International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 8 (2017) pp. 1664-1670

DLL Based Clock Generator with Low Power and High Speed Frequency Multiplier

Thutivaka Vasudeepthi[1,] P.Malarvezhi[2] and R.Dayana[3]

[1-3]Department of ECE, SRM University
SRM Nagar, Kattankulathur, Kancheepuram District,
Tamil Nadu, India.

Abstract

An effective less power, and enhancement of frequency multiplier for a delay-locked loop clock generator is used to produce an increased frequency by multiplied clocks. Here edge combiner which we have used to enhance multiplied frequency gives more speed and effective operation is done as we use different structure and overlap canceller. On other hand by applying the logic that satisfies our requirement of pulse generator and multiplication-ratio control logic design, we reduces the delay difference between positive- and negative-edge of the generated pulse, which causes a known jitter which we called as deterministic jitter.

Keywords: Clock generator (clk), delay locked loop (DLL), differential current voltage switch logic (DCVSL), edge combiner (EC), and frequency multiplier.

INTRODUCTION

The clock generator is generally implemented using a phase-locked loop (PLL) to change the output clock frequency. PLLs have several backlogs such as the difficulty of design, expensive loop filters, and accumulation of jitter [4]. Delay-locked loops (DLLs) gives better results compared to PLLs, they overcome the drawbacks of PLL; however, because a DLL uses a delay line instead of an oscillator, its output clock frequency is always same as its input clock frequency. Therefore, a DLL alone cannot be used as generator of clocks. Many DLL dependant clock generators have been used to remove these problem of clock generations [4]–[13].

The DLL based clock generator has a core DLL and a frequency multiplier, and this frequency multiplier again has two blocks one for combining edges and other for combining both positive and negative edges. To generate different frequencies of various ranges, a controlled logic of multiplication ratio is implemented. The DLL core produces multiphase clocks using a reference clock in the DLL core. The pulse generator gives the appropriate number of pulses from the multiphase clocks as per the controlled logic of multiplication, and the combined edges provides a multiplied clock by the generated pulses.

Usually, the maximum multiplication ratio of the frequency multiplier is half of its number of multiphase clocks. Because the frequency multiplier produces the multiplied clock by gathering the multiphase clocks. There will be no jitter accumulation. There by, the frequency multiplier can easily change multiplication ratios. To have a increased maximum multiplication ratio, the logic used or output loading of the frequency multiplier have to be increased. However, it gradually degrades the maximum multiplied clock frequency. To control the defects of the frequency multiplier used before, a new effective frequency multiplication technique is used in the following paper. A new rearranged structure and canceller of overlapped signal are used in the edge combining process. Here the technique used in frequency multiplier increases the range of the frequency of the multiplied clock by consuming less power per frequency ratio and higher reliability than previous frequency multipliers.

PRE-EXISTING TECHNIQUES:

The generated multiplied clock pulses are generated parallely [i.e., the *k*th pulse is generated directly following the (*k*−1) th pulse], the pulses might overlap for getting process variation or layout mismatch as they come through the multiplication-ratio control logic; this may cause a short-circuit current to flow in the edge combiner, which in turn may lead to bulk power consumption or improper usage of the frequency multiplier. The edge combining property of the edge combiner parallely increases with the controlled logic of multiplication ratio, as a PU-N and PU-P are given after the pulse generation. Hence the maximum multiplication ratio increases simultaneously by one simultaneously by one.

The frequency multiplication technique used in this gives better results than the pre–existing methods. The frequency multiplication techniques majorly uses a D-flip–flop to generate pulses, a controlled logic of multiplication-ratio, and one push–pull-stage for the combination of edges. Normally combining of edges structure, is used to give effective multiplication of frequencies with consumption of low power and reliability. A 50% duty cycle for its multiplied clocks is guaranteed, as the generation of pulses is done respectively [i.e., the *k* th pulse is generated directly following the (*k*−1) th pulse], the pulses might overlap as the process variation or layout mismatch occurs because they come across the multiplication-ratio control logic; this may cause a short-circuit current to flow in the edge combiner, which may lead to more power consumption of the frequency multiplier.

The edge combining property of the edge combiner parallely increases with the controlled logic of multiplication ratio, as a PU-N and PU-P are given after the pulse generation. If the maximum multiplication ratio increased by one then this pmos and nmos are added to the output of the edge combiner. We have a controlled logic of multiplication-ratio, a pulse generator with AND gate, and a differential cascade voltage switch (SW) logic (DCVSL)-stage for the combination of edges. The multiplication of frequency technique gives many different clocks by low power consumption. We add only one PD-N to each different clocks of the multiplied frequency technique so that the maximum multiplication ratio is increased by one, hence the combination of edges in this is increased by one when compared with the pre-existing techniques.

