British Journal of Applied Science & Technology
3(4): 1187-1200, 2013

SCIENCEDOMAIN ***international***
www.sciencedomain.org

A Fast-Lock Low-Jitter DLL with Double Edge Synchronization in 0.18μm CMOS Technology

Saman Mohammadi Mohaghegh[1] and Reza Sabbaghi-Nadooshan[1*]

[1]Department of Electrical Engineering, Islamic Azad University Central Tehran Branch, Tehran, Iran.

Authors' contributions

This work was carried out in collaboration between all authors. All authors designed the study. Author SMM performed the statistical analysis, wrote the protocol, and wrote the first draft of the manuscript. Author RSN managed the analyses of the study and literature searches. All authors read and approved the final manuscript.

Research Article

Received 19th March 2013
Accepted 12th July 2013
Published 30th July 2013

ABSTRACT

Aims: This paper describes a fast-lock, low-power, low-jitter and good duty-cycle correction capability delay locked loop with double edge synchronization which is mainly used in clock alignment process. A clock aligner's task is to phase-align a chip internal clock with a reference clock. The main advantage of delay locked loop rather than phase locked loop is related to good jitter performance of it. Double edge synchronization method leads to more power consumption and it can increase rms and peak-to-peak jitter therefore, in this work rms jitter, peak-to-peak jitter and power consumption are implemented to understand if this statement is always true or not. So, this case became one of our aims.
Study Design: Double edge synchronization delay locked loop.
Place and Duration of Study: Department of Electrical Engineering (Islamic Azad University, Central Tehran Branch), between February 2012 and September 2012.
Methodology: Comparing with single edge synchronization delay locked loops, double edge synchronization method has its own advantages and disadvantages. Using two phase frequency detectors, two charge pumps and two loop filters in double edge delay locked loops, increases the jitter and power consumption. In this paper, to overcome these challenges for the introduced delay locked loop circuit, proper blocks with suitable characteristic for each MOSFET were used which took a lot of time to find ones with the help of HSPICE simulator.
Results: All the simulation results are based on 0.18μm CMOS technology with 1.8V

**Corresponding author: E-mail: r_sabbaghi@iauctb.ac.ir;*

supply voltage. The HSPICE simulation results show the proposed delay locked loop circuit generates clock signals ranging from 750MHz to 1GHz. The maximum power consumption of the DLL circuit at 1GHz is 3.4mW. The maximum and minimum of rms jitters are 9.12 and 0.463ps and the maximum and minimum of peak-to-peak jitters are 124.89 and 2.52ps, respectively. The locking time of proposed delay locked loop is less than 20ns within the operating frequency band. Another feature of this architecture is that it has good duty cycle correction capability (50±0.9%). It should be note that, in double edge DLL it is so important to find a balance between duty cycle (should be around 50%), jitter and power consumption. Rms jitter, peak-to-peak jitter, power consumption and also duty cycle error are calculated by HSPICE simulator. (Cosmosscope program in HSPICE simulator can be used for these measurements).

Conclusion: Although designing double edge synchronization method in delay locked loops is challenging and it takes more area than single edge delay locked loops (which is mentioned as the main disadvantages of double edge delay locked loops and we all agree on this), by choosing suitable blocks it can be used without jitter performance or power consumption challenges. In other word, the results of this paper shows that all the effective and important items of introduced double edge delay locked loop (such as power consumption, rms jitter and peak-to-peak jitter) are as well as single edge delay locked loops in most articles. So when it is suitable to use double edge delay locked loop instead of single edge delay locked loop, it should be no concern about these items.

Keywords: Delay locked loop (DLL); phase locked loop (PLL); jitter; multistage clock buffer; voltage controlled delay line (VCDL).

1. INTRODUCTION

In the past decades, PLLs and DLLs have been widely used in high-speed applications, such as memory ICs, communication ICs, microprocessors, network processors, etc.

Ordinarily, if there is no frequency multiplication, using DLL for signal synchronization would be the better choice than PLL. Because DLL is a first order control system, it’s more stable and easier to design. Moreover, PLL suffers from a later locking time and jitter accumulation due to the closed loop Voltage Controlled Oscillator (VCO). On the other hand, the DLL using the VCDL instead of the VCO does not accumulate over many clock cycles therefore, DLL exhibits better jitter performance than PLL. In addition, DLL have smaller area and faster locking time than the PLL [1]. Low power, wide lock range, short locking time, and low jitter are focuses of the DLL design. In order to achieve low jitter operation, DLL designs require delay stage design with low supply and substrate noise sensitivity and good matching between the up/down CP currents [2].

In this work, a DLL structure with double edge synchronization with clock alignment capability of both rising and falling edges is proposed. In the rest of the paper, Section 2 of the paper describes architecture of DLL with clock alignment capability of both rising and falling output pulse edges and also concentrates on implementation of proposed structure. Section 3 includes simulation results by HSPICE simulation, and conclusions are given in Section 4.

2. MATERIALS AND METHODS

The building block of a conventional analog delay loop with double edge synchronization is shown in Fig. 1. This structure of a DLL circuit has clock alignment capability of both leading and trailing output pulse edges. A clock aligner's task is to phase align a chip internal clock with a reference clock, effectively removing the variable buffer delay and reducing uncertainty in clock phase between communicating VLSI IC constituents [3].

Constituents of Fig. 1 are: a VCDL, two differential Charge Pumps (CPs), CP1 and CP2, two first order low pass filters (LPFs), LPF1 and LPF2, and a multistage clock buffer (MCB). The reference clock, CLKref, is propagated through VCDL and MCB. The output signal, CLKout, is compared through with the reference input. If the delay difference from integer multiples of clock period is detected, the closed loop will automatically correct it by changing the delay time of the VCDL [4].

