SERDES TRANSCEIVER DESIGN FOR

MULTICORE COMMUNICATION

A Thesis Presented to the Faculty
Of
Nile University

In Partial Fulfillment
of the Requirements for the Degree
of Master of Science
in Micro-electronics System Design

By
Sally Safwat Moawad Amin
August 2011

CERTIFICATION OF APPROVAL

SERDES TRANSCEIVER DESIGN FOR

MULTICORE COMMUNICATION

by

Sally Safwat Moawad Amin

Dr Yehea Ismail Associate Professor at Nile University	Date
Dr Maged Ghoneima Assistant Professor at Nile University	Date
Dr Hani Fikry Ragai Professor at Ain Shams University	Date

To Mum and Dad

Acknowledgment

It is a pleasure to thank those who made this thesis possible.

First, I would like to express my gratitude to my profound advisor Prof. Yehea Ismail for his expert advice and criticism. I am grateful to him for all the efforts he has made to help me complete my first paper from which this thesis started.

I would like to thank my co-advisor, Dr. Maged Ghoneima, who has helped me in the completion of this thesis. He read my drafts with incredible strictness and carefulness. His review and edits were important for this thesis to achieve its present end.

My special thanks go to my colleague Ezz El-Din Hussein who has participated in my research work and has invaluable contribution to my thesis. A special word of thanks must go to Amr El-Sherif for his CAD tools support and his patience to solve tools problems, which were the main reason for having results. I would like to thank my colleague Amr Lotfy for helping me in understanding the basic concept of all digital PLLs from which I started my innovation.

I wish to thank my respectable academic Professor Rafik Guinidi, for his valuable instructions and suggestions in the Microelectronics System Design (MSD) master program.

I am indebted to all my colleagues at Nano-electronics Integrated System Center (NISC). The friendly environment they have created encouraged me to work in the Lab and made the experience enjoyable.

Above all, I owe my deepest gratitude to my Mother and Father, for always being supportive of my academic choices, even though it meant more stress and worries for them.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

ADCDR	All Digital Clock and Data Recovery
ADPLL	All Digital Phase Locked Loop
BBADPLL	Bang-Bang All Digital Phase Locked Loop
BBPD	Bang-Bang Phase Detector
BPD	Binary Phase Detector
CDR	Clock and Data Recovery
CPPLL	Charge Pump Phase Locked Loop
DCO	Digital Controlled Oscillator
DETFF	Double Edge Trigger Flip Flop
DFE	Decision Feedback Equalizer
DLF	Digital Loop Filter
FFE	Feed Forward Equalizer
PD	Phase Detector
PLL	Phase Locked Loop
SerDes	Serializer and Deserializer
SiP	System in Package
SoC	System on Chip
TDC	Time to Digital Converter
TSV	Through Silicon Via
VCO	Voltage Controlled Oscillator

Abstract

System in package (SiP) is a technology involving multi-core integration in 3D ICs. SiP has evolved as an attractive alternative to system on chip (SoC). Three dimensional ICs do not only allow more cores to be integrated on a single chip, but also provide potential performance advances, because each core can be connected to a larger number of adjacent cores. SiP offers the opportunity to stack adjacent cores on adjacent layers and connects them vertically. Therefore, long interconnects in 2D ICs are replaced by a shorter set of interconnects in addition to vertical through silicon via (TSV) interconnects in 3D ICs. However, 3D ICs still suffer from wiring complexity and routing congestion due to the parallel buses connecting hundreds of modules. Moreover, 3D ICs suffer from the increase in the power density resulting from stacking cores on top of each other. The increase in the power density results in high peak temperatures, which can cause IC failure. According to the above scenario, conventional parallel bus communication schemes are no longer suitable in terms of area, power, and performance. One promising solution is to replace the parallel bus by a serial link with serialization and deserialization (SerDes) transceiver in 2D and 3D ICs.

Serial links have been used for decades in off-chip communication between ICs, due to the limited number of I/O pins. A similar problem exists for on-chip communication in current designs due to the limited number of metal tracks and other timing errors due to skew and jitters on parallel buses that make the synchronization at the receiver difficult and limits the bandwidth. However, the requirements for the design of the on-chip serial link are different from off-chip. The on-chip interconnects are very lossy, compared to the lossless off-chip transmission lines.

This thesis presents the design and analysis of three different architectures of an all-digital

SerDes transceiver for three different signaling schemes. Two new signaling schemes multiplex the clock and the data into a three-level signal. The new-three level signal enables the receiver to recover the clock from the data using simple circuitry. This eliminates the need for power hungry blocks at the receiver, such as the clock and data recovery (CDR) at the receiver or analog phase locked loop (PLL) at the transmitter. Moreover, this technique is insensitive to jitter accumulated during signal propagation or at the receiver input, because the clock signal is extracted from the multiplexed data stream. Hence, timing errors in the received signal will affect both the data and the extracted clock, and accordingly the data will be sampled correctly. The SerDes transceivers for the three schemes were implemented for a 3mm long lossy on-chip transmission line in a 65nm TSMC CMOS technology. In addition, this thesis presents the design and analysis of a bang-bang all digital PLL, which was used to provide the high frequency clock at the transmitter.

The implementation of a bang-bang all digital phase locked loop (BBADPLL) in a frequency synthesizer has proven a reduction in the power and, area. This due to the elimination of complex, and power and area hungry blocks, such as an analog to digital converter (ADC) or a time to digital converter (TDC), which is used to convert the average analog output of the phase detector (PD) to digital bits for the digital controlled oscillator (DCO). However, the non-linearity of the BBADPLL makes the traditional Laplace transform used in modeling the PLL invalid. Hence, there are a serious design challenges in managing the tradeoffs between the tracking bandwidth, jitter, and lock time. In this thesis, a new design methodology that adjusts the digital loop filter (DLF) coefficients according to the digital controlled oscillator (DCO) frequency step is presented. The DLF coefficients are used to control the closed loop dynamics of the PLL. Useful expressions that model the DLF are presented for the design and optimization of the programmable DLF coefficients.

Chapter 1

Introduction

1.1. Motivation

In the past, scaling resulted in performance improvements, as well as reductions in power, area, and cost. Today scaling alone does not ensure improvement of all four items. Figure 1.1 shows the International Technology Roadmap for Semiconductors (ITRS) prediction [1] for the gate and interconnect delay versus technology. As technology scales, the interconnect delay increases, while the gate delay decreases. Thus, computational blocks become faster, but communication between these on-chip blocks become slower. Hence, the requirements for on-chip communication links become more challenging in deep submicron technologies. Therefore, the research effort becomes directed towards the analysis and the design of on chip communication links [2], [3],[4].

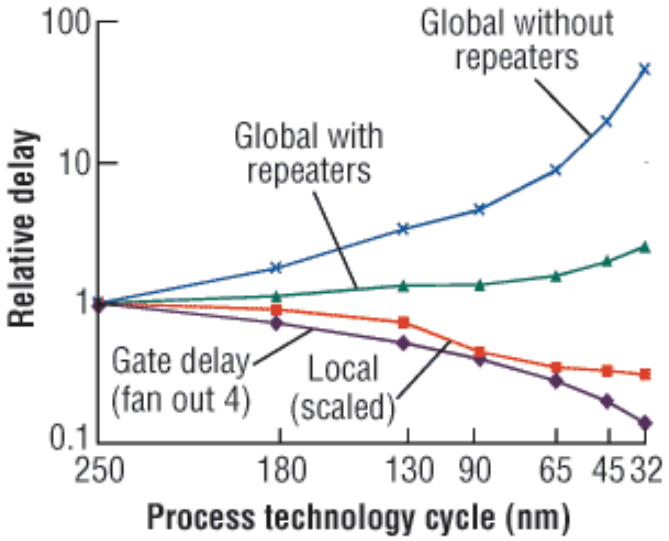


Figure 1.1 The relative delay of the transistor, local, and global interconnect wires plotted versus process technology [1]

The advancement in CMOS technology scales down transistors, allowing more transistors to be placed on a single chip. System on chip (SoC) provides a path for continued improvement in performance, by integrating hundreds or possibly thousands of cores on a single chip. However, the rate of increase in performance is not the same as that of the number of processing elements, as shown in Figure 1.2. A popular example of the change in performance scaling was demonstrated by the Pentium 4, which was first implemented in the same technology as the Pentium 3 (0.18-micron). Even though the Pentium 4 had 50% more transistors than the Pentium 3, its performance, based on the SPECint 2000, was only 15% greater. This situation introduced Moore's Gap at 2002 as shown in Figure 1.2. Moore's Gap is defined as the increasing difference between the exponentially growing number of transistors on a single chip and the delivered performance of the chip [5].

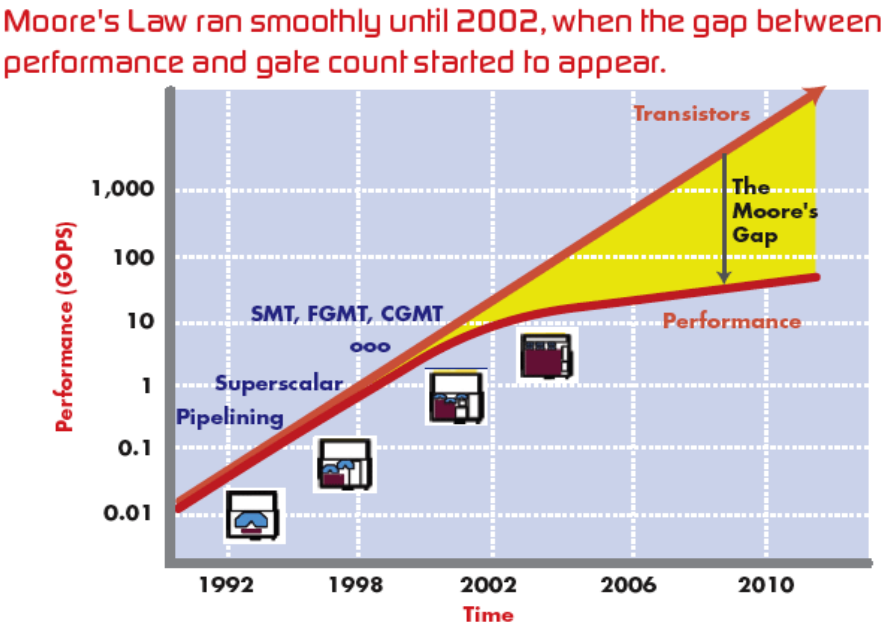


Figure 1.2 Moore's gap; gap between the performance and the number of transistors

1.2. System in Package (SiP)

System in package (SiP) has evolved as an attractive alternative to SoC for electronics integration in three-dimensional integrated circuits (3D ICs) rather than two-dimensional (2D) ICs [6]. Three-dimensional ICs consists of multiple layers of active devices that have the potential to enhance chip performance, functionality, and device packing density. Figure 1.3 shows a 3D IC consisting of three silicon layers vertically connected using through silicon vias (TSVs). Each layer can potentially have multiple processing cores (IPs) and memories (MEM) [7].

Three-dimensional ICs do not only allow more cores to be integrated on a chip, but also provide potential performance advances, because each core can be connected to a larger number of adjacent cores. SiP offers the opportunity to vertically stack cores, and connect them vertically rather than horizontally. Therefore, long interconnects in 2D ICs are replaced by a combination of shorter interconnects and vertical TSVs in 3D ICs. This enables faster and more power efficient inter-core communication across multiple silicon layers [7]. Moreover, data traffic between the processor chip and the memory chip in SiP will have reduced noise due to the shorter path length enabled by SiP technology. There is a tremendous amount of recent research efforts done on 3D ICs such as [8], [9], [10].

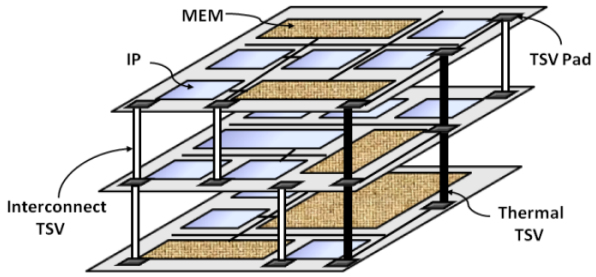


Figure 1.3 3D ICs multilayer connected TSV [7]

However, 3D IC technology faces thermal challenges due to higher power densities and routing congestion due to the distribution of TSV pads on each layer. Vertically stacking cores result in an increase in the power density and peak temperatures that may cause IC breakdown. Thermal path TSVs were proposed as a mean to dissipate heat from the core of the chip to the heat sink, as shown in Figure 1.3. Serialization of the vertical TSVs in 3D ICs can be considered a promising solution for better thermal distribution, resulting in lower peak temperatures, as well as more efficient layout that reduces routing congestion [7].

1.3. Parallel vs. Serial On-Chip Communication

In parallel communication, the physical channels are parallel line buses, which consist of parallel wires; one wire for each bit as shown in Figure 1.4. Each parallel line operates at a low frequency, hence the wires are treated as RC interconnects rather than transmission lines [2]. This results in a need for a large number of repeaters along the lines, which consume large power and area. Therefore, adding more parallel wires to increase the data rate means adding more repeaters, drivers, and buffers that increase the power dissipation and area. In addition, crosstalk between the parallel wires, as well as jitter and skew on the transmitted parallel data make the synchronization at the receiver more difficult, and reduce the reliability of the received data.

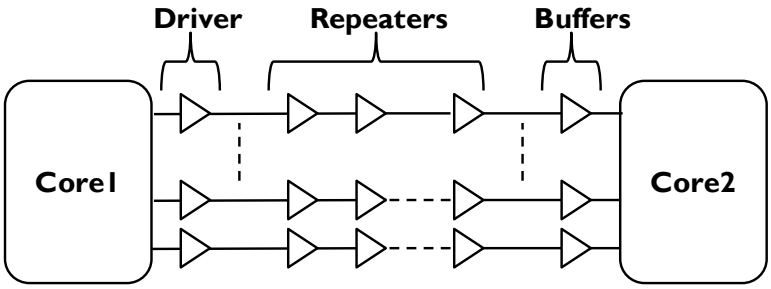


Figure 1.4 Parallel based communication

The problems that result from having parallel-based buses increase significantly in deep submicron technology, because interconnects do not scale in the same trend as transistors. Hence, parallel busses occupy a larger area as compared to processing elements, which results in an increase in routing complexity. Therefore, it will be difficult to boost the performance by integrating hundreds of cores on a single chip, due to the limited availability of metal tracks.

Serial link based communication provides a promising solution for parallel bus problems. In serial communication, the physical channel is a single link replacing N parallel wires. Thus, the data rate of the serial link is N times the data rate of the parallel wire to provide the same bandwidth. Therefore, serial links operate at high frequencies, and they behave more like transmission lines. This gives us the opportunity to benefit from transmission line characteristics for high data rate transmission, and eliminate the need for bulk, and power hungry repeaters [2].

Serial links are a viable solution to reduce the routing complexity, the routing area, and save metal tracks dramatically, hence on-chip serial link design is a hot topic in research. The design of SerDes transceivers is very complex and challenging in terms of power, area, and data reliability.

1.4. Thesis Organization

This thesis is constructed as following; Chapter 2 presents the three proposed signaling schemes and the corresponding SerDes architectures. It also presents the design of different building blocks of the SerDes architectures. After that, a comparison between the three different schemes is presented to show the trade offs. Chapter 3 presents the design of the bang-bang all digital phase locked loop (BBADPLL) for the SerDes transceiver, its different building blocks and design flow. Chapter 4 presents a summary, the conclusions and the anticipated future work.

Chapter 2

On-Chip SerDes Transceiver Architectures and Signaling Schemes

2.1. Introduction

This chapter presents the design of a serializer and deserializer (SerDes) transceiver architecture for three different signaling schemes. Due to many limitations, the data sent on parallel line buses should be transmitted on a single serial link for some distance. A SerDes transceiver helps serialize the data from parallel form to serial form, then deserializes the data at the receiver side to parallel form again. This serial link can be a single line or two lines if differential signaling is used. The SerDes design example in this chapter interfaces an 8-bit parallel data bus to a serial link.

Figure 2.1 shows the main functional blocks of a SerDes transceiver connecting two cores. The transmitter consists of a serializer, which converts the parallel data to a serial data stream, and a driver is used to drive the transmission line. The transmission line can be single ended or

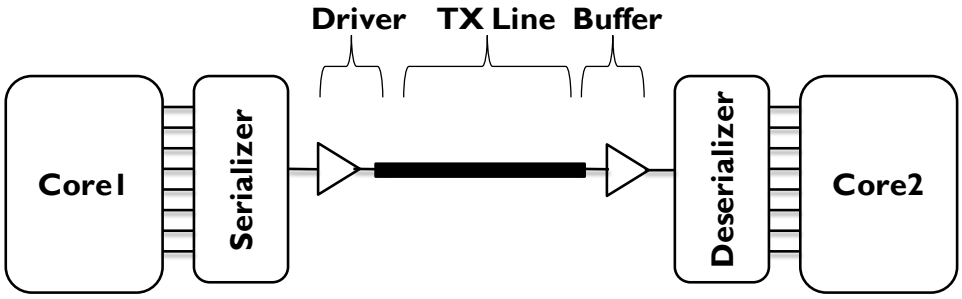


Figure 2.1 Main functional blocks of SerDes transceiver; it consists of Serializer, driver at the transmitter and buffer, deserializer at the receiver

*This chapter was done in collaboration with Ezz El-Din Hussein, a graduate research assistant and master student at NISC lab, Nile University, Egypt.

differential depending on the signaling scheme, and the chosen architecture. The receiver consists of a buffer to drive the receiver blocks, and a deserializer that converts the serial data stream to parallel form at the receiver.

A significant challenge in the implementation of the serial links is achieving a high data rate with minimum power. This chapter introduces three different schemes with different signaling schemes and their corresponding SerDes transceiver architecture implementations.

2.2. Signaling Schemes

In this thesis, three different signaling schemes are presented along with their corresponding SerDes architecture designs. Each signaling scheme has its advantages and disadvantages. Figure 2.2 shows the conventional two-level signaling scheme named *scheme 1*, where each bit is represented by a two-level symbol in any given clock cycle; *"0"* is represented by 0 and *"1"* is represented by *VDD*. This scheme does not need a special encoder. Figure 2.3 shows a three-level signaling scheme, named *scheme 2,* which preserves the *DC* level of the signal. Each bit is encoded by a three-level symbol for duration of two clock cycles; *"0"* is serially represented by "zero, *VDD/2*, *VDD*, then *VDD/2*", while a *"1"* is serially represented by "*VDD*, *VDD/2*, 0, then *VDD/2*". Figure 2.4 shows a three level signaling scheme named *scheme 3*. Each bit is encoded in a three level signal for duration of one clock cycle; *"0"* is serially represented by "zero, then *VDD/2*", while *"1"* is serially represented by "*VDD*, then *VDD/2*".

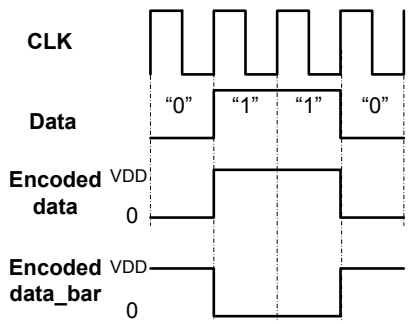


Figure 2.2 Scheme 1: conventional two-level signaling scheme

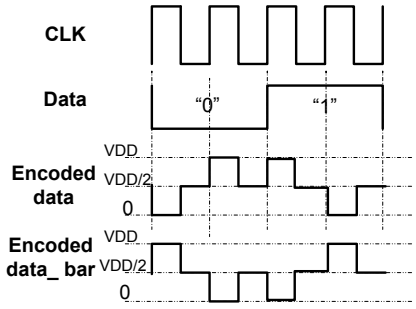


Figure 2.3 Scheme 2: three-level DC preserving signaling scheme

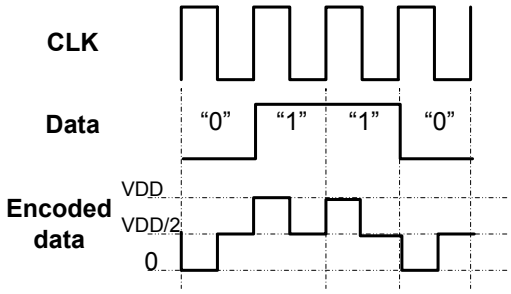


Figure 2.4 Scheme 3: three-level signaling scheme

2.3. Transmission Line

Modeling on-chip wires depends on the operating frequency. At low frequencies, on-chip wires are treated as *RC* interconnects, while at high frequencies, on-chip wires need to be treated as an *RLC* network i.e. a transmission line [2]. Since, serial links operate at high frequencies, they are treated as transmission lines and are modeled as an *RLC* network.

