Techniques for SPWM using LUTs on low-cost microcontrollers

Article · March 2009

DOI: 10.1109/CSPA.2009.5069244

CITATION
1

READS
456

4 authors:

Muhammad Irfan

23 PUBLICATIONS **154** CITATIONS

SEE PROFILE

Naveed Ahmed

Firat University

9 PUBLICATIONS **23** CITATIONS

SEE PROFILE

M.T. Qadri

Sir Syed University of Engineering and Technology

45 PUBLICATIONS **325** CITATIONS

SEE PROFILE

Mubeen Sirhindi

UNSW Sydney

1 PUBLICATION **1** CITATION

SEE PROFILE

Techniques for SPWM Using LUTs on Low-Cost Microcontrollers

M. Irfan Anis, Naveed Ahmed, Muhammad Tahir Qadri and Mubeen Sirhindi
Department of Electronics Engineering, Sir Syed University of Engineering & Technology,
Karachi, Pakistan.
Email: mirfananis@yahoo.com, naveedahmed001@hotmail.com, mtahirq@hotmail.com,sirhindi@gmail.com

***Abstract*- This paper examines the techniques which can be used to generate Sinusoidal Pulse Width Modulation (SPWM) utilizing minimal resources of a low-end microcontroller. Different techniques are evaluated, bit-banging look-up table (LUT) routines are advocated for certain purposes, programming algorithms are considered for efficiency, and methods of harmonic suppression are reflected upon. Finally, feedback methods are considered and a practical application is presented. This paper is targeted towards inverter and motor drives operating in H-bridge configuration. Microcontroller based voltage source inverters.**

Keywords – Sinusoidal Pulse Width Modulation (SPWM), Inverters, Motor Drive, Microcontroller, Harmonic Measurement & Suppression.

I. INTRODUCTION

There has been a significant proliferation of small, low-cost microcontrollers in recent years, which has lead to their application in even the cheapest of consumer electronics. While a wide array of functionality can be derived from these, their applications often demand some manner of analog interfacing. Manufacturers have heeded this demand and have adapted by fabricating ADCs into microcontrollers. However, analog outputs are another matter. This is an especially interesting aspect since low-cost general purpose microcontrollers are encroaching on areas previously reserved for low-end DSPs.

Most microcontroller manufacturers suggest, rather off-hand, the use of D/A ladders and PWM modules, but these have high opportunity costs in terms of pin-count and code space. An alternative can be found in bit-banging a look-up-table (LUT). Although bit-banging is considered passé, perhaps a ritual of passage in the learning the use of such devices, it can be of great purpose, considering the limited resources available, in terms of processing time, RAM and code space, in a low-cost controller.

Sinusoidal pulse width modulation (SPWM) is the most commonly used control scheme for the voltage source inverter [1]. In sinusoidal PWM, instead of maintaining the width of all the the pulses uniform as in case of multiple PWM, the width of each pulse is varied in proportion to the amplitude of the sine wave evaluated at centre of the same pulse [2].

II. DISCUSSION

Three generally accepted approaches need to be considered for the generation of analog waveforms. First, is the use of DACs. Whether serial or parallel, DACs have the disadvantage of cost, which can be a significant factor knowing the applications under consideration. Moreover, parallel DACs, of any significant resolution, have a trade-off in terms of the pin-count, which is rarely affordable. The disadvantage of serial DACs lies in their diminished speed and high interfacing overhead. Second, is the use of D/A ladders or resistance ladders. Besides the disadvantage of a pin-count that they share with parallel DACs, D/A ladders almost always need to be buffered, introducing another component into the circuit, which in most cases is unacceptable. PWM modules, that are now so often built into microcontrollers, are another matter altogether. A lot of devices feature 8-bit, 10-bit or even 12-bit PWM modules. Operating at a substantial frequency, they are able to replicate analog signals rather well. The downside to PWM modules is often the LUT that they require. An 8-bit PWM module, which might require 200-256 reloads, will necessitate a LUT just as long i.e. 200-256 bytes, and thus may be undesirable. A similar application, using a bit-bang routine, might utilize one-tenth as much space.

It would also be pertinent to note that the microcontrollers under discussion, i.e. at the low-end of the spectrum, would probably not feature PWM modules, in which case the discussion is redundant.

