Available online at www.elixirpublishers.com (Elixir International Journal)

Power Electronics Engineering

Elixir Power Elec. Engg. 47 (2012) 8710-8714

ISSN: 2229-712X

FPGA based digital space vector controller of voltage source inverter

Bahram Rashidi and Mehran Sabahi
University of Tabriz, Iran.

ARTICLE INFO

Article history:
Received: 24 March 2012;
Received in revised form:
18 May 2012;
Accepted: 4 June 2012;

Keywords
SVPWM,
FPGA,
Voltage Source Inverter.

ABSTRACT
This paper presents a digital method to develop the three phase voltage source inverter control unit using field programmable gate array (FPGA) based digital Space Vector Pulse Width Modulation (SVPWM) algorithm. In this paper the complicate equations of SVPWM algorithm are simplified to effortless instructions such as shift, addition and subtraction operands in the proposed digital design. Furthermore low power digital circuit to make FPGA based SVPWM control is implemented. The achieved dynamic power consumption is about 63 mW in FPGA clock frequency of 100 MHz. The proposed digital SVPWM was synthesized and implemented using Xilinx ISE and Virtex IV FPGA, with target device XC4VFX100. Also power is analyzed using XPower analyzer. The simulation results demonstrate that proposed method has reduced power consumption.

Introduction

Pulse width modulation (PWM) is one of the switching techniques widely used to control the output of the inverter especially to overcome to the harmonics problem. It is known that unwanted frequency components can be moved to a higher frequency region by PWM method. Conventional method of generating PWM signal is achieved by using ratio of a high frequency carrier signal and fundamental frequency signal generated via analogue circuit. Advancement in the digital technology enables PWM switching schemes to be generated by means of digital controller (i.e. using microcontrollers). Through this technique, the harmonics content of the output voltage can be minimized and reduced significantly by simply adjusting the switching angles of the pulses using the programming language [1]. Recently, developments in power electronics and semiconductor technology have lead improvements in power electronic systems. Hence, different circuit configurations namely multilevel inverters have become popular and considerable interest are given on them by researchers. Variable voltage and frequency supply to ac drives is invariably obtained from a three-phase voltage source inverter. A number of PWM schemes are used to obtain variable voltage and frequency supply. The most widely used PWM schemes for three-phase voltage source inverters are carrier-based sinusoidal PWM and space vector PWM (SVPWM). There is an increasing trend of using SVPWM because of their easier digital realization and better dc bus utilization [2]. Iimplementation of SVPWM by using the FPGA has been proposed by some researchers [3-5]. In [4] a resource efficient SVPWM algorithm is proposed which reduces computational overheads and solves the problem of high sampling time in real time applications. The proposed algorithm uses only adders, subtractors, comparators, shifters, etc to be implemented using FPGA's and takes less number of resources and execution time than conventional algorithm. In [5] a simple realization of 5-segment discontinuous SVPWM with a difference approach based on FPGA, has been presented in which the judging of sectors and the calculation of the firing time to generate the SVPWM waveform is simple, and also the switching losses is low.

This paper focuses on development and realization a digital SVPWM on FPGA board with optimized digital circuits. In Section II, the model of a three phase voltage source inverter is discussed based on space vector theory. Section III presents the proposed implementation of SVPWM algorithm on FPGA with comparison of proposed implementation with some of the previous works.

SVPWM Algorithm

The basic 3-phase inverter structure is shown in Fig. 1.

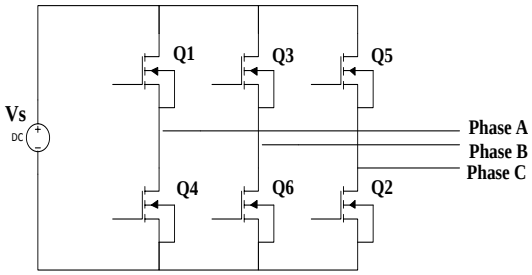

Fig. 1: Full bridge 3-phase inverter.

