DPWM-S3 Software Control for Three-phase Inverters

Cristian Aghion[1], Mihai Lucanu[2], Ovidiu Ursaru[3]
[1,2,3] "Gh. ASACHI" Technical University of Iasi Faculty of Electronics, Telecommunications and Information Technology
Bvd. Carol I, nr.11, RO-700506, Iasi
aghion@etc.tuiasi.ro

***Abstract* — In this paper we describe a DPWM-S3 software control method implemented on a microcontroller. This technique was tested by simulations and in practice on a three-phase inverter with an AC motor (0.37KW) as output load.**

I. Introduction

DPWM-S3 [1] command technique is a high performance computational control method, having as a main advantage the reduction of the number of commutations of the power transistors within the three-phase inverter, as compared to other modulation techniques, e.g. Sinusoidal Pulse Width Modulation (SPWM) [2], Space Vector Modulation (SVM) [3], etc. The number of commutations is reduced as, considering equations (1) for four time intervals (out of twelve), some transistors are in continuous conduction, and power losses in commutation should be in theory by $\frac{1}{3}$ smaller if SPWM or SVM techniques were used. With a reduced number of commutations, power losses on the transistors are decreasing, caloric power is decreasing too and, as a result, smaller heat-sinks are needed on the same transistors.

II. Theoretical Considerations Regarding The Modulation Signal

The main equations of the DPWM-S3 signal are presented in (1) and the mathematical representation is shown in Figure 1.

$$s_3 = \begin{cases} \sqrt{3}m_a \cos\omega_m t + m_a \sin\omega_m t - 1\,; & 0 \le \omega_m t \le \pi/6 \\ 1; & \pi/6 \le \omega_m t \le \pi/3 \\ \sqrt{3}m_a \cos\omega_m t - m_a \sin\omega_m t + 1; & \pi/3 \le \omega_m t \le \pi/2 \\ \sqrt{3}m_a \cos\omega_m t + m_a \sin\omega_m t - 1; & \pi/2 \le \omega_m t \le 2\pi/3 \\ -1; & 2\pi/3 \le \omega_m t \le 5\pi/6' \\ \sqrt{3}m_a \cos\omega_m t - m_a \sin\omega_m t + 1; & 5\pi/6 \le \omega_m t \le \pi \\ \sqrt{3}m_a \cos\omega_m t + m_a \sin\omega_m t + 1 & \pi \le \omega_m t \le 7\pi/6 \\ -1; & 7\pi/6 \le \omega_m t \le 4\pi/3 \\ \sqrt{3}m_a \cos\omega_m t - m_a \sin\omega_m t - 1; & 4\pi/3 \le \omega_m t \le 3\pi/2 \\ \sqrt{3}m_a \cos\omega_m t + m_a \sin\omega_m t + 1; & 3\pi/2 \le \omega_m t \le 5\pi/3 \\ 1; & 5\pi/3 \le \omega_m t \le 11\pi/6 \\ \sqrt{3}m_a \cos\omega_m t - m_a \sin\omega_m t - 1; & 11\pi/6 \le \omega_m t \le 2\pi \end{cases} \quad (1)$$

Figure 2 shows the waveforms of the modulator signals (s_{1a}, s_{1b} and s_{1c}) obtained for the DPWM-S3 technique and the control signals for all six transistors within the power inverter.

