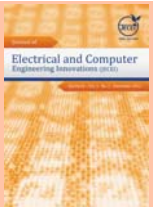
SRTTU

Journal of Electrical and Computer Engineering Innovations

JECEI, Vol. 1, No. 1, 2013

Regular Paper



High Speed Delay-Locked Loop for Multiple Clock Phase Generation

A. Ghanbari[1,*], A. Sadr[2], and M. Nikoo[3]

[1]Graduate School of Electrical Engineering, Qazvin Islamic Azad University (QIAU), Qazvin, Iran

[2]Department of Electrical Engineering, Iran University of Science and Technology (IUST), Tehran, Iran

[3]Department of Computer and Electrical Engineering, Islamic Azad University (MIAU)

[*]Corresponding Author: ghanbari.ashkan@yahoo.com

ARTICLE INFO

ARTICLE HISTORY:
Received 15 September 2012
Revised 15 August 2013
Accepted 22 August 2013

KEYWORDS:
Delay locked loop (DLL)
Phase detector (PD)
Charge pump (CP)
Filter
CMOS

ABSTRACT

In this paper, a high speed delay-locked loop (DLL) architecture is presented which can be employed in high frequency applications. In order to design the new architecture, a new mixed structure is presented for phase detector (PD) and charge pump (CP) which can be triggered by double edges of the input signals. In addition, the blind zone is removed due to the elimination of reset signal. Therefore, operating frequency of the whole system is improved which can be mentioned as notable advantage of the proposed DLL. To obtain more accurate phases at the output signal, a new delay cell is introduced which is controlled by a single voltage. This control voltage, through equalizing the rise and fall time, regulate duty cycle of output clock. These features along with simplicity and low power consumption qualify the proposed architecture to be widely used in high speed systems. For better realization of the designed circuit's behavior, simulation results are presented based on TSMC 0.35μm CMOS technology and 3.3-V power supply for a type II filter which demonstrate accuracy and perfect performance of this work.

1. Introduction

Along with the increment of clock frequency in synchronous VLSI circuits, necessity of correctly aligned system clocks becomes evident. The most important concern is to suppress the skew and jitter of the clock signal. As the clock period is reduced, if jitter and skew remain the same, total clock phase error will be increased, which can affect many aspects of a synchronous system, including setup and hold times, data access times and accuracy of internal control signals. Many applications require accurate placement of phase of a clock or data signal. Although simple delaying the signal could shift the phase, this phase shift is not robust to variations in processing, voltage or temperature. For more precise control, designers incorporate phase shift into a feedback loop that locks the output phase with an input reference signal which indicates the desired phase shift. In essence, the loop is identical to a phase-locked loop (PLL), except that phase is the only state variable and that a variable-delay line replaces the oscillator. Such a loop is commonly referred to as a delay-line phase-locked loop or delay-locked loop (DLL) [1]. One major difference between a PLL and DLL is that, rather than a voltage-controlled oscillator (VCO), a voltage-controlled delay-line (VCDL) is used in a DLL structure.

If output of the delay cell is fed back to the input (forming an oscillator), the DLL could become a PLL. Another significant difference is that the DLL has a first-order response whereas PLL has a second-order response which is the main motivation for using DLLs in clock synchronization circuits. Therefore, design of the control loop is simplified by having only phase as the state variable. DLLs are widely employed in microprocessors, memory and communication ICs in order to reduce chip clock buffering delays. They can be also used to generate multiple clock signals on chip for different applications. The essential function of a DLL is to achieve phase alignment between the input clock and output clock from the final stage of the VCDL. After phase alignment is achieved, each VCDL delay stage is able to provide a stable clock signal which is phase shifted from the input clock. However, the rising clock speed and integration levels of digital circuits have made the phase alignment task increasingly difficult [2]. Although aperiodic signals can be also delayed by the delay line in a DLL, the inputs of delay lines are typically clock signals. Therefore, using a periodic signal, delay lines do not need arbitrarily long delays and typically they only need to span the period of the clock in order to generate all possible phases. A data signal can be delayed by sampling the data by the appropriately delayed clock. By assuming the phase as a variable state and considering delay cell as the final component before the output node, the main building blocks of a DLL system are phase detector (PD) and VCDL. Therefore, careful design of these components will directly affect performance of the DLL. So, many works are reported in the literature on realizing the circuit level implementation of DLL architecture in different forms. In this work, the main emphasis was on rebuilding basic components of a DLL to achieve higher operating frequencies along with correct phase alignment in output signals. In Section II, a brief review of a DLL system and its design considerations are presented. In Section III, the newly designed circuits for PD, charge pump (CP) and VCDL are discussed. Simulation results are shown in Section IV to prove theoretical assumptions and the conclusions are made in Section V.