III. SYSTEM MODEL

Firstly, due to the quadrant symmetry of a sine wave, the LUT needs only to incorporate one-fourth of the total wave i.e. the first quadrant will suffice.

Secondly, the rate at which the bits will be dumped is rather important. This can be anywhere from 32 to 256 bits per quadrant. The bit rate of the system, in relation to frequency of the reproduced sine wave and the bit rate per quadrant, is given by:

$$b_r = 4 f_{sine} b_Q \quad (1)$$

Considering the generation of a 50Hz sine wave, at 160 bits per quadrant, results in a bit rate of 32k bits/sec. Thus the minimum time period of a pulse, given by:

$$T_{p(\min)} = T_s = \frac{1}{b_r} \quad (2)$$

will be about 31.25us. This is also the sampling time. A simple algorithm can be used to generate a sinusoidal pulse width modulation (SPWM) LUT using any math CAD software. A generalized algorithm can be stated as below.

for $t_0 = T_s$ to $\frac{T_{sine}}{4}$ step T_s
  if $\left(\int_0^{t_0-T_s} a \sin 2\pi f_{sine} t \, dt > \int_0^{t_0-T_s} p(t) \, dt\right)$
    $p(t_0) = 1$
  end if
end for

As is immediately obvious, the algorithm tries to make the integral of the SPWM wave follow the integral of the sine wave. When the integral of the SPWM wave starts to fall behind the integral of the sine wave, a high pulse of time period T_s should cause the integral of the SPWM wave to rise above the integral of the sine wave. If not, the subsequent pulse is also high. If it rises substantially higher, the subsequent pulse will be withheld until the sine integral catches up and exceeds the integral of the SPWM wave.

However, it is found that this scheme results in an over-modulation of the SPWM wave. This can be improved somewhat with a slight modification:

for $t_0 = T_s$ to $\frac{T_{sine}}{4}$ step T_s
  if $\left(\sqrt{\frac{\int_0^{t_0-T_s}(a \sin 2\pi f_{sine} t)^2 dt}{t_0-T_s}} > \sqrt{\frac{\int_0^{t_0-T_s} p(t)^2 dt}{t_0-T_s}}\right)$
    $p(t_0) = 1$
  end if
end for

Now the characteristic under consideration is the RMS value of the sine wave. The algorithm tries to make the RMS of the SPWM wave follow the RMS of the sine wave. When the integral of the SPWM wave starts to fall behind the integral of the sine wave, a high pulse of time period T_s should cause the RMS of the SPWM wave to rise above the RMS of the sine wave. If not, the subsequent pulse is also high. If it rises substantially higher, the subsequent pulse will be withheld until the sine RMS catches up and exceeds the RMS of the SPWM wave.

The difference between the two modulation schemes is shown in figure 1 (a) and (b). While both replicate the characteristics of the sine wave, the second avoids the pitfalls of over-modulation by closely following the RMS value, thus maintaining the equivalent effective energy transfer. This is seen to been a consistent difference, and subsequent improvement, of about 12-14%.

It is not necessary, indeed unadvisable, to use the bit addressable memory of a microcontroller to store the look-up table so generated. It is a precious resource and usually little is available. Rather, a significant improvement would be its storage in a compressed form in the code memory. Two look-up tables, one of amplitude 0.5 and the other of amplitude 1.0, are reproduced here.

```
SPWM[2][8] ={{0x02, 0x22, 0xA5, 0xAA},
             {0x42, 0x6A, 0xF7, 0xFF}};
```

The least significant bit of the lowest byte is the first sample in the table and the most significant bit in the highest byte is the last sample in the table. The advantage of this scheme is immediately obvious. 2 look-up tables have been stored in only 8 bytes, each of four bytes or 32 bits long (i.e. $b_Q = 32$ samples). Considering f_{sine} to be 50Hz, in this small space, a sampling capability of 6.4kSps has been achieved. In this manner, even a small microcontroller can be made capable of a reproducing a large number of pre-stored waveforms of many amplitudes and frequencies, perhaps a very crude form of direct digital synthesis (DDS). This technique is gaining popularity as a method of generating sinusoidal signals and modulated signals in digital systems [3].