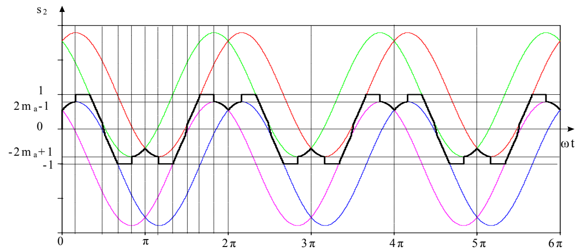

Figure 1. The modulator signal DPWM-S3

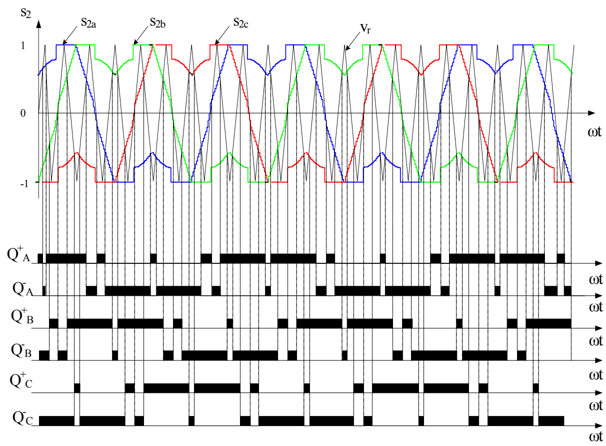

Figure 2. Waveforms of the modulator and control signals

Figure 3 shows the power inverter schematics used for the simulation.

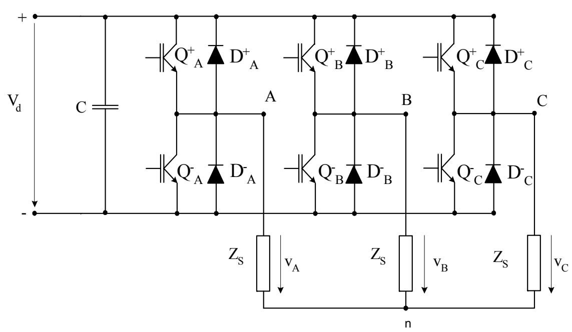

Figure 3. Power inverter architecture

III. Command and Control Algorithm

The DPWM-S3 software takes into consideration equations (1) and is suited for C8051F120 microcontroller made by Silicon Laboratories. The main flowchart of the control software is shown in Figure 4a. Figure 4b shows the flowchart of the T3 Timer interruption (Service Routine Interruption), where the main equations are solved.

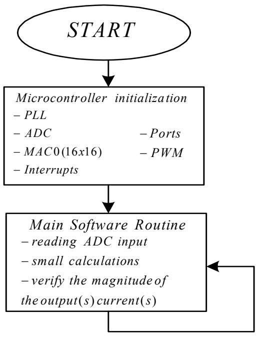

Figure 4a. Software control flowchart

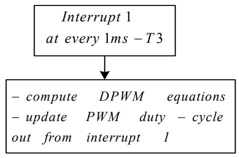

Figure 4b. Flowchart of Interrupt 1

The first step is the microcontroller initialization (Input/Output Ports, ADC, PLL, Timers, PWM, etc.), the second step is the execution of the main software routine, written in C language, where the value of the Analog to Digital Converter is continually read. This value is useful for the calculation of the magnitude of the modulator wave for the desired signal. The software also contains an interrupt routine:

Interrupt 1 appears at every 1ms because the Timer T3 surpasses itself. In this routine all three command signals are computed (with 120° phase-shift between them); there are three signals because it is necessary to command transistors Q_A^+, Q_B^+ and Q_C^+ , and for Q_A^-, Q_B^- and Q_C^- transistors, the command signals are obtained by the complementation of the original signals.

