ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 3, March 2015

An Efficient NAND Gate Based Glitch-free All-Digital Duty-Cycle Corrector Architecture

S.Theivanayaki[1], M.Sathiskumar[2]

PG Scholar, VLSI Design, P.A. College of Engineering and Technology, Pollachi, India[1]

Head of the Department, PG-ES, P.A. College of Engineering and Technology, Pollachi, India[2]

Abstract: Duty-cycle correctors are widely used in high-speed devices and system-on-a-chip applications. Because both the positive and negative edges of the clock are utilized for sampling the input data, these systems require an exact 50% duty cycle of the input clock. However, as the clock signal is distributed over the entire chip with clock buffers, the duty cycle of the clock is often far from 50%. Because of the unbalanced rise and fall times, as a result of variations in process, voltage, and temperature. To resolve this problem and to correct the duty-cycle error many approaches are presented. In this work, wide-range NAND gate based glitch-free all-digital duty-cycle corrector is presented. The proposed NAND gate based glitch-free ADDCC not only achieves the desired output/input phase alignment, but also maintains the output duty cycle at 50% with a short locking time. In addition, the proposed method can mitigate the delay mismatch problem and produce minimum delay without any glitch. The circuit complexity also is reduced in the proposed NAND gate based ADDCC. The simulation is carried out in 0.65 nanometer CMOS process. An experimental result shows that the power consumed, delay and area is reduced in the proposed architecture. Simulation results are obtained using MODELSIM 6.3f. The power, area and delay are obtained using the XILINX ISE 8.1 software.

Keywords: All-Digital Duty-cycle corrector (ADDCC), half-cycle delay line, High resolution, Phase alignment.

I. INTRODUCTION

Delay-locked loops have been widely used to minimize timing skews and jitters of the clock signals. The traditional analog DLLs generally have better jitter and skew performances, but they need accurate devices. It makes the analog DLLs sensitive to the process, voltage and temperature (PVT) variations. The PWCLs [2] are realized by the analog approach. The analog approach suffers from several shortcomings. First, more design effort is needed. For example, several design iterations are usually taken to determine the values of the loop gain and the loop filter. Second, the circuit may have the problem of loop instability. An inappropriate selection of the loop gain and the loop filter makes the loop difficult to converge. Third, the circuit usually takes a long transient time to reach the desired duty cycle. Fourth, the analog circuit is not easy to apply voltage scaling for power saving. And last, the duty cycle is fixed at the design time, and cannot be changed during operation. After the analog approach a digital approach was used for designing a 50% duty-cycle corrector.

In all-digital pulse width control loop because of the unbalanced rise and fall times of the clock buffers, as a result of variations in process, voltage, and temperature. To resolve this problem many approaches to correct the duty-cycle error and meet system requirements are presented. For example, an analog pulse-width control loop (PWCL) [1], an all-digital PWCL [6], and an all-digital duty-cycle corrector (ADDCC) [9]. In addition, in some applications, the DCC is combined with delay-locked loop (DLL) located on the output side [11] to eliminate the phase error caused by the DCC circuit. A conventional analog PWCL [1] uses a feedback approach to adjust the duty-cycle of the input clock. Based on the architectural requirements, it requires a ring oscillator to provide a 50% duty-cycle reference clock, and thus, the operating range and the acceptable input duty-cycle error are very restricted in this architecture [2]. A high linearity PWCL that employs a linear control stage and a digitally controlled charge pump is proposed for extending the range of both input and output duty cycles over a wide frequency range [1]. However, an analog PWCL usually requires a large on-chip capacitor that occupies a large chip area. In addition, the analog PWCL has a relatively longer lock-in time, and the leakage current problem of the charge pump makes it unsuitable for use in a nanometer CMOS process.

