FPGA based Design and Implementation of Different Approaches for High Resolution Synchronous DPWM

Yashoda M S
M.Tech, Dept. of Electronics and Communication
Sir.M Visvesvaraya Institute of Technology ,
INDIA

Shalini p
Asso.Prof. Dept. of Telecommunication
Sir.M Visvesvaraya Institute of Technology,
INDIA

***Abstract*—Advantages of digital control in power electronics have led to an increasing use of digital pulse-width modulators (DPWM). While using classical DPWM architectures, the clock frequency requirements may exceed the operational limits when the power converter switching frequency is increased. This paper is about designing and implementing the four synchronous digital pulse-width modulator (DPWM) to increase the resolution of the designs implemented on field-programmable gate arrays (FPGAs). The first two architectures are based on the on-chip digital clock manager present in the low-cost Spartan-3 FPGA series and the I/O delay element respectively. The third architecture uses the on chip PLL blocks to generate fixed delays and selector to choose the one corresponding with the desired duty-cycle. In the fourth architecture, a serializer-deserializer (SERDES) module is used to serialize a thermometer-coded representation of the LSB portion of the input. The serialization technique is commonly used for high speed data transmission like LVDS and is extensively supported by FPGA providers. All four architectures have been analyzed, implemented and compared to verify the performance of these architectures.**

***Keywords*—FPGA, DPWM, LVDS, PLL, SERDES**

I. INTRODUCTION

The development towards digital control in high-frequency dc-dc converters has created a need for a new class of modulator: the digital pulse-width modulator (DPWM). The input to this basic building block is a digital word and outputs a square wave of fixed period with the a duty-cycle proportional to the input value. Digital pulse width modulators (DPWMs) have become an integral part of almost all embedded systems. It has been widely accepted as control technique in electronic appliances. One of the applications of DPWM lies in power electronics for controlling power converters (DC/DC, DC/AC, etc.). PWM operations are extensively used in the power converters. It outputs a square waveform with a varying on to off ratio. This ratio is called duty cycle, and the average duty cycle can vary from 0 to 100. There are basically two PWM techniques, Analog PWM generation technique and Digital PWM generation technique. When compared to analog controls, digital controls are increasingly used in power converters because of their advantage. The main advantages of using digital controls over analog are the ability to perform more advanced and sophisticated functions that potentially result in improving power conversion efficiency and/or dynamic performance of the power converter, reduced sensitivity to component variations, the ease of digital control function and loop upgradeability compared to analog controllers. Digital pulse width modulators mainly work as basic building block in digital control architectures of any power converters.

The DPWM resolution determines the accuracy in the output voltage/current control where as DPWM frequency is mainly determined by the power converter operating conditions. As a consequence, the DPWM resolution has a direct impact in the power converter performance. Traditional DPWM implementations generate the power converter gating signals according to several predefined thresholds and are based on counters and comparators. In the PWM design, the synchronization and resolution determines the accuracy in the output voltage/current control of any power converters.

The four architectures explained in this project based on fully synchronous designs. The first proposed architecture, is a generalization of the DCM based circuit, and it allows operating the circuit at higher clock frequencies. The second proposal is based on the I/O delay element (IODELAYE1) and it provides higher resolution with a straightforward implementation. The third architecture is PLL based, as expected, post-fitting adjustment to PLL delays was needed to obtain a linear response. The forth implementation is based on a serializer block driving an LVDS high-speed serial I/O available in most of modern FPGA families and does not need manual routing nor post-fitting calibration. Besides, these are glitch free designs, which improve the circuit reliability and the number of paths to equilibrate in order to achieve a monotonic behavior is minimized.

The main idea of this paper is to identify and investigate best architecture for DPWM that give better resolution for power converters and objective of these paper is to propose four fully synchronous high-resolution DPWM architectures in order to avoid the need of using unfeasible high clock frequencies, providing a more convenient final implementation. All four are based on the resources available in modern FPGAs and these architectures mainly used to achieve higher resolution PWMs.

II. RELATED WORK

Before surveying the published literature, there are numerous other papers are associated with this paper. To make things easier, this section will focus on few papers that are related with this literature.

Some FPGA-based DPWM implementations were reported in [1]-[5]. The implementations presented in [1],[2] use delay lines in combination with a counter-comparator architecture, emulating in some sense many of the ASIC implementations found in the literature. These implementations obtain excellent resolution results, but they require manual routing or postfitting adjustments, and they lack of flexibility as the obtained resolution must be a multiple of technology-dependent delays. A recent work [7] uses Altera's synthesis tool features to avoid the need for manual routing but still suffer from the second drawback.

