FPGA based Digital Pulse Width Modulator with Time Resolution under 2 ns

R. Murugesan,
Assistant Professor, Department of EEE,Ranipettai Engineering college ,
vellore,Tamilnadu,India.
R. Madhusudhanan,
Research scholar,Sathyabama university,
madhujeppiaar@india.com

Abstract

This work proposes a new DPWM architecture that takes advantage of FPGA's advanced characteristics, especially the DLLs (Delay-Locked Loop) present in almost every FPGA. The proposed DPWM combines a synchronous (counter-based) block with an asynchronous block for increased resolution without unnecessarily increasing the clock frequency. The experimental results show an implementation in a low cost FPGA (Xilinx Spartan-3) that uses an external 32 MHz clock for a final time resolution under 2 ns.

1. Introduction

An Digital control of switching mode power supplies (SMPS) has obtained great research attention due to their now well known advantages [1-3], such as programmability, advanced control algorithms, reduced component count, low sensitivity to external factors or aging, ease of design and prototyping, etc. However, their disadvantages are also well known, being the two more important the processing/sampling delay and the limited resolution [4]. Regarding the second factor, resolution is limited mainly by the ADC and the PWM. However, the ADC resolution is becoming a less important problem thanks to the windowed ADC technique [3] and because the PWM resolution needs to be higher than ADC resolution for avoiding limit cycling [4-5].Traditional Digital PWMs (DPWMs) are based on counters (see section II.B). The advantage of these DPWMs is that they are very simple and obtain high linearity. Given the interest of obtaining high resolution DPWMs, a lot of different architectures have been proposed in the last years [6-7], and even a classification of them has appeared . Although all these architectures differ from each other, almost all of them make use of delay-lines. The main advantages of delay-line DPWMs are high resolution and low power consumption. However, they have lower linearity and even non-monotonic behavior in some cases. Trying to obtain a trade-off between the advantages of counter-based and delay-line DPWMs, hybrid architectures have also been proposed. This paper proposes a new DPWM hybrid architecture.

The synchronous block is counter-based. However, the asynchronous block does not implement a delay-line, but it uses FPGA internal resources. Taking advantage of these resources, it obtains an excellent trade-off between linearity and time resolution. The proposed DPWM is, in principle, only intended for FPGA implementation, but it is very simple to design and can be implemented even in the lowest cost FPGAs, as shown in the experimental results. The paper is organized as follows: next section describes the proposed DPWM architecture, while section III shows the experimental results. Finally, section IV gives the conclusions.

2. Proposed dwpm architecture

A.DLL block

The key of the proposed DPWM architecture is that it takes advantage of the advanced DLL features that are available in almost every FPGA nowadays. Digital devices like FPGAs have specific blocks that can manage clock signals: DLL (Delay-Locked Loop) or PLL (Phase-Locked Loop). Using these DLLs or PLLs, it is possible to multiply or divide the clock frequency. Many of these DLLs can also generate four phase-shifted clocks (shifted 0º, 90º, 180º and 270º) directly (Spartan and Virtex families of Xilinx) or allow to generate phase-shifted versions of the clock (Cyclone and Stratix families of Altera, ProAsic3, Fusion or Axcelerator families of Actel).The first feature of these DLLs that is used in the proposed DPWM is multiplying the clock frequency. The advantage of doing so is that a high clock frequency can be internally used in the DPWM while an external lower frequency is generated and also used in the rest of the digital controller.

B. Synchronous block

The synchronous block is a counter-based DPWM that uses the most significant bits (MSBs) of the duty cycle, d[n-1,2], being n the total number of bits.the synchronous block is based on a counter and comparison structure. The functionality of this block is the following: if the duty cycle command is over the counter value, the output is in the on-state, and when the counter reaches duty cycle the output is turned off. This is a simple block and, therefore, it can work at high clock frequencies.

In the synchronous block, resolution is given by both the clock and the switching frequencies. This resolution can be obtained as:

$$\text{Resolution} = f_{clk} / f_{sw} \qquad (1)$$

where f_{clk} is the clock frequency and f_{sw} is the switching frequency.

C. Asynchronous block

The two least significant bits (LSBs), d[1,0], are used in the asynchronous block. These two bits are used to select between the four phase-shifted clocks generated by the FPGA's DLL. In fact, these signals are combined using and gates (see Fig. 1) in order to obtain other four phase-shifted signals that are high only a quarter of a cycle instead of half a cycle (see Fig. 4).The basic idea of using these four phase-shifted signals is obtaining four possible switching instants during each clock cycle. Therefore, resolution is multiplied by four. In general, using m asynchronous bits, the total resolution is calculated as:

$$\text{Resolution} = 2m \bullet fclk / fsw \qquad (2)$$

where m is the number of asynchronous bits, fclk is the clock frequency and fsw is the switching frequency.

