RESEARCH ARTICLE OPEN ACCESS

FPGA Based Power Efficient VLSI Architecture for Phase Measurement

Nagavalli vegesna*,K.Yugandhar**,Sai Susmitha Kota***, Jagadeesh Koppisetti****, Akash kakumani*****, Madan Sai NiteeshPulletikurthi******
**(Department of Electronics and Communication Engineering , SRKR Engineering college, Bhimavaram*
Corresponding Author: Nagavalli Vegesna

ABSTRACT
Reconfigurable hardware such as FPGAs (Field Programmable Gate Array) is used since the designers can prefer digital architecture. High performance and reliability are important aspects that are expected in electronic appliances. High-speed serial transceivers of FPGA circuitry do not maintain constant chip latency after each power cycle, reset cycle (or) firmware upgrade. These cause phase differences between the recovered signals. To eliminate these phase shift changes in FPGA, a phase measurement logic core having resolution and precision in the range of a few picoseconds is designed.

Keywords- Field programmable gate array(FPGA), phase polarity, synchronization, systematic sampling, XOR-based phase detector.

Date Of Submission: 25-04-2019 Date Of Acceptance: 07-06-2019

I. INTRODUCTION

The measuring phase is needed in electronic applications in which the synchronous relationship between the signals is required to be preserved. Synchronization of signals between different circuit elements is necessary for accurate results. Hence phase information to calibrate and synchronize signals is important. In some experiments, the preservation of phase relationship between critical signals throughout the experiment runtime is a necessary condition. Nowadays for the implementation of full digital architectures, reconfigurable hardware technologies such as field programmable gate arrays (FPGAs) play a very dominant role. Latency-critical communication link standards used in HEP (High Energy Physics) experiments such as gigabit transceiver and timing-trigger and control system over passive optical network technology are implemented in FPGAs directly[1]. These links carry trigger and timing information needed for time-stamp generation and event building. It is necessary that the latency critical protocols maintain constant phase differences in the recovered signals for the entire experiment's runtime. High-speed serial transceivers of FPGA's do not maintain constant phase shift with each round of power cycle, reset cycle, loss of lock in the transceiver, firmware upgrade, or aging of clock circuitry in phase locked loop(PLL). Logic design is needed for phase monitoring capability to register phase shift changes between the signals in the range of 20-100 ps inside the FPGA circuitry[2]. This helps in extracting the relative phase information and recalibrating the system when needed, to maintain the constant phase relationships. Using the oversampling technique is inadequate to measure a relative phase difference between the two high-frequency clocks inside an FPGA fabric, as the frequency exceeds the maximum limit supported by the fabric(<500MHz). Sampling it externally using the analog-to-digital converter and feeding it back to the FPGA for computation is one of the solutions but it requires additional hardware. Without the use of additional hardware, a phase measurement approach had been proposed using dynamic phase alignment feature of the FPGA PLL, but the resolution is limited to 1/8th of the voltage controlled oscillator (VCO) frequency which is the drawback of this method. So in this paper, we use subsamples collected by the systematic sampling[3] over the XOR-based phase detector (PD) signal. The XOR-based PD introduces the least timing jitter because of the simplicity of its design. This technique can predict the relative phase difference with accuracy, precision and high resolution between clocks of low frequencies and high frequencies in the range of few picoseconds.

II. ARCHITECTURE

The block diagram of FPGA based phase measurement is as shown in fig. 1. The block diagram comprises five blocks namely synchronizer, XOR-based phase detector, duty cycle computation, phase value computation, phase polarity detection.

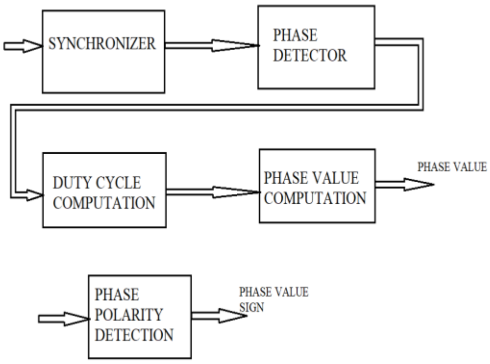


Fig.1. Block Diagram

2.1 Synchronizer

The two high-frequency clock signals inside FPGA must be sampled and synchronized at first. Random sampling holds no synchronous relationship between the sampling clock and the reference clock(CLK1). So we use the systematic sampling method for phase measurement. The necessary condition in systematic sampling is to maintain the synchronous relationship between the sampling clock and the reference clock. For this reason, we synchronize the high-speed digital clocks. The two synchronous high-speed clock signals are sampled with another independent clock of much slower frequency i.e. sampling clock which is referred from [4]. The architecture of the synchronizer is shown in fig.2.