A different approach to SPWM, is based on the space vector representation of voltages in the d and q plane. The dq components are found by Park transform, where the total power, as well as the impedance, remains unchanged. The transformation is equivalent to an orthogonal projection of a-b-c reference frame onto two dimensional perpendiculars to the equivalent d-q plane in a three dimensional co-ordinate system. As a result six nonzero and two zero vectors are obtained. The six non zero vectors are shaped to hexagonal as shown in Fig. 2. The angle between any two non zero adjacent vectors is 60°. The objective of SVPWM is to approximate reference voltage

Tele:
E-mail addresses: b_rashidi86@yahoo.com

vector V_{ref} using eight switching pattern. Therefore, space vector PWM can be implemented by following three steps:

- Determining V_d, V_q, V_{ref} and angle α for sector
- Determining the time duration T1, T2 and T0
- Determining the switching time of each switching devices (S1 ~ S6)

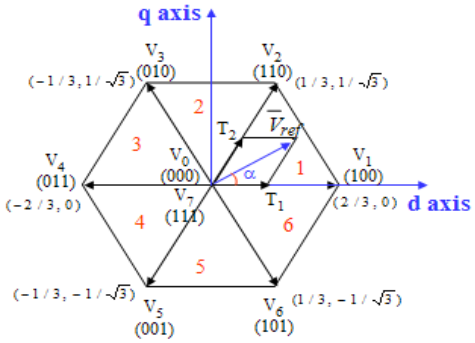

Fig. 2: Basic switching vectors and sectors

The relationship between switching variable vector and phase voltage vector [V_a V_b V_c] can be expressed as follows:

$$V_d = V_{an} - V_{bn}\cdot\cos 60 - V_{cn}\cdot\cos 60$$
$$= V_{an} - \frac{1}{2}V_{bn} - \frac{1}{2}V_{cn}$$

$$V_q = 0 + V_{bn}\cdot\cos 30 - V_{cn}\cdot\cos 30$$
$$= \frac{\sqrt{3}}{2}V_{bn} - \frac{\sqrt{3}}{2}V_{cn}$$

$$\therefore \begin{bmatrix} V_d \\ V_q \end{bmatrix} = \frac{2}{3}\begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}\begin{bmatrix} V_{an} \\ V_{bn} \\ V_{cn} \end{bmatrix} \quad (1)$$

Fig. 3 shows reference vector as a combination of adjacent vectors at sector 1.

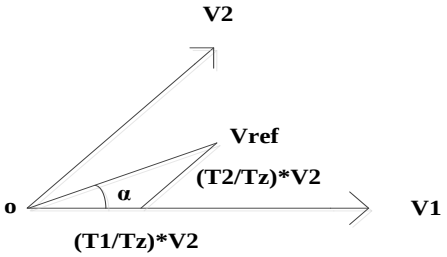

Fig. 3: Reference vector as a combination of adjacent vectors at sector 1

The reference vector V_{ref} is calculated as follows [4]:

$$|\bar{V}_{ref}| = \sqrt{V_d^{\,2} + V_q^{\,2}}$$
$$\alpha = \tan^{-1}\left(\frac{V_q}{V_d}\right) = \omega_s t = 2\pi f_s t \quad (2)$$

Where, f_s is the fundamental frequency. In general, V_{ref} in 'n' sector is obtained by two adjacent non-zero vectors (V_j, V_{j+1}) and two-zero vectors. V_{ref} is expressed as:

$$V_{ref} = \frac{T_n}{T_Z}.V_n + \frac{T_{n+1}}{T_Z}V_{n+1} \quad (3)$$

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. Fig 3 (b) shows the space vector time representation with V_{ref} for sector 1.

The magnitude of non zero space vector is always $||V_j|| = 2V_{dc}/3$. The sectors are identified depending on values of angle 'α'. The switching time duration T_1, T_2 and T_0 for a particular sector can be calculated from following equations:

$$\int_0^{T_z}\bar{V}_{ref} = \int_0^{T_1}\bar{V}_1 dt + \int_{T1}^{T_1+T_2}\bar{V}_2 dt + \int_{T_1+T_2}^{T_z}\bar{V}_0$$
$$\therefore T_z\cdot\bar{V}_{ref} = (T_1\cdot\bar{V}_1 + T_2\cdot\bar{V}_2)$$
$$\Rightarrow T_z\cdot|\bar{V}_{ref}|\cdot\begin{bmatrix}\cos(\alpha)\\ \sin(\alpha)\end{bmatrix} = T_1\cdot\frac{2}{3}\cdot V_{dc}\cdot\begin{bmatrix}1\\0\end{bmatrix} + T_2\cdot\frac{2}{3}\cdot V_{dc}\cdot\begin{bmatrix}\cos(\pi/3)\\ \sin(\pi/3)\end{bmatrix}\backslash$$
$$\therefore T_1 = T_z\cdot a\cdot\frac{\sin(\pi/3-\alpha)}{\sin(\pi/3)} \quad ; \quad \therefore T_2 = T_z\cdot a\cdot\frac{\sin(\alpha)}{\sin(\pi/3)}$$
$$\therefore T_0 = T_z - (T_1 + T_2),$$
$$\text{where,} \quad T_z = \frac{1}{f_s} \quad \text{and} \quad a = \frac{|\bar{V}_{ref}|}{\frac{2}{3}V_{dc}} \quad (4)$$