Signaling schemes can have either a single-ended or a differential implementation. Both flavors have their advantages and disadvantages in terms of area, power, and noise. However, the choice depends on the architecture [11]. There are two significant advantages of using a differential line. First, using a differential line is immune to the problems of skew, and crosstalk. Second, differential signaling improves the receiver sensitivity, enabling it easily detect highly attenuated signals. Thus, differential signaling is the choice for high bandwidth signaling on long interconnects. In a two-level signaling scheme, such as scheme 1, differential signaling is very

attractive due to its improved common mode noise immunity, and the ease of detection at the receiver using a simple differential amplifier. However, crosstalk noise can be better suppressed using a shielded or well-spaced single ended line.

A transmission line is modeled as an *RLC* network, such that *R* is the resistance per unit length, *L* is the inductance per unit length, and *C* is the capacitance per unit length. An on-chip link shown in Figure 2.5 consists of a transmission line with characteristic impedance Z_0, a driver with output impedance Z_S, and a load impedance Z_L [4].

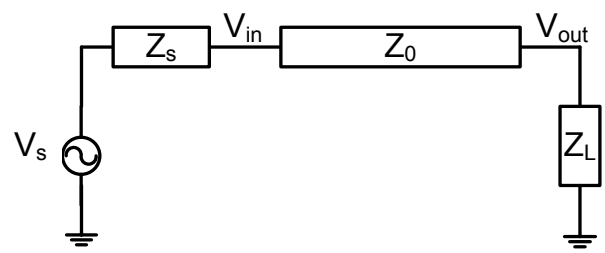


Figure 2.5 Transmission line equivalent

The characteristic impedance Z_0 of a transmission line is given by:

$$Z_0 = \sqrt{\frac{L}{C}}\sqrt{1+\frac{R}{j\omega L}} \tag{2-1}$$

where the values of *R*, *L* and *C* depend greatly on the technology, and metal layer used for transmission line. The voltage seen along the transmission line at time *t*, and a distance *z* from the source with propagation constant γ is given by:

$$Vout\ (z,t) = Re\{Vin(z,t)e^{-\gamma z}\} \tag{2-2}$$

where γ is the propagation constant given by:

$$\gamma = j\omega\sqrt{LC}\sqrt{1+\frac{R}{j\omega L}} = \alpha + j\beta \tag{2-3}$$

The real part of γ is the attenuation constant α, and the imaginary part of γ is the phase constant β given by:

$$\beta = \omega\sqrt{LC}, \quad \alpha_{sat} = \frac{R}{2}\sqrt{\frac{C}{L}} \tag{2-4}$$

The phase velocity v can be calculated using:

$$v = \frac{1}{\sqrt{LC}} \tag{2-5}$$

Figure 2.6 and Figure 2.7 shows the cross section of a single ended, and a differential transmission line, respectively. Transmission lines should be shielded using static *VDD* or ground lines to reduce coupling between signal lines. The interconnect width w and spacing s are the design parameters that determine the values of R, L, C, and the characteristic impedance Z_0 of a transmission line. For a given technology, the design parameters w and s are chosen based on the operating frequency, and the length of the line. The characteristics of the transmission line can be controlled by tuning w and s.

Despite the fact that transmission lines offer the potential of a speed close to the speed of light communication, practical on-chip transmission lines are formed of standard lossy metal layers, which cause dispersion, and limit the effective bandwidth of on-chip serial links [2]. On-chip transmission lines have a series resistance that causes two problems: attenuation and

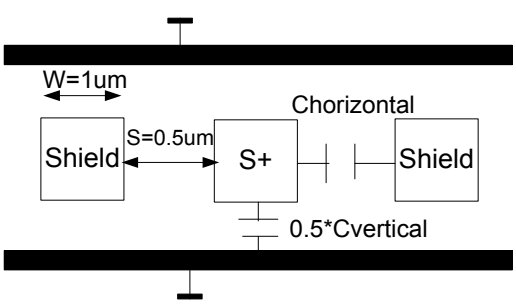


Figure 2.6 Cross section of single ended transmission line

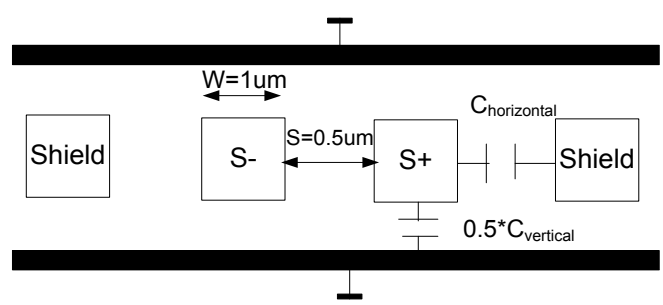


Figure 2.7 Cross section of differential ended transmission line

dispersion. Signal dispersion is a more serious problem than attenuation. For the high frequency component, the inductance effect is significant and the line appears as a transmission line, hence the signal tends to propagate with the speed of light. On the other hand, for the low frequency component, the line appears as *RC* interconnects and the signal propagates with different speed. Dispersion is a result of different propagation speeds between the high frequency component and the low frequency component. This results in inter-symbol interference (ISI), where successive symbols can interfere at the receiver side.

Random data streams cover the entire frequency spectrum, and low frequency components will introduce a large distortion to the signal. Equalization is required to suppress this distortion effect either using a Feed-Forward Equalizer (FFE) at the transmitter or using a Decision Feedback Equalizer (DFE) at the receiver [12]. In the design of the transmission line used in the three schemes, matching and termination techniques are used to suppress signal dispersion and distortion rather than equalizers, which eliminates the need to use these complex and power hungry equalizers.

2.3.2. Transmission Line Matching Techniques

Line termination is a common technique that can be used to achieve a large bandwidth, and a reduction in inter symbol interference or signal reflections. Terminating the transmission line with a resistance equal to the characteristic impedance prevents signal reflections. However, reflections are less important in long resistive lines, because the energy of any reflected signal is dissipated by resistive loss.

2.3.2.1. Source Matching Technique

This matching technique is based on having the source impedance at the input of the transmission line Zs equal to the characteristic impedance of the line Z_0. The main advantage of

using source matching technique over other matching techniques is doubling the amplitude of the signal at the input of the line due to the reflected signal. This matching technique eliminates the need for bulky drivers at the transmitter, which consumes a lot of power. In the source matching technique, the attenuation that takes place across the transmission line is compensated by the amplitude increase at the input of the line. However, this technique does not reduce the dispersion or eliminate the distortion. Therefore, to apply this technique, the transmitted signal should have a high data rate with a constant *DC* to benefit from the transmission line characteristics at high frequency [4]. This matching technique is applied to the signaling scheme 2, which preserves the *DC* level of the transmitted data. However, this matching technique limits the minimum transmitted data rate.

2.3.2.2. Resistive Termination Technique

In this termination technique, the transmission line is not terminated by a resistance equal to the characteristic impendence. A resistive termination technique was presented in [2], and applied to reduce inter-symbol interference. The step response of a lossy transmission line with length *l* is given by:

$$Vout = Vin\, e^{-\alpha l} \tag{2-6}$$

The low frequency components face a voltage divider formed between the total resistance of the line and the terminated resistance and are thus given by:

$$Vout = Vin\, \frac{R_l}{R_{tot} + R_l} \tag{2-7}$$

There is an optimum value for R_L that masks the effect of the slow frequency component, and reduces inter-symbol interference. This optimum value of R_L can be calculated by equalizing the *DC* and *AC* attenuation of the transmitted signal as following:

$$e^{-\alpha l} = \frac{R_l}{R_{tot} + R_l} \tag{2-8}$$

This matching technique does not put constraints on maintaining the *DC* level of the transmitted signal. Hence, this termination technique is applied on signaling schemes 1 and 3. However, the transmitted signal suffers from high attenuation along the transmission line.

2.4. Proposed SerDes Transceiver Architectures

2.4.1. Scheme 1: Conventional Two-Level Code

Signaling scheme 1, shown in Figure 2.2, is the conventional two-level signaling scheme in the literature [13], [14]. Figure 2.8 shows the block diagram of the proposed SerDes architecture designed for 10.5 Gbps data rate. This system consists of a transmitter, receiver, and a lossy on-chip transmission line. The transmitter consists of a 6-8 encoder, a serializer, a driver, and a 7GHz all digital phase locked loop (ADPLL). At the transmitter, a 6-8 encoder is used to generate data transitions, which eliminates long streams of 0's and 1's to simplify recovery at the receiver using the clock and data recovery (CDR). The serializer operates at half the data rate. It serializes the 1.75GHz 8-bit parallel data into 14Gbps serial data using a 7GHz clock generated from the ADPLL. Then, a driver is used to drive the lossy transmission line.

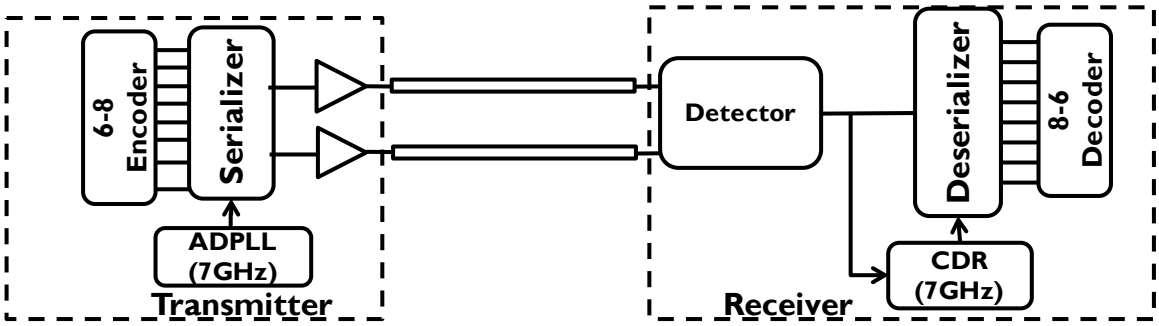


Figure 2.8 SerDes transceiver architecture scheme1

At the receiver side, a differential amplifier is used as a detector to detect the two-level received signal. After that, the CDR recovers the data from the incoming data stream, and the deserializer is used to convert the 14Gbps serial data stream into a stream of 8-bit parallel data. Then, the decoder is used to decode the eight parallel bits to six parallel bits. The use of 6-8 encoder results in about 25% loss in the data rate, so this architecture's data rate is 10.5Gbps.

2.4.1.1. Encoder and Decoder

There are many encoding schemes used in data transmission, and the choice of the encoding scheme depends on the application. Encoding can be used for many reasons: error detection and correction, preserving the DC level of the signal, reducing inter-symbol interference along the transmission line, or generating data transitions. Data transitions are important for proper CDR operation, because transitions avoid the long streams of 0's and 1's. In this architecture, the decoder is designed to generate data transition for the CDR to function properly.

The proposed simple 3-4 encoder is shown in Figure 2.9 . The proposed encoder adds a bit to every 3 bits, where the fourth bit is the inversion of the third bit. At the receiver, the decoder just takes the three bits and discards the fourth bit.

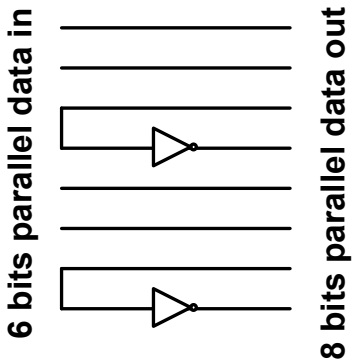


Figure 2.9 Proposed 6-8 encoder

2.4.1.2. Serializer

The shift-register type serializer, shown in Figure 2.10 is one of the simplest conventional serializer designs. It consists of MUXs and serial D-FFs. The D-FFs load parallel data bits and shift them to one side using a high-speed clock. The critical path delay is one MUX and one D-FF delay. Despite the design simplicity and the area reduction of the shift register serializer, this conventional architecture has two problems: First, the maximum clock frequency is limited by the delay of the D-FF and MUX, so it can't be used for high data rate transmission. Second, operating all DFFs at the high-speed clock for the serialization is a system overhead [15].

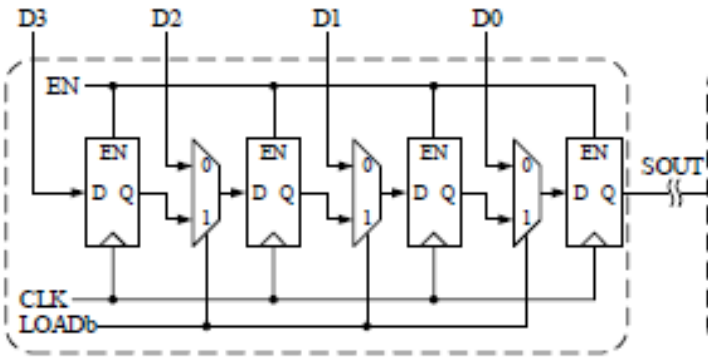


Figure 2.10 Conventional shift-register type Serializer [15]

Figure 2.11 shows the block diagram of the proposed serializer that is very similar to the serializer proposed in [16]. Double-edge-triggered flip-flops (DETFF) are used to serialize the data. The flip-flops were implemented using C^2MOS (Clocked CMOS) registers [17], which are clock overlap insensitive. Although the proposed serializer consumes more area than the conventional shift-register type, it consumes less power and can overcome the previously stated problems of the conventional shift-register type. This is because the registers operate at lower frequencies (divided versions of the high-speed clock), and reduce the load on the high-speed clock.

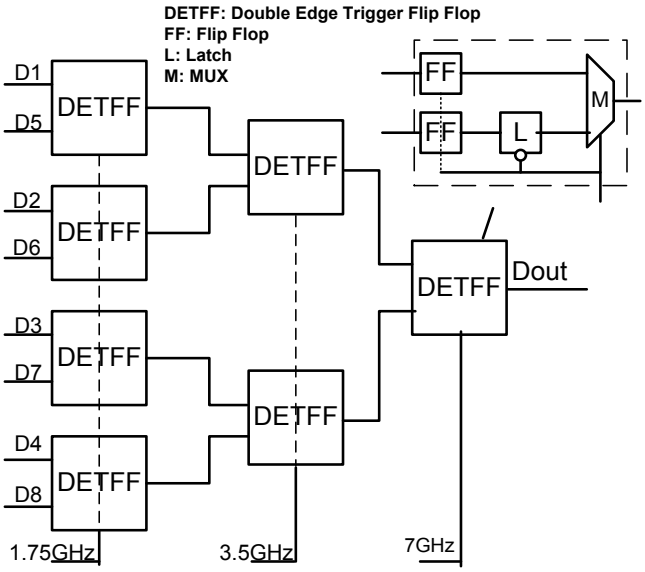


Figure 2.11 Serializer block diagram

2.4.1.3. Deserializer

The simple conventional way of implementing a deserializer is the shift register based deserializer operating on the high data rate. However, Figure 2.12 and Figure 2.13 show the

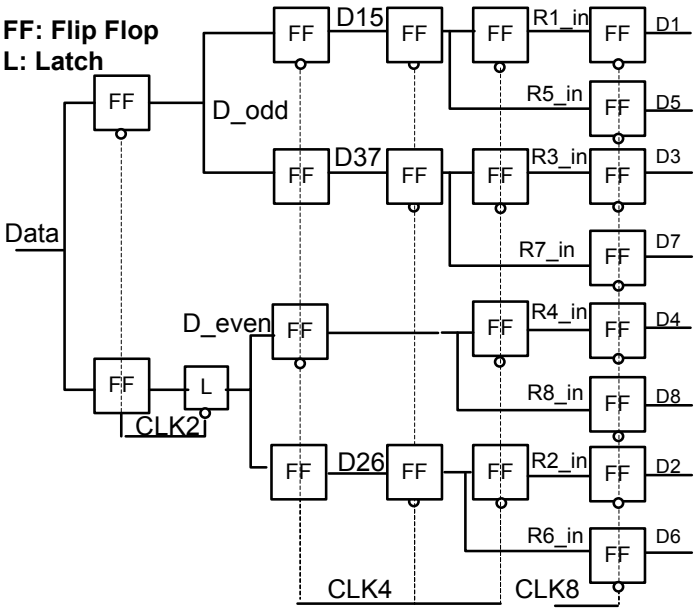


Figure 2.12 Deserializer block diagram

proposed deserializer and the equivalent waveforms, respectively. The proposed deserializer utilizes both rising and falling edges of the 7GHz, 3.5GHz, and 1.75GHz clocks to deserialize the serial 14GHz data stream into eight parallel data stream. In addition to its low power dissipation, the proposed deserializer reduces the latency resulting from the shift-register based implementation.

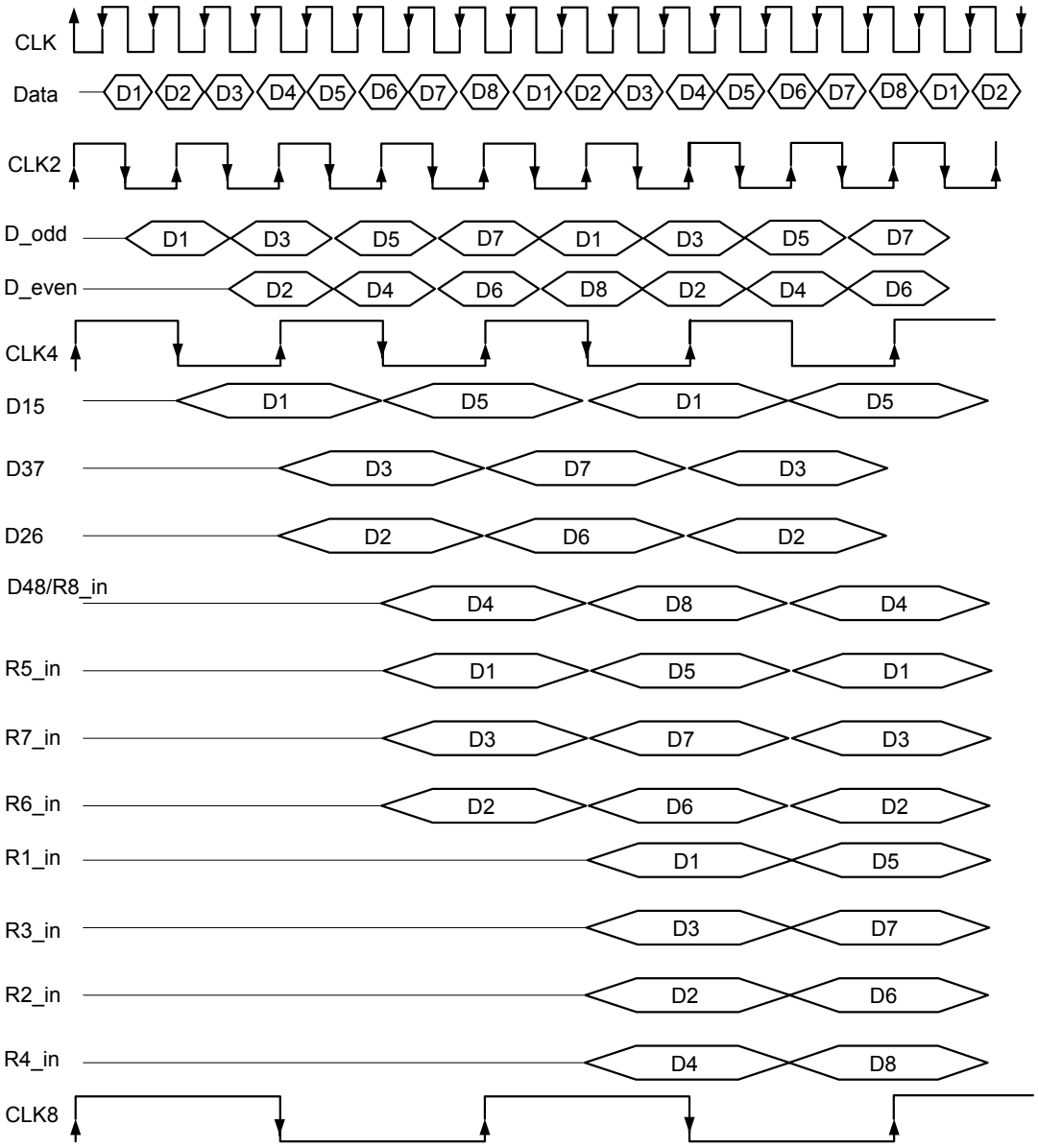


Figure 2.13 Deserializer waveforms

2.4.1.4. Clock and Data Recovery (CDR)

The clock and data recovery (CDR) is a critical building block in SerDes transceivers. The CDR is responsible for recovering the data from the received signal with minimum bit error rate. To achieve the minimum bit error rate, the CDR needs to sample the data at the middle of the eye as shown in Figure 2.14. The CDR design is complex, because the received data stream is not aligned with the clock at the receiver. There are two reasons for the misalignment. First, the different time domains in the transmitter and the receiver results in a deterministic phase offset and the jitter accumulated during the channel transmission results in time uncertainty. The second reason is the timing uncertainty (jitter) of the incoming data [18].

