International Journal of Electrical, Electronic and Communication Sciences
ISSN: 2517-9438
Vol:2, No:6, 2008

Low Jitter ADPLL based Clock Generator for High Speed SoC Applications

Moorthi S.[1] Meganathan D.[2] Janarthanan D.[2] Praveen Kumar P.[2] J. Raja paul perinbam[3]

***Abstract* —** An efficient architecture for low jitter All Digital Phase Locked Loop (ADPLL) suitable for high speed SoC applications is presented in this paper. The ADPLL is designed using standard cells and described by Hardware Description Language (HDL). The ADPLL implemented in a 90 nm CMOS process can operate from 10 to 200 MHz and achieve worst case frequency acquisition in 14 reference clock cycles. The simulation result shows that PLL has cycle to cycle jitter of 164 ps and period jitter of 100 ps at 100MHz. Since the digitally controlled oscillator (DCO) can achieve both high resolution and wide frequency range, it can meet the demands of system-level integration. The proposed ADPLL can easily be ported to different processes in a short time. Thus, it can reduce the design time and design complexity of the ADPLL, making it very suitable for System-on-Chip (SoC) applications.

***Keywords*—** All Digital Phase Locked Loop (ADPLL), System-on-Chip (SoC), Phase Locked Loop (PLL), Very High speed Integrated Circuit (VHSIC) Hardware Description Language (VHDL), Digitally Controlled Oscillator (DCO), Phase frequency detector (PFD) and Voltage Controlled Oscillator (VCO).

I. Introduction

In data transmission systems, the transmitter and the receiver must be synchronized to access transmitted data. Phase-Locked Loop (PLL) circuits have been widely used to achieve such synchronization. A typical PLL circuit receives a reference input and performs a feedback control operation to adjust the output signal in phase with the reference signal. An analog PLL circuit continuously tests the output of a voltage controlled oscillator (VCO) through a feedback loop, and when the output of VCO drifts away from the incoming signal, an error voltage is generated to pull the VCO back into synchronization with the reference signal. However, analog PLLs have some drawbacks including noise coupling, signal shading and power supply noise effects. Integration of analog parts in newer deep submicron technologies is much tougher and additionally complicated because the usable voltage ranges decrease with every new integration step.

If it is possible to realize a PLL as 'pure' digital circuit, no effort would be needed to scale the device for every new CMOS process technology and furthermore, [1] the full integration advantage of a digital circuit would be feasible. No special silicon process and test technology known as 'mixed mode' would be needed as well.

The most important use of Phase Locked Loop Circuit is to recover the clock from a given data stream [2]. The data coming into recovering circuit is jittered due to inter-symbol interference and other undesirable effects that happen in the real world such as power supply noise, component tolerance and any added noise at the Voltage Controlled Oscillator (VCO) input. This means that the received data edges (i.e., zero crossing, transitions, etc.) are not happening at a fixed time period but are varying around the ideal. The PLL, being a narrow-band system, will tend to average out these variations and produce a clock, which is closer to the ideal. Therefore low jitter performance is essential for any PLL.

To cover a wide range of applications with different frequency specifications, and to allow the adaptation for different semiconductor technologies, the design of the ADPLL should be portable. This means it has to rely on standard cells only and must not contain library specific components that have to be redesigned when the technology changes. The design is adapted solely by readjusting parameters that define its performance [3]. Thereby the redesign of the ADPLL components is avoided. A hardware description language (HDL)-based design of the ADPLL will be beneficial as it permits design debugging at a high level, and enables system simulation in a digital simulator including the ADPLL. Furthermore, an HDL offers an easy and fast facility to modify the parameters of a design.

The paper is organized as follows: Section II introduces the proposed ADPLL architecture and describes about the implementation of individual blocks. Section III deals with jitter and its performance. Section IV shows the implementation and measurement results and finally, section V provides the conclusion.

II. Proposed Adpll Architecture

A. ADPLL

Digital implementation of phase detector, loop filter and frequency divider are presented [4]. These functional blocks can easily be made portable. The digital implementation of ADPLL requires design of a high resolution wide range DCO and a small dead zone PFD. The main difficulty to realize a portable ADPLL is to implement a portable DCO, which has a high frequency resolution and a small jitter.

[1] Lecturer, Dept of EEE, National Institute of Technology, Trichy – 15, TamilNadu, India. (corresponding author) phone: +91-94432 10281; **e-mail: srimoorthi@nitt.edu**).
[2] Dept of Electronics Engineering, Madras Institute of Technology, Chennai – 44, TamilNadu, India.
[3] Dept of Electronics and Communication Engineering, CEG, Anna University, Chennai – 25, TamilNadu, India.

