Arab J Sci Eng (2013) 38:3059–3070
DOI 10.1007/s13369-012-0464-2

RESEARCH ARTICLE - ELECTRICAL ENGINEERING

Design and Implementation of Space Vector PWM Inverter Based on a Low Cost Microcontroller

Mahmoud M. Gaballah

Received: 31 October 2011 / Accepted: 4 February 2012 / Published online: 13 December 2012

Abstract The rapid development of high switching frequency power electronics in the past decade leads towards wider application of voltage source inverters in AC power generation. Therefore, this prompts the need for a modulation technique with less total harmonic distortion, fewer switching losses, and wider linear modulation range. Space vector pulse width modulation (SVPWM) provides a better technique compared to the more commonly used PWM or sinusoidal PWM (SPWM) techniques because of their easier digital realization and better DC bus utilization. This paper aims to achieve two goals. One is to introduce an SVPWM technique based on a reduced computation method which is much simpler and faster than conventional means. The other is presenting a practical design and implementation of space vector PWM inverter based on a low cost microcontroller to overcome many of the issues experienced using the conventional methods.

Keywords Space vector pulse width modulation (SVPWM) · Uninterruptible power supplies (UPS's) · Voltage source inverter (VSI)

الخلاصة

إن التطور السريع في صناعة المكونات الإلكترونية عالية القدرة والتردد في العقد الماضي أدى الى استخدامها بصورة كبيرة في تطبيقات العواكس "جهد مستمر - جهد متردد" لتوليد الطاقة الكهربية، لذلك دعت الحاجة الى استحداث طرق تحكم جديدة لتوليد النبضات اللازمة لتشغيل العاكس بحيث تضمن معدلات تشويه متدنية للجهد: سهولة وسرعة فى التنفيذ، وأيضا عمر أطول للمكونات الإلكترونية .

وقد تم - في هذا البحث - تطوير وتنفيذ طريقة لتوليد النبضات اللازمة لتشغيل العاكس " جهد مستمر- جهد متردد " ثلاثي الأوجه باستخدام طريقة جديدة لتعديل المتجه الفراغي لعرض النبضة. وتعد هذه الطريقة أفضل وأقل تعقيدا من الطرق الأخرى، لأنها لا تستخدم أنظمة الجدولة للحصول على قيم تعديل عرض النبضة ولا تحتاج كذلك إلى عمليات منطقية معقدة.

وقد تم عمل أنموذج رياضي للنظام ككل ومحاكاته من خلال برنامج الماتلاب عند نسب تعديل مختلفة وبأحمال مختلفة، وعليه فقد أثبتت النتائج النظرية أن هذه الطريقة صالحة لتشغيل عاكس مصدر الجهد ثلاثي الأوجه. وتم أيضا بناء النظام السابق من خلال محاكاته عمليا، وتم تطبيق الطريقة المستخدمة فى البحث على متحكم دقيق ميكروكنترولر رخيص الثمن من النوع "PIC18F4431". وقدمت النتائج العملية على عاكس جهد "تيار مستمر- تيار متردد" ثنائي المستوى ثلاثي الأوجه يتبعه مرشح ثلاثي الأوجه من النوع "LC". وقد وجد تطابق النتائج العملية مع النتائج المحاكاتية مما يثبت صلاحية هذه الطريقة لتشغيل عاكس مصدر الجهد بطريقة مبسطه وسهلة التنفيذ.

1 Introduction

Voltage source inverter (VSI) synthesizes AC voltage and frequency from a constant DC voltage using PWM techniques. Nowadays, VSI is used in large applications such as variable speed drives (VSDs), uninterruptible power supplies (UPS's), frequency converters, and active filters [1,2]. PWM techniques have been studied extensively during the last few decades. A large variety of methods, differing in concept and performance, have been developed to achieve one or more of the following objectives: wide linear modulation range, fewer

M. M. Gaballah (✉)
Department of Industrial Electronics and Control Engineering,
Faculty of Electronic Engineering, Minufiya University,
Menufia 32852, Egypt
e-mail: mahmoud_g@sigma-electronics.com

M. M. Gaballah
Research and development, Sigma Electronic Industries,
9 Obour buildings, Salah Salem St, Cairo, Egypt

switching losses, less total harmonic distortion (THD), easy implementation and less computation time [3,4].

Several modulation strategies differing in concept and performance have been developed. With the development of microprocessors, space vector modulation has become one of the most important PWM methods for three-phase converters [5]. It uses the space vector concept to compute the duty cycle of the switches. It is simply the digital implementation of PWM modulators. An aptitude for easy digital implementation and wide linear modulation range for line to line voltages are the noticeable features of space vector modulation.

Many methods have been developed to implement the space vector pulse width modulation (SVPWM) for driving VSI's. Generally, the SVPWM implementation involves sector identification, switching time calculation, switching vector determination, and optimum-switching-sequence selection for the inverter voltage vectors [6,7]. Sector identification can be done by coordinate transformation as introduced in [8–10] or by repeated comparison of the three-phase reference voltages as introduced in [6–11]. The lookup tables can be used for determining the switching vectors in best switching sequence as introduced in [12]. Calculating the duration of the switching vectors can be simplified by mapping the sector of the multilevel inverter to a corresponding sector of the two-level inverter as introduced in [13,14].

The objective of this paper is to introduce a simplified SVPWM technique in which the inverter leg switching times are directly obtained from the instantaneous sampled reference phase voltages, and the inverter switching vectors are generated automatically. This method is much simpler and more executable than conventional means without lookup tables or complex logical judgments. In addition, an objective of this paper is to introduce a practical SVPWM inverter design based on a low cost microcontroller. The practical design is modeled using the MATLAB SIMULINK software package, and experimentally implemented on the low cost microchip PIC microcontroller 18F4431 platform.

2 Three-Phase Inverter

The major purpose of the PWM inverter is to generate a variable-voltage variable-frequency (VVVF) three-phase voltage from a DC voltage. Two-level VSI consists of six power semiconductor switches with antiparallel diodes. In the widely used pulse width modulation (PWM) methods, the inverter output voltage approximates the reference value through high frequency switching for the six power semiconductor switches. The circuit model of a typical two-level inverter is as shown in Fig. 1. S1–S6 are the six power switches that shape the output, these are controlled by the signal to terminals a, à, b, b̀, c, and c̀.