2. DLL DESIGN CONSIDERATIONS

A DLL is essentially a nonlinear negative feedback system. However, it can be easily characterized by linear analysis. Although linear analysis is not able to produce a very accurate result, it can still serve as a reasonable first-order approximation and lead to some useful insights into a DLL's operation. Basic loop's building blocks are similar to those of a PLL'PD, which is often combined with a CP, Loop Filter (LP) and a variable-delay line, especially VCDL. Since phase is the only state variable, a control loop higher than first-order is not needed to compensate for a fixed phase error. The resulting transient impulse response is a simple exponential one. Despite simple loop characteristics are the advantage of DLLs over PLLs, the design is complicated by additional circuitry that is needed to overcome having a limited delay range and not producing its own frequency [1]. In a DLL, the input clock signal travels through the VCDL and develops phase shift in every delay stage of the VCDL which is controlled by voltage of a loop filter. Phase of the output signal which is taken from one of the delay stages is compared with phase of the input clock in the PD. Then, the generated phase error in PD is transferred to the CP. The CP uses phase error information to adjust voltage of the loop filter and thus change delay of the VCDL. Due to the negative feedback mechanism, phase error is gradually reduced until finally becomes zero. At the same time, delay of the whole VCDL line becomes equal to one clock period and voltage of the loop filter is stabilized, which indicates that a locked state has been established. The DLL's loop dynamics were completely discussed in [3, 4]. Two common architectures of DLLs are shown in Fig. 1. In the first type, the input reference clock C_{Kref} is compared with its delayed version [4]. This DLL structure is often used for frequency synthesis, clock generation and signal synchronization. In the second type, the reference signal C_{Sref} is compared with a delayed version of another uncorrelated signal C_{Sref} [4]. This type of DLL is often used in some clock recovery circuits. Since the objective of this paper was generating different phases of a signal for clock synchronization, therefore, configuration of Fig. 1(a) was employed. By ignoring the sampling nature of the PD, the DLL's loop dynamics could be analyzed by continuous time (s-domain) approximation. It must be considered that the DLL's lower limit of the bandwidth is about a decade or more below the operating frequency [3]. In Fig. 2, the DLL block diagram is illustrated in terms of the input and output phases.

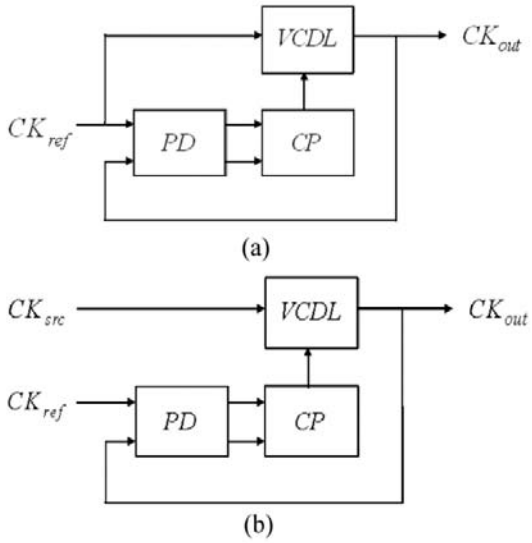


Figure1: Common architectures of DLL (a) For signal synchronization purposes (b) For clock recovery purposes

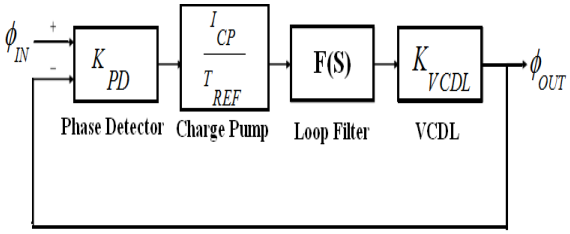


Figure2: S-domain model of a DLL in terms of phase

In the structure of Fig.2, K_{PD} , I_{CP} and K_{VCDL} are phase detector gain, charge pump current and VCDL gain, respectively. The gain of VCDL is proportional to the number of the delay cells. The loop filter transfer function is represented by F(s) and period of the input reference clock is defined by T_{REF}. The input and output phases are denoted by φ_{in} and φ_{out}, respectively. Many applications have used a single capacitor C_f as the loop filter. Therefore, the transfer function F(s) can be written as:

$$F(s) = \frac{1}{SC_f} \tag{1}$$

Using S-domain model in Fig. 2 and Eq. (1), the close-loop transfer function of a DLL will be:

$$H(s) = \frac{\phi_{out}}{\phi_{in}} = \frac{1}{1 + \frac{S}{\omega_n}} \tag{2}$$

In which ω_n is close-loop bandwidth (3-dB frequency). By defining ω_{in} as frequency of the input signal,ω_n is equal to:

$$\omega_n = \frac{K_{PD}.I_{CP}.K_{VCDL}}{2\pi C_f}\omega_{in} \tag{3}$$

Eq. (2) shows that the DLL has a first-order transfer function which is illustrated in Fig. 3. The type II filter adds a zero and a pole to the transfer function of the DLL.