The table, of course following a uni-polar scheme, needs to be reproduced on two separate outputs, once forwards and once backwards. Using a small interrupt-driven counter, the required bit can be easily found, e.g.:

```
SPWM[i>>3]&(1<<(i&0x07))
```

The calculation shown above might be disarmingly simple. However, complications have only begun if the table exceeds 256 bits per quadrant. Generally speaking, for the more common applications of inverters and motors drives at 50Hz, this translates into a bit rate of 51.2kSps, which is much more than enough. Performance considerations are of course paramount. The feasibility of this system is entirely dependent upon the kind of load that it places on the microcontroller. The processor utilization, is considered in terms of the number of instructions required to reproduce a sample, the overall sampling rate of the system and the processor speed (in terms of instructions per second), and is given below.

$$PU = \frac{N_i b_r}{I_r} \times 100\% \quad (3)$$

For example, if 15 instructions (inclusive of interrupt service routine entry and exit) are required to sample a sine wave at a rate of 6.4kSps, on a 1MIPS microcontroller, the utilization of the microcontroller will be 9.6%. By this, not only can the processing load be judged, but the trade-off between resolution or the sampling rate, and the processor utilization becomes immediately apparent. For all the pros that have been discussed, a mention of a con of this approach is only fair. While PWM modules have been excluded from our consideration, nonetheless had one been available it would have allowed for a real-time rescaling of the amplitude of the output waveform simply by performing a floating-point multiplication. The alternate to this scheme, which is presented here, as perhaps the astute would have noted, is the inclusion of several LUTs, each of a discrete amplitude, thus

limiting the choices for and flexibility in the amplitude of the sinusoidal waveform being generated.

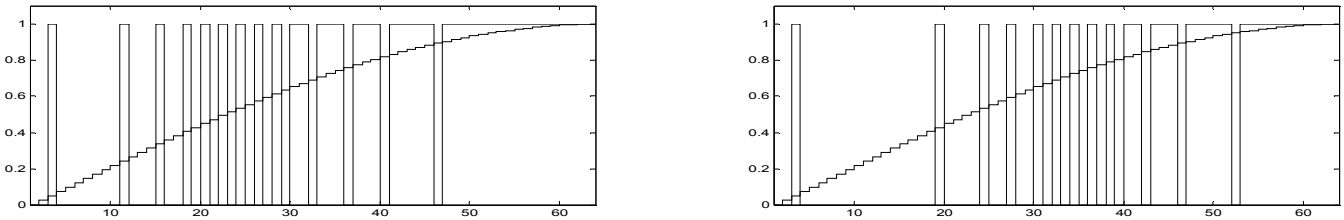

Fig. 1. 64 bits per quadrant SPWM LUTs (a) Following the Integral (b) Following the RMS

IV. Analysis

Although the LUT is uni-polar, its intended application is in a bi-polar scheme through an H-bridge or a similar arrangement resulting in an output across a resistive load as shown in figure 2.

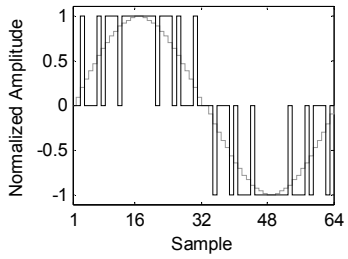

Fig. 2. A 16 bits per quadrant LUT shown over whole waveform

A. Harmonic Measurement

Ideally, only the fundamental harmonic should exist. Unfortunately, this is never quite the case. Here, the waveform will comprise of the fundamental harmonic and the odd multiples thereof. This is quite easily confirmed by performing a quick FFT of a large enough sample of the waveform, as shown in figure 3.

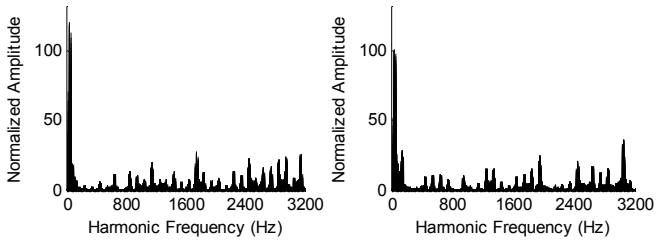

Fig. 3. Harmonic content (f_{sine}=50, b_Q=32) of (a) "Integral Following" and (b) "RMS Following" waveforms

It is seen that the harmonic distribution is more or less even across the spectrum and there is no specific pattern to it. Moreover, even a precursory comparison between the FFTs of the "integral following" and the "RMS following" waveforms reveal that the latter has a greater harmonic content. This also confirmed by later calculations. The fundamental harmonic is also seen to be stronger in the "integral following" waveform.