According to Fig. 1, in first stage two Phase Frequency Detectors (PFDs) are shown. Their function is to compare phase frequency of rising or falling edges between the input (CLKref) and output (CLKout). Note that the PFD1 is sensitive to a rising and PFD2 is sensitive to a falling pulse edge. These PFDs are high speed and have small dead zone. Hence, the DLL circuit has very fast lock feature compared to other dynamic Phase Detectors (PDs) [3, 5]. In the next stage, two ideal CPs and two LPFs are sketched in Fig. 1 UP1 and UP2 pulses cause Ip to add charge to the capacitors of LPF1 and LPF2, while DN1 and DN2 function is to discharge the capacitors. The output of CP1 and CP2 are Vctrl1 and Vctrl2, and they are connected to the VCDL control input (Vbn and Vbp). In the last stage of this figure, VCDL and MCB blocks are shown. The control input voltages of VCDL (Vbn and Vbp) can regulate the rising or falling clock pulse edge.

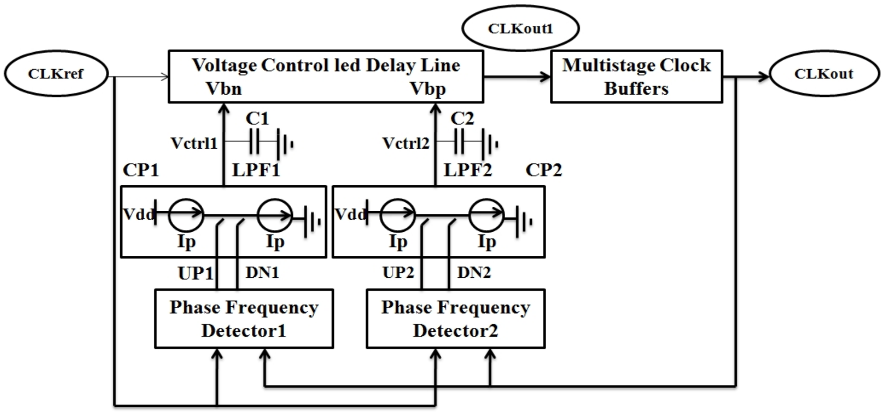
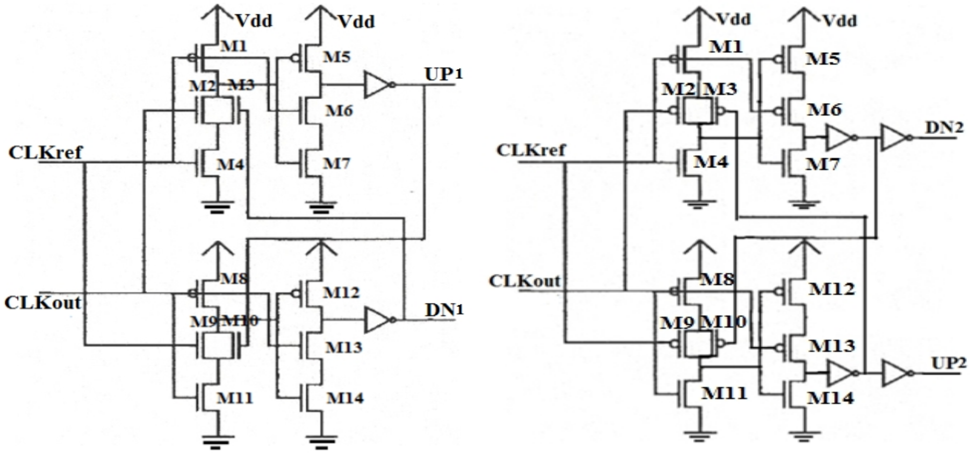


Fig. 1. DLL's structure with double edges synchronization
Circuit structure is discussed in Section 2.1, 2.2 and 2.3.

2.1 Phase-Frequency Detector

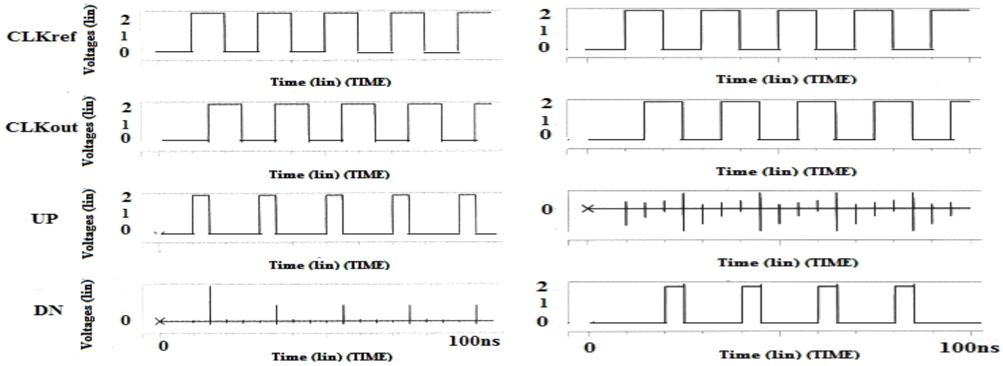
The PD function is to detect the phase difference between the reference clock signal and the feedback clock. PD can detect the skew of the clock, and it can be analog or digital as well. Nowadays digital phase detectors have become more popular. As its name indicates, PDs are sensitive to the phase difference between two signals, but they are not sensitive to frequency. Practically PD can work as frequency detector but with limited range. Thus, it is preferred to replace the PD with PFD. On the other hand, many PFDs have a large dead zone. As we know, dead zone occurs when the loop does not respond to small phase errors. Each width of the dead zone directly feeds to jitter in the DLL and should be avoided. Hence, this kind of PFDs cannot be used at high frequencies.