The switching time of each switching devices (S_1~ S_6) per sector is given by Table I

Table I: Switching time at each sector

Sector	S_1, S_3, S_5	S_2, S_4, S_6
1	$S_1 = T_1 + T_2 + T_0/2$ $S_3 = T_2 + T_0/2$ $S_5 = T_0/2$	$S_4 = T_0/2$ $S_6 = T_1 + T_0/2$ $S_2 = T_1 + T_2 + T_0/2$
2	$S_1 = T_1 + T_0/2$ $S_3 = T_1 + T_2 + T_0/2$ $S_5 = T_0/2$	$S_4 = T_2 + T_0/2$ $S_6 = T_0/2$ $S_2 = T_1 + T_2 + T_0/2$
3	$S1 = T0/2$ $S3 = T1 + T2 + T0/2$ $S5 = T1 + T2$	$S_4 = T_1 + T_2 + T_0/2$ $S_6 = T_0/2$ $S_2 = T_1 + T_0/2$
4	$S_1 = T_0/2$ $S_3 = T_1 + T_0/2$ $S_5 = T_1 + T_2 + T_0/2$	$S_4 = T_1 + T_2 + T_0/2$ $S_6 = T_2 + T_0/2$ $S_2 = T_0/2$
5	$S_1 = T_2 + T_0/2$ $S_3 = T_0/2$ $S_5 = T_1 + T_2 + T_0/2$	$S_4 = T_1 + T_0/2$ $S_6 = T_1 + T_2 + T_0/2$ $S_2 = T_0/2$
6	$S_1 = T_1 + T_2 + T_0/2$ $S_3 = T_0/2$ $S_5 = T_1 + T_0/2$	$S_4 = T_0/2$ $S_6 = T_1 + T_2 + T_0/2$ $S_2 = T_2 + T_0/2$

Implementation of The Proposed SVPWM Algorithm On FPGA

Today digital designers use hardware description languages (HDLs) to design digital systems. The most widely used HDLs are VHDL and Verilog.. The program can be used to both simulate the operation of the circuit and synthesize an actual implementation of the circuit in a CPLD, an FPGA, or an application specific integrated circuit (ASIC). In this paper we use the hardware description language VHDL for designing and implementing SVPWM based on combinational logic circuits with low resource utilization and for reducing the execution time. The simplified mathematical equations are used here which require only addition, subtraction, shifting (combinational shift) operators and the SVPWM algorithm is implemented for three phase voltage source inverter on FPGA. The proposed method is described as follows: the conventional SVPWM algorithm includes d-q transformation of three input voltages

V_{an}, V_{bn} and V_{cn} as given by equation (4), require $\sqrt{3}$ to be calculated but it is difficult to implement floating point numbers using FPGA. Hence instead of d-q transformation, intermediate transformation vectors are used. The (1) can be written as follows [4]:

$$\therefore V_d = \frac{1}{3}[2V_a - V_b - V_c] \qquad (5)$$
$$V_q = \frac{1}{\sqrt{3}}[V_b - V_c]$$

By defining intermediate variables as X_d and X_q, (6) is obtained as [3]:

$$X_d = 2V_a - V_b - V_c \qquad (6)$$
$$X_q = V_b - V_c$$

The proposed circuit and proposed VHDL code for implementing of X_d and X_q are based on (6) thus in this work the index $2V_a$ is used alone with shift and subtraction operators. Hence the proposed method to calculate X_d and X_q needs only simple operands namely shifter (for $2V_a$ i.e. multiplication by 2), and subtractor that all are combinational circuits without clock signal and extra hardware, so the hardware of the controller unit is optimized with low power consumption.