IV. Software Control

The command software is written in C language, applied to C8051F120 microcontroller. The software routine for the calculation of the PWMA signals (s_{1a}, s_{1b} and s_{1c}) is shown below. It should be taken into consideration that all these calculations are made in Interrupt 1 routine (having a lower priority), which is executed at every 1ms when Timer T3 surpasses itself. The necessary computing time is approximately 400µs, performance obtained using MAC 16x16 (Multiply and Accumulate) block and the high working speed of the microcontroller (100MHz-100MIPS / millions of instructions per second).

```
void Timer3_ISR (void) interrupt 14   // at every 1ms
{
        unsigned char SFRPAGE_SAVE = SFRPAGE;
        gigi();
        SFRPAGE = TMR3_PAGE;
        TF3  = 0;
        SFRPAGE = SFRPAGE_SAVE;
}
//---------------------------------------------------
// Compute modulation signals 1
//---------------------------------------------------
void gigi(void)
{
        signed char s;          // signed sine
        unsigned char o;        // output value
        unsigned int p;         // 16 bit product

        unsigned char SFRPAGE_SAVE = SFRPAGE;

        float f0, f1, f2, tr0, tr1, tr2;
        Sum += (freq << 6);
        q0 = (Sum >> 8);
        tr0 = (q0/40.6);

                if(!reverse)
                {
                        tr1 = tr0 + xx1;
                        tr2 = tr0 + xx2;
                }
                else
                {
                        tr2 = tr0 + xx1;
                        tr1 = tr0 + xx2;
                }
SFRPAGE   = PCA0_PAGE;
f0 = lac(tr0);
s = f0*0x7F;
p = amplitude * (signed int)s;       //multiply by v
o = p>>8;                    // throw away low byte
o += 0x80;                   // center sinewave at 50%
PCA0CPH0 = o;
f1 = lac(tr1);
s = f1*0x7F;
p = amplitude * (signed int)s;       //multiply by v
o = p>>8;                    // throw away low byte
o += 0x80;                   // center sinewave at 50%
PCA0CPH1 = o;
f2 = lac(tr2);
s = f2*0x7F;
p = amplitude * (signed int)s;       //multiply by v
o = p>>8;                    // throw away low byte
o += 0x80;                   // center sinewave at 50%
PCA0CPH2 = o;
SFRPAGE = SFRPAGE_SAVE;     }

//---------------------------------------------------
// Compute DPWM-S3 signal
//---------------------------------------------------

float lac(float tr)
{
        float iti;
        if(tr<0.5235 && tr>0)
                {
                iti = 1.732*ma*cos(tr)+ma*sin(tr)-1;
                }
        if(tr<1.0471 && tr>=0.5235)
                {
                iti = 1;
                }
        if(tr<1.5707 && tr>=1.0471)
                {
                iti = 1.732*ma*cos(tr)-ma*sin(tr)+1;
                }
        if(tr<2.0943 && tr>=1.5707)
                {
                iti = 1.732*ma*cos(tr)+ma*sin(tr)-1;
                }
```

```
        if(tr<2.61799 && tr>=2.0943)
                {
                iti = -1;
                }
        if(tr<3.1415 && tr>=2.61799)
                {
                iti = 1.732*ma*cos(tr)-ma*sin(tr)+1;
                }
        if(tr<3.6651 && tr>=3.1415)
                {
                iti = 1.732*ma*cos(tr)+ma*sin(tr)+1;
                }
        if(tr<4.1887 && tr>=3.6651)
                {
                iti = -1;
                }
        if(tr<4.7123 && tr>=4.1887)
                {
                iti = 1.732*ma*cos(tr)-ma*sin(tr)-1;
                }
        if(tr<5.2359 && tr>=4.7123)
                {
                iti = 1.732*ma*cos(tr)+ma*sin(tr)+1;
                }
        if(tr<5.7595 && tr>=5.2359)
                {
                iti = 1;
                }
        if(tr<6.28 && tr>=5.7595)
                {
                iti = 1.732*ma*cos(tr)-ma*sin(tr)-1;
                }

        return iti;
}
```

V. Simulation And Experimental Results

After implementing the above software on the microcontroller, the signal s1a is obtained and shown in Figure 5.

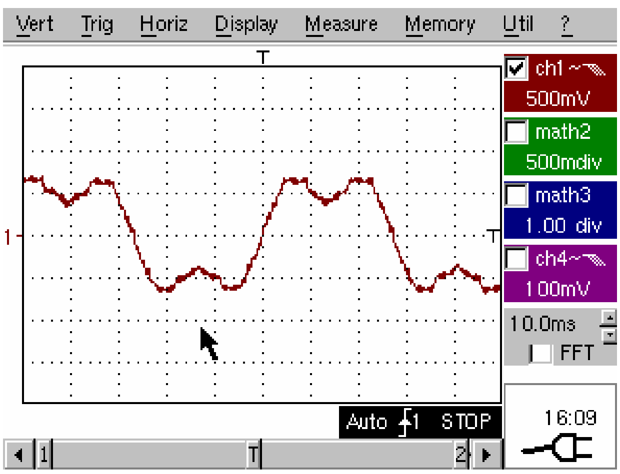

Figure 5. Modulation signal s_{1a} obtained by measurement

Figure 6 shows the waveforms of the command signals for Q_A^+ *and* Q_B^+ transistors within the three-phase inverter. It is important to mention that the modulation was a triangular modulation wave with 17.25KHz carrier frequency.

