A New FPGA-Based Segmented Delay-Line DPWM With Compensation for Critical Path Delays

Xin Cheng, Ruifeng Song, Guangjun Xie, Yu Zhang, and Zhang Zhang

***Abstract*—The duty cycle in digital pulse width modulation (DPWM) whose time resolution is as high as hundreds of picosecond will be increased since the propagation delays of internal logics and interconnects are superimposed on the on-time. The duty-cycle-increment phenomenon affects the regulation performance of converters, especially when the delays are the same magnitude level as switching period. In this paper, a compensation method based on tool command language commands is used to offset the propagation delays. Furthermore, the DPWM applies a counter, phase locked loop, and carry chain to constitute segmented delay-line, while it maintains a wide duty cycle range and a high time resolution. This DPWM is implemented by a low-cost field-programmable gate array Cyclone IV. It achieves a good linearity where R2 is 0.9949 for an 11-bit, 9.375 MHz switching frequency DPWM. The duty cycle range is from 1.52% to 97.81%. Meanwhile, the time resolution and average duty difference are 53 ps and 3.08%, respectively.**

***Index Terms*—Compensation, digital pulse width modulation (DPWM), field-programmable gate array (FPGA), segmented delay-line, time resolution.**

I. Introduction

DIGITAL control in power converters has obtained wide research attention due to the rapid development of semiconductor technology and its advantages, such as programmability, flexibility, expandability, and advanced control algorithms [1]–[5]. In digital control systems, the resolution of digital pulse width modulation (DPWM) must be higher than analog-to-digital converter's for avoiding the limited cycling [6], [7]. In order to meet the requirement of regulation accuracy and avoid the unexpected limit cycling, a DPWM system needs to maintain high standard in monotonicity and resolution. Hence, it imposes great challenges on the design of high performance DPWM system.

In order to enhance regulation accuracy, digital–analog hybrid PWM architectures are designed to realize advanced control algorithms in a custom integrated circuit environment [8]–[11], which also improve system stability and reduce power dissipation. But the complex control algorithms implemented in a custom integrated circuit environment increase design difficulty and cost consumption. For example, Sun *et al.* [9] use the MPPT algorithm to obtain the peak reference voltage, which controls a current generated by operational transconductance amplifier, then, the P-DPWM module detects the variation of the current to generate pulse signal. On the other hand, passive devices such as capacitors and inductors increase the chip area, and these designs always are easily affected by the variation of process, voltage, and temperature (PVT) [10], [11].

Taking into account these problems, many literatures propose various algorithms to simplify the design, which are usually based on field-programmable gate array (FPGA) where standard internal logic resources like phase-locked loop (PLL), digital clock manager (DCM), and carry chain are embedded. Due to the requirement for time to market and programmability, utilizing FPGA device for designing DPWM system is more attractive and more widely used [12]–[18]. These methods always use time resolution and linearity to quantize the regulation accuracy of converter, and all of these designs achieve high time resolution, which means high regulation accuracy. All of these architectures are based on counter. A DPWM-based on delay-locked loop (DLL) and counter in [12] can obtain high time resolution of 2 ns. A cascade segmented DCM-based DPWM in [15] and a DCM-based DPWM in [17] keep a resolution as high as 625 ps. Costinett *et al.* [18] come up with a carry chain-based DPWM block, which gets an even higher resolution about 60 ps. However, when a DPWM architecture runs with switching frequency about tens of MHz or higher and time resolution under nanosecond level, the delays of critical path including logic elements and interconnects are the same magnitude level as switching period, thus the on-time of PWM signal and duty cycle are increased due to the superposition of critical path delays. This duty-cycle-increment phenomenon greatly affects the regulation performance of converter.

Motivated by the above concerns, this paper presents a compensation technique for critical path delays. Actually, implementing DPWM designs in FPGA with time resolution of ns-level requires manual placement and routing. To make sure the critical path delays are equal in all cases, an improved multiplexer in critical path with symmetrical structure is presented. In addition, logic assignments are used to assign the logics in the critical path by tools, the on-chip global resources are used to get an equal routing delay. To compensate the critical path delays, a *set_net_delay* command is presented. In a sense, the

Manuscript received April 25, 2017; revised August 10, 2017; accepted October 3, 2017. Date of publication October 15, 2017; date of current version September 28, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 61401137, Grant 61674049, and Grant 61404043, and in part by the Anhui Science and Technology Major Project under Grant 16030901007. Recommended for publication by Associate Editor K.-H. Chen. *(Corresponding author: Zhang Zhang.)*

The authors are with the School of Electronic Science and Applied Physics, Hefei University of Technology, Hefei 230009, China (e-mail: ceciliacheng1017@163.com; 271277182@qq.com; gjxie8005@hfut.edu.cn; 874213521@qq.com; zhangzhang@hfut.edu.cn).

Color versions of one or more of the figures in this paper are available online at http//ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TPEL.2017.2763750

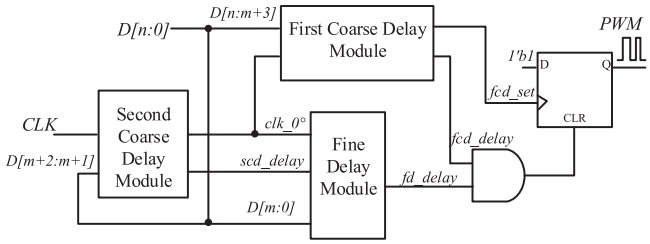


Fig. 1. Top-level architecture of the proposed segmented delay-line DPWM.

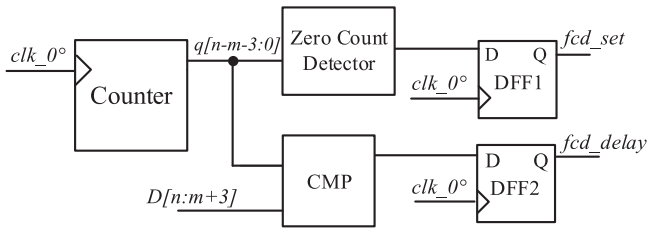


Fig. 2. Proposed first coarse delay module.