Figure 1 shows the block diagram of the proposed ADPLL. There are four major building blocks, namely the phase/frequency detector (PFD), the Digitally Controlled Oscillator (DCO), the ADPLL controller, and the digital loop filter.

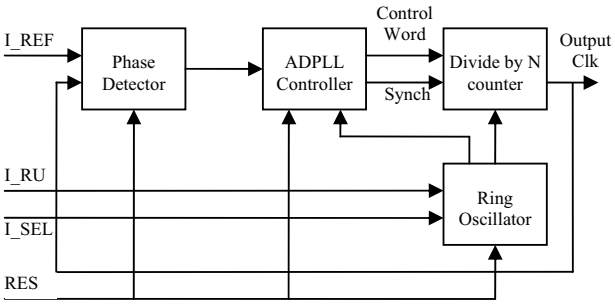

Fig. 1 Proposed ADPLL Block Diagram

The PFD detects the frequency difference and phase error between an input clock signal and the generated DCO signal, and delivers the UP and DOWN signals. Accordingly the ADPLL controller generates control word and sync signal. The DCO frequency gets updated with respect to these signals from ADPLL controller.

B. Phase Frequency Detector

The schematic of the PFD is shown in Figure 2. Due to the PFD's dead zone and the reference clock noise, the DCO control code has small variations in the phase acquisition mode. In order to reduce the jitter, the proposed ADPLL uses a pulse amplifier to diminish such kind of effects. Therefore the PFD has reduced dead zone. When the divided output clock (FB_CLK) leads the reference clock (IN_CLK), flagD generates a low pulse and flagU remains high. Oppositely, when FB_CLK lags IN_CLK, flagU generates a low pulse and flagD remains high. The ADPLL controller will be triggered by those signals. Since the ADPLL controller only needs to know that the FB_CLK leads IN_CLK or FB_CLK lags IN_CLK, the digital pulse amplifier can effectively minimize the dead zone of the PFD. The schematic of the digital pulse amplifier is given in [3]. It uses the cascaded two-input AND architecture to increase the pulse width of OUTU and OUTD. The digital pulse amplifier enlarges the phase error between IN_CLK and FB_CLK, thus, the following D-flip-flops can detect it. When the phase error is less than 50 ps, both flagU and flagD will remain in high, and no trigger signal is sent to the ADPLL controller.

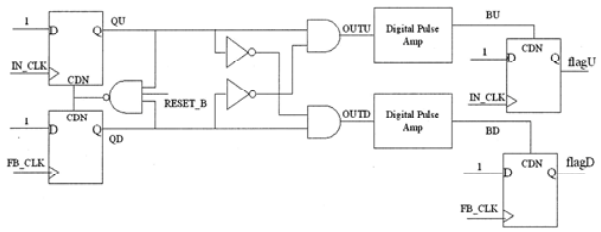

Fig. 2 Phase Frequency Detector

C. ADPLL Controller

The ADPLL controller contains a up/down counter that uses the high rate clock from the ring oscillator to measure the length of one reference period. Output from the up/down counter is a measurement of the previous reference period in number of oscillator periods. The measurement is based on the error signal from the PFD. Also, 1-bit signal 'sync' generated at every transition from low to high of the reference clock. The signal 'sync' is used in the output generation counter to synchronize output clock edges to clock edges of the reference clock.