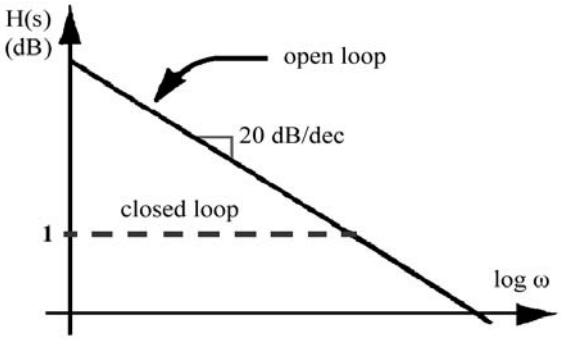


Figure3: First-order transfer function of a DLL

On the stability analysis, a PLL contains at least two states to store both frequency and phase information. Therefore, to maintain loop stability, an additional zero is needed. But in a DLL, loop gain directly determines the desired bandwidth. The only stability consideration is when the loop bandwidth is very close to the reference frequency [1]. Periodic sampling nature of the phase detection and delay in the feedback loop degrade the phase margin.

On the design of the delay line, the following notations must be considered:

I. If the desired phase of the output signal is beyond the delay range of delay line, the loop will not lock properly.
II. Due to the delay-line input propagates to the DLL output, tracking jitter and the output's duty cycle depends not only on delay-line design but also on delay-line input.
III. The basic DLL cannot generate new frequencies different from those of the delay-line input.

3. The Proposed Circuit

A. PD

The PD is one of the most critical components within a DLL which generates an output signal proportional to the phase difference of its two inputs. In modern PLL/DLL design, it is followed by a difference detector

(DD) or a CP circuit. However, during recent years, CP-based architectures have been widely used to design high speed microprocessors or communication systems for their low-complexity zero-steady-state phase error and infinite pulling range [5]. Therefore, in a PLL/DLL-based system design, the PD and CP are mostly considered the complementary part of each other.

Basic implementation of the PD is an XOR gate [6], in which DC value of the output signal is linearly proportional to the phase difference of the two input signals. But, it has its own drawbacks. Since there is only one output from the XOR PD, it is difficult to be interfaced with the subsequent CP circuit. In addition, duty cycle dependence problem with the XOR gate makes this circuit unavailable in modern design because signal edge detection has more conformity with duty cycle of the input signals. If the flip-flop-based PD is employed, it can detect both phase and frequency difference, which is known as Phase Frequency Detector (PFD) and increases acquisition range and locking speed [7]. To design a CP-based DLL system, a tri-state PFD must be employed which is usually built using memory elements. Therefore, a reset signal is needed to clear memory elements [5]. The designed circuit for this PFD is illustrated in Fig. 4. One of the most important issues in the flip-flop-based PFDs is the existence of dead zone. The minimum pulse width of the PFD output which is needed to completely turn on the charge pump is the main reason of generating the dead zone.

For very small phase errors, output voltage of the PFD in Fig. 4 will not be proportional to the phase error of input signals because the PFD internal components' delay dominates phase delay between the inputs [5]. This effect which happens around zero phase error is known as dead zone effect. For tri-state PFDs built by Flip-Flops, when phase difference between the two inputs is close to 2π, rising edge of the leading phase can fall into the reset region. Therefore, during the reset process, the PFD cannot detect the leading signal. It treats the following lagged signal as the leading one and generates reversed phase information. Because of this effect, phase comparison range of the PFD reduces to $2\pi - \Delta$, where Δ is the length of the blind zone normalized to 2π. The third type of PD which is widely used in high speed DLL designs is dynamic PD. Basic structure of a dynamic PD consists of two blocks, which are used to separately generate UP and DOWN signals.

Each block consists of two cascaded stages with a pre-charge PMOS in each stage. Pre-charge activity of the second stage is often controlled by output of the first stage, as completely discussed in [8]. In order to reach higher operating frequencies, the third method is used to design the new PD. But, most of the reported works in this area use one edge triggering. Therefore, dynamic power consumption will be high because the PFD and CP circuits must work until the next rising or falling edge of the input signals. In addition, more clock cycles are needed to equalize phase error between UP and Down signals. To resolve this, feature of double edge triggering is considered which means that, in contrast to other designs, some modifications must be applied to the structure of dynamic PD.

The proposed circuit for this purpose is shown in Fig. 5, in which A and B are input signals. In order to better analyze the proposed PD circuit, it was considered that the input signal in node A was the leading one. Since two cascaded clocked inverters were used on each side, therefore, the leading edge of signal would charge DOWN node to V_{dd} value and discharge UP node to zero value on the other side to continue this process, as demonstrated in Fig. 6(a). If the input signal in node B became the leading signal, the state of UP and DOWN nodes would be complemented, as illustrated in Fig. 6(b). The main advantage of the proposed PD was that the UP and DOWN signals contained constant values, which means that, their average value was equal to V_{dd} or zero.