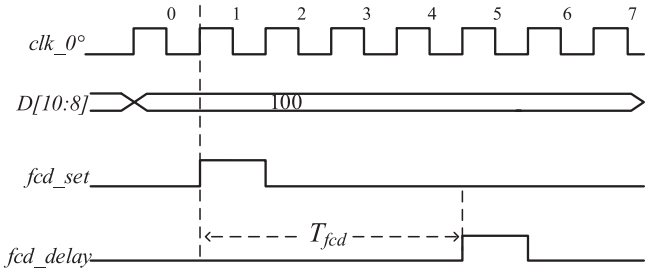


Fig. 3. Timing diagram of first coarse delay module.

increased duty cycle can be adjusted by the optimization and compensation for critical path.

The rest of the paper is organized as follows. The new segmented delay-line DPWM architecture is proposed in Section II. The duty-cycle-increment phenomenon is explained and the solution methods including optimization and compensation techniques for critical path are introduced in Section III. The experimental results and conclusion are given in Sections IV and V, respectively.

II. Proposed DPWM Architecture

The top-level architecture of the proposed DPWM is shown in Fig. 1. It includes a counter-based first coarse delay module, a PLL-based second coarse delay module and a carry-chain-based fine delay module. CLK is the external clock signal of DPWM, $D[n:0]$ is the digital stream signal. The PLL in second coarse delay module generates a synchronous clock $clk\_0°$ by frequency multiplication and 0° phase shifting from CLK. When the counter value in first coarse delay module is equal to 0, the first delay module outputs a rising edge so that the output signal PWM is set to 1. Simultaneously, $D[n:m+3]$ $(n > m+3)$ selects an appropriate multiple of clock period accordingly as first coarse delay. At the same time, the second coarse delay module delays a second coarse delay which is controlled by $D[m+2:m+1]$. Finally, the fine delay module delays a fine delay, which is controlled by $D[m:0]$. After delaying the three propagation delays, the PWM is reset to 0, thus obtaining the required duty cycle. The details about operation principle will be stated in next paragraphs.

A. First Coarse Delay Module Based on Counter

The proposed first coarse delay module, as shown in Fig. 2, mainly consists of three blocks: a counter, a zero-count-detector, and a digital comparator. With the synchronous clock $clk\_0°$, the synchronism between the first coarse delay module and the other two modules can be guaranteed by two D flip-flops (FFs).

The key function of first coarse delay module is to generate a first coarse delay (T_{fcd}) with clock period unit, which depends on $D[n:m+3]$ $(n > m+3)$ and satisfies the following relationship:

$$T_{fcd} = x \cdot t_{fcd} (0 \le x \le 2^{n-m-2} - 1) \tag{1}$$

$$t_{fcd} = T \tag{2}$$

where x is the decimal number corresponding to $D[n:m+3]$, t_{fcd} is the first coarse delay unit and T is the period of $clk\_0°$.

Fig. 3 shows a timing diagram of the first coarse delay module when $n = 10$ and $m = 5$. At first, counter counts the synchronous clock $clk\_0°$ and outputs a 3-bit binary number $q[2:0]$. For $D[10:8] = 3'\text{b}100$, when zero count detector detects that the counter output $q[2:0]$ is 3'b000, it outputs 1 to the D-terminal of D flip-flop1 and the output of this FF fcd_set goes high in the next clock. When $q[2:0] = 3'\text{b}100$ (equals $D[10:8]$), the output signal of flip-flop2 fcd_delay goes high in next clock and lasts for a clock period. Finally, as shown in Fig. 1, fcd_delay is combined with fd_delay through an AND gate to generate the reset signal of D flip–flop. According to the above analysis, the first coarse delay module delays four cycles of $clk\_0°$, namely $T_{fcd} = 4T$ as the first coarse delay.

B. Second Coarse Delay Module Based on PLL

As the digital clock management unit of Altera FPGA internal resources, on-chip PLL is the key part of second coarse delay module. It can divide and multiply the frequency of input clock signal, configure the duty cycle of clock signal, and generate four clock signals with different phase shifts 0°, 90°, 180°, and 270°. The on-chip PLL also has the ability to provide a clock with zero transmission delay, low jitter, and high phase-shift resolution [19]. As a default, the generated phase shift from PLL can be viewed as the delayed version if input clock with duty cycle of 0.5. However, when four signals with 25% phase shifts between each other goes through a multiplexer, it brings nonlinear behavior [12]. To overcome this problem, the duty cycles of $clk\_0°, clk\_90°, clk\_180°$, and $clk\_270°$ should be configured to be less than 25%.

The structure of second coarse delay module is shown in Fig. 4. Its function is to generate a second coarse delay (T_{scd}), which is determined by $D[m+2:m+1]$ and it satisfies the

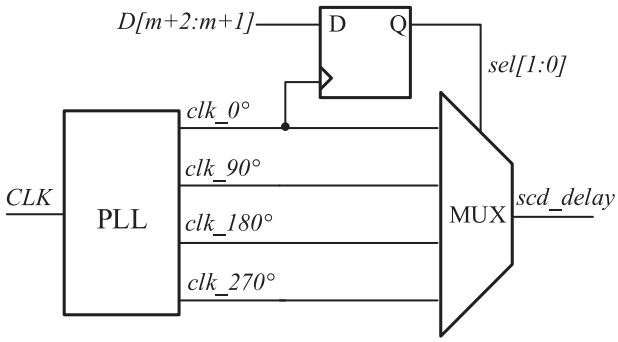


Fig. 4. Proposed PLL-based second coarse delay module.

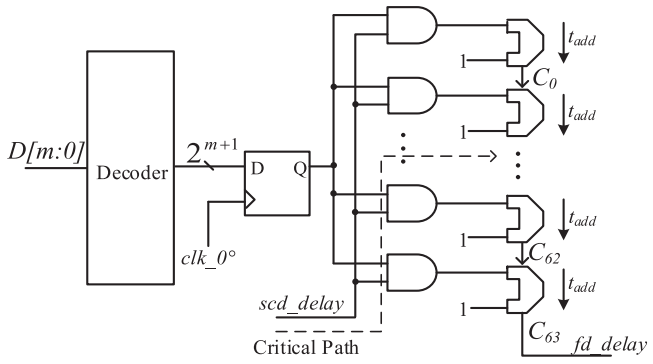


Fig. 5. Proposed fine delay module.

following relationship:

$$T_{scd} = y \cdot t_{scd} (0 \le y \le 3) \tag{3}$$

$$t_{scd} = T/4 \tag{4}$$

where y is the decimal value corresponding to $D[m+2:m+1], t_{scd}$ is the second coarse delay unit. The PLL phase shift is used to match clk_0°. In other words, one clock period T is divided into four second coarse delay units. The PLL is used to configure four clocks: $clk\_0^\circ, clk\_90^\circ, clk\_180^\circ, clk\_270^\circ$ with duty cycle less than 25%. At the same time, $D[m+2:m+1]$ selects the corresponding phase shift clock as the second coarse delay after synchronism with other modules by D FFs.