To satisfy the above said condition, the PD-N pull down N transistor is kept unchanged till the positive and negative edges of the multiplied different clocks are given by the combination of edges. If we turn on the PD-N is, for the edge combination a small-sized PU-P to prevent controversy between the PU-P and the PD-N. Due to these contra versions, the frequency multiplier may not give high-speed operation and cannot guarantee 50% duty cycle for the multiplied clock. At the end, interphase timing distortion may occur when the multiphase clocks go across the controlled logic multiplication-ratio, which may give pulses in overlapping condition similar to that of pre-existing technique. Fig. 1 shows the structures of different frequency multipliers which are used to multiple the frequency.wecan see different frequency multipliers in this figure[11]–[13].

The frequency multiplier [11] contains a generator of pulses using D-flip-flop a controlled logic of multiplication-ratio, and a edge combining stage using push pull stage, as shown in Fig. 1(a).

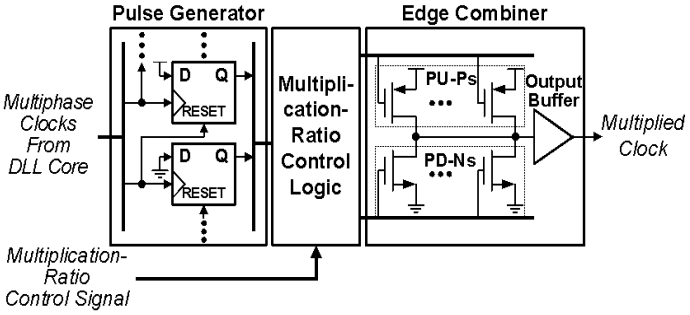

Figure 1(a)

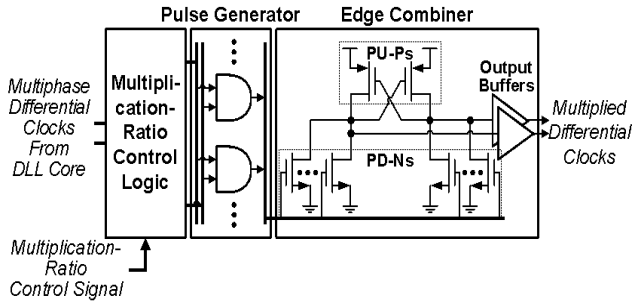

Figure 1(b)

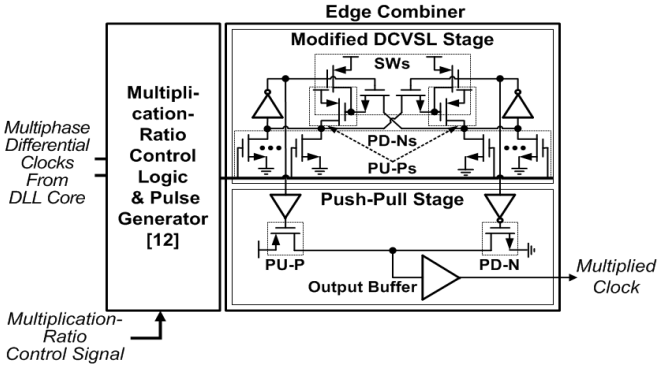

Figure 1(c)

Fig. 1. Structures of the multiplied frequencies in (a) [11], (b) [12], and (c) [13]. Fig. 1(b) shows the structure of the frequency multiplication in [12]. The multiplication of frequencies starts with pulse generator using D-flip–flop, a controlled logic of a multiplication-ratio, and an edge combiner using push–pull-stage. Basic combination of edges structure, the frequency multiplier give multiplied frequencies of high range and the power utilized for this process of getting high rang frequencies is low[11]. The frequency multiplier gives clocks with different multiplied frequencies. To satisfy the above said condition, the PD-N pull down N transistor is kept unchanged till the positive and negative edges of the multiplied different clocks are given by the combination of edges. . If we turn on the PD-N is, for the edge combination a small-sized PU-P to prevent controversy between the PU-P and the PD-N. Due to these contraversions, the frequency multiplier may not give high-speed operation and cannot guarantee 50% duty cycle for the multiplied clock. At the end, interphase timing distortion may occur when the multiphase clocks go across the controlled logic multiplication-ratio, which may give pulses in overlapping condition similar to that of pre-existing technique. Fig. 1 shows the structures of different frequency multipliers which are used to multiple the frequencies. We can see different frequency multipliers in this figure [11]–[13].