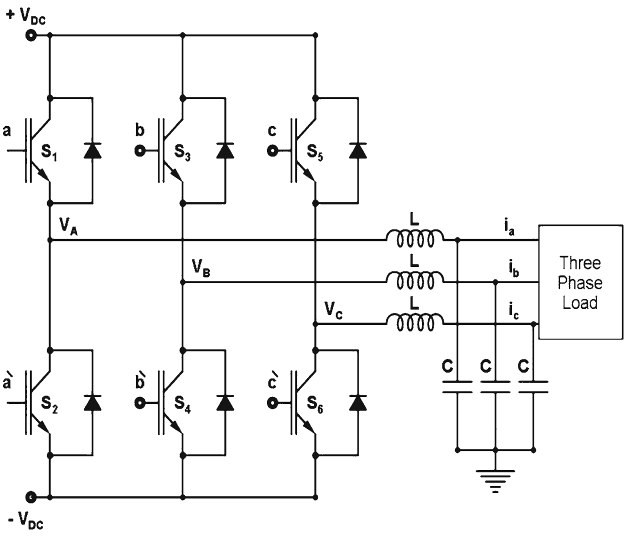


Fig. 1 Two-level voltage source inverter

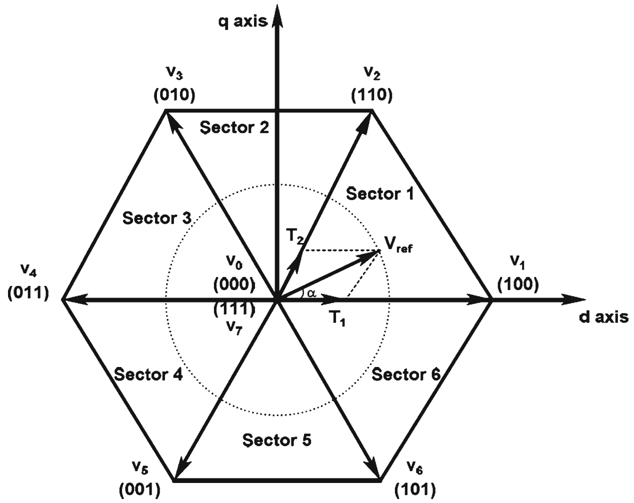


Fig. 2 Output voltage space of the two-level inverter in dq coordinates

It is assumed that S1 and S2, S3, and S4 as well as S5 and S6 are switched in a complementary way. There are only eight possible switching vectors. Six out of these eight vectors produce a non-zero voltage and are known as non-zero switching states: the remaining two vectors produce zero output voltage and known as zero switching states. The output voltages of the inverter are composed by these eight switch states. The six active vectors divide the space vector plane into six equal sized sectors of 60° with equal magnitude which forms an origin centered hexagon, and two zero space vectors found at the origin as shown in Fig. 2.

The hexagon is the maximum boundary of the space vector, and the circle is the trajectory of the regular sinusoidal outputs in linear modulation. Table 1 lists all of the possible switching vectors and the respective line to line/line to neutral voltages.

Table 1 Possible switching vectors, phase voltage, and output line to line voltage

Voltage vectors	Switching vectors			Pole voltage			Line voltage		
	a	b	c	V_{a0}	V_{b0}	V_{c0}	V_{ab}	V_{bc}	V_{ca}
V_0	0	0	0	0	0	0	0	0	0
V_1	1	0	0	2/3	−1/3	−1/3	1	0	−1
V_2	1	1	0	1/3	1/3	−2/3	1	0	−1
V_3	0	1	0	−1/3	2/3	−1/3	−1	1	0
V_4	0	1	1	−2/3	1/3	1/3	−1	0	1
V_5	0	0	1	−1/3	−1/3	2/3	0	−1	1
V_6	1	0	1	1/3	−2/3	1/3	1	−1	0
V_7	1	1	1	0	0	0	0	0	0

All the respective voltage should be multiplied by V_{DC}

To obtain a sinusoidal waveform from the VSI, a voltage reference V_{ref} is provided in terms of a revolving space vector. The magnitude and the frequency of the fundamental component are specified by the magnitude and frequency, respectively, of the reference vector. The reference vector is sampled once in every sub-cycle. The inverter is maintained in different states for appropriate durations such that an average voltage vector equal to the sampled reference vector is generated over a given sub-cycle.

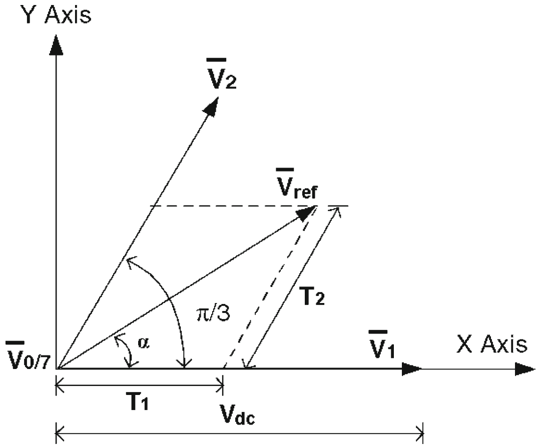


Fig. 3 Reference vector in sector 1

3 Conventional SVPWM Algorithm

In the space vector approach, the inverter states used are the two zero states, and the two active states. These voltage vectors are the closest to the commanded voltage vector. The SVM algorithm has four switching rules: (a) the trajectory of V_{ref} should be a circle, (b) only one switching by state transition, (c) not more than three switching in one sampling period, and (d) the final state of one sample must be the initial state of the next sample.

These rules help in limiting the number of switching actions and therefore, there is a decrease in the switching losses. In addition, they maintain symmetry in switching waveforms at the VSI output to achieve the lower THD. For example, for a commanded vector in sector 1 as shown in Fig. 3, switching states 0, 1, 2, and 7 can be used. If we assume that during the sampling interval T_{s}, the reference voltage $\bar{V}_{\mathrm{ref}}$ remains steady.

For implementing the conventional SVPWM using the SVM rules $\bar{V}_{\mathrm{ref}}$ can be expressed as follows:

$$\bar{V}_{\mathrm{ref}} = \left(\frac{T_1}{T_{\mathrm{s}}} \times \bar{V}_1\right) + \left(\frac{T_2}{T_{\mathrm{s}}} \times \bar{V}_2\right) + \left(\frac{T_{0/7}}{T_{\mathrm{s}}} \times \bar{V}_{0/7}\right) \quad (1)$$

Equation 1 means the inverter is in active state 1 for a period T_1 and in active state 2 for a period T_2. For the remaining time of the sampling interval period T_{s} there is no voltage applied. This can be achieved by applying inactive state 0 or 7 for the remaining time T_0 or T_7. To generate this vector in an average sense, the durations for which the active state 1, the active state 2, and the two zero states together must be applied which are given by T_1, T_2 and T_{Z}, respectively, obtained as:

$$T_1 = \bar{V}_{\mathrm{ref}}\ \sin(60^\circ - \alpha) \quad (2)$$

$$T_2 = \frac{\bar{V}_{\mathrm{ref}}\ \sin(\alpha)}{\sin(60^\circ)} \quad (3)$$

$$T_{\mathrm{Z}} = T_{\mathrm{s}} - T_1 - T_2 \quad (4)$$

where α is the angle of rotating vector $\bar{V}_{\mathrm{ref}}$.

The division of the duration T_{Z} between the two zero vectors T_0 and T_7 is a degree of freedom in the space vector approach. This division of T_{Z} in a sub-cycle is equivalent to adding a common mode component to the three-phase average pole voltages. The typical VSI switching waveforms in sector 1, as defined in Eq. 1 are as given in Fig. 4. Realization of conventional SVPWM involves the following steps: (1) Coordinate transformation for the reference vector $\bar{V}_{\mathrm{ref}}$ from rotating reference frame to stationary reference frame.