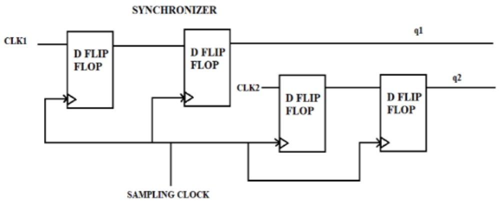


Fig.2.Architecture of synchronizer

2.2 Phase Detector

The synchronized high-speed clocks are fed to the XOR-based phase detector for the detection of the phase difference between the high-frequency clock signals[5-6]. The architecture of the phase detector is shown in fig.3.

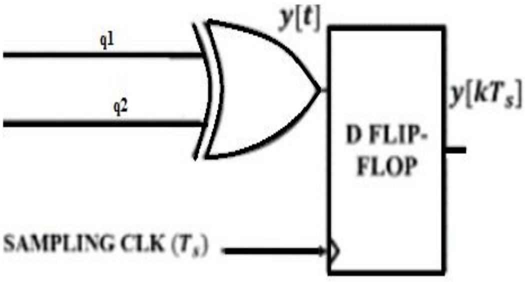


Fig.3. The architecture of XOR-based phase detector

2.3 Duty cycle computation

The number of samples acquired for phase computation is referred to as the sample population size (N). For counting of samples up to N and for the XORed signal duty cycle and the reference clock duty cycle, the counters are used in this block. The architecture of duty cycle computation is shown in fig.4.

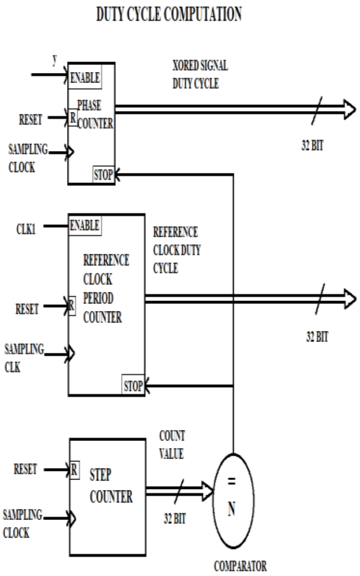


Fig.4. The architecture of Duty cycle computation

2.4 Phase value computation

The computation of phase value involves two operations mainly division and multiplication. For ease of calculation which is mentioned in the works of [7], we use additional signals such as α_i, β_i, K, $\emptyset_i$, $\tilde{\theta}_i$, $\delta[i]$. The architecture of phase value computation is shown in fig.5.

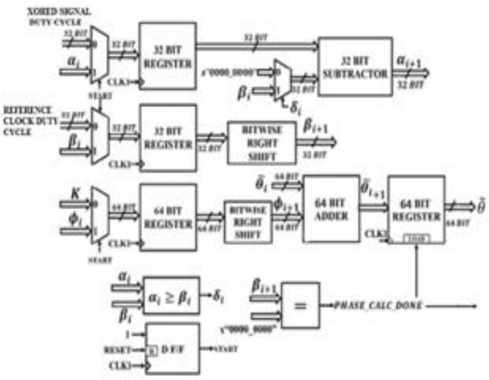


Fig.5. The architecture of phase value computation

2.5 Phase polarity detection

The measured phase difference cannot be distinguished between the leading or lagging phase shift. To resolve the phase polarity ambiguity, the reference clock (CLK1) is used to sample the phase shifted clock (CLK2) to determine whether the phase magnitude (θ̃) is positive (leading) or negative (lagging). The architecture of phase polarity detection is shown in fig.6. Bank skew cancellation can be measured in this block.