In [3]-[6] specific resources encountered in modern FPGAs are used, namely phase-locked loops (PLL), also complemented by a counter-comparator architecture. One approach [3],[5] uses on-chip PLL blocks to generate fixed delays and a selector to choose the appropriate delay. Manual routing or post-fitting delay adjustments are needed to compensate delay differences between the different propagation paths. Other works [4],[6] use the fine phase shifting of the clock available in the on-chip PLL of most FPGA families, obtaining resolution values similar to the delay lines solutions. However, the procedures to modify the phase on PLL outputs is usually slow, and this is undesirable when a fast response is needed.

Moreover, recent developments in semiconductor technology enable the use of higher switching frequencies through SiC [19] and GaN [20] power devices. This allows the design of power converters with reduced size and cost, and improved dynamic behavior and power density, as shown in [21] and [22]. However, these designs require high-frequency high-resolution PWMs (HRPWMs) in order to take the most of the power converter.

Another field of application for HRPWMs is the dc–dc converters, where either the output voltage [voltage regulator modules (VRMs)], the duty cycle for output-power control [23], or the switching delay mismatch between power devices [24], [25] need to be accurately tuned. As a conclusion, the evolution of both power electronics and digital control techniques makes the development of higher resolution DPWMs [26] necessary. To overcome this problem, different solutions have been proposed depending on whether the digital controller is implemented on a digital signal processor (DSP), an application specific integrated circuit (ASIC), or a field programmable gate array (FPGA).

In the case of DSPs, some of them include HRPWM peripherals [27]. The HRPWM module extends the time resolution capabilities of the conventional PWM allowing a minimum time step that is a fraction of the system clock. Besides, several architectures have been proposed for IC implementation [28]–[30]. They are usually based on a tapped delay line in combination with a multiplexer [28] or a hybrid counter/delay line [30].

Several FPGA-based solutions have also been proposed in the literature [31]–[38]. One common solution is to use a coarse resolution counter-based stage plus one or several on-chip digital clock manager (DCM) blocks. The PWM signal is set at the beginning of the counter period, and it is reset after a given number of clock cycles plus a certain fraction of the clock period established by the DCM.

Apart from [35] and [38], the circuits previously published for delaying the reset signal are not fully synchronous. Asynchronous circuits make harder to perform static timing analysis and can result in glitching since controlling the logic and routing delays in an FPGA is more difficult than in ASIC implementations. A synchronous design, therefore, improves the reliability of the circuit and eases the design process. Besides, it makes the design more independent of the technology, easing the design portability.

III. HRDPWM ARCHITECTURE USING DCM BLOCKS

The DCM-based architecture uses the Digital Clock Manager(DCM) block present in the Spartan-3 FPGA. These architectures mainly used to achieve higher resolution PWMs. This is the glitch free design, which improve the circuit reliability and the number of paths to equilibrate in order to achieve a monotonic behavior is minimized.

A. Digital clock manager

DCM is the clock management block present in the Spartan-3 FPGA series. It provides advanced clocking capabilities to Spartan-3 FPGA applications. DCM solves the common clocking issues by

- Shifting the clock
- Condition the clock
- Multiplying the incoming clock signal
- Eliminating the clock skew

Simplified pin out description of DCM block is shown in figure 1

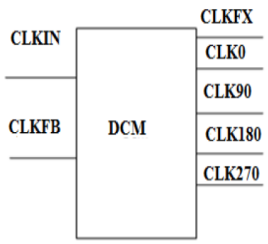


Fig. 1: Simplified Pin Out Of DCM.

DCM mainly consist of four functional blocks namely

- Phase Shifter
- Digital Frequency Synthesizer
- Delay Locked Loop
- Status logic

The key of this architecture is the on-chip DCM block provided in almost every state of the art FPGA. The DCM provides four quadrant phase-shifted clock signals derived from the source clock CLKIN. In addition to CLK0 for zero-phase alignment to the CLKIN signal, the DCM also provides the CLK90, CLK180 and CLK270 outputs for 90,180and 270 phase-shifted signals, respectively. Besides, all the outputs of the DCM can be phase shifted with finer resolution. It can

generate a wide range of output clock frequencies (CLKFX output port), performing clock frequency division and multiplication. Besides this phase shifting, the DCM is able to obtain clock outputs with 50% duty cycle. The clock feedback signal CLKFB is used to compare and lock the output signals with the input CLKIN signal. The fine phase shifting can be fixed or variable. It is set by means of the DCM attribute PHASE_SHIFT, an integer in the range [−255, +255]**.** Fixed fine phase shifting effect is shown in the figure 2.

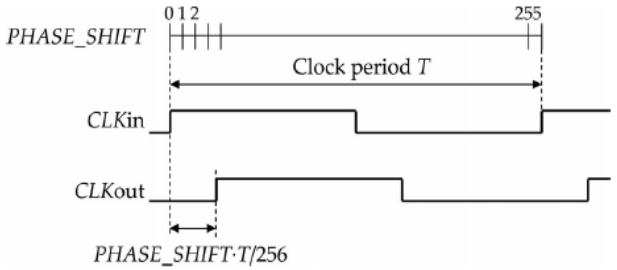


Fig. 2: Fixed fine phase shifting effect.