C. Fine Delay Module Based on Carry-Chain

A large number of logic elements are embedded in Altera FPGA, which is composed of combinational logic (COMB) and FF. For COMB, it can be configured in either normal mode or arithmetic mode with a look-up table structure. When a carry-chain is used in a design, the COMB is configured in arithmetic mode [19]. The carry-chain is a low-latency path, which propagates the carry bit through two continuous 2-bit adders whose routing delay is zero, and achieves high-speed operation. The propagation delay for each adder is predictable with a range from 10 to 100 ps, depending on FPGA process and speed. As fine delay unit, the delay t_{add} determines the minimum time resolution of DPWM.

As shown in Fig. 5, this fine delay module architecture employs carry-chain, which is similar to literature [18]. C_i $(i = 0, \ldots, 63)$ is the carry for each adder, the D FFs play a role in keeping synchronization among the three delay modules, so that the critical path is no longer from decoder to carry chain, but from the second coarse delay module to carry-chain. AND gates are used to control the number of fine delay units when the scd_delay signal goes high. The sum of selected fine delay units constitutes the final fine delay T_{fd}, in the following manner:

$$T_{fd} = z \cdot t_{add} (0 \le z \le 2^{m+1} - 1) \tag{5}$$

where z is the decimal value corresponding to $D[m:0], t_{add}$ is the delay of adder. In order to ensure the continuity between second coarse delay unit and fine delay unit, the design need meet the following relationship as much as possible:

$$t_{scd} = 2^{m+1} \cdot t_{add}. \tag{6}$$

Considering the variation of logic delay in physical implementation, designer must modify the frequency of synchronous clock generated by PLL iteratively according to the delay of carry-chain in order to satisfy the (6) as much as possible.

D. Functionality of the DPWM Architecture

The three modules mentioned above constitute the whole segmented delay-line DPWM, which can be seen in Fig. 6. The on-time of output signal is generated by the sum of the three delays. Therefore, when ignoring critical path delays, the DPWM duty cycle is expressed as

$$\begin{aligned} \text{DutyCycle} &= \frac{T_{\text{on}}}{2^j \cdot T} \\ &= \frac{T_{fcd} + T_{scd} + T_{fd}}{2^j \cdot T} \\ &= \frac{x \cdot t_{fcd} + y \cdot t_{scd} + z \cdot t_{add}}{2^j \cdot T} \\ &\approx \frac{x \cdot T + y/4 \cdot T + (z/2^{m+3}) \cdot T}{2^j \cdot T} \\ &= \frac{x + y/4 + z/2^{m+3}}{2^j} \end{aligned} \tag{7}$$

where j presents counter bits, x, y, z are the decimal values corresponding to $D[n:m+3], D[m+2:m+1]$, and $D[m:0]$, respectively. Note that there is an approximate equal in (7), the frequency of synchronous clock generated by PLL must be modified iteratively according to the delay of carry-chain since t_{add} changes with PVT variations.

Fig. 7 illustrates the DPWM timing operation when $n = 10, m = 5$. Assuming that $D[10:0] = 11'\text{b}011\_01\_110000$, when zero-count-detector detects the counter value is zero, fcd_set goes high in the next synchronous clock so that the PWM signal becomes high. Equation (1) becomes $T_{fcd} = 3T$ as first coarse delay. Simultaneously, the second coarse delay is $T_{scd} = (1/4)T$ according to (3) and (4). Subsequently, the fine delay module offers a fine delay $T_{fd} = 48t_{add}$ based on (5). In such a case, the high-levels of fcd_delay and fd_delay are fed

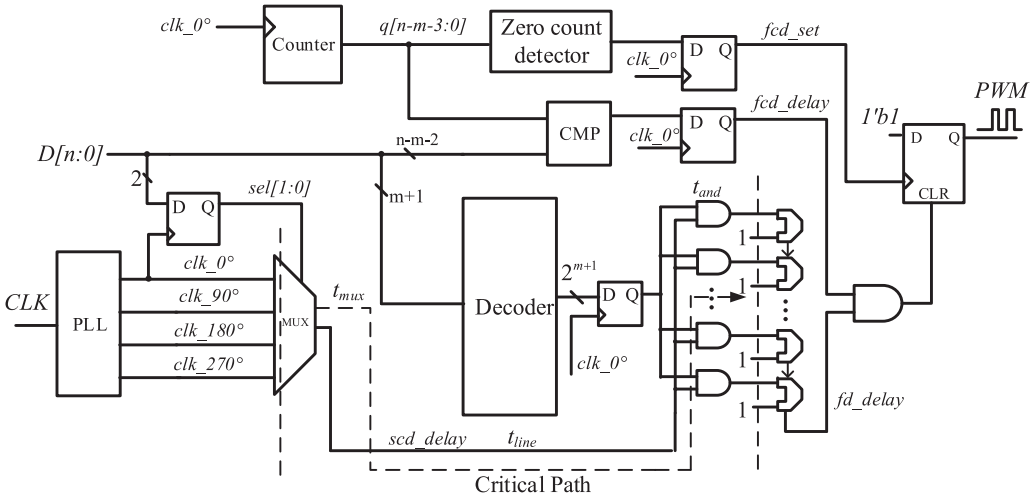


Fig. 6. Proposed counter-based, PLL-based, and carry-chain-based segmented delay-line DPWM.