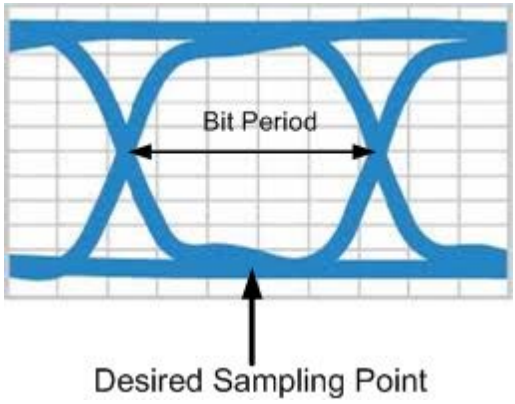


Figure 2.14 Eye diagram of the data, the Y-axis is the voltage and the X-axis is the time

The conventional CDR design based on an analog Charge Pump PLL (CPPLL) is shown in Figure 2.15 [18]. The architecture of the CDR has been realized as a two-loop structure consisting of a frequency loop and a phase loop. At start up, the frequency loop provides fast locking to the system frequency with the help of the reference clock. After that, the VCO clock reaches a frequency very close to the data rate. Then a *frequency lock* signal is generated and the frequency loop is turned off, while the phase loop is turned on. The phase loop tracks the phase of the VCO generated clock with respect to the data and aligns them together, such that the clock

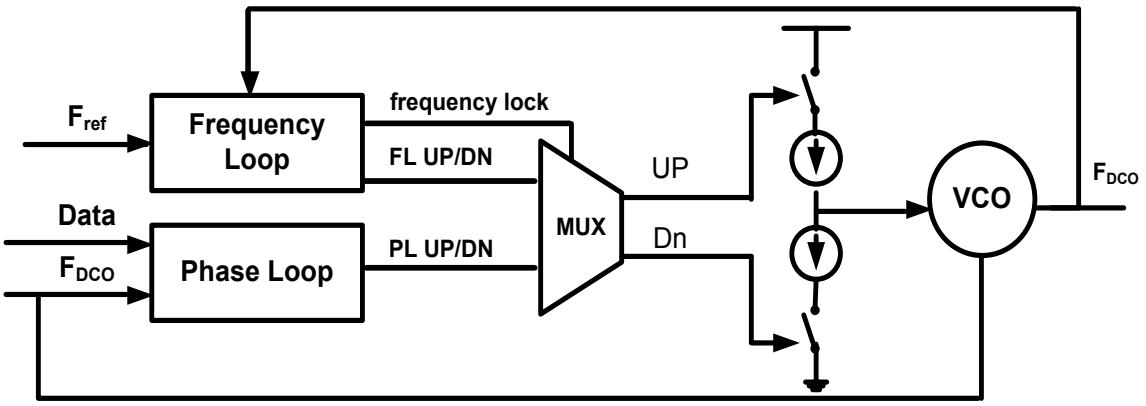


Figure 2.15 Conventional charge pump based CDR

can sample the data at the middle of the eye. For the phase loop to function properly, we need as many data transitions as possible for continuous tracking and alignment. Hence, long streams of zeros or ones may cause loop instability. Therefore, a 6-8 bit encoder is used to generate data transitions, and guarantee a maximum of four successive 0's or 1's. However, this encoder results in about 25% loss in the data rate.

The charge pump based CDR (CPCDR) has proven good jitter performance and low power consumption. However, in deep submicron technology, the design of an analog CPPLL faces many challenges [19]. First, the performance of analog blocks degrades due to low voltage headroom. Second, there is a large area overhead due to the metal capacitors needed for implementing the analog loop filter. Third, the CPCDR faces difficulty in achieving good jitter performance with a noise-susceptible analog control, which is induced by the reduced supply voltages and the large noise generated from adjacent digital circuits. The challenges encountered in analog-based CDRs implemented in deep submicron technologies, make the all-digital CDR (ADCDR) an attractive alternative.

2.4.1.4.1.All Digital CDR Proposed Architecture

Figure 2.16 shows the proposed architecture of a dual loop, half-rate ADCDR. The proposed ADCDR architecture consists of a digital controlled oscillator (DCO), and two loops: a frequency acquisition loop and a phase loop. The frequency acquisition loop is used to set the initial coarse word of the DCO at power up, which results in an output frequency very close to the target frequency. The frequency acquisition loop consists of counter-based logic that compares between the frequencies of the DCO output and the reference clock. After that, a binary search algorithm (BSA) is used to adjust the coarse word (4 bits), and the fine word (5 bits) of the DCO. After the frequency acquisition is done, the frequency loop generates a *frequency lock* as an enable signal for the phase loop. The phase loop aligns the clock and data by controlling the DCO input fine word. The phase loop consists of proportional and integral paths. The proportional path is a direct path from the output of the phase detector to the control bits of the DCO. The proportional path controls the frequency of the DCO directly at a speed higher than the integral path to provide stability. This is because the integral path is slow in a

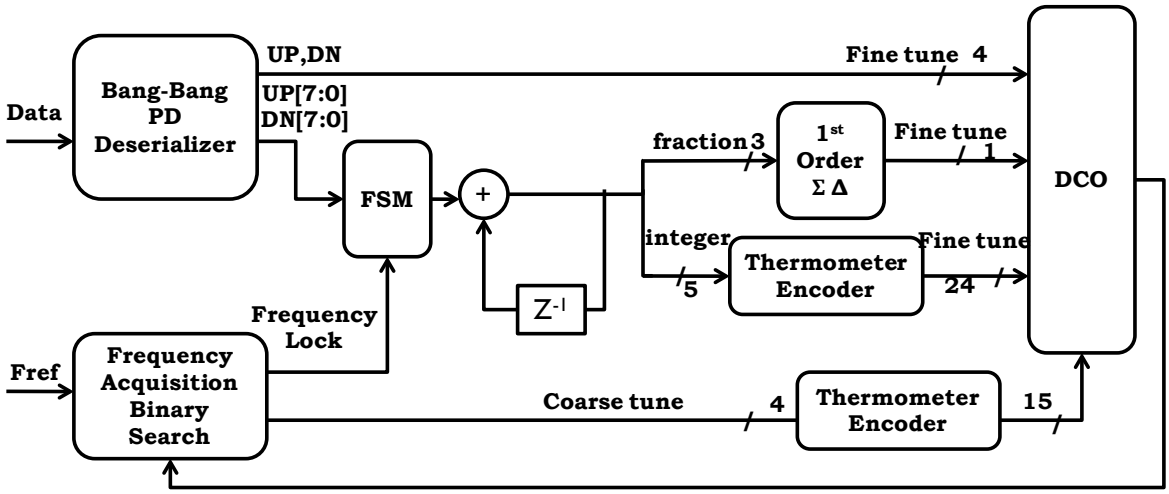


Figure 2.16 Proposed CDR block diagram

way that the clock may go out of phase before the integral paths responds. The integral path consists of an 8-bit deserializer, a finite state machine (FSM), and a first order infinite impulse response (IIR) filter. The phase loop tracks the phase difference between the data and the output DCO clock using a bang-bang phase detector (BBPD), which indicates whether the clock leads or lags the data by generating *UP* and *DN* signals. Then, the deserializer is used to convert the *UP/DN* stream to one-eighth of the data rate for the integral path. After that, the FSM takes the average of 64 bits, and feeds it into a first order IIR integrator.

Since, most of the building blocks of the All Digital CDR are made of synthesized digital circuits except for the DCO, and the BBPD, they can be easily scaled from one technology to another [19] and [20]. One of the challenges in the CDR implementation is the speed limitation; however, implementing half-rate CDR architecture overcomes this problem [21]. Half-rate CDR architecture can be implemented using a half-rate BBPD that uses at least two clock phases. Implementing a half-rate CDR offers many advantages. First, the lower frequency of operation reduces the power and puts fewer constraints on the digital blocks of the CDR. Second, as the clock frequency is reduced, the design of the DCO is relaxed, so the jitter performance will be better, and the loop stability can be achieved easily. On the other hand, the half-rate CDR architecture needs multiphase clock signals; in-phase and quadrature phase clocks. This means that any mismatch between the in-phase and the quadrature phase clock will result in a reduction in the overall system performance. Moreover, implementing a multiphase BBPD increases the circuit design complexity. However, a half-rate CDR implementation is a very attractive alternative for a high data rate transmission.

2.4.1.4.2.Bang-Bang Phase Detector

Figure 2.17 and Figure 2.18 show the schematic and the transfer function of the linear phase detector (PD) and the bang-bang PD, respectively [18]. The linear PD output is proportional to the magnitude of the input phase error, while the BBPD output indicates only whether the clock leads or lags the data regardless of the magnitude of the input phase error. The first BBPD was published by Alexander in [23] and often referred to as the binary PD. The BBPD has several advantages as compared to the linear PD. First, its binary output makes the integration with the digital loop filter easier and enables sampling the data using multiphase clock, such that the CDR can operate at half the data rate. Second, there is no need for an analog to digital converter (ADC) or a time to digital converter (TDC) to convert the average analog output to digital bits for the DCO.

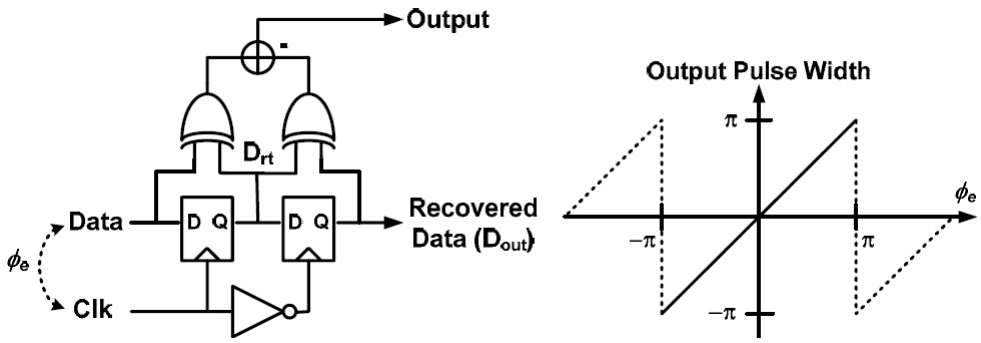


Figure 2.17 Linear Phase Detector: Schematic and Transfer function [18]

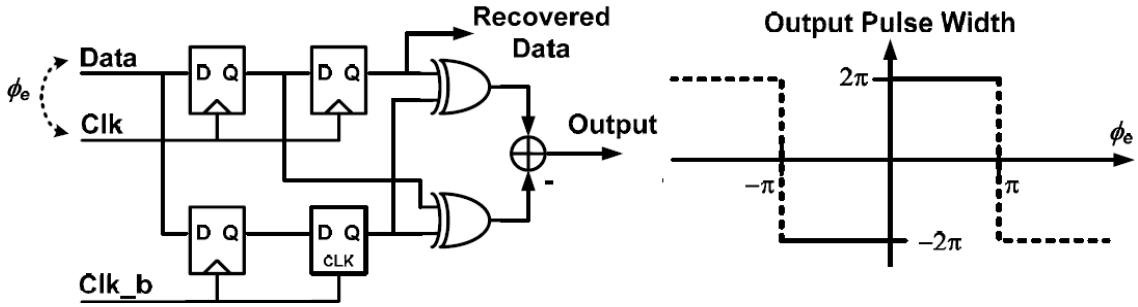


Figure 2.18 Bang-Bang Phase Detector: Schematic and Transfer function [18]

The first half-rate linear PD and BBPD were proposed by Savoj and Razavi in [21]. Figure 2.19 shows the schematic of the proposed tri-state half-rate BBPD. The in-phase and quadrature phase clocks are used to sample the data three times as indicated by *S0*, *S1,* and *S2* in Figure 2.20. The *UP* signal is the result of an XOR between the current state *S1* and the previous state *S0*. The *DN* signal is the result of an XOR between the current state *S1* and the next state *S2*. The three possible states are indicated in Table 2.1; if the clock lags the data, *UP* signal will be high and *DN* signal will be low. If the clock leads the data, the *UP* signal will be low and the *DN* signal will be high. In case of no transitions, both *UP* and *DN* are low.

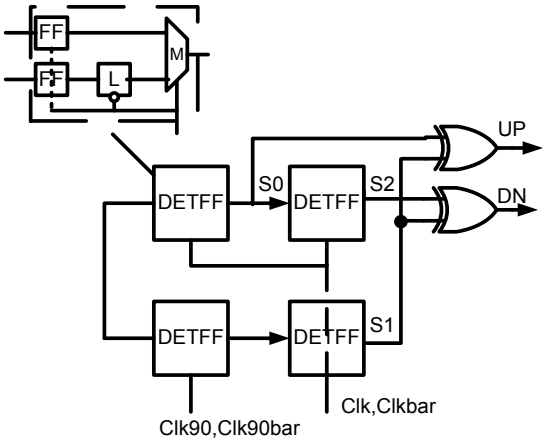


Figure 2.19 Proposed half-rate BBPD

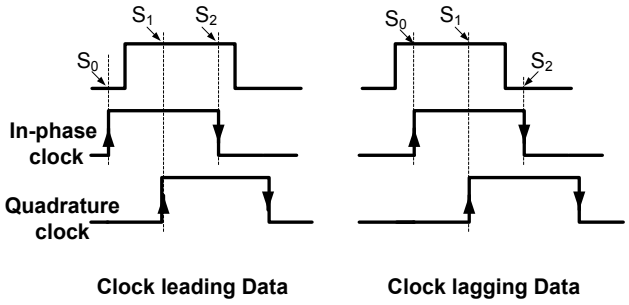


Figure 2.20 Data sampled by the clock at 3 states

Table 2.1 Different states of clock and data are indicated by UP and DN signals

UP	***DN***	**State**
1	0	Clock lag data
0	1	Clock lead data
0	0	No transition

2.4.1.4.3.Digital Controlled Oscillator (DCO)

Figure 2.21 shows the proposed DCO. The ring oscillator based DCO consists of four differential inverters. The DCO generates multiphase clocks, whose frequencies are controlled by the input integral word. The in-phase and quadrature phase clocks are fed back to the half-rate BBPD for phase comparison. The DCO output frequency is controlled by a 4-bit coarse word and a 5 bits fine word. The oscillator is controlled by a thermometer code rather than a binary code to ensure monotonic increase. Figure 2.22 shows the output frequency versus the fine word for different coarse words in typical conditions at 125°c.

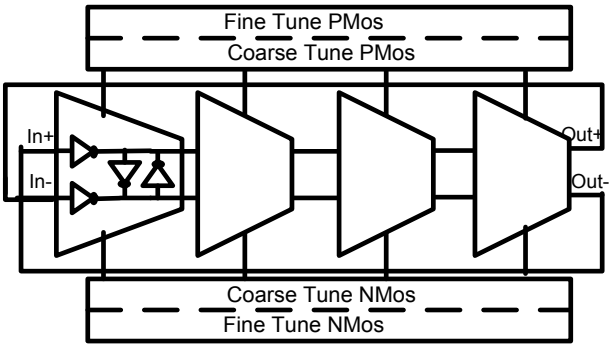


Figure 2.21 DCO

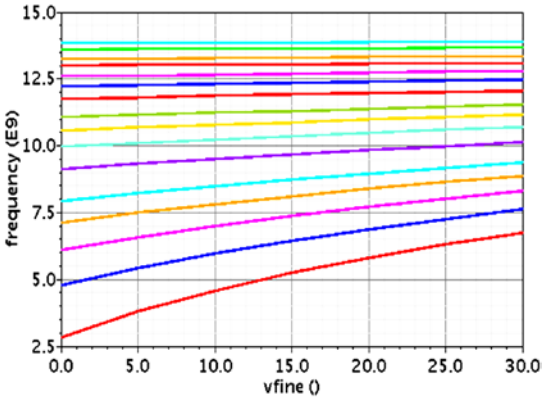


Figure 2.22 DCO tuning curves

2.4.2. Scheme 2: Three-Level code with Constant DC

In scheme two, the data and the clock are multiplexed using a new three-level coding technique shown in Figure 2.3. This coding technique enables splitting the clock and the data stream at the receiving side with a simple circuit. This coding maintains the signal *DC* level at $V_{DD}/2$ independent of the data pattern, thus eliminating the need for equalization circuits as long as the circuit operates at a high frequency. Furthermore, this new signaling scheme eliminates the need for an extra link to send the clock on, or using the complex and power hungry CDR at the receiver side. Accordingly, the proposed signaling scheme results in lower area, power, and metal track usage as compared to the other conventional SerDes designs presented in [13], [14], and scheme 1.

Moreover, this signaling scheme is insensitive to jitter accumulated during the signal transmission, because the clock signal is extracted from the data stream. Hence, any timing errors in the received signal will also be reflected in the extracted clock and the data will be sampled correctly. Therefore, there is no high constraint on the jitter performance of the PLL at the transmitter side, which makes an all-digital PLL (ADPLL) a very attractive frequency synthesizer for this SerDes architecture. Replacing the analog PLL at the transmitter side with an ADPLL reduces the design complexity, power, and area overhead significantly.

Figure 2.23 SerDes transceiver architecture for 3 level signaling preserving the DC level in scheme 2

Figure 2.23 shows the block diagram of the proposed 12Gbps SerDes architecture for scheme 2 signaling. The system consists of a transmitter, a receiver, and a lossy on-chip wire that will be used as a transmission line. The transmitter serializes 1.5GHz 8-bit parallel data, generates the differential three-level code shown in Figure 2.3, and drives the lossy transmission line. The receiver decodes the three-level coded data, and extracts the clock from the received signal using a simple phase detector. A deserializer is then used to recover the 8-bit parallel data from the 12Gbps serial data stream.

2.4.2.1. Three-level Encoder and Driver

The three-level encoder is shown in Figure 2.3. The Latch-MUX configuration implemented by fast XOR and XNOR gates, shown in Figure 2.24 multiplexes the 12Gbps data with the 12GHz clock. After that, the transmission gates are used to multiplex the 24GHz clock with the data and drive the transmission line. It is very interesting to notice that the *VDD/2* source doesn't dissipate power, because the DC level of the signal is *VDD/2* and the differential pair outputs are charged and discharged to *VDD/2* at the same time.

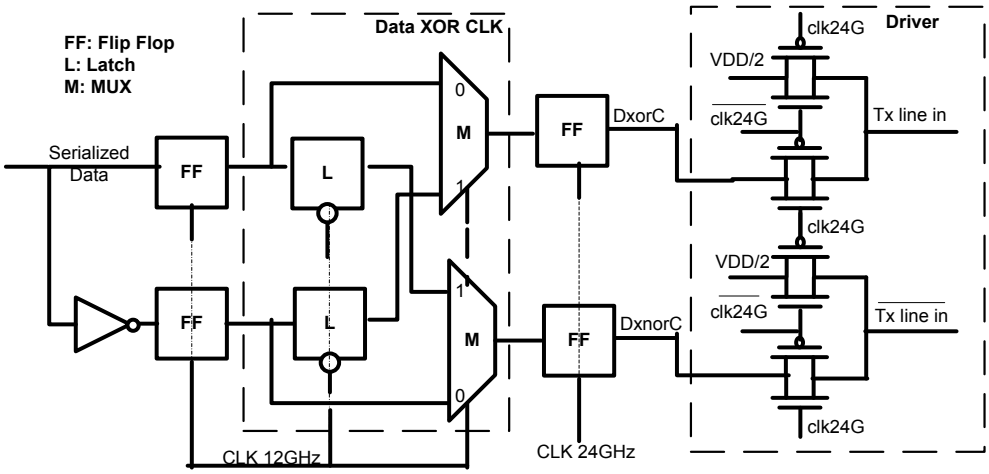


Figure 2.24 Scheme 2: Three-level Encoder and driver

In this scheme, source matching is used at the transmitter side to make use of the transmission gate (driver) resistance, and benefit from the signal reflections at the receiver side. The signal reflections double the amplitude of the received signal. The sizes of the transmission gates are adjusted in such a way that that the source impedance of the transmission line Z_s equals the characteristic impedance of the transmission line Z_0. Figure 2.25 shows the internal signals of the three-level encoder, and the transmission line input and output signals. Note that the amplitude of the input signal of the transmission line approximately equals the amplitude of the amplitude of the output signal of the transmission line.