The total harmonic distortion (THD) of a waveform, which is given by [4],

$$THD = \frac{1}{V_{01}}\sqrt{\sum_{n=2,3,\ldots}^{\infty}(V_{0n})^2} \quad (4)$$

is not an adequate measurement of the harmonic content in this case because higher order harmonics can be easily filtered. Indeed, the inductive and capacitive loads that we are targeting can themselves filter most of the higher order harmonics. Instead, THD can be modified slightly by considering the order of the filter attenuation to get the distortion factor (DF), given by [5],

$$DF = \frac{1}{V_{01}}\sqrt{\sum_{n=2,3,\ldots}^{\infty}\left(\frac{V_{0n}}{n^{order}}\right)^2} \quad (5)$$

and by measuring the lowest order harmonic (LOH), which is the lowest harmonic with an amplitude of or greater than 3% of the fundamental harmonic. Two general points to consider when measuring harmonics include the distribution of harmonics and the relation of the fundamental harmonic to the sampling frequency. When harmonics are inevitable, higher-order harmonics are preferred to lower-order harmonics due to the ease with which they can be filtered by discrete components. Hence, if the choice exists, the distribution of harmonics should be pressed towards the higher side as much as possible. This will make little difference in the voltage THD, but a significant difference in the effective current THD. It might also be tempting to push the number of bits per quadrant (i.e. the sampling rate) as high as possible. However, one should be aware that this leads to a weakening of the fundamental harmonic, therefore a balance needs to be struck.

B. Harmonic Suppression

Harmonics are not only unwanted but also harmful causing harmonic torque, heating, interferences and oscillations. One of two approaches to harmonic suppression can be adopted. Firstly, a hardware filter across the load can be considered i.e. a C, LC, pi-CLC or some other such filter arrangement [6]. In CLC, a good approximation of the sinusoidal voltage waveform can be produced by a set of triangular carrier signals and a sinusoidal modulating signal using a SPWM [7]. This is quite effective for filtering higher order harmonics and can be specifically tuned to target certain harmonics. A consideration to keep in mind here is that the filter itself will be an additional load, thus reducing efficiency. Simple

implementations of the capacitive and inductive reactance are quite enough for this method. The second approach would be to modify the LUT to specifically eliminate certain harmonics. A method, presented in [8], is to use notches calculated by the following formula:

$$B_n = \frac{4V_d}{n\pi}\left[1 + \sum_{k=1}^{m}(1-k)^k \cos(n\alpha_k)\right] \quad (6)$$

where the n^{th} harmonic is suppressed with a notch starting at angle α. For notches to suppress multiple harmonics, the actual mathematics of this task can be often daunting. An iterative/heuristic incremental improvement algorithm to solve the problem can be a great solution. Although pre-calculated notches in the LUT have the disadvantage being process intensive, this is a one-time expense on a platform that can afford it. Usually a combination of the two approaches, a hardware filter for higher order harmonics and a LUT modification for lower order harmonics, yields the best results i.e. strategically placed notches in the LUT for the more prominent lower-order harmonics and a carefully tuned hardware filter for the more prominent higher-order harmonics.

V. Proposed Model

In all fairness, open-loop SPWM techniques are often a poor substitute for an actual sine. Since the original LUT calculations do not, and in many cases cannot, take into account the load being driven, especially if it is greatly variable in its magnitude and nature. This is especially true of inverters. However, several modifications to an implementation can greatly improve the quality of the output. Two aspects of the proposed model are considered: closed loop control and real time harmonic suppression. The proposed model, while greatly enhancing the capabilities, does pose several drawbacks in terms of component count. This is because several variable aspects of the system are left to be controlled by an analog hardware circuit, rather than a software algorithm.

