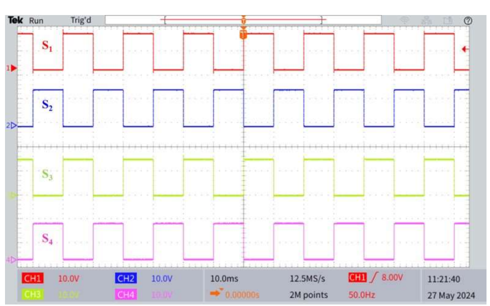


***Fig.12**. Level shifting of switching pulses obtained from Basys3 FPGA Controller*

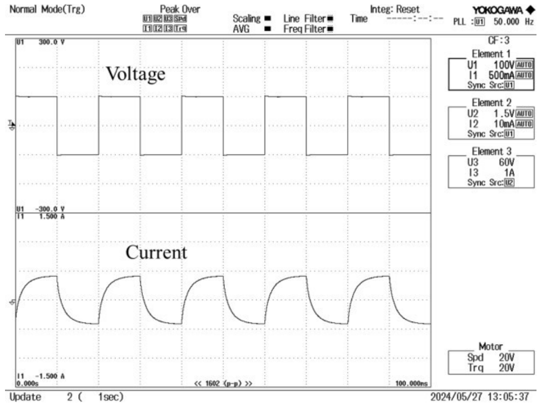


Fig.13. *Output voltage and current of single phase inverter with Basys3 FPGA Controller*

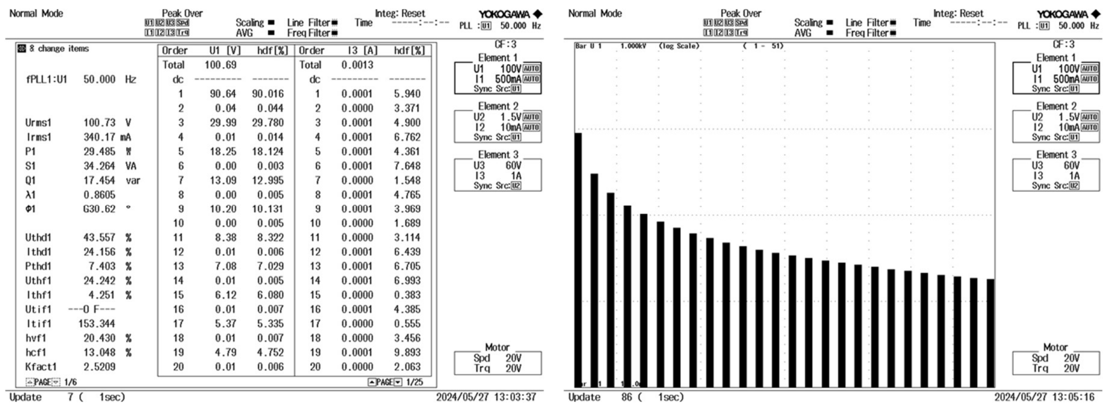


Fig.14. *FFT analysis of output voltage with Basys3 FPGA Controller*

The level-shifted switching pulses are shown in Fig. 12. The voltage level is now 15 V. The output voltage and current of single-phase square wave inverter with Basys3 FPGA Controller is depicted in Fig. 13. The FFT analysis of voltage and current are shown in Fig. 14 and Fig. 15 respectively. The fundamental peak of voltage and current are 90.64 V and

0.33 A respectively. Their %THDs are 43.557% and 24.156% respectively. The hardware setup of a single-phase inverter with WAVECT FPGA controllers is depicted in Fig. 16. A 100 V DC is applied to the single-phase inverter through an uncontrolled rectifier. The voltage level of the switching pulses is 15 V and depicted in Fig. 17. The output voltage and current of a single-phase square wave inverter with WAVECT FPGA Controller are depicted in Fig. 18. The output voltage and current of a single-phase square wave inverter with WAVECT FPGA Controller are depicted in Fig. 18.