To overcome the speed limitation and reduce the dead zone, we proposed high speed PFDs which are sketched in Fig. 2. These schematics have fast lock loop feature. In this work, PFD1 [6] is used to detect a rising pulse edge. We needed another PFD for falling edge, therefore, we proposed new PFD (PFD2) to detect a falling pulse edge. Hence, PFD1 (Fig. 2 (a)) is sensitive to rising clock pulse edge, while PFD2 (Fig. 2 (b)) is sensitive to the falling edge. The width of UP and DN signals are proportional to the phase difference of the input signals [7]. PFD1 and PFD2 have three states. Compare to both PDs in reference [3], our proposed PFDs for rising and falling edge have smaller dead zone and they can also work as frequency detector since they can work in higher frequencies. As shown in Fig. 3(a), (b) and 4(a), (b) both reference and output have the same frequency but with phase difference, while in Fig. 5(a), (b) and 6(a), (b) both signals have different frequencies and phases. There is another state, which is when both signals have the same frequency and phase (Fig. 7). As can be seen in Fig. 3 and 5, CLKref leads and CLKout lags while in Fig. 4 and 6 CLKref lags and CLKout leads. In Fig. 7, both CLKref and CLKout are started from the same point and they are equal. Behaviour of up and down are shown in these figures. The point of sketching these figures is to make sure that our proposed PFDs work properly.



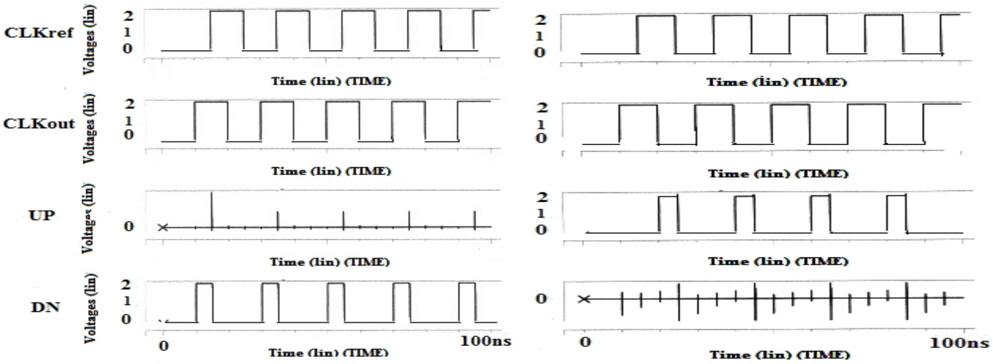
(a) (b)

Fig. 2. PFDs for (a) raising and (b) falling edges



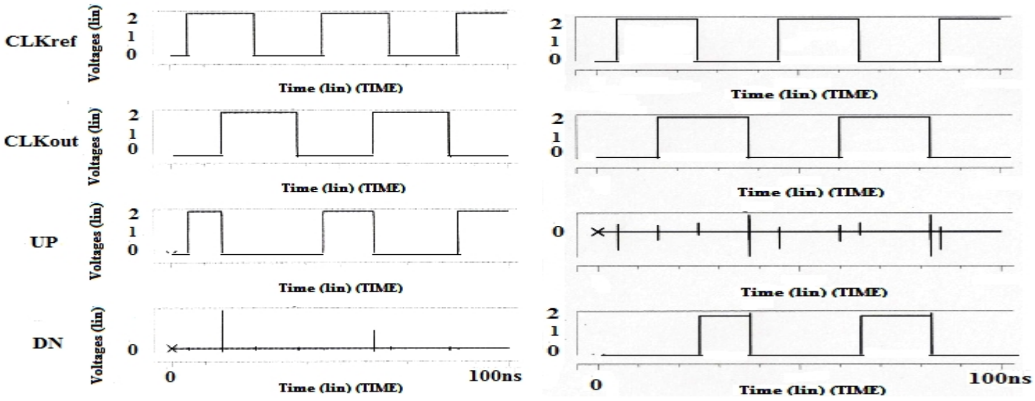
(a) (b)

Fig. 3. Behaviour of (a) PFD1 and (b) PFD2 with same frequency (CLK-ref leads)



(a) (b)

Fig. 4. Behaviour of (a) PFD1 and (b) PFD2 with same frequency (CLK-ref lags)



(a) (b)

Fig. 5. Behaviour of (a) PFD1 and (b) PFD2 with different frequencies (CLK-ref leads)