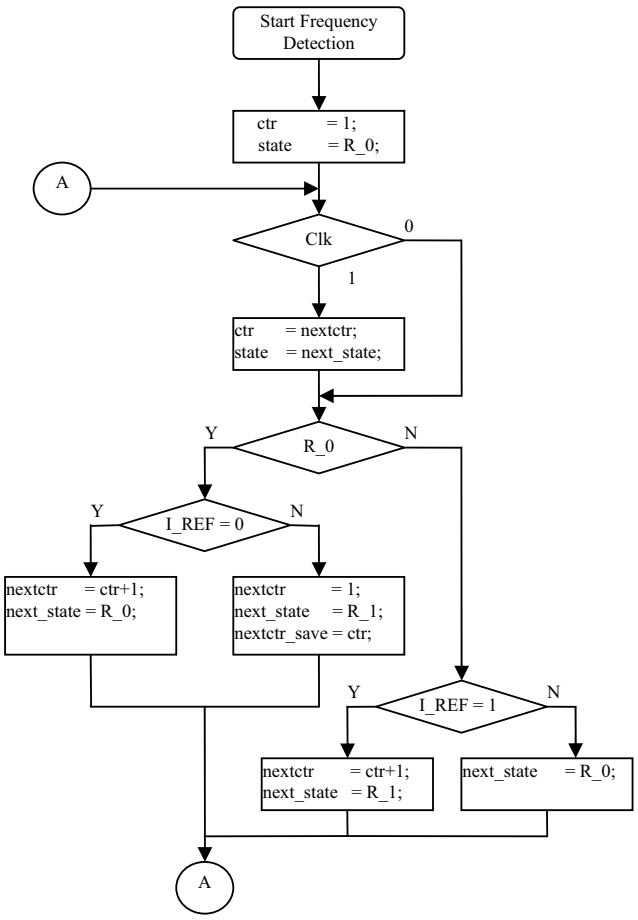

Fig. 3 ADPLL Controller

The flow chart describing the operation of ADPLL controller is shown in the Figure 3. When the ref clock is zero then the state is zero (R_0). When the ref clock is one then the state is one (R_1). Whenever the state of high frequency clock changes the control (ctr) value is changed to the next ctr. The next ctr value is thus incremented during each positive edge of high frequency clock. When the reference clock changes state from state R_0 to R_1, the control value is saved. This control value is sent to the DCO.

D. Ring Oscillator

The oscillator is implemented as a ring oscillator with one NAND-gate [5] as inverter and with variable delay elements as shown in Figure 4. The NAND-gate is to enable power down by shutting down the oscillator. A frequency divider is inserted to give ability to select between 4 clock-rates *f*, *f*/2, *f*/4 and *f*/8, where *f* is the output frequency from the ring oscillator.

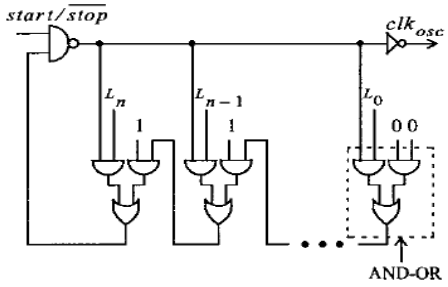

Fig. 4 Ring Oscillator Structure

In general a ring oscillator with adjustable frequency consists of a variable number of delay elements (d_e) and a frequency selector (f_s). A control word (*L)* is applied to the selector which defines the number of the delay elements in the ring and thereby the frequency f_{OSC} of the ring oscillator. This frequency is given by

$$f_{osc} = 1/2(L.d_e + f_s) \quad (1)$$

The number of delay elements determines the frequency of the ring oscillator. Also to get 50% duty cycle only one delay element must be enabled at any time. Selection of the last delay element gives the lowest frequency and vice versa. The number of bits in the control word 'L' could be modified to get extended frequency range, thereby portability is achieved. Therefore L value is set for synthesize flow, so that portability is achieved by only changing it for different frequencies which is also technology independent [5].

E. Digitally Controlled Oscillator

The Divide by N counter DCO is used with slight modifications (with additional input sync for accuracy). As shown in the Figure 1 the Divide by N counter DCO is used to scale down the signal generated by a high frequency oscillator operating at a fixed frequency [1]. The N-bit parallel output signal of a digital loop filter is used to control the scaling factor N of the Divide by N counter DCO.

The output generation counter produces output clock using measured "length" of the reference period and the "sync" signal. It counts up by two times the multiplication factor, each oscillator period. When the output generation counter has reached the measured "length" of the reference period, the output clock changes polarity and twice the value of multiplication factor is subtracted from the counter. Each time a sync pulse arrives, the counter is reset and next output is set high.

III. Jitter

The short-term variation of a signal with respect to its ideal position in time is termed as Jitter. This deviation in a clock's output transition from its ideal position can negatively impact data transmission quality [6]. The jitter can be expressed as, timing jitter, period jitter and Long Term (Accumulation Jitter), where long term jitter and accumulation jitter are defined as the deviation of clock cycle. The period jitter is defined for one clock cycle whereas long term jitter is defined for multiple clock cycle. Due to its random nature, this jitter can be measured peak-to-peak or by Root Mean Square (RMS). Defining the clock rising-edge crossing point at the threshold V_{TH} as $T_{PER}(n)$ and n is the time domain index, the J_{PER} can be expressed as:

$$J_{PER} = T_{PER}(1) - T_0 \quad (2)$$

Where T_0 is the period of the ideal clock cycle.