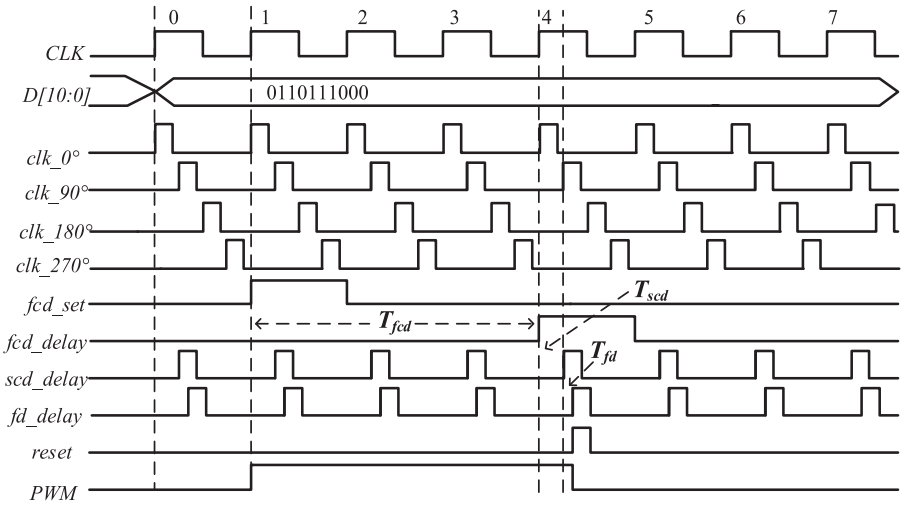


Fig. 7. Timing operation of the DPWM when $n = 10, m = 5$.

into the AND gate and $PWM = 0$. According to (7), the duty cycle is

$$\text{DutyCycle} = \frac{(3 + 1/4)T + 48/256 \cdot T}{8 \cdot T} = 0.4297.$$

III. DPWM Implementation

A. Duty-Cycle-Increment Phenomenon

When a DPWM architecture runs with switching frequency about tens of MHz or higher and time resolution under nanosecond level, the critical path delays including logic elements and interconnects are the same magnitude level as switching period. Therefore, the on-time of PWM signal is enlarged due to the superposition of T_{cpath}, which is denoted as

$$T_{\text{on}} = T_{fcd} + T_{scd} + T_{fd} + T_{\text{cpath}} \tag{8}$$

where $T_{\text{cpath}} = t_{\text{mux}} + t_{\text{line}} + t_{\text{and}}$ is the critical path delays as shown in Fig. 6. This causes a longer T_{on} and a larger duty cycle, which is expressed as

$$\text{DutyCycle} = \frac{x + y/4 + z/2^{m+3}}{2^j} + \frac{T_{\text{cpath}}}{2^j \cdot T}. \tag{9}$$

Compared with (7), the duty cycle difference is

$$\Delta(\text{DutyCycle}) = \frac{T_{\text{cpath}}}{2^j \cdot T}. \tag{10}$$

Fig. 8 shows this problem where $PWM1$ does not consider the critical path delays while $PWM2$ does. In other words, $PWM1$ is ideal signal without physical delay. It can be seen that compared with $PWM1$, the delays of critical path enlarge the duty cycle of $PWM2$. To adjust the duty cycle, the paper presents a compensation method as $PWM3$ shows, that is making fcd_set delay T_{cpath} to offset the increased duty cycle.

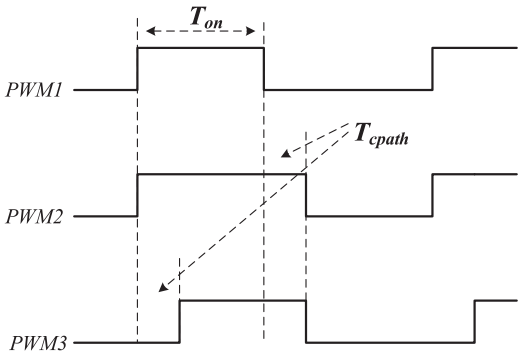


Fig. 8. Duty-cycle-increment phenomenon and its solution.

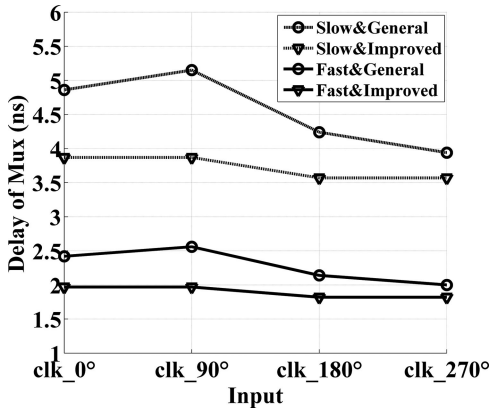


Fig. 9. Delays of general and improved MUX.

B. Optimization for the Critical Path

As mentioned above, the DPWM architecture in Fig. 6 has a larger duty cycle due to the delays of critical path T_{cpath}. With the increasing time resolution and switching frequency in DPWM, the difference between two adjacent duty cycle values becomes smaller, so it is more necessary to optimize and compensate the critical path, namely manual place and route of critical path are required.

First of all, the synthesis tool generally uses a four input look up table for multiplexer whose logic elements become asymmetric after synthesis. Therefore, it cannot obtain equal propagation delays of the signals passing through multiplexer in all cases. To solve this problem, the optimization method in this paper is to modify the hardware description language into a symmetric multiplexer structure, that is, configuring *ALTCLKCTRL*. A multiplexer code in Verilog is given in Appendix A. Configuring *ALTCLKCTRL* also can avoid glitches and any possible hold time problem on the device due to logic delay on the clock line [20], [21]. For multiplexer delays, the simulation results on multiplexer are shown in Fig. 9. In fast-corner, the signal delays are 2.0∼2.56 ns and 1.82∼1.97 ns, and the largest delay differences are 560 and 150 ps for general and improved multiplexer, respectively. In slow-corner, the signal delays are 3.94∼5.15 ns and 3.57∼3.87 ns, the largest delay differences are 1210 and 300 ps, respectively. Compared with general multiplexer, both signal delays and delay difference of the improved multiplexer

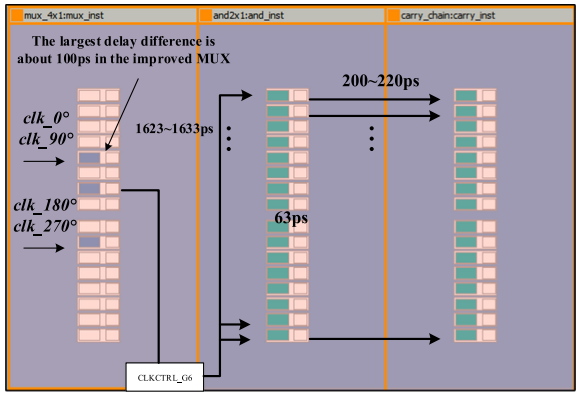


Fig. 10. Optimization for critical path with matched delays in fast-corner simulation environment.

are smaller. It is important to optimize the multiplexer because that the delay difference of critical path is mainly provided by multiplexer.