In contrast to PWCLs, all-digital PWCL and ADDCC do not utilize any passive components and use digital design approaches, making their integration into digital and low-supply voltage systems easy. There are two major types of architecture in the digital approach: synchronous-mirror-delay (SMD) and time-to-digital converter (TDC). The SMD-based DCC that consists of a half-cycle delay line (HCDL) [7] and a match delay line. The HCDL consists of a full-cycle delay line and a mirror delay line. The full-cycle delay line is used for detecting CLK_IN's period information, and the mirror delay line then generates the half-cycle delay time according to CLK_IN's period information. Subsequently, the 50% duty cycle clock is generated by an SR latch. However, the two-delay-line architecture has a delay mismatch problem, particularly in the nanometer CMOS process with on-chip variations (OCVs). In addition, a high-resolution delay line is required for reducing the output duty-cycle error. Hence, the operating frequency range and the final duty-cycle error are limited in this architecture.

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 3, March 2015

The TDC-based DCC quantizes CLK_IN's period information into a digital code, and then this digital code is divided by two to control the delay line for generating the half-cycle delay [14]. The TDC-based DCC has a short locking time. However, the output digital code is divided by two from the quantization digital code, and thus, the duty-cycle correction resolution worsens along with the TDC quantization error. In addition, it is difficult to design a high-resolution TDC while maintaining a wide operation frequency range with low power and a small chip area. Although the TDC resolution has improved in recent years, this architecture is still not suitable for applications with a wide operating-frequency range.

After that, two wide-range ADDCCs with an output clock phase alignment are presented. First, the phase-alignment ADDCC (PA-ADDCC) that consists of a DCC and a DLL not only achieves the desired output/input phase alignment but also maintains the output duty cycle at 50% with a short locking time. But in this delay mismatch problem present and to mitigate this second method (HR-ADDCC) presented. Second, the high-resolution ADDCC (HR-ADDCC) uses a novel correction method without a HCDL to improve the delay resolution and mitigate the delay mismatch problem in a nanometer CMOS process but this method has maximum delay and also glitch effect due to MUX based DCDL block. As compared with the SMD-based and TDC-based ADDCCs, the above two designs can achieve high duty-cycle correction resolution and a wide operating frequency range easily while maintaining the phase alignment.

This paper presents a glitch-free NAND-based DCDL which overcome the limitation by MUX-based DCDLs in a wide range of applications. The proposed NAND-based DCDL maintains the same resolution and minimum delay compared to previously present MUX-based DCDL.

II. Analysis Of Proposed Architecture

In this paper, wide-range NAND gate based ADDCC with an output clock phase alignment is presented with minimum power, area and delay.

A. Block diagram of proposed ADDCC

The block diagram of proposed NAND gate based glitch-free ADDCC is shown in Fig. 1. It is composed of an ADDCC and an all-digital DLL. The ADDCC consists of a DCD, a coarse-tuning digitally controlled duty-cycle correction delay line (coarse DDCC), a fine-tuning digitally controlled duty-cycle correction delay line (fine DDCC), and a DCC controller (DCC_CTRL). The all-digital DLL consists of a PD, a coarse DCDL, a fine DCDL, and a DLL_CTRL. After the system is reset, both the DUTY_SELECT signal and the PHASE_SELECT signal are set to 0. The input clock (CLK_IN) is passed through DCC's delay line and output as an X signal. Subsequently, the inverted X signal is then passed through DLL's delay line and output as a Y signal. The PD of the DLL compares the phase error between the positive edges of X and Y, and then, it outputs DLL_UP/DLL_DOWN control signals to DLL_CTRL. DLL_CTRL adjusts the DCDL_CODE to compensate for the phase error. When the phase error between X and Y is eliminated, the DLL is locked. Then, two clocks (i.e., X and Y) with complementary duty cycles are generated.

After the DLL is locked, the existing all-digital DCC starts to compensate for the duty-cycle error of the output clock (CLK_OUT). The DCD detects the phase error between the negative edges of X and Y, and then, it outputs DCC_UP/DCC_DOWN control signals to DCC_CTRL. DCC_CTRL adjusts the duty-cycle correction DDCC_CODE to increase the pulse width of the X signal according to the outputs of the DCD.

