Criteria for Implementing Space Vector PWM Modulators Using DSP

José Torrico Altuna
torrico@dsce.fee.unicamp.br

Zanoni Dueire Lins
zanoni@dsce.fee.unicamp.br

Edson Bim
bim@dsce.fee.unicamp.br

UNICAMP/FEEC/DSCE
CEP. 13081-970, Campinas, SP – Brasil

***Abstract* - The paper presents implementation strategies of Space Vector Pulse Width Modulation (SVPWM) based on current Digital Signal Processors (DSPs). The modulator performance is analysed targeting to minimise execution time without sacrificing harmonic motor currents. Simulation and experimental tests illustrate the performance of the PWM waveforms for a three-phase inverter. Practical considerations for implementing real time SVPWM are discussed. Experimental implementation is performed using a development system based on the DSP96002 digital signal processor.**

I. Introduction

The strategy of Space Vector Pulse Width Modulation (SVPWM) is nowadays frequently used in drives with small and medium power. The reason of this success is mainly due to the great performance comparing with conventional methods. In other side, technological advances allow to design compact, flexible and efficient modulators that have easy integration with the other parts of the control system. Although the modulator has been accepted and used largely, the implementation strategies have not been clearly defined and systematized. The initial hardware complexity SSI and MSI has been replaced by powerful VLSI integrated circuits.

Analysing the first works it can be observed the great effort to implement this modulator: in 1987 Holtz J. et al.[1] used Discrete-Digital hardware (SSI and MSI) and ROM memories to store the modulation times; in 1988, Van Der Broeck H. W. et al.[2] combined hardware and software using Intel 8086 processor to implement the vectorial modulator with low switching frequency. In recent years, ASIC technology has been used to minimize the execution time of the processor, but with very expensive development costs [3]. Technologies of programmable integrated circuits (CPLD or FPGA) offer currently decreasing costs of development systems and integrated circuits. Therefore, they are a good option for implementation in laboratory [4] although, they need additional external hardware to connect with CPU. In parallel with the preceding alternatives, Digital Signal Processors (DSPs) present an excellent option to implement both the modulator and the control system allowing their easy integration. The development of semiconductor technology has increased considerably: the MIPs, the parallelism and the peripherals built in the DSP, besides of decreasing chip size and manufacturing costs.

This article shows a real time implementation of the space vector PWM modulator that uses a digital signal processor DSP96002 from Motorola, establishing criteria for implementation and, bringing out the relevant characteristics of the system.

II. Space vector pulse width modulation

The SVPWM is based on vector representation of three-phase stator voltages in the $\alpha\beta$ stationary reference frame. A voltage space vector represents the three machine voltages. There are eight states available according to the position of the inverter switches. The Figure 1 shows the three-phase inverter, the switching states are obtained considering the state of each upper leg switch being the complement of each lower leg switch.

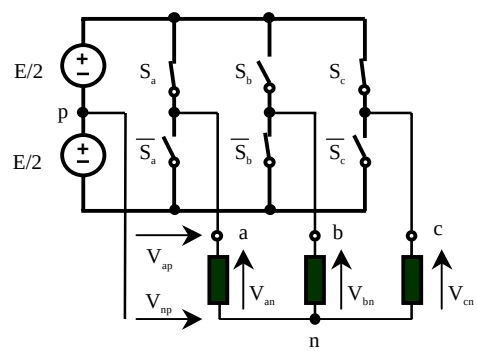


Fig. 1. Three-phase power converter diagram

The plane is divided into six sectors which are numbered I, II, …,VI and limited by vectors V_1, V_2, V_3, V_4, V_5 e V_6 with equal voltage magnitudes. The switching state vectors describe the inverter output voltages. Zero vector V_0 is associated to those inverter states with all lower half-bridge switches closed and the zero vector V_7 to all upper. The three machine terminals are then short-circuited, and the voltage vector assumes zero magnitude. The Figure 2 shows the position of these vectors on the $\alpha\beta$ reference frame.