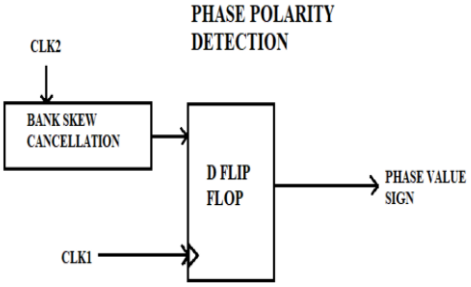


Fig.6. The architecture of phase polarity detection

III. RESULTS

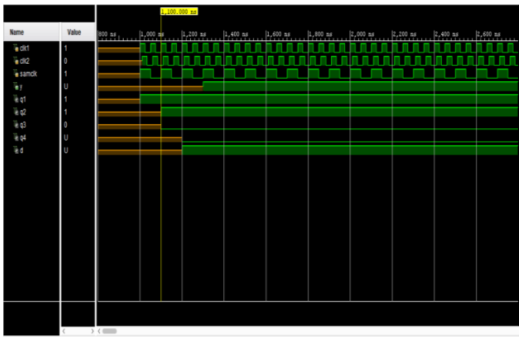


Fig.7 Performance analysis of synchronizer and phase detector

From fig.7 we can see the input and output waveforms of synchronizer and phase detector blocks. CLK1 of 50nsec, CLK2 0f 50nsec with 10nsec offset value and samclk of 100nsec are applied as inputs. During the rising edge of samclk, the input CLK1 is buffered to q1 and with some delay introduced by D flip-flop, q1 is buffered to q2. Similarly, during the rising edge of samclk, the input CLK2 is buffered to q3 and with some delay introduced by D flip-flop, q3 is buffered to q4. d shows the output waveform of q2 xor q4. And finally, with the rising edge of samclk, d is buffered to y.

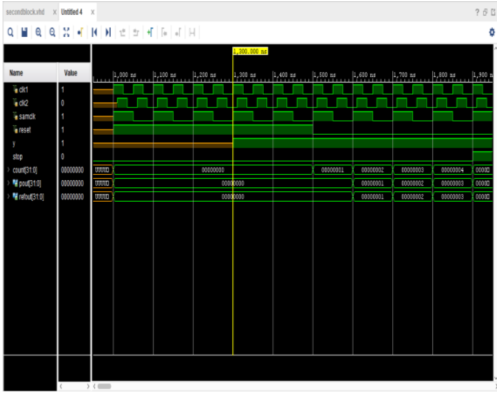
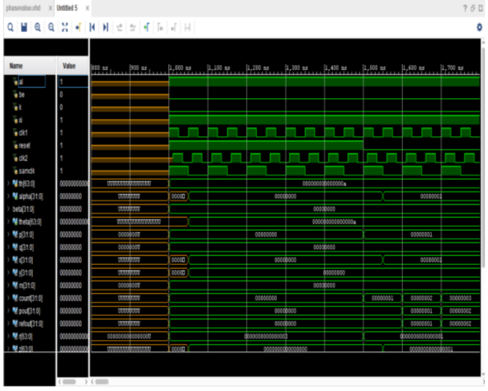


Fig.8. Performance analysis of duty cycle computation block

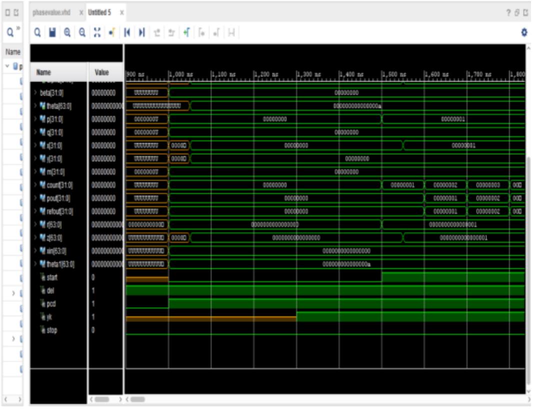
From fig.8., we can observe the input and output waveforms of the duty cycle computation block. The output of the phase detector is fed to this block with the same time periods of CLK1, CLK2, and samclk.