In the first cycle, the DCC extends the pulse width of the X signal. Then, in the next cycle, the positive edge of the Y signal lags behind the positive edge of the X signal because of the pulse extension in the previous cycle. Thus, in the second cycle, DCDL_CODE is decreased for aligning the positive edges of X and Y. The same process is repeated until both the positive edges and the negative edges of X and Y is phase aligned then, the DCC is locked. For loop stability, in each clock cycle, only one loop (DCC or DLL) is working. Therefore, after the DCC is locked, the duty cycle of CLK_OUT is 50%. Further, once the DCC is locked, the PHASE_SELECT signal is set to 1. The inputs of DLL's PD are switched to CLK_IN and CLK_OUT. Then, DLL adjusts DCDL_CODE to compensate for the phase error between CLK_IN and CLK_OUT. Therefore, the output clock (CLK_OUT) can be phase aligned with the input clock (CLK_IN).

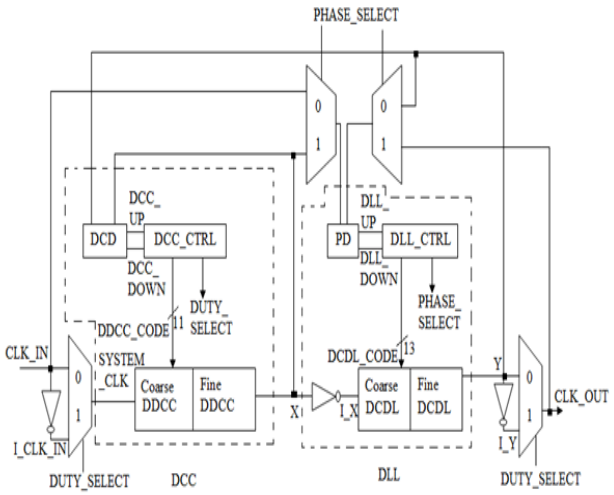

Fig. 1. Block Diagram of Proposed NAND gate based glitch-free ADDCC

After the DLL is locked, the DCC starts to correct the duty cycle of the output clock. Before the DCC is locked, the correction action is the same as in the case when the input clock's duty cycle is less than 50%. Nevertheless, the positive edge of CLK_OUT is not aligned with the positive edge of CLK_IN. Therefore, I_Y signal is selected as the CLK_OUT signal and the positive edge of inverted Y (I_Y) lags to CLK_OUT. Thus, DLL just reduces DCDL_CODE, which in turn reduces the delay time of DCDL until the positive edges between CLK_IN and CLK_OUT are aligned.