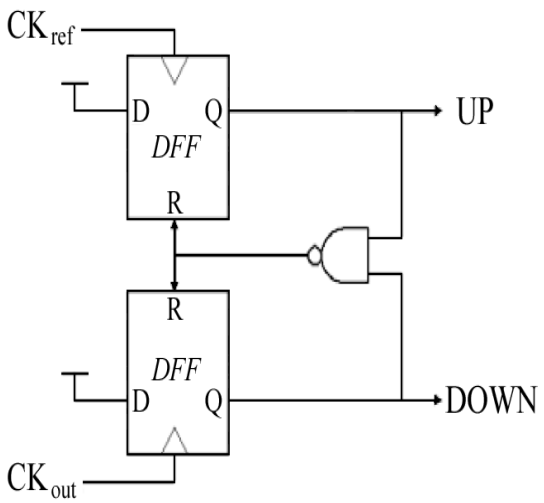


Figure 4: Flip-flop based PFD

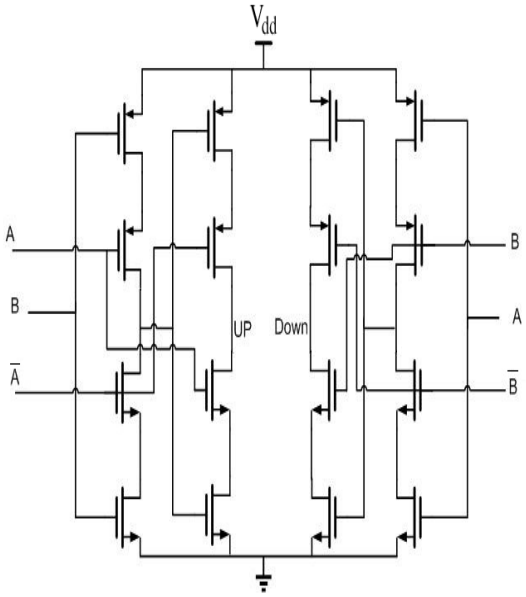


Figure 5: Designed PD circuit

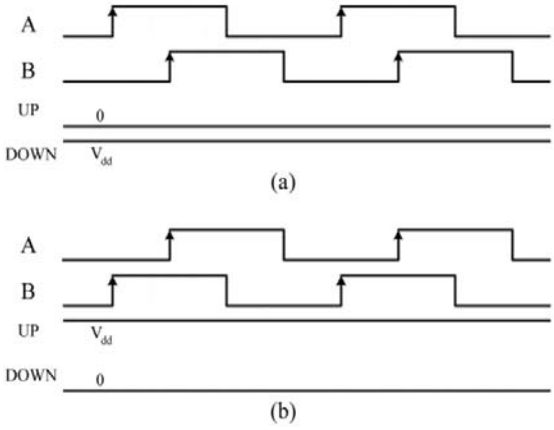


Figure6: Different states of designed PD, (a) Signal A is the leading signal, (b) Signal B is the leading signal

B. CP

Another important issue in the PD/CP-based DLL design is architecture of CP to reach less acquisition time and enhance speed performance of the DLL structure. The CP performs the function of adjusting voltage of the loop filter and thereby altering the VCDL delay according to the phase error information from the PD. Several methods have been suggested for design the CP. In the primary concept, the CP simply consists of two controlled switches, one current source and one current sink, as shown in Fig. 7.

The main drawbacks of this concept are:

I. The UP signal needs to be inverted.

II. Because of the mismatch between current sources, noise of the whole system will be increased.

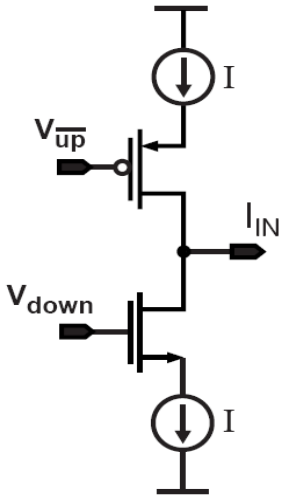


Figure 7: Principle of CP

In order to resolve these problems, some corrections were suggested in [9]. An efficient architecture was proposed in [10,11], in which the charge sharing problem between common drain transistors was resolved. This configuration is illustrated in Fig. 8. The cascade current mirrors are used at the output node to increase the output resistance so that the charging or discharging currents could not be significantly disturbed. But, the mismatch problem is not completely resolved in this configuration. To overcome this issue, differential pair-based CPs showed better performance although static power dissipation increased total power consumption of the DLL. A combination of the above ideas was used to design the new CP, in which the mismatch problem was completely resolved and the static power dissipation was reduced. The proposed CP is shown in Fig. 9. According to the circuit in Fig. 5, if signal A was the leading signal, then the value of UP would be zero and the value of DOWN would be V_{dd}. Therefore, the node O_1 would be charged via M_1 and the node O_2 would be discharged, in which control voltage of the VCDL would vary until the two input signals of PD contained the same phase. The same reason was established for node O_2 when signal B in the circuit of Fig. 5 became the leading one. It had to be mentioned that the output of node O_1 is used as the control voltage for VCDL. But, since the proposed CP had two differential output nodes, it could be flexibly used with active LPs to bias their voltage differentially, which was another notable advantage of the designed CP.