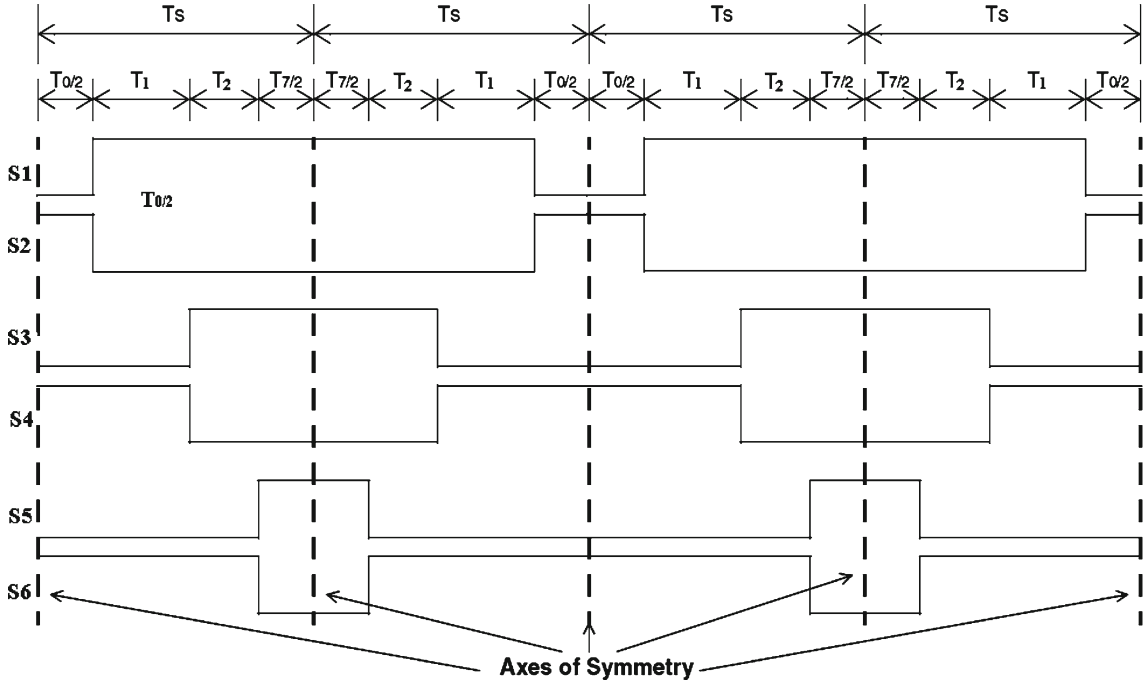


Fig. 4 Typical VSI switching waveforms in sector 1

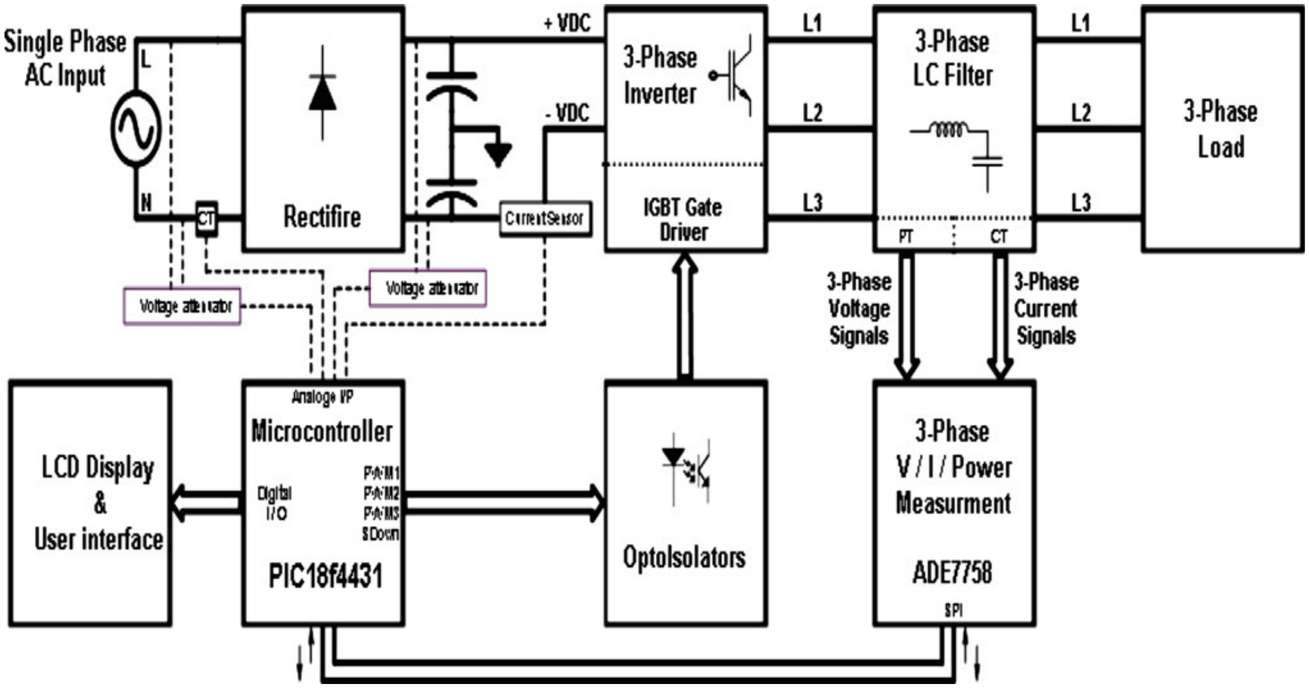


Fig. 5 Overall block diagram of the experiment

(2) Determine time durations T_1, T_2, and T_0. (3) Determine the switching time of each transistor (S1–S6).

4 SVPWM Using a Reduced Computation Method

This method is based on the principle of equivalence of SVPWM with sinusoidal PWM (SPWM) and can generate the SVPWM signals directly from the instantaneous reference phase voltages. In the sinusoidal PWM scheme for a two-level inverter, each reference phase voltage is compared with the triangular carrier, and the individual pole voltages are generated independent of each other [3]. To obtain the maximum possible peak amplitude of the fundamental phase voltage in linear modulation a common

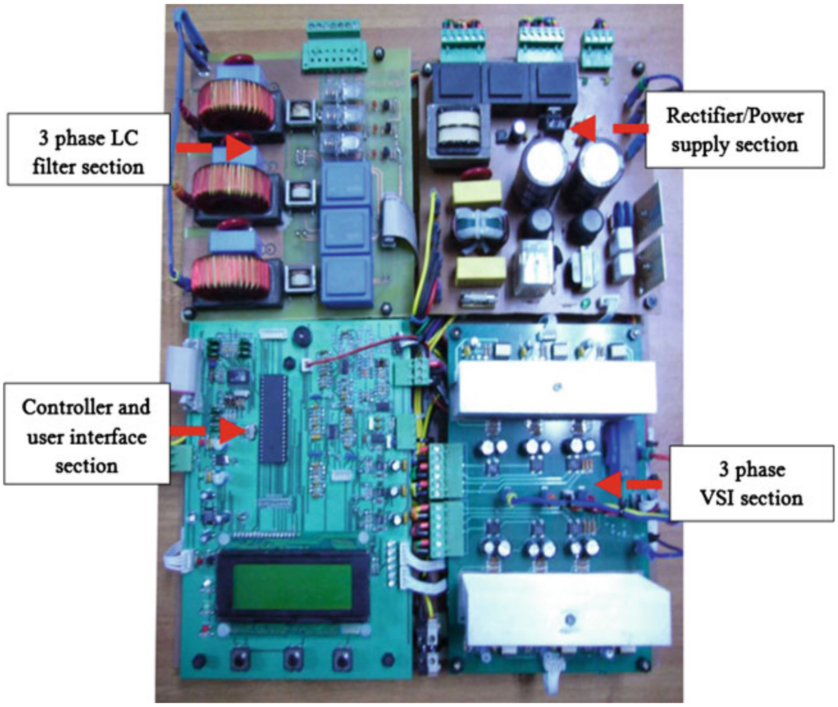

3 phase LC filter section
Rectifier/Power supply section
Controller and user interface section
3 phase VSI section