Determination of The Sectors

The determination of sector depends only on the sign of X_d and X_q as per following rules. Determination of the sectors can be done by simply checking 3 conditions:

Condition 1: sign of X_d
Condition 2: sign of X_q
Condition 3: $|X_d| > |X_q/2|$

This part is done by simple combination logic circuits. First condition 1 and then condition 2 is considered and then expression $X_q/2$ is achieved by shifting one bit to right. Than $|X_d|$, $|X_q/2|$ are obtained without using the ***abs()*** function. Total of three conditions 1, 2 and 3 are designed with optimized and combinational logic circuit. The rules to find sectors are base on the following statements:

Rule 1: if (X_d>0 & X_q>0 & $|X_d|$>$|X_q/2|$) Sector-1
Rule 2: if (X_d>0 & X_q>0 & $|X_d|$<$|X_q/2|$) Sector-2 OR if (X_d<0 & X_q>0 & $|X_d|$<$|X_q/2|$) Sector-2
Rule 3: if (X_d<0 & X_q>0 & $|X_d|$>$|X_q/2|$) Sector-3
Rule 4: if (X_d<0 & X_q<0 & $|X_d|$>$|X_q/2|$) Sector-4
Rule 5: if (X_d>0 & X_q<0 & $|X_d|$<$|X_q/2|$) Sector-5 OR if (X_d<0 & X_q <0 & $|X_d|$<$|X_q/2|$) Sector-5
Rule 6: if (X_d>0 & X_q<0 & $|X_d|$>$|X_q/2|$) Sector-6

To compute $X_q/2$, the signed one bit shift to right is applied, thus a signed shifter is made for ordering of input number to be inversed. Proposed VHDL code for this controller unit is shown as follows:

```
if(x_q(7)='1')then
    x_q_reg<=x_q(7)&x_q(7 downto 1);
else
    x_q_reg<='0'&x_q(7 downto 1);
end if;
if(x_d(7)='1')then
    not_x_d<=not(x_d)+x"01";
else
    not_x_d<=x_d;
end if;
if(x_q_reg(7)='1')then
    not_x_q<=not(x_q_reg)+x"01";
else
    not_x_q<=x_q_reg;
end if;
if(not_x_d>not_x_q)then
    u<='1';
else
    u<='0';
end if;
t(0)<=((not(x_d(7))) and (not(x_q(7))) and u);
t(1)<=((not(x_d(7))) and (not(x_q(7))) and not(u));
t(2)<=( x_d(7) and (not(x_q(7))) and not(u));
t(3)<=(x_d(7) and (not(x_q(7))) and u);
t(4)<=(x_d(7) and x_q(7) and u);
t(5)<=((not(x_d(7))) and x_q(7) and not(u));
t(6)<=(x_d(7) and x_q(7) and not(u));
t(7)<=((not(x_d(7))) and x_q(7) and u);
if(t(0)='1')then
    sector<="001";
end if;
if(t(1)='1')then
   sector<="010";
end if;
if(t(2)='1')then
   sector<="010";
end if;
if(t(3)='1')then
   sector<="011";
end if;
if(t(4)='1')then
   sector<="100";
end if;
if(t(5)='1')then
   sector<="101";
end if;
if(t(6)='1')then
   sector<="101";
end if;
if(t(7)='1')then
   sector<="110";
end if;
```

Determination Switching Times

For a symmetric space vector PWM, the output voltage i.e. X_d and X_q can be in any of the sector 1 to sector 6, which is given by (7):

$$\begin{bmatrix} T_n \\ T_{n+1} \end{bmatrix} = T_{PWM} M_0 \begin{bmatrix} X_d \\ X_q \end{bmatrix} \qquad (7)$$

Equation (7) shows that in each PWM period, the output voltages are approximated as (T_z/V_{dc}) by switching between the two non-zero basic vectors that border the sector of the output voltages. The sum of T_n and T_{n+1} should be less than or equal to T_{PWM} and rest of period of the switching time should be equal with T_0. The switching time can be calculated by M_0, which is called as decomposition matrix, given by (8). If reference vector is located in sector I, e.g. if 'n'=1, then M_{00}=1, M_{01}= -1/2, M_{10}=0 and M_{11}=1. The coefficients of decomposition matrix according to sector are given in Table II.