A. Closed Loop Control

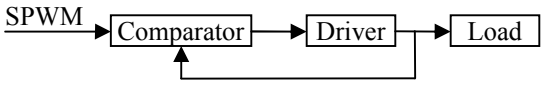

Fig. 4. Closed Loop Control

As shown in figure 4, an SPWM output is given to a comparator, the output of which is used to control a driver circuit translating into the correct modulation for the load. However, since the nature and the magnitude of the load can distort the voltage across or current through the load, a feedback loop is constructed by measuring the voltage or the current. In this arrangement, if the output falls below the expected level determined by the SPWM wave, the comparator compensates by extending the pulse or inserting another pulse until the output catches up. Similarly, if the output jumps above the expected level determined by the SPWM wave, the comparator compensates by inserting notches allowing the output falls in line.

B. Real Time Harmonic Suppression

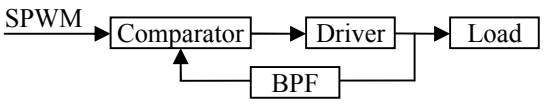

Fig. 5 Real Time Harmonic Suppression

As shown in figure 5, the closed loop circuit shown in figure 4 can be modified to suppress specific harmonics in real time. This can be achieved by placing a band-pass filter in the feedback path, which can isolate the specific harmonic. The level of this harmonic, beyond a set point, can be used to trigger a notch. This works along pretty much the same principle as the earlier circuit. Real time harmonic suppression will tend to distort the harmonic distribution of the waveform. This is method is more suitable for lower order harmonics since while it will suppress the lower order harmonic, it will generally induce more higher-order harmonics.

C. A Basic Practical Drive

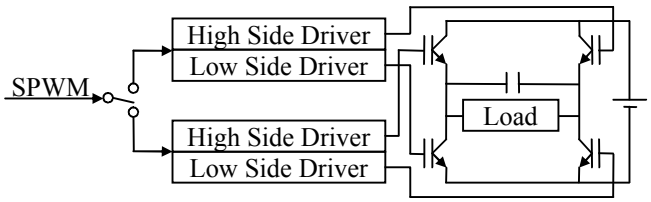

Fig. 6 A Basic Practical Drive

A basic practical drive should incorporate the suggested improvements of a C filter as well as harmonic suppression within the LUT. The SPWM should be switched between either sides of the H bridge arrangement, wherein high and low side drivers manage the electrical aspects of switching the devices on and off in line with the LUT.

VI. Conclusion

The paper has examined how low-cost microcontrollers can be used to great effect for generating sinusoidal power transfers through properly modulated PWMs. In a departure from previous thinking, a newer mode of LUT table generation is considered, with its advantages and drawbacks, it is analyzed for common harmonic problems along with appropriate solutions, and finally practical applications and circuits have been presented.

REFERENCES

[1] Yaow-Ming Chen, Yuan-Ming Cheng," Amplitude distortion compensation for voltage source dc/ac inverters", INT. J. Electronics, Vol. 91, No. 2, February 2004, pp. 83–96

[2] P M Menghal, A S Sindekar," Harmonic Analysis for Some Prominent PWM Strategies in Inverters", IE(I) Journal-EL, Vol 84, September 2003, pp. 79-85

[3] Xinjun Zhang, Xin Jiang, Wentao Song, Hanwen Luo," A Novel Direct Waveform Synthesis Technique With Carrier Frequency Programmable", Volume 1, 17-21 March 2002, pp. 150 – 154

[4] Stephen J. Chapman, *Electric Machinery Fundamentals*, Fourth Edition, McGraw Hill.

[5] P. C. Sen, *Principles of Electric Machines and Power Electronics*, Second Edition, John Wiley & Sons.

[6] B. L. Theraja, A. K. Theraja, *A Textbook of Electrical Technology*, 2004 Edition, S. Chand & Company Ltd.

[7] S. M. Bashi," Simulation and Development of Low Harmonics High Voltage Chain Power Inverter", European Journal of Scientific Research, Vol.25 No.1, 2009, pp. 111-117

[8] Muhammad H. Rashid, *Power Electronics: Circuits, Devices, and Applications*, Third Edition, Prentice Hall Publications