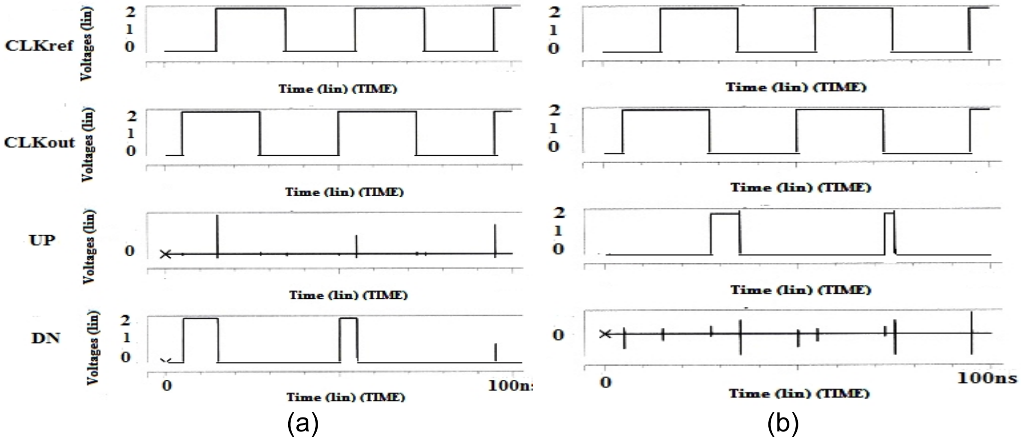


Fig. 6. Behaviour of (a) PFD1 and (b) PFD2 with different frequencies (CLK-ref lags)

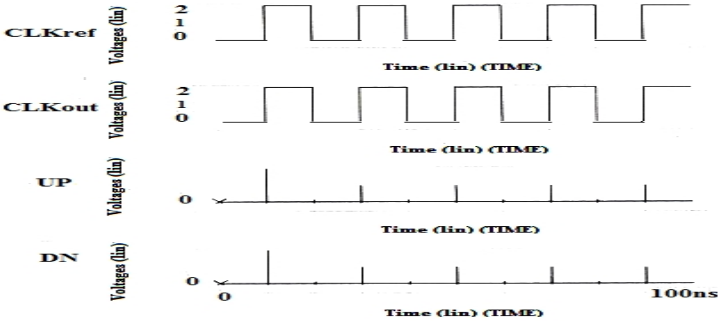
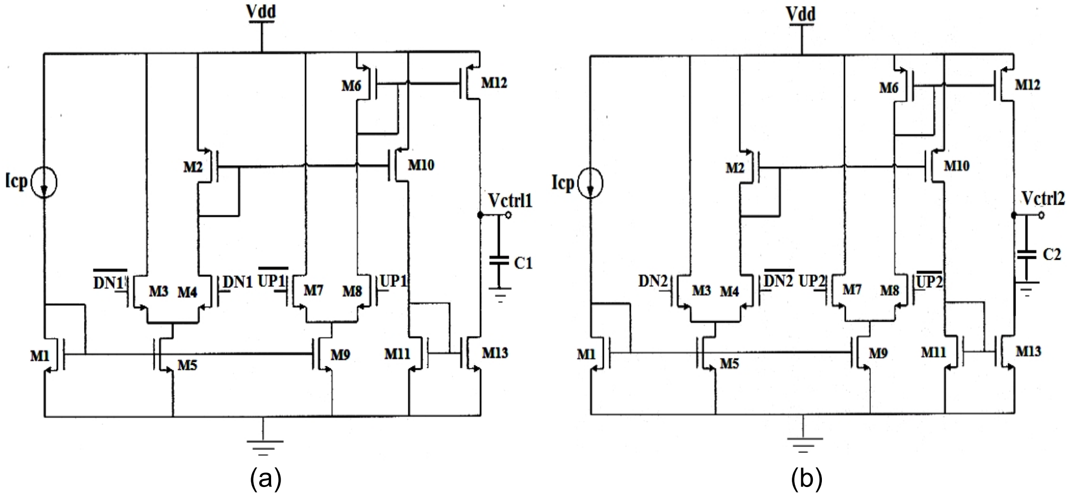


Fig. 7. Behaviour of PFD1/PFD2 with same frequency and phase

2.2 Charge Pump and Loop Filter

CP design is one of the most complicated parts of the DLL structure. The CP controls the charging/discharging current by UP/DN signal from PFD, and uses the phase difference between the up and down signals from the PFD to convert the phase error into current. Then, loop filter converts the current into the control voltage, by charging or discharging the capacitor and sending it to Vctrl to set the VCDL delay. Two differential CPs (CP1 and CP2) are used, because in this work, they seem more proper choice and the advantage of these structures is that switching time is improved by using the current steering switches. CP1 is depicted in Fig. 8(a). This CP can be used for rising edge [8]. Therefore, we needed another CP for falling edge, and we proposed CP2 (Fig. 8(b)) to detect a falling pulse edge.

One of the possible filters is a RC low-pass filter, like the filter mentioned in [9]. But in this work two simple capacitors are used as LPFs, and they are adjusted to be C1=C2=1.88pF. As we can seen in Fig. 1, the equivalent model of the CP and LPF consists of a source current, a sink current and two switches controlled by PFD output. When the output phase of the DLL circuit leads the reference phase, the current source switch opens and the current sink switch closes. Thus, the voltage in the capacitor decreases. The voltage in the capacitor increases if the reference phase leads the output phase. In this work, we adjusted source current and sink current to be ICP-up=ICP-down=100μA. This amount of source and sink current is chosen by HSPICE simulation test which works suitably in both mentioned CPs. Each CP can charges or discharges their filter capacitors. Vctrl1 and Vctrl2 (Vbn and Vbp) are the voltages on capacitors C1 and C2 respectively, and sets the VCDL stage propagation delay.



(a) (b)

Fig. 8. Implementation of CP for (a) raising and (b) falling edges

2.3 Voltage Controlled Delay Line and Multistage Clock Buffer

The most critical component in the performance of DLL is VCDL. A VCDL can influence DLL action. Therefore, VCDL can influence DLL stability and jitter performances. The implementation of a VCDL is composed of several variable delay elements connected in series. There are several examples of buffer elements [10], such as cascade delay cell, differential delay cell, shunt capacitor delay cell, etc.