The frequency multiplier [11] contains a generator of pulses using D-flip-flop a controlled logic of multiplication-ratio, and a edge combining stage using push pull stage, as shown. Fig. 1 shows the structures of the recently proved frequency multipliers which perform better than most previous frequency multipliers [11]–[13]. The multiplied frequencies in [13] has the same structure as in [12], with hope that its edges combined consists of a modified DCVSL stage and a push–pull stage, as shown in Fig. 1(c). The modified DCVSL stage has switches that turn the PU-P OFF when the PD-N is ON, in order to prevent from contraverse between the PU-P and the PD-N. Hence, there is no need of a small-sized PU-P anymore; this property effectively overcomes the slow operation problem of the DCVSL-stage edge combiner in [12]. Instead, by adopting a push–pull stage, 50% duty cycle can be guaranteed for the multiplied clock. Finally, as the modified DCVSL stage maintains the characteristics of a DCVSL structure, only one PD-N is used in each differential output of

the modified DCVSL stage where as the maximum multiplication ratio is increased by one, as in the edge combiner in [12]. Even though, there exists some conflicts in the frequency multiplier in common with the frequency multiplier in [11], [13] and [12], including cancellation of pulse overlapping.

PROPOSED MULTIPLIER

Our designed clock generator based on DLL has a DLL core and the frequency multiplier is shown in the fig (2). Lock time is the major parameter in clock generators. To acquire that a phase detector which is dual edge triggered is used in the DLL core [14]. As we have seen in the pre-existing techniques for the multiplication of frequency basically we have a generator of pulses, controlled logic of multiplication ratio and a combiner of edges.Fig.3.shows the working of the DLL in [14] and the proposed multiplication of frequencies technique.

The phase-detector of double edge triggered compares the positive and the negative edges of CLKREF, DCK and CLKOUT, DCK, which are the duty cycle came from the clocks of CLKREF and CLKOUT with the help of duty-cycle keeper.

The DLL is locked within the 300 cycles in all voltage temperature ends of the dual-edge detection characteristic, and gives 32-phase differential clocks (CLK0:32 and /CLK0:32*)*. Using the 32-phase differential clocks, the pulse generator makes pulses (PG0:31 and /PG0:31*)* from the combination of positive- and negative-edges. The controlled logic of multiplication ratio accepts only the suitable pulses from PG0:31 and /PG0:31 and gives a multiplied phase clocks MC*P,* 0:15 and MC*N,* 0:15 as per the logic i.e. the controlled logic of multiplication ratio.