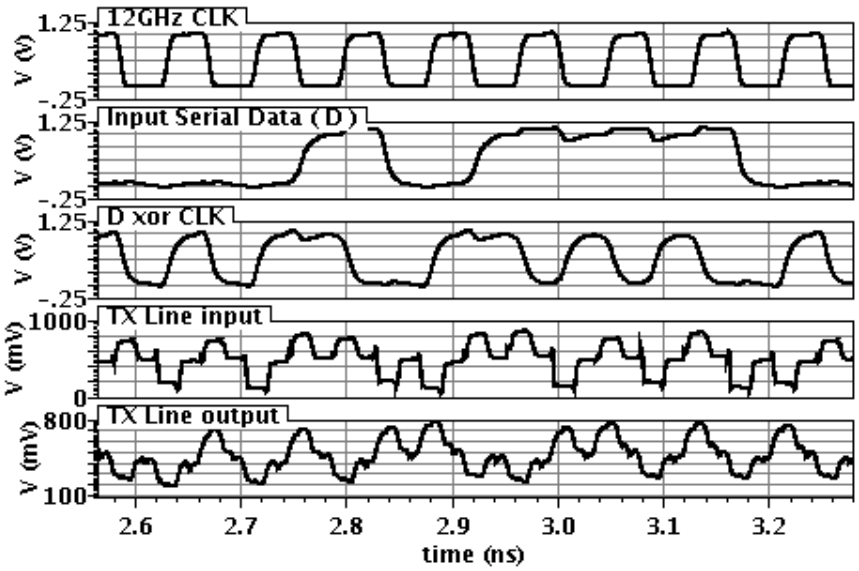


Figure 2.25 The encoder internal signals and the transmission line input/output signals

2.4.2.2. Transmission Line Design

An on-chip differential transmission line with source impedance Z_s and characteristic impedance Z_0 is shown in Figure 2.26. In this scheme, source matching is used and Z_s is designed to be equal to Z_0.

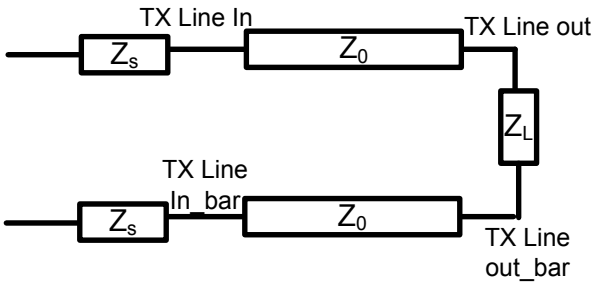


Figure 2.26 Scheme 2, source matching technique is used

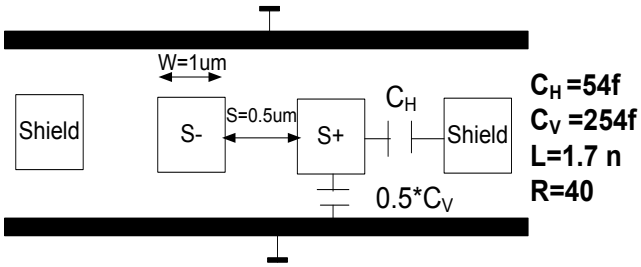


Figure 2.27 cross section of differential transmission line and design parameters of scheme 2

Figure 2.27 shows the cross section of the differential transmission line and its parasitic parameters. The transmission line is designed to be 3mm long and 1μm wide to have the following parasitics: R= 40Ω/mm, L=1.7nH/mm, C_V=254fF/mm, C_H= 54fF/mm and $C=C_v+2C_H$ = 364fF/mm.

Using the parasitic values of the interconnect, the characteristic impedance Z_0 calculated using (2-1) equals 68Ω and the propagation speed v calculated using (2-5) equals 40.2mm/ns. Table 2.3 summarizes the design parameters of the transmission line. The attenuation constant α calculated from (2-4) equals to 0.293mm^{-1}, results in a total attenuation of $(1-e^{-\alpha l})$=58.4% of the signal amplitude. Applying source matching doubles the signal amplitude and results in an attenuation of only $(1-2e^{-\alpha l})$=16.8%. The propagation time for the signal throughout the 3mm long, transmission line is 74.6ps. Figure 2.25 shows the three-level code at the input and the output of transmission line. The peak-to-peak amplitude of the three-level code at the input of the transmission line equals to 600mV, while the peak-to-peak amplitude of the signal at the output of the transmission line equals 500mV. Thus, using source matching resulted in only 16% attenuation loss.

Table 2.2 Design parameters of scheme 3 transmission line

Parameter	*Symbol*	*Value*
Resistance per unit length (Ω/mm)	R	40
Inductance per unit length (nH/mm)	L	1.7
Capacitance per unit length (fF/mm)	C	364
Attenuation constant (mm^{-1})	α_{sat}	0.293
Attenuation along 3mm line	$e^{-\alpha l}$	41.6 %
characteristic impedance (Ω)	Z_0	68
Phase velocity (mm/ns)	v	40.2
Propagation time along 3mm line (ps)	t_p	74.6

2.4.2.3. Decoder

At the receiver side, low switching threshold inverters shown in Figure 2.28, are used as a decoder to convert the three-level signal to a two-level signal. As shown in Figure 2.29, *"0"* is represented by a pulse on *A* followed by a pulse on *B*, while a *"1"* is represented by a pulse on B followed by a pulse on *A*. The switching thresholds of these inverters are susceptible to process, voltage and temperature variations. A calibration scheme will be presented later to adjust the threshold voltage of the inverters.

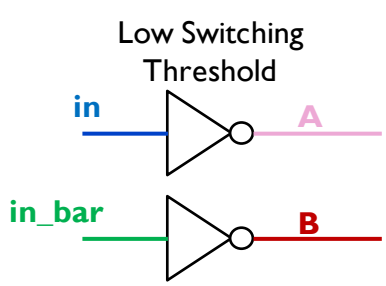


Figure 2.28 Low switching threshold inverters convert the three-level signals to two-level signal *A* and *B*

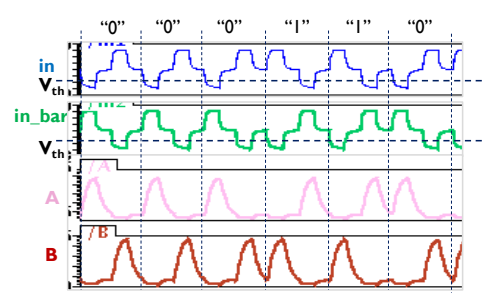


Figure 2.29 *A* and *B* are the output signals of the low switching threshold inverters; *"0"* represented by a pulse on A followed by a pulse on B, while *"1"* represented by a pulse on *B* followed by a pulse on *A*.

2.4.2.4. Phase Detector

The Phase Detector (PD) shown in Figure 2.30 is very common in an analog PLL. It works as a three state machine, starting with both *QA* and *QB* equal to zero representing the reset state. When a pulse is received on *A*, the PD will assert *QA*, and then wait for a pulse on *B* to reset. The same will be done with *QB*, when the PD receives a pulse on *B* followed by a pulse on *A*. An SR latch is used to set the output using *QA* and reset it using *QB*, while a NAND gate is used to extract the 12 GHz output clock. After that, the deserializer is used to convert the 12Gbps serial data to 8-bit parallel data strams at a clock frequency of 1.5 GHz. Figure 2.31 shows how the phase detector works. If a pulse on *A* is received before a pulse on *B*, the PD detects a *"0",* and if a pulse on *B* is received before a pulse on *A*, PD detects a *"1"*. A NAND gate is used to extract the clock and a SR latch is used to recover the data.

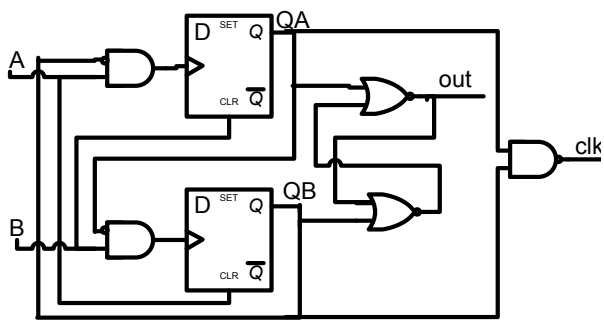


Figure 2.30 Phase Detector

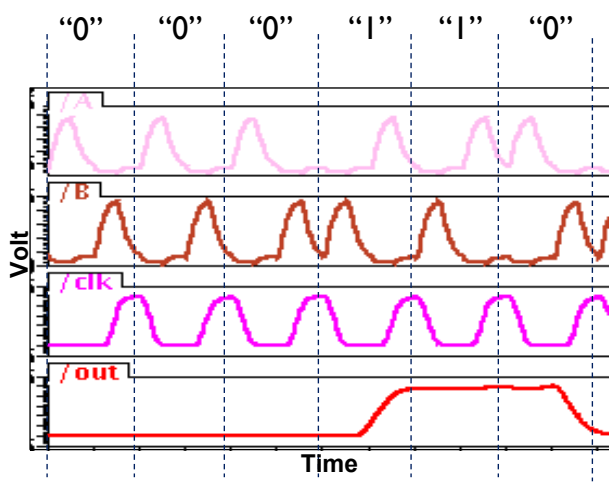


Figure 2.31 Simple Phase detector extract the clock and recover the data

2.4.3. Scheme 3: Three-Level Code

Scheme 3 is proposed to overcome many problems encountered in scheme 2. Although, the signaling scheme 2 with source matching offers many advantages mentioned before, it suffers from many problems. First, each bit is encoded in two clock cycles, thus this signaling scheme loses half the data rate. Second, the use of source matching technique puts a limitation on the minimum frequency of the transmission line, which makes it difficult to operate probably in slow corners. Third, sending differential signals has no advantages in terms of noise immunity, because at the receiver, there are two separate low switching threshold inverters used to detect the signals. Thus, there is a great probability of mismatch at the receiver side, which will make extracting the clock and recovering the data difficult.

Signaling scheme 3 presents a three-level coding technique shown in Figure 2.4 that overcomes the problems of signaling scheme 2, while offering the same advantages provided by scheme 2. The data and the clock are multiplexed using a three-level code that enables recovering the clock from the data at the receiving side using simple circuitry. This also eliminates the need of sending the clock using an extra wire, or using a complex conventional clock data recovery (CDR) at the receiver side. Since the clock and data are multiplexed, this technique is also insensitive to jitter accumulated during signal transmission, which eliminates any huge requirements on the jitter performance of the PLL at the transmitter.

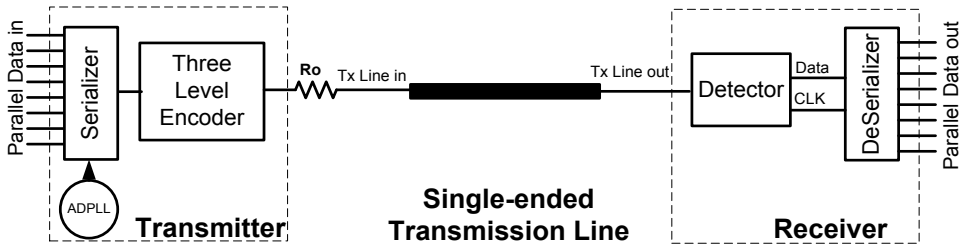


Figure 2.32 Scheme 3 SerDes transceiver architecture for 3 level signaling

The first advantage of this scheme is that each bit is encoded in only one clock cycle, which overcomes the problem of losing half the data rate is overcome. Figure 2.32 shows the block diagram of the proposed SerDes architecture for scheme 2. Resistive termination is used to reduce dispersion and avoid distortion. Hence, this signal doesn't put a limitation on the minimum frequency. The system consists of a transmitter, a receiver, and a lossy single-ended on-chip wire that will be used as a transmission line. The transmitter serializes 8-bit parallel data and generates the three-level code shown in Figure 2.4, and drives the lossy transmission line. The receiver decodes the three-level coded data, and extracts the clock from the received signal using a very simple detector. A deserializer is then used to recover the 8-bit parallel data from the serial data stream.

2.4.3.1. Driver and Three-level Encoder

Figure 2.33 shows the proposed driver and the three-level encoder used to multiplex the clock and data by adding a third level, using a *VDD/2* source.

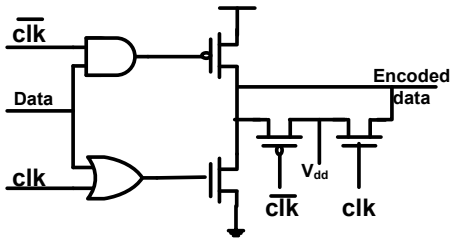


Figure 2.33 Scheme 3: Three-level Encoder and driver

2.4.3.2. Transmission Line Design

An on-chip single ended transmission line with characteristic impedance Z_o is shown in Figure 2.34. Resistive termination is used to reduce the inter symbol interference, where the termination resistance R_L is calculated using (2-8). The termination resistance is connected to a *VDD/2*

source rather than ground, so there is no *DC* power dissipation. Figure 2.35 shows the cross section of a single-ended transmission line and the parasitic parameters. The transmission line is designed to be 3mm long and 3μm wide. For a 3μm wide interconnect, the line parasitics: R= 18.2Ω/mm, L=1nH/mm, C_V=763.5fF/mm, C_H= 52.2fF/mm and C=C_v+2C_H= 868fF/mm.

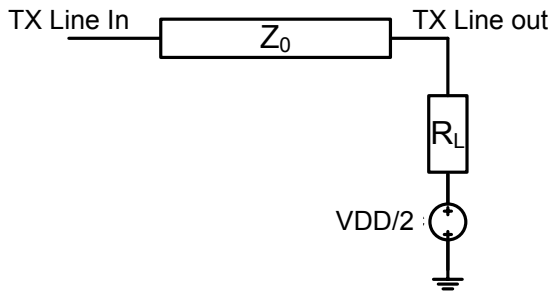


Figure 2.34 Scheme 3 transmission line termination resistance connected to VDD/2 source rather than ground

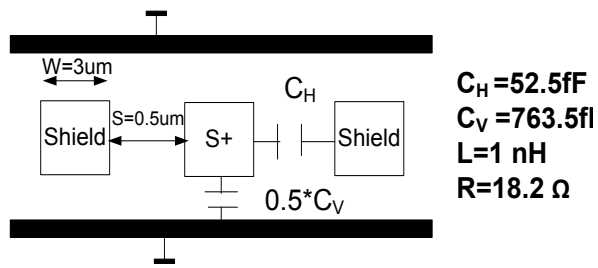


Figure 2.35 cross section of single ended transmission line

From the parasitic values of the interconnect, the attenuation constant α calculated using (2-4) equals 0.268 mm^{-1}, the characteristic impedance Z_0 calculated using (2-1) equals 64Ω and the propagation speed v calculated using (2-5) equals 33.9 mm/ns. Table 2.3 summarizes the design parameters of the transmission line. The attenuation of the signal along the transmission line is 45%. The value of R_L needed for applying the resistive termination technique is calculated using (2-8) and equals 40Ω.

The propagation time for the signal transmitted along the 3mm long transmission line is 88ps, which can be seen in Figure 2.36. The three-level signals at the input and the output of transmission line are shown in Figure 2.36. The peak-to-peak amplitude of the three-level code at the input of the transmission line equals to 600mV, while the peak-to-peak amplitude of the signal at the output of the transmission line equals to 300mV. There is a 50% attenuation loss compared to only 16% attenuation in scheme 2 due to using the source matching technique.

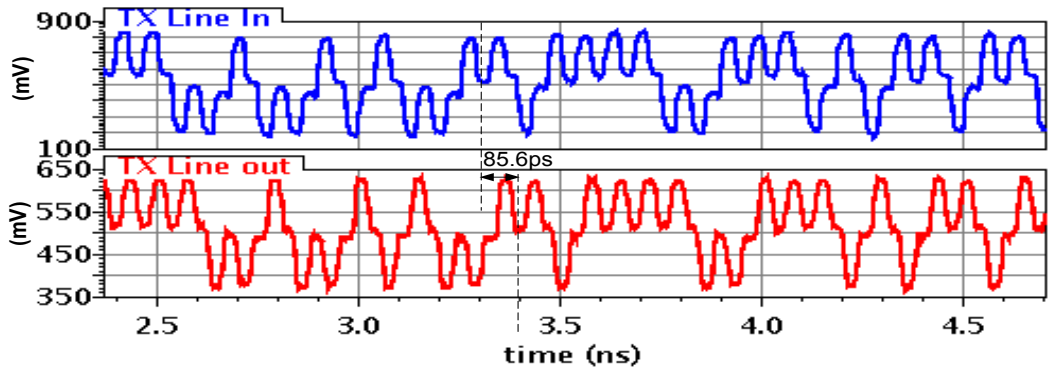


Figure 2.36 Scheme 3, three-level code at the input and output of the transmission line

Table 2.3 Design parameters of scheme 3 transmission line

Parameter	***Symbol***	***Value***
Resistance per unit length (Ω/mm)	R	18.2
Inductance per unit length (nH/mm)	L	1
Capacitance per unit length (fF/mm)	C	868
Attenuation constant (mm^{-1})	α_{sat}	0.268
Attenuation along 3mm line	$e^{-\alpha l}$	0.45
characteristic impedance (Ω)	Z_0	34
Phase velocity (mm/ns)	v	33.9
Propagation time along 3mm line (ps)	t_p	88.4

2.4.3.3. Detector

Figure 2.37 shows the schematic of a simple detector used at the receiver side to extract the clock and recover the data from the three-level encoded signal. The detector consists of high switching threshold and low switching threshold inverters, which are used to convert the three-level codes into two-level signals *A* and *B* as shown in Figure 2.38. A NAND gate is used to

generate the clock and an SR latch is used to recover the data. Figure 2.38 shows the simulation results of the simple detector in Figure 2.37.

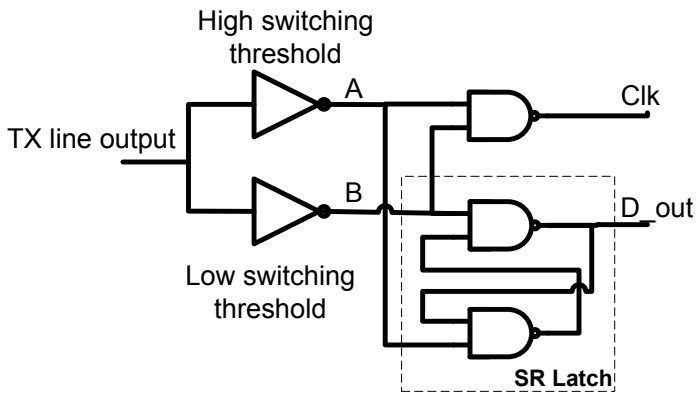


Figure 2.37 Scheme 3 simple detector

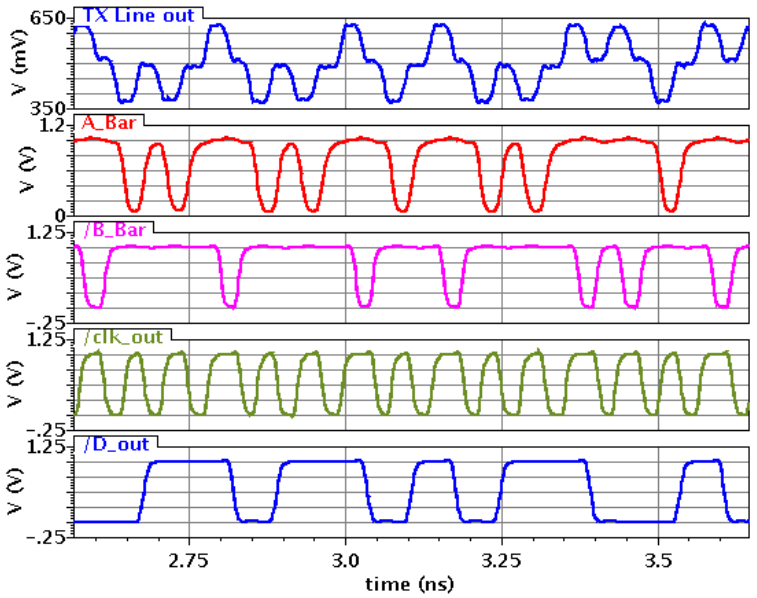


Figure 2.38 Detector at the receiver detects the 3 level code, extract the clock and recover the data

2.5. Calibration Technique Applied to Scheme 2 and Scheme 3

The reliability of detecting the three-level signals at the end of the transmission line depends on the switching threshold of the inverters. The switching threshold of the inverters is influenced greatly by the process, voltage, and temperature variation. Calibration of the switching threshold

of the inverters at the start up is important to overcome the process variation problem. A binary search algorithm is used to select a word that controls the switching threshold of the inverters. Figure 2.39 and Figure 2.40 show the schematic of low and high switching threshold inverters used for calibration, respectively. The number and the size of the control transistors depend on the range of variations and the required resolution.

A duty cycle measurement circuit is used to measure the duty cycle of the extracted clock for a preamble of successive 1's and a preamble of successive 0's, which are used to calibrate the control bits of the high switching threshold inverter and the low switching threshold inverter, respectively.