The VCDL used in this work (Fig. 9) [9] consists of cascaded variable delay stage, is driven by the reference input clock, and the output is CLKout1, which is an input voltage for MCB circuit. As we can see, in this circuit, Vbp drives gates of M1 and M5 transistors, while Vbn drives gates of M4 and M8. A few (or many) cascade delay stage can be implemented in the structure of the proposed VCDL. We should note that, the load capacitance represented in the previous stage, is the input capacitance of the next inverter.

In high-speed design a multistage clock buffer implemented with a long inverter chain is often needed to drive a heavy capacitive load. For these designs, as well as for applications in which the timing of both edges of the clock is critical, it is difficult to keep the clock duty cycle at its ideal value of 50%, primarily due to various asymmetries in signal paths and unbalances of the p and n transistors in the long buffer. As a consequence the clock duty cycle will deteriorate from 50%, and in the worst case, the clock pulse may disappear inside the clock buffer, as the pulse width becomes too narrow or too wide [3]. As can be seen in Fig. 1 the output of VCDL (CLKout1), is an input for MCB and the output of MCB (CLK-out), interns to PFD1 and PFD2. The VCDL and MCB were implemented as a chain of ten delay elements (five VCDL and five MCB). It should be noted that proposed VCDL and MCB are tested by HSPICE simulation for their jitter performances. Compare to reference [3], Selected VCDL and MCB take much less area than mentioned reference since they have fewer delay cells.

Time delay variation of the leading and trailing pulse edge term of control voltage Vctrl (Vbn and Vbp) is presented in Fig. 10. If the control voltage Vctrl, decreases, the time delay of the trailing edge increases and time delay of leading edge decreases and vice versa.

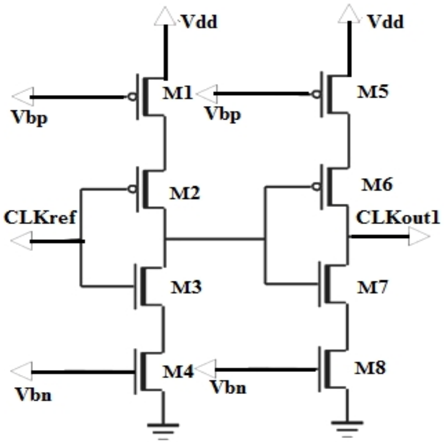


Fig. 9. VCDL Implementation

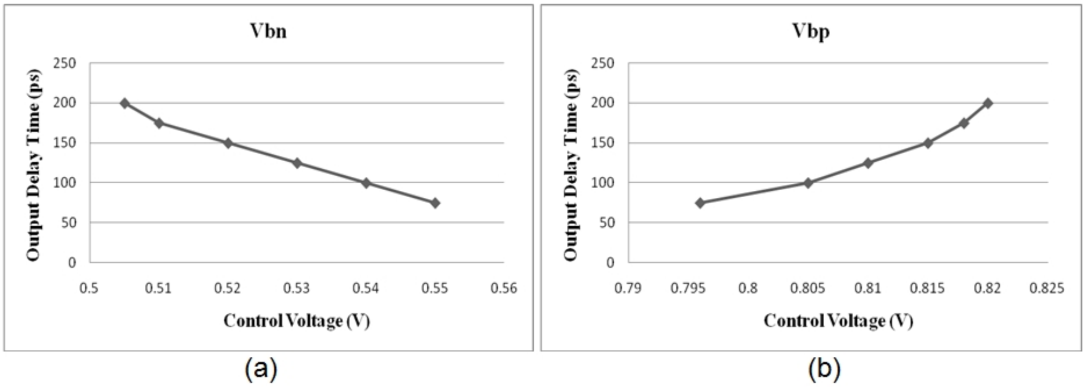
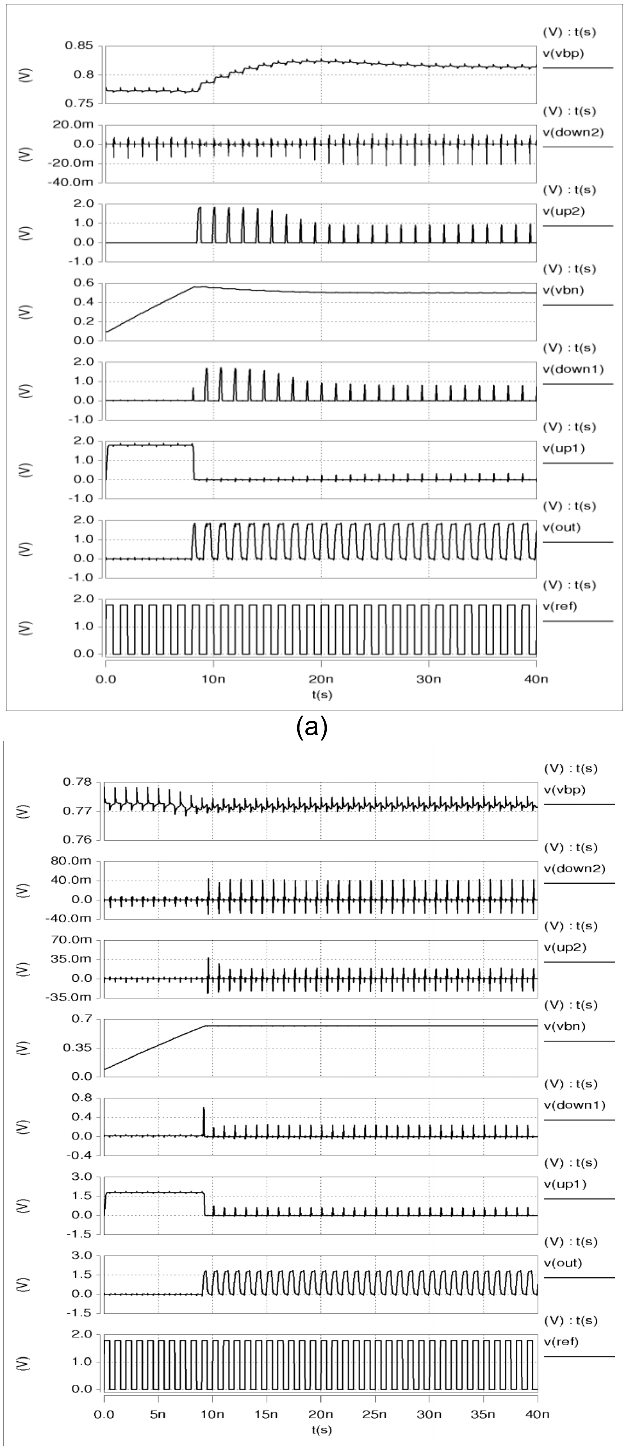


