SEMARAK ILMU PUBLISHING

Semarak Engineering Journal

Journal homepage:
https://semarakilmu.com/
ISSN: 3036-0145



Design of an Intelligent Solar SPWM Inverter Based on FPGA

Murtadha Raoof Aldahhan[1,*], Hanan Abdul-Reda Akkar[1]

[1] Department of Electrical Engineering, University of Technology, Baghdad, Iraq

ARTICLE INFO

Article history:
Received 20 July 2025
Received in revised form 29 November 2025
Accepted 10 December 2025
Available online 5 January 2026

Keywords:

Inverter; Photovoltaic (PV); Sinusoidal Pulse-Width Modulation (SPWM); Artificial Neural Network (ANN); FPGA

ABSTRACT

This paper presents the development and implementation of an intelligent hybrid solar inverter with feedback. The inverter converts the applied direct current (DC) from a photovoltaic (PV) array combined with an energy storage device (Battery) into pure sinusoidal alternating current (AC) at the specified output voltage of 220V and 50Hz frequency in conjunction with an AC power source (Grid) to supply sufficient power during a power outage. This work aims to construct a bipolar Sinusoidal Pulse Width Modulation (SPWM) employing a Field Programmable Gate Array (FPGA) for a single-phase full-bridge inverter with feedback to maintain the output power. In MATLAB, an Artificial Neural Network (ANN) is created and trained using the Levenberg-Marquardt (LM) algorithm to find the ideal configuration for the SPWM inverter's modulation index. The control method for the proposed inverter with feedback has been designed using MATLAB/SIMULINK and ISE Suite 14.7. The controller with ANN feedback was converted to HDL code and uploaded on the Xilinx Spartan 6 FPGA board. Simulation results have shown that the inverter's efficiency at a full load increased to 91.95% and by adjusting the modulation index and switching frequency, the Total Harmonic Distortion (THD) of the sinusoidal output waveform can be minimized.

1. Introduction

Solar energy is regarded as one of the cleanest and most accessible renewable energy sources for power generation. Renewable energy sources such as solar are combined with energy storage devices, such as a battery, to operate as a standalone unit [1]. The idea is simple, constructing a smart inverter using bipolar Sinusoidal Pulse Width Modulation (SPWM) employing a Field Programmable Gate Array (FPGA) for a single-phase full-bridge inverter to maintain the output power by adjusting the duty cycles depending on the feedback system. SPWM has two switching strategies, unipolar and bipolar techniques. Inverters transform Direct Current (DC) from a solar panel or battery into Alternating Current (AC). According to the design of the power electronic switches, inverters can be constructed in half-bridge or full-bridge topologies [2]. Total Harmonic Distortion (THD) is affected by changes in Modulation Index (MI) and carrier frequency (fc). An artificial neural network presents the feedback system to maintain the output sinusoidal amplitude [3]. There are two types of Neural

* *Corresponding author.*
E-mail address: eee.19.26@grad.uotechnology.edu.iq

Networks (NN) based on their learning strategy: supervised (where the output values are known in advance) and unsupervised (where the output values are unspecified). The most popular NN training algorithm is the Back Propagation (BP) algorithm, where the outputs of the NN are compared to the desired output. If the comparison is unsatisfactory, the connections (weights) between the layers are adjusted, and the procedure is repeated until the error is sufficiently reduced [4]. Field Programmable Gate Array (FPGA) is an Integrated Circuit (IC) that comprises configurable (programmable) logic blocks, configurable interconnections between these blocks, and programmable input and output. FPGA may be programmed once or several times, depending on its implementation method [5]. A solar charge controller equipped with Maximum Power Point Tracking (MPPT) technology may increase the battery charging current by monitoring the maximum power point (MPP) in order to extract the maximum quantity of energy possible from the PV arrays [6].

2. Related work

Many studies tried to reduce the Total Harmonic Distortion (THD) of the output for a DC to AC full-bridge inverter system and get smooth sinusoidal and solve the drop voltage at the load with a fast response by feedback. The feedback is controlled using an artificial neural network (ANN) through changing modulation Index (MI). Employing Solar energy as a DC power source is preferred as a renewable energy power source because it is easily available in every part of the country with a battery to be utilized standalone or in combination with a grid-connected AC power supply. Due to variable solar radiation, the battery is hardly fully charged from a single photovoltaic source or limited grid power, particularly in rural areas. As a result, the solar energy system must be connected to other alternating current power sources to provide 24-hour electricity. The need for an FPGA has emerged as a device with high-speed switching, resolution, and accuracy to reduce cost and circuit complexity [7,8]. Hassaine and Bengourina, 2020 [9] presented a photovoltaic system that is connected to the grid. The major function of the inverter is regulating the amount of solar-generated power sent into the grid via the inverter. The quality of the power delivered to the grid and the efficacy of the converter system are both affected by the current inverter management. A Digital Sinusoidal Pulse-Width Modulation (DSPWM) control process is presented to modify the power factor of a grid-connected solar system and synchronize the sinusoidal output current with the grid voltage. A single-phase bipolar PWM inverter with lineal current control serves as the foundation for this control. Utilizing VHDL, this controller is built as a digital design on an FPGA. The ability of the recommended SPWM to considerably improve the power quality of grid-connected solar inverter outputs while simultaneously reducing the cost of the hardware used in these systems. Sarker, Datta, and Debnath, 2020 [10] developed an FPGA-based generator design utilizing the SPWM technique for Voltage Source Inverter (VSI) systems with an adjustable modulation index and high-frequency switching. The modulation index is controlled by a mathematically developed FPGA architecture, which modifies the duty cycle values of the SPWM pulses in response to inverter output measurements. The experimental setting indicated that the suggested Voltage-Source Inverter (VSI) application could effectively manage the Modulation Index (MI) value of the SPWM generation across a variety of input and load situations employing an FPGA-based architecture. Ali, Hassan, and Azmy, 2020 [11] proposed research on multilevel inverter (MLI) using two PWM approaches for AC devices. Five switching devices make up the MLI's primary and secondary circuits. The basic circuit controls output voltage polarity with a single-phase H-bridge inverter. The input DC bus voltage is regulated by the secondary circuit, which modifies the DC count of cells in the primary circuit at the front end. Efficiency ranges from 88.5% to 90.3%, while THD is 3%. Ahmed, N. and Khan, Z.R., 2021 [12] proposed a pure sine wave single-phase inverter based on a microcontroller. The system outputs

220V/50Hz. The design uses sinusoidal pulse width modulation. The inverter has lower harmonics when tested on varied AC loads up to 150W. The feedback control system improves voltage regulation and efficiency (72%–80%). Due to the high switching frequency at 20kHz, the THD of output is also decreased. Syahputra et al., 2022 [13] designed a square inverter utilizing an IC SG3525 in order to increase the input voltage to 220V with feedback for a maximum load of 100W, as an oscillator generator, an IRFZ44 MOSFET driver as a power amplifier, and a step-up transformer. In order to ensure a nearly constant switching frequency by load, Chacko et al., 2023 [14] proposed a new control technique for single-phase switching frequency regulation by using the low-pass-filtered inverter output voltage as the feedback signal with a low-cost programmable microcontroller. The results of the simulation and experiment show that the suggested control strategy can maintain the output voltage in the presence of load and source side changes.

3. Methodology

3.1 The Proposed System Design

The system design is illustrated in Figure 1 consists of an FPGA, H-bridge inverter, DC-to-AC rectifier, and three inputs (Grid, PV, battery). The FPGA controls the priority of these three input sources. During PV and battery-working time, the Xilinx Spartan 6 FPGA (XC6SLX45) board attached to the driver circuit generates PWM signals.