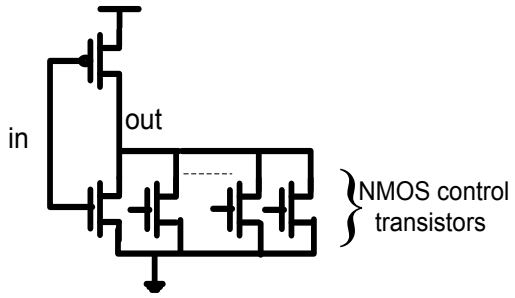


Figure 2.39 Calibrated low switching threshold inverter

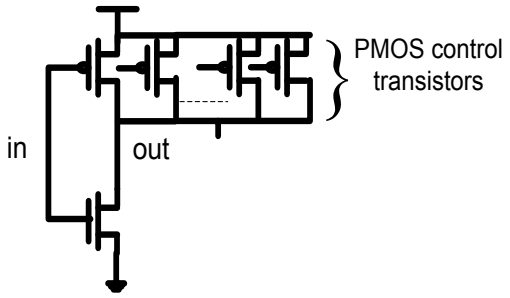


Figure 2.40 Calibrated high switching threshold inverter

2.6. A Comparison between the Three Discussed Signaling Schemes

Point of comparison	*Scheme 1 (conventional scheme)*	*Scheme 2 (proposed scheme)*	*Scheme 3 (proposed scheme)*
Signaling scheme	two-level	three-level Manchester	three-level
Transmission Line	differential	differential	single ended
Matching technique	resistive termination	source matching	resistive termination
Limitation on minimum frequency	No limitations	There is a minimum allowed data rate	No limitations
Bulky building blocks	CDR at the receiver Good jitter performance PLL at the transmitter	Simple circuits, no need for bulk blocks such as CDR	Simple circuits, no need for bulk blocks such as CDR
Jitter effect on data reliability	very sensitive to transmitted data jitter	insensitive to jitter accumulated during transmission	insensitive to jitter accumulated during transmission
Detector calibration	not needed	needed	needed
Driver size	small	large	very large
Maximum frequency of operation (GHz)	5	24	16
Maximum data rate (GHz)	10	12	16
Power	high	low	low
Design complexity	complex	simple	simple
Area	large	small	small

2.7. Simulation Results

2.7.1. Scheme 2 Simulation Results

Figure 2.41and Figure 2.42 show the eye diagram of the three-level code at the input and the output of the transmission line, respectively. It can be noticed that the amplitude of the signal at the input and the output of the transmission line is approximately the same. This results from the use of the source matching technique, where there is no matching at the receiver side (only a small capacitive load), the reflections will double the signal amplitude resulting in attenuation of only $(1-2e^{-\alpha l})$=16.8% instead of $(1-e^{-\alpha l})$=58%. Figure 2.43 shows the eye diagram of the extracted 12 GHz clock from the phase detector. Figure 2.44 shows the eye diagram of the extracted 12Gbps data stream. Although there is a significant amount of jitter in the received signal shown in Figure 2.42, the data was received and sampled correctly.

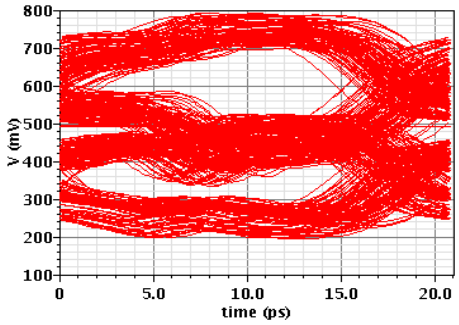


Figure 2.41 Eye diagram of the three-level code at the input of the transmission line

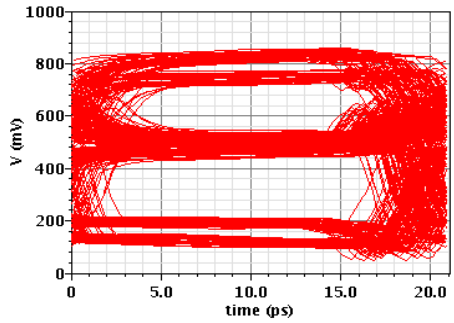


Figure 2.42 Eye diagram of the three-level code at the output of the transmission line

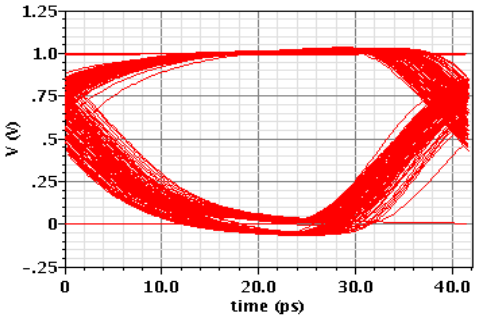


Figure 2.43 Eye diagram of the extracted clock from the phase detector

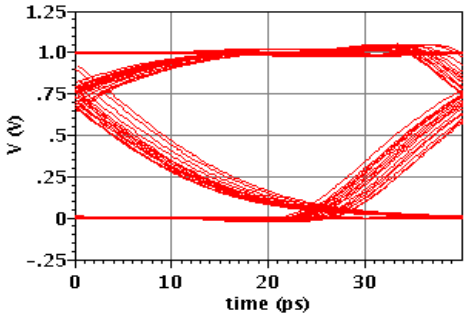


Figure 2.44 Eye diagram of the recovered data from the phase detector

Table 2.4 summarizes the total power and the power consumed by each building block in scheme 2 SerDes transceiver. The total power consumed is only 15.5mW, which is very small compared to that of conventional SerDes transceivers [13], [14].

Table 2.4 Power Consumption of scheme 2 componnents in typical conditions prelayout

	Component	***Power (mW)***
TX	ADPLL	4.5
	Serializer	4
	Line Driver	4
RX	Phase detector	2.5
	Deserializer	0.5
Total power		15.5

2.7.2. Scheme 3 Simulation Results

Table 2.5 summarizes the total power and the power consumed by each building block in scheme 3 SerDes transceiver. The total power consumed is only 18.1mWatt, which is very small compared to the conventional SerDes transceivers [13], [14].

Table 2.5 Power Consumption of scheme 3 components in typical conditions

	Component	***Power (mW)***
TX	ADPLL	2.9
	Serializer	1
	Line Driver	9.4
RX	Detector	4.2
	Deserializer	0.32
	Duty cycle correction circuits	0.33
Net power		18.1

2.8. Layout

Figure 2.45 and Figure 2.46 show the layout of the serializer and deserializer, respectively used in SerDes transceiver.

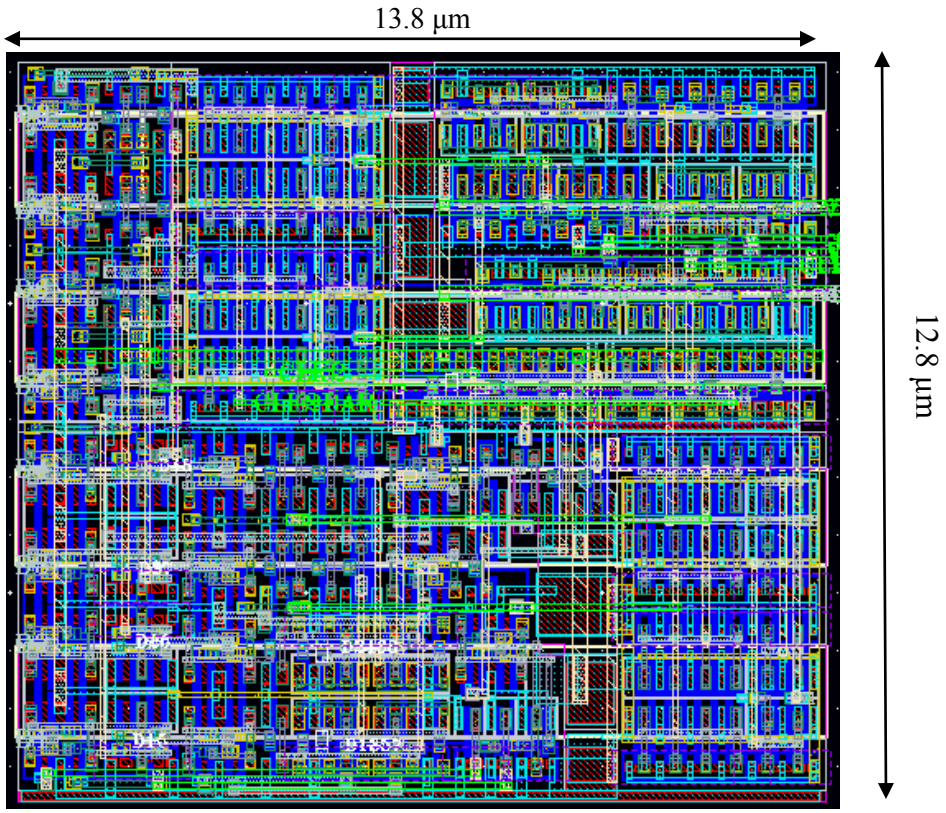


Figure 2.45 Serializer layout

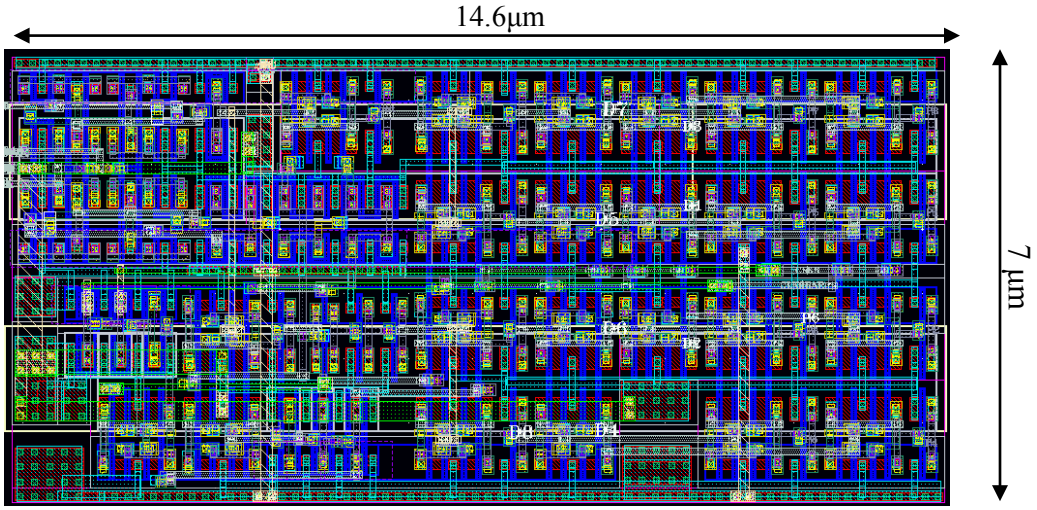


Figure 2.46 Deserializer Layout

2.9. Design Comparison

Table 2.6 compares between the results of this work and other published work in [13], [14]. The proposed signaling techniques in this work have proven a significant reduction in power and area. This is because the three-level coding techniques enable extracting the clock from the data using simple circuits rather than using complex power hungry blocks such as CDR.

Table 2.6 Design comparison of the proposed signaling techniques with other work in published literature

	This work		[13]	[14]
	Scheme 2	*Scheme 3*		
Technology	65nm	65nm	130nm	130nm
Core voltage (V)	1	1	1.2	1.2
Data rate (Gbps)	12	10	3.125	9
Power (mW)	15.5	18.1	100	600

Chapter 3

All Digital Phase Locked Loop

3.1. Introduction

Frequency synthesizers are used in RF applications to generate a high frequency periodic signal from a pure reference clock, generated from crystal oscillators. In our proposed SerDes transceiver, a high frequency clock signal is needed for the serializer at the transmitter to convert the parallel bit streams to a single serial bit streams. There are many approaches available to implement frequency synthesizers such as Direct Digital Frequency Synthesizers (DDFS) [25], [26], Delay Locked Loops (DLL) [27], [28] and Phase Locked Loops (PLL). However, the PLL is the common synthesizer used in RF applications.

The concept of phase locking was invented in the 1930s and found wide usage in electronics and communication. While the basic PLL has remained nearly the same since then, its implementation in different technologies and for different applications continues to challenge designers [29]. The block diagram of a conventional architecture of a PLL is shown in Figure 3.1. The Voltage Controlled Oscillator (VCO) is used to generate the high frequency output clock F_{out} determined by the input control voltage V_{LPF}:

$$F_{out} = F_o + K_{vco} * V_{LPF} \tag{3-1}$$

where V_{LPF} is the output voltage of the Low Pass Filter (LPF) and the input control voltage of the VCO. F_O is the free running oscillator frequency when the input controlled voltage V_{LPF} = zero. K_{VCO} is the VCO gain that depends on the design of the VCO.

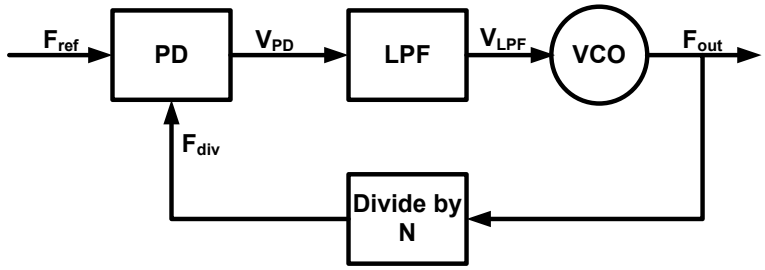


Figure 3.1 The conventional block diagram of PLL based frequency synthesizer

A PLL is a negative feedback system that compares the reference clock F_{ref} and the divided version of the output clock F_{div} ($F_{div} = F_{out}/N$) using a Phase Detector (PD). The PD is used to generate a voltage proportional to the phase error. The average of the PD output voltage generated from the LPF is used to control the output frequency of the VCO. At the lock state, $F_{out} = N \times F_{ref}$ and the phase difference between F_{ref} and F_{div} is very small, and approximately equals zero. In general, there are well known PLL specs that are used to evaluate the performance of a PLL:

1) ***Acquisition Lock Time***: It is the time needed for the PLL to lock the output frequency to the input frequency with minimum phase error.
2) ***Acquisition Lock Range***: It is the input frequency range that the PLL can detect and acquire lock. This is a very important spec, because frequency variation is expected due to the process, voltage, and temperature (PVT) variation. Therefore, the acquisition lock range should be wide enough to account for PVT variation.
3) ***Jitter***: It is the time uncertainty in the output clock period as shown in Figure 3.2, and it is measured in ps.
4) ***Phase noise***: Jitter in the time domain is translated into phase noise in the frequency domain as shown in Figure 3.2, and it is measured in dBc/Hz. Figure 3.2 shows how the output clock and its frequency response differs practically from the ideal case.

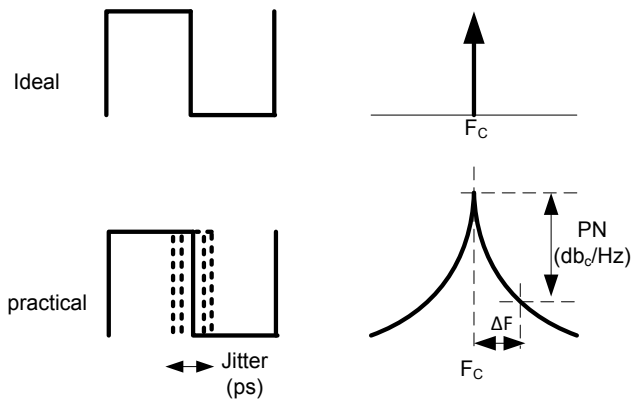


Figure 3.2 Jitter in time domain translated to phase noise in frequency domain

5) ***Bandwidth***: The bandwidth of a PLL determines the acquisition lock time, the acquisition lock range, and the phase noise of the PLL. There is extensive research effort spent in modeling the PLL and deriving expressions for the PLL transfer functions that show the impact of bandwidth on the other PLL specs [30], [31].
6) ***Power Consumption***: Nowadays we are talking about multiples of cores integrated on a single chip, so reducing power is important.
7) ***Portability***: The fast pace of technology scaling, forces the designers to find a quick and easy way to transfer the design from one technology to another. The digital design described using Hardware Description Language (HDL) is portable and desired for fast transfer.

The design of an RF PLL meeting all the predefined specs is very challenging. However, the specs that the PLL must meet depend heavily on the application, so there is no ideal implementation for frequency synthesizers. The required specs for the application determine the choice of the PLL architecture and the circuit design.

3.2. Motivation for All Digital PLLs

Analog PLLs may differ in their implementation, however they all have a charge pump based architecture [32]. In the charge pump based PLL shown in Figure 3.3 , the phase and frequency detector generate a pulse width modulated signal proportional to the phase error. Then, the charge pump generates a current proportional to this signal, and the analog loop filter converts this current to a voltage signal that controls the output frequency of the VCO. Many good jitter performance charge pump PLLs are published in recent literatures [33], [34], [35]. Usually achieving low jitter is at the expense of area and power. Increasing the current in the analog blocks of the PLL reduces the thermal noise at the expense of an increase in the power dissipation. In addition, in noisy applications, differential ended circuit design is used for noise immunity and good jitter performance at the expense of high area and power overhead.

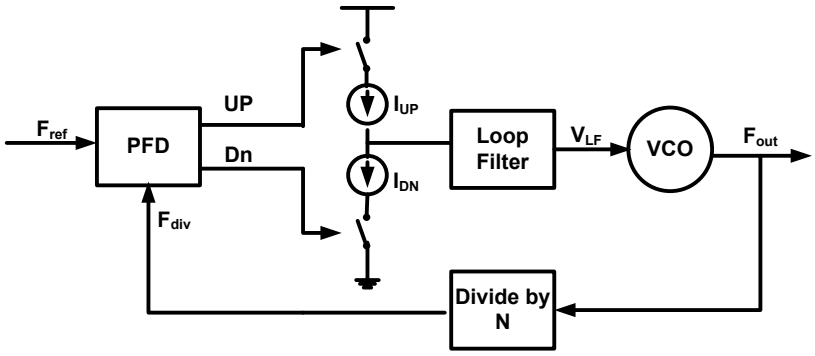


Figure 3.3 Block diagram of charge pump based PLL

3.2.1. Analog PLL vs. Digital PLLs

In an all digital PLL (ADPLL), the building blocks of the analog PLL shown in Figure 3.1 are replaced with digital blocks as shown in Figure 3.4 . The VCO is replaced by a digital controlled oscillator (DCO), the LPF is replaced by a digital loop filter (DLF) and in some architectures, a time to digital converter (TDC) replaces the charge pump. Despite research efforts spent in digital PLLs [20], [36], [37], [38], [39] , they were never practically implemented in RF applications. Although ADPLLs consume less power and area than analog PLLs, they still have a bad jitter performance compared to analog PLLs. The bad jitter performance of digital PLLs results from the limited DCO resolution. However, as technology scales, the design of analog PLLs encounters many challenges, which makes the digital PLL an attractive approach for many reasons [20], [37] . First, low voltage headroom reduces the performance of analog circuits and increases the impact of noise significantly. Second, leakage current increases in deep submicron, so the leaky MOS capacitors used in the implementation of loop filters need to be replaced by metal capacitors, which occupy more area. Third, current leakage leads to current mismatch in the charge pump, which in turn introduces ripples and increases jitter. Fourth, it becomes difficult to design a VCO with a good jitter performance, because the reduced voltage supply results in a highly noise susceptible voltage control.

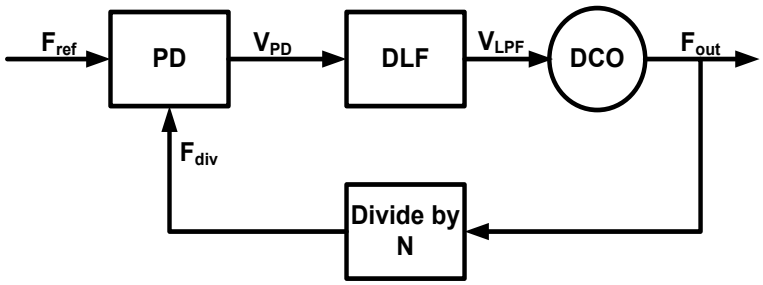


Figure 3.4 The block diagram of Digital PLL

On the other hand, ADPLLs overcome the mentioned analog PLL problems in deep submicron technologies and offer many advantages. First, the digital implementation reduces the sensitivity of the supply to noise. Second, the control bits of a DCO are in digital form, so the digital tuning voltage control of the DCO is less sensitive to noise and makes the design more robust to process, voltage and temperature variation. Third, the building blocks of ADPLLs are made of digital synthesizable blocks except for the DCO, so the design can be easily ported as technology scales. Fourth, the DLF occupies a small area as compared to the analog loop filter, because it does not contain any passive elements. Finally, the absence of analog blocks and passive elements reduces the area and power significantly.