Fig. 6 Real image for the experiment

mode voltage V_{offset} is added to the reference phase voltages [15,16], where the magnitude of V_{offset} is given by Eq. (5)

$$\mathrm{V}_{\text{offset}} = \frac{-(V_{\max} - V_{\min})}{2} \tag{5}$$

In Eq. (5), $V_{\max}$ is the maximum magnitude of the three sampled reference phase voltages, while $V_{\min}$ is the minimum magnitude of the three sampled reference phase voltages. In a sampling interval the addition of the common mode voltage V_{offset} results in the active inverter switching vectors being centered in a sampling interval, making the SPWM technique equivalent to the SVPWM technique [3]. Equation (5) is based on the fact that in a sampling interval, the reference phase which has lowest magnitude (termed the min-phase) crosses the triangular carrier first and causes the first transition in the inverter switching state, while the reference phase which has the maximum magnitude (termed the max-phase) crosses the carrier last and causes the last switching transition in the inverter switching states in a two-level SVPWM scheme [15,16]. Thus, the switching periods of the active vectors can be determined from the (max-phase and min-phase) sampled reference phase voltage amplitudes in a two-level inverter scheme [17]. The idea behind this SVPWM technique is to determine the sampled reference phase, from the three sampled reference phases, which crosses the triangular first (first-cross) and the reference phase which crosses the triangular carrier last (third-cross). This SVPWM technique presents a simple way to determine the time instants at which the three reference phases cross the triangular carriers using only the instantaneous reference phase amplitudes. These time instants are sorted to find the offset voltage [18,19]. This voltage is then added to the reference phase voltages, so the middle inverter switching vectors are centered (during a sampling interval), as in the conventional two-level SPWM scheme [20]. Implementing this SVPWM method for driving a two-level VSI involves three steps:

A: Read the sampled reference phase amplitudes of VAN, V_{BN} and V_{CN} for the present sampling interval and then calculate the time equivalents of phase voltages, i.e. T_{as}, T_{bs} and T_{cs} as:

$$T_{\text{as}} = V_{\text{AN}} \times \frac{T_{\text{s}}}{V_{\text{DC}}} \tag{6}$$

$$T_{\text{bs}} = V_{\text{BN}} \times \frac{T_{\text{s}}}{V_{\text{DC}}} \tag{7}$$

$$T_{\text{cs}} = V_{\text{CN}} \times \frac{T_{\text{s}}}{V_{\text{DC}}} \tag{8}$$

where T_{s} is the sampling time period and V_{DC} is the DC link voltage across inverter terminals.

B: Find T_{offset} as:

$$T_{\text{offset}} = 0.5T_{\text{s}} - 0.5(T_{\max} - T_{\min}) \tag{9}$$

where $T_{\max}$ and $T_{\min}$ are the maximum and minimum of T_{as}, T_{bs} and T_{cs}.

C: Find T_{ga}, T_{gb} and T_{gc} as:

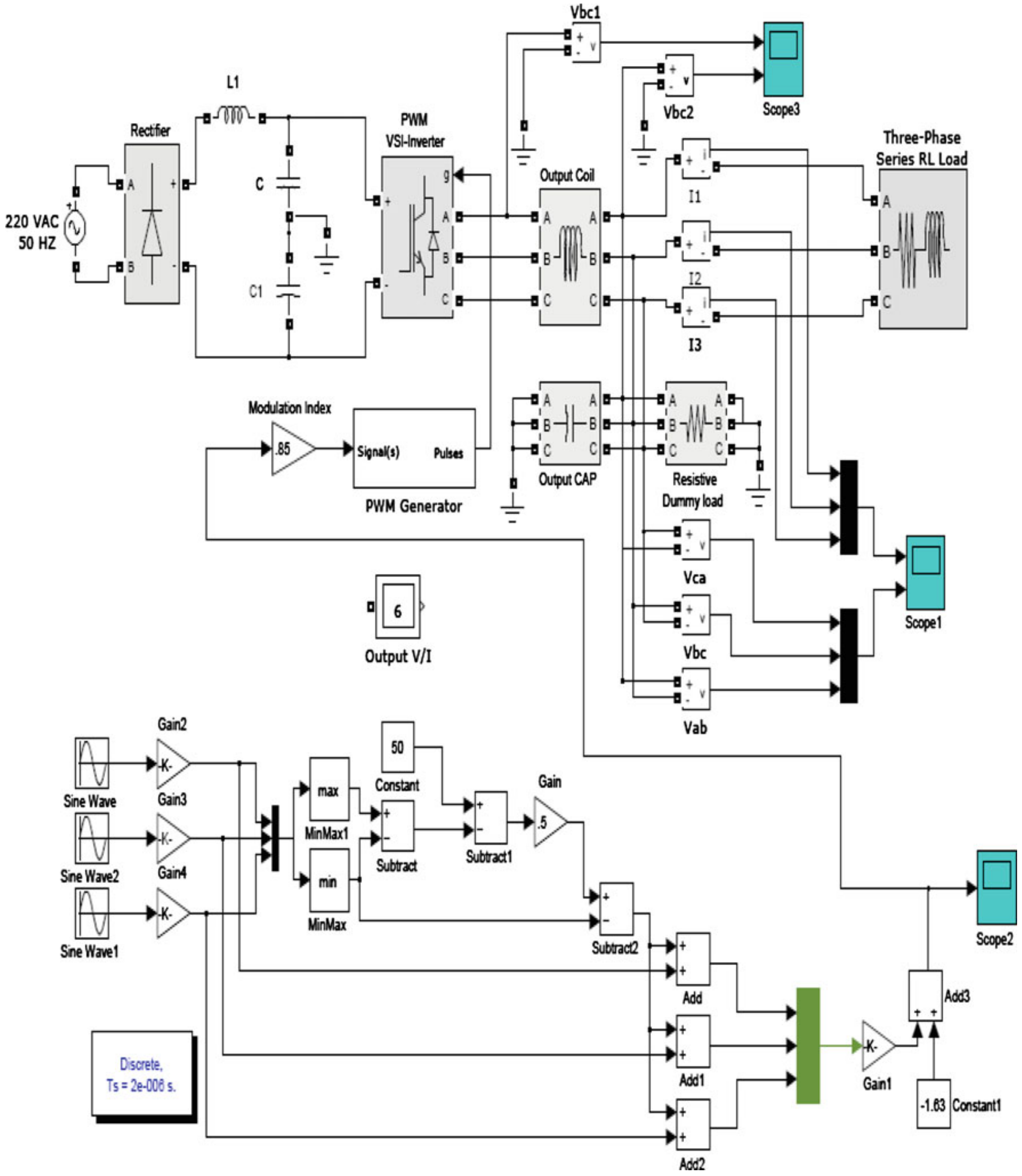


Fig. 7 MATLAB SIMULINK model for the system

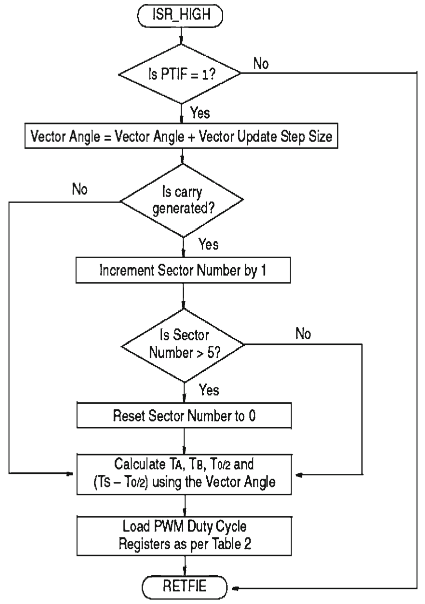
$$T_{\mathrm{ga}} = T_{\mathrm{as}} + T_{\mathrm{offset}} \tag{10}$$

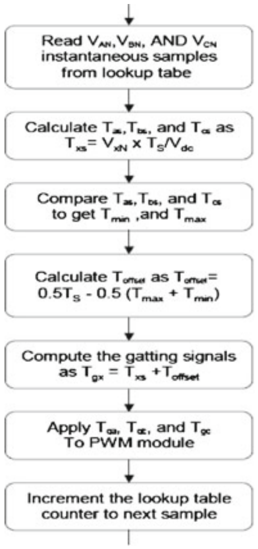
$$T_{\mathrm{gb}} = T_{\mathrm{bs}} + T_{\mathrm{offset}} \tag{11}$$

$$T_{\mathrm{gc}} = T_{\mathrm{cs}} + T_{\mathrm{offset}} \tag{12}$$

where T_{ga}, T_{gb} and T_{gc} are the gating signals during which the top switches in a leg are turned on.