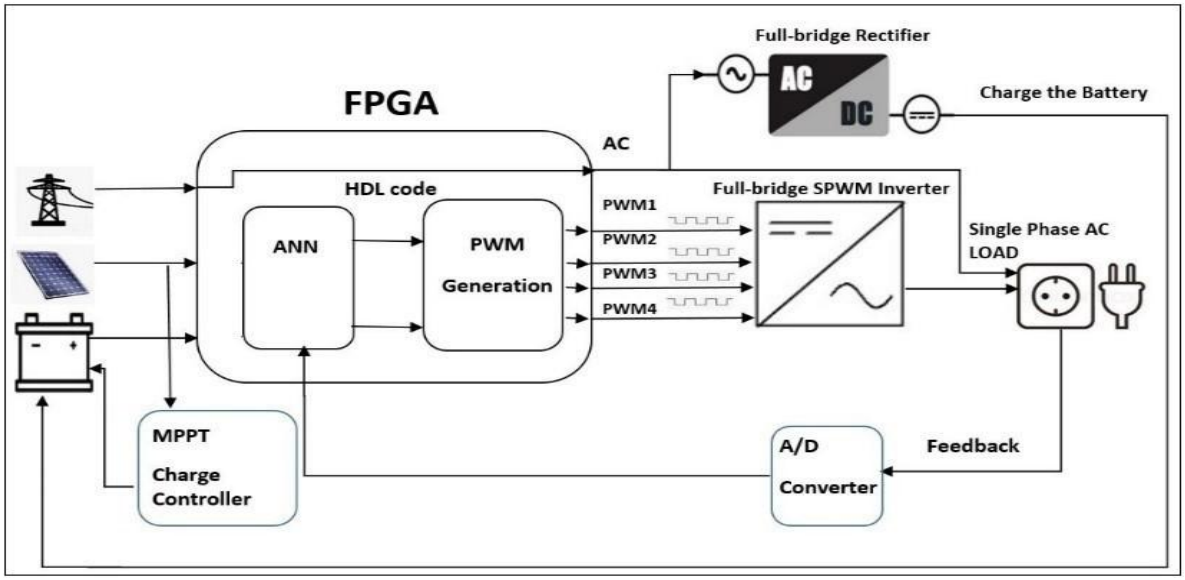
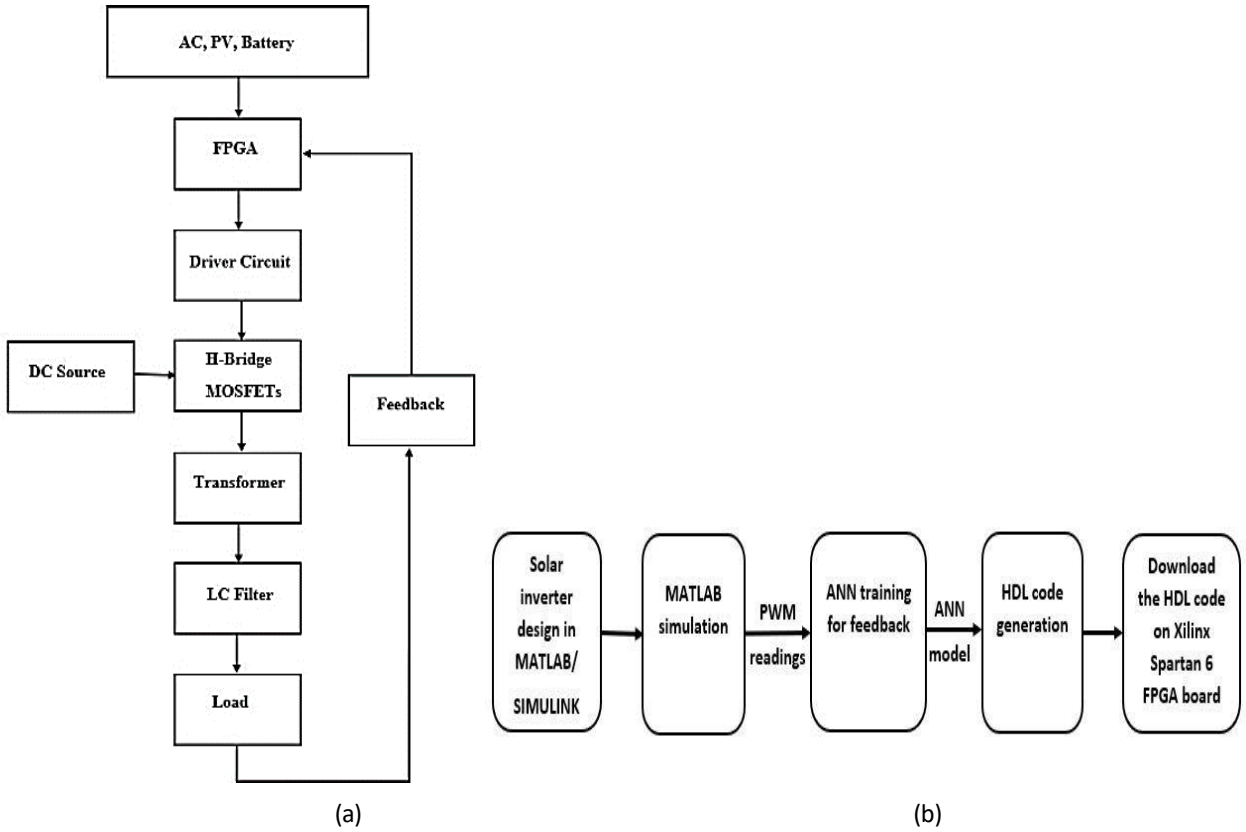


Fig. 1. The proposed system design

This driver circuit is connected to a full-bridge inverter to generate a sinusoidal output voltage. The priority is assigned to the AC Grid, photovoltaic solar panel, and battery, respectively. The system selects AC power from the Grid directly to power the residence and charges the battery when AC exists. The second priority is given to PV panels to provide DC power to the full-bridge inverter. Otherwise, the power from the battery is selected to be transmitted to the inverter's DC source during a power outage or when solar irradiation is inadequate. When solar and Grid power is insufficient to meet the demand, a backupbatteryprovides the required DCinput to the inverter circuit until any of the two primary power sources (PV, Grid) is restored. AC power from the Grid charges the battery with a full-bridge AC/DC rectifier, while PV panels charge the battery through an MPPT device. The driver circuit consists of 4 optocouplers, which amplify the output signals from FPGA from 3.3 to 15 Vdc.

The driving circuit outputs are wired to the gate terminals of the MOSFETs. An AC transformer (24 to 220 volts) is connected to the middle legs of the MOSFETs. A low-pass filter is connected to the output of the transformer to minimize the THD and eliminate high-order frequencies from the inverter's output signal. The intelligent controller implemented on FPGA is designed to analyse the output voltage at the load by feedback with ANN (Artificial Neural Network) to maintain the output power under different loads. The FPGA board generates the PWM signals linked to the driver circuit. The outputs from the driving circuit are linked to the gate terminals of each of the four MOSFETS. The output of the inverter is then filtered out to produce a pure and smooth sine wave signal. The block diagram shown in Figure 2(a) describes the inverter's block diagram and Figure 2(b) shows the design workflow, illustrating the working steps associated with the proposed design.



(a) (b)

Fig. 2. (a) Block diagram of the inverter (b) Design workflow

3.2 Sinusoidal Pulse-Width Modulation (SPWM)

This thesis focuses on using Sinusoidal Pulse-Width Modulation (SPWM), as it has proven successful, relatively easy, and less complex to implement in the control system. The primary advantage of employing SPWM switching for power electronic devices such as MOSFETs and IGBTs in the inverter is the decrease in harmonics [15]. Figure 3 shows the simulation for the Bipolar SPWM inverter in MATLAB/SIMULINK. This MATLAB simulation compares the reference or fundamental wave with a sawtooth carrier wave to generate a PWM pulse with its complement, where switching angles are obtained from a MATLAB code and then used later in FPGA. The Modulation Index (MI) is set to 1. The carrier frequency (fc) was set to 10kHz, and the reference sinewave was adjusted to 50 Hz.

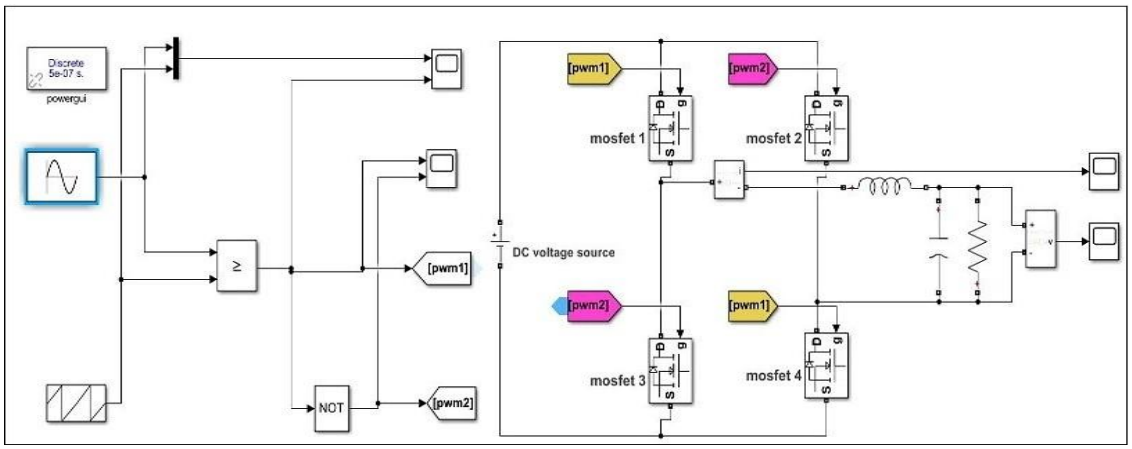


Fig. 3. MATLAB simulation for the full-bridge bipolar inverter

Two Pulse Width Modulation (PWM) are generated (PWM1 and PWM2). The first pulse width (PWM1) drives MOSFET 1 and MOSFET 4, and the second (PWM2) drives MOSFET 2 and MOSFET 3. With an effective low-pass filter, a smooth sine wave signal is filtered out from the inverter's output. The total generated pulses (n) for one time period (20 ms) are 200 pulses as follows [16]:

$$f_0 = \frac{f_s}{n} \tag{1}$$

Where f_s is the switching frequency, f_0 is the desired output frequency, and n is the number of sampling pulses.