B. Architecture

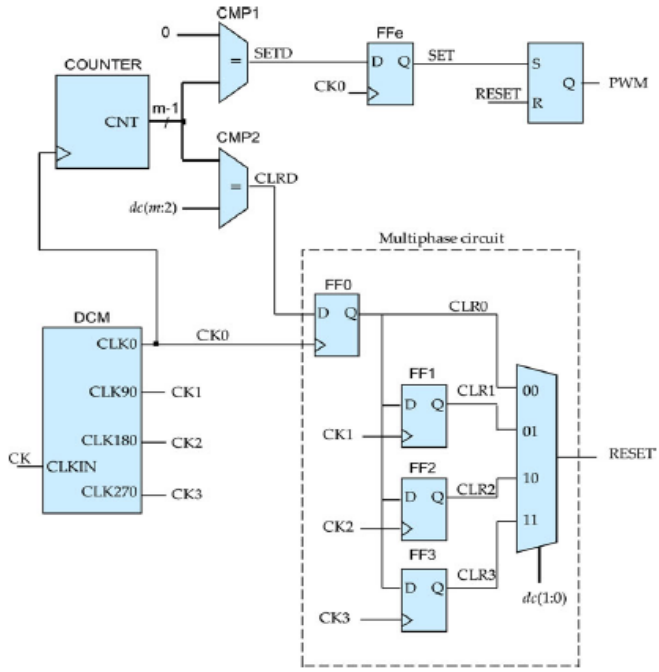


Fig. 3: HRDPWM Architecture using DCM block

The DCM-based high resolution DPWM(HRPWM) architecture is as shown in figure 3. In this first version, the quadrant phase-shifted outputs of a single DCM are used. The duty cycle command dc(*m*:0) has *m*+1 bits, ranging from *m* to 0, and the counter "CNT" has *m*−1 bits. "CLRD" signal is set when the *m*−1 most-significant bits (MSBs), dc(*m*:2), are equal to CNT; and SETD signal is set when CNT is equal to zero and dc(*m*:2) is different from zero.

Fig. 3 shows architecture of High resolution DPWM using DCM blocks. Basically, when the counter CNT is equal to the *m*−1 MSBs of the duty command dc, signal CLRD activates. The resulting pulse is captured in the next clock cycle by FF0, and phase shifted 90◦, 180◦, and 270◦ by flip-flops FF1, FF2, and FF3, respectively. These four FFs implement a multiphase synchronous circuit. The two least-significant bits (LSBs) of the duty command are used by the multiplexer to select the phase-shifted signal that clears the SR latch. The advantage of this architecture is that the digital circuit that generates the reset of the SR latch is synchronous. The use of asynchronous circuits to reset the latch makes harder to calculate timing using static timing analysis and can result in glitching since controlling the logic and routing delays in an FPGA is more difficult than in ASIC implementations.

IV. HRDPWM ARCHITECTURE USING IODELAYE1 BLOCK

In the second approach we make use of the I/O delay block (IODELAYE1) which is present in Virtex-6 series FPGAs or as IODELAY2 present in Virtex-7 FPGAs. The IODELAYE1 block provides a certain delay for the input signals according to the parameters and mode we use the block thus by allowing us to generate the signals which has been delayed by a certain number of tap delays with respect to the input. The factor which calculate the resolution is the tap resolution and for IODELAYE1 the tap resolution is given by, $t_{tap} = 1/(32 \cdot 2 \cdot f_{CK\_REF})$ by providing a fine delay-time t_d adjustment.

The range of the IODELAY1 Block is determined by the reference clock frequency which is the main attribute in this design. The range of clock frequency is from 200 ± 10. The IODELAY1 block can run on different modes depending on the input ie, the attributes which we set. In increment/decrement mode (CE), increment/decrement delay (INC), and reset (RST), are the attributes which determines the mode of operation of the IODELAY1 to control the desiring delay as needed.

One of the advantages of the block is that it uses a clock signal C which makes the design synchronous which ultimately helps us to determine the delay and predict the output precisely. The IDELAYCTRL block should be instantiated with the IODELAY1 block since IDELAYCTRL is responsible for the continuous calibration of the delay elements which helps to reduce the influence of process, voltage, and temperature on the output signal with the help of supplied reference clock frequency (REFCLK). Fig. 4 shows the pin out description of the IODELAYE1 and IDELAYCTRL blocks in vertex 6 series.