Fig. 10. Output (a) rising and (b) falling edge delay versus frequency

3. RESULTS AND DISCUSSION

The proposed DLL structure with double edge synchronization is implemented in 0.18µm CMOS technology, with the supply voltage of 1.8V. The operational frequency range is from 750MHz to 1GHz. Fig. 11 shows the result of DLL operation at (a) 750MHz and (b) 1GHz. In Fig. 11 illustrates the behaviour of CLK-ref and CLK-out, and waveforms of UP1 (UP2), DN1 (DN2). Also, this figure shows the behaviour of CP's output (Vctrl1 and Vctrl2). As can be seen from UP1 and DN1 (UP2 and DN2) signals define the control voltages Vctrl1 (Vctrl2). The locked time of DLL is less than 20ns within the operating frequency band. This circuit also has good duty cycle correction capability. Duty cycle error is also measured and as shown in Fig. 12, within the full operating range, the duty cycle error is less than ±0.9%.



(a)

(b)

Fig. 11. Simulation of DLL with double edge synchronization at (a) 750MHz (b) 1GHz

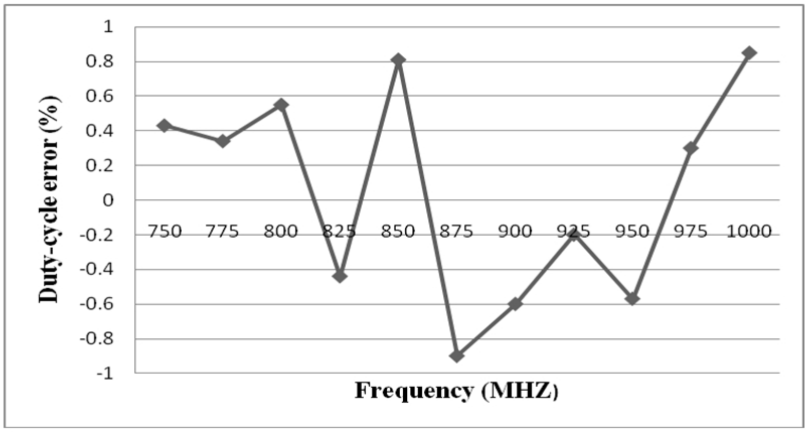


Fig. 12. Duty cycle error versus frequency

Fig. 13 and Fig. 14 show output rms jitter and peak-to-peak jitter versus operation frequency. As can be seen, when frequency becomes bigger, both jitters are decreasing.

The maximum and minimum of rms jitters are 9.12 and 0.463ps and the maximum and minimum of peak-to-peak jitters are 124.89 and 2.52ps, respectively (these results seems acceptable for even single edge DLLs). Fig. 15 shows power consumption variation during the frequency range of DLL. Power consumption increases proportional to frequency in whole of operation range (It can be seen from the picture that the frequency range is calculated from 750 775…….up to 1000MHz). The maximum and minimum power consumption at 750MHz and 1GHz are 2.2mW and 3.4mW, respectively. As we compared this range with other power consumption range in other articles, it's been realized that this range is the normal range in single edge DLLs (it should be mentioned that there are some works that could achieve lower power consumption in single edge DLLs than our work).