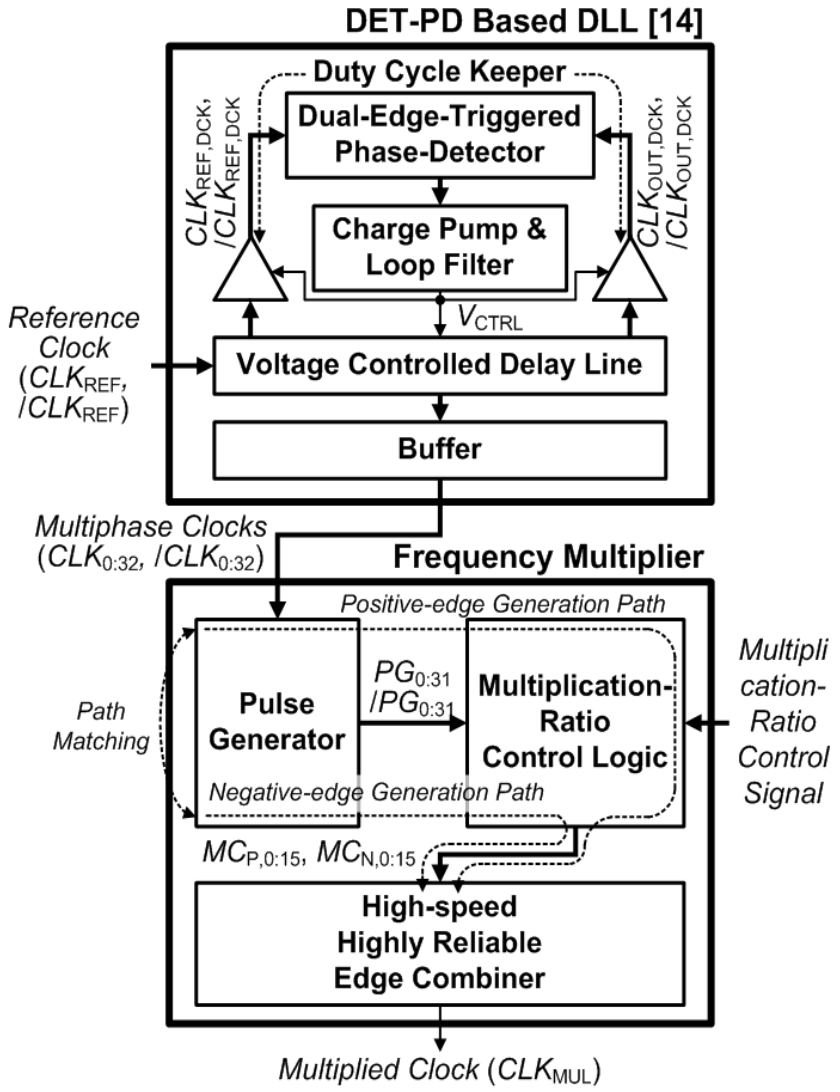

Figure 2. Structure of the multiple clock generator

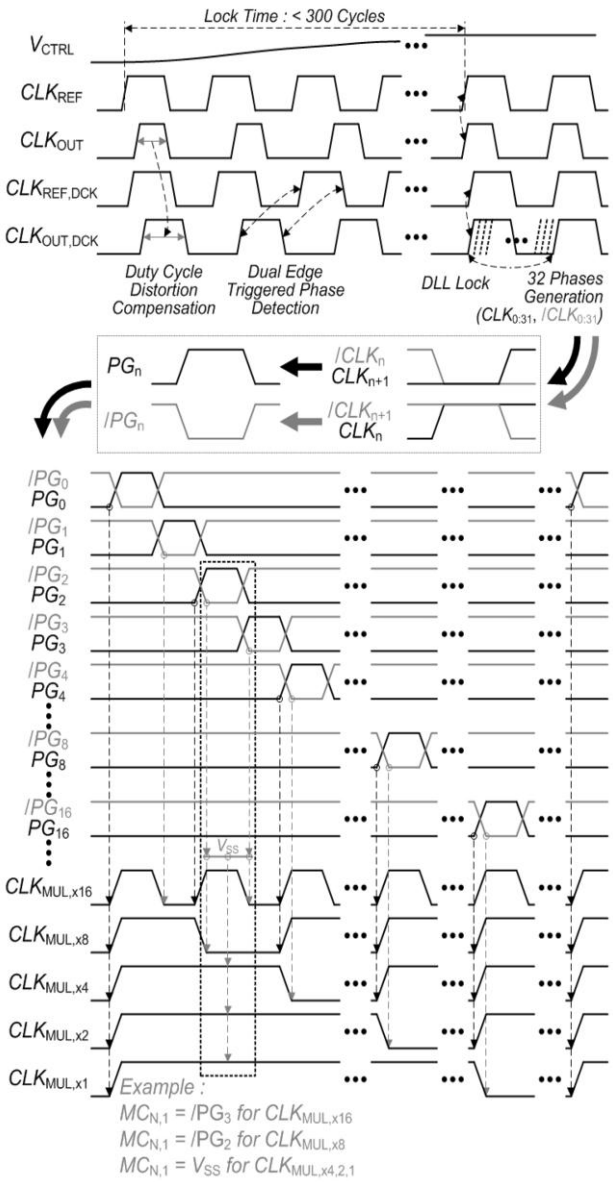

Figure 3. Functioning of the multiple clock generator.

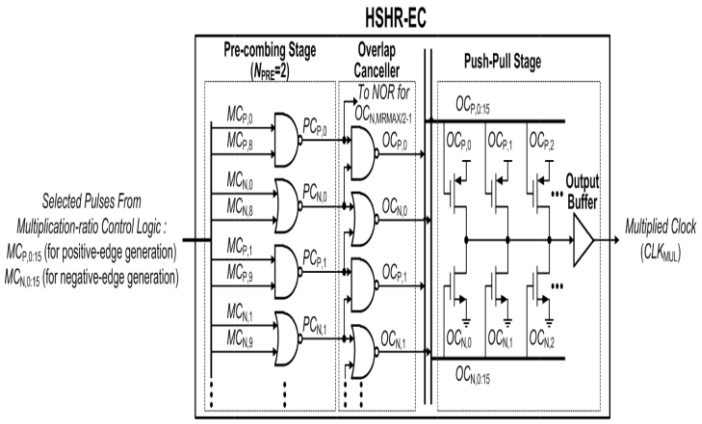

Figure 4. Structure of the edge combiner.

Fig.4. consists of prearranged signal stage, overlap canceller, and push–pull stage. Here the dual edge combiner, prearranging, and push–pull stage can have the more number of multiplied clock frequency. The overlap canceller has a property to cancel the signals to be overlapped and it guarantees the stability.