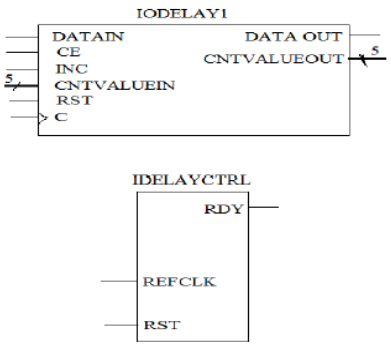


Fig. 4: Pin out description of IODELAY1 and IDELAYCTRL.

The IODELAYE1 block offers three different operational modes when operating in the unidirectional input delay configuration, depending on the mechanism used to select the number of delay taps.

1) *Fixed:* The delay is fixed in this mode at the time of design and the delay cannot be changes in between an operation.

2) *Variable:* In this mode the delay factor depends on the input control signals CE and INC. Increment/decrement signal (CE) determines the delay by incrementing or decrementing the number of delay taps depending on the INC signal is activated or not. The RESET input when activated resets the delay block to a predefined delay value.

3) *Loadable variable:* This mode is similar to the variable mode operation except the delay factor is determined by a 5 bit input value provided to the CNTVALUEIN in port. When in this mode, the IODELAYE1 reset signal resets the delay value to a value set by the CNTVALUEIN, as shown in Fig. 5. The delay time is, therefore, calculated as $t_d = t_{tap}$. Similar to the variable operation the reset signal is activated to reset the delay value i.e., the 5 bit data given to CNTVALUEIN. The delay value when updated can be read from CNTVALUEOUT port.

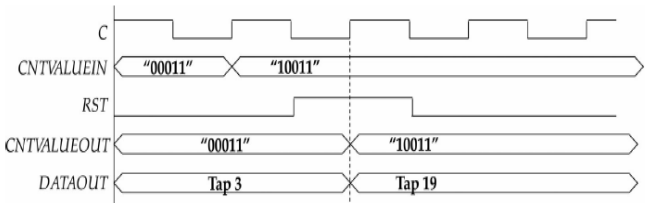


Fig. 5 IODELAY1 operation in the loadable variable mode.

A. Architecture

Fig.6 shows the implementation for an m+1-bit HRPWM using the IODELAY1 block. The design comprises of a counter and comparators as the traditional DPWM technique. Similar to the DCM DPWM technique this design uses Mixed-Mode Clock Manager(MMCM) block to handle the clock signals of this design.

The difference of this design is that it replaces multiplexers from the traditional designs and uses IODELAYE1 block for increasing the resolution and to make the implementation easier. Signals SETD and CLRD are generated comparing the counter output with zeros for SETD and the most significant bits (dc(m:5) in the figure) for CLR. The CLR signal then is given to the IODALAY1 block for setting up the reset signal for the SR latch.

In this design we use the IODELAY1 block in variable mode so that we can use the CNTVALUEIN port to set up the delay using 5 bit input data. The CNTVALUEIN sets up the delay using the 5 bit data and is delay factor can be updated using the reset signal (RST). Since the design is synchronous it provides more precision in providing and predicting the output. The clock signal provided in the input port C. The maximum delay which can be provided by the IODELAY1 block is 32- tap delay. The same delay factor demand the counter clock frequency to be the double of the IODELAY1 clock frequency. This demand is met by MMCM which does the same function as DCM block in the first design.

As mentioned above, when we use IODELAY1 block we have use IDELAYCTRL block also with it whose main function is to auto calibrate the tap delay efficiently. MMCM is responsible for providing different output clock frequency and this is done by using the attributes like M, D, and O since $f_{CKO} = M/(D \cdot O)$.

In addition to this, the IDELAYCTRL is instantiated in order to autocalibrate the delay tap as previously explained.

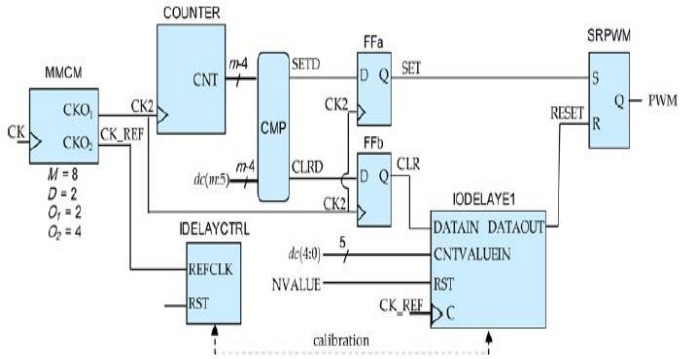


Fig. 6 : DPWM design using IODELAY1.

Figure 7 shows the basic operation of the proposed IODELAYE1- based HRPWM architecture with dc =“10110100.” The CLRD signal is activated when dc(7:5) =CNT (7:5) = “100” = 4. The resulting pulse is captured in the next clock cycle by D flip-flop, which generates the input signal for the IODELAYE1 block to the input DATAIN. The IODELAYE1 block then will generate the RESET signal by delaying the CLR signal according to the CNTVALUEIN input which is the tap cycles dc(4:0) = “10011” = 19. This signal clears the SR latch to generate the desired PWM signal.