Equations (6)–(12) show that the centering of the middle inverter switching vectors of the SVPWM in this method reduces the computation time required to determine the switching times for inverter legs, making the algorithm suitable for real-time implementation. Furthermore, the complicated calculations for inverter switching vector times and lookup tables for selecting the inverter switching vector are avoided in this scheme. A comparison between the conventional SVPWM algorithm and the SVPWM based on equivalence with SPWM is introduced below:

<table>
<tr><th>Conventional SVPWM</th><th>SVPWM Based on Equivalence with SPWM</th></tr>
<tr><td>Generates the gating drive signals using switching rules as listed below:
<table>
<tr><th>Sector</th><th>T_{ga}</th><th>T_{gb}</th><th>T_{gc}</th></tr>
<tr><td>1</td><td>$T_0/2$</td><td>$T_0/2 +T_1$</td><td>$T_S-T_0/2$</td></tr>
<tr><td>2</td><td>$T_0/2 +T_2$</td><td>$T_0/2$</td><td>$T_S-T_0/2$</td></tr>
<tr><td>3</td><td>$T_S-T_0/2$</td><td>$T_0/2$</td><td>$T_0/2+T_1$</td></tr>
<tr><td>4</td><td>$T_S-T_0/2$</td><td>$T_0/2$ +T2</td><td>$T_0/2$</td></tr>
<tr><td>5</td><td>$T_0/2+ T_1$</td><td>$T_S-T_0/2$</td><td>$T_0/2$</td></tr>
<tr><td>6</td><td>$T_0/2$</td><td>$T_S-T_0/2$</td><td>$T_0/2+T_2$</td></tr>
</table></td><td>The gating drive signals are generated directly from the instantaneous samples of the reference phase amplitudes.</td></tr>
<tr><td>Needs sector identification to define the switching rules in each sector.</td><td>It needs no sector identifications.</td></tr>
<tr><td>Needs Vref and α angle information.</td><td>Vref and α angle information is not needed.</td></tr>
<tr><td>Uses look-up tables for switching times calculations.</td><td>Do not use any look-up tables for switching times calculations.</td></tr>
<tr><th colspan="2">Digital Implementation</th></tr>
<tr><td>
ISR_HIGH
Is PTIF = 1?
No
Yes
Vector Angle = Vector Angle + Vector Update Step Size
No
Is carry generated?
Yes
Increment Sector Number by 1
Is Sector Number > 5?
No
Yes
Reset Sector Number to 0
Calculate TA, TB, T0/2 and (TS – T0/2) using the Vector Angle
Load PWM Duty Cycle Registers as per Table 2
RETFIE

</td><td>
Read VAN,VBN, AND VCN instantaneous samples from lookup tabe
Calculate Tas,Tbs, and Tcs as Txs= VxN x TS/Vdc
Compare Tas,Tbs, and Tcs to get Tmin ,and Tmax
Calculate Toffset as Toffset= 0.5TS - 0.5 (Tmax + Tmin)
Compute the gatting signals as Tgx = Txs +Toffset
Apply Tga, Tgb, and Tgc To PWM module
Increment the lookup table counter to next sample

</td></tr>
<tr><td>When the microcontroller is running at 20 MHZ: this routine consumes processing time = 25.3uSec. This means:
• High CPU usage
• High execution time
• Big code size</td><td>When the microcontroller is running at 20 MHZ: This routine consumes processing time = 15.2 uSec. This means:
• low CPU usage
• low execution time
• small code size</td></tr>
</table>

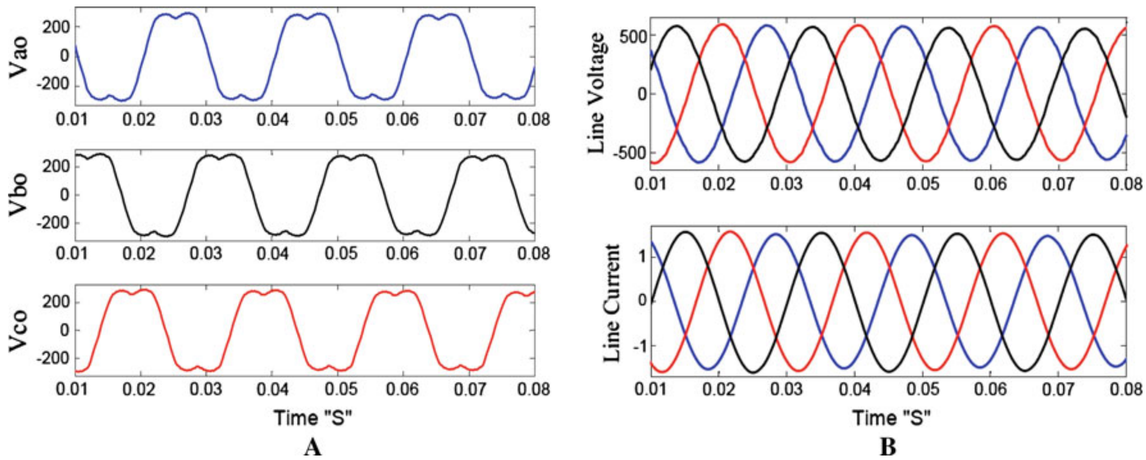


Fig. 8 The simulation results at modulation index = 0.1. **a** Output pole voltage waveforms, **b** line voltage/current waveforms across load terminals

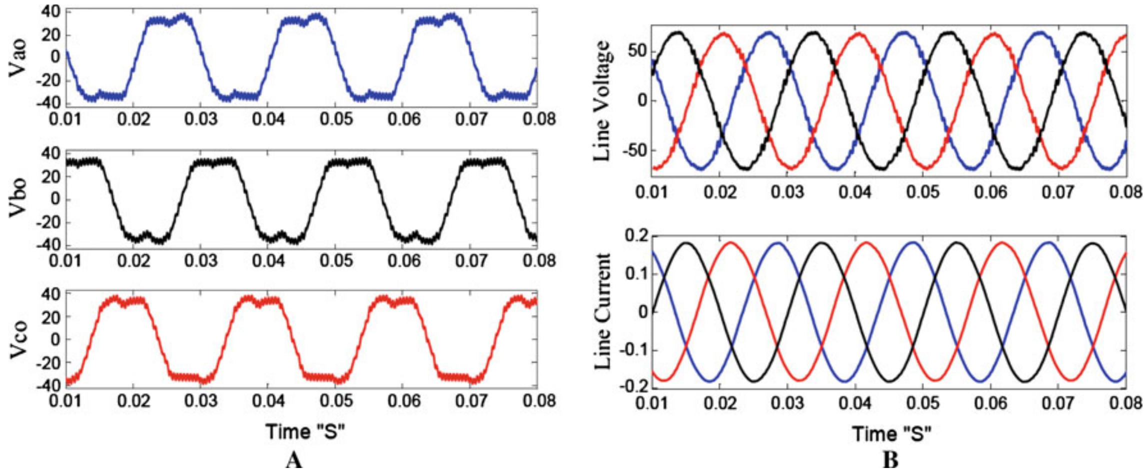


Fig. 9 The simulation results at modulation index = 0.85. **a** Output pole voltage waveforms, **b** line voltage/current waveforms across load terminals