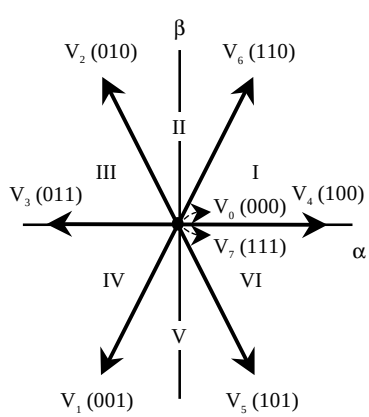


Fig. 2. Switching state vectors in the $\alpha\beta$ plane

The strategy of vector modulation defines a space vector reference voltage V* which is assumed constant during the switching period. The reference vector is sampled at the fixed clock frequency f_s (switching frequency) for symmetrical modulation or is sampled at the fixed clock frequency $2f_s$ for asymmetrical modulation. To expect an optimum PWM modulation, the switching period and the ripple of the current vector must be as small as possible. These conditions are carried out if two adjacent vectors to the reference vector V* are chosen. The combination in time of these vectors generates an average vector that follows the reference vector. The Figure 3 illustrates the procedure mentioned: to generate an equivalent average vector is necessary to switch the vectors V_A and V_B proportionally at angle formed with V* (θ to V_A and (60-θ) to V_B). For the first sector, V_A corresponds to the vector V_4 and V_B corresponds to the vector V_6.

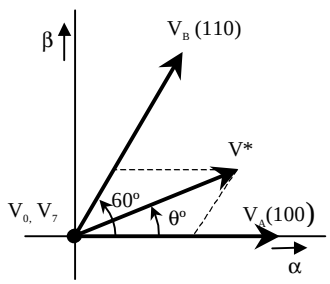


Fig. 3. Switching state vectors of the first sector

Expressing this property mathematically:

$$T_A V_A + T_B V_B = V^* . \left(\frac{T_S}{2}\right) \quad (1)$$

and placing the equation (1) in complex form, it follows that:

$$T_A |V_M| e^{j0} + T_B |V_M| e^{j60} = |V_M| a \left(\frac{T_s}{2}\right) e^{j\theta} \quad (2)$$

Where: $0° \le \theta \le 60°$, $a = \frac{|V^*|}{|V_M|}$ and $|V_M|$ is the module of voltage space vector obtained from a three-phase to two-phase transformation.

Real and complex parts of the equation (2), yields:

$$T_A = \frac{T_s}{2} . a . \frac{\text{sen}(60° - \theta)}{\text{sen}\, 60°} \quad (3)$$

$$T_B = \frac{T_s}{2} . a . \frac{\text{sen}\, \theta}{\text{sen}\, 60°} \quad (4)$$

$$T_0 = T_7 = \frac{T_s}{2} - T_A - T_B \quad (5)$$

For the sectors II-VI are applied the same equations.

Considering the linear operation zone, the value of "a" varies between 0 and $\sqrt{3}/2$. The modulation index is defined as $m = (\pi/3).a$ and its variation is linear in the interval $0 \le m \le 0.907$.

III. Implemented System

The development system consists of a personal computer (PC), the Motorola Application Development Software (ADS) and the Application Development Module (ADM) that contains the DSP chip. The communication between PC and the ADM hardware is done through a parallel port. This windows driven system is useful to debug easily hardware and software environments based on DSP96002.

The SVPWM is integrally implemented using DSP96002 assembly language. Two interrupt routines are used: The DSP timer interrupt for triggering the output of IGBTs command pulses and the external hardware interrupt routine (IRQA) for incrementing the angle of the voltage reference vector.

A. Switching state vectors generation

The Figure 4 shows the interrupt request instants for one switching period $T_s=1/f_s$ corresponding to the sector I. S_a, S_b, and S_c are the command signals of inverter upper switch. The switch pairs are complementary. The DSP timer is programmed free running to achieve more precision in generation of the switching signals. Inside of one switching period T_s, the DSP executes eight routines synchronised with zero value of timer counter. Beginning each interrupt, a new value is loaded into the time constant register corresponding to the time of the next switching vector. In the Figure 4, the signal Int indicates the instant where each routine is executed. The routine 8 calculates the times $T_0/2$, T_A, T_B and $T_7/2$ for the next switching period.