The THD for different carrier frequencies and modulation indexes are calculated and compared through this simulation. Then the pulse widths of generated pulses during the positive half-time period are extracted from the PWM1, which are complementary to the negative half-wave. In this proposed system, there are three switching frequencies employed with a 10 kHz carrier frequency, the first one when modulation index = 1, the second when modulation index = 0.9, and the third calculation applied for modulation index = 0.8. These three-switching frequency calculations were used later in FPGA programming, which drives the full-bridge MOSFETs according to the desired output voltage.

3.2.1 Switching angle computations

The input data of generated PWM signals are analyzed mathematically by MATLAB, and at the intersection of the carrier and the signal reference signal, PWM pulses with duty cycles ranging from αn to ßn are generated for each pulse, as illustrated in Figure 4 by bipolar SPWM method for positive half. The carrier frequency or switching frequency (fc) is set to 10KHz, and the Modulation Index (MI) is set to 1. The illustrated figure shows 200 pulses at one time period (20ms), 100 PWM pulses for each half ranging from (t1-t100). The time period (T) of the reference signal can be calculated as follows [16]:

$$T = \frac{1}{f_r} \tag{2}$$

The switching Angles (αn to ßn) and the duty cycle (pulse width) for each pulse can be expressed as follows [17]:

$$\alpha_n(t) = \alpha_n{}^{\circ}\frac{T/2}{180^0} \tag{3}$$

$$ß_n(t) = ß_n{}^{\circ}\frac{T/2}{180^0} \tag{4}$$

$$Duty\ cycle = ß_n - \alpha_n \tag{5}$$

Where fr is the frequency of the input reference signal, α_n is the switching angle of the pulse for the positive edge, and $ß_n$ is the switching angle of the pulse for the negative edge.

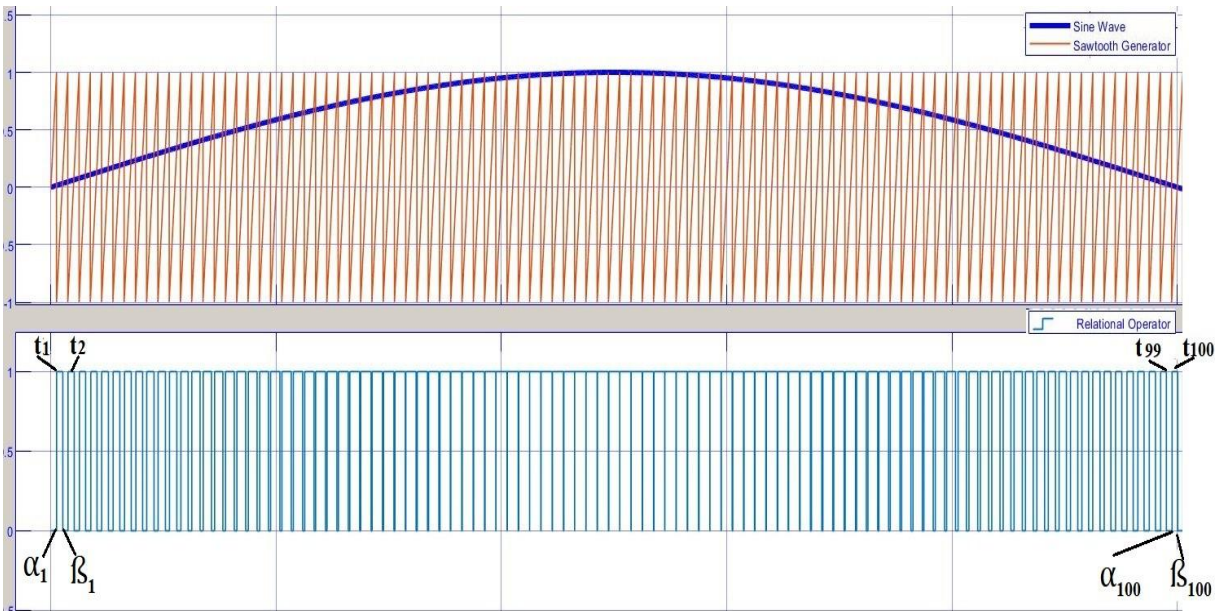


Fig. 4. PWM switching angles for the positive half (fc = 10kHz, MI=1)

3.2.2 Total Harmonic Distortion (THD)

The THD analysis is performed using FFT (Fast Fourier Transform) using MATLAB to determine the ratio of the RMS value of its total harmonic component of the output voltage to the RMS value of the fundamental component. The variation in %THD of output sinusoidal voltage wave concerning changes in carrier frequency and modulation index has been compared for the bipolar scheme. Total harmonic distortion is frequently used in cases when the dc term is zero. THD may be used in these cases to quantify both voltage and current waveforms [18].

The THD in the voltage is defined as:

$$THD_{v\%} = \frac{;\overline{\sum_{n=2}^{\infty} V_n{}^2}}{V_1} * 100\% \tag{6}$$

Where V_n is the amplitude of the n^{th} order harmonic of the inverter output voltage, and V_1 is the fundamental voltage.

Based on the simulation of this scheme, the THD of the output voltage in bipolar is determined for different modulation indexes (MI=1), (MI=0.9), and (MI=0.8), respectively with three carrier frequencies (fc = 10 kHz), (fc = 8 kHz) and (fc = 4 kHz). Figure 5 shows the %THD percentage of output voltage for the bipolar scheme in FFT analysis with Modulation Index (MI=1) and a carrier frequency of (10kHz). The obtained results are summarized as shown in Table 1. Figure 6 illustrates the THD analysis from Table 1 concerning different switching frequencies for the bipolar scheme as a graphical curve.

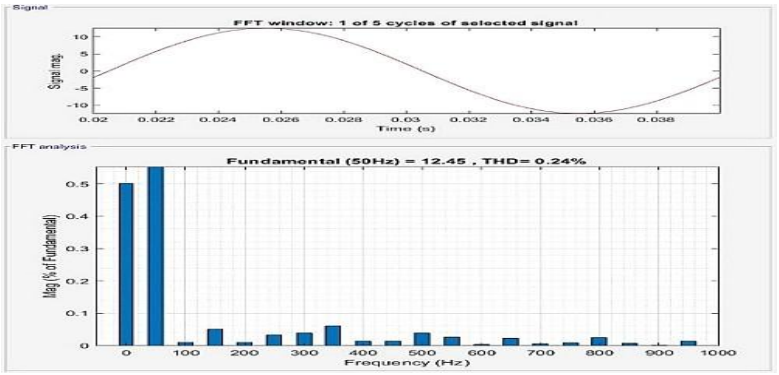


Fig. 5. FFT analysis of output voltage for the bipolar inverter with (MI =1), carrier frequencies (10kHz)

Table 1
THD analysis for bipolar SPWM inverter

SPWM Scheme	Frequency of Fundamental Signal	Modulation Index (MI)	Carrier Frequency (fc)	%THDv
Bipolar	50Hz	1	10kHz	%0.24
			8kHz	%0.35
			4kHz	%1.28
		0.9	10kHz	%0.28
			8kHz	%0.41
			4kHz	%1.53
		0.8	10kHz	%0.34
			8kHz	%0.49
			4kHz	%1.87