If the number of signals gets combined in the precombining stage (*N*PRE*)* increases, the number of PU-Ps and PD-Ns used in the push–pull stage can be reduced by a factor of *N*PRE.we can also see that, by increasing *N*PRE, the maximum multiplied clock frequency of the HSHR-EC can be collected; as the logic used and the number of NAND and NOR gates in the precombining stage are equal to log2*N*PRE and 32(1–1/*N*PRE*)*, respectively, a large *N*PRE causes the precombining stage to be effective to process variation, which gives us large deterministic jitter effortlessly. Thus, *NPRE* is limited to two, which corresponds to a logic of one in the HSHR-EC, and thus, the precombining stage can be simply done by using NAND and NOR gates. Fig. 5 gives a brief information on the operation of the edge combiner. As the number of signals gets combined in the precombining stage (*N*PRE*)* increases, then the number of PU-Ps and PD-Ns which are used in the push–pull stage can be reduced by a factor of *N*PRE.the maximum number of multiplied clocks are generated by increasing *N*PRE; because of the logic used and the number of NAND and NOR gates in the precombining stage are equal to log2*N*PRE and 32(1–1/*N*PRE*)*, respectively, a large *N*PRE causes the precombining stage to be effective for the process variation, which in turn could cause a large deterministic jitter. Thus, *N*PRE is limited to two, which corresponds to a logic used of one in the HSHR-EC, and thus, the precombining stage can be simply done by us using NAND and NOR gates. As we know that for the frequency multipliers in [11]–[13], the techniques of frequency multiplier may be caused from the pulse overlapping to the controlled logic of multiplication-ratio. To overcome this, an overlap canceller is added in between the pre-combining and the push–pull stages. Its operation is also shown in Fig. 5.As shown in Fig. 4, the overlap canceller consists of simple NAND and NOR gates. If the delay of PC*P*,0 (PC*N*,0*)* pulse path is less (more) than that of PC*N*,0 (PC*P*,0*)* pulse path because of the process variations or layout mismatches, overlap between the high level of PC*P*,0 and the low level of PC*N*,0 can be the outcome. The NAND gate with PC*P*, 0 and PC*N*, 0 inputs makes OC*P*, 0 low, only when both PC*P*, 0 and PC*N*, 0 are high. This remove the problem of pulse overlapping.in the same way, if the delay of the PC*N*,0 (PC*P*,1*)* pulse path is less (more) than that of PC*P*,1 (PC*N*,0*)* pulse path due to process variations or layout mismatches, overlap can be obtained in between the low level of PC*N*,0 and the high level of PC*P*,1. As a NOR gate makes OC*N*, 0 high only when both PC*N*, 0 and PC*P*, 1 are low, pulse overlapping is removed. Hence, a stable and highly effective operation of the frequency multiplier can be generated.