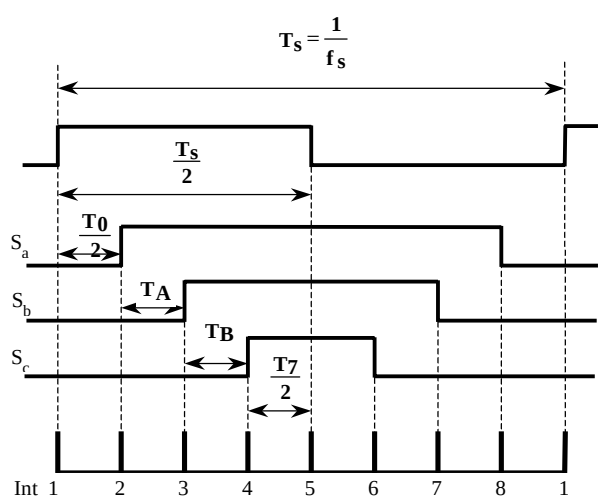


Fig. 4. Interrupt instants for the sector I

Each timer interrupt has two main functions: send the switch state vector to the inverter and load the time constant register for the next interrupt. The switch state

vector is obtained from the sector and the interrupt number of the timer as stated in Table I.

TABLE I
VECTORS SEQUENCE

Sector	Interruption 1	2	3	4	5	6	7	8
I	V_0	V_4	V_6	V_7	V_7	V_6	V_4	V_0
II	V_7	V_6	V_2	V_0	V_0	V_2	V_6	V_7
III	V_0	V_2	V_3	V_7	V_7	V_3	V_2	V_0
IV	V_7	V_3	V_1	V_0	V_0	V_1	V_3	V_7
V	V_0	V_1	V_5	V_7	V_7	V_5	V_1	V_0
VI	V_7	V_5	V_4	V_0	V_0	V_4	V_5	V_7

For each sector, the voltage vectors V_A and V_B are obtained from adjacent vectors to the reference vector. In the interrupt routines 1, 4, 5 and 8, the vectors V_0 and V_7 are chosen to achieve minimum switching. Thus, there is one bit of difference between two consecutive switching states.

The second function of the interrupt routines is load the time constants of the timer with the values T_0, T_A, T_B and T_7, where each time value is associated with an angle between 0° and 60°. These values are calculated according equations (3), (4).and (5).

The flowchart of the timer interrupt routine is shown in Figure 5.