D. Functionality of the DPWM architecture

The proposed DPWM, as shown in Fig. 1, is composed of two blocks: a synchronous block and an asynchronous block, as explained in the last sections. The functionality of the proposed hybrid DPWM is described as follows (see Fig. 1). The output of the synchronous block (counter-based block using the external frequency multiplied by 4) sets the output of the DPWM depending on the MSBs of d (duty cycle command). The proposed DPWM architecture is intended for a synchronous multiphase buck converter, so it creates driving signals for both the high side MOSFETs (HSM) and low side MOSFETs (LSM).Of course, it can be easily adapted to any other SMPS topology. The idea is that the turn-on instant of the HSM is always coincident with a 0º clock edge, while the turn-off instant can be at any of the four clocks edges, depending on the LSBs. The opposite is done for the LSM: turned-on with any of the four clock edges while turned-off with the 0º shifted clock.

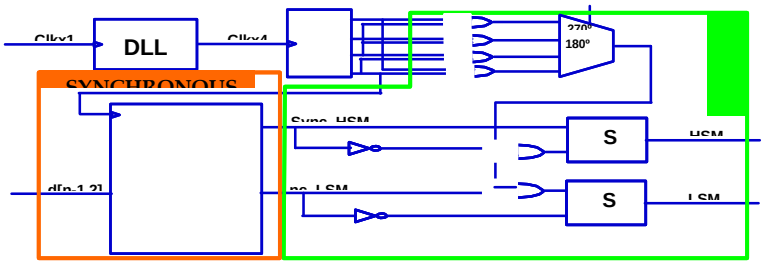

Figure 1. Proposed DPWM architecture (basic outline).

As it can be seen in Fig. 1, the asynchronous block generates the signal named QuarterCycle, which corresponds to a quarter of the clock cycle, starting in the rising edge of one of the four clocks, depending on the value of the 2 LSBs of the duty cycle . For the HSM, the output is reset when the synchronous block is already off and QuarterCycle arrives. Therefore, the output is active and integer number of clock cycles (as generated by the synchronous block using the n-2

MSBs) plus 0 to 3 quarters of a cycle, depending on the 2 LSBs. However, using asynchronous techniques involves delay problems. In this case, signal QuarterCycle suffers from these problems. Its value corresponding to the 270º clock also has a short on-time at the beginning of the 0º clock . This problem can produce a non-monotonic behavior (duty cycle commands ending in "11" would be similar to "00" commands). In order to avoid the non-monotonic behavior some modifications in the DPWM architecture are proposed in the figure 2.

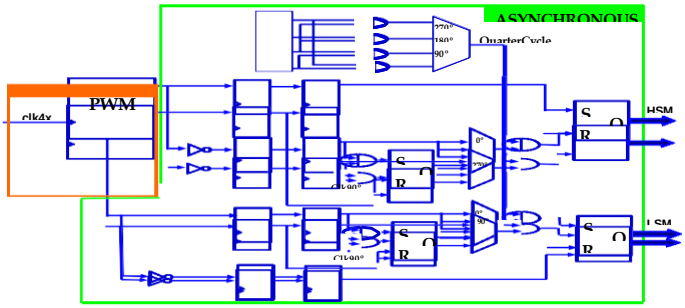

Figure 2. Proposed DPWM architecture avoiding non-monotonic behavior

E. Modifications for avoiding non-monotonic behavior

As explained before, the proposed basic DPWM shown in Fig. 1 can have problems of non-monotonic behavior. In order to guarantee monotonicity, it needs some additional resources used for avoiding the problems caused by asynchronous delays. The proposed block diagram is shown in Fig.5. In order to avoid the non-monotonic behavior, the reset of the output when the 2 LSBs of the d are "11" is only allowed after the arrival of the 90º clock. For that reason, additional RS registers and multiplexers are added . As a conclusion, the proposed DPWM achieves high resolution thanks to the use of 4 phase-shifted clocks while maintaining high linearity and monotonic behavior avoiding delay-lines. Resolution is increased 4 times compared to only counter-based DPWMs.

3. Experimental results

The software used for simulation was ModelSim. In order to assure a correct VHDL description of the proposed design, behavioral simulations (these simulations do not include delays) were done. However, in order to check linearity and monotonic behavior, post-place and route simulations were carried out. These simulations include the delays of each signal. Using these simulations, non-monotonic behavior was identified in the design. The proposed DPWM has been implemented in a Spartan-3 FPGA (Xilinx), occupying only 22410 equivalent gates for a 4-phases buck converter (including the DLLs). The Spartan families are the low-cost low-speed FPGAs of Xilinx, so even better time resolution results can be obtained using Virtex FPGAs.