DOI 10.17148/IJARCCE.2015.4390

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 3, March 2015

B. Flow chart of proposed ADDCC

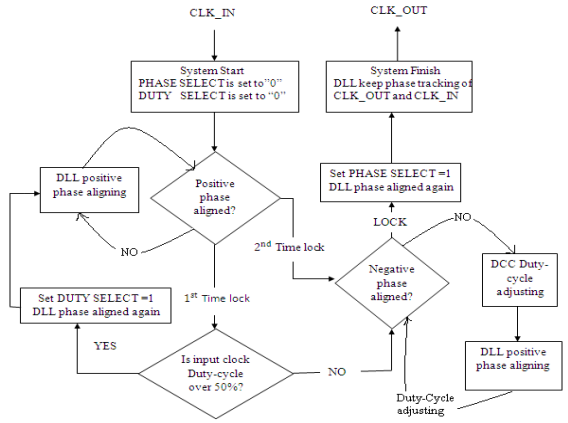

Fig. 2. Operation Flowchart of proposed NAND gate based glitch-free ADDCC

Fig. 2. Shows the operation flowchart of the proposed NAND gate based glitch-free ADDCC. At the beginning, DLL performs a positive-edge phase alignment. When DLL is at the first time lock, DCC determines whether CLK_IN's duty cycle is < 50% or more. If CLK_IN's duty cycle is < 50%, then DCC duty cycle is adjusted. Otherwise, DCC sets DUTY_SELECT to 1, and the DLL phase alignment is carried out until the second time lock. When DLL is locked, we start the DCC duty-cycle adjustment. In the first cycle, DCC extends the pulse width of the X signal. Then, in the next cycle, the positive edge of the Y signal lags behind the positive edge of the X signal because of the pulse extension in the previous cycle. Thus, in the second cycle, DLL aligns the positive edges of X and Y. The same process is repeated until both the positive edge and the negative edge of X and Y is phase aligned, and then, DCC is locked. After DCC is locked, DLL sets PHASE_SELECT to 1 and keeps tracking the phase between the output clock (CLK_OUT) and the input clock (CLK_IN).

C. Circuit Implementation for DLL

The key functional blocks in DLL including a PD and NAND gate based glitch-free digitally controlled delay line (DCDL). The circuit operating path is controlled by two control signals. The DCDL has n + 1 coarse-tuning delay elements which consist of NAND gates as shown in Fig. 3.

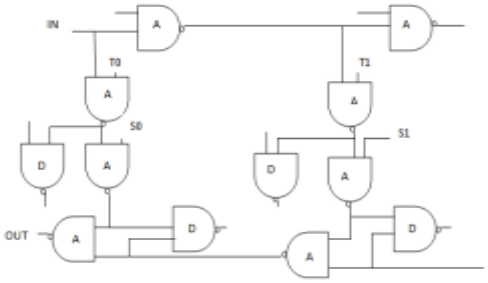

Fig. 3. NAND gate based glitch- free DCDL

The NAND gate based DCDL has control bits to control the delay elements. In this A denotes the fast input of the NAND gate and D denotes the dummy cell for the load balancing. Two control bits Ti and Si are used to synchronize the arrival of the input and the arrival of the control bits. It has three possible states as shown in Table I.

TABLE I
LOGIC STATES OF EACH DE IN PROPOSED DCDL

S_i	T_i	DE STATE
0	1	PASS
1	1	TURN
1	0	POST-TURN

Figure 4 shows the fine-tuning component of DCDL, which is based on digitally controlled varactors (DCVs). Each DCV cell is composed of four NAND gates. The fine DCDL can provide m + 1 different delay times by controlling the DCV cells. When the control bit of a DCV cell is enabled, the capacitance at the output node of the inverter is changed and the delay time is increased accordingly. Thus, the resolution of DCDL can be improved by the use of fine DCDL.

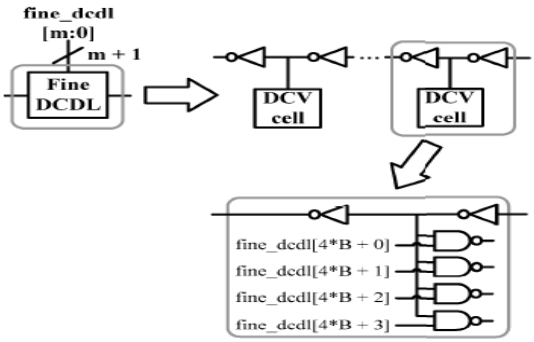

Fig. 4. Fine-Tuning Component of Proposed DCDL

Figure 5 and Figure 6 shows the proposed PD. It detects the positive-edge phase error between COMP and BASE. The proposed PD is composed of a sampled-based PD (SBPD) and a sense- amplifier-based PD. To improve the detectable phase error, a sense-amplifier-based PD that can detect a small phase error that is larger than1ps.