3.2.2. Target Application Requirements

As mentioned in section 3.1, the specs of the PLL depend heavily on the target application. Hence, the application determines the choice of the PLL architecture, and the circuit design. The target of our PLL is to provide a high frequency square wave clock signal for the serializer and the encoder at the transmitter of the SerDes transceiver.

Table 3.1 SerDes transceiver frequency synthesizer requirements

Reference frequency (F_{ref})	100MHz
Output frequency range	5→10 GHz across PVT
Output frequency step	500 MHz
Jitter	Flexible somehow
Power & area	Minimum

The requirements of the target PLL summarized in Table 3.1. Given that, the reference clock frequency from the crystal oscillator equals 100MHz, the output frequency range is 5 to 10 GHz with a frequency step 500MHz. To get the frequency range; the proposed PLL should be

programmable. The new signaling scheme doesn't put a heavy requirement on the jitter specs, because it embeds the clock in the data, any phase shift in the data will be reflected in the extracted clock and the data will be sampled correctly. Therefore, the output clock jitter is accepted to a limit that does not affect the performance of the serializer. The PLL should be power and area efficient. Therefore, the best candidate for these specs is an ADPLL. A bang-bang ADPLL is chosen to achieve a fast and simple design, as will be shown later.

3.3. Bang-Bang ADPLL

The block diagram of a bang-bang ADPLL is shown in Figure 3.5 . The PLL consists of a binary phase detector (BPD), a digital loop filter (DLF), a digital controlled oscillator (DCO), and a feedback divider. The BPD determines if the divided clock leads or lags the reference clock and generates an *UP/DN* signal. The DCO is the heart of the PLL; it generates the output high clock frequency according to the input control word. The DLF is considered the brain of the PLL that receives the *UP/DN* signal from the BPD, and adjusts the control bits of the DCO according to its programmable coefficients α and β. The feedback divider is programmable to generate the target range of frequencies, such that $F_{out} = N \times F_{ref}$. For our application: the reference frequency F_{ref} equals to100 MHz, the output frequency F_{out} ranges from 5 to 10 GHz, with frequency resolution ΔF equals to 500MHz, then N ranges from 50 to 100 with a step of 5.

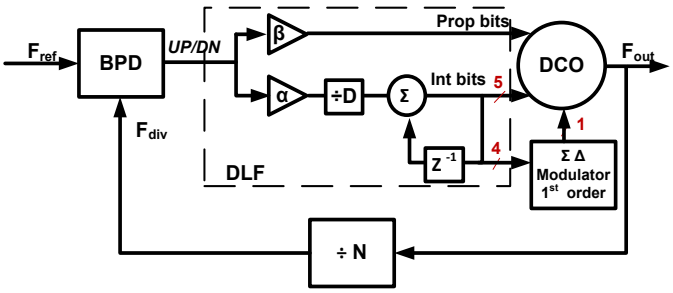


Figure 3.5 Block diagram of bang-bang ADPLL

3.3.1. Proposed BBADPLL Architecture

Figure 3.6 shows the block diagram of the proposed bang-bang ADPLL (BBADPLL). The proposed BBADPLL consists mainly of a DCO, a frequency divider and two loops: a frequency acquisition loop and a phase loop. The acquisition lock time is directly proportional to the initial frequency difference. Consequently, to reduce the acquisition time, a small initial frequency difference is achieved through the frequency acquisition loop. The frequency acquisition loop used during power up to select the DCO input coarse word, which in turn generates an output frequency close to the target frequency, but not identical. Hence, it reduces the initial frequency difference. After the frequency acquisition is achieved, the frequency loop generates a *frequency lock* as an enable signal for the phase loop. The phase loop tracks the phase difference between the reference clock and the divided clock using the PD, and tunes the fine input word of the DCO to account for temperature, and voltage variations.

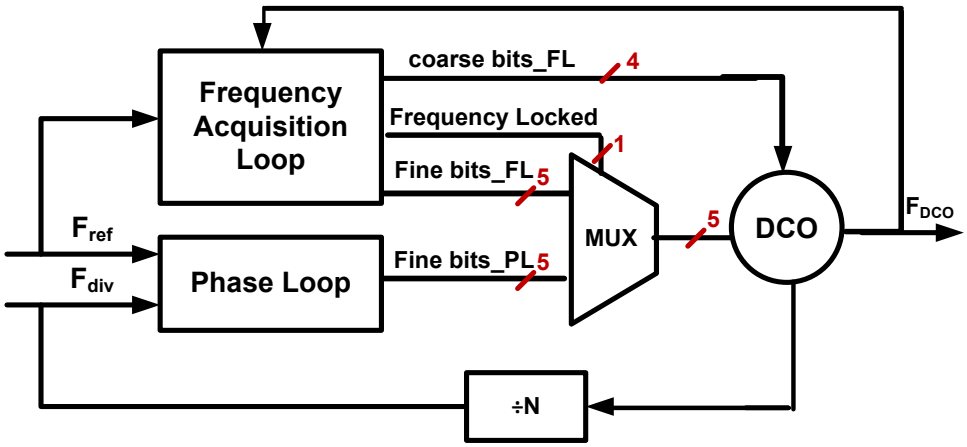


Figure 3.6 Architecture of the proposed BBADPLL, it consists mainly of frequency acquisition loop and phase loop

3.3.2. BBADPLL Building Blocks

Figure 3.7 shows the detailed block diagram of the proposed BBADPLL. The proposed BBADPL consists mainly of a DCO, a frequency divider and two loops: frequency acquisition loop and phase loop. The frequency acquisition loop consists of a frequency detector, a MUX, a thermometer encoder, a DCO, and a divide-by-five. The frequency acquisition loop compares between the frequencies of the DCO output clock and the reference clock, after that a binary search algorithm (BSA) is used to adjust the coarse word (4 bits) and the fine word (5 bits) of the DCO. This results in a frequency very close to the target frequency, but not identical. After the frequency acquisition is done, the phase loop is enabled. The phase loop consists of a binary phase detector (BPD), a DLF, a multiplexer (MUX), a thermometer encoder, a first order sigma delta (ΣΔ) modulator, a DCO, a divide-by-five, and a divide by *N*. The BPD detects whether the divided clock lags or leads the reference clock, and generates an *UP/DN* signal. The DLF receives the *UP/DN* signal and generates 12 bits: five fine bits, three proportional bits, and four fractional bits. The ΣΔ modulator employs a dithering scheme; it accepts the fractional bits at a low rate, and generates a single bit with a high rate.

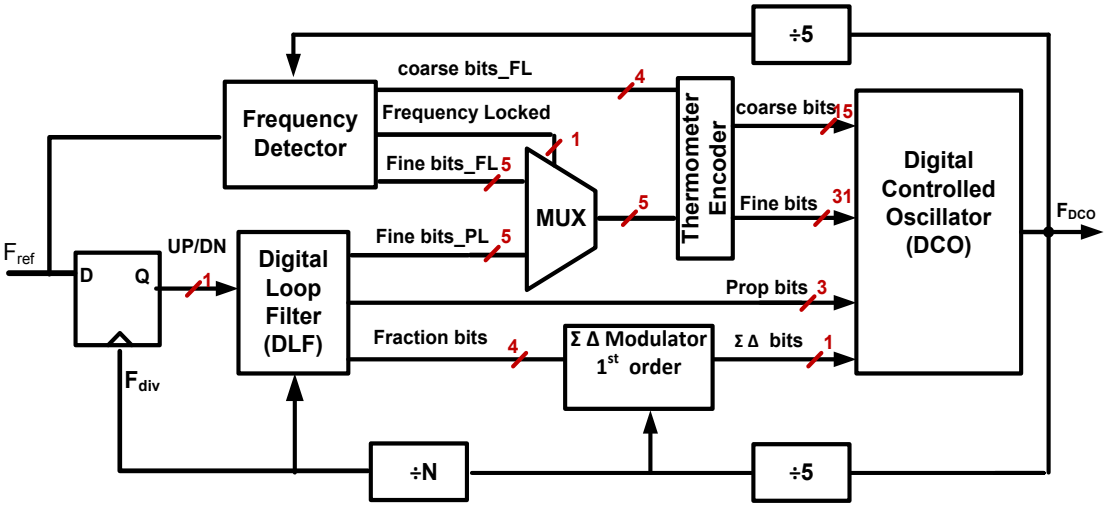


Figure 3.7 Detailed Block diagram of the proposed BBADPLL

3.3.3. Digital Controlled Oscillator (DCO)

The digital controlled oscillator (DCO) in an ADPLL is the replacement of the conventional VCO in analog PLLs. The control input of DCO is in digital form; hence, the noise immunity of these control digital bits is high compared to the analog control of the VCO.

The resolution of the DCO is the primary factor that determines the output clock jitter; therefore, the DCO is the most critical component in the ADPLL design. However, since the new signaling schemes embed the clock in the data, any phase shift in the data will be reflected in the extracted clock, and the data will be sampled correctly. Therefore, the output clock jitter associated with the limited resolution of the DCO is accepted to a limit that does not affect the performance of the serializer. Moreover, jitter in the output clock frequency of the oscillator will be reduced after the clock division at the transmitter. Since, this signaling scheme does not put high constraints on jitter specs; the ring oscillator is the best choice. Ring oscillators have a wider range, and a lower cost than LC oscillators [40]. Since the target frequency should be covered over all possible operating conditions, the wide tuning range of the ring oscillators is required to compensate for the variation of the center frequency across PVT.

There are two popular design techniques for a ring oscillator based DCO. The first technique is changing the driving strength of the MOSFET by controlling the current. The other technique is tuning a shunt capacitor load for different output frequencies. There are many ways of applying these techniques on a DCO design published in the literature [41], [42], [43], [44]. The first technique mentioned is used in our DCO design.

Figure 3.8 shows the block diagram of the proposed simple DCO. It consists of three inverter stages sharing the same fine and coarse tune transistors. Figure 3.9 shows the differential inverter

based ring oscillator. Cross-coupled inverters are used to guarantee a differential output. The ring oscillator generates a clock whose frequency is controlled by four binary coarse bits for a coarse frequency step, and five binary fine bits for a fine frequency step. Figure 3.10 shows the proposed DCO output frequencies for different coarse and fine words at a typical process corner, 1V and 125°c. There are 15 curves; each curve represents the output DCO frequency versus the fine word for a given coarse word.

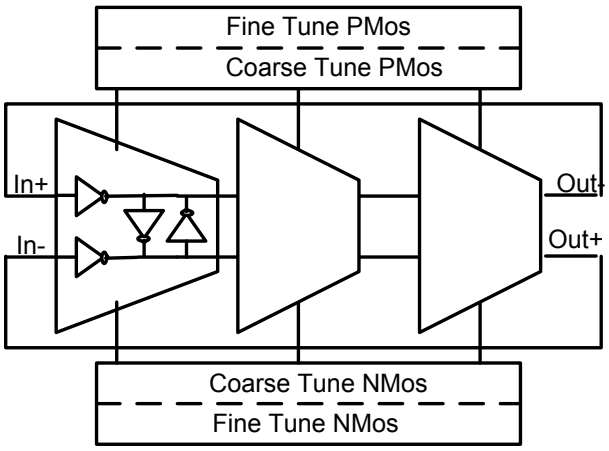


Figure 3.8 Block diagram of the proposed DCO

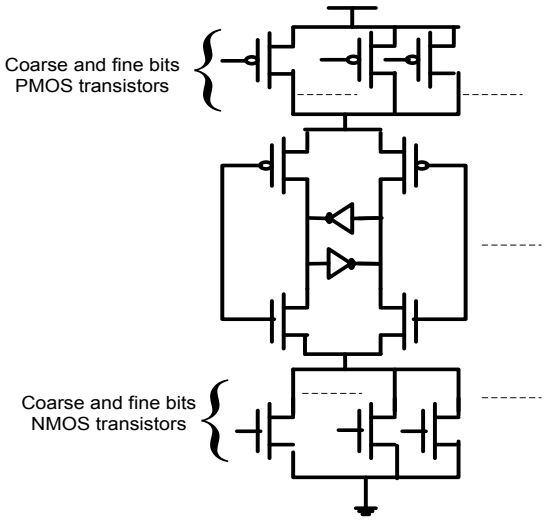


Figure 3.9 DCO cell schematic

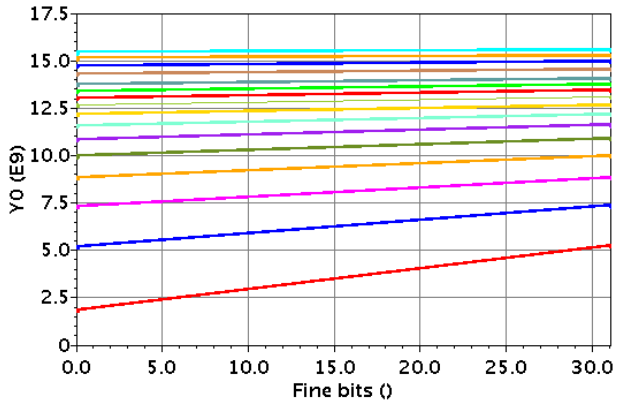


Figure 3.10 Post-layout simulation results showing the output frequency curves for different coarse and fine words

The output frequency of the digital controlled oscillator is given by:

$$F_{out} = F_o + K_{fine}W_{fine} + K_{coarse}W_{coarse} \quad (3\text{-}2)$$

where F_{out} is the DCO output frequency, W_{Coarse} and W_{Fine} are the DCO input coarse and fine word, respectively. K_{Coarse} and K_{Fine} is the DCO gain for the coarse and fine words, respectively. F_o is the oscillator free running frequency at $W_{Coarse} = 0$ and $W_{Fine} = 0$. The linearity of the DCO curves will help define a constant K_{Fine} for each coarse word. K_{fine} defines the frequency step upon which the programmable DLF coefficients are adjusted. The length of the fine bits transistors is set to non-minimum length, to achieve a linear frequency response and improve the DCO resolution for better jitter performance.

Coarse tuning bits provide coarse frequency steps for a broad range of frequencies that account for the PVT variation. Fine tuning bits provide fine frequency steps for a narrow range of frequencies that account for the phase tracking mode. The oscillator is controlled by a thermometer code rather than a binary code to ensure monotonic increase. Thermometer decoder is used to convert the binary bits to thermometer bits.

3.3.4. Frequency Acquisition Loop

The frequency acquisition loop consists of a frequency detector, a MUX, a thermometer encoder, a DCO, and a divide-by-five. The divide-by-five is used as a frequency prescaler for the high output DCO frequency, so that the frequency detector can operate at the prescaled frequency and digitally implemented. The frequency detector shown in Figure 3.11 consists of a 6-bit counter, a control unit, a comparator, and a BSM state machine. The counter counts the number of the DCO clock cycles in one reference clock cycle. Then, the comparator compares between the number of the DCO clock cycles and *N*, and accordingly generates the *FAST/SLOW*

signal. The binary search algorithm accepts the *FAST/SLOW* signal, and selects the coarse word that results in an output frequency close to the target frequency.

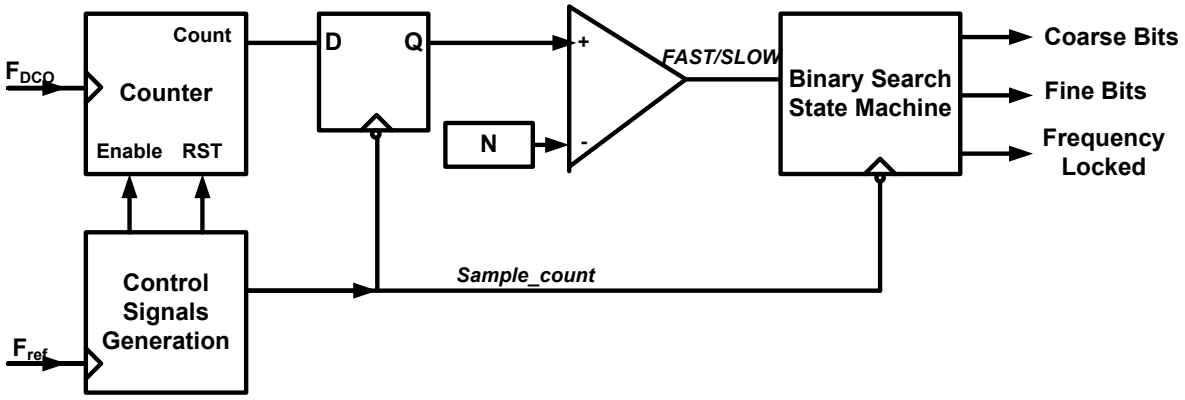


Figure 3.11 Block diagram of the frequency detector

3.3.4.1. Counter and Comparator

A 6-bit counter is used to count the number of the DCO output clock cycles in a single reference clock period. Since the counter is implemented using standard cells, it cannot operate at such high frequency (10GHz). Thus, the DCO output clock is prescaled by dividing it by five.

The counter is based on three states: reset, count, and hold. The two control signals; 'enable' and 'reset' are generated from the reference clock, and are used to create the three states. Therefore, the counter is asynchronous; it operates at two different clock domains: the high DCO clock frequency, and the reference frequency. Metastability is an inherent property in any asynchronous block. In a metastable condition, the circuit is unable to settle at a defined logic value. This metastability state may lead to an error in the number of cycle's count, which may result in an output frequency far from the target frequency. A large initial frequency difference will increase the acquisition lock time, and may lead to instability. Synchronizers are designed to

resolve the metastability, and reduce the time of an undefined state. As shown in Figure 3.12, two D flip-flops are clocked with the high clock frequency, and are used to synchronize the generated 'enable' and 'reset' signals with the DCO frequency. The expected maximum count error is ±1.

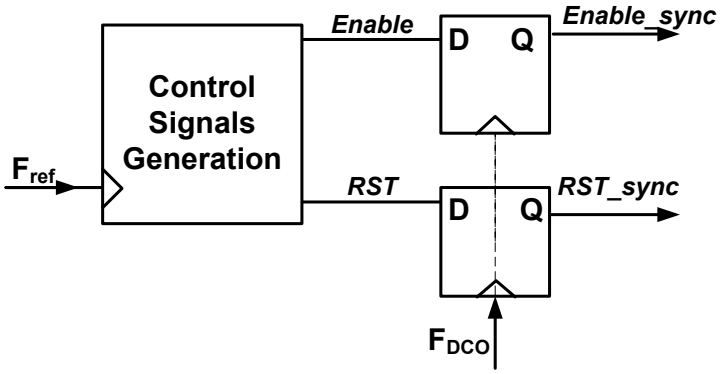


Figure 3.12 Frequency detector synchronizer used to resolve metastability, and reduce the period of undefined state.

3.3.4.2. Control Signals

Figure 3.13 shows the control signals generated from the reference clock, the counter different status, and the comparator status. There are three control signals generated from the reference clock: *Enable*, *RST*, and *Sample_Count* . The *Enable* and *RST* signals create the three states of the counter, as summarized in Table 3.2. In the first reference clock cycle; *Enable = 1* and *RST = 0*, so the counter resets for a reference clock cycle. When *RST* is set to *1* in the second reference clock cycle, the counter starts counting the number of DCO clock cycles in one reference clock period. After that, the *Enable* signal goes low, and the counter holds the cycles count for another reference clock cycle. *Sample_Count* signal is used to sample the DCO cycles count in the middle of the hold state. Thus, the comparator operates at a low frequency $F_{ref}/3$.

