HARDWARE-RESOURCE SAVING FOR REALIZATION OF SPACE VECTOR PWM BASED ON FPGA USING BUS-CLAMPING TECHNIQUE

Tole Sutikno
Department of Electrical Engineering, Faculty of Industrial Technology
Universitas Ahmad Dahlan (UAD), Yogyakarta
e-mail: tole@ee.uad.ac.id

Abstract
The space vector pulse width modulation (SV-PWM) is more suitable and can increase the obtainable DC voltage utilization ratio very much compared to others PWM. Moreover, the modulation can obtain a better voltage total harmonic distortion (THD) factor. But until now, no studies that concern at hardware resources saving to realize SV-PWM based on FPGA. This paper proposes a new technique to realize SV-PWM based on FPGA. In order to get hardware resource saving, a simple technique to judge sectors, to calculate the firing pulses and to generate SV-PWM waveform without calculation of trigonometric function using bus-clamping technique is proposed. The technique has been implemented successfully based on APEX20KE FPGA to drive three phase induction machine 1.5 kW with low ripples in current and voltage, and has been shown that the proposed SVM method required the most minimum hardware resources compared to others research.

Keywords: bus-clamping, FPGA, SV-PWM, hardware-resource saving

1. INTRODUCTION
The main aim of any modulation technique is to obtain variable output having a maximum fundamental component with minimum harmonics and less switching losses. The Space Vector Pulse Width Modulation (SV-PWM) method is an advanced PWM method and it is possibly the best among all the PWM techniques for variable frequency drive applications. Compared to the Sinusoidal Pulse Width Modulation (SPWM), SV-PWM is more suitable and can increase the obtainable DC voltage utilization ratio very much. Moreover, it can obtain a better voltage total harmonic distortion factor [1-13].

In most engineering practice, the SV-PWM algorithm is mainly implemented with software based on microcontroller [2, 13-18] or digital signal processors (DSP) [19-21] are widely adopted. They perform control procedure sequentially by exploiting their mathematically
oriented resources. That is the instructions of different procedures are executed one after the other. Thus, the purely software-based technique is not an ideal solution. Differ to software implementation; FPGA performs the entire procedures with concurrent operation by using its reconfigurable hardware. A FPGA is considered as an appropriate solution to boost system performance of a digital controller including an SV-PWM algorithm [2, 6, 13, 22-26].

However, the conventional SV-PWM suffers from the drawbacks like computational burden, inferior performance at high modulation indices and high switching losses of the inverter. Hence to reduce the switching losses and to improve the performance in high modulation region, several bus-clamping SV-PWM methods have been proposed [5, 21, 27-31]. Unfortunately, those are based on the conventional SV-PWM without considering hardware resource saving. This paper present the design and implementation of bus-clamping SV-PWM based on FPGA using new technique. To get hardware resource saving, the simplification of sector identification method, re-arrange the dwelling time to avoid the complex trigonometric calculations, and a novel method to calculate the duration of active vector are proposed.

2. A NEW APPROACH TO IMPLEMENT 5-SEGMENT BUS-CLAMPING SV-PWM ALGORITHM

This section present symmetrical 5-segment bus-clamping switching sequence with new judge method of sectors, and new SV-PWM generating method based on calculation of the duration of active vectors to avoid complicated computations with trigonometric function.

2.1. Proposed SV-PWM switching Pattern (5-segment bus-clamping switching pulses)

There has been reported many bus-clamping SV-PWM pattern [5, 6, 26-28]. However, not all those patterns have lower switching losses, simpler algorithm and can be implemented based on FPGA easily. In this paper, a novel symmetric 5-segment bus-clamping SV-PWM design, which it is always a channel staying constant for the entire PWM period is proposed.