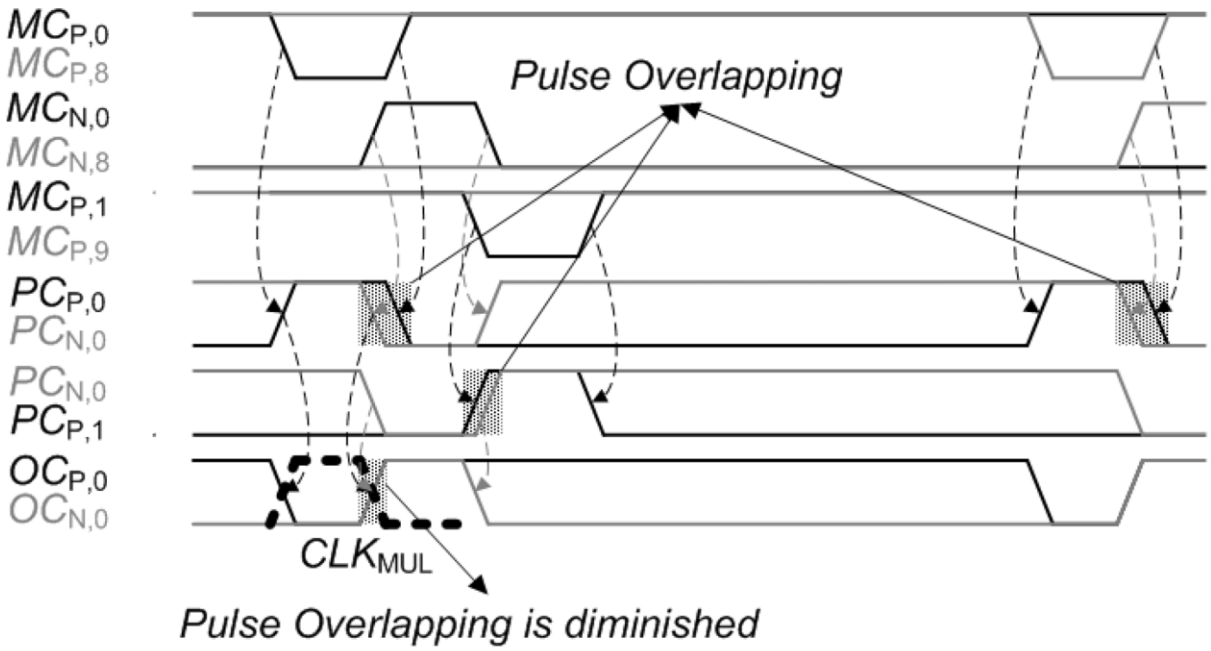

Figure 5. Operation of the edge combiner.

The maximum multiplied clock frequencies of the frequency multipliers in [11] and [12] are less than the frequency multiplier in [13], and the technique of frequency multiplier because of the structural problems explained above. The frequency multiplier in [13] has the best performance among the previous frequency multipliers, because the modified DCVSL stage in its edge combiner has better advantages than that of the edge combiners in [11] and [12], i.e., the driving strength of the PU-P and the PD-N in the edge combiner is the same as edge combiner based on the push–pull-stage-based in [11], and the generated output is lower than that of the DCVSL-stage based combing edges technique [12].

Fig.4. consists of prearranged signal stage, overlap canceller, and push–pull stage. Here the dual edge combiner, prearranging, and push–pull stage can have the more number of multiplied clock frequency. The overlap canceller has a property to cancel the signals to be overlapped and it guarantees the stability. While the modified DCVSL stage in the edge combiner in [13] overcomes the weak PU-P usage problem of the DCVSL-stage dependant edge combiner in [12], even though it still has performance conflicts in the structural characteristics of the DCVSL; namely, the positive edge can be obtained only after the negative edge is generated. Hence, even though the HSHR-EC has a greater output capacity and the same PU-P and PD-N driving strengths as the modified DCVSL stage in the edge combiner in [13], the frequency multiplier technique can gives a higher maximum multiplied clock frequency than the frequency multiplier in [13].

FINAL SIMULATIONS:

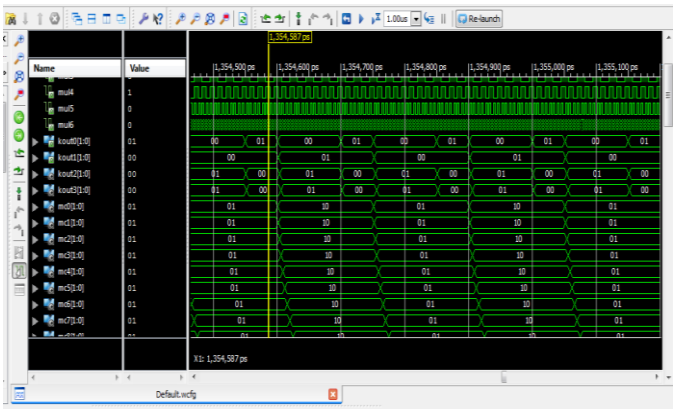

Figure 6(a) Dual edge triggered based edge combiner

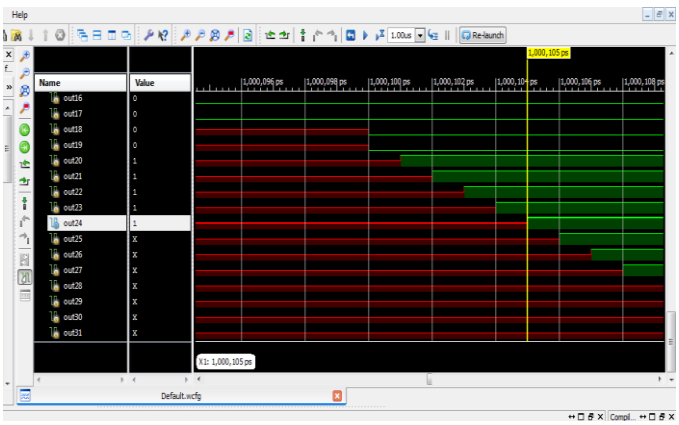

Figure 6(b). Phase generations.