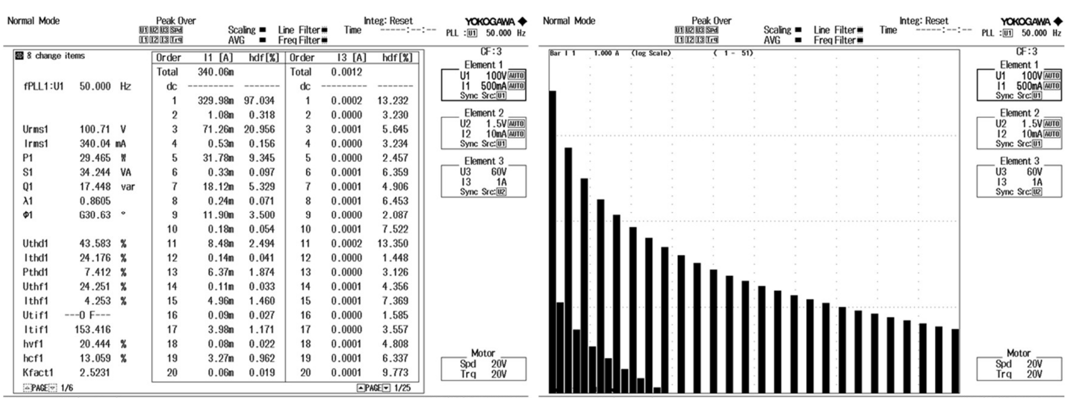


Fig.15. *FFT analysis of output current with Basys3 FPGA Controller*

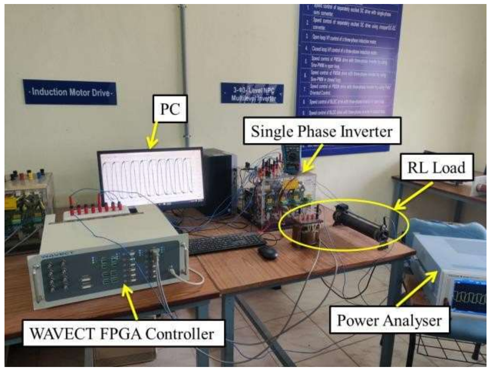


Fig.16. Hardware setup of single-phase inverter with WAVECT FPGA controllers

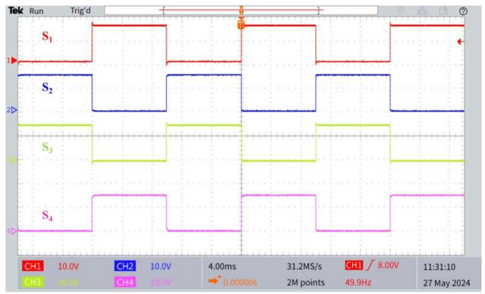


Fig.17. Switching pulses from WAVECT FPGA Controller.

The FFT analysis of voltage and current are shown in Fig. 19 and Fig. 20 respectively. The fundamental peak of voltage and current are 90.61 V and 0.32 A respectively. Their %THDs are 43.925% and 24.433% respectively. The comparison of fundamental peak and %THD for Basys3 and WAVECT FPGA Controllers is depicted in Table 1.

***Table 1** Comparison of Fundamental Peak And %THD of Basys3 and WAVECT FPGA Controllers*

S. No	Parameter	Fundamental Peak		%THD	
		Basys3 Controller	WAVECT Controller	Basys3 Controller	WAVECT Controller
1	Voltage	90.64 V	90.61 V	43.557	43.925
2	Current	0.33 A	0.32 A	24.156	24.433