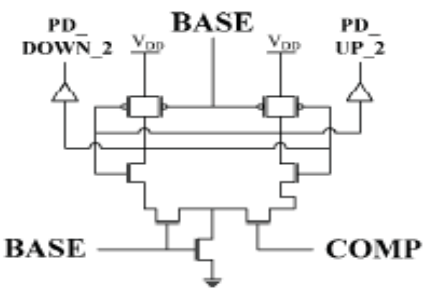

Fig. 5. Sense-Amplifier-Based Proposed Phase Detector

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 3, March 2015

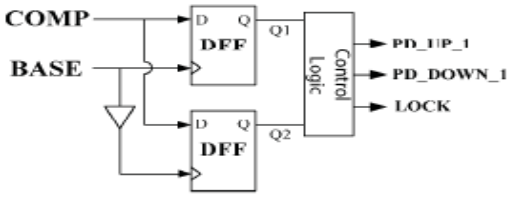

Fig. 6. Sample-Based Proposed Phase Detector

Although the sense-amplifier-based PD can detect a small phase error, it has incorrect detection results when the phase error between two inputs is large because of the leakage current of the transistor in the 65-nm CMOS process. For this reason, the SBPD is used to detect the large phase error at the beginning and then use the sense-amplifier-based PD to improve the overall phase error detection capability. Although the SBPD does not have a small dead zone because of the setup/hold time requirements of the D-Flip/Flops, the SBPD can be designed easily and can be built with standard cells.

It can prevent the incorrect operation of the sense-amplifier-based PD. At the beginning, the PD controller receives SBPD's outputs (PD_UP_1 and PD_DOWN_1). After the SBPD is locked, the PD controller switches to receive sense-amplifier-based PD's outputs (PD_UP_2 and PD_DOWN_2). Therefore, the proposed PD can correctly detect a small phase error between COMP and BASE.

D. Circuit Implementation for DCC

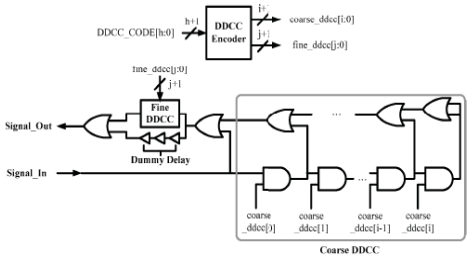

Fig. 7. AND-OR-Type Coarse-Tuning Component of Proposed DDCC

Figure 7 shows the AND–OR-type DDCC architecture, which is combined with the coarse-tuning component (coarse DDCC) and the fine-tuning component (fine DDCC). The circuit operating path is from Signal_In to Signal_Out, which is selected by control code coarse_ddcc [i: 0]. The Coarse DDCC has i + 1 coarse-tuning delay cells, and each coarse-tuning delay cell is combined with an AND cell and an OR cell. Therefore, coarse DDCC can provide i +1 type of pulse-width adjustments and easily cover the wide pulse-width adjustment range. Nevertheless, the resolution of coarse DDCC is not sufficiently good. For this reason, the fine-tuning component is added to increase the DDCC resolution. The fine DDCC fine-tuning component is based on the architecture of DCV, which is the same as that of fine DCDL.

Each DCV cell is combined with four NAND gates, which are controlled by enable code fine_ddcc [j: 0]. the fine-tuning component has j+1 types of delay times brought about by controlling the DCV cells. An OR gate is connected after the DCV output and the dummy delay output. The dummy delay is used for reducing DCV's intrinsic delay.

The architecture of the proposed DCD is similar to that of the proposed PD, as shown in Fig. 5. and Fig. 6. It also consists of a sample-based DCD and a sense-amplifier-based DCD. However, the proposed DCD detects the negative-edge phase error between COMP and BASE. Thus, in the proposed DCD, there are two inverters in front of PD's inputs (COMP and BASE). Then, PD can easily transform into the proposed DCD, and the operation behavior is same as that of the proposed PD.

III. EXPERIMENTAL RESULTS

The simulation results obtained from Modelsim SE 6.3f tool for two architectures are presented. Xilinx 8.1i version is used to measure the Area, power and time delay.

A. *Timing Diagram*

Figure 8 shows the waveform obtained on simulating the proposed NAND based ADDCC. The range for input duty-cycle is 30 and in between 20 to 80 respectively. The positive and negative edges of both the CLK_IN and CLK_OUT in the timing diagram are perfectly matched.