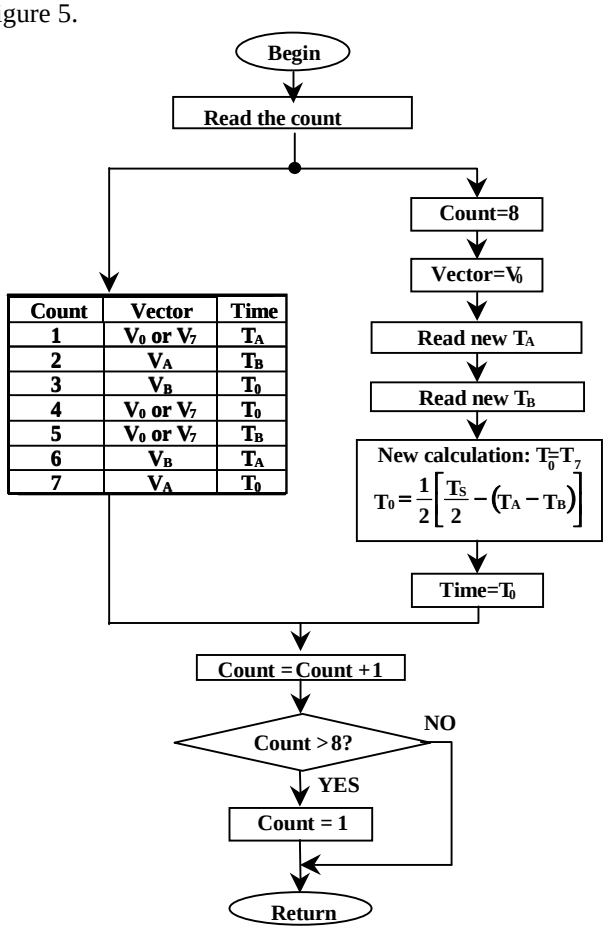


Fig. 5. Routine to generate the switching vectors

A software counter is implemented to choose the voltage vector that must be switched and the time constant that will be used in the next interrupt.

The main program waits for an interrupt in an infinite loop. When the current interrupt finishes, the execution backs to the waiting loop.

Knowing the sector (supplied by angle and sector updating routine) and the interrupt number, the voltage vector is chosen from the Table I. The time constants T_A and T_B are generated from the Table II loaded in RAM memory. With an eye to never loose any clock cycle, the timer is never restarted and the time constants are always loaded in the previous interrupt. In the interrupt 8, the time constants T_A, T_B and T_0 are updated. These times will be used in the next switching period using the angles obtained from the IRQA interrupt.

B. Angle and sector updating

To avoid large processing time, the switching instants for the vectors V_A and V_B respectively are calculated off-line. Therefore, the tables with sin(59.5-θ) and sin(θ) are placed into the X and Y memories with 120 angles (θ = 0.0°, 0.5°,..., 59.5°). The Table II shows the loaded values in these memories which are used to update the times T_A and T_B in each switching period.

TABLE II
SINE FUNCTIONS LOADED IN RAM MEMORY

Angle θ	X Memory Sin (59.5-θ)	Y Memory Sin θ
0°	Sin 59.5°	Sin 0°
0.5°	Sin 59°	Sin 0.5°
1°	Sin 58.5°	Sin 1°
. . .	. . .	. . .
59.5°	Sin 0°	Sin 59.5°

The external interrupt IRQA is used to update the angle and sector. The Figure 6 shows the IRQA flowchart.

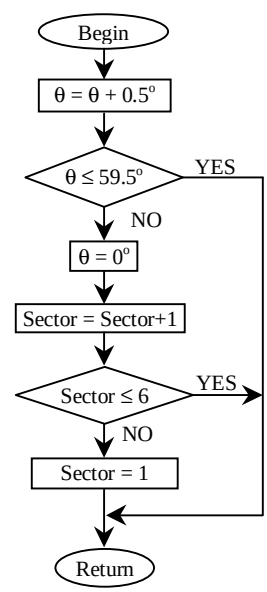


Fig. 6. Angle and Sector updating routine

Every execution of the interrupt routine, the memory content that stores the angle value is incremented of 0.5º. When the value of the angle reaches 60º, the sector is updated and the angle is started again with 0º.

IV. Simulation and Experimental Results

To study the modulator behaviour, a simulation was carried out considering the real conditions of the system. All calculations were made with a 50ns step size (Instruction cycle of the DSP96002).

Eight ramps with reset capability were built to generate the PWM waveforms. A look up table is emulated by sampling 120 values of the first sector.

The Figure 7 shows simulated waveforms of V_{ap}, V_{bp} and V_{cp} , for modulation index m=0.5, switching frequency 4kHz and fundamental frequency current of 60Hz.