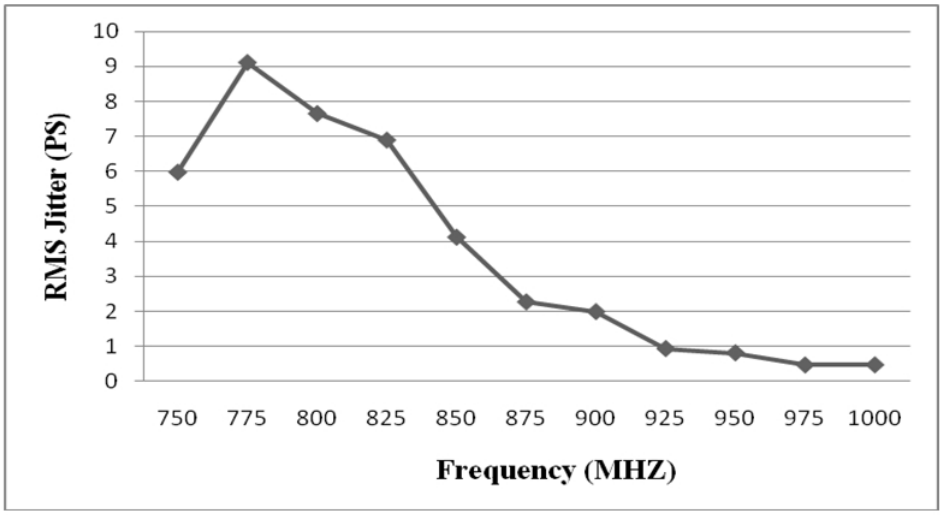


Fig. 13. RMS Jitter versus frequency

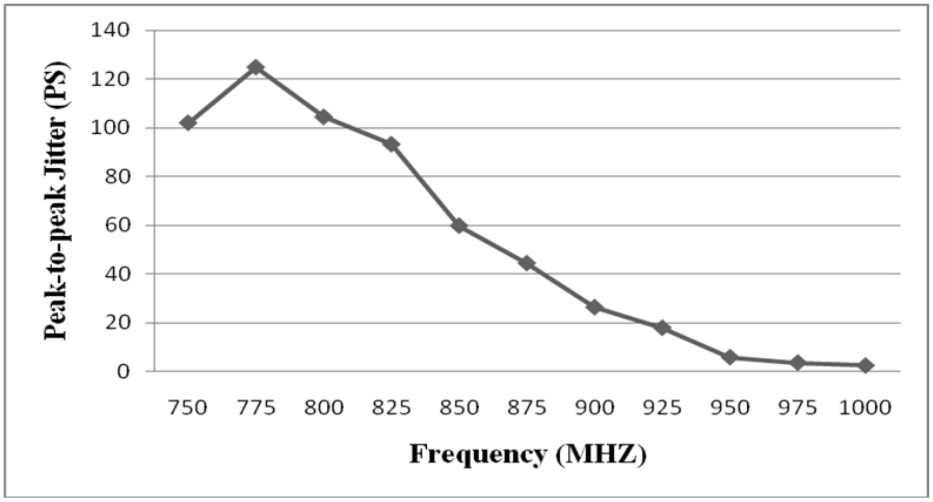


Fig. 14. Peak-to-peak Jitter versus frequency

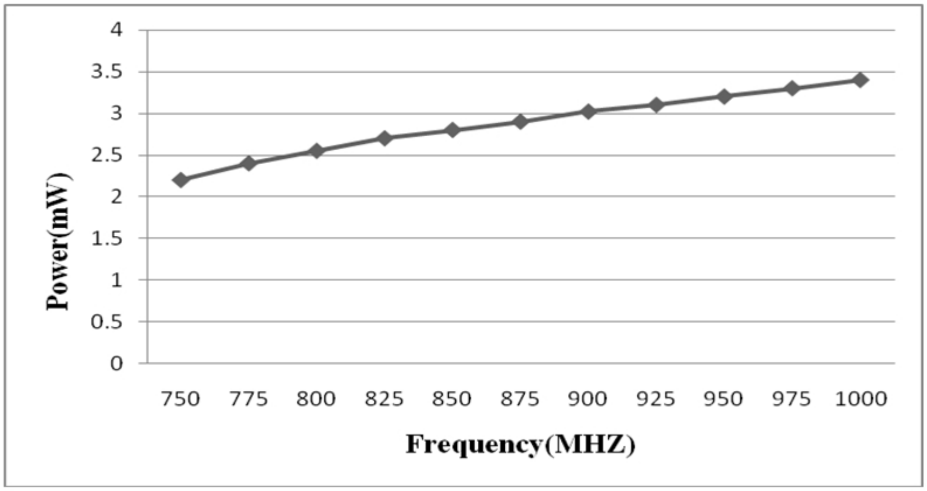


Fig. 15. Power consumption versus frequency

Table 1, gives the performance summary of the proposed PWCL and the characteristics of other published PWCLs. As it can be seen, reference [3] is also double edge synchronization DLL. In this work, approximate architecture proposed in this reference is used, but with more proper blocks and different process to improve important items of DLL. Therefore, compare to this reference, different technology is used, locking time is faster, wider range of frequency is achieved. On the other hand, rms jitter, peak-to-peak jitter and power consumption were computed, which did not mention in reference [3] at all and as we know these two items are the most important ones in DLLs. It can obtain that in our work, good jitter performance and low power consumption are achieved too. In the rest of the table, our work is compared with other references [11,12,13,14]. It should be mentioned that our previous work [15] is also used to make the results of this work better (specially the locked time, rms jitter and peak-to-peak jitter).

It must also be pointed out that the reported information for this work are extracted from the simulation results, whereas some of those previously reported works are from the experimental results. So some problems such as parasitic elements, impedance mismatch and calibration errors have been ignored, which could influence the performance of proposed system for future fabrication and test setup.

