Design of 21-level LUO Progression based Multi-Level Inverter using Cross-compiling

R. Nithya
Research Scholar, Department of Electronics and Instrumentation Engineering, Annamalai University, Chidambaram, Tamil Nadu, India. email: nithiazure@gmail.com

Dr. T. Anitha
Associate Professor, Department of Electronics and Instrumentation Engineering, Annamalai University, Chidambaram, Tamil Nadu, India.

Dr. Joseph Anthony Prathap
Associate Professor, Department of Electronics and Communication Engineering, School of Engineering, Presidency University, Bengaluru, India

Abstract
This paper proposes the design of a 21-level LUO progression-based inverter. The driving switch patterns are generated using the VHDL code and cross-compiled in the MATLAB SIMULINK tool. Though there are several inverter topologies, the LUO progression-based inverter consists of less number of switches and low design complexity. The proposed method includes the utilization of carrier-free pulse width modulation signals for the switches to generate the 21-level Multi-Level Inverter. The carrier-free modulation technique is referred to as Switching Angle Method. The performance of the proposed method is evaluated for THD%, VPEAK, and VRMS. The proposed method is developed with the Integrated Circuit layout and its analysis using the Cadence tool.

Keywords: Switching Angle Method, Multi-Level Inverter, Hardware Description Language, Cross Compiling.

I. Introduction
The inverters are power circuits that are used in the transformation of DC to AC signals. The AC output generation of the power converter depends on the switching strategy and the topology of the circuitry used. In recent times, the inverter evolution has been predominantly based on the number of sources, the number of power switches, cascading of circuits, hybridization of DC values (unequal DC sources), and PWM generation schemes.

Among these factors, the inverter with an unequal DC source and less number of switches has a %THD of 2.03 and an efficiency of 95.2% in real-time implementation [1]. The influence of the hybrid DC sources achieves more levels of the inverter compared to the conventional equal DC sources. Also, the evaluation time for the inverters significantly varies based on the modeling methods [2]. The power inverter performance is evaluated based on the THD% of the AC output obtained. The regular carrier-based PWM generation is unsatisfactory with the efficiency of the inverter. The carrier-based PWM generation induces spike error in the AC output affecting the THD of the inverter. Hence, the alternative non-carrier based PWM generation that includes the events at which the change of angles for the inverter are defined by the trigonometric formulation The THD is as low as 1% with the inverter efficiency of
97.55% by making use of the angles evolution technique for switching events [3].

In most applications, the power inverters are controlled for the load variation that occurs due to the non-linearity of the real environment. The power inverters demand many steady-state responses to acknowledge the demands of industries. The power converters are regulated to the reference values under steady-state and dynamic load variation for the control of nanogrid systems [4]. The reliability of the inverter with multi-objective decision-making has good dynamic compensation under droop control [5].

Now with the advent progression formula for the definition of the DC values in inverter circuits, the levels of the inverter are improvised with a less number of switches and components. The inverter is superior in terms of cost, power, and efficiency using the progression-based technique [6].

The advent of digital controller has provided opportunity in controlling the inverter switches accurately. The control of inverter switches in real time is challenging and by making use of the Digital control, the switches are precisely ON and OFF as per the switch pattern requirement of the LUO progression inverter. Among the digital controller, the FPGA is suitable for features namely parallel processing, programmability, high performance and accuracy. The VHDL code is developed for the DPWM generation and its control of the DC-DC voltage regulator and implemented in the FPGA for real-time validation [7]. The FPGA-based control for the PV-fed DC-DC-AC converter gives accuracy and low THD% in the implementation of 81-level MLI [8].

This paper proposes the Luo progression-based inverter using the non-carrier switching angle method for the generation of the 21-level inverter. The switch patterns using the non-carrier-based switching angles technique are derived and converted digitally with the resolution of 8 bits to develop the VHDL code for the 21-level LUO progression-based inverter. The developed VHDL code for the switch pattern of the 21-level LUO progression-based inverter is validated by the cross-compiling in the MATLAB SIMULINK environment.

II. The Proposed Method: LUO PROGRESSION BASED INVERTER

The proposed method includes the generation of the Digital Pulse Width Modulation signals to drive the Luo progression-based inverter. This method uses the non-carrier event angle estimation algorithm to change the levels of the progression inverter. The events generated by the angles calculated are based on the Half Height switching angle method that could generate the angles in sequence till the half and in alternate sequence after the half of the positive and negative cycles.

\[
\alpha_i = \sin^{-1}\left[\left(i - \frac{1}{2}\right)\frac{2}{m-1}\right]
\]  

Where \(i = 1, 2, \ldots, \frac{m-1}{2}\)

The angle of changing events is calculated as per the formulation given in (1) and converted to digital values with a resolution of 28 bits. The x-axis angles from 0 to 360 of a cycle are converted as 0 to 255 bits in digital format. The LUO progression-based inverter consists of two parts of the voltage switches namely DC voltage ladder switches and polarity switches as shown in fig.1. The DC voltage ladder switches combine the DC voltages through the switches as required for the level in the inverter design. The LUO Progression based inverter utilizes the formula as depicted below in (2) & (3)

\[
V_i = i, \quad \text{if } i \leq 2;
\]

\[
= 7 \times 3^{i-2}, \text{if } i \geq 3
\]

Where \(V_i\) is the input voltages of the Luo Progression based Inverter.
In this work, the DC voltages are considered as $V_0 = 1V; V_1 = 2V$ and $V_2 = 7V$. The switches at the respective DC voltages are fed with the patterns to activate the 21-levels of the inverter. The proposed inverter controls 4 switches at a given time to generate the levels. The switch control for the LUO progression-based inverter is based on Table 1 which combines the 10 positive and negative levels and zero level of the 21-level MLI. The VHDL code is developed for the generation of these switches patterns in the mixed style of coding. The 28 counter is utilized to drive the switch patterns for a cycle of MLI output. By using the Conditional-if block, the ON-OFF patterns for the switches are defined precisely in the code.