Figure 6. Waveforms from the oscilloscope of the command signals for Q_A^+ *and* Q_B^+ transistors.

Considering that the supply voltage of the three-phase is 305V and the output impedance is of 0.37 KW for the simulations and on the practical stand, we obtain the following.

Figure 7 shows on the left side the line to line voltage between A and B phases read on the oscilloscope, and the harmonic content of this voltage is presented below. On the right side, is presented the line to line voltage between A and B phases, obtained by simulations, and the harmonic spectrum of the line to line voltage between A and B phases, obtained also by simulations.

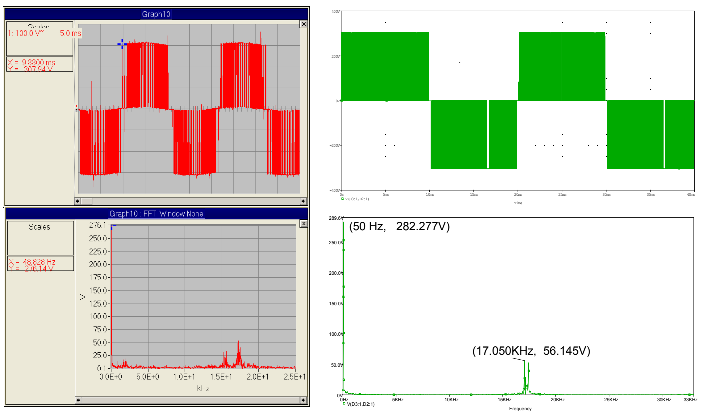

Figure 7. – left side: waveforms and harmonic spectrum of the line to line voltage, obtained from oscilloscope
– right side: waveforms and harmonic spectrum of the line to line voltage, obtained by simulations

Figure 8 shows on the left side the phase voltage A, read on the oscilloscope, and the harmonic content of this voltage is presented below. On the right side, is presented the phase voltage A, obtained by simulations, followed by the harmonic spectrum of the phase voltage A, obtained by simulations.

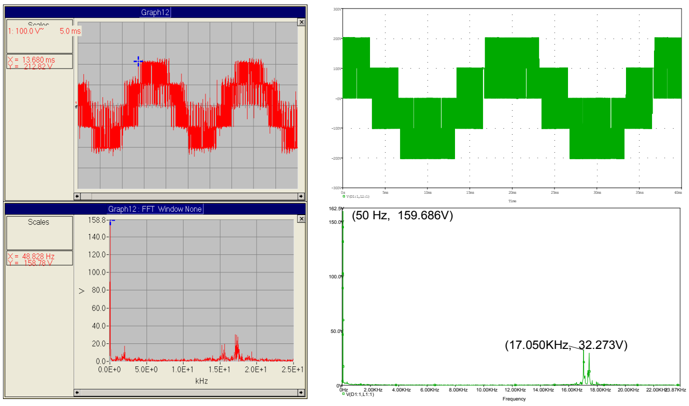

Figure 8. – left side: waveforms and harmonic spectrum of the phase voltage A, obtained from oscilloscope.
– right side: waveforms and harmonic spectrum of the phase voltage A, obtained by simulations.

Figure 9 shows on the left side the phase current A, read on the oscilloscope, and the harmonic content of this current is presented below. On the right side is presented the phase current A, obtained by simulations, followed by the harmonic spectrum of the phase current A, obtained by simulations.

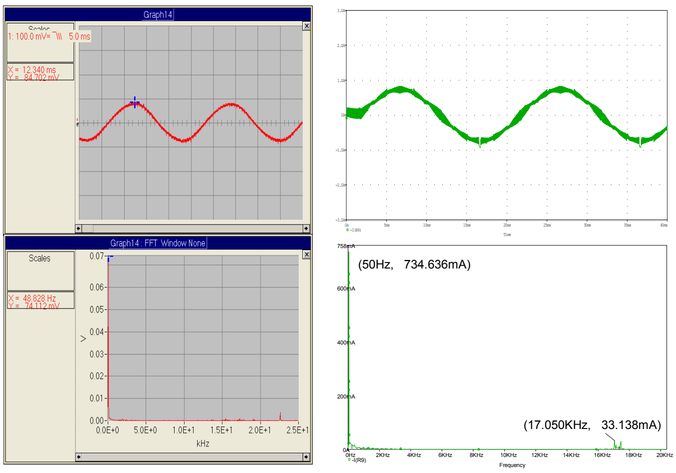

Figure 9. – left side: waveforms and harmonic spectrum of the phase voltage A, obtained from oscilloscope
– right side: waveforms and harmonic spectrum of the phase voltage A, obtained by simulations.