3.2.3 SPWM Simulation

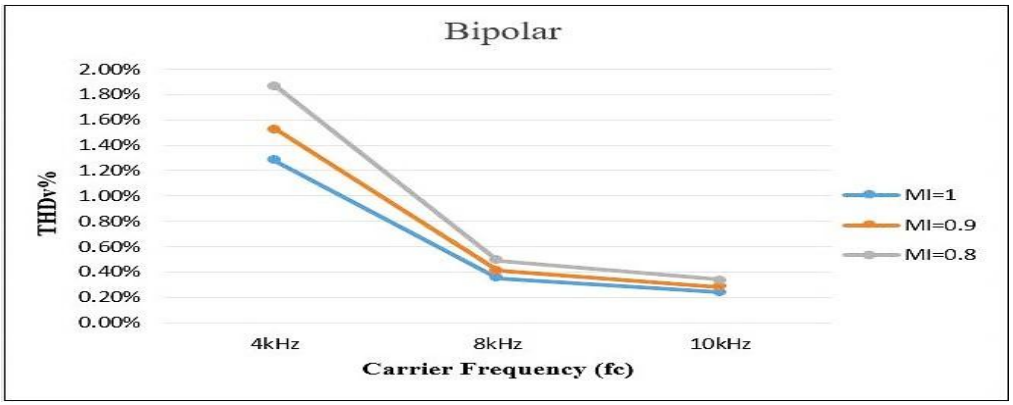


Fig. 6. THD analysis of the output voltage for bipolar inverter under different switching frequencies

The bipolar SPWM technique was simulated in MATLAB/SIMULINK with different switching frequencies (4khz,8khz,10khz) and various modulation indexes (0.8,0.9,1) to obtain the most sinusoidal output wave with minimum THD for output voltage. The obtained switching pulses from the bipolar simulation for a half-wave, which presents 100 pulses of the positive half-wave, are shown in Table 2 with their details. These datasets were collected for a switching frequency of 10 kHz and a modulation index of 1. Using MATLAB/SIMULINK, the datasets were derived from generated bipolar PWM, as shown in Figure 7. The calculation was performed by MATLAB coding. The PWM shows 200 pulses for a period of 0.02 seconds to satisfy a frequency of 50 Hz. Where the width of each pulse (duty cycle) is recorded, which is taken for use in experimental work later in FPGA. The datasets are utilized in the FPGA board's HDL code to acquire the PWM signals that drive the MOSFETs. The switching angles are also calculated for modulation indexes of (MI=0.9) and (MI=0.8), respectively. The same steps are repeated to generate 100 datasets by changing the modulation index to (0.8 and 0.9). The datasets are used to create the HDL code of the PWM signals (PWM1, PWM2, PWM3, PWM4), which will control the gate of the MOSFETs. The first half-wave datasets will be complemented to obtain the switching angles of the negative half-wave. The prototype system design employed a bipolar SPWM method with a switching frequency of 10kHz and modulation indexes of 0.8, 0.9, and 1 rather than a unipolar technique due to the simplicity of creating HDL code and generating the implementing PWM signals. When an effective low pass filter is used, the THD % of bipolar and unipolar schemes will be similar at high switching frequencies (10kHz and above); hence selecting bipolar over unipolar will not significantly alter the output sinusoidal wave at these frequencies.

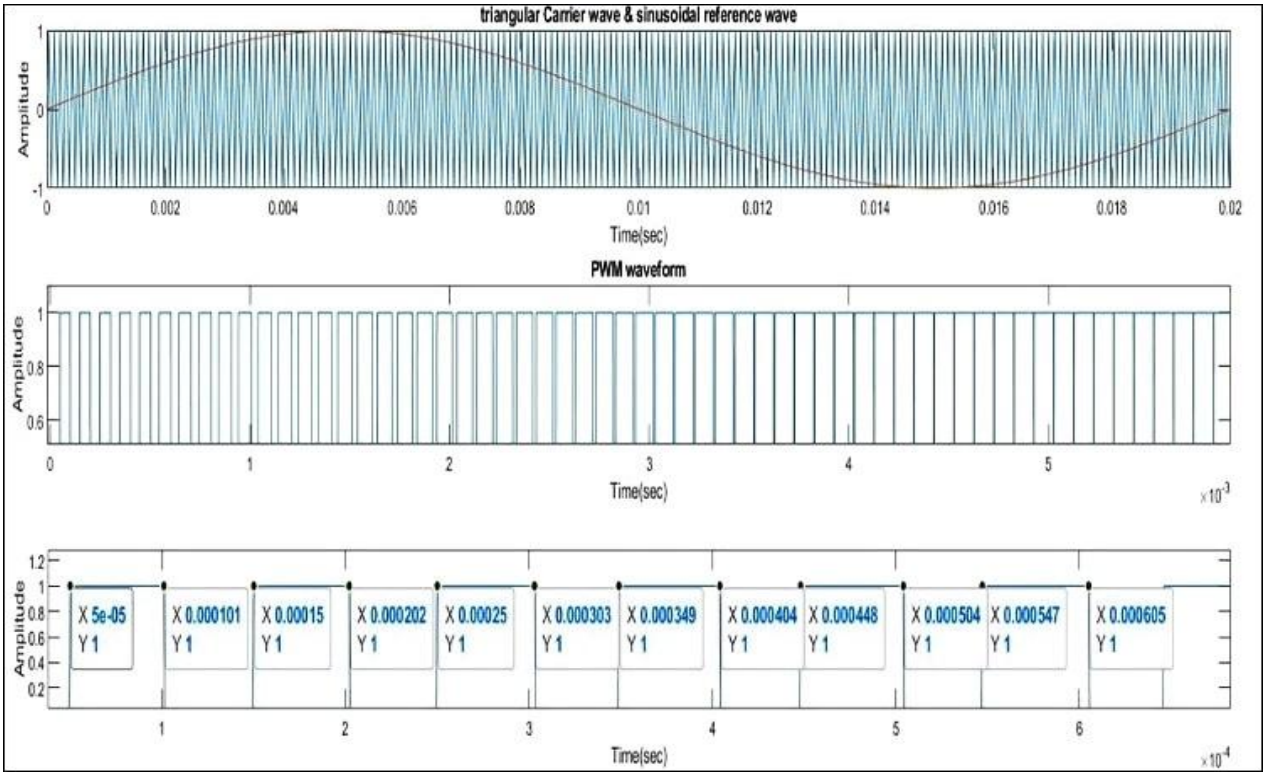


Fig. 7. Bipolar PWM generation (fc = 10 kHz, MI= 1) by MATLAB

Table 2
Switching angles of Bipolar SPWM for fc=10 kHz and MI =1

Switching pulses	α˚	Ɓ˚	α(t) second $*10^{!5}$	Ɓ(t) second$*10^{!5}$	Pulse width(t) second $*10^{!5}$
t1	0.9	1.8288	5	10.16	5.16
t2	2.7	3.6576	15	20.32	5.32
t3	4.5	5.4864	25	30.48	5.48
t4	6.3	7.3152	35	40.64	5.64
t5	8.1	9.1422	45	50.79	5.79
t6	9.9	10.971	55	60.95	5.95
t7	11.7	12.7998	65	71.11	6.11
t8	13.5	14.6268	75	81.26	6.26
t9	15.3	16.4556	85	91.42	6.42
t10	17.1	18.288	95	101.6	6.6
:	:	:	:	:	:
:	:	:	:	:	:
t90	161.1	162.27	895	901.5	6.5
t91	162.9	164.052	905	911.4	6.4
t92	164.7	165.816	915	921.2	6.2
t93	166.5	167.598	925	931.1	6.1
t94	168.3	169.362	935	940.9	5.9
t95	170.1	171.144	945	950.8	5.8
t96	171.9	172.908	955	960.6	5.6
t97	173.7	174.69	965	970.5	5.5
t98	175.5	176.454	975	980.3	5.3
t99	177.3	178.236	985	990.2	5.2
t100	179.1	180	995	1000	5

3.3 Feedback and Design Procedure of ANN

ADC (Analog to Digital Converter) is necessary to convert analog signals to digital numbers so that FPGA can read and process the feedback signals. If there is a voltage drop at the output by a higher load, the ANN with the backpropagation algorithm is responsible for fixing the issue and maintaining the output power by increasing the modulation index to boost the output voltage. One way to increase the efficiency of the output power is by increasing the modulation index, which leads to an increase in the output voltage [16]. The Modulation Index is given as:

$$M_i = \frac{V_r}{V_{tri}} \tag{7}$$

Where, V_{tri} is the triangular carrier-wave signal, V_r is the sinusoidal reference signal, and M_i is the modulation index. By altering the Modulation Index (M_i), the output voltage can be altered. The following equation estimates the output voltage (V_0) amplitude [16].

$$V_0 = M_i.V_d \tag{8}$$

Where, V_d is the voltage DC input source.

The feedback from the ADC, which converts analog to digital, is supplied as an input to the ANN intelligent controller, which can change the Modulation Index (MI) to an appropriate one, allowing the output to fix the drop voltage and maintain the output power. When the inverter's output voltage drops, the Vin of the ADC is also affected, and its value will change accordingly. The output digits of the ADC which are the inputs to FPGA are the ANN input datasets. The input-output datasets of the constructed and trained network based on the Levenberg-Marquardt (LM) algorithm are shown in Table 3. LM algorithm has a faster training time than other gradient descent training algorithms. Satlins function is used as an activation function in the training process [19].