These DCMs also generate the four phase- shifted clocks. Hence the internal clock frequency of the DPWM is 128 MHz. The final resolution of this DPWM can be calculated as a time step using:

$$\Delta T = Tclk4x / 2m \qquad (3)$$

where Tclk4x is the period of the 128 MHz clock frequency (7.81 ns), and m is the number of asynchronous bits (2 in this case). Therefore, the final resolution is a quarter of the 128 MHz clock period, that is, 1.95 ns.As the switching frequency was 250 kHz for these experimental results, the total resolution can be calculated using (2) for a result of 2048 different duty cycle solutions. This is equivalent to 11 bits of resolution, 9 for the synchronous block and 2 for the asynchronous block. However, the number of bits of resolution changes from application to application depending on the switching frequency. Higher resolutions can be achieved using faster devices. Synthesis results for other FPGAs are shown in Table I. All these results were obtained using XST (Xilinx Synthesis Tool) and Xilinx ISE v8.1.

TABLE I

RESOLUTION OF THE PROPOSED DPWM FOR DIFFERENT FPGAS

Device	Max. freq. (MHz)	Resolution (ns)
Spartan 3 (slow) XC3S200-4FT256	171.2	1.460
Spartan 3 (fast) XC3S200-5FT256	196.3	1.274
Virtex 2 (slow) XC2V40-4CS144	147.2	1.698
Virtex 2 (fast) XC2V40-6CS144	206.7	1.209
Virtex 4 LX (slow) XC4VLX15-10SF363	253.5	0.986
Virtex 4 LX (fast) XC4VLX15-12SF363	331.3	0.754

TABLE II

EXPERIMENTAL ON-TIME FOR DIFFERENTDUTY CYCLES

Duty cycle	*On-time*
001 0000 0000	502.2 ns
001 0000 0001	504.2 ns
001 0000 0010	506.4 ns
001 0000 0011	508.3 ns
001 0000 0100	510.0 ns
001 0000 0101	512.4 ns
001 0000 0110	514.2 ns
001 0000 0111	516.1 ns

Linearity and monotonic behavior have also been tested. Table II shows the measured on-time for different duty cycles. These results are close to theoretical ones (1.95 ns steps), showing the feasibility of the proposed DPWM architecture. Furthermore, the accumulated step every four duty cycle solutions (discarding the 2 LSBs) shows an even higher. linearity, as it is caused by the synchronous block which is highly linear.

4. CONCLUSIONS

A new hybrid counter-asynchronous DPWM architecture has been proposed. This DPWM, which is easy to design, is intended for FPGA implementation, as it takes advantage of the internal DLL available in almost every FPGA nowadays. The DLL raises the resolution of the DPWM in two ways. The external clock frequency is internally multiplied for a higher resolution of the counter-based block of the DPWM. Once the maximum possible resolution is achieved in the synchronous block, it is multiplied by four using four phase-shifted clock outputs of the DLL. The proposed DPWM has been verified through experimental results using a low-cost FPGA implementation (Spartan-3), which shows the feasibility of the method not only for prototyping purposes but also for final products. A time resolution under 2 ns has been obtained while keeping high linearity and monotonic behavior.

References

1. IEEE Trans. Power Electron. (Special Issue on Digital Control in Power Electronics), vol. 18, n. 1, part II, Jan. 2003.

2. B. J. Patella, A. Prodic, A. Zirger, D. Maksimovic, “High-Frequency Digital Controller IC for DC-DC Converters”, IEEE Trans. Power Electron., vol. 18, n. 1, pp. 438-446, Jan. 2003

3.A. V. Peterchev, J. Xiao, S. R. Sanders, “Architecture and IC Implementation of a Digital VRM Controller”, IEEE Trans. Power Electron., vol. 18, n. 1, pp. 356-364, Jan. 2003.

4. A. V. Peterchev, S. R. Sanders, “Quantization Resolution and Limit Cycling in Digitally Controlled PWM Converters”, IEEE Trans. Power Electron., vol. 18, n. 1, pp. 301-308, Jan. 2003.

5.H. Peng, A. Prodic, E. Alarcon, D. Maksimovic, "Modeling of Quantization Effects in Digitally Controlled DC-DC Converters", Proc. IEEE Power Electronics Specialists Conf. (PESC), Jun. 2004, Aachen, Germany, pp. 4312-4318

6. A.P. Dancy, A.P. Chandrakasan, “Ultra low power control circuits for PWM converters”, Proc. IEEE Power Electronics Specialists Conf. (PESC), June 1997, St. Louis, Missouri, USA, vol. 1, pp. 21-27.

7. E. O'Malley, K. Rinne, "A Programmable Digital Pulse Width Modulator Providing Versatile Pulse Patterns and Supporting Switching Frequencies Beyond 15 MHz", Proc. IEEE Applied Power Electronics Conf. Expo (APEC), vol. 1, Feb. 2004, Anaheim, CA, pp. 53-59.