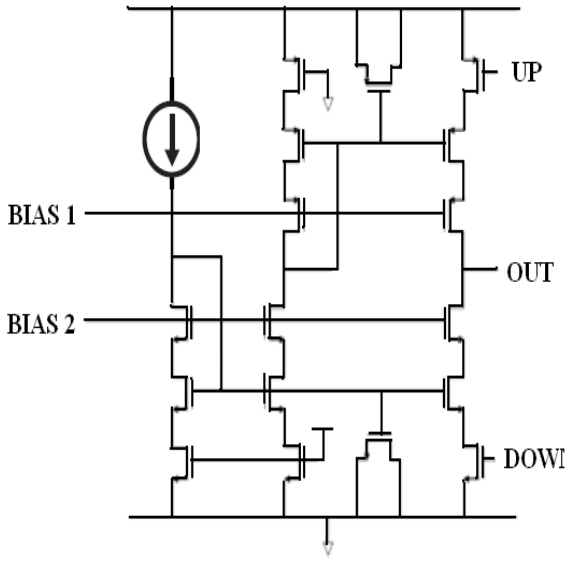


Figure 8: Principle of CP

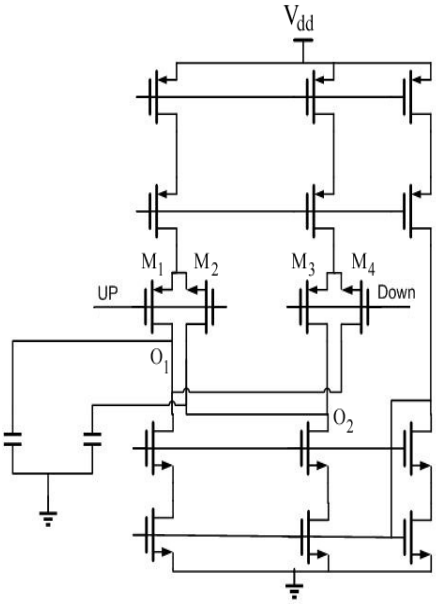


Figure 9: Designed CP circuit

C. VCDL

The VCDL takes two inputs, a control voltage and a clock. The output is a clock of the same frequency as the input; but, phase shifts by some amount proportional to the control voltage. As shown in Fig. 10, a VCDL typically consists of a number of delay stages (cells) which are connected in series. So, it is an open loop configuration and does not oscillate. Thereby, it is different from the voltage-controlled oscillator (VCO) in a PLL. The simplest form of a delay cell is a current driven inverter which is illustrated in Fig. 11. The maximum current supplied to the inverters is controlled by the bias voltage. A higher bias voltage increases the cell's current, thereby reduces the delay. The delay of this circuit is inversely related to the current, which is proportional to square of the control voltage. Therefore, the gain curve versus control voltage will be a non-linear curve which is one of the design challenges on the design of a DLL system. Another drawback of this circuit is inequality of the paths, through which the control voltage passes, to bias the upper and lower current sources. Differential pair-based delay cells are more widely used owing to their inherent advantages such as better immunity to common-mode noise and improved spectral purity. The most popular differential delay stage involves a source-coupled differential pair with two load elements and a biasing tail current source. The load elements are MOS load elements such as diode-connected transistors or triode biased transistors. They are chosen based on design considerations, such as control over the cell delay, output signal swing and dynamic supply noise rejection [12-16]. But, the following drawbacks limit their use in some DLL configurations: Active loads consume voltage headroom; so, the output voltage swing is limited. The output DC voltage is not controlled. Although the non-linear relationship between gain and control voltage of the inverter-based VCDLs is a major problem, since these structures occupy much smaller space on the chip and are much more power efficient in comparison with differential VCDLs, some corrections are made to the structure of Fig. 11 under some design considerations to implement the desired VCDL for the proposed DLL architecture. The designed circuit for delay cell is illustrated in Fig. 12. The proposed delay cell had a clear advantage over the circuit in Fig. 11. Since the control voltage was applied to the transistor which was in series with diode-connected PMOS and NMOS transistors, the bias voltage was applied to the current sources simultaneously. Therefore, the output noise would be considerably reduced. In the absence of input signals, there was no control voltage, which indicated that the whole transistors were off and there was no static power dissipation in the circuit. It must be mentioned that, because of poor power supply rejection, the current starved inverter delay line is not practical for all designs. Since in the proposed design, high speed feature along with low power consumption were the key factors, inverter-based VCDL were chosen as the desired block.

D. LP

To complete design process of the DLL, an appropriate LP must be chosen. Type II filter was chosen to add an extra zero and pole to the closed loop transfer function of the DLL for more control abilities. Then, Eq. (1) would be changed as follows:

$$F(S) = \frac{1 + RC_2S}{(C_1 + C_2)S + RC_1C_2S^2} \quad (4)$$

in which C_2 is in series with R and the set is in parallel with C_1. The transfer function would be:

$$H(s) = \frac{1}{1 + \frac{1}{AF(s)}} \text{ , A} = \frac{K_{PD}.I_{CP}.K_{VCDL}}{T_{REF}} \quad (5)$$

The resulting system would be a type II system.