In many cases, other signal deviations, like signal skew and coupled noise are combined and labeled as jitter. Deviation (expressed in ±*ps*) can occur on either the leading edge or the trailing edge of a signal. Excessive jitter can increase the bit error rate (BER) of a communications signal by incorrectly transmitting a data bit stream. In digital systems, jitter can lead to a violation of timing margins, causing circuits to behave improperly. Accurate measurement of jitter is necessary for ensuring the reliability of a system. Different techniques have been reported for the design and implementation of de-jitter circuits or low jitter clock recovery circuits, for example, modifying the filter design to narrow the PLL bandwidth and make the phase noise at the DCO input as low as possible, reducing power supply noise, eliminating ground bounce, using a voltage Controlled Crystal Oscillator (VCXO), using dual phase frequency detector and using charge pump as phase frequency detector. But the problem with most of the design is that we have to use an oscillator that has the same reference frequency [7].

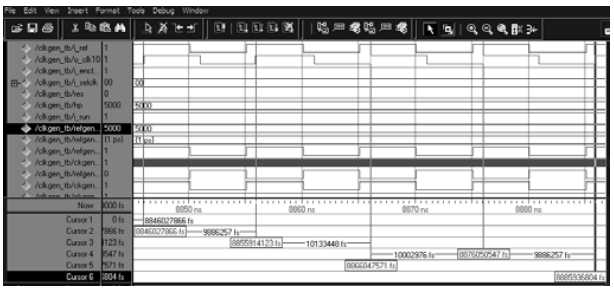

Fig. 5 ADPLL Output at 100MHz with cycle to cycle jitter of 164ps

Recent analysis shows that the jitter increases due to dead zone in PFD and low resolution of the DCO. The most important design consideration of the ADPLL is how to design a high-resolution wide-range DCO and a small dead zone PFD. From the analysis of jitter it is found that the major jitter source of the ADPLL comes from the DCO and PFD, the

high-resolution DCO and the small dead zone PFD can reduce the jitter of ADPLL significantly.

IV. Experimental Results

The proposed ADPLL is designed by cell-based design flow. The entire architecture is described using Hardware Description Language (HDL).

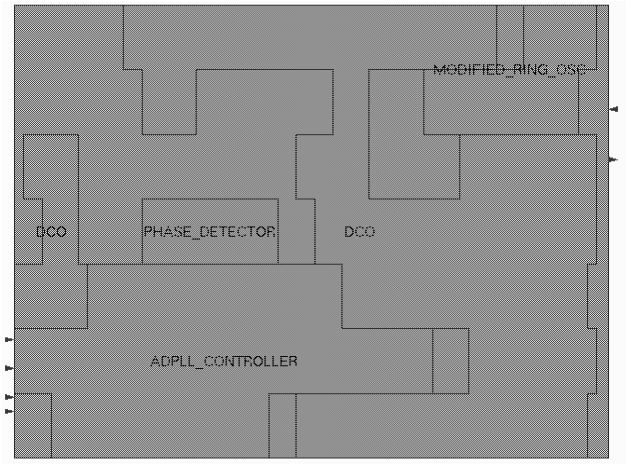

Fig.. 6. ADPLL Layout

The entire design has been implemented using cadence digital tools. The simulation and synthesis (Front end flow) were carried out using Nc-Sim and RTL Compiler respectively. All the synthesis reports have been obtained in the 90 nm CMOS technology library. The proposed ADPLL consumes 1.976 mW of power @ 100MHz with 1V power supply. The worst case lock in time of the PLL is 14 reference cycles. Figure 5 shows the ADPLL output at 100MHz with $T_{c\ to\ c}$ of 164 ps measured using markers in MODEL SIM. The physical design of ADPLL is carried out using First Encounter. The physical view of proposed ADPLL is shown in figure 6. The PLL lock range is (10 to 200) MHz for the centre frequency of 4GHz. The ring oscillator centre frequency varies from 500MHz to 4GHz. The variable centre frequency is achieved by using equation (1) by changing the number of active delay elements. The phase error of ADPLL output is increasing with frequency above 200MHz.

Table 1. Design Summary

Process Technology	TSMC 180nm CMOS [8]	TSMC 90nm CMOS
PLL Lock Range	0.1 to 282 MHz	(10 – 200)MHz
Ring Osc Freq Range	-	(0.5 – 4) GHz
Lock-in Time	< 5 cycles	< 14 cycles
Cycle to Cycle Jitter	190 ps @ 100MHz	164ps
Period Jitter	126 ps	100ps
Area	2501 μm^2	1622 μm^2
Power Dissipation	6.4 mW	1.976 mW

V. Conclusion

This paper proposes an efficient ADPLL architecture with low jitter suitable for high speed SoC applications. The ADPLL is designed using standard cells and described by Hardware Description Language (HDL). The ADPLL has lock range of (10 to 200) MHz with worst case frequency acquisition in 14 cycles. It has cycle to cycle jitter of 164ps at 100MHz. The complete design features are summarized in the Table 1. The ADPLL can be implemented with standard cells, and it has good portability over different processes. The table I compares the conventional ADPLL [8] with the proposed one. The proposed ADPLL can reduce design time and circuit complexity. Therefore it is very suitable for System-on-Chip applications.