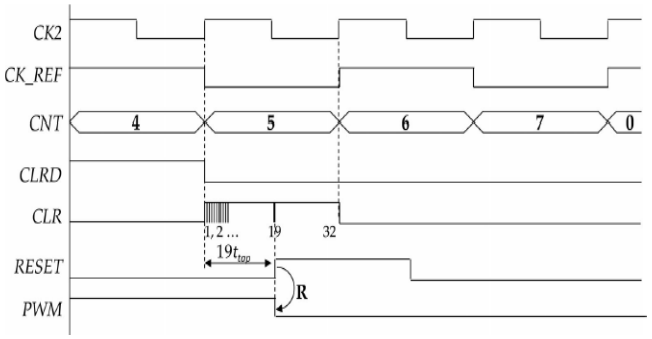


Fig. 7: IODELAY1-based HRPWM operation with dc="10010011".

V. PLL BASED HRPWM

The main components of this architecture are shown in Fig.8. and described next. The key in this DPWM is the flexibility in the management of clock signals provided by the PLLs. With these blocks it is possible to change the phase of clock signals, as well as to multiply and divide its frequency in a controlled manner. The PLL multiplies the system’s clock frequency from 25MHz to 250MHz. Besides the main output P0, three additional outputs P90, P180 and P270 are generated

with phase shifts of 90º, 180º, and 270º respectively. The use of these signals allow for an effective increase in the resolution of the DPWM, adding two more input bits providing a fine adjustment of the duty-cycle reaching 1ns resolution.

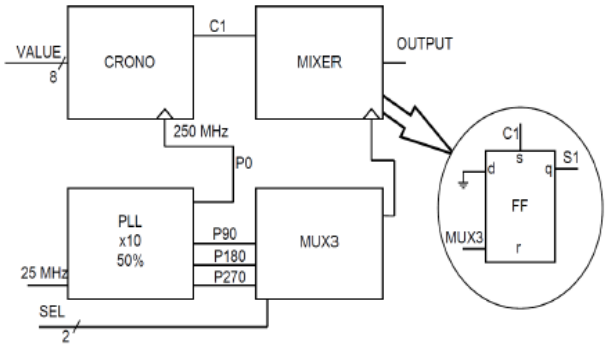


Fig. 8: Simplified diagram of the PLL-based DPWM architecture.

Delay differences due to different paths inside the multiplexor may lead to non-monotonic and non-linear behavior. These undesired effects are eliminated by a calibrating procedure: a post-fitting simulation is run to estimate the delay differences and the phase shifts at the PLL outputs are adjusted accordingly. Special care must be taken to lock on-chip placement of the DPWM module resources after this calibration is completed. Otherwise the insertion of additional circuit elements (as the JTAG interface module used during the experiments to provide module inputs) can completely modify the placement of the circuit inside the chip and therefore also the timing and delays, leaving the previous calibration useless.

VI. SERDES BASED HRDPWM

This architecture is based on the use of a serializer-deserializer (SERDES) module and the low voltage differential signal (LVDS) I/O standard. This serialization technique is commonly used for data transmission over high speed serial I/O lines and is extensively supported by FPGA providers. It can also be applied to enhance DPWM resolution as explained next.

A. Serializer-deserializer(SERDES)

It is a pair of functional blocks commonly used in high speed communications to compensate for limited input/output. These blocks convert data between serial data and parallel interfaces in each direction. The term "SERDES" generically refers to interfaces used in various technologies and applications. The primary use of a SERDES is to provide data transmission over a single/differential line in order to minimize the number of I/O pins and interconnects.

Figure 9 shows the Serializer and Deserilizer Pulses. The basic SERDES function is made up of two functional blocks: the Parallel In Serial Out (PISO) block and the Serial In Parallel Out (SIPO) block. There are 4 different SERDES architectures:

(1) Parallel clock SERDES.

(2) Embedded clock SERDES.

(3) 8b/10b SERDES.

(4) Bit interleaved SERDES.

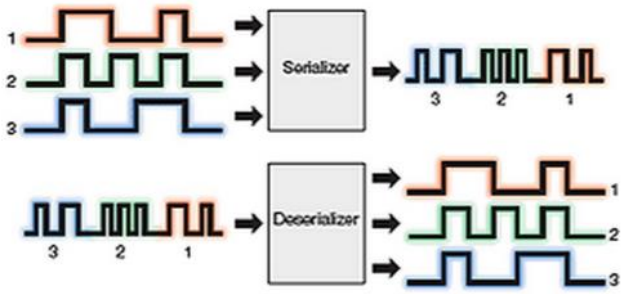


Fig. 9: Serializer and Deserilizer Pulses.