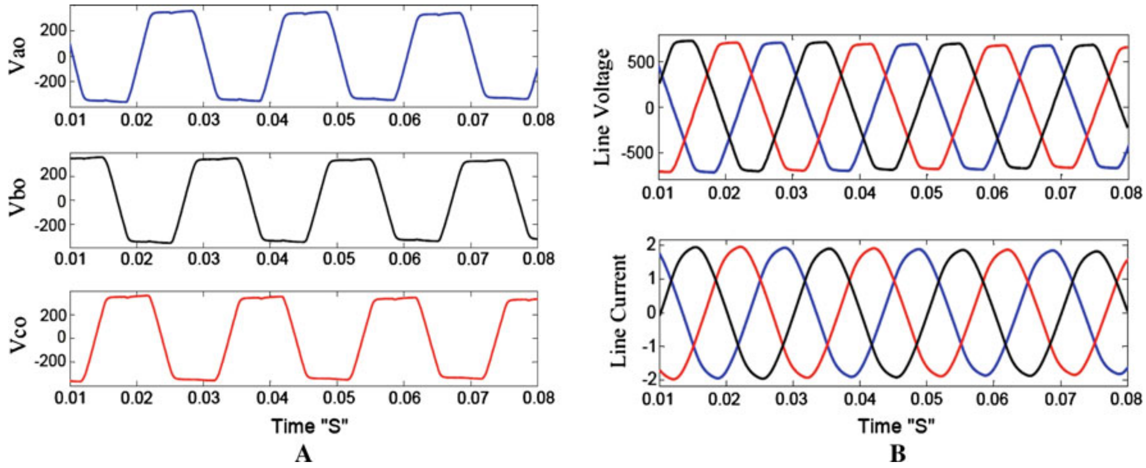


Fig. 10 Shows the simulation results at modulation index = 1.15. **a** Output pole voltage waveforms, **b** line voltage/current waveforms across load terminals

5 Experimental Setup

Hardware: The practical work is as shown in Fig. 5. The input stage consists of a dual half bridge diode rectifier, which provides the DC bus voltage ±310 V from the 220 VAC input and controlled by enable signal from a microcontroller (MCU). The output is provided by a three-phase two-level VSI followed by a three-phase LC filter and an output enable relay. The inverter converts the DC bus voltage back to a sinusoidal voltage using the SVPWM technique. The output inverter is fully controlled by the MCU and generates a pure sinusoidal waveform, free of any disturbance.

The system consists of four printed circuit board (PCB), as shown in Fig. 6.

(1) A rectifier/power supply PCB which provides the DC link voltage via the rectifier circuit and provides all DC power supply voltages for the various control circuits such as the main control circuits and the inverter driving circuits. (2) A three-phase two-level inverter PCB, which converts the DC voltage of the DC bus to a three-phase sinusoidal voltage with the required amplitude and frequency via six power IGBTs derived from the control circuit. (3) A three-phase LC filter PCB which removes the undesired component within the output waveforms coming from inverter and maintains the fundamental waveform with pure sinusoidal voltage. (4) A controller/user interface PCB, which manages all control algorithms and measurements of the system, also it contains a user interface that includes a 4 × 20 characters LCD with three input switches and three indication LEDs.

Software: The firmware is developed for a microchip PIC18F4431 microcontroller using ASSEMBLY language, Microchip MPLAB IDE ver. 8.3 compiler, and a PICKIT 3 programmer/debugger kit. The reference sine waveform is generated using the built-in PWM module within the microcontroller. The sine reference is stored in a lookup table. The table values are periodically taken from the table and then multiplied by the required amplitude. The resulting value gives the duty cycle of the PWM output. The pointer to the table is incremented by a value which corresponds to the desired output frequency. All the values over one period give a sinusoidal modulated square wave output. If such a signal passes through an LC filter, a pure sine-wave voltage is generated on the inverter output. At every instant of reading sample from the sine table, the program will compute the minimum and maximum phase amplitudes then compute the offset time and then get/apply the new duty which will drive the VSI switches.

6 Simulation Results

The practical system introduced in Sect. 5 is modeled/simulated using MATLAB SIMULINK software package. The MATLAB SIMULINK model is as shown in Fig. 7.

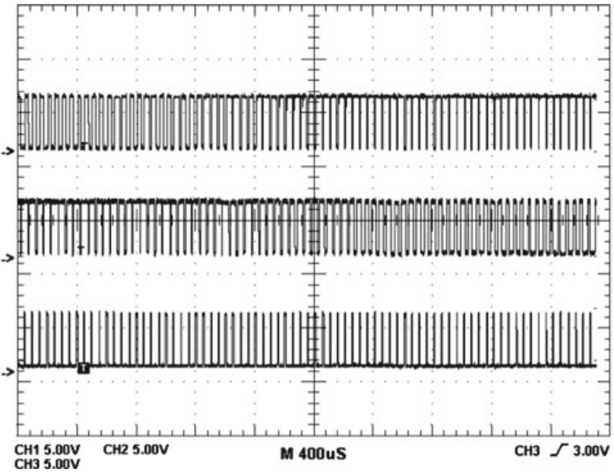


Fig. 11 Experimental SV-PWM gating signals. CH1 = T_{ga}, CH2 = T_{gb}, and the CH3 = T_{gc}, X axis 400 μs/div, Y axis 5 V/div

The simulation is performed under the following conditions: input voltage = 220 VAC, V_{DC} = 620, output voltage fundamental harmonic f = 50 Hz, switching frequency f_{sw} = 20 kHz, and variable modulation index “0.1, 0.85 and 1.15”, also the output of the system is connected to 1.5 KVA 0.7 P.F Inductive load.

The simulation is done for three load modulation indexes. Figure 8 shows the simulation results for 1.5 KVA inductive load with 0.7 power factor at modulation index = 0.1, Fig. 9 shows the simulation results at modulation index = 0.85, and Fig. 10 shows the simulation results at modulation index = 1.15.

From these simulation results we can say that the SVPWM signal generation using the algorithm in Sect. 4 can work in the under-modulation region with some small harmonics in pole and line voltages as shown in Fig. 8, also it can work in the over-modulation region with some small distortions in the output line voltage at the largest modulation index of SVPWM as shown in Fig. 10, and finally it works good in the linear modulation region as shown in Fig. 9.