Second, in order to guarantee that the routing delays from the output of second coarse delay module to each AND gate in fine delay module are equal, the global signal constraints for these paths are required. Also, location assignments are used to fix the location of the logic elements including ANDs and Adders. More specifically, the locations of ANDs[x] and $\mathrm{Adders[x]}\ (\mathrm{x} = 0, 1, \ldots, 63)$ are arrayed in a straight line to guarantee an equal routing delay between each other in all cases. These constraint methods ensure equal critical path delays in all cases.

Actually, the critical path delays mainly concentrate on the routing between second coarse and fine delay modules. Fig. 10 shows the optimization for critical path by timing analysis tool in fast-corner simulation environment. It can be seen that approximately equal delays of critical path are acquired with a small delay difference about 100 ps, which is mainly supplied by the improved multiplexer. Obviously, 100 ps in fast-corner can be ignored with relative to the critical path delays.

C. Compensation for Critical Path

Although the above optimization techniques acquire equal propagation delays in all possible cases, it cannot eliminate duty-cycle-increment phenomenon. The critical path is optimized very well in [18], but the delays of the critical path are also added to duty cycle, which affects the overall regulation accuracy. A typical compensation method employs a multistage synchronizer to offset the propagation delays. The approach is very accurate, but it requires a fast system clock that causes timing violation. In addition, it is a clock domain crossing method causing the instability of DPWM. For example, for compensating critical path delays of 1 ns, a system clock frequency of at least 1 GHz is needed.

Taking into account the accuracy and reliability of DPWM, the proposed compensation method adds set_net_delay into synopsys design constraints (SDC) File. More especially, a keep attribute is used to the wire in which the output pin of the register in first delay module is fed into the clock pin of the register

Slow 1200mV 85C Model

	Name	Slack	Required	Actual	From	To	Type
1	set_net_delay	-0.337	6.800	6.463	[get_nets {my_pwm|pwmset}]		min
2	set_net_delay	-0.188	6.900	7.088	[get_nets {my_pwm|pwmset}]		max

Fig. 11. Compensation result for critical path delays.

in next stage. As explained in [22], the node shall not be subject to optimization with the presence of this attribute. Then, set_net_delay commands are used to the node to offset critical path delays. The design uses set_clock_groups commands to specify clocks that are asynchronous, which lets the tool cut timing between the specify groups, because set_net_delay commands might cause timing violation if set_clock_groups commands are not added to SDC file. The simulation result is shown in Fig. 11.

It can be seen that the command successfully compensates the critical path delays although it has a slight error, actually, 6.9 ns, which includes two parts. One part is the critical path delays. In order to make the output consistent with the theory, the design also considers the other part delay which is generated from fine delay module to the reset terminal of D flip–flop. Actually, if set_net_delay commands are not used to compensate the critical path delays, the wire itself has a random delay amount to offset the delays. Of course, the design must iterate simulation for critical path delays due to the PVT variations in physical implementation.

IV. Experimental Results

The whole DPWM has been described in Verilog HDL and verified in Cyclone IV, EP4CE15F23C8N FPGA development board. The test instrument is Tektronix MDO4054C oscilloscope. Quartus prime built-in Timequest Timing Analyzer and Chip Planner are used to optimize critical path, compensate critical path delays, and predict the minimum time resolution of the DPWM. Each adder delay is 34 ps in fast-corner and 73 ps in slow-corner verified by Timequest Timing Analyzer.

After modifying the frequency of synchronous clock generated by PLL with the tools, a frequency $fclk\_0^\circ = 75$ MHz is set as synchronization clock of test prototype. Since the counter output in the first coarse delay module $q[2:0]$ is a 3-bit number, the switching frequency f_{sw} is set to 9.375 MHz. Fig. 12(a) shows a smallest duty cycle of 1.52% and a largest duty cycle of 97.81%, which mean a wide duty cycle range. Fig. 12(b) shows the behavior of the DPWM for several duty cycle commands that are increased from 010_00_000000 to 010_10_111000 with step of 8 units. For $D[5:0] > 63$, the least significant bit (LSB) rolls over and the most significant bit (MSB) is increased. The delay difference with step of 8 units is about 400 ps showing a good consistency relationship. When LSB rolls over, the delay-increment is about 800 ps showing a slight nonlinear behavior, which is inevitable due to the (6).

In order to verify that critical path delays increase the DPWM duty cycle, Fig. 13 compares the compensated DPWM output waveforms with uncompensated when the duty cycle commands are increased from 010_00_000000 to 010_10_000000 with one

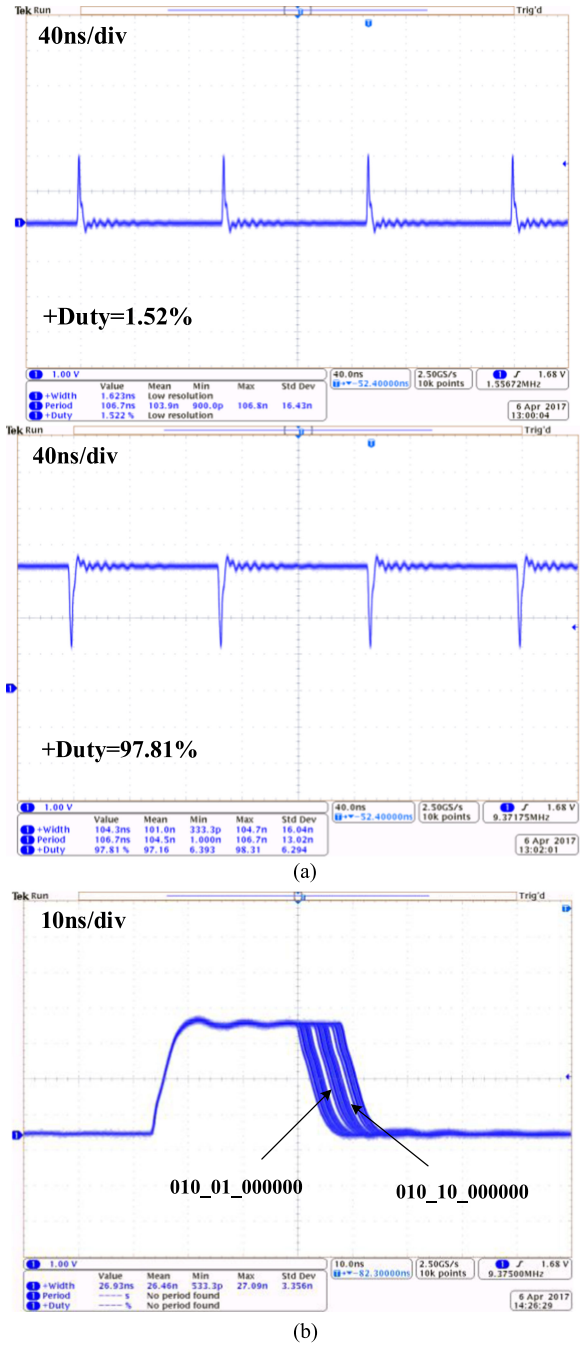


Fig. 12. (a) Smallest and largest duty cycle generated by the DPWM. (b) Details of the waveforms for several duty cycles. Vertical, 1 V/div.