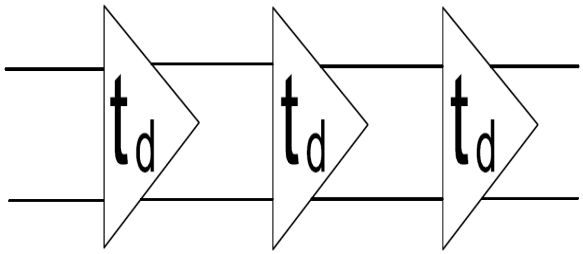


Figure 10: Typical VCDL

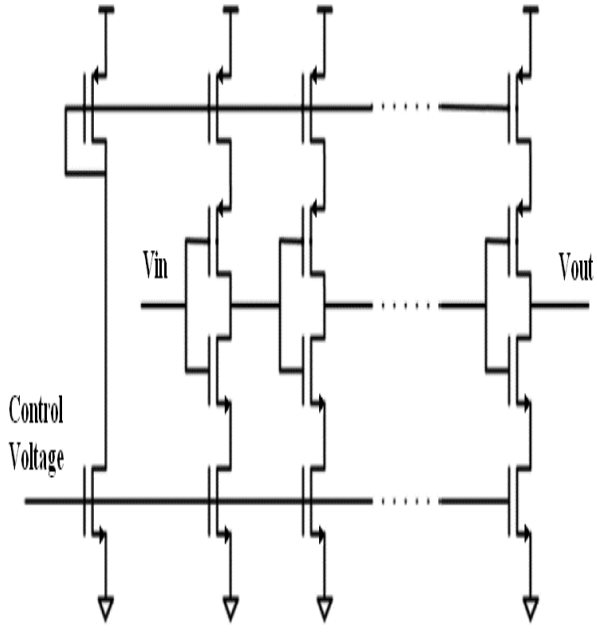


Figure 11: Inverter based VCDL

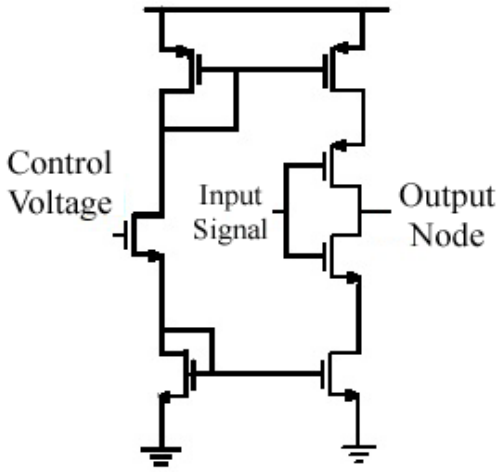


Figure 12: The designed VCDL

4. SIMULATION RESULTS

In order to better analyze the designed system, all the designed components were separately simulated to show their accurate behavior. Then, the DLL system was simulated for final conclusion. Simulations were performed based on TSMC 0.35μm CMOS technology and 3.3-V power supply. Each output node was loaded with buffers to provide a realistic simulation environment. Simulation results for flip-flop-based PFD of Fig. 4 are presented in Fig. 13. It is clear that, for small phase errors, the PFD could not extract precise values for UP and DOWN which indicated that the next CP could be correctly driven. Simulation results for the proposed PD of Fig. 5 are presented in Fig. 14. Since B is the leading signal, UP is in high state. As expected, the UP and DOWN signals can only accept logic 1 or zero values. Therefore, the acquisition time would be reduced which would enhance speed performance of the DLL. Simulation results of the proposed CP of Fig. 9 are given in Fig. 15. To better realize performance of the designed CP, UP and DOWN were considered square wave signals. The result indicated correct behavior of the designed CP on both of its output nodes. The proposed delay cell was simulated by two different control voltages in order to evaluate the non-linear relationship between delay time and control voltage. The simulation was done for $250MHz$ input frequency, the result of which is shown in Fig. 16. For $V_{control} = 1v$ according to Fig. 16(a), the delay time equaled to $255ns$. But, for $V_{control} = 2v$ according to Fig. 16(b), the measured delay time was equal to $238ns$. Finally, the whole system was simulated using a type II

LP. The parameters were set by the way that the step response of the system was similar to a first-order DLL. Input frequency was set to $500MHz$. Two delay cells were used to generate 180^0 phase shift at the output signal. Simulation results of this purpose, which are illustrated in Fig. 17, showed correct phase shift of the proposed system.