Table 1. Performance comparisons of PWCLS

References	This work	[3]	[11]	[12]	[13]	[14]
Cmos process	0.18μm	1.2μm	0.13μm	0.18μm	0.18μm	0.18μm
Supply voltage	1.8V	5V	N.A.	1.8V	0.6V	1.8V
Operational frequency	0.75-1GHz	55-166MHz	15-600MHz	150-400MHz	85-550MHz	200-400MHz 1-2GHz
Locked time	<20ns	<200ns	N.A.	13 cycles	N.A.	>6 clock cycles
Duty-cycle error	±0.9%	±0.8%	N.A.	N.A.	N.A.	N.A.
Rms Jitter	0.468ps@1GHz	N.A.	9ps@600MHz	25ps@400MHz	3.8ps@550MHz	3.22ps@300MHz
Peak-to-peak jitter	2.52ps@1GHz	N.A.	N.A.	N.A.	25.6ps@550MHz	23.6ps@300MHz
Power consumption	3.4mW@1GHz	N.A.	20mW @600MHz	N.A.	2.4-4.2mW	31.5mW @ 300MHz , 1.5GHz
Singel/double edge correction	double	double	single	single	single	single

4. CONCLUSION

In this paper, DLL architecture with double edge synchronization based on 0.18μm CMOS technology at 1.8V power supply is proposed. Operating frequency range is from 750MHz to 1GHz. Fast-lock double edge synchronization DLL (maximum 20ns) is achieved by using high speed double edge PFD1 and PFD2. Proposed PFDs also have small dead zone. Also, differential CPs (CP1 and CP2) are used, because in this work, they seem more proper choice and the advantage of these CPs is that switching time is improved by using the current steering switches. Another feature of this structure is that it has good duty cycle correction capability (50±0.9%). On the other hand, as we know, double edge synchronization method leads to more power consumption and it can increase rms and peak-to-peak jitter, because of using two PFDs, two CPs and LPFs instead of one, therefore, in this work analysis of rms jitter, peak-to-peak jitter and power consumption is also implemented. The maximum and minimum power consumption at 750MHz and 1GHz are 2.2mW and 3.4mW, respectively. The maximum and minimum of rms jitters are 9.12 and 0.463ps and the maximum and minimum of peak-to-peak jitters are 124.89 and 2.52ps, respectively. Although designing double edge synchronization method in DLLs is challenging and it takes more area than single edge DLLs (which is mentioned as the main disadvantages of double edge DLLs and we are working on this part, to minimize the chip area for our next DLL project), these results show that double edge DLLs can be used without jitter performance or power consumption challenges. In other word, the results of this paper shows that most of the effective and important items of introduced double edge DLL are as well as single edge DLLs in most articles. So when it is suitable to use double edge DLL instead of single edge DLL, it should be no concern about these items.

COMPETING INTERESTS

Authors have declared that no competing interests exist.

REFERENCES

1. Chen P-T, Chang CH-CH, Liu H-Y, Lo Y-L. A fast-lock analog multiphase delay-locked loop using a dual-slope technique. IEEE Computer Sociaty, International Symposium On Computer, Consumer and Control. 2012:954-957.
2. Moazedi M, Abrishamifar A, Ghanaatian A. A low-power multiphase-delay-locked loop with a self-biased charge pump and wide-range linear delay element. Cyber Journals, Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Microelectronics (JSAM). 2011:1-8.
3. Stojcev M, Jovanovic G. Clock aligner based on delay locked loop with double edge synchronization. Microelectronics Reliability. 2008;48(1):158-166.
4. Chang H-H, Lin J-w, Yang C-Y. A wide-range delay-locked loop with a fixed latency of one clock cycle. IEEE Journal of Solid-State Circuits. 2002;37(8):1021-1027.
5. Jovanovic G, Stojcev M, Krstic D. Delay Locked Loop with linear delay element. In Proceeding of Seventh International Conference. TELSIKS, Nis, Serbia. 2005:397-400.
6. Johansson H. A simple precharged CMOS phase frequency detector. IEEE Journal of Solid-State Circuits. 1998;33(2):295-299.
7. Weng RM, Liu CY, Lu YC. A low jitter DLL-Based pulsewidth control loop with wide duty cycle adjustment. APCCAS. 2008;418-421.
8. Noh J, Jeong H. Charge-pump with a regulated cascode circuit for reducing current mismatch in PLLs. World Academy of Science, Engineering and Technology. 2007:185-187.
9. Ayat M, Atani R, Mirzakuchaki S, Zamanidoost A. Design and simulation of a CMOS DLL-based frequency multiplier. IEEE Symposium on Industrial Electronics and Applications (ISIEA), Penang, Malaysia. 2010:450-455.
10. Ken Yang C.K. Delay-locked loops-an overview. phase-locking in high-performance systems. IEEE Press. 2003:14-21.
11. Hoyos S, Tsang CH, Vanderhaegen J, Chiu Y, Aibara Y, Khorramabbadi H, Nikolic B. A 15MHz to 600MHz, 20mW, 38mm Split-Control, Fast Coarse, Locking Digitall DLL in 0.13µm CMOS. IEEE on VLSI Systems. 2012;20(3):564-568.
12. Jia CH, Milor L. A DLL Design for Testing I/O Setup and Hold Times. IEEE Transactions in VLSI Systems. Nov. 2009;17(11):1579–1592.
13. Lu C-T, Hsieh H-H, Lu L-H. A 0.6 V Low-Power Wide-Range Delay-Locked Loop in 0.18 um CMOS. IEEE Microwave and Wireless Copmonents Letters. 2009;19(10):662-664.

14. Kuo C-H, Lai H-J, Lin M-F. A Multi-B and Fast-Locking Delay-locked loop with jitter bounded feature. IEEE Transaction on Ultrasonics, Ferroelectrics, and Frequency Control. 2011;58(1):51-59.
15. Mohammadi Mohaghegh S, Sabbaghi-Nadooshan R. Design and simulation of DLL with double edge synchronization in 0.13µm CMOS technology. International Journal of Scientific and Engineering Research. 2013;4(1).

Peer-review history:
The peer review history for this paper can be accessed here:
http://www.sciencedomain.org/review-history.php?iid=226&id=5&aid=1780