MSB. It can be seen that the positive width of the uncompensated case is obviously larger than the compensated one, that is, the critical path delays increase duty cycle.

In order to measure the time resolution and monotonicity of DPWM and calculate the duty cycle increment caused by critical path delays, the positive width as a function of the commands in the DPWM is captured. Fig. 14(a) shows the positive width for the proposed DPWM, which includes the uncompensated and compensated results when $D[10:6]$ ranges from 0 to 31, corresponding to the MSBs of 5-bit duty command in

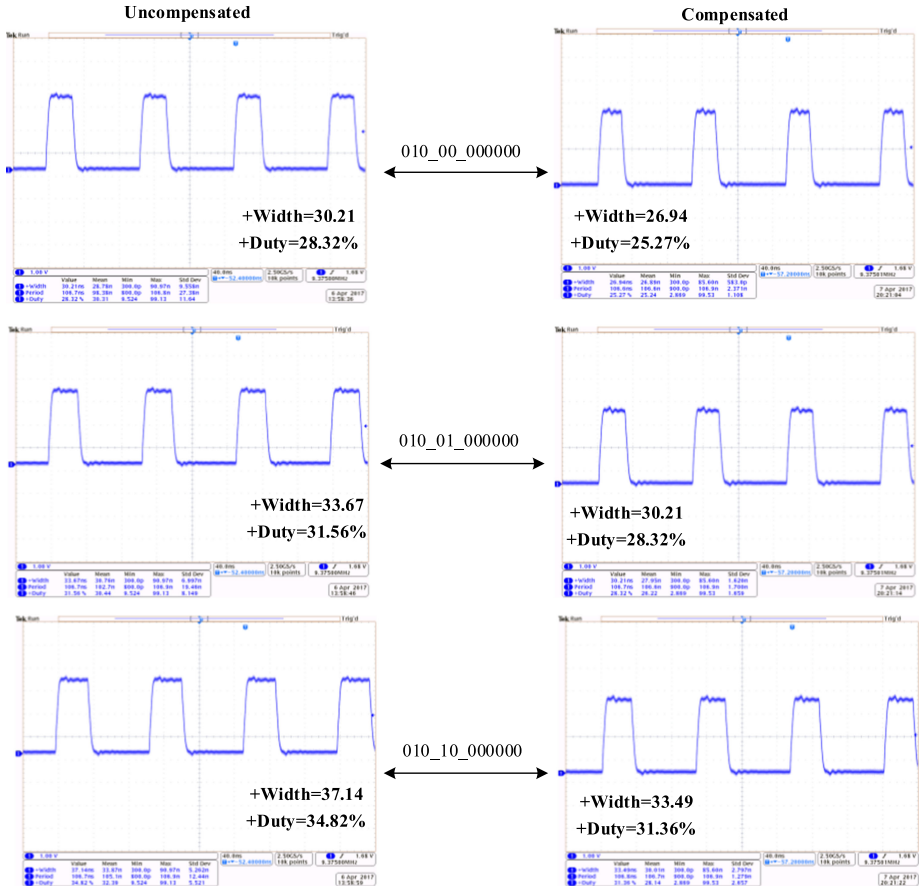


Fig. 13. Measured DPWM output waveforms when the duty varies from 010_00_000000 to 010_10_000000 with compensated and uncompensated critical path delays. Vertical scale, 1 V/div. Horizontal scale, 40 ns/div.

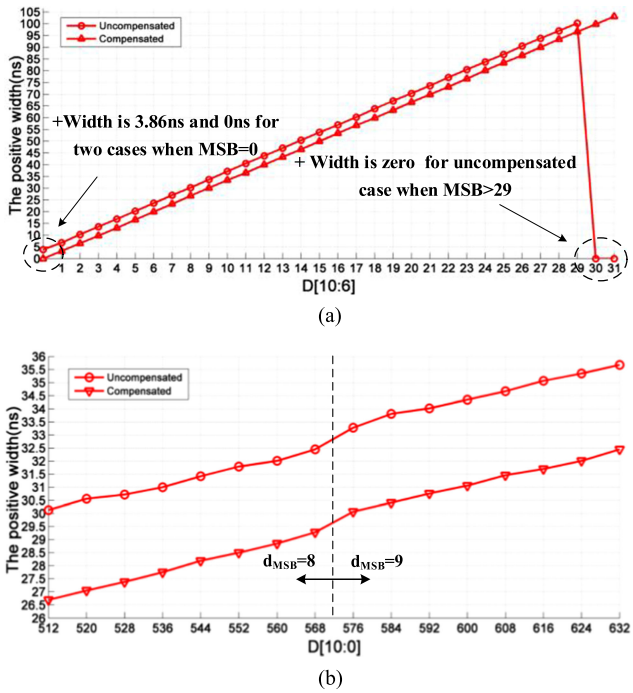


Fig. 14. (a) Positive width as a function of the MSBs. (b) Detailed experimental results. For $D[5:0] > 63$, the LSB rolls over.