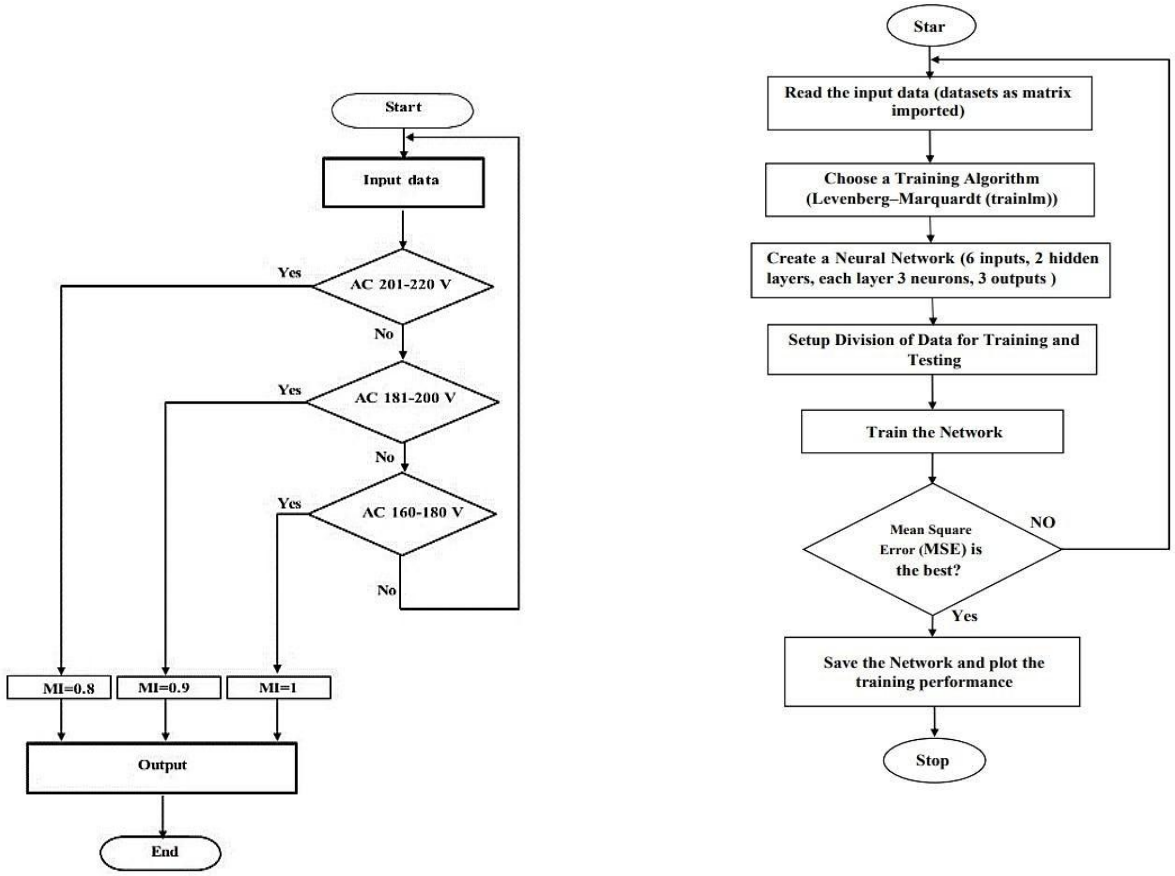
Table 3
Input-output training data for ANN

Inputs						Outputs		
A5	A4	A3	A2	A1	A0	MI=0.8	MI=0.9	MI =1
1	1	1	1	1	1	0	1	1
1	1	1	1	0	0	0	1	1
1	1	1	0	1	1	0	1	1
1	1	1	0	0	0	0	1	1
1	1	0	1	1	1	0	1	1
1	1	0	1	0	0	0	1	1
1	1	0	0	1	1	0	1	1
1	1	0	0	0	0	0	1	1
1	0	1	1	1	1	0	1	0
1	0	1	0	0	0	0	1	0
1	0	0	1	1	1	0	1	0
1	0	0	1	0	0	0	1	0
1	0	0	0	1	1	0	1	0
1	0	0	0	0	0	0	1	0
0	1	1	1	1	1	1	0	1
0	1	1	1	0	0	1	0	1
0	1	1	0	1	1	1	0	1
0	1	0	1	1	1	1	0	1
0	1	0	1	0	0	1	0	1
0	1	0	0	1	1	1	0	1
0	1	0	0	0	0	1	0	1
0	0	0	1	0	0	1	0	0
0	0	0	0	1	0	1	0	0

The multilayer Neural Network (NN) consists of six inputs, three outputs, and two hidden layers. For each training procedure, the Mean Square Error (MSE) was determined using some hidden layers with neurons taken as (3). Setting the modulating index to 0.8 with a switching frequency of 10kHz will generate a sinusoidal wave with a (40V) decrease in amplitude, and modulating index to 0.9 with a switching frequency of 10kHz will generate a sinusoidal wave with a (20V) decrease in amplitude. Figure 8(a) presents the proposed controller's flowchart. The AC transformer used in this design has one single primary winding and three secondary windings. The default secondary winding is selected at high voltage winding (260V) with a default modulation index of 0.8, so the generated output voltage is (220V).

The proposed Back Propagation Neural Network (BPNN) consists of three primary layers. An input layer, an intermediate layer (hidden layer), and an output layer [20]. The input layer consists of six inputs (A0 to A5). The last two bits (A6 and A7) are always binary 1 in this design, and to minimize the inputs to FPGA, ADC output pins are reduced to 6 bits. The ANN network was trained with two training algorithms Levenberg-Marquardt (LM) algorithm and the Fletcher-Powell Conjugate Gradient (CGF) algorithm. The best with high accuracy training has been obtained when using (LM)

algorithm; the transfer function used in training is "satlins", and the network training function is "trainlm". The output layer generates three values (MI1, MI2, MI3), selects the modulation index according to backpropagation, and adjusts neuron weight for each layer based on error propagation. Figure 8(b) illustrates the block diagram of MATLAB's algorithm for ANN training.



(a) (b)

Fig. 8. (a) Flowchart of the proposed feedback controller (b) BPNN training procedure for feedback

3.3.1 ANN training and HDL code generation for feedback

The Artificial Neural Network (ANN) is highly efficient in terms of feedback response in comparison to other methods, due to its minimal training time, high accuracy, and high-speed response. MATLAB R2019b was utilized to train the BP neural network's algorithms. After experimenting with various activation functions, the command (newff), which stands for (feed-forward backpropagation network), is used to build the used network and applied "satlins" as an activation function. Many attempts during training are made before choosing a specific algorithm, activation function, and several hidden layer neurons. The training algorithm has been executed and retrained several times to get the ideal result. The training is performed by several techniques to choose the optimal performance based on Mean Square Error (MSE), the number of neurons in the hidden layer, and the number of epochs necessary to achieve this performance. To select the intelligent controller with the best performance and to acquire the fewest neurons with the lowest mean square error, two configurations of the ANN network were trained using Levenberg-Marquardt (LM) algorithm [21]. The training results show that the best performance for three neurons (N=3) at MSE equals (1.89×10^{-26}) and six epochs. Figure 9 illustrates the ANN's training results.

