FPGA implementation of an efficient space vector pulse width modulation algorithm

R. Subha, Nitin Kumar, Navreet Kaur, Pragathi M.S and Sudhanshu Agrawal
Department of Electrical & Electronics, Sir M Visvesvaraya Institute of Technology, Bangalore.

Introduction
Space Vector Pulse Width Modulation (SVPWM) technique has been widely investigated for decades. It gives higher output voltage for the same DC bus voltage, lower switching losses and better harmonic performance in comparison with the carrier based sine triangle PWM, a simple and widely used scheme which generated PWM output based on the principle of comparing the triangular carrier signal with a sinusoidal reference waveform. Therefore, SVPWM scheme has become the preferred PWM technique for various Three Phase Power Converter applications.

Although the conventional scheme is theoretically simple the implementation is difficult especially with the advance of switching frequency (low sampling period). The complicated procedures will result in long processing time and extensive computational resources. Using this resource efficient SVPWM algorithm, the conventional SVPWM is decomposed into fast integer operations entirely by using an intermediate vector, which will properly counteract the redundant calculations of the remaining procedures. Since it can be implemented without any multiplier or divider, the fast algorithm is especially suitable for field programmable gate array applications. It ensures lower hardware resource usage, and at the same time, operates faster. Also the Field Programmable Gate Arrays (FPGAs) offer high computational ability and flexibility due to their parallel execution and reconfigurable hardware. Hence the scheme suggests the implementation of a resource efficient algorithm for SVPWM generation using FPGA and obtains higher sampling rates with minimal use of hardware resources. Hence a modified SVPWM algorithm is developed by using intermediate vectors and the transformed mathematical equations, from the conventional SVPWM algorithm. This modified algorithm is then implemented using VHDL, downloaded on XILINX Spartan 2E kit XC3S200E and the waveforms were observed on the oscilloscope.

Keywords
Three Phase voltage, XILINX Spartan, VHDL, SVPWM.
Conventional SVPWM Algorithm

In d, q coordinates the combination of 3 phase inverter output voltage form eight space vectors \((V_k, k=0,1,\ldots,7)\) as shown in figure 2. There are six non-zero space vectors \((V_k, \ k=1,2,\ldots,6)\), forming an origin centered hexagon and two zero space vectors \((V0, V7)\), located at the origin. Space vector PWM can be implemented by the following STEPS:

Step 1. Determine \(V_d, V_q, V_{ref}\), and angle \((\alpha)\)
Step 2. Determine time duration \(T_1, T_2, T_0\)
Step 3. Determine the switching time of each transistor (S1 to S6).

The above set of equations explains the calculation of the line to neutral voltages. The \(V_{ref}\) can be calculated as given below:

\[
V_{ref} = \frac{T_n}{T_z} \cdot V_n + \frac{T_{n+1}}{T_z} \cdot V_{n+1}
\]

where \(T_n \& T_{n+1}\) are on time of \(V_n\) and \(V_{n+1}\) during each sampling period \((T_z)\) respectively and ‘n’ is the sector number in which \(V_{ref}\) resides. The dwell time coefficients can be calculated as given by the above set of equations. The conventional SVPWM algorithm includes d-q transformation of three input voltages \(V_{an}, V_{bn}\) and \(V_{cn}\). This transformation requires value of \(\sqrt{3}\) to be calculated, it is difficult to implement floating point numbers using FPGA.

\[
T_k = \frac{\sqrt{3}T_z}{V_k} \left[ \sin \left( \frac{\pi}{3} k \right) V_\alpha - \cos \left( \frac{\pi}{3} k \right) V_\beta \right]
\]

\[
T_{k+1} = \frac{\sqrt{3}T_z}{V_k} \left[ -\sin \left( \frac{\pi}{3} (k-1) \right) V_\alpha + \cos \left[ \frac{\pi}{3} (k-1) \right] V_\beta \right]
\]

\[
T_0 = T_k - T_{k+1}
\]

because of the irregular coefficients of \(T_k, T_{k+1}, T_0\) complexity of the algorithm increases. Hence a modified version of the SVPWM algorithm has been developed.