TABLE I
EXPERIMENTAL CRITICAL PATH DELAYS FOR DIFFERENT DUTY COMMANDS

Duty Command	Switching Period(ns)	Uncompensated Width(ns)	Compensated Width(ns)	Tcpath (ns)	Duty Difference
010_00_000000	106.7	30.13	26.69	3.44	3.22%
010_00_001000	106.7	30.56	27.05	3.51	3.29%
010_00_010000	106.7	30.72	27.38	3.34	3.13%
010_00_011000	106.7	31.00	27.75	3.25	3.05%
010_00_100000	106.7	31.42	28.19	3.23	3.03%
010_00_101000	106.7	31.79	28.50	3.29	3.09%
010_00_110000	106.7	32.01	28.85	3.16	2.96%
010_00_111000	106.7	32.45	29.28	3.17	2.97%
010_01_000000	106.7	33.28	30.07	3.21	3.01%

first and second coarse delay modules. Because of the proposed optimization techniques for critical path, the calculated R^2 for the linear fit is greater than 0.9999, demonstrating a very high monotonicity. More importantly, the positive widths are 3.86 and 0 ns, respectively, in two cases when $\mathrm{MSB} = 0$, demonstrating that the critical path delays are 3.86 ns and only the critical path delays are superimposed on pulse width, namely they enlarge the positive width. In other MSB cases, the critical path delays are also superimposed on the width causing larger duty cycle. In addition, the duty cycle never exists in uncompen-

TABLE II
BRIEF COMPARISON OF HRDPWM ARCHITECTURES

Reference	Architecture	Operating Frequency	Synchronous	Time Resolution	Critical Path	Compensation for Critical Path
[12]	DLL-based	128 MHz	No	2 ns	4 -to-1 MUX and interconnects	No
[15]	Cascade DCM-based	100 MHz	Yes	625 ps	4-to-1 MUX and interconnects	No
[17]	DCM-based	200 MHz	Yes	625 ps	8-to-1 MUX and	No
	IODELAY1-based	200 MHz	Yes	78 ps	interconnects	No
					interconnects	
[18]	Carry (Altera) chain	50 MHz	Yes	60 ps	Interconnects ANDs	No
	(Xilinx)	178 MHz	Yes	90 ps	and interconnects	No
This paper	PLL and carry chain based	75 MHz	Yes	53 ps	4-to-1 MUX and interconnects	Yes

sated design when $\mathrm{MSB} > 29$. In other words, compared with uncompensated case, this design covers a wide range of duty cycle after compensating the critical path delays.

Fig. 14(b) gives the detailed results that pulse width produced by the modulator as a function of the 11-bit duty cycle commands that are increased from 512 to 632 with step of 8 units. The calculated R^2 is 0.9949, which indicates a good linearity. A linear fit shows a slope of about average 423 ps per 8-LSB duty cycle command unit so that each average adder delay is about 53 ps, which means a high time resolution has been achieved. Notice that the delay gap between the two cases is the critical path delays. Table I shows a part of the detailed data in Fig. 14(b). It can be concluded that the critical path delays are about 3.29 ns, which means the duty cycle is increased by 3.08%, indicating a significant improvement for the accuracy of the DPWM duty cycle after compensation. Small deviations can be acceptable in the boundaries when the LSB rolls over because the frequency of synchronous clock generated by PLL must be modified iteratively to meet (6).

Because certain numbers of bits are assigned to counter and fine delay module, the switching period is fixed. Once the bit numbers are changed, the switching period and duty cycle difference will also change, but critical path delays are equal in any cases. That is, if the design does not compensate the critical path delays, they will always exist in the design and increase the duty cycle. Furthermore, the critical path delays are much larger than a time resolution of 53 ps, which means that fine delay module needs 63-unit adder delay to offset the impact of critical path.

Table II briefly lists the architectures and parameters of some advanced DPWMs based on FPGA. All of previous DPWM architectures have not considered duty-cycle-increment phenomenon caused by the critical path, while this paper solves the problem by the proposed compensation techniques, and maintains a good monotonicity and a high time resolution at the same time.

V. CONCLUSION

This paper presents a new DPWM architecture based on PLL and carry chain in FPGA. It mainly consists of three parts: a counter-based first coarse delay module, a PLL-based second coarse delay module, and a carry-chain-based fine delay module. The paper presents a duty-cycle-increment phenomenon due to the superposition of critical path delays, which is more serious for the DPWM architecture whose switching frequency is about tens of MHz or higher. The proposed DPWM can solve this problem by adjusting the duty cycle with a compensation method based on tool command language commands. In addition, the proposed architecture shows a wide duty range by the compensation techniques, and obtains a high time resolution. All of these advantages were verified through an 11-bit DPWM with 9.375-MHz switching frequency. It achieves a good linearity where R^2 is 0.9949. The duty cycle range is from 1.52% to 97.81%, the time resolution and average duty difference are 53 ps and 3.08%, respectively.

APPENDIX A
VERILOG HDL CODE FOR OPTIMIZED MUX

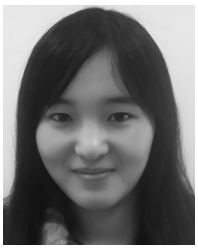
```
module mux_4x1
(
    input[3:0] in,
    input[1:0] sel,
    output  scd_out
);
    wire out_wire0,out_wire1/*synthesis keep*/;
    wire scd_out/*synthesis keep*/;

    altclkctrl m0( .clkselect(sel[0]), .ena(1'b1),
                   .inclk(in[1:0]), .outclk(out_wire0) );
    defparam m0.width_clkselect = 1;
    defparam m0.number_of_clocks = 2;

    altclkctrl m1( .clkselect(sel[0]), .ena(1'b1),
                   .inclk(in[3:2]),  .outclk(out_wire1) );
    defparam m1.width_clkselect = 1;
    defparam m1.number_of_clocks = 2;

    assign scd_out = sel[1] ? out_wire1 :out_wire0;
endmodule
```

REFERENCES

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

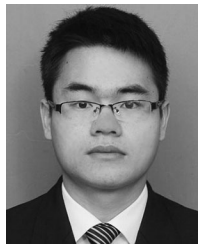
[2] A. V. Peterchev, J. Xiao, and S. R. Sanders, "Architecture and IC implementation of a digital VRM controller," *IEEE Trans. Power Electron.*, vol. 18, no. 1, pp. 356–364, Jan. 2003.