The PISO (Parallel Input, Serial Output) block typically has a parallel clock input, a set of data input lines, and input data latches. It may use an internal or external phase-locked loop (PLL) to multiply the incoming parallel clock up to the serial frequency. The simplest form of the PISO has a single shift register that receives the parallel data once per parallel clock, and shifts it out at the higher serial clock rate. Implementations may also make use of a double-buffered register to avoid metastability when transferring data between clock domains.

The SIPO (Serial Input, Parallel Output) block typically has a receive clock output, a set of data output lines and output data latches. The receive clock may have been recovered from the data by the serial clock recovery technique. However, SERDES which do not transmit a clock use reference clock to lock the PLL to the correct Transmission frequency, avoiding low harmonic frequencies present in the data stream. The SIPO block then divides the incoming clock down to the parallel rate. Implementations typically have two registers connected as a double buffer. One register is used to clock in the serial stream, and the other is used to hold the data for the slower, parallel side.

Some types of SERDES include encoding/decoding blocks. The purpose of this encoding/decoding is typically to place at least statistical bounds on the rate of signal transitions to allow for easier clock recovery in the receiver, to provide framing, and to provide DC balance.

B. Architecture

The architecture is shown in Figure 10 . The SERIAL block receives a 4-bit parallel vector synchronized with the system's clock (200 MHz), serializing these bits at the output beginning with the most significant bit. In this case the output rate is 800Mbps, which will have a resolution of 1,25 ns. During the relevant clock period at the end of the pulse on the counter comparator output, a thermometer encoding of SEL is presented to the serializer module input, thus providing the fine adjustment resolution control.

The limitation to the resolution is imposed by the maximum data rate tolerated by the FPGA's high speed LVDS

output pins, that was 805MHz on the Cyclone II chip used during measurements. This limit has been increased up to 1080MHz on newer low-cost chips.

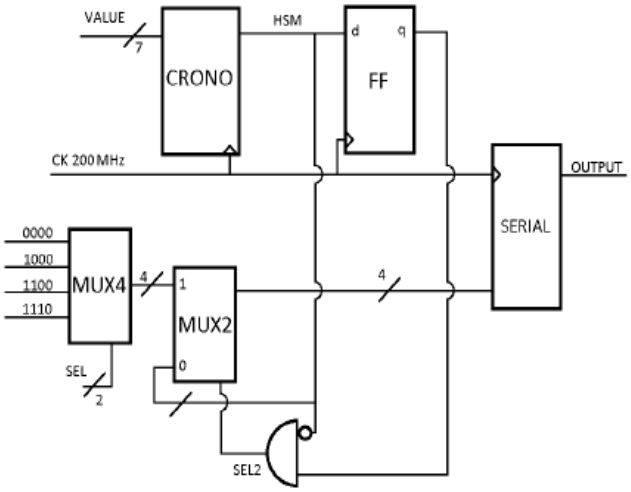
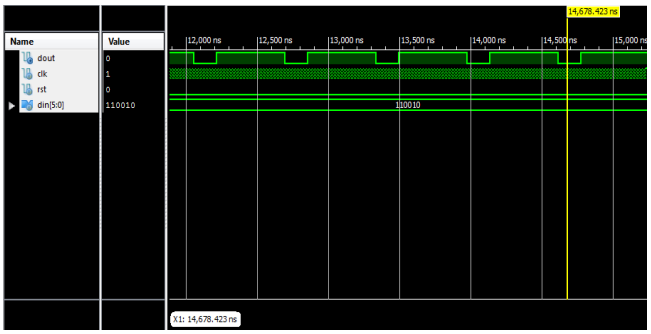


Fig. 10: Simplified diagram of SERDES-based DPWM architecture.

VII. RESULTS

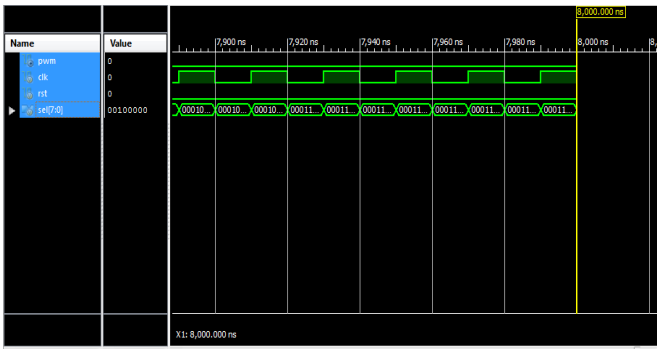
A. *HRDPWM architecture using DCM block*

DPWM output of DCM-based HRPWM operation with dc = “110010”