The criteria for sector identification in d, q coordinates can be given by:

<table>
<thead>
<tr>
<th>Sector</th>
<th>Vector angle</th>
<th>(V_{\alpha}, V_{\beta}) conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>([0^\circ, 60^\circ])</td>
<td>(0 \leq V_\beta &lt; \sqrt{3}V_\alpha)</td>
</tr>
<tr>
<td>II</td>
<td>([60^\circ, 120^\circ])</td>
<td>(0 \leq \sqrt{3}V_\alpha \leq V_\beta)</td>
</tr>
<tr>
<td>III</td>
<td>([120^\circ, 180^\circ])</td>
<td>(0 &lt; V_\alpha \leq \sqrt{3}V_\beta)</td>
</tr>
<tr>
<td>IV</td>
<td>([180^\circ, 240^\circ])</td>
<td>(\sqrt{3}V_\alpha &lt; V_\beta \leq 0)</td>
</tr>
<tr>
<td>V</td>
<td>([240^\circ, 300^\circ])</td>
<td>(V_\beta \leq \sqrt{3}V_\alpha \leq 0)</td>
</tr>
<tr>
<td>VI</td>
<td>([300^\circ, 360^\circ])</td>
<td>(-\sqrt{3}V_\alpha \leq V_\beta &lt; 0)</td>
</tr>
</tbody>
</table>

Table 1: Criteria of the sector identification in d,q coordinates

Modified SVPWM Algorithm

In the conventional method the procedure of sector identification and switching time calculation are complicated. Hence instead of d-q transformation, intermediate transformation vectors are used.

\[
\begin{align*}
V_d &= \frac{2}{3} V_{an} - \frac{1}{3} V_{bn} - \frac{1}{3} V_{cn} \\
V_q &= \frac{1}{\sqrt{3}} V_{bn} - \frac{1}{\sqrt{3}} V_{cn}
\end{align*}
\]

The modified algorithm adopts new intermediate variables \(X_d\) and \(X_q\) defined as:

\[
\begin{align*}
X_d &= V_{an} - V_{bn}/2 - V_{cn}/2 \\
X_q &= V_{bn} - V_{cn}
\end{align*}
\]

Eqn 1

The above set of equations normalizes the coefficients by \(\sqrt{3}\) so that the calculations are digitally made and are easy to implement. By adding the sign detection of \(X_d\) and \(X_q\), the sector finding criterion is directly related to \(|X_d|\) and \(|X_q|/2\) instead of the relationship between \(\sqrt{3}V_d\) and \(V_q\). More importantly the coefficients \(X_d\) and \(X_q\) will precisely counteract other irregular coefficients of the remaining signal procedures To identify sector location, an improved sector identification can be deal with the simple criterion as listed below instead of the complicated conditions as listed in Table 1.
Table 2: Simplified Criterion of the sector identification

| Sector | Xd>0 | Xq>0 | |Xd|> |Xq|/2 |
|--------|------|------|--------|------------------|
| I      | 1    | 1    | 1      |                  |
| II     | X    | 1    | 0      |                  |
| III    | 0    | 1    | 1      |                  |
| IV     | 0    | X    | 0      |                  |
| V      | X    | 0    | 1      |                  |
| VI     | 1    | 0    | 1      |                  |

Table 2 Symmetrical PWM outputs when reference locates in sector 1

The dwell time calculation can be modified as:

\[ T_{\text{TT}} = T \begin{bmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{bmatrix} \begin{bmatrix} X_d \\ X_q \end{bmatrix} \]

Where, T is the dwell time transformation matrix given by:

\[
T = \begin{bmatrix}
T_{00} & T_{01} \\
T_{10} & T_{11}
\end{bmatrix}
= \begin{bmatrix}
\frac{2}{\sqrt{3}} \sin \left( \frac{\pi}{3} k \right) & -\cos \left( \frac{\pi}{3} k \right) \\
-\frac{2}{\sqrt{3}} \sin \left[ \frac{\pi}{3} (k - 1) \right] & \cos \left[ \frac{\pi}{3} (k - 1) \right]
\end{bmatrix}
\]