![Fig.1 Inverter topology for the proposed 21-level LUO progression based inverter](image)

### Table 1 Voltage combinations for the positive and negative levels for the proposed 21-level LUO progression based inverter

<table>
<thead>
<tr>
<th>Level</th>
<th>Positive Voltage Combination</th>
<th>Level</th>
<th>Negative Voltage Combination</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>+1+2+7</td>
<td>-10</td>
<td>-1-2-7</td>
</tr>
<tr>
<td>9</td>
<td>+2+7</td>
<td>-9</td>
<td>-2-7</td>
</tr>
<tr>
<td>8</td>
<td>+1+7</td>
<td>-8</td>
<td>-1-7</td>
</tr>
<tr>
<td>7</td>
<td>+7</td>
<td>-7</td>
<td>-7</td>
</tr>
<tr>
<td>6</td>
<td>-1+7</td>
<td>-6</td>
<td>+1-7</td>
</tr>
<tr>
<td>5</td>
<td>-2+7</td>
<td>-5</td>
<td>+2-7</td>
</tr>
</tbody>
</table>

### III. RESULTS AND DISCUSSION

The proposed 21-level LUO progression-based Inverter is developed using the VHDL code to generate the DPWM signals. The events of the DPWM driving signals are derived from the digitized version of the SAM algorithm to ON-OFF according to the respective 21 levels of the inverter. Fig.2 shows the DPWM signals for the 21-level LUO Progression-based inverter using the ModelSim Tool.

For the sake of validation for the generated DPWM signals of the inverter, the VHDL code is cross-compiled with the MATLAB SIMULINK tool. The switch patterns for the inverter in the MATLAB scope window are given in Fig.3. The switching patterns are fed into the black box configured with the MATLAB SIMULINK. The switching patterns for the 6 switches control the Luo Progression DC voltages of 1V, 2V, and 7V to generate the AC response for the proposed 21-level LUO progression-based inverter as depicted in Fig.4. The FFT analysis is performed on the obtained AC levels to evaluate the %THD, VPeak, and VRMS for the 21-level inverter. The parametric values for the 21-level inverters is 7.30% as %THD, 9.619V as VPeak and 6.802V as VRMS as depicted in Fig. 5(a) and 5(b). The developed HDL code for the 21-level LUO progression-based Inverter is utilized to derive the IC layout using the Cadence Tool. Fig. 6 depicts the RTL schematic of the proposed method in the GENUS cadence tool. The synthesizable code for the proposed method is graphically presented in the RTL schematic. Fig. 7 shows the clock tree synthesis for the proposed method the gives the clock path from the clock source to the clock terminal using the clock tree debugger tool in cadence. The IC layout is developed using the INNOVUS tool of the Cadence to prove that the proposed
Design of 21-level LUO Progression based Multi-Level Inverter using Cross-compiling

method could be fabricated into tap IC. Fig. 8 shows the IC layout for the proposed 21-level LUO progression-based Inverter. The power analysis is derived for the proposed method that gives the Internal power as 60.9509%, Switching power as 39.0448% and the total leakage power of 0.0043% as depicted in the Table 2. The power consumption in the cadence analysis is very low and proves to be advantages for real time implementation.

Fig. 2 Switch pattern for the proposed 21-level LUO progression based inverter in ModelSim Software

Fig. 3 Switch pattern for the proposed 21-level LUO progression based inverter in MATLAB SIMULINK Software
Fig. 4 AC output response for the proposed 21-level LUO progression based inverter

Fig. 5 (a) %THD and (b) parametric evaluation for the proposed 21-level LUO progression based inverter

Fig. 6 RTL schematic of the proposed 21-level LUO progression based inverter using the GENUS cadence Tool
Design of 21-level LUO Progression based Multi-Level Inverter using Cross-compiling

Fig. 7 Clock Tree Debugger for the proposed 21-level LUO progression based inverter using the cadence Tool

Fig. 8 IC Layout for the proposed 21-level LUO progression based inverter using cadence Tool
Table 2 Power Analysis for the proposed 21-level LUO progression based inverter using cadence Tool

<table>
<thead>
<tr>
<th></th>
<th>Internal Power</th>
<th>Switching Power</th>
<th>Leakage Power</th>
<th>Total Power</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Internal Power:</td>
<td>0.57666688</td>
<td>0.1278</td>
<td>1.245e-05</td>
<td>0.4742</td>
<td>49.93</td>
</tr>
<tr>
<td>Total Switching Power:</td>
<td>0.37679287</td>
<td>0.0448%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total Leakage Power:</td>
<td>0.60064661</td>
<td>0.0043%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total Power:</td>
<td>0.94966037</td>
<td></td>
<td></td>
<td></td>
<td>100</td>
</tr>
</tbody>
</table>

IV. CONCLUSION

The proposed 21-level LUO Progression-based inverter has been verified using the cross-compiled MATLAB SIMULINK Model. The VHDL code is developed for the 6 switch patterns of the 21-level LUO inverter and validates with the lower %THD and higher VPeak voltage of 9.619V. Also the power consumption for the proposed method using the cadence tool is low and IC layout can be fabricated in real time. The proposed LUO progression-based inverter could be developed for higher levels and verified experimentally using the FPGA device.

References


for photovoltaic voltage regulator system using FPGA", Microprocessors and Microsystems, vol. 65, pp:107-120, 2019, 
doi:https://doi.org/10.1016/j.micpro.2019.01.004.

doi:https://doi.org/10.1016/j.compeleceng.2022.107738