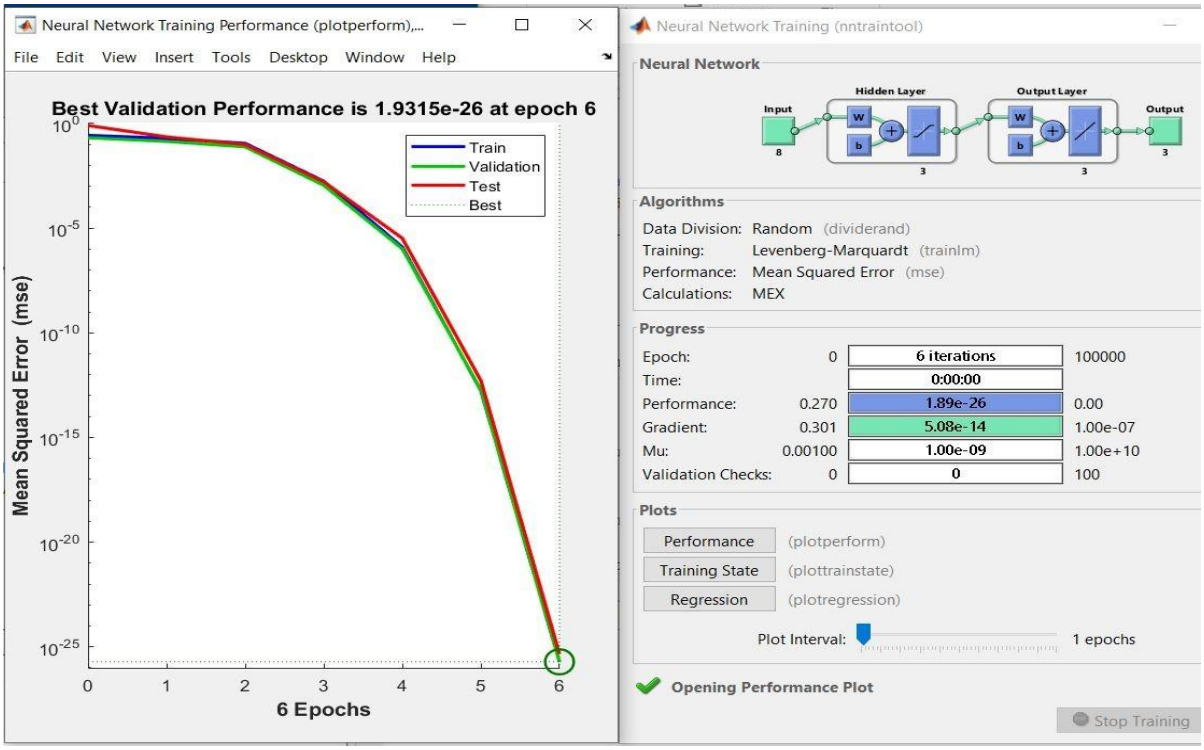
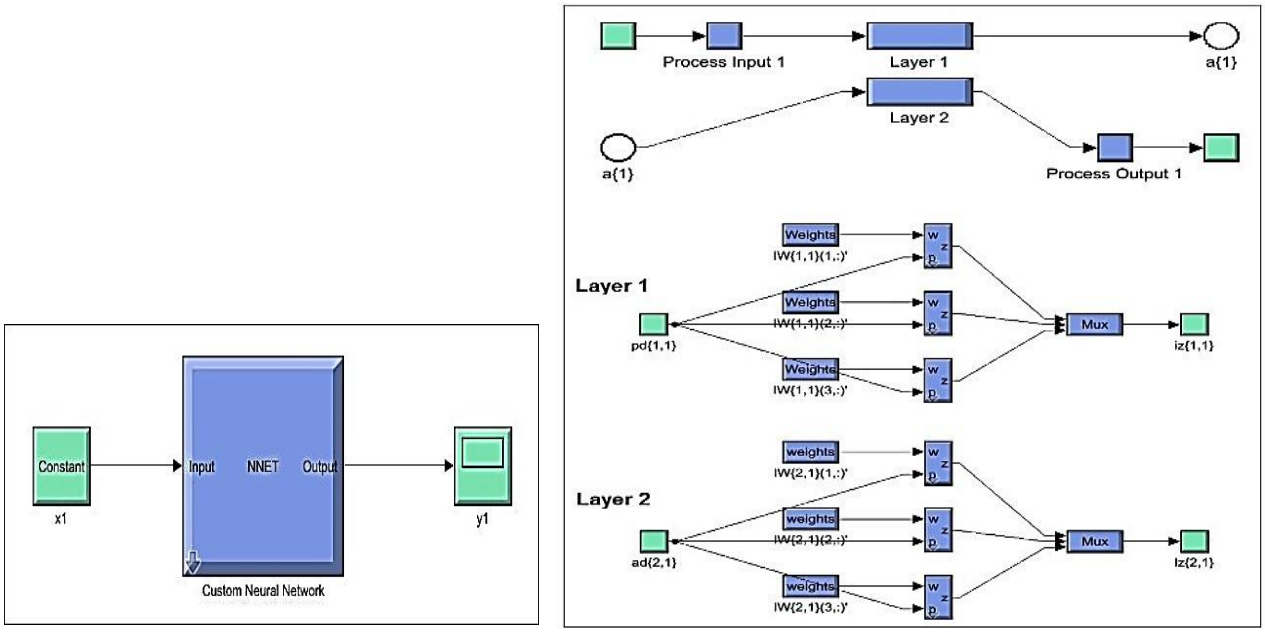


Fig. 9. The training results of N=3 using LM algorithm

After finalizing the training for artificial neural networks, the next stage is translating the taught neural network into HDL code for FPGA implementation. The following processes used to create HDL code will be implemented in hardware design. As previously described, the Levenberg-Marquardt method is used to train Neural Networks (NN). Figure 10(a) illustrates the neural network model built-in Simulink by "genism" code after training. The model containing new weights is transformed into HDL code for FPGA implementation. The internal structure of the neural network model is seen in Figure 10(b).

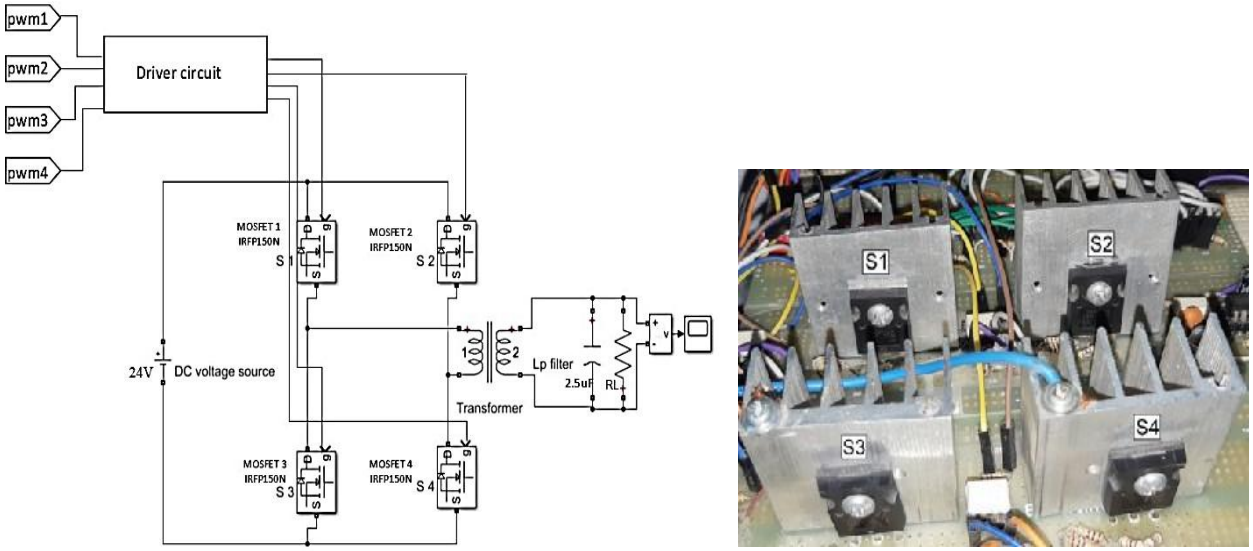


(a) (b)

Fig. 10. (a) Simulink model of neural network (b) The internal structure of neural network model

3.4 Implementation of Hardware Design

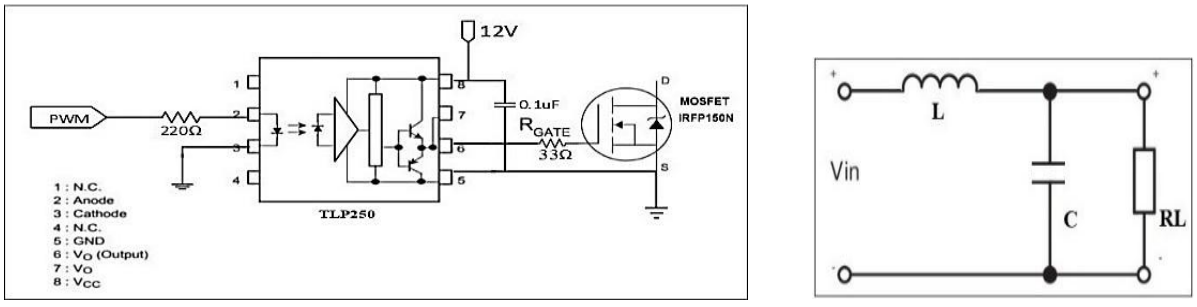
To convert direct current to alternating current, an H-bridge inverter is employed, and it consists of four MOSFET transistors, and in this work, IRF840 MOSFET is used as a power switch. As MOSFET is very heat sensitive so, for cooling purposes, four heat sinks are used. The MOSFET drain is attached to the heat sink, and the transformer is connected to the inverter's output. The transformer's inductance L and the output capacitor can form an L-C low pass filter. The output is supplied from the drains of the two lower MOSFETs as the upper MOSFET source is connected to the drain of the lower MOSFET, as shown in Figure 11(a) and the hardware circuit setup for the H-Bridge MOSFETs is shown in Figure 11(b).