VI. Conclusion

This paper aims at implementing the DPWM-S3 algorithm using a microcontroller for three-phase inverter command. Although this DPWM-S3 technique is of high performance, we cannot say that this technique is the best (considering also the frequency response of the output voltage), but it offers a great advantage as it determines a reduction of the number of commutations of the power transistors within the three-phase inverter. This will lead to small power losses, increasing the efficiency of the inverter.

References

[1] Aghion C, Ursaru O, Lucanu M, Dimitriu L, Vornicu L, „Software control for PWMA (Angular Pulse Width Modulation)", International Symposium on Signals Circuits and Systems – ISSCS 2007, Iasi, Romania, July 12-13, 2007, Volume 2 of 2, pp. 429-432, IEEE Catalog Number 07EX1678, ISBN 1-4244-0968, Library of Congress 2007920356.

[2] Hava A. "Carrier based PWM-VSI drives in the overmodulation region", PhD Thesis, University of Wisconsin – MADISON, 1998.

[3] Bose K. B., ''Microcomputer Control of Power Electronics and Drive'' IEEE Press. New York, 1987

[4] Hava A., Kerkman R., Lipo T. "AHigh-performance generalized discontinuous PWM algorithm" High Performance Generalized Discontinuous Pwm Algorithm. Ieee Transactions On Industry Applications, 34(5), 1059-1071, 1998.

[5] Alexa D., Hrubaru O.,''Aplicatii ale Convertoarelor Statice de Putere'', Ed Tehnica, Bucuresti,1989.

[6] Bose K. B., ''Tehnology Trends in Microcomputer Control of Electrcal Machines'' IEEE Trans. On IE , vol.1E-35, NO.2, pp. 160-177,1992.

[7] Aghion C., Ursaru O., Vornicu L., "BLDC Software Control", Trans& Motauto'05+, Twelfth International Scientific-Tehnical Conference on Transport, Technologies and Military and Military-Educational Problems, 23 - 25 November 2005, Veliko-Tarnovo, Bulgaria , vol.4, pp. 84-87.

[8] Dimitriu L., Lucanu M., Aghion C., Ursaru O., '' Control with Microcontroller for PWM Single-Phase Inverter ''International Symposium on SCS 2003 ,vol I, pp. 265-269, IASI.

[9] Motorola – AN 1857, A 3-Phase ac Induction Motor Control System Based on the MC68HC908MR32.

[10] Bowes S. R., Clark P.R.,''Simple Miroprocessor Implementation of New Regular-Sampled Harmonic Elimination PWM Tehniques'', IEEE Trans. On IAS , vol.28, NO.1, pp. 89-95,1992.

[11] Hava A., Kerkeman R.J., Lipo T.A., „A Hight Performance Generalized Discontinuous PWM Algorithm„ ''IEEE Trans. On Ind. Appl, vol. 34, no. 5, pp1059-1071, 1998

[12] ***C8051F120 datasheet microcontroller, Silicon Laboratories – Texas, Austin, 2004.

[13] ***80C51 Family Architecture, Philips Semiconductors, Sunnyvale, California, U.S.A., March 1995.

[14] A. Hava, R. J. Kerkman and T. A. Lipo, " Simple Analytical and Graphical Methods for Carrier based PWM VSI Drives, *IEEE Trans. on Power Electronics*, vol. 14, no. 1, pp. 49-61, January 1999.

[15] H. Van Der Broeck, H. Skudelny and G. Stanke, "Analysis and Realization of a Pulse Width Modulator Based on Voltage Space Vectors," *IEEE Trans. on Industry Applications*, Vol. IA-24, No. 1, pp. 142-150, January 1988.

[16] R. Erickson and D. Maksimovic, "Fundamentals of Power Electronics", Kluwer Academic Publishers, Norwell, USA, 2001.