The coefficients corresponding to the sectors can be got from the following table:

| Sector | Xd>0 | Xq>0 | |Xd|> |Xq|/2 |
|--------|------|------|--------|------------------|
| I      | 1    | 1    | 1      |                  |
| II     | X    | 1    | 0      |                  |
| III    | 0    | 1    | 1      |                  |
| IV     | 0    | X    | 0      |                  |
| V      | X    | 0    | 1      |                  |
| VI     | 1    | 0    | 1      |                  |

FPGA implementation

Refering to the Fig. 1, there are four main modules as below.

Transformation Module:

Values of three phase voltages i.e. Vab, Vbn, Vcn is input to this block. This module transforms three phase voltage reference frame to two phase voltage reference frame. To implement in FPGA the resources requires are only adders, and shifters.

Sector determination module:

The input to this module is Xd and Xq values generated by three to two phase transformation module. This module checks three conditions as explained earlier and the corresponding sectors get determined as per the rules explained.

- Result of condition Xd>0 is stored in 1 bit variable as C1.
- Result of condition Xq>0 is stored in 1 bit variable as C2.
- And result of condition |Xd|>|Xq/2| is stored as C3.

Switching time calculation module:

Switching time calculation requires values of Xd, Xq and sectors. These values are calculated by two blocks as explained above. The values of parameters T1 and T2 are determined. The coefficients of decomposition matrix and their values are obtained are based on sector number.

Symmetrical PWM generation module:

This module is the heart of this algorithm which is responsible for generation of three symmetrical PWM signals. This module requires sector number and switching time values calculated by previous blocks. The PWM switching frequency can be

Adjusted in this module

By using the above concept a VHDL algorithm is developed so that each block can be determined and digitally implementable. The algorithm was tested with three values of modulation index that is 0.7, 0.5 and 0.3. The simulation was done using Xilinx ISE kit and the simulation results were approximately equivalent to those obtained in the conventional method.

Fig. 4 Compact architecture of the modified SVPWM algorithm

RESULTS
ISE SIMULATOR RESULTS

The simulation result for modulation index 0.7 is shown below:

The simulation results for modulation index 0.5

The simulation results for modulation index 0.3
Hardware setup
Waveforms seen on the oscilloscope pulses for modulation index 0.7
PWMA

PWMB

PWMC

Conclusion
To facilitate the digital implementation of SVPWM scheme for three-phase inverter application, a compact algorithm has been developed in this paper.

By employing two specific intermediate variables, the algorithmic strength is tremendously reduced, because the proposed algorithm can be decomposed into fast integer operations completely without requiring any multiplier or divider. Thus, making the algorithm a lot easier to implement digitally. The use of FPGA (Xilinx Xc2S200E), we see that the resource utilization is minimal and efficient.

Table 3: resource utilization table for 0.7 modulation index
Comparison between conventional and modified scheme

<table>
<thead>
<tr>
<th>Architecture</th>
<th>Multipliers</th>
<th>Dividers</th>
<th>Adders</th>
<th>Utilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conventional scheme</td>
<td>8</td>
<td>1</td>
<td>24</td>
<td>12%</td>
</tr>
<tr>
<td>Modified scheme</td>
<td>0</td>
<td>0</td>
<td>12</td>
<td>5%</td>
</tr>
</tbody>
</table>

From the Table-4 it was observed very less resources were utilized for the implementation of the complicated SVPWM algorithm after the algorithm was modified. Table-5 compares the multipliers, dividers and adders required for conventional algorithm and modified algorithm. The number of multipliers and dividers used in modified algorithm are none as against the conventional algorithm which uses 8 multipliers, 1 divider. The use of adders also is reduced in modified algorithm which utilizes only 12 adders whereas the conventional algorithm uses 24.