The counter(count) counts until the reset is 0 and it stops when the reset is 1. Now the comparator compares the count value to the population size(N) when the condition is satisfied the stop signal gets activated.

Now the refout and pout outputs get counted when reset is 0 and stop is 0. When the stop signal becomes high the counting gets halted. When the reset becomes high, the counter is set to 0.



(a)



(b)

Fig.9. Performance analysis of phase value computation block

(a) and (b) indicates the input and output waveforms

The outputs of counters in the duty cycle computation block are fed as input to the multiplexers and the external inputs al, be,k, xi, th are given as 1,0,0,1 and 00000000000a as shown in Fig,9. Registers receive the inputs from the multiplexers with the select line(start) accordingly which is activated by the comparator. The inputs of registers get processed by the respective subtractor, bitwise right shift, and adder. Later the final phase value is obtained from the comparator.

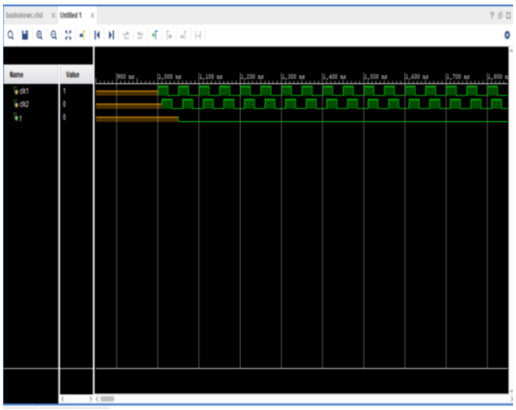


Fig.10. Performance analysis of phase polarity detection

During the rising edge of CLK1, the CLK2 is buffered to y which indicates the signal is lagging (0) or leading(1) as shown in Fig.10.

IV. CONCLUSION

Implemented the simulation of a sensitive phase detection logic core for FPGA, having precision, accuracy, and resolution in the range of few picoseconds. This simulation can be utilized in FPGA as a monitoring device of the phase relationship between digital clock pulses, without any additional circuitry. This design allows designers to modify different components for more lustiness in the design, like replace XOR- based phase detector with other comparators.

REFERENCES

[1]. Liu, X., Deng, Q. X., & Wang, Z. K. (2014). Design and FPGA implementation of high-speed, fixed-latency serial transceivers. IEEE Transactions on Nuclear Science, 61(1), 561-567.

[2]. Hidvégi, A., Geßler, P., Kay, H., Petrosyan, V., Petrosyan, G., Petrosyan, L., ... & Bohm, C. (2013, October). FPGA based phase detector for high-speed clocks with pico-seconds resolution. In 2013 IEEE Nuclear Science Symposium and Medical Imaging Conference (2013 NSS/MIC) (pp. 1-4). IEEE.

[3]. Choi, H., Gomes, A. V., & Chatterjee, A. (2010). Signal acquisition of high-speed periodic signals using incoherent sub-sampling and back-end signal reconstruction algorithms. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 19(7), 1125-1135.

[4]. Lu, S. J., Siqueira, P., Vijayendra, V., Chandrikakutty, H., & Tessier, R. (2013). Real-time differential signal phase estimation for space-based systems using FPGAs. IEEE Transactions on Aerospace and Electronic Systems, 49(2), 1192-1209.

[5]. Guntoro, A., Zipf, P., Soffke, O., Klingbeil, H., Kumm, M., & Glesner, M. (2006, March). Implementation of realtime and highspeed phase detector on FPGA. In International Workshop on Applied Reconfigurable Computing (pp. 1-11). Springer, Berlin, Heidelberg.

[6]. Rong, L., Zheng, W., Wei-Min, P., Guang-Wei, W., Hai-Ying, L., Peng, S., & Ri-Hua, Z. (2009). FPGA-based amplitude and phase detection in DLLRF. Chinese physics C, 33(7), 594.

[7]. Mitra, J., & Nayak, T. K. (2017). An FPGA-based phase measurement system. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 26(1), 133-142.

Nagavalli Vegesna" FPGA Based Power Efficient VLSI Architecture for Phase Measurement" International Journal of Engineering Research and Applications (IJERA), Vol. 09, No.05, 2019, pp. 63-66