$$M_0 = \begin{bmatrix} M_{00} & M_{01} \\ M_{10} & M_{11} \end{bmatrix} \qquad (8)$$

Table II: Coefficients of decomposition matrix

Sector 'n'	1	2	3	4	5	6
M_{00}	1	1	0	-1	-1	0
M_{01}	-1/2	1/2	1	1/2	-1/2	-1
M_{10}	0	-1	-1	0	1	1
M_{11}	1	1/2	-1/2	-1	-1/2	1/2

So form (8) the switching times are calculated as:

If(sector=1)=> $T_n=X_d$*-shift_to_right(*X_q*),*$T_{n+1}=X_q$*,*$T_0=\sqrt{3}$ - T_n+T_{n+1}

If(sector=2)=> $T_n=X_d$*+shift_to_right(*X_q*),*T_{n+1}= *shift_to_right(*X_q*)-*X_d*,*$T_0=\sqrt{3}$ -T_n+T_{n+1}

If(sector=3)=> $T_n=X_q$*,*$T_{n+1}=-X_d$*-shift_to_right(*X_q*),*$T_0=\sqrt{3}$ - T_n+T_{n+1}

If(sector=4)=> $T_n=-X_d$*+shift_to_right(*X_q*),*$T_{n+1}=-X_q$*,*$T_0=\sqrt{3}$ - T_n+T_{n+1}

If(sector=5)=> $T_n=-X_d$*-shift_to_right(*X_q*),*$T_{n+1}=X_d$*-shift_to_right(*X_q*),*$T_0=\sqrt{3}$ -T_n+T_{n+1}

If(sector=6)=> $T_n=-X_q$, $T_{n+1}=X_d$*+shift_to_right(*X_q*),*$T_0=\sqrt{3}$ - T_n+T_{n+1}

SVPWM Pulses Generation Module

Time duration of any PWM pulse is dependent on the number of sectors and switching times T0, T1 and T2. To generate PWM pulses a counter is used, this counter operates by the following conditions:

If(Sector=1) => {
If (counter =T0/2) => PWM_A='1', PWM_B='0', PWM_C='0';
If (counter = T0/2+T1) => PWM_A='1', PWM_B='1', PWM_C='0';
If (counter = T0/2+T1+T2) => PWM_A='1', PWM_B='1', PWM_C='1';
If (counter = T0+T1+T2) => PWM_A='1', PWM_B='1', PWM_C='1';
If (counter = T0+T1+T2+T0/2) => PWM_A='1', PWM_B='1', PWM_C='0';
If (counter = T0+T1+2T2+T0/2) => PWM_A='1', PWM_B='0', PWM_C='0';
If (counter = T0+2T1+2T2+T0/2) => PWM_A='0', PWM_B='0', PWM_C='0';
If (counter =2T0+2T1+2T2) => PWM_A='0', PWM_B='0', PWM_C='0';
}

The proposed calculation for this block is realized based on simple operators such as shifters and adders, thus reduces the complexity, power consumption and increases total performance of conventional SVPWM algorithm. Waveform of proposed FPGA based SVPWM controller is shown in Fig. 4.