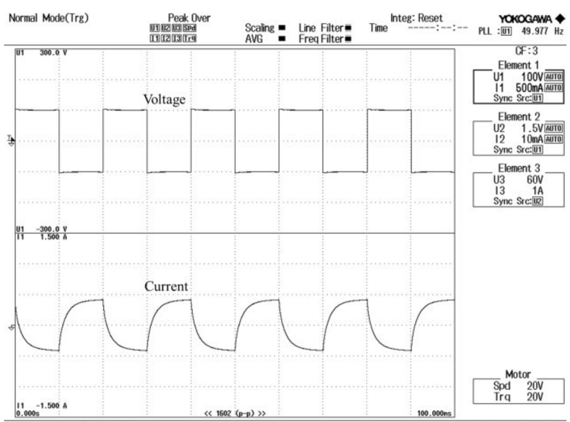


***Fig.18.** Output voltage and current of single phase inverter with WAVECT FPGA Controller*

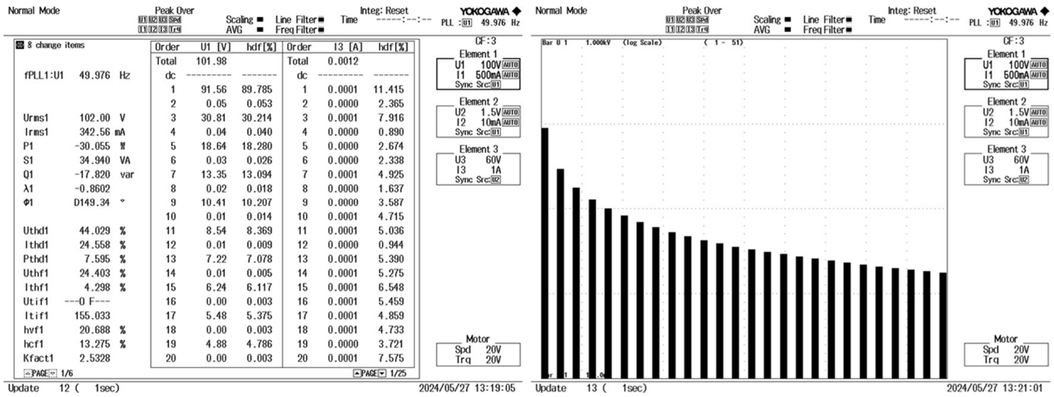


Fig.*19. FFT analysis of output voltage with WAVECT FPGA Controller*

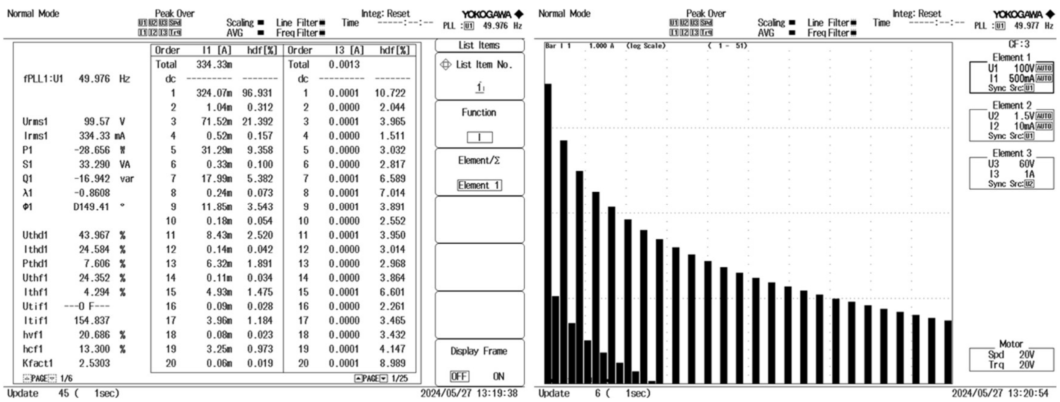


Fig. *20. FFT analysis of output current with WAVECT FPGA Controller*

5. Conclusion

This report presents the design and implementation of switching pulse generation for a single-phase square wave inverter using two different FPGA platforms: the Artix-7-based Basys3 and the Zynq-based Wavect controller. The study focused on evaluating the effectiveness of these FPGA controllers in generating switching pulses to drive an RL load and managing the inverter's output voltage and waveform quality, specifically in terms of THD. The comparison between the Basys3 Controller and the WAVECT Controller reveals that both controllers have nearly identical fundamental peak values for voltage and current. However, the WAVECT Controller shows a slightly higher THD in both voltage and current compared to the Basys3 Controller. This indicates that while both controllers perform similarly in terms of voltage and current output, the WAVECT system exhibits marginally More harmonic distortion. The study concluded that the Basys3 is appropriate for simple, budget-conscious projects, while the Zynq-based Wavect platform is more suited for applications that demand superior waveform performance and flexibility, such as those in renewable energy and power electronics. Overall, the report highlights the importance of selecting an FPGA platform based on the specific needs of the application, balancing cost, complexity, and performance requirements in inverter system design.

Compliance with Ethical Standards

Conflicts of interest: Authors declared that they have no conflict of interest.

Human participants: The conducted research follows the ethical standards and the authors ensured that they have not conducted any studies with human participants or animals.

References

[1] M. Rahman and M. Al-Sheikh, "Comparison of Analog and Digital Control Methods for DC-DC Converters", International Journal of Power Electronics and Drive Systems (IJPEDS), Vol. 1, no. 2, pp. 129-137, 2011.

[2] M. Ghamari and M. Gohari, "Digital vs. Analog Controllers for PWM Inverter Systems: A Comparative Analysis", Journal of Electrical Engineering & Technology, Vol. 8, no.3, pp. 651-659, 2013.

[3] S. Saha and J. Lee, “Design and Implementation of Digital Controllers for Power Electronics Circuits”, IEEE Transactions on Power Electronics, Vol. 29, no.12, pp. 6448-6456, 2014.

[4] S. Chien and L. Wang, “A Comparative Study of Analog and Digital Control for Brushless DC Motor Drive”, International Journal of Control, Automation, and Systems, Vol. 14, no. 6, pp. 1840-1848, 2016.

[5] J. Smith, and L. Zhao, “FPGA-based three-phase inverter for PWM control”, IEEE Transactions on Power Electronics, Vol. 34, no.5, pp. 5871-5880, 2019.

[6] A. Kumar and S. Gupta, “FPGA implementation of square wave inverter for power electronics applications”, International Journal of Electronics, Vol. 105, no.3, pp. 321-329, 2018.

[7] R. Patel, and N. Khanna, “FPGA implementation of a PWM inverter for renewable energy systems”, IEEE Access, Vol. 8, pp. 73421-73429, 2020.

[8] B. Singh, and R. Verma, “Zynq-based three-phase inverter design for renewable applications”, Journal of Power Electronics, Vol. 16, no. 6, pp. 1994-2003, 2017.

[9] Y. Wang and L. Zhang, “Zynq FPGA-based controller for grid-connected solar inverters”, Renewable Energy, Vol. 92, pp. 435-445, 2016.

[10] S. Lee, and M. Kim, “Reduction of harmonic distortion in FPGA-based inverters using digital signal processing”, IEEE Transactions on Industrial Electronics, Vol. 67, no.8, pp. 6578-6587, 2020.

[11] Basys 3 FPGA Controller reference manual, provided by Digilent, 2016.

[12] S. R. Kunduru, N. K. Reddy Chegireddy, and S. B. Choppavarapu, "Implementation of Nearest Vector Control and Nearest Level Control for Three Phase, Five Level CHB MLI with WAVECT Controller", 2022 International Conference on Recent Trends in Microelectronics, Automation, Computing and Communications Systems (ICMACC), Hyderabad, India, pp. 194-199, 2022