7 Experimental Results

The algorithm in Sect. 4 is implemented on a microchip PIC18F4431 microcontroller platform and the experimental results are presented for a two-level VSI with output LC filter as shown in Fig. 1. The modulation index is varied from the under-modulation region to the over-modulation region. The experiment is done under the following conditions: DC link = 400 V is used for the inverter, output voltage fundamental harmonic f = 50 Hz, switching frequency f_{sw} = 20 kHz, and a three-phase 1.5 KVA/0.7 power factor load. The experimental results are presented in Figs. 11, 12,

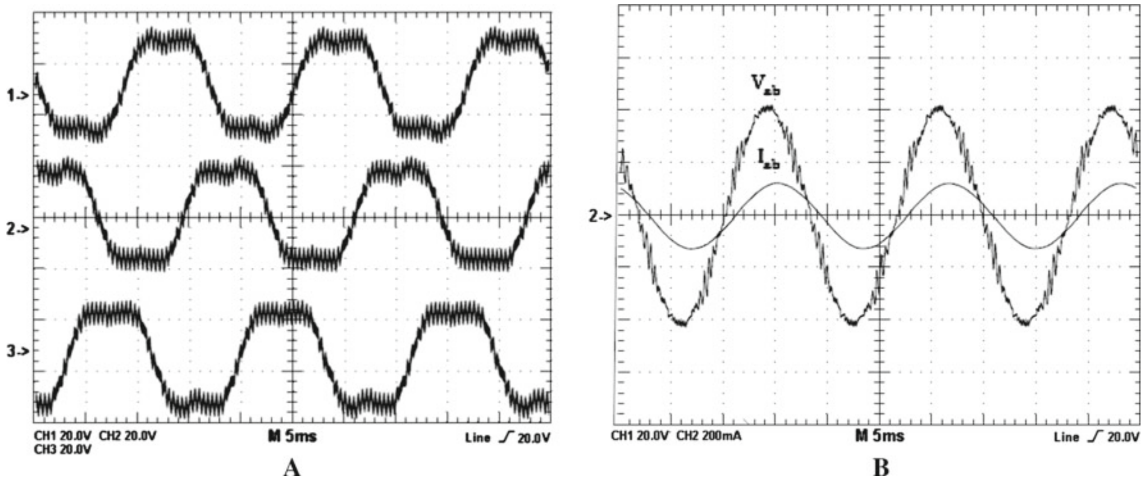


Fig. 12 Experimental results for modulation index = 0.1, **a** pole voltage CH1 = V_{ao}/CH$_2$ = V_{bo}/CH$_3$ = V_{co} waveforms, **B** line voltage CH1 = V_{ab}, line current CH2 = I_{ab} waveforms

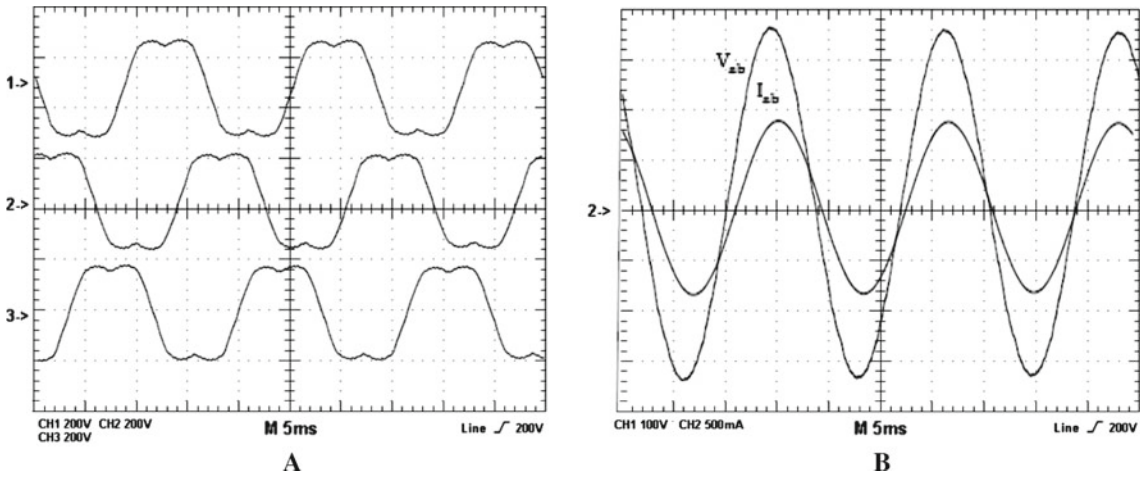


Fig. 13 Experimental results for modulation index = 0.85, **a** pole voltage CH1 = V_{ao}/CH$_2$ = V_{bo}/CH$_3$ = V_{co} waveforms, **b** line voltage CH1 = V_{ab}, line current CH2 = I_{ab} waveforms

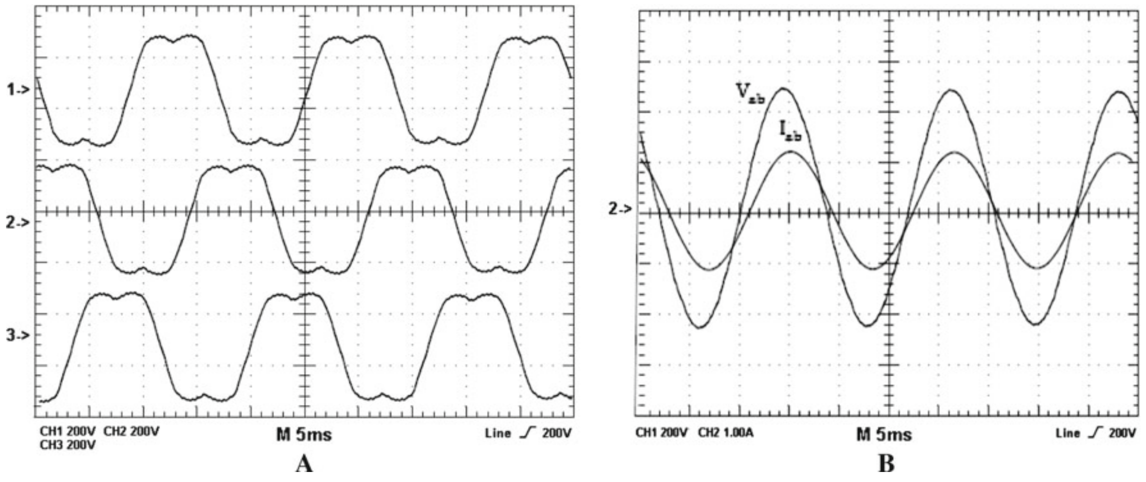


Fig. 14 Experimental results for Modulation index = 1.00, **a** pole voltage CH1 = V_{ao}/CH$_2$ = V_{bo}/CH$_3$ = V_{co} waveforms, **b** line voltage CH1 = V_{ab}, line current CH2 = I_{ab} waveforms

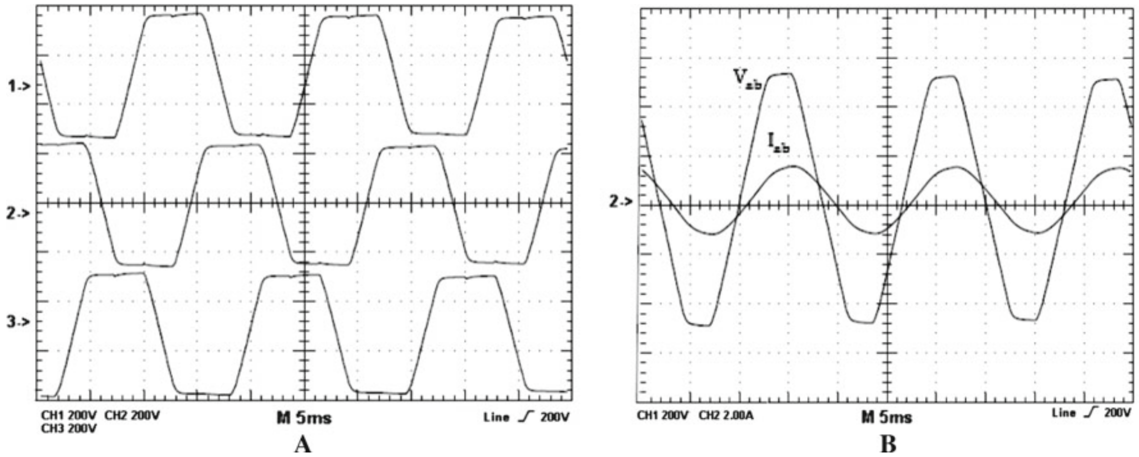


Fig. 15 Experimental results for modulation index = 1.15, **a** pole voltage CH1 = V_{ao}/CH$_2$ = V_{bo}/CH$_3$ = V_{co} waveforms, **b** line voltage CH1 = V_{ab}, line current CH2 = I_{ab} waveforms

Table 2 Summary of measured parameters

Parameter	Load	
	Linear load	Non-linear
Efficiency	93 %	92 %
Output voltage THD without load	0.4 %	
Output voltage THD	1 %	5 %
Power factor	0.99	
Output frequency	50 Hz ± 0.5 %	

13, 14 and 15. Figure 11 shows the generated SVPWM signal used to drive the inverter switches SW1, SW3 and SW5, respectively. Figures 12, 13, 14 and 15 show the pole voltage/line voltage/line current waveforms at modulation index 0.1, 0.85, 1.00, and 1.15, respectively.