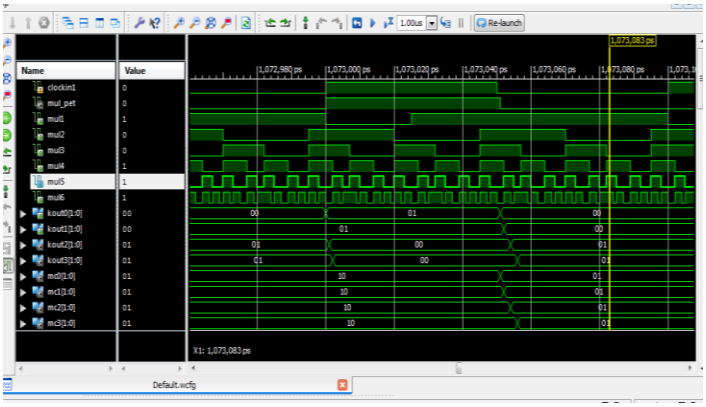

Figure 6(C). **Final expected multiplied frequencies result.**

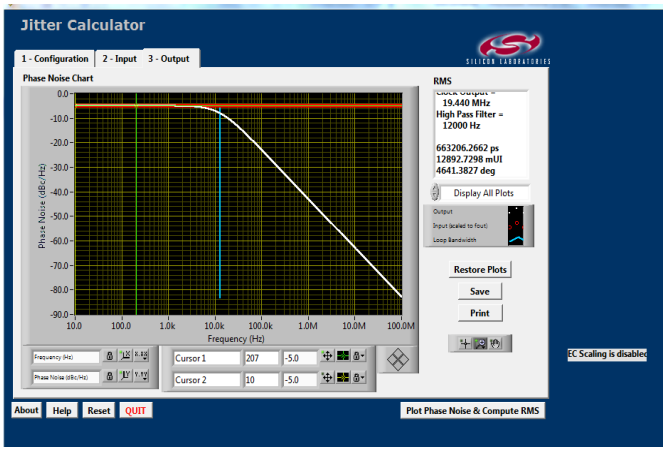

Figure 7. Jitter calculation

The above figures 6(a), 6(b), 6(c) gives the results expected as we see the fig6 (a) gives the dual edge triggered output and fig (b) phase generations to eliminate and find the jitter noise introduced in our device. By the phases generated we can calculate the jitter noise as shown in fig.7. And the proposed frequency multiplier has the multiplication ratios of 1, 2, 4, 8, and 16, and a maximum multiplied clock frequency of 3.3 GHz is generated as shown in fig6(c). The structure of the edge-combiner imposes limitation of having only integer multiplication factor. The integer multiplication factor together with the high reference frequency results in a very low frequency selectivity (channel spacing or frequency resolution.

The frequency multiplier technique is implemented using a 0.14-μm CMOS process technology and the frequency multiplier technique has the ratios of multiplication as 1, 2, 4, 8, and 32, and a maximum multiplied clock frequency of 3.5 GHz. As the operating less frequency of the DLL core is 103 MHz, the multiplied clock frequency has a range from 100 MHz to 3.3 GHz. At 3.3 GHz, the frequency multiplier and the overall DLL-based clock generator consumed 10.6 and 23.4 mW, respectively. The duty-cycle error of the multiplied clock is −0.7%–0.3%, and the rms and the peak-to-peak jitter are 1.65 and 11.6 ps, respectively. The power-frequency ratio increases as the multiplication ratio decreases, because the power consumption of the frequency multiplier almost linearly scales down with decreasing the multiplication ratio, but that of the DLL core is fixed Second, the jitter slightly decreases as the multiplication ratio decreases, because the jitter induced by the delay cell mismatch in Voltage Controlled Delay Line (VCDL) reduces. Instead, jitter is largely dependent on DLL reference clock frequency. Because the VCDL generates one period delay of DLL reference clock, the slope of the clocks in VCDL should be degraded at a low DLL reference clock frequency. Due to this characteristic, jitter increases at a low DLL reference clock frequency. Finally, the duty-cycle error decreases as the multiplication ratio and the DLL reference clock frequency decreases, because the duty-cycle error is deterministic. Thus, the duty-cycle error becomes the largest value at the highest multiplied output clock frequency.

CONCLUSION

The frequency multiplier for a DLL-based clock generator is proposed. The proposed HSHC-EC guarantees high-speed operation of its hierarchical edge-combiner structure and highly reliable operation to its use of an overlap canceller. The optimized pulse generator and the multiplication-ratio control logic are proposed to reduce the delay difference between positive and negative-edge generation paths. Finally, a numerical analysis is performed to validate its performance. The frequency multiplier, which is fabricated using the 0.13-μm CMOS process technology, has the multiplication ratios of 1, 2, 4, 8, and 16, an output range of 100 MHz–3.3 GHz, and a power consumption to frequency ratio of 2.4 μW/MHz's

ACKNOWLEDGMENT

The authors wish to thank SRM University, kattankulathur at Chennai for their support in finding results and acquiring the expected results.