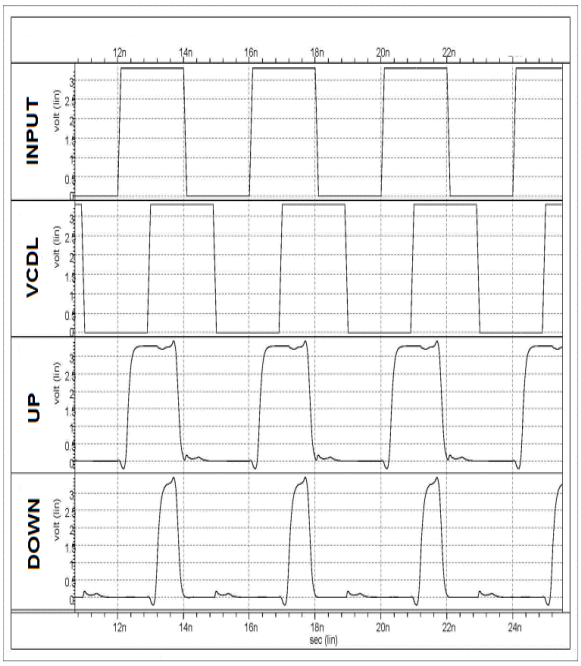


Figure 13: Simulation results for flip-flop based PFD

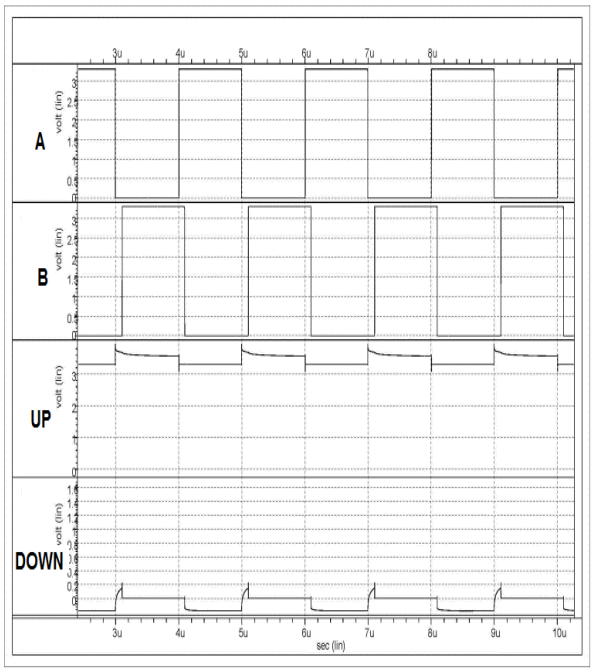


Figure 14: Simulation results of proposed PD

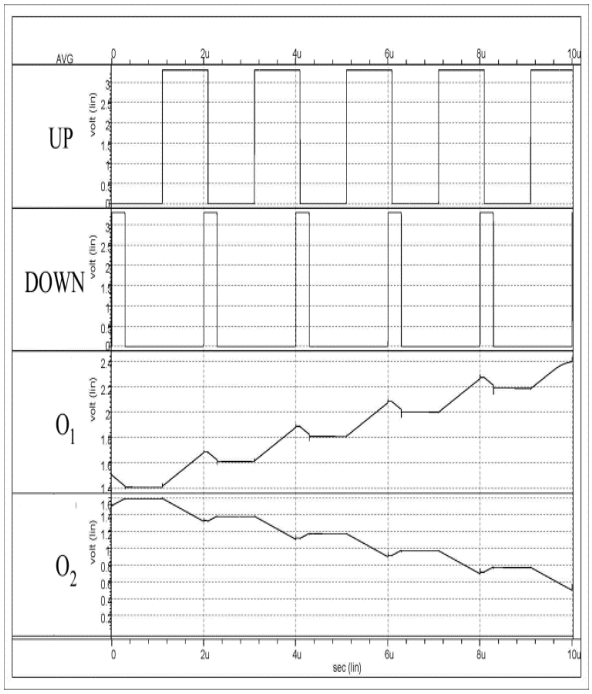


Figure 15: Simulation results of proposed CP.

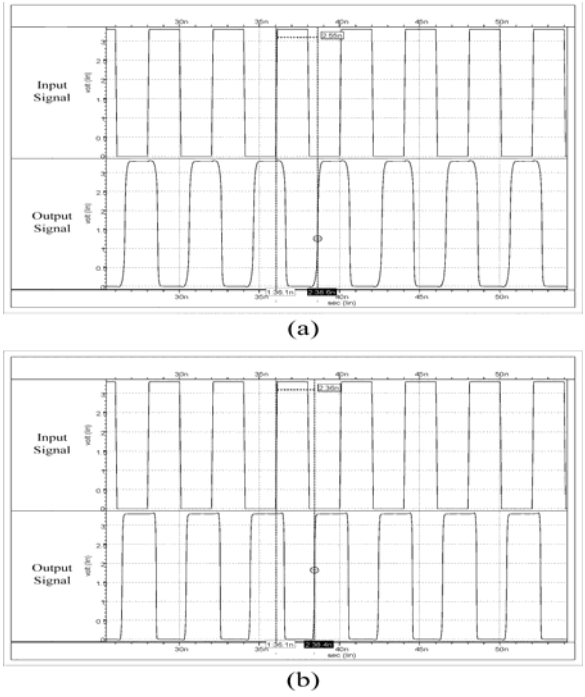


Figure 16: Simulation results of proposed delay cell (a) For $V_{control} = 1v$ (b) For $V_{control} = 2v$

5. Conclusion

A novel DLL was designed which could be widely used in high speed microprocessors. To design the new architecture, a new mixed structure was presented for

PD and CP, which could be triggered by double edges of the input signals. In addition, the blind zone was removed due to the elimination of reset signal. The static power dissipation of CP was minimized which reduced total power dissipation of the whole system. Operating frequency of the whole system was improved because the UP and DOWN signals were always in high or low logic states. To obtain more accurate phases at the output signal, a new delay cell was introduced which was controlled by a single voltage. This control voltage, by equalizing the rise and fall time, regulated duty cycle of the output clock. These features along with simplicity and low power consumption qualified the proposed architecture to be widely used in high speed systems.