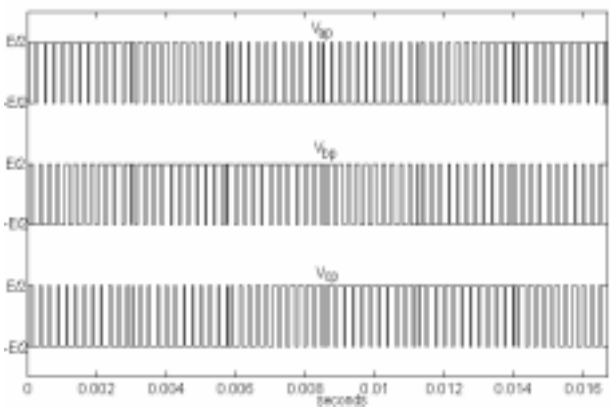


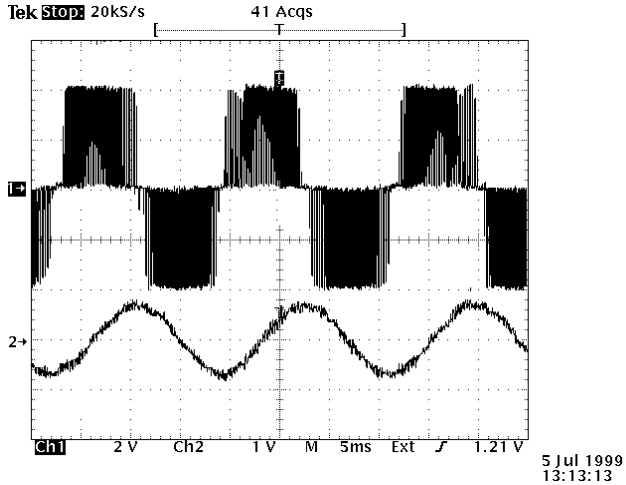
Fig. 7. Simulated results: phase voltages

Experimental tests were carried out to characterise the performance of the Space Vector Modulator. The laboratory tests were performed with an induction motor: 3Hp, 60Hz, 220/380 Δ/Y, rotor cage.

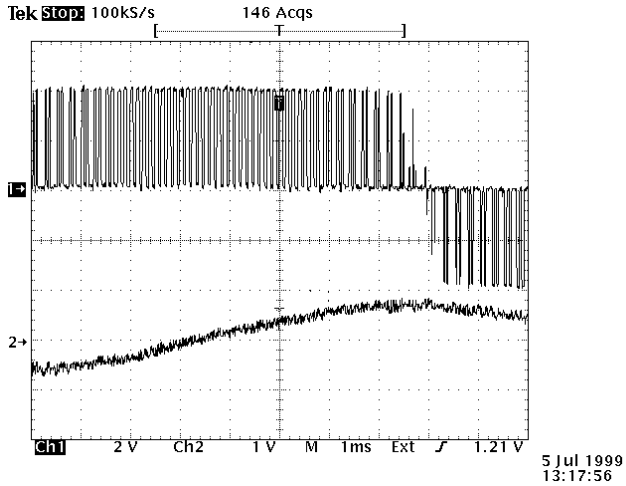
The power inverter is mounted with SKM40 GDL 123D IGBT's obtained from SEMIKRON. The command pulses are sent to the inverter through the integrated circuits SKHI 22 that provides isolation, interlock circuit, short circuit protection and dead time compensation.

The switching frequencies were programmed to be 2.5 or 4 kHz. The tests were made in open loop by controlling the fundamental frequency with IRQA signal. The switching frequency for all experimental tests was 4 kHz and the modulation index was m=0.5.

The Figure 8 shows the 60 Hz line voltage and the current waveforms when no load is applied to the motor. The line voltage appears distorted due to the frequency response of voltage sensors.



(a)



(b)

Fig. 8 Voltage V_{ab} and current I_a – 60Hz
(Vertical scale: 155V/div-channel 1 and 2.2A/div- channel 2)

The Figure 9 presents the measure of the 60 Hz line voltage waveform from oscilloscope without using sensors.

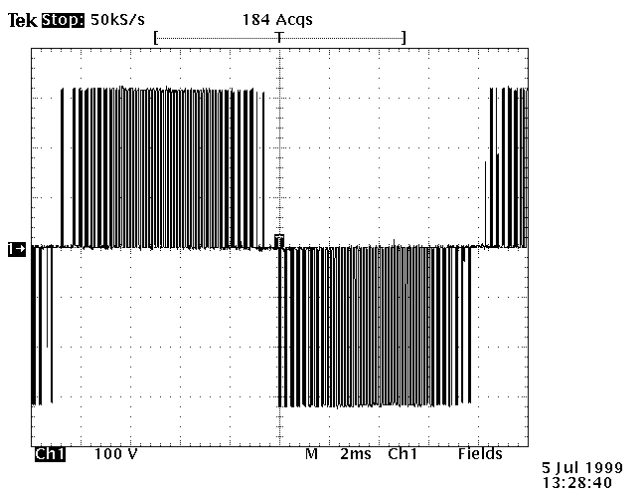


Fig. 9. Voltage V_{ab} before the hall sensors– 60Hz

The Figure 10 shows the 30 Hz line voltage and the current when no load is applied to the motor.