(a) (b)

Fig. 11. (a) H-bridge inverter circuit (b) H-Bridge hardware setup

For driving the gate of the MOSFET, the TLP250 gate driver is used. The Gate driver circuit pin diagram is shown in Figure 12(a). As FPGA output pins deliver 3.3V and which is not sufficient to drive the MOSFET gate, which is not active until it is fed with 12V and high current. The Gate driver is powered by low-voltage signals from the FPGA and provides the necessary high-voltage gate drive for the MOSFETs. TLP250, which also functions as an isolation circuit providing separation between the low-power circuit FPGA outputs and high-power MOSFETs circuit. Figure 12(b) shows the low-pass filter circuit, while the following equation describes the relationship between the filter components and the cut-off frequency of a low-pass filter [22].



(a) (b)

Fig. 12. (a) TLP250 gate driver circuit configuration (b) Low-pass filter circuit [22]

$$f_C = \frac{1}{2\pi\sqrt{L\,C}} \tag{9}$$

Where L is the filter inductance, C is the filter capacitance, and f_C is the cut-off frequency.

In this project, MATLAB R2019b and Xilinx ISE 14.7 are utilized for FPGA programming. Figure 13 shows the hardware layout of the proposed system hardware design for the proposed intelligent solar inverter. The parameters of the hardware components are reported in Table 4.

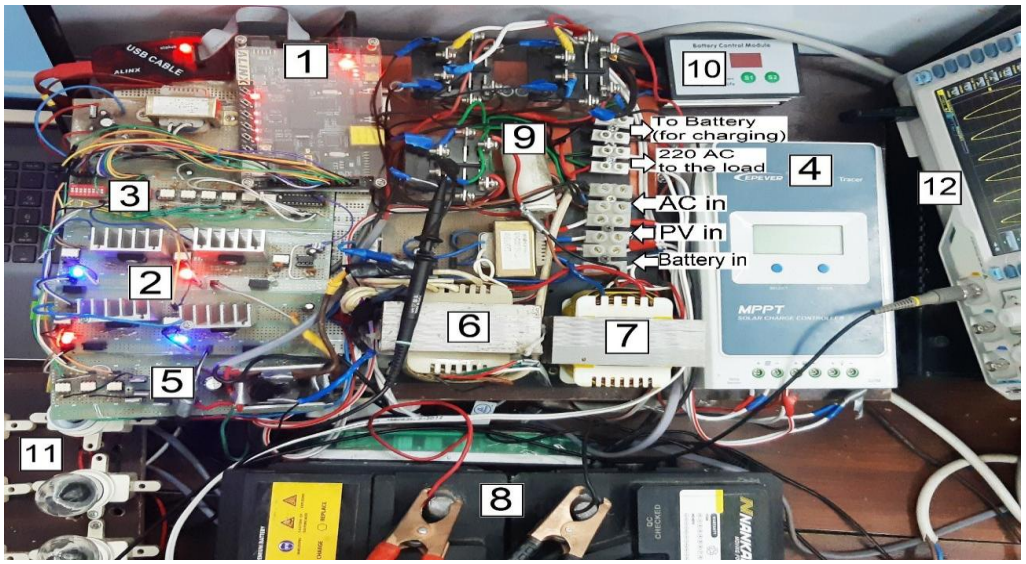


Fig. 13. Hardware design of the proposed system

Table 4
Experimental hardware setup parameters

No.	Component	Specification
1	FPGA	Xilinx Spartan 6 (XC6SLX45)
2	H-Bridge MOSFET circuit	IRFP150N
3	Feedback circuit	A/D converter (ADC0804) and optocouplers
4	MPPT	solar charge controller
5	Full-bridge rectifier diodes	30A
6	Transformer	220V - 24V AC
7	Transformer	24V- 220V AC (rectifier)
8	DC batteries	24V
9	capacitor	2.5 uF
10	Battery charge controller	30A- 24V
11	Resistive loads (lamps)	(60W,120W,180W,240W,300W)
12	Oscilloscope	Rigol, 100MHz bandwidth

4. Results

Xilinx ISE 14.7 Design Suite is the software tool used to implement the design on the FPGA board. The circuit functioning is tested after downloading the design on FPGA and turning on the inverter system, completed successfully. The hardware design generally operated normally without any load, as shown in Figure 14. The output wave was measured using an oscilloscope, as shown in Figure 15. It was pure sinusoidal with a frequency of (50 Hz) and (223v) RMS.

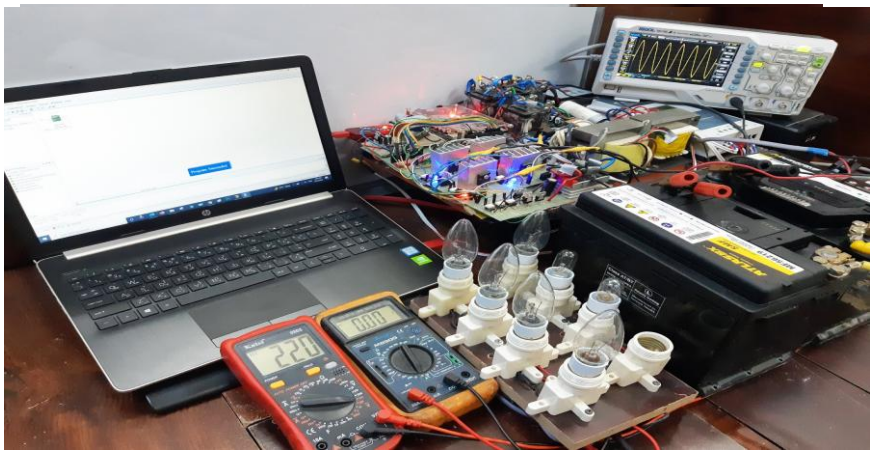


Fig. 14. Implementing the proposed system design without load (f=50Hz, Vrms=220 V)



Fig. 15. Oscilloscope output measurement without load (f=50Hz, Vrms=223 V)

The experimental results by using an oscilloscope and digital multimeters were measured for different loads. Table 5 illustrates a summarization of these values, and Figure 16 compares the output power delivery of the inverter.

Table 5
Experimental power measurement

Nominal load (W)	Without feedback			With feedback		
	Vout (V)	Iout (A)	Power (W)	Vout (V)	Iout (A)	Power (W)
60	205	0.22	45.1	205	0.22	45.1
120	193	0.46	88.78	204	0.47	95.88
180	181	0.68	123.08	202	0.73	147.46
240	173	0.9	155.7	204	0.99	201.96
300	162	1.02	165.24	203	1.14	231.42

Based on Table 5, the output power improvement can be calculated as follows:

$$\triangle Power\% = \frac{|721.82 - 577.9|}{577.9} * 100\% = 24.9\%$$

Where, Sum of the power delivered with feedback = 721.82 W, and the sum of power delivered without feedback equals to 577.9 W.

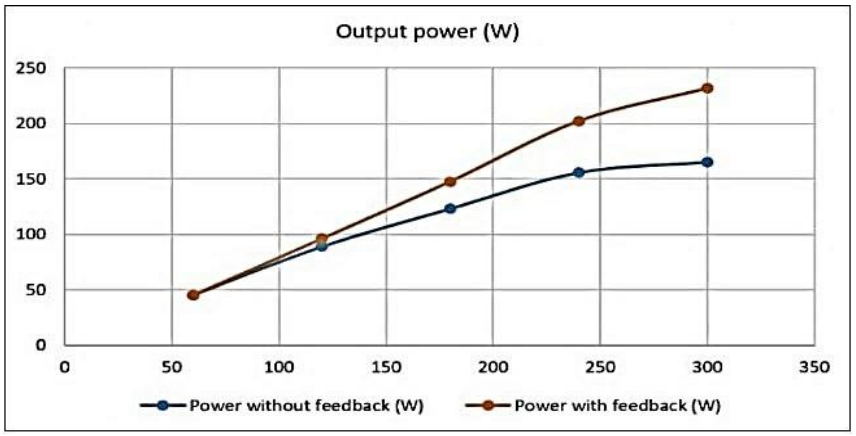


Fig. 16. Power comparison curve with and without ANN feedback system

It shows that the solar inverter with ANN feedback can increase power generation by 24.9% higher than that produced without ANN feedback. The inverter's efficiency was calculated at a maximum load (300 W) according to the following equation [23].

$$Efficiency\ (\eta) = \frac{P_{AC}}{P_{DC}} * 100\% \quad (10)$$

Where, P_{AC} is the AC output power of the inverter, and P_{DC} is the DC input power of the inverter. The inverter's efficiency is calculated for a maximum resistive load of 300 W at about 91.95%.