REFERENCES

[1] C-K. K Yang, "Delay-Locked Loops - An Overview" from "Phase Locking in High Performance Systems," IEEE Press, 2003.

[2] M.-J. E. Lee, et al., "Jitter Transfer Characteristics of Delay-Locked Loops Theories and Design Techniques," IEEE J. Solid-State Circuits, vol. 38, pp. 614-621, Apr. 2003.

[3] J. Maneatis, "Low-Jitter Process-Independent DLL and PLL Based on Self-Biased Techniques," IEEE J. Solid-State Circuits, vol. 31, pp. 1723-1732, Nov. 1996..

[4] A. Chandrakasan, W. J. Bowhill, and F. Fox, "Design of High-Performance Microprocessor Circuit," New York, IEEE Press, 2001.

[5] Giovanni Bianchi, "Phase-Locked Loop Synthesizer Simulation," McGraw-Hill Electronic Engineering, 2005.

[6] R. E. Best, "Phase-Locked Loops: Design, Simulation, and Applications," McGraw-Hill, 2003.

[7] M. Mansuri, D. Liu and C.-K. K. Yang, "Fast Frequency Acquisition Phase-Frequency Detectors for GSamples/s Phase-Locked Loops," IEEE J. Solid-State Circuits, vol. 37, pp. 1331-1334, Oct. 2002..

[8] H. Notani, H. Kondoh, and Y. Matsuda, "A 622-MHz CMOS Phase-Locked Loop with Precharge-Type Phase Frequency Detector," VLSI Symp. Dig. Tech. Papers, June 1994, pp. 129-130.

[9] F. M. Gardner, "Charge-Pump Phase-Lock Loops," IEEE Transactions on Communications," COM-28, No. 11, pp. 1849-1858, November 1980.

[10] W. Rhee, "Design of High Performance CMOS Charge Pumps in Phase-Locked Loops," Proc. IEEE Intl. Symp. On Circuits and Systems, vol. 1, 1999, pp. 545-548.

[11] B. Razavi, Design of Analog CMOS Integrated Circuits, McGraw-Hill, New York, NY, 2001.

[12] Chen, Wu-Hsin, Maciej E. Inerowicz, and Byunghoo Jung. "Phase Frequency Detector with Minimal Blind Zone for Fast Frequency Acquisition," IEEE Transactions on Circuits and Systems II Express Briefs, 2010.

[13] Raghav, Himadri Singh, SachinMaheshwari, MolaSrinivasarao, and B. P. Singh. "Design of low power, low jitter DLL tested at all five corners to avoid false locking," 2012 10th IEEE International Conference on Semiconductor Electronics (ICSE), 2012.

[14] Chu Peng. "A novel Digital DLL and its implement on the FPGA.," International MultiConference of Engineers & Computer Scientists 2007 (Volume 1), 20070101.

[15] Jia Chen,''A delay lock loop for multiple clock phase/delays generation',' PHD thesis, Georgian statute of technology, 2005.

[16] Chih-Kong Ken Yang, ''an over view on Delay lock loops (DLLs),'' University of California, 2000.

BIOGRAPHIES

Ashkan Ghanbari was born in Karaj,Iran in May 1985. He received in 2009 the degreeof Engineer in Telecommunication Engineering from Tehran Emam Hussein University, and in March 2012 he obtained the Master of Science (MSc) in Electronics Engineering from the Department of Electrical Engineering of the Qazvin Islamic Azad University *(QIAU)*.

He intends to start PHD in Electronic Engineering from the beginning of 2013. His principal research areas are, Fiber Optics, MEMS and Electronic Systems. He has published several reports in nationaland international journal and conference proceeding.

Email: ghanbari.ashkan@yahoo.com

Ali Sadr received in 1987 the degree of Engineer in electronic engineering from Amir Kabir University and in 1991 he received Master of Science in electronic engineering from Iran University of Science and Technology. Also in 2002 he received PHD degree in Electronic engineering from Department of Electrical Engineering of the UMIST University (ENGLAND).

His principal research areas are, Fiber Optics, Ultrasonic and Electronic Systems. He has published several reports in national and international journal and conference proceedings

Author's picture is not available

Mehran Nikoo was born in Tehran, Iran in May 1975. He received in 2002 the degree of Engineer in Computer Engineering from Islamic Azad University, and in March 2012 he obtained the Master of Science (MSc) in Computer Engineering from the Department of Computer Engineering of the Islamic Azad University (MIAU).

His principal research areas are, Signal Processing, Network Communication and Electronic Systems.