REFERENCES:

[1] S. Paek, W. Shin, J. Lee, H.-E. Kim, J.-S. Park, and L.-S. Kim, "Hybrid temperature sensor network for area-efficient on-chip thermal map sensing," *IEEE J. Solid-State Circuits*, vol. 50, no. 2, pp. 610–618, Feb. 2015.

[2] A. Elshazly, R. Inti, B. Young, and P. K. Hanumolu, "Clock multiplication techniques using digital multiplying delay-locked loops," *IEEE J. Solid-State Circuits*, vol. 48, no. 6, pp. 1416–1428, Jun. 2013.

[3] S. Hwang, K.-M. Kim, J. Kim, S.-W. Kim, and C. Kim, "A selfcalibrated DLL-based clock generator for an energy-aware EISC processor," *IEEE Trans. Very Large Scale (VLSI) Syst.*, vol. 21, no. 3, pp. 575–579, Mar. 2013.

[4] K. Ryu, D. H. Jung, and S.-O. Jung, "A DLL with dual edge triggered phase detector for fast lock and low jitter clock generator," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 59, no. 9, pp. 1860–1870, Sep. 2012.

[5] S. Paek, J. Oh, S.-H. Chung, and L.-S. Kim, "Area-efficient dynamic thermal management unit using MDLL with shared DLL scheme for many-core processors," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, 2011, pp. 1664–1667.

[6] K. Ryu, D. H. Jung, and S.-O. Jung, "A DLL based clock generator for low-power mobile SoCs," *IEEE Trans. Consum. Electron*, vol. 56, no. 3, pp. 1950–1956, Aug. 2010.

[7] T. D. Burd, T. A. Pering, A. J. Stratakos, and R. W. Brodersen, "A dynamic voltage scaled microprocessor system," *IEEE J. Solid-State Circuits*, vol. 35, no. 11, pp. 1571–1580, Nov. 2000.

[8] M. Elgebaly and M. Sachdev, "Variation-aware adaptive voltage scaling system," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 15, no. 5, pp. 560–571, May 2007.

[9] C. Kim, I. C. Hwang, and S.-M. Kang, "A low-power small-area •}7.28-ps-jitter 1-GHz DLL-based clock generator," *IEEE J. Solid-State Circuits*, vol. 37, no. 11, pp. 1414–1420, Nov. 2002.

[10] G. Chien and P. R. Gray, "A 900-MHz local oscillator using a DLL-based frequency multiplier technique for PCS applications," *IEEE J. Solid-State Circuits*, vol. 35, no. 12, pp. 1996–1999, Dec. 2000.

[11] T.-C. Lee and K.-J. Hsiao, "The design and analysis of a DLL-based frequency synthesizer for UWB application," *IEEE J. Solid-State Circuits*, vol. 41, no. 6, pp. 1245–1252, Jun. 2006.

[12] C.-N. Chuang and S.-I. Liu, "A 40 GHz DLL-based clock generator in 90 nm CMOS technology," in *IEEE Int. Solid-State Circuit Conf. Dig. Tech. Paper*, 2007, pp. 178–595.

[13] P. C. Maulik and D. A. Mercer, "A DLL-based programmable clock multiplier in 0.18-μm CMOS with −70 dBc reference spur," *IEEE J. Solid-State Circuits*, vol. 42, no. 8, pp. 1642–1648, Aug. 2007.

[14] C.-C. Wang, Y.-L. Tseng, H.-C. She, and R. Hu, "A 1.2 GHz programmable DLL-based frequency multiplier for wireless applications," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 12, no. 12, pp. 1377–1381, Dec. 2004.

[15] J.-H. Kim, Y.-H. Kwak, M. Kim, S.-W. Kim, and C. Kim, "A 120-MHz-1.8-GHz CMOS DLL-based clock generator for dynamic frequency scaling," *IEEE J. Solid-State Circuits*, vol. 41, no. 9, pp. 2077–2082, Sep. 2006.

[16] J. Koo, S. Ok, and C. Kim, "A low-power programmable DLL-based clock generator with wide-range antiharmonic lock," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 56, no. 1, pp. 21–25, Jan. 2009.

[17] K. Ryu, D. H. Jung, and S.-O. Jung, "A DLL with dual edge triggered phase detector for fast lock and low jitter clock generator," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 59, no. 9, pp. 1860–1870, Sep. 2012.

[18] I. Sutherland, R. F. Sproull, and D. Harris, *Logical Effort: Designing Fast CMOS Circuits*. San Mateo, CA, USA: Morgan Kaufmann, 1999.