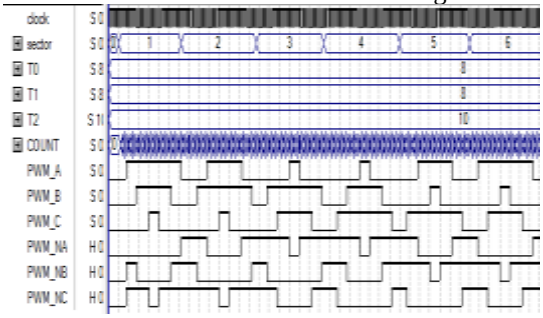

Fig. 4: Waveform of proposed FPGA based SVPWM

Fig.5 shows waveform of proposed PWM generation for sectors 1 and 2

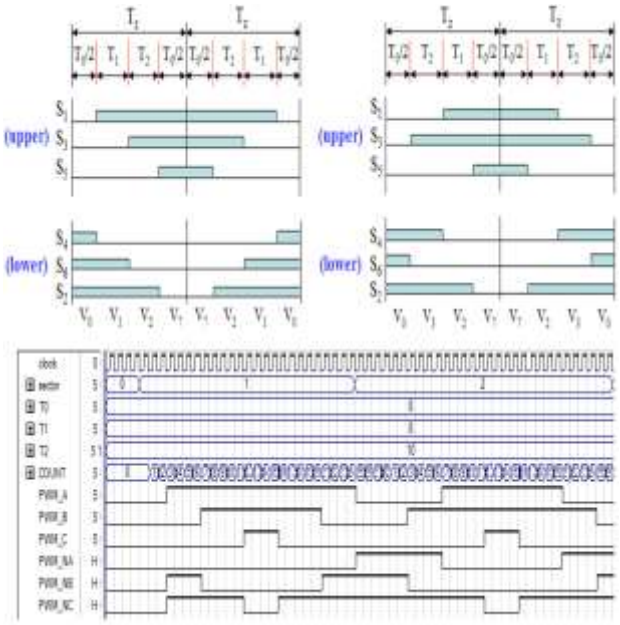

Fig.5: Waveform of proposed PWM generation for sectors 1 and 2

Proposed design is an optimized realization of SVPWM on FPGA and this work is accomplished via VHDL hardware description language by using Xilinx ISE software synthesized and implemented on FPGA in Virtex IV family. Simulation results are achieved using Quartus II software. Also power consumption is analyzed using Xilinx XPower analyzer. Table III and table IV show the comparison between results and previous works. Power consumption, numbers of LUTs, numbers of Slices and FFs and the type of device that has been used in proposed design, are shown in tables V and table VI.

Table III: Utilized hardware on FPGA in [5]

Implementation	Device	Total of logic elements	Total memory bits	LC Register
[5]	EP20k200 EFC484-2X	520(6%)	9216(9%)	31

Table IV: Utilized hardware on FPGA in [4]

Implementation	Multipliers	Dividers	Adder/ subteractor	CLBs
[4]	0	0	32	490

Table V: Utilized hardware on FPGA in proposed design and [6]

Implementation	Number of Slices	Number of FFs	LUTs	Device	Number of IOBs
Proposed Method	209	83	384	XC4VFX100	36
[6]	100	360	Max RAM bits 3600	Xc4003A	80

Table VI: Power consumption of our proposed method

Clock Frequncy (MHz)	Dynamic power consumption(mW)
100	63
75	58
50	52
25	44

Conclusion

In This paper a new design of FPGA based SVPWM controller with low power has been presented. Combined digital circuits have been used to reduce power consumption and area. Also the comparison between proposed system and other works has been shown, It is clear that the proposed approach is most effective to execute with the low cost and low power consumption. The proposed method has been synthesized and implemented using Xilinx ISE Virtex IV FPGA and power is analized using Xilinx XPower analyzer.

References

[1] N. M. Nayan, "Design of a Microcontroller-Based Converter for 3-Phase Brushless DC Motor Drives", Thesis for Masters of Science, UNIVERSITI SAINS MALAYSIA, September 2009.

[2] Sunil Panda, Anupam Mishra, B. Srinivas, "Control of Voltage Source Inverters using PWM/SVPWM for Adjustable Speed Drive Applications", the award of the degree of BACHELOR , Department of Electrical Engineering National Institute Of Technology Rourkela, May 2009.

[3] Óscar López, Jacobo Álvarez, Jesús Doval-Gandoy, and Francisco D. Freijedo, "Multilevel Multiphase Space Vector PWM Algorithm", IEEE Trans. On Industrial Electronics, Vol. 55, No. 5, pp. 1933-1942, May 2008.

[4] D. N. Sonawane, M. S. Sutaone, B. N. Choudhari and Abhijeet Badurkar, "FPGA Implementation of Simplified SVPWM Algorithm for Three Phase Voltage Source Inverter", International Journal of Computer and Electrical Engineering, Vol.2, No.6, pp. 1010-1017, December, 2010.

[5] Tole Sutikno, Auzani Jidin, and Mohd Farriz Basar, "Simple Realization of 5-Segment Discontinuous SVPWM Based on FPGA", International Journal of Computer and Electrical Engineering, Vol. 2, No. 1, pp. 147-157, February, 2010.

[6] Ying-Yu Tzou, Member, IEEE, and Hau-Jean Hsu, "FPGA Realization of Space-Vector PWM Control IC for Three-Phase PWM Inverters", IEEE Trans. On Power Electronics, Vol. 12, No. 6, pp. 953-963, Nov. 1997.