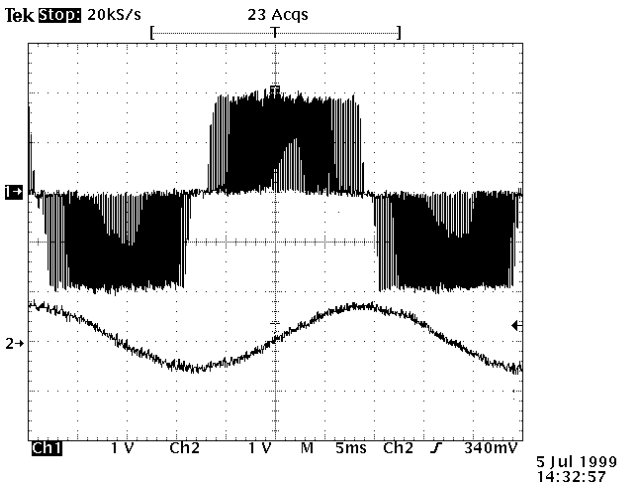


Fig. 10. Voltage V_{ab} and current I_a – 30Hz
(Vertical scale: 77.5V/div-channel 1 and 2A/div- channel 2)

The Table III shows the relative results of the processor time consumption to execute the SVPWM.

TABLE III
RELATIVE EXECUTION TIMES TO GENERATE THE SVPWM

fs (kHz)	Timer Routine (%)	IRQA Routine (%)	DSP (%)
2.5	3.42	4.21 (30Hz)	7.63
		8.42 (60Hz)	11.84
4	5.48	4.21 (30Hz)	9.69
		8.42 (60Hz)	13.9

It can be observed for 4 kHz switching frequency and 60Hz fundamental current frequency a 13.9% DSP processing time.

V. CONCLUSIONS

It was implemented an efficient modulation strategy based on space vectors with low CPU consumption using only one interrupt routine.

Commercial microcontrollers and DSPs with built-in timer could use this strategy without lost of system performance.

The algorithm is easily integrated with motor control routines. The modulator parameters such as modulation index and switching frequency could be changed on-line.

VI. ACKNOWLEDGEMENT

The authors acknowledge the financial support provided by São Paulo State Foundation for Research Support (FAPESP) for the development of this research through contract N° 96/11923-5.

VII. REFERENCES

[l] J. Holtz, P. Lammert and W. Lotzkat. “High-speed drive system with ultrasonic MOSFET PWM inverter and single-chip microprocessor control”, *IEEE Transactions on Industry Applications*, vol. IA-23, no. 6, pp. 1010-1015, Nov/Dec 1987.

[2] H. W. Van Der Broeck, H. C.Skudelny, and G. V. Stanke “Analysis and Realization of a Pulsewidth Modulator Based on Voltage Space-Vectors”, *IEEE Transactions on Industry Applications*, vol. 24, no. 1, pp. 142-150, Jan/Feb 1988.

[3] M. G. Egan, J.M. Murphy, E.J. Heffernan, S.U. Lidholm, and M. L. McGrath, “An ASIC-based PWM waveform generator for AC motor control applications”, in IEEE Int. Symp. On Circuits and Systems. Proc., vol.2, 1988, pp.1369-1372.

[4] Tzou Y. Y., Hsu H. J. , “FPGA Realization of Space-Vector PWM Control IC for Three-Phase PWM Inverters”, *IEEE Transaction on PowerElectronics*, Vol. 12, no. 6, Nov. 1997.