[3] V. Yousefzadeh, N. Wang, Z. Popovic, and D. Maksimovic, "A digitally controlled dc/dc converter for an RF amplifier," *IEEE Trans. Power Electron.*, vol. 21, no. 1, pp. 164–172, Jan. 2006.
[4] H. C. Foong, Y. Zheng, Y. K. Tan, and M. T. Tan, "Fast-transient integrated digital DC-DC converter with predictive and feedforward control," *IEEE Trans. Circuits Syst. I, Reg. Paper*, vol. 59, no. 7, pp. 1567–1576, Jul. 2012.
[5] M. P. Chan and P. K. T. Mok, "On-chip digital inductor current sensor for monolithic digitally controlled DC-DC converter," *IEEE Trans. Circuits Syst. I, Reg. Paper*, vol. 60, no. 5, pp. 1232–1240, May 2013.
[6] A. V. Peterchev and S. R. Sanders, "Quantization resolution and limit cycling in digitally controlled PWM converters," *IEEE Trans. Power Electron.*, vol. 18, no. 1, pp. 301–308, Jan. 2003.
[7] H. Peng, A. Prodic, E. Alarcon, and D. Maksimovic, "Modeling of quantization effects in digitally controlled dc-dc converters," *IEEE Trans. Power Electron.*, vol. 22, no. 1, pp. 208–215, Jan. 2007.
[8] Z. Lukic, C. Blake, S. C. Huerta, and A. Prodic, "Universal and fault-tolerant multiphase digital PWM controller IC for high-frequency dc-dc converters," in *Proc. Int. Conf. IEEE Appl. Power Electron.*, Feb. 2007, pp. 42–47.
[9] Z. Sun, K. W. R. Chew, H. Tang, and L. Siek, "A 0.42-V input boost dc-dc converter with pseudo-digital pulsewidth modulation," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 61, no. 8, pp. 634–638, Aug. 2014.
[10] C. Huang and P. K. T. Mok, "A delay-line-based voltage-to-duty-cycle controller for high-frequency PWM switching converters," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 62, no. 8, pp. 751–755, Aug. 2015.
[11] K. Wu, B. Hwang, and C. C. Chen, "Synchronous double-pumping technique for integrated current-mode PWM DC-DC converters demand on fast-transient response," *IEEE Trans. Power Electron.*, vol. 32, no. 1, pp. 849–865, Jan. 2017.
[12] S. C. Huerta, A. de Castro, O. Garcia, and J. A. Cobos, "FPGA-based digital pulse-width modulator with time resolution under 2 ns," *IEEE Trans. Power Electron.*, vol. 23, no. 6, pp. 3135–3141, Nov. 2008.
[13] M. G. Batarseh, W. Al-hoor, L. Huang, C. Iannello, and I. Batarseh, "Segmented digital clock manager FPGA based digital pulse width modulator technique," in *Proc. Int. Conf. Power Electron. Spec.*, 2008, pp. 3036–3042.
[14] A. de Castro and E. Todorovich, "High resolution FPGA DPWM based on variable clock phase shifting," *IEEE Trans. Power Electron.*, vol. 25, no. 5, pp. 1115–1119, May 2010.
[15] M. G. Batarseh, W. Al-Hoor, L. Huang, C. Iannello, and I. Batarseh, "Window-masked segmented digital clock manager-FPGA-based digital pulse-width modulator technique," *IEEE Trans. Power Electron.*, vol. 24, no. 11, pp. 2649–2660, Nov. 2009.
[16] M. Scharrer, M. Halton, T. Scanlan, and K. Rinne, "FPGA-based multiphase digital pulse width modulator with dual-edge modulation," in *Proc. IEEE Appl. Power Electron. Conf. Expo.*, 2010, pp. 1075–1080.
[17] D. Navarro, O. Lucia, L. Barragan, J. I. Artigas, I. Urriza, and O. Jimenez, "Synchronous FPGA-based high resolution implementations of digital pulse width modulators," *IEEE Trans. Power Electron.*, vol. 27, no. 5, pp. 2515–2525, May 2012.
[18] D. Costinett, M. Rodriguez, and D. Maksimovic, "Simple digital pulse width modulator under 100ps resolution using general-purpose FPGAs," *IEEE Trans. Power Electron.*, vol. 28, no. 10, pp. 4466–4472, Oct. 2013.
[19] *Cyclone IV Device Handbook*, ch. 2, Altera, San Jose, CA, USA, Feb. 2013, vol. 1.
[20] *Advanced Synthesis Cookbook*, ch. 16, Altera, San Jose, CA, USA, Jul. 2011, vol. 11.0.
[21] *Clock Control Block (ALTCLKCTRL) Megafunction User Guide, UG(v. 10)*. Altera, San Jose, CA, USA, Sep. 2010.
[22] *Verilog Register Transfer Level Synthesis, IEEE Standard 1364.1(E)*, 2002.



Xin Cheng received the B.S. and M.S. degrees in microelectronics from Hefei University of Technology, Hefei, China, in 2006 and 2009 respectively, and the Ph.D. degree in microelectronics from the Institute of Electronics, Chinese Academy of Sciences, Beijing, China, in 2012.

She currently serves as an Associate Professor in Hefei University of Technology. Her research interests include dc–dc converter and analog integrated circuit design.



Ruifeng Song received the B.S. degree in microelectronics from Hefei University of Technology, Hefei, China, in 2015. He is currently working toward the M.S. degree in microelectronics at Hefei University of Technology.

His research interests include digital control switching mode power supplies, field programmable gate arrays, and digital circuit design.

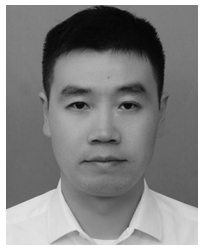
Guangjun Xie received the B.S. and M.S. degrees in microelectronics from Hefei University of Technology, Hefei, China, in 1992 and 1995, respectively, and the Ph.D. degree in information processing from the University of Science and Technology of China, Hefei, in 2002.

He was a Visiting Scholar in IMEC, Katholieke Universiteit Leuven and Fudan University. In 2003, he became a Professor in Hefei University of Technology. He has authored about 120 papers and holds seven Chinese patents. His current research interests include IC design and nanoelectronics.

Dr. Xie is a senior member of the Chinese Institute of Electronics.

Yu Zhang received the B.E. degree in electronics science and technology from Jiangxi University of Science and Technology, Jiangxi, China, in 2015. She is currently working toward the M.E. degree in integrated circuits and systems in Hefei University of Technology, Hefei, China.

Her research interest includes digital dc–dc converter design.



Zhang Zhang received the B.S. degree in electronic science and technology from Hefei University of Technology, Hefei, China, in 2004, and the Ph.D. degree in microelectronics from Fudan University, Shanghai, China, in 2010.

Since 2011, he has been working in the School of Electronic Science and Applied Physics, Hefei University of Technology, and he is currently an Associate Professor. His current research interests include mixed signal circuits and ultra-low power biomedical circuits.