5. Conclusions

In this research, an intelligent SPWM inverter for solar cells based on FPGA with a feedback system is designed to perform a DC/AC inverter and maintain the output power on different loads. Using the ISE 14.7 design suite, the complete design was implemented on the FPGA board. The inverter is supplied by a low DC supply voltage from solar panels and batteries to generate a sinusoidal AC voltage to the load with lower harmonics content and higher efficiency. The experimental results show that the THD can be reduced by increasing the modulation index, switching frequency, and setting a proper LC low pass filter. The Artificial Neural Network (ANN) is highly efficient in terms of feedback response in comparison to other methods, due to its high accuracy, high-speed response, and lowest training time since the best performance response time performed at MSE equals (1.89×10^{-26}). Implementing the ANN feedback and PWM controller of the full-bridge MOSFETs is utilized by FPGA for its high-speed, better performance, and less hardware usage. Increasing the modulation index will increase the power delivered to the output of the inverter by increasing the AC voltage amplitude and current. ANN feedback increases the output power efficiency by 24.9%, and the efficiency of the inverter at a full load of 300W increased to 91.95%. Experimental results from the oscilloscope showed smooth sinusoidal, less harmonic, with desired

frequency (50Hz) and RMS voltage (220V), which were the same as simulation results in MATLAB/SIMULINK.

Acknowledgment

This research would like to thank the support of the Department of Electrical Engineering, University of Technology, Baghdad, and the Iraqi Ministry of Higher Education and Scientific Research.

References

[1] Asumadu-Sarkodie, Samuel, and Phebe Asantewaa Owusu. "The relationship between carbon dioxide emissions, electricity production and consumption in Ghana." *Energy Sources, Part B: Economics, Planning, and Policy* 12, no. 6 (2017): 547-558. https://doi.org/10.1080/15567249.2016.1227885.

[2] Luo, Fang Lin, and Hong Ye. *Advanced DC/AC inverters: applications in renewable energy*. Crc Press, 2017. https://doi.org/10.1201/b13750

[3] Aboadla, Ezzidin Hassan Elmabrouk, et al. "Effect of modulation index of pulse width modulation inverter on Total Harmonic Distortion for Sinusoidal." 2016 International Conference on Intelligent Systems Engineering (ICISE). IEEE, 2016. https://doi.org/10.1109/INTELSE.2016.7475119

[4] Cilimkovic, Mirza. "Neural networks and back propagation algorithm." *Institute of Technology Blanchardstown, Blanchardstown Road North Dublin* 15, no. 1 (2015).

[5] Hauck, Scott, and Andre DeHon. *Reconfigurable computing: the theory and practice of FPGA-based computation*. Elsevier, 2010.

[6] Ahmed, Oday A., Hussain K. Sayed, Kanaan A. Jalal, Dhari Y. Mahmood, and Waleed H. Habeeb. "Design and implementation of an indoor solar emulator based low-cost autonomous data logger for PV system monitoring." *International Journal of Power Electronics and Drive Systems* 10, no. 3 (2019): 1645. https://doi.org/10.11591/ijpeds.v10.i3.pp1645-1654

[7] Rodriguez-Andina, Juan J., Maria J. Moure, and Maria D. Valdes. "Features, design tools, and application domains of FPGAs." *IEEE Transactions on Industrial Electronics* 54, no. 4 (2007): 1810-1823 https://doi.org/10.1109/TIE.2007.898279

[8] Joost, Ralf, and Ralf Salomon. "Advantages of FPGA-based multiprocessor systems in industrial applications." In *31st Annual Conference of IEEE Industrial Electronics Society, 2005. IECON 2005.*, pp. 6-pp. IEEE, 2005. https://doi.org/10.1109/IECON.2005.1568946

[9] Hassaine, Linda, and Mohamed Rida Bengourina. "Control technique for single phase inverter photovoltaic system connected to the grid." *Energy Reports* 6 (2020): 200-208. https://doi.org/10.1016/j.egyr.2019.10.038

[10] Sarker, Rishiraj, Asim Datta, and Sudipta Debnath. "FPGA-based variable modulation-indexed-SPWM generator architecture for constant-output-voltage inverter applications." *Microprocessors and Microsystems* 77 (2020): 103123. https://doi.org/10.1016/j.micpro.2020.103123

[11] Ali, Ahmed Ismail M., Mahmoud A. Sayed, Takaharu Takeshita, Alaaeldien MM Hassan, and Ahmed M. Azmy. "A single-phase modular multilevel inverter based on controlled DC-cells under two SPWM techniques for renewable energy applications." *International Transactions on Electrical Energy Systems* 31, no. 1 (2021): e12599. https://doi.org/10.1002/2050-7038.12599

[12] Ahmed, Nasim, and Ziaur Rahman Khan. "Microcontroller based pure sine wave inverter." In *2021 IEEE International Conference in Power Engineering Application (ICPEA)*, pp. 173-177. IEEE, 2021. https://doi.org/10.1109/ICPEA51500.2021.9417841

[13] Syahputra, Ramadoni, Eman Sulaeman, Kunnu Purwanto, Yudhi Ardiyanto, Agus Jamal, and Sudarisman Sudarisman. "Design of a single-phase power inverter with voltage controller using IC SG3525." In *AIP Conference Proceedings*, vol. 2499, no. 1. AIP Publishing, 2022. https://doi.org/10.1063/5.0105933

[14] Chacko, Geethu, Lakshmi Syamala, Nithin James, Bos Mathew Jos, and Mathew Kallarackal. "Switching frequency limited hysteresis based voltage mode control of single-phase voltage source inverters." *Energies* 16, no. 2 (2023): 783. https://doi.org/10.3390/en16020783

[15] Algaddafi, Ali, Khalifa Elnaddab, Abdullah Al Ma'mari, and Abdelrahim Nasser Esgiar. "Comparing the performance of bipolar and unipolar switching frequency to drive DC-AC Inverter." In *2016 International Renewable and Sustainable Energy Conference (IRSEC)*, pp. 680-685. IEEE, 2016. https://doi.org/10.1109/IRSEC.2016.7984067

[16] Rusdi, Muhamad, Faizal Arya Samman, and Rhiza S. Sadjad. "FPGA-based electronic pulse generator for single-phase DC/AC inverter." In *2019 International Conference on Information and Communications Technology (ICOIACT)*, pp. 756-760. IEEE, 2019. https://doi.org/10.1109/ICOIACT46704.2019.8938571

[17] Bakar, Afarulrazi Abu, Md Zarafi Ahmad, and Farah Salwani Abdullah. "Design of FPGA based SPWM Single phase inverter." In *Malaysian Technical Universities Conference on Engineering and Technology MUCEET2009*, pp. 20-22. MS, Garden, 2009.

[18] Sayed, Khairy, Abdulaziz Almutairi, Naif Albagami, Omar Alrumayh, Ahmed G. Abo-Khalil, and Hedra Saleeb. "A review of DC-AC converters for electric vehicle applications." *Energies* 15, no. 3 (2022): 1241. https://doi.org/10.3390/en15031241

[19] Le, Tien-Thinh. "Surrogate neural network model for prediction of load-bearing capacity of CFSS members considering loading eccentricity." *Applied Sciences* 10, no. 10 (2020): 3452. https://doi.org/10.3390/app10103452

[20] Mahmood, Dhari Y., Mohammed Kdair Abd, and Kanaan A. Jalal. "Wind Energy Output Prediction Model Based on DPSO-BP Neural Network." *International Journal of Intelligent Engineering & Systems* 14, no. 1 (2021): 242-254. https://doi.org/10.22266/ijies2021.0630.21

[21] Hussein, Areeg F., and Hanan AR Akkar. "Intelligent controller Design based on wind-solar system." *Eng. Technol. J* 39 (2021): 326-337. https://doi.org/10.30684/etj.v39i2A.1761

[22] Chander, A. Hema, and Lalit Kumar. "Multiple input converter for Standalone photovoltaic applications." In *2017 14th IEEE India Council International Conference (INDICON)*, pp. 1-6. IEEE, 2017. https://doi.org/10.1109/INDICON.2017.8488110

[23] Rashid, Muhammad Harunur, ed. *Power electronics handbook: devices, circuits, and applications handbook*. Butterworth-Heinemann, 2018.