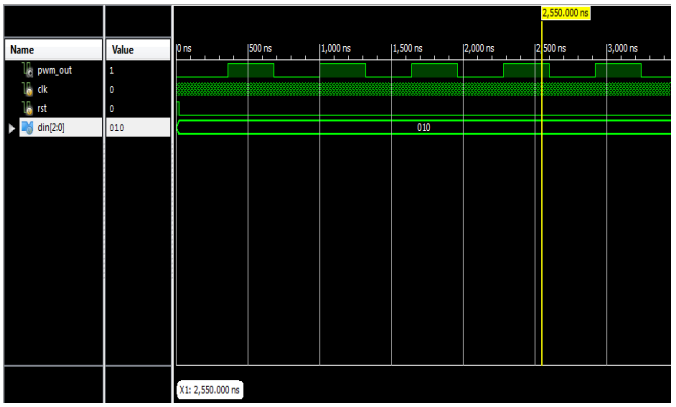
B. *HRDPWM architecture using IODELAYe1 block*

DPWM output for HRPWM using IODELAYe1



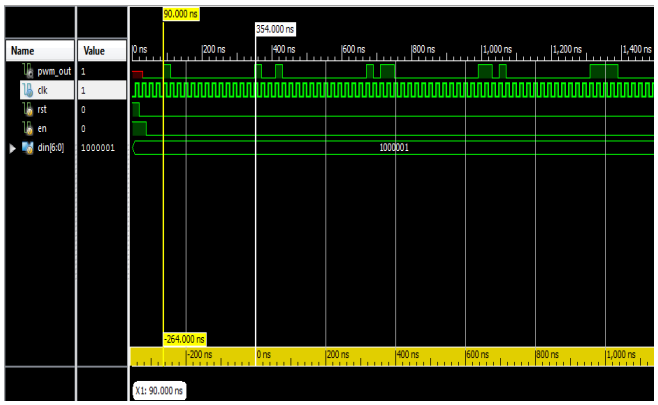
C. *PLL based HRPWM*

DPWM output for dc="010



D. *SERDES based HRDPWM*

DPWM output for SERDES based HRPWM dc="1000001"



VIII. COMPARISION

Table 1 shows the comparative study of four HRDPWM

Table 1 Comparative study of four HRDPWM

Types of Architecture	DCM	IODELAY	PLL	SERDES
Maximum Frequency	98.122 MHz	122.942 MHz	274.397 MHz	276.14 MHz
Minimum input arrival time before clock	5.26 ns	4.203 ns	2.724 ns	4.537 ns
Maximum output required time after clock	6.231 ns	6.216 ns	6.141 ns	6.21 ns

IX. CONCLUSION

Advances in power electronics and digital control have made necessary development of high-resolution DPWMs to take the most of either high-frequency or high-precision power converters. The Four DCM-based synchronous architectures have been designed and verified its results. PLL based DPWM uses less components compared to other three architectures, where as SERDES based DPWM can be used where applications does not need DCM blocks. Both PLL based and SERDES based can operate in the high frequency.

REFERENCES

[1] "Analysis and Implementation of Low-cost FPGA Based Digital Pulse-width Modulators", Ignacio de Leon, Gonzalo Sotta, Gabriel Eirea, Julio Perez Acle, 2014 IEEE, Facultad de Ingenieria Universidad de la Republica Uruguay.

[2] "Synchronous FPGA-Based High-Resolution Implementations of Digital Pulse-Width Modulators", Denis Navarro, O´ scar Luc´ıa, Member, IEEE, Luis Angel Barraga´n, Jose´ Ignacio Artigas, Isidro Urriza, and O´ scar Jime´nez, Student Member, IEEE transactions on power electronics, vol. 27, no. 5, may 2012.

[3] "A Novel FPGA Based Synchronous High Resolution Digital Pulse Width Modulators for Inverters", Shijiny Basheer, Chithra M, IOSR Journal of Engineering (IOSRJEN) Vol. 04, Issue 03 (March. 2014),

[4] "A versatile digital pulsewidth modulation architecture with area-efficient FPGA implementation." R.F. Foley, R.C. Kavanagh, W.P. Marnane and M.G. Egan, Power Electronics Specialists Conference, PESC 2005.

[5] "Hybrid DPWM with digital delay-locked loop." V. Yousefzadeh, T. Takayama, D. Maksimovic. IEEE Computer in Power Electronics Workshop, COMPEL 2006.

[6] "FPGA based digital pulse width modulator with time resolution under 2 ns."S.C. Huerta, A. de Castro, O. Garcia, J.A. Cobos. IEEE Tran. on Power Electronics, Vol. 23, No. 6, November 2008.

[7] "DPWM based on FPGA clock phase shifting with time resolution under 100 ps."A. de Castro, E. Todorovich. Power Electronics Specialists Conference, PESC 2008.

[8] "Segmented digital clock manager FPGA based digital pulse width modulator technique." M.G. Batarseh, W. Al-Hoor, L. Huang, C. Iannello, I. Batarseh. Power Electronics Specialists Conference, PESC 2008.