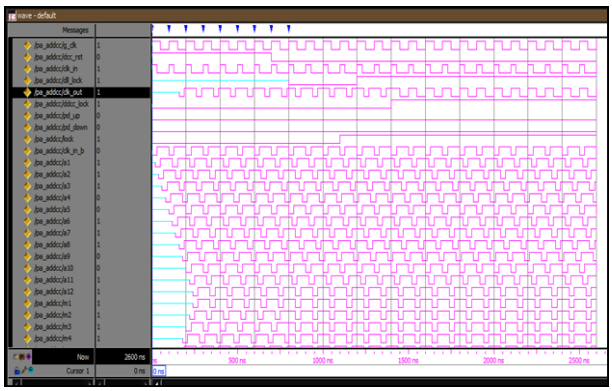

Fig .8. Timing Diagram of PA-ADDCC (existing method)

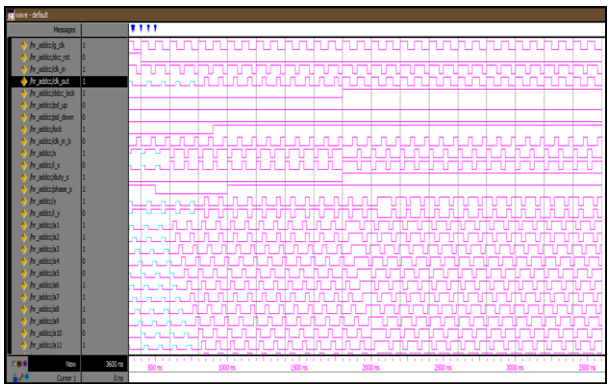

Fig .9. Timing Diagram of HR-ADDCC (existing method)

DOI 10.17148/IJARCCE.2015.4390

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 3, March 2015

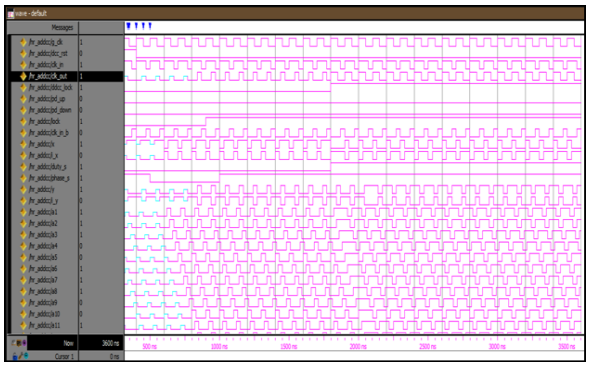

Fig .8. Timing Diagram of Proposed NAND based glitch-free ADDCC

TABLE II
POWER COMPARISION OF EXISTING AND PROPOSED ARCHITECTURES

BLOCK	POWER(mW)
DIGITAL DLL **[GARLEPP B.W. AND DONNELLY K.S. ET AL (1999)]**	340
FAST LOCKING PWCL **[HAN S.R. AND LIU S.I. ET AL (2004)]**	270
ANALOG DLL **[KIM J. AND CHAU P.S. ET AL (1999)]**	175
PA-ADDCC	168
HR-ADDCC	145
PROPOSED NAND GATE BASED ADDCC	71

TABLE III
POWER, AREA AND DELAY COMPARISION OF EXISTING AND PROPOSED ARCHITECTURES

BLOCK	POWER (mW)	AREA(GATE COUNT)	DELAY (ns)
PA-ADDCC	168	479	9.978
HR-ADDCC	145	465	9.925
Proposed NAND gate based ADDCC	71	375	5.747

Table III. Shows power, area and delay comparison of existing and proposed architectures. The proposed NAND gate based ADDCC is 51% power efficient than HR-ADDCC. The proposed NAND gate based ADDCC is power and area efficient architecture compared to all existing methods. Also the proposed architecture produces lower delay than the existing methods.