Table 3.2 Three states of the counter

State	*Enable*	*RST*
reset	1	1
counting	1	0
hold	0	x

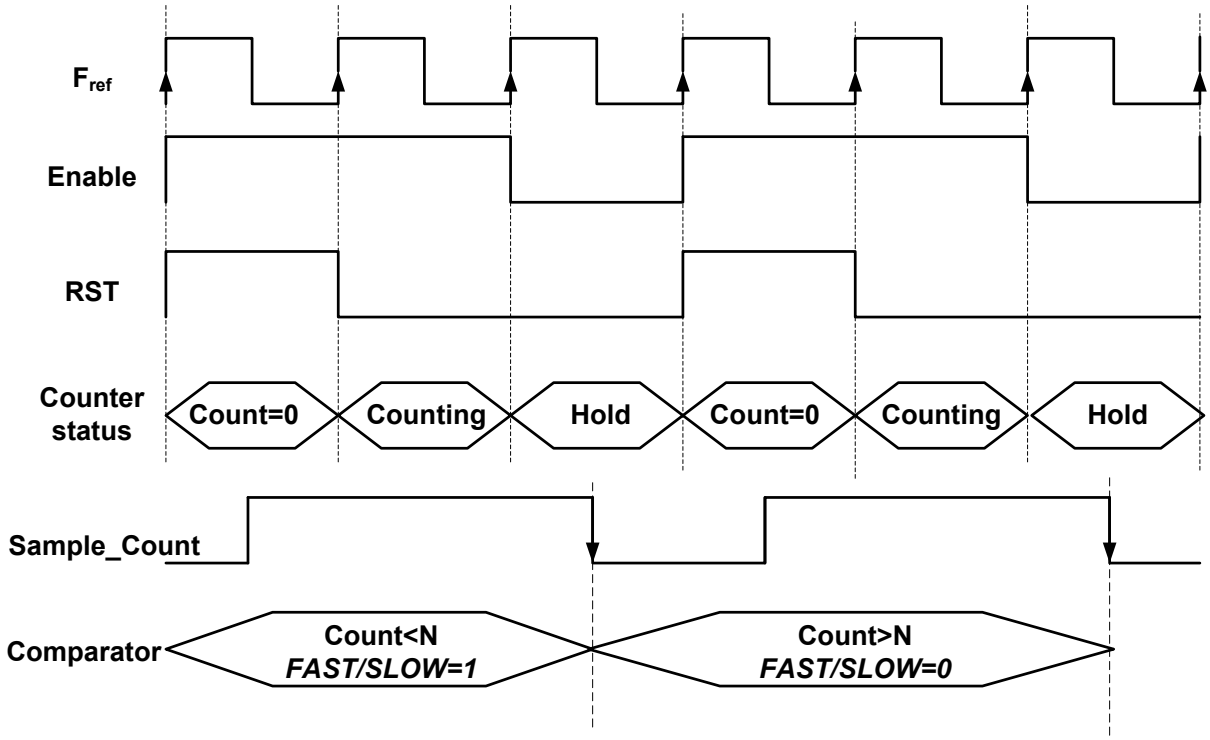


Figure 3.13 Control signals of the counter; 'reset' and 'enable' signals, counter status and comparator output

3.3.4.3. Binary Search Finite State Machine

A binary search algorithm (BSA) acts as a frequency search algorithm [45], [46], [47] to achieve a faster lock in time. The acquisition lock time is directly proportional to the initial frequency difference. Consequently, to reduce the acquisition time, a small initial frequency difference is achieved through the binary search algorithm. The binary search algorithm searches for the coarse, and fine bits that achieve an output frequency very near to the target frequency. It takes four reference clock cycles to find the coarse word, and five reference clock cycles to find the fine word. Since the comparator operates at the rate of $F_{ref}/3$, the frequency lock time is 27 reference clock cycles.

Figure 3.14 shows the state diagram of the binary search algorithm. *S0* is the initial state, where the coarse and fine bits are initialized to be in the middle of the range *1000* and *10000*, respectively. *S1*–*S4* determine the coarse bits and *S5*–*S9* determine the fine bits. In each state, the algorithm determines a bit, starting from the most significant bit to the least significant bit.

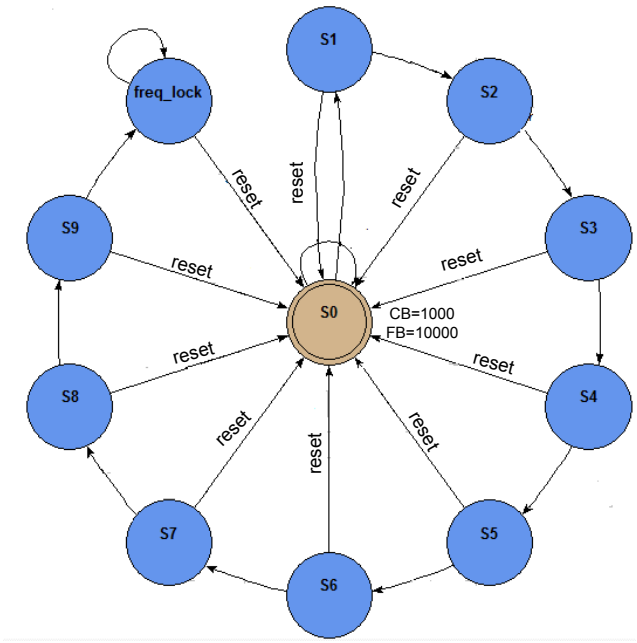


Figure 3.14 State machine of the binary search algorithm

3.3.5. Phase Loop

3.3.5.1. Phase Detector

A single flip-flop shown in Figure 3.15 is used as a Binary Phase Detector (BPD) that generates an *UP/DN* signal to indicate whether the divided DCO clock frequency leads or lags the reference clock frequency. Figure 3.15 shows the transfer function of BBPD (the *UP/DN* signal versus the phase difference).

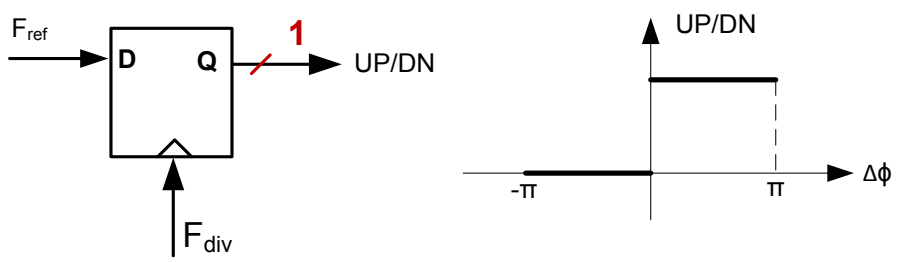


Figure 3.15 (a) Single flip flop act as binary phase detector (b) Output transfer function of BPD

3.3.5.2. Frequency Divider

The frequency divider is used in the feedback path of the PLL, dividing the DCO output clock frequency by *N*. The frequency divider needs to be programmable to generate a range of output frequencies. At each subsequent stage of the frequency divider, the speed is lower and so is the power. Since we are planning to synthesize the ADPLL digital blocks, a frequency prescaler is needed to reduce the maximum operating frequency of the standard cells. The frequency prescaler is the divider block that the oscillator drives. This frequency prescaler is a challenging block, because it operates at the oscillator frequency.

The output frequency range is 5-10GHz with 500MHz frequency step needs a division ratio range from 50 to 100 with a step of five. Thus, a divide-by-five is used as a frequency prescaler

followed by a digital synchronous programmable divider with a range from 10 to 20. Figure 3.16 shows the schematic of the prescaler block (divide-by-five). The prescaled clock is used as the input clock for the frequency acquisition loop, and the first order ΣΔ modulator.

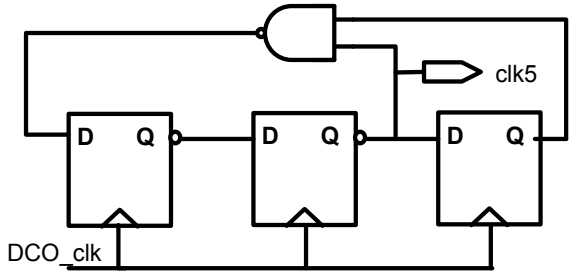


Figure 3.16 Divide-by-five; frequency prescaler

3.3.5.3. Sigma Delta Modulator

The fractional part of the DLF integral path bits employs a time averaged dithering mechanism to increase the frequency resolution. Dithering is performed by a digital Sigma Delta (ΣΔ) modulator that accepts the fractional bits and produces high rate integer stream, whose average value equals to the low rate fractional input [48]. The ΣΔ modulator is considered an essential block to improve the DCO frequency resolution, and randomize the output frequency components that results in spurious tone.

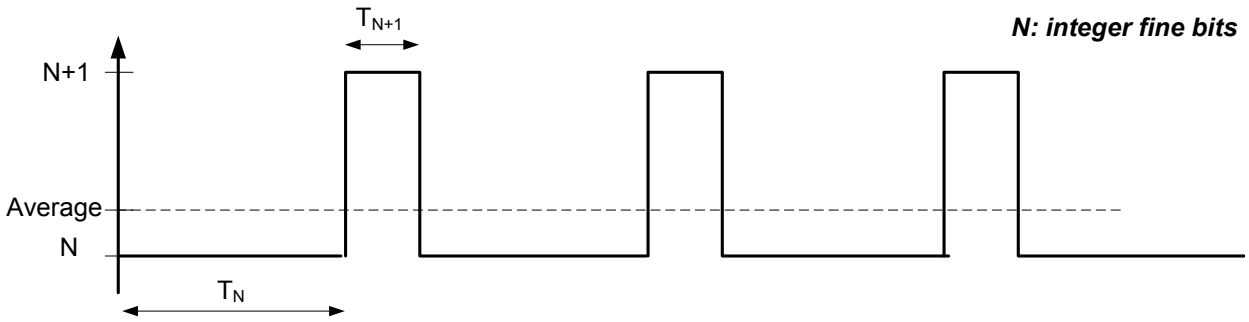


Figure 3.17 DCO dithering by changing the fine bits at high rate

Figure 3.17 reveals the principle in which the integer fine bit is periodically altered from N to N+1. The resulting average fine bit is

$$= \frac{NT_N + (N+1)T_{N+1}}{T_N + T_{N+1}}$$

$$= N + \frac{T_N}{T_N + T_{N+1}} \tag{3-3}$$

where N represents the integer bits, T_N is the period that the control fine word equals to N, T_{N+1} is the period that the control fine word equals to $N+1$ and $\frac{T_N}{T_N+T_{N+1}}$ is equivalent to the fractional bits.

Figure 3.18 shows the block diagram of the first order ΣΔ modulator. It is accumulator based [49], and consists of a 4-bit adder and D-FFs. The accumulator in our design accepts four fractional bits, and operates at the prescaled frequency $F_{DCO}/5$. The accumulator value X_n increases by the value of the input fractional bits every clock cycle until it reaches the maximum value. After that, the accumulator will overflow and set the *Carry*. *Carry* is the *ΣΔ_bit* that is used to control the fine bits for dithering. For example, if the fractional bits = *0100* i.e. 4, the accumulator adds an increment value of 4 every fast clock cycle, until it overflows and sets the *carry*. Therefore, the carry out will be high for one cycle and low for 3 cycles generating a

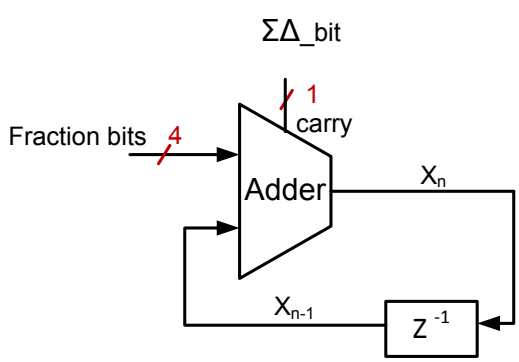


Figure 3.18 Block diagram of ΣΔ modulator, it consists of adder and D-FFs

fraction of ¼. The accumulator value and its carry are summarized cycle by cycle for the fractional bits *0100* and *0110* in Table 3.3 and Table 3.4, respectively.

Table 3.3 Example illustrates the operation of 1st order ΣΔ modulatorwith fractional bits =0100=1/4

Clock cycle *n*	1	2	3	4	5	6	7	8	9	0	11	12	13
X_n	0	4	8	12	0	4	8	12	0	4	8	12	0
X_{n-1}	N/A	0	4	8	12	0	4	8	12	0	4	8	12
carry/ ΣΔ_bit	1	0	0	0	1	0	0	0	1	0	0	0	1

Table 3.4 Example illustrates the operation of 1st order ΣΔ modulatorwith fractional bits =0110=3/8

Clock cycle *n*	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
X_n	0	6	12	2	8	14	4	10	0	6	12	2	8	14	4	10	0
X_{n-1}	N/A	0	6	12	2	8	14	4	10	0	6	12	2	8	14	4	10
carry/ ΣΔ_bit	1	0	0	1	0	0	1	0	1	0	0	1	0	0	1	0	1

The dithering rate has to be higher than the update cycle of the fraction bits, and the resolution of the time-averaged value relies on the dithering speed. Hence, there is a proportional relationship between the frequency resolution improvement and the dithering rate. Therefore, it is preferable to clock the ΣΔ modulator with the highest possible frequency. The limitation here is in the digital design of the ΣΔ modulator; the maximum frequency can be achieved using the standard cells. In our design, the ΣΔ modulator is clocked with $F_{DCO}/5$.

3.3.5.4. Digital Loop Filter (DLF)

The digital loop filter (DLF) is the second most challenging block in the DPLL design, because the loop dynamics of the DPLL is defined through the DLF coefficient. Since, the DLF coefficients are programmable; the loop dynamics are also programmable, which is a very interesting property not found in analog loop filters. The binary phase information is fed from the

BPD into the DLF. The DLF is clocked by the divided clock F_{DIV}, which is in a locked condition synchronized with the reference clock F_{ref}.

The proposed DLF is shown in Figure 3.19; it consists of proportional and integral paths with coefficients β and α, respectively. The proportional path is a direct path from the output of the phase detector to the control bits of the DCO. The proportional path is important for loop stability, because the integral path is slow in a way that the clock can go out of phase before the integral paths respond.

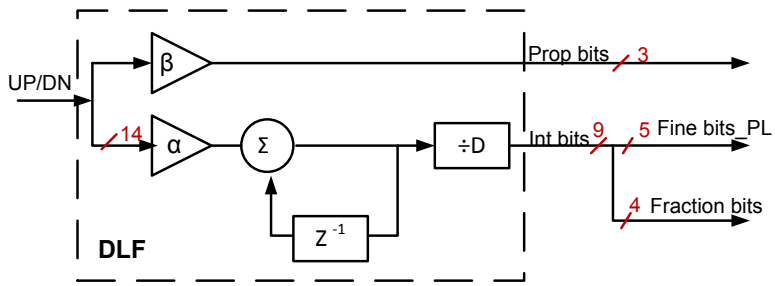


Figure 3.19 Block diagram of the digital loop filter

The presence of the binary phase detector introduces non-linearity in the PLL loop. Despite the efforts and the research spent on deriving a linear model for the bang-bang ADPLL in the literature [50], [51], there is no definite design methodology for the BBADPLL. Usually, the nonlinear systems are just simulated massively in order to derive their performance. The DLF coefficients α and β are programmable and their values are designed to depend on the coarse word i.e. the slope of DCO curves.

The proportional path accepts a single bit *UP/DN* from the BPD and generates a thermometer code depending on the value of β. The integral path accepts a single bit *UP/DN* from the BPD and generates nine bits; the five MSB are the fine bits, the four LSB are the fractional code. The coefficient *D* determines the update rate of the integral word. The coefficient *D* is programmable

to be either one or two; D= 1 means that the integral word is updated every one reference clock cycle, D= 2 means that the integral word is updated every two reference clock cycle. The coefficient D is important for loop stability. The integral path coefficient α determines the integral path gain and is controlled through shifting the bits to the right or left.

3.3.6. Proposed BBADPLL Model

The presence of a binary phase detector introduces a non-linearity in the PLL loop. Thus, the traditional Laplace transform that is used in modeling the linear PLLs is no longer valid. Usually, the nonlinear systems are just simulated massively in order to derive their performance. However, in this thesis, Useful expressions that model the DLF are presented for the design and optimization of the programmable DLF coefficients. The derived expressions for the PLL model give an insight on the effect of changing the DLF coefficients on the loop dynamics. The proposed DLF based BBADPLL model is shown in Figure 3.20.

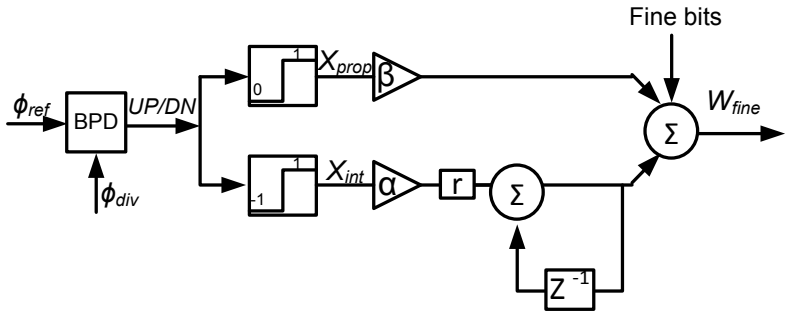


Figure 3.20 BBADPLL Model

The output angular frequency of the DCO at time t is:

$$\omega_{DCO}(t) = \omega_o + \Delta\,\omega(t) \tag{3-4}$$

where ω_o is the initial DCO output angular frequency resulted from the frequency acquisition loop, $\Delta\omega$ is the frequency step resulted in response to the input phase error. The output phase $\emptyset_{Div}$ of the divided frequency is given by:

$$\emptyset_{Div} = \frac{1}{N}\int_0^{\infty} \omega(t)\, dt \tag{3-5}$$

where N is the division ratio. The time domain of the output phase $\emptyset_{Div}$ of the divided frequency is converted to the s-domain using Laplace transform as follows:

$$\emptyset_{Div}(s) = \frac{1}{N}\left(\frac{\Delta\, \omega(s)}{s}\right) \tag{3-6}$$

X_{prop} and X_{int} are defined as following:

$$X_{prop} = \begin{cases} 0 & if\ up/dn = 0 \\ 1 & if\ up/dn = 1 \end{cases} \tag{3-7}$$

$$X_{int} = \begin{cases} -1 & if\ up/dn = 0 \\ 1 & if\ up/dn = 1 \end{cases} \tag{3-8}$$

since, the control inputs X_{prop} and X_{int} are piecewise constant over the sampling period $Ts=1/F_{ref}$, where F_{ref} is the reference frequency. Zero-order hold is used to convert from continuous-time to discrete-time model as follows:

$$\emptyset_{Div}(z) = \frac{T_s}{N}\frac{\Delta\, \omega(z)}{(1 - z^{-1})} \tag{3-9}$$

The z-domain of the fine word step ΔW_{fine} results in response to the transfer function of the digital loop filter is given by:

$$\Delta W_{fine}(z) = \beta X_{prop} + \frac{\alpha r}{1 - z^{-1}} X_{int} \tag{3-10}$$

where β is the proportional path gain, α is the integral path gain, and r is the fractional word step.

The frequency change $\Delta\,\omega(z)$ is given by:

$$\Delta\,\omega(z) = 2\pi K_{fine}\Delta W_{fine}(z) \tag{3-11}$$

From (3-9), (3-10), and (3-11), the output phase $\emptyset_{Div}\,(z)$ can be found as:

$$\emptyset_{Div}\,(z) = \frac{2\pi\,K_{fine}}{Nf_{ref}}.\frac{\beta X_{prop}(1-z^{-1}) + \alpha r X_{int}}{(1-z^{-1})^2} \tag{3-12}$$

The z-domain of the output phase $\emptyset_{Div}$ given by (3-12) is converted to discrete time domain as follows:

$$\emptyset_{Div}\,[n] = \frac{2\pi\,K_{fine}}{Nf_{ref}}\begin{pmatrix} \beta(X_{prop}[n] - X_{prop}[n-1]) \\ +\alpha r X_{int}[n] \\ -\emptyset_{Div}\,[n-2] + 2\emptyset_{Div}\,[n-1] \end{pmatrix} \tag{3-13}$$

The output phase error $\Delta\emptyset_{Div}$ in discrete-time domain is given by:

$$\Delta\emptyset_{Div}\,[n] = \emptyset_{Div}\,[n] - \emptyset_{Div}\,[n-1] \tag{3-14}$$

The derived expressions of the output phase $\emptyset_{Div}$ and the phase error $\Delta\emptyset_{Div}$ presented in (3-13) and (3-14), respectively, have been analyzed using Matlab scripts and verified on ADPLL model using Modelsim. The DLF programmable coefficients α and β are adjusted based on the DCO frequency step i.e. K_{fine}. Figure 3.21 shows the effect of changing β on the acquisition lock time and the jitter performance. For a defined coarse word i.e. DCO frequency step K_{fine}, increasing β reduces the lock in time, but increases the output jitter Figure 3.22 shows the linear dependence of the output RMS jitter on β. Increasing β, increases the output RMS jitter linearly; hence, large value of β may lead to loop instability (overdamped response).