[9] "High Resolution FPGA DPWM Based on Variable Clock Phase Shifting,"A, De Castro, E. Todorovich Power Electronics, IEEE Transactions on , vol.25, no.5, pp.1115,1119, May 2010.

[10] "Simple digital pulse width Modulator with 60 picoseconds resolution using a low-cost FPGA," D. Costinett, M Rodriguez and D. Maksimovic, 15th International Power Electronics and Motion Control Conference, EPE-PEMC 2012.

[11] "SiC versus Si evaluation of potentials for performance improvement of inverter and dc–dc converter systems by SiC power semiconductors," J. Biela, M. Schweizer, S. Waffler, and J. W. Kolar, IEEE Trans. Ind. Electron., vol. 58, no. 7, pp. 2872–2882, Jul. 2011.

[12] "Switching characteristics of GaN HFETs in a half bridge package for high temperature applications," T. Nomura, M. Masuda, N. Ikeda, and S. Yoshida, IEEE Trans. Power Electron., vol. 23, no. 2, pp. 692–697, Mar. 2008.

[13] "Ultra high frequency dc-to-dc converters using GaAs power switches," S. Ajram and G. Salmer, IEEE Trans. Power Electron., vol. 16, no. 5, pp. 594–602, May 2001.

[14] "Modeling digital control laws for high-frequency VRM applications," A. Simon-Muela, Y. Elbasri, C. Alonso, V. Boitier, and J. L. Chaptal, in Proc. IEEE Power Electron. Spec. Conf., 2008, pp. 1560–1565.

[15] "High efficiency dc–dc converter with twin-bus for dimmable LED lighting," W. Yu, J. S. Lai, H. Ma, and C. Zheng, IEEE Trans. Power Electron., vol. 26, no. 8, pp. 2095–2100, Jul. 2011.

[16] "Architecture and IC implementation of a digital VRM controller," A. V. Peterchev, X. Jinwen, and S. R. Sanders, IEEE Trans. Power Electron., vol. 18, no. 1, pp. 356–364, Jan. 2003.

[17] "Phase-shift control of dual half-bridge inverter feeding coupled loads for induction heating purposes," C. Carretero, O. Luc´ıa, J. Acero, and J. M. Burd´ıo, Electron. Lett., vol. 47, no. 11, pp. 670–671, May 2011.

[18] "High-frequency digital PWM controller IC for dc–dc converters," B. J. Patella, A. Prodic, A. Zirger, and D. Maksimovic, IEEE Trans. Power Electron., vol. 18, no. 1, pp. 438–446, Jan. 2003.

[19] "High-resolution pulse width modulator (HRPWM)," TMS320x28xx, Reference Guide, Texas Instruments SPRU924C, Dallas, TX, 2007.

[20] "Digital pulse width modulator architectures," A. Syed, E. Ahmed, D.Maksimovic, and E. Alarcon, in Proc. IEEE Power Electron. Spec. Conf., Jun. 2004, vol. 6, pp. 4689–4695.

[21] "Universal and fault tolerant multiphase digital PWM controller IC for high-frequency dc–dc converters," Z. Lukic, C. Blake, S. C. Huerta, and A. Prodic, in Proc. IEEE Appl. Power Electron. Conf., Feb./Mar. 2007, pp. 42–47.

[22] "FPGA-based digital pulse-width modulator with time resolution under 2 ns,"S. C. Huerta, A. de Castro, O. Garcia, and J. A. Cobos, IEEE Trans. Power Electron., vol. 23, no. 6, pp. 3135–3141, Nov. 2008.

[23] "FPGA-based multiphase digital pulse width modulator with dual-edge modulation," M. Scharrer, M. Halton, T. Scanlan, and K. Rinne, in Proc. IEEE Appl. Power Electron. Conf. Expo., 2010, pp. 1075–1080.

[24] "FPGA-based digital pulse width modulator with optimized linearity," M. Scharrer, M. Halton, and T. Scanlan, in Proc. IEEE Appl. Power Electron. Conf. Expo., 2009, pp. 1220–1225.

[25] "Segmented digital clock manager FPGA based digital pulse width modulator technique," M. G. Batarseh,W. Al-hoor, L. Huang, C. Iannello, and I. Batarseh, in Proc. Power Electron. Spec. Conf., 2008, pp. 3036–3042.

[26] "DPWM based on FPGA clock phase shifting with time resolution under 100 ps," A. de Castro and E. Todorovich, in Proc. IEEE Power Electron. Spec. Conf., 2008, pp. 3054–3059.

[27] "FPGA based digital control with high-resolution synchronous DPWM and high-speed embedded A/D converter," J. Quintero, M. Sanz, A. Barrado, and A. Lazaro, in Proc. IEEE Appl. Power Electron. Conf. Expo., 2009, pp. 1360–1366.