Acknowledgment

The Authors express their gratitude to the Technical Education Quality Improvement Programme (TEQIP) funded by World Bank for partially supporting this work and the authorities of NIT - Trichy and Anna University - Chennai for their constant support in completing this work.

References

[1] R.E. Best, **"**Phase-Locked Loops Principle, Design and Applications". New York; McGraw-Hill, Feb 2000.

[2] T. Hsu, B. Shieh and C. Lee, "An all-digital phase-locked loop (ADPLL)-based clock recovery circuit," IEEE Journal of Solid-State Circuits, vol. 34, pp.1063-1073, Aug 1999.

[3] Ching-Che Chung, Duo Sheng and Chen-Yi Lee, "An All-Digital Phase-Locked Loop with High-Resolution for SoC Applications", VLSI Design Automation and Test, 2006 International Symposium, pg. 1 – 4, April 2006.

[4] E. Mokhtari and M.Sawan,"CMOS High Resolution All-Digital Phase Locked Loop", 46th IEEE International symposium on Circuits and systems, pg 221-224, Vol 1, 27-30 Dec, 2003.

[5] Read Stefo, Jorg Schreiter, Jens-Uwe SchliiOler and Rene SchiiRny, "High Resolution ADPLL Frequency Synthesizer for FPGA- and ASIC-based Applications", Proceedings. IEEE International Conference, pg 28 – 34, Dec 2003.

[6] Masami Kihara, Sadayasu Ono and Pekka Eskelinen, "Digital Clocks for Synchronization and Communications", John Wiley & Sons, 2003.

[7] A. Telbal, J. M. Noras2, M. Abou El Elal, and B.AlMasharyl "Jitter minimization in Digital Transmission using dual phase locked loops", 17th International Conference on Microelectronics, pg 270 – 273, Dec 2005.

[8] B. Sai Pramod Reddy, N. Krishnaprasad, S. Moorthi, J. Raja paul perinbam, "An All Digital Phase Locked Loop for Ultra fast locking", Proceedings of National Conference on Emerging Trends in Engineering and Technology, April 2008.

About The Authors

S. Moorthi received his B.E (EEE) degree from University of Madras and ME (Applied Electronics) from PSG College of Technology, Coimbatore. He currently works towards his PhD in the area of VLSI for Communication Circuits. His area of interest includes VLSI, Embedded Systems and Signal processing. He has served as Lecturer in the Department of Electronics Engineering, MIT Campus of Anna University, Chennai for three years. He is currently a Lecturer in the Department of Electrical and Electronics Engineering, National Institute of Technology, Tiruchirappalli, India. He is a member of professional bodies like IEEE, ISTE and IETE.

D. Meganathan received his AMIE degree from Institution of Engineers, India and ME (Applied Electronics) from PSG College of Technology, Coimbatore. He currently works towards his PhD in the area of VLSI for Data Converter Circuits. His area of interest includes VLSI and analog circuit design. He serves as Lecturer in the Department of Electronics Engineering, MIT Campus of Anna University, Chennai. He is a member of professional bodies like IE and IETE.

Janarthanan D received his BE (ECE) degree from Madras Institute of Technology Campus of Anna University, Chennai during April 2007. He is currently working in Wipro Technologies. His area of interest includes Digital VLSI circuits, Signal Processing. He has undertaken several projects in the area VLSI Technology and won in the Altera Design Contest 2006. He is the student member of IEEE and IETE.

Praveen Kumar P received his BE (ECE) degree from Madras Institute of Technology Campus of Anna University, Chennai during April 2007. He is currently working in Infosys Technologies. His area of interest includes Digital system design, Signal Processing. He is the student member in IEEE and IETE.

Dr J Raja paul perinbam, Professor, ECE, CEG, Anna University, Chennai-25 has wide experience in teaching VLSI and Embedded systems related subjects at College of Engineering, Guindy. He has undertaken several Projects and developed embedded hardware involving microcontrollers, Digital Signal processors and FPGAs. His area of research is on low power VLSI design, VLSI routing algorithms and embedded architecture.