The experimental results for the under-modulation region with a modulation index 0.1 are as shown in Fig. 12. Figure 12a shows the pole voltage of three phases, Fig. 12b shows the line voltage and line current for phase A; it may be noted that the appearance of harmonics in voltage waveforms is due to low modulation index.

The experimental results for the modulation region with the modulation indexes 0.85 and 1.00 are as shown in Figs. 13 and 14. The line voltage and line current signals seem to be good with low harmonic and no distortion.

Finally the experimental result for the over-modulation region with a modulation index 1.15 is as shown in Fig. 15. Figure 15a shows the pole voltage of three phases; it may be noted that these waveforms tend to be square waves due to over-modulation. Figure 15b shows the line voltage and line current for phase A, the small distortion within line voltage is due to large modulation index.

All measured parameters for the inverter performance are summarized in Table 2.

The simulation and experimental waveforms are identical. It demonstrates that the simplified implementation of SVPWM is feasible and effectual in driving three-phase two-level inverter, and it is much faster "about 1.66" and more executable than conventional means without lookup tables or complex logical judgments.

8 Conclusions

A SVPWM technique based on a reduced computation method was presented. The SVPWM scheme can drive the inverter gating signals from the sampled amplitudes of the reference phase voltages. The switching vectors for the inverter are derived using a simple digital logic which does not involve any complex computations and hence reduces the implementation time. A practical design and a real implementation for a SVPWM inverter, including MATLAB SIMULINK model, and simulation results for different modulation indexes were also presented. The SVPWM scheme was implemented on a microchip PIC microcontroller 18F4431 platform and the experimental results were presented for a two-level VSI with a three-phase LC filter. The practical results show a good performance with less computation time "about 1.66 faster" and easy software implementation for the presented SVPWM scheme rather than conventional SVPWM.

References

1. Channegowda, P.; John, V.: Filter optimization for grid interactive voltage source inverter. IEEE Trans. Ind. Electron. **157**, 4106–4114 (2010)

2. Yang, S.; Lei, Q.; Peng, F.Z.; Qian, Z.: Robust control scheme for grid interactive voltage source inverter. IEEE Trans. Ind. Electron. **158**, 202–212 (2011)
3. Holtz, J.: Pulse-width-modulation—a survey. IEEE Trans. Ind. Electron. **39**, 410–419 (1992)
4. Bowesand, S.R.; Lai, Y.S.: The relationship between space-vector modulation and regular-sampled PWM. IEEE Trans. Ind. Electron. **44**, 670–679 (1997)
5. Van Der Broeck, H.W.; Skudenly, H.-C.; Stanke, G.V.: Analysis and realization of a pulse width modulator based on voltage space vector. IEEE Trans. Ind. Electron. **24**, 142–150 (1988)
6. Celanovic, N.; Boroyevich, D.: A fast space-vector modulation algorithm for multilevel three-phase converters. IEEE Trans. Ind. Electron. **37**(2), 637–641 (2001)
7. Tekwani, N.; Kanchan, S.; Gopakumar, K.: A dual five-level induction motor drive with common-mode voltage elimination and DC-link capacitor voltage balancing using only the switching-state redundancy—Part I. IEEE Trans. Ind. Electron. **54**(5), 2600–2608 (2007)
8. Seixas, P.F.; Severo Mendes, M.A.; Donoso Garcia, P.; Lima, A.M.N.: A space-vector PWM method for three-level voltage source inverters, vol. 1. In: Proceedings of IEEE APEC, pp. 549–555 (2000)
9. Yao, W.; Hu, H.; Lu, Z.: Comparisons of space-vector modulation and carrier-based modulation of multilevel inverter. IEEE Trans. Ind. Electron. **23**(1), 45–51 (2008)
10. Somasekhar, V.T.; Gopakumar, K.; Baiju, M.R.; Mohapatra, K.K.; Umanand, L.: A multilevel inverter system for an induction motor with open-end windings. IEEE Trans. Ind. Electron. **52**(3), 824–836 (2005)
11. Shivakumar, E.G.; Gopakumar, K.; Sinha, S.K.; Pittet, A.; Ranganathan, V.T.: Space-vector PWM control of dual inverter fed open-end-winding induction motor drive. In: Proceedings of the IEEE APEC, pp. 399–405 (2001)
12. Somasekhar, V.T.; Srinivas, S.; Kumar, K.K.: Effect of zero-vector placement in a dual-inverter fed open-end-winding induction motor drive with a decoupled space-vector PWM strategy. IEEE Trans. Ind. Electron. **55**(6), 2497–2505 (2008)
13. Shivakumar, E.G.; Gopakumar, K.; Sinha, S.K.; Pittet, A.; Ranganathan, V.T.: Space-vector PWM control of dual inverter fed open-end-winding induction motor drive. In: Proceedings of the IEEE APEC, pp. 399–405 (2001)
14. Gopinath, A.; Mohamed A.S.A.; Baiju, M.R.: Fractal based space vector PWM for multilevel inverters—a novel approach. IEEE Trans. Ind. Electron. **56**(4), 1230–1237 (2009)
15. Holmes, D.G.: The general relationship between regular sampled pulse width modulation and space vector modulation for hard switched converters. In: Conf. Rec. IEEE Industry Applications Society (IAS) Annual Meeting, pp. 1002–1009 (1992)
16. Baiju, M.R., Mohapatra, K.K., Somasekhar, V.T., Gopakumar, K., Umanand, L.: A five-level inverter voltage space phasor generation for an open-end winding induction motor drives. IEE Proc. Electr. Power Appl. **150**(5), 531–538 (2003)
17. Kim, J.; Sul, S.: A novel voltage modulation technique of the Space Vector PWM'. In: Proc. Ind. Power Electronics Conf., Yokohama, Japan, pp. 742–747 (1995)
18. Baiju, M.R.; Gopakumar, K.; Somasekhar, V.T.; Mohapatra, K.K.; Umanand, L.: A space-vector-based PWM method using only the instantaneous amplitudes of reference phase voltages for three-level inverters. EPE J. **13**(2), 35–45 (2003)
19. Kanchan, R.S.; Baiju, M.R.; Mohapatra, K.K.; Ouseph, P.P.; Gopakumar, K.: Space-vector PWM signal generation for multilevel inverters using only the sampled amplitudes of reference phase voltages. Proc. Ind. Elect. Eng. Elect. Power Appl. **152**(2), 297–309 (2005)
20. van der Broeck, H.W.; Skudelny, H.Ch.; Stanke, G.: Analysis and realization of a pulse width modulator based on voltage space vectors. IEEE Trans. Ind. Electron. **24**, 142–150 (1988)