2.2. Proposed identification of the sector

The methods to judge the sector have been introduced which the reference space voltage vector lies in. Zhi-pu [32] has compared the reference space vector’s angle with 0°, 60°, 120°, 180°, 240°, and 300° to obtain the number of the sector that the $V_{ref}$ in. The others, Yu [33], Jiang [23] and Xing [7] have analyzed the relationship between $V_a$ and $V_\beta$ to determine the sector. They have calculated the projections $V_a$, $V_b$ and $V_c$ of $V_a$ and $V_\beta$ in (a,b,c) plane by using inverse Clark transformation, as follow:

$$\begin{align*}
V_a &= V_\beta \\
V_b &= \frac{-\sqrt{3}V_a - V_\beta}{2} \\
V_c &= \frac{-\sqrt{3}V_a - V_\beta}{2}
\end{align*}$$

(1)

Then, based on equation (4) above, they calculate $N=\text{sign}(V_a) + 2*\text{sign}(V_b) + 4*\text{sign}(V_c)$. Map $N$ to the actual sector of the output voltage reference by referring to the following relationship:

<table>
<thead>
<tr>
<th>N</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>sector</td>
<td>2</td>
<td>6</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>5</td>
</tr>
</tbody>
</table>

In [34], Zeliang has adopted the new intermediate vectors $X_a$ and $X_\beta$ that he has defined $X_a = \frac{3}{2}V_a$ and $X_\beta = \sqrt{3}V_\beta$ as decompose of the conventional SV-PWM, which will
properly counteract the redundant calculations to identify sector location, but it imported the complicated matrix calculations. Hence, through the analyzing of SV-PWM mentioned above, this research has created a simpler method to determine the sectors of voltage vectors based on the comparison between $V_\beta, \sqrt{3}V_\alpha, -\sqrt{3}V_\alpha$ and 0 as shown in Fig. 1. By using the comparison, we can determine the sectors of voltage vectors simpler than [7, 23, 32-34].

![Diagram](image)

**Fig. 1. A simpler method to determine the sectors**

### 2.3. The proposed calculating of the duration of active vectors

In this research, through the analyzing of reference [12], a new set of equation to calculate the duration of active vectors for each sector has re-arranged in order to easier to implement based on FPGA. It is shown in Table 2.

<table>
<thead>
<tr>
<th>Sector</th>
<th>$T_a$</th>
<th>$T_b$</th>
<th>$T_a + T_b$</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>$\frac{3T}{4} \left( \frac{V_\alpha - V_\beta}{V_{dc} - \sqrt{3} V_{dc}} \right)$</td>
<td>$\frac{3T}{4} \left( \frac{2V_\beta}{\sqrt{3} V_{dc}} \right)$</td>
<td>$\frac{3T}{4} \left( \frac{V_\alpha + V_\beta}{V_{dc} + \sqrt{3} V_{dc}} \right)$</td>
</tr>
<tr>
<td>II</td>
<td>$\frac{3T}{4} \left( \frac{V_\alpha + V_\beta}{V_{dc} + \sqrt{3} V_{dc}} \right)$</td>
<td>$\frac{3T}{4} \left( -\frac{V_\alpha + V_\beta}{V_{dc} + \sqrt{3} V_{dc}} \right)$</td>
<td>$\frac{3T}{4} \left( \frac{2V_\beta}{\sqrt{3} V_{dc}} \right)$</td>
</tr>
<tr>
<td>III</td>
<td>$\frac{3T}{4} \left( \frac{2V_\beta}{\sqrt{3} V_{dc}} \right)$</td>
<td>$\frac{3T}{4} \left( -\frac{V_\alpha - V_\beta}{V_{dc} - \sqrt{3} V_{dc}} \right)$</td>
<td>$\frac{3T}{4} \left( -\frac{V_\alpha + V_\beta}{V_{dc} + \sqrt{3} V_{dc}} \right)$</td>
</tr>
<tr>
<td>IV</td>
<td>$\frac{3T}{4} \left( -\frac{V_\alpha + V_\beta}{V_{dc} + \sqrt{3} V_{dc}} \right)$</td>
<td>$\frac{3T}{4} \left( \frac{2V_\beta}{\sqrt{3} V_{dc}} \right)$</td>
<td>$\frac{3T}{4} \left( -\frac{V_\alpha - V_\beta}{V_{dc} - \sqrt{3} V_{dc}} \right)$</td>
</tr>
<tr>
<td>V</td>
<td>$\frac{3T}{4} \left( -\frac{V_\alpha - V_\beta}{V_{dc} - \sqrt{3} V_{dc}} \right)$</td>
<td>$\frac{3T}{4} \left( \frac{V_\alpha - V_\beta}{V_{dc} - \sqrt{3} V_{dc}} \right)$</td>
<td>$\frac{3T}{4} \left( -\frac{2V_\beta}{\sqrt{3} V_{dc}} \right)$</td>
</tr>
<tr>
<td>VI</td>
<td>$\frac{3T}{4} \left( -\frac{2V_\beta}{\sqrt{3} V_{dc}} \right)$</td>
<td>$\frac{3T}{4} \left( \frac{V_\alpha - V_\beta}{V_{dc} - \sqrt{3} V_{dc}} \right)$</td>
<td>$\frac{3T}{4} \left( \frac{V_\alpha - V_\beta}{V_{dc} - \sqrt{3} V_{dc}} \right)$</td>
</tr>
</tbody>
</table>
2.4. Proposed method to generate SV-PWM switching pulses

The bus-clamping 5-segment bus-clamping SV-PWM has symmetrical switching pulses as in the 7-segment conventional SV-PWM. Therefore, the similar method to generate conventional SV-PWM switching pulses can be adopted in the proposed SV-PWM. To easier implement it based on FPGA, in this research has created a graphical method to generate the SV-PWM switching pulses as illustrated in Fig. 2.

![Diagram of proposed method to generate set of SV-PWM switching pulses](image)

Fig. 2 The proposed method to generate set of SV-PWM switching pulses

By considering equation \( y = \frac{2m}{T} x \) in Fig. 3, the PWM generating for odd sector were implemented through comparison between triangle and \( T_a \), and between triangle and \( T_a + T_b \) with other switching was set equal to 1; while for even sector, the PWM generating are implemented through complement of comparison between triangle and \( T_a \), and complement of comparison between triangle and \( T_a + T_b \) with other switching is set equal to 0. For example in sector I, if \( x = \frac{T_a}{2} \) then \( y = \frac{2m}{T} \left(\frac{T_a}{2}\right) \) and if \( x = \frac{T_b}{2} \) then \( y = \frac{2m}{T} \left(\frac{T_b}{2}\right) \). For simpler of circuit design, in this paper the term \( \frac{2m}{T} \) is set equal to 1, so if \( x = \frac{T_a}{2} \) then \( y = \frac{T_a}{2} \) and also if \( x = \frac{T_b}{2} \) then \( y = \frac{T_b}{2} \). Obviously, if \( x = \frac{T_a}{2} + \frac{T_b}{2} = T_a + T_b \) then \( y = \frac{T_a}{2} + \frac{T_b}{2} = T_a + T_b \).

Therefore, the PWM generating for \( S_b \) and \( S_c \) channels in sector I can be obtained through comparison between triangle and \( T_a \), and between triangle and \( T_a + T_b \) respectively, with \( S_a \) Channel is set equal to 1. The PWM generating in other sectors can be obtained in a similar way.

3. FPGA REALIZATION OF A PROPOSED NOVEL SV-PWM

In previous section, the principle of SV-PWM has analyzed. In this section, the implementation of proposed SV-PWM based on FPGA will be presented. The overall of
proposed SV-PWM design is shown in Fig. 3. This top module has divided into 5 sub modules, namely adjust_freq, Vbeta_Valfa, find_sector, SVM_generator and deadtime_system module.

3.1. Adjustable Frequency Module
In this module, the source of clock generating is designed to connect L6 pin. In this research, carrier signal frequency is set to 40 kHz and reference signal frequency is set to 50 Hz. To get desired carrier and reference signal frequency, the clock dividing is done.

3.2. Vα and Vβ Module
In this research, Vα and Vβ is generated through look up table (LUT) sine and cosine function with memory mapping 360 addresses. The lower, base, and higher numbers of sine and cosine function are 96, 224 and 352 respectively (in 9 unsigned bits).

3.3. Sector Identification Module
This module is used to determine sector based on Table 2. The simplification of truth table for comparison results as shown in Table 3 were used to determine number of sector in “sector” sub-module.

Table 3. Conversion of comparison result between $\sqrt{3}V_\alpha$, $-\sqrt{3}V_\alpha$ and 0 to number of sector

<table>
<thead>
<tr>
<th>Sector</th>
<th>Vector Angle</th>
<th>Input</th>
<th>Output (S2,S1,S0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>(0°, 60°)</td>
<td>101</td>
<td>001</td>
</tr>
<tr>
<td>II</td>
<td>(60°, 120°)</td>
<td>111</td>
<td>010</td>
</tr>
<tr>
<td>III</td>
<td>(120°, 180°)</td>
<td>110</td>
<td>011</td>
</tr>
<tr>
<td>IV</td>
<td>(180°, 240°)</td>
<td>010</td>
<td>100</td>
</tr>
<tr>
<td>V</td>
<td>(240°, 300°)</td>
<td>000</td>
<td>101</td>
</tr>
<tr>
<td>VI</td>
<td>(300°, 360°)</td>
<td>001</td>
<td>110</td>
</tr>
</tbody>
</table>

3.4. Three phase SV-PWM signal generator module
This module can be divided into 4 sub-modules, namely Triangle, Duration_Ta, Duration_TaTb, and SVM pattern sub-module. Triangle sub-module was used to generate triangle carrier signal. In this research, triangle signal generator was sampled 32 times per period and 9 unsigned bits were used to represent, which lower number (equal to base number of reference signal) and higher number each are 224 and 352 respectively. The Duration_Ta and Duration_TaTb sub-module has created based on digital solution of second and fourth column in Table 3 respectively. Then, SVM pattern sub-module was used to generate set of SV-PWM pulses refer to section 2.4 and Fig. 2 above.
4. SIMULATION AND EXPERIMENT RESULTS

The software to design, compilation, verification and hardware realization based on FPGA APEX20KE in this research has used Quartus II Version 9.0 Web Edition.

4.1. Simulation

The parameters has used in this research are listed as follows: \( V_{dc}/T=1 \), switching frequency 40 kHz, and reference frequency 50 Hz. The compilation report of proposed SV-PWM generator has shown that the design requires 520 logic elements and 9.216 memory bits. If the requirement of hardware resource to generate proposed SV-PWM is compared to other researches, the method requires is most saving as shown in Fig. 4. It has proved that the proposed SVM method required the most minimum hardware resources.

![Fig. 4 The comparison of required hardware resources](image)

4.2. Experiment

The advanced testing of FPGA based proposed SV-PWM generator design has been done to drive a three phase inverter system with induction machine 1.5 kW. The results are shown in Fig. 5. They are each shown stator current output \( (I_a) \), phase-to-phase voltage output \( (V_{ab}) \) and frequency spectrum. The practical results from test-rig were in good agreement to drive induction machine 1.5 kW, with low ripples in current and voltage.

![Fig.5 The performance of proposed SV-PWM generator design based on FPGA.](image)
5. CONCLUSION

This paper has presented the realization of a novel 5-segment bus-clamping SV-PWM based on FPGA at switching frequency 40 kHz, in which the simple judging of sectors, the rearrange calculation method of the firing time and the simple method to generate SV-PWM pulses without complicated computations with trigonometric function have been proposed. The technique has been implemented successfully based on APEX20KE FPGA to drive three phase induction machine 1.5 kW with low ripples in current and voltage, and has been shown that the proposed SVM method required the most minimum hardware resources compared to others research.

REFERENCES