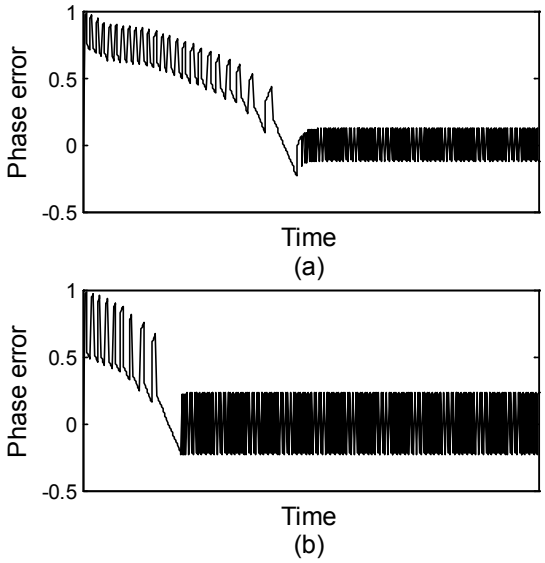


Figure 3.21 phase error transient response for coarse word=1 (a) β=1 (b) β=2

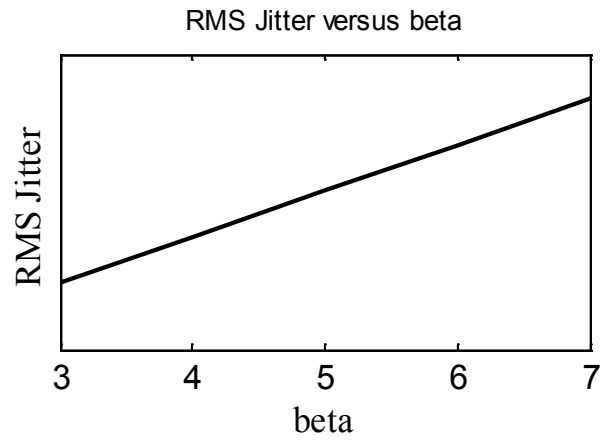


Figure 3.22 The effect of increasing β on the output RMS jitter

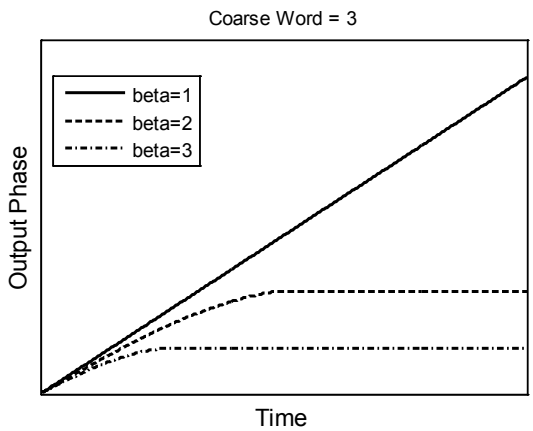


Figure 3.23 Effect of β on loop stability for coarse word=3

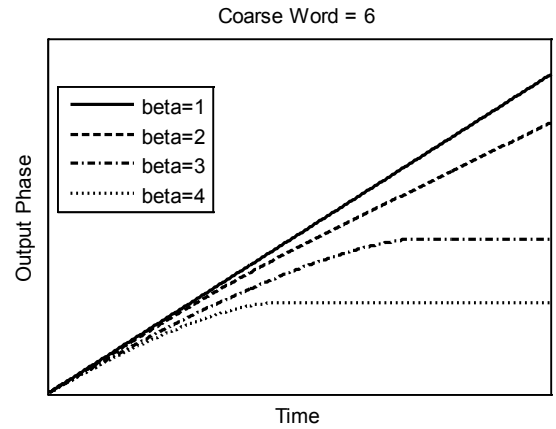


Figure 3.24 Effect of β on loop stability for coarse word=6

Figure 3.23 and Figure 3.24 show the output phase vs. time for different values of β for coarse word equals to 3 and 6, respectively. It shows the effect of changing β on loop stability for different DCO frequency steps. Figure 3.23 shows that for coarse word equals to three, the minimum value of β for a stable loop equals to two, while Figure 3.24 shows that for a coarse word equals to six, the minimum value of β for a stable loop equals to three. Hence, the

minimum value of the proportional coefficient β for a stable loop depends on the DCO frequency step i.e. DCO fine bits gain K_{fine} i.e. coarse word.

Figure 3.25 shows the output phase vs. time for different values of α. For a given coarse word i.e. DCO slope K_{fine}, as α increases, the output phase starts to oscillate. Since $F_{out} = d\emptyset/dt$, the oscillations in the output phase resulted in variations in the output frequency, hence, the output frequency not more locked to the target frequency. Thus, for a given frequency step and β, there is a minimum value of α to guarantee loop stability. The coefficient D enhances the loop stability by reducing the rate of the integral path response.

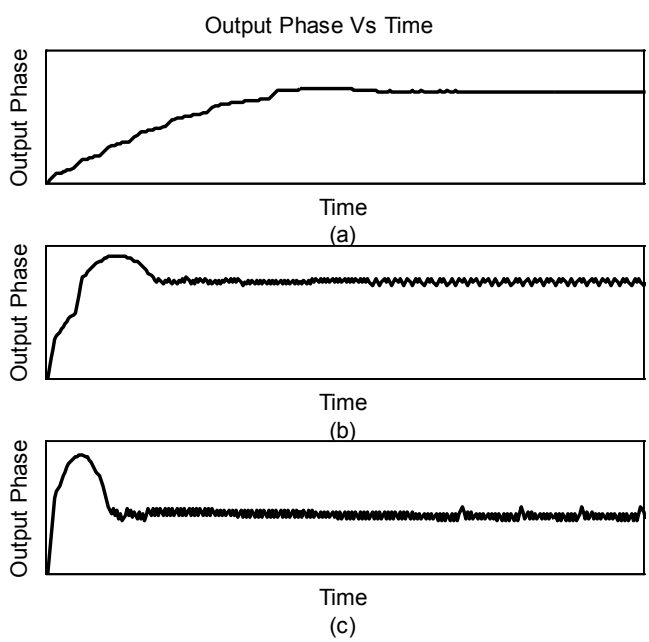


Figure 3.25 The effect of changing α on the output phase transient response for coarse word=1, β=3 (a) α=0.25, (b) α=1, (c) α=2

Intuitively, we can summarize the effect of α, β, and D as following: The small value of the proportional coefficient β may lead to loop instability (underdamped response), because the loop responds slowly to the phase difference. Increasing the value of β will stabilize the loop, but will increase the jitter. Large values of β may also cause loop instability (overdamped response).

Hence, there is an optimum value of β, which stabilizes the loop and minimizes the jitter. Moreover, β plays an important role in determining the acquisition lock time. The tradeoff in β can be described as following: the acquisition lock time is inversely proportional to β, while jitter is directly proportional to β. Thus, increasing β increases the jitter and reduces the lock in time, and vice versa. For stability: the α to β ratio should be small, so if β is increased, α should be reduced to avoid overdamped response to the phase difference. The coefficient D enhances the loop stability by reducing the rate of the integral path response. The methodology of our design is to program α, β, and D according to the DCO frequency step (K_{Fine}) i.e. the coarse word. The frequency step will determine α, β, and D for a stable loop with minimum jitter.

3.4. Simulation Results

Mixed signal simulations on cadence for ADPLL may take one or two days, while Modelsim only takes about 5 minutes in each run. The linearity of the DCO curves was important to develop an equivalent model very close to the actual ADPLL. For DCO modeling: a linear equation is generated for each curve (coarse word) in the form presented in (3-2). The equivalent model used in Modelsim had a response very close to the actual ADPLL response. Therefore, initially when we started to design the DLF coefficients required for loop stability and minimum lock in time, we needed a very fast simulator. Modelsim is used to verify the functionality of the digital blocks and optimize the DLF coefficients for a stable loop. After adjusting the coefficients of the DLF, the loop is simulated using the mixed signal simulator in cadence to verify Modelsim simulations.

The proposed BBADPLL was implemented using a TSMC CMOS 65nm technology with a frequency range 5−10GHz and a frequency resolution of 500MHz.

3.4.1. DCO Frequency Curves and The Equivalent Power Consumption

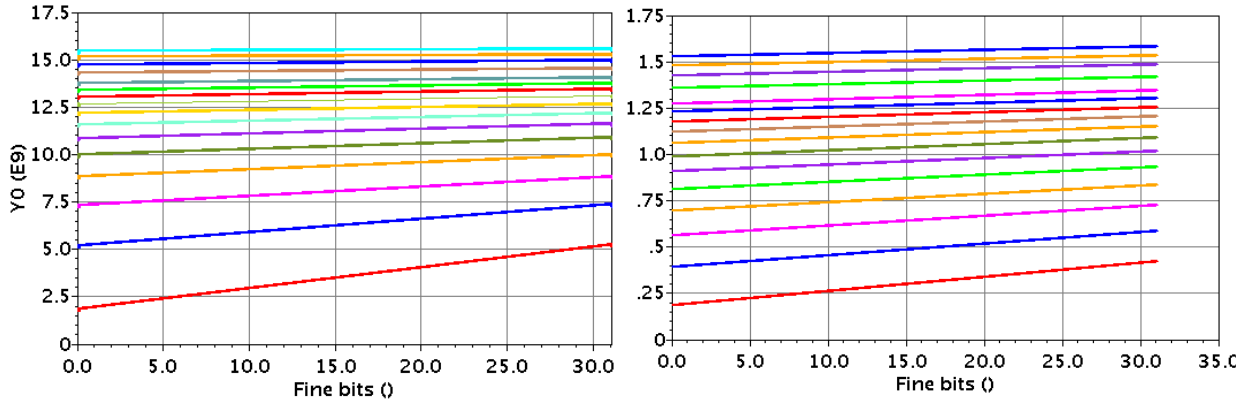
Output frequency (GHz) vs. fine word & Coarse word

DCO power consumption vs. fine word & Coarse word

TT, 1V, 125°c

2 → 15GHz

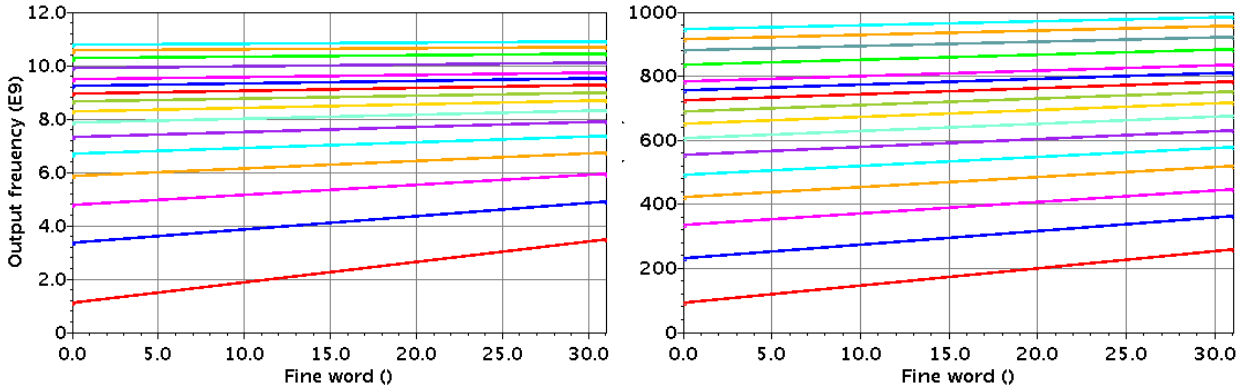
0.2 → 1.6mW



SS, 0.9V, 125°c

1 →11GHz

0.1 → 1mW



FF, 1.1V, 0°c

3 → 23GHz

0.4 → 2.5mW

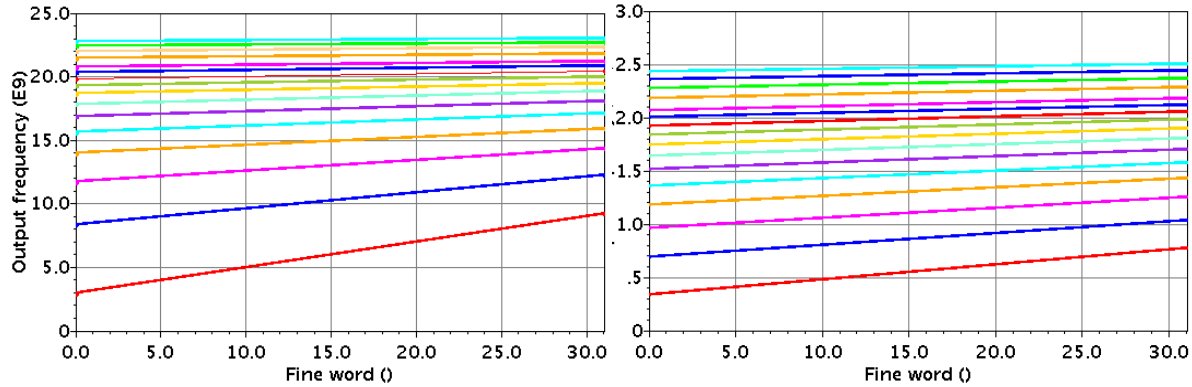


Figure 3.26 DCO output frequency curves and power consumption for different coarse and fine words across PVT

3.4.2. ADPLL Closed Loop Simulation Results

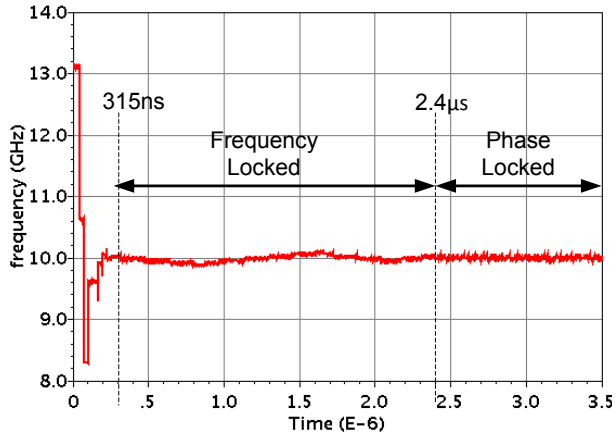


Figure 3.27 Closed loop post layout simulations for BBADPLL

Figure 3.27 shows the ADPLL closed loop simulation for a 10GHz DCO output frequency at a typical process corner, 125°c, and 1V. It takes 315ns (30 reference clock cycles) to acquire a frequency lock state, and 2.4µs to acquire a phase lock. The frequency lock time depends on the binary search algorithm. The phase lock time depends on the DLF programmable coefficient α, β, and D. Table 3.5 summarizes the simulation results for output clock frequencies of 7.5GHz and 10GHz. The total power consumed at 10GHz is only 2.7mWatt. The peak to peak period jitter, and the RMS jitter at 10GHz is 1.49ps and 0.19ps, respectively. The total area of the proposed ADPLL is only 4372µm^2, which is very small compared to other published architectures [52], [53], [54], [55].

Table 3.5 The closed loop simulation results at 7.5 and 10GHz

Output frequency (GHz)	7.5	10
Acquisition lock time (µs)	1.6	2.4
Peak to peak jitter (ps)	3.29	1.49
RMS jitter (ps)	0.58	0.19
Total power (mW)	2.3	2.7
Total area (µm^2)	4372	

3.5. Layout

3.5.1. Full Custom Blocks

There are three full custom blocks in the ADPLL: the DCO, the frequency prescaler (divide-by-five), and the output clock buffer.

3.5.1.1. Floor Plan

Figure 3.28 shows the floorplan of the proposed BBADPLL. The digital blocks were synthesized using Synopsys and automatically placed and routed using SoC Encounter, while the layout of the DCO, frequency prescaler and the clock buffer is a full custom layout.

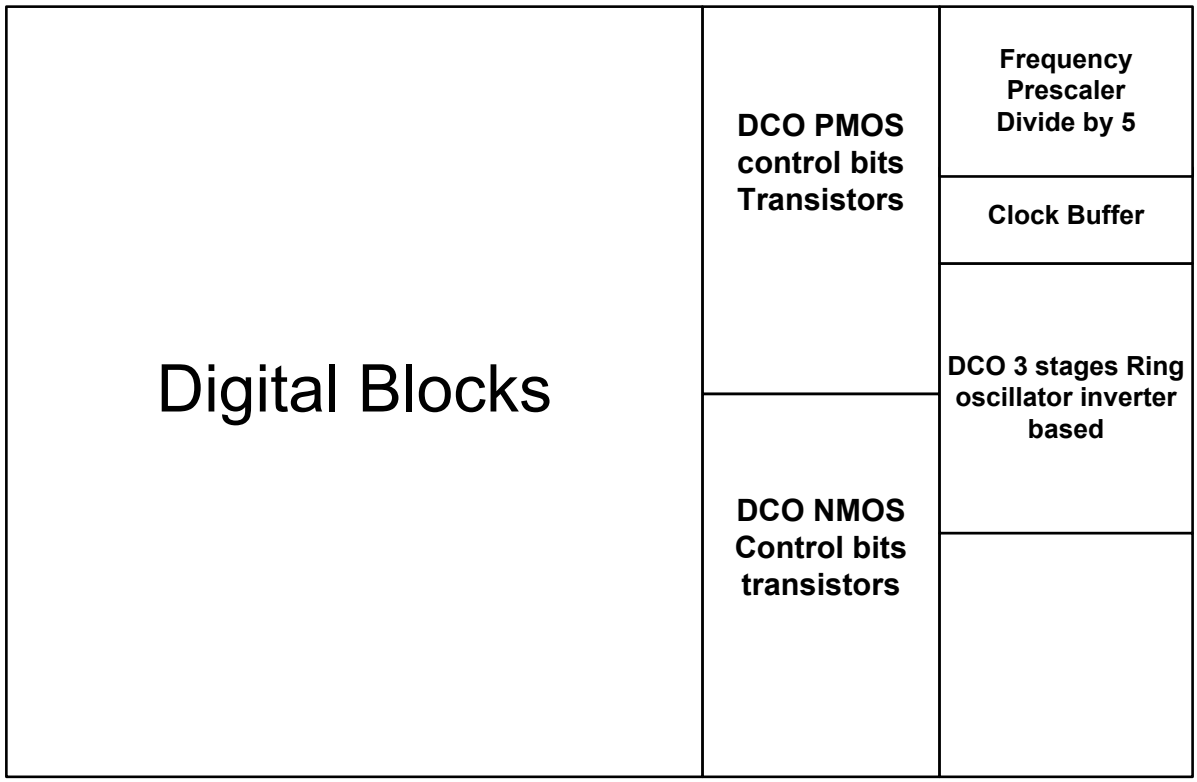


Figure 3.28 ADPLL Floorplan

3.5.1.2. DCO, Buffer and Prescaler Layout

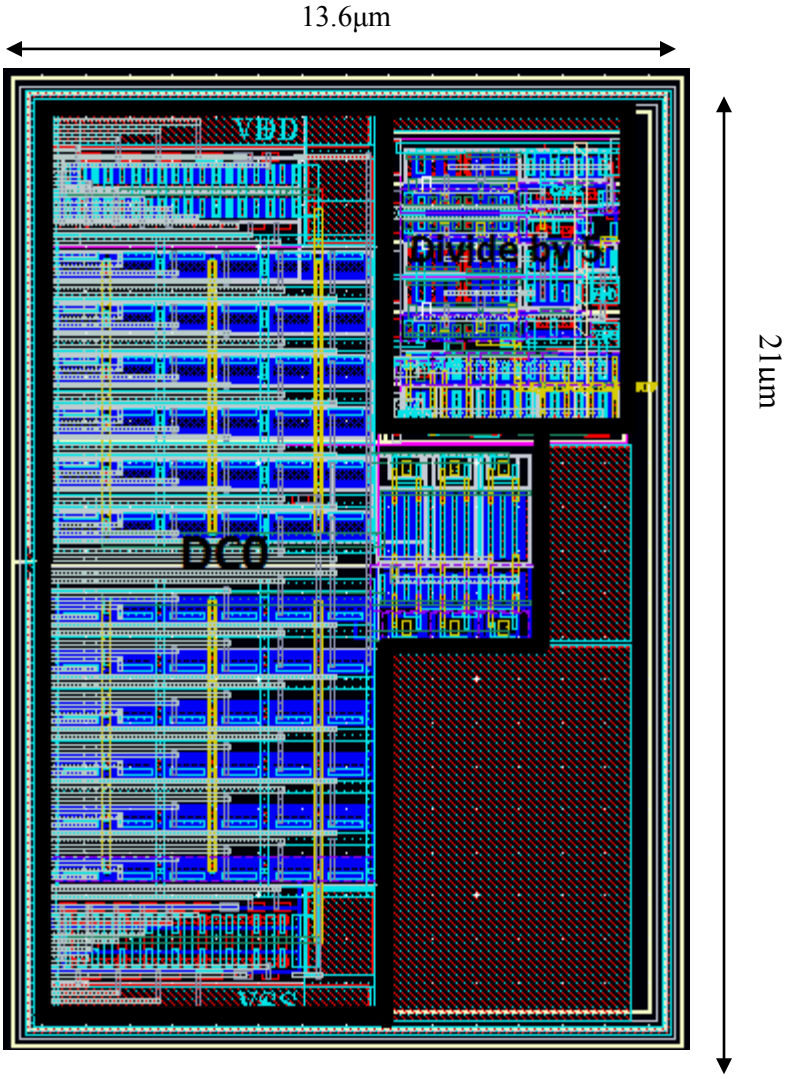


Figure 3.29 DCO and divide-by-five full custom layout

Figure 3.29 shows the full custom layout of the DCO, the clock buffer, and the frequency prescaler. The area of the full custom layout is only 13.6µm × 21µm.

3.5.2. Digital Blocks

3.5.2.1. Floor Plan