IV. CONCLUSION

The existing PA-ADDCC not only exhibited the phase alignment of the input and output clocks and also corrected the duty cycle of the output clock to 50% with a short locking time but delay mismatch problem present in it. The existing HR-ADDCC uses a novel correction method without a half-cycle delay line, which can solve the delay mismatch problem but in this delay is high. So, the proposed NAND gate based ADDCC has minimum delay and glitch-free compared to all of the existing architectures. Thus, it is very suitable for duty-cycle correction applications such as the DDR memory, I/O bus interface, and SOC applications. The proposed NAND gate based ADDCC architecture eliminates the redundant power dissipation with lower area. A comparison of the existing architectures with the proposed architecture showed that it exhibits lower power dissipation. The simulation results proved that the proposed architecture is well suited for modern high performance designs where area and power dissipation are major concern.

REFERENCES

[1] J. H. Bae and J. H. Seo, "An All-digital 90-degree phase-shift DLL with loop-embedded DCC for 1.6Gbps DDR Interface", in Proc. IEEE Custom Integrated Circuits Conf., vol. 25, pp. 373–376, Sep. 2007.
[2] F. M. Gardner, "Charge-pump phase-lock loops", IEEE Trans. on solid-circuits vol. 28, pp. 1849– 1858, Nov. 1980.
[3] B. W. Garlepp and K. S. Donnelly, "A portable digital DLL for high-speed CMOS interface circuits", IEEE Journal on Solid-State Circuits, vol. 34, No.5, pp. 632–644, May. 1999.
[4] J. Gu and J. W, "All-digital wide range precharge logic 50% duty cycle corrector," IEEE Trans. on VLSI Systems, vol. 20, No.4, pp. 760–764, April. 2012.
[5] S. R. Han and S. I. Liu , "A 500-MHz-1.25-GHz fast-locking pulse-width control loop with presettable duty cycle", IEEE Journal on Solid-State Circuits, vol. 39, No.3, pp. 463–468, March. 2004.
[6] S. R. Han and S. I. Liu, "A single-path pulse-width control loop with a built-in delay-locked loop", IEEE Journal on Solid-State Circuits, vol. 40, No.5, pp. 1130–1135, May. 2005.
[7] S. K. Kao and S. I. Liu, "All-digital fast-locked synchronous duty-cycle corrector", IEEE Trans. on Circuits Syst. II, vol. 53, No.12, pp. 1363–1367, Dec. 2006.
[8] J. W. Ke and S. Y. Huang , "A high-resolution all-digital duty-cycle corrector with a new pulse-width detector", in Proc. IEEE Electron Devices and Solid-State Circuits Conf., vol. 52 , pp. 1–4, Dec. 2010.
[9] B. G. Kim and K. I. Oh, "A 500MHz DLL with second order duty cycle corrector for low jitter", in Proc. IEEE Custom Integration Circuits Conf., vol. 40, pp. 325–328, Jan. 2006.
[10] F. Mu and C. Svensson (2000), "Pulse-width control loop in high-speed CMOS clock buffers", IEEE Journal on Solid-State Circuits, vol. 35, No.2, pp. 134–141, Feb. 2000.
[11] D. Sheng and C. C. Chung ,"An ultra-low-power and portable digitally controlled oscillator for SOC applications", IEEE Trans. on Circuits Syst. II, vol. 54, No.11, pp. 954–958, Nov. 2007.
[12] Y. M. Wang and J. S. Wang , "An all-digital 50% duty-cycle corrector", in Proc. IEEE Int. Symp., vol. 53, pp. 925–928, May. 2004.
[13] Y. J. Wang and S. K. Kao, "All-digital delay-locked loop/pulse-width-control loop with adjustable duty cycles", IEEE Journal on Solid-State Circuits, vol. 41, No.6, pp. 1262–1274, June. 2006.
[14] P. H. Yang and J. S. Wang , "Low-voltage pulse-width control loops for SOC applications", IEEE Journal on Solid-State Circuits, vol. 37, No.10, pp. 1348–1351, Oct. 2002.
[15] Young-Chan Jang , "A Digital CMOS PWCL with Fixed-Delay Rising Edge and Digital Stability Control", IEEE Trans. on Circuits and Syst. II, vol. 53, pp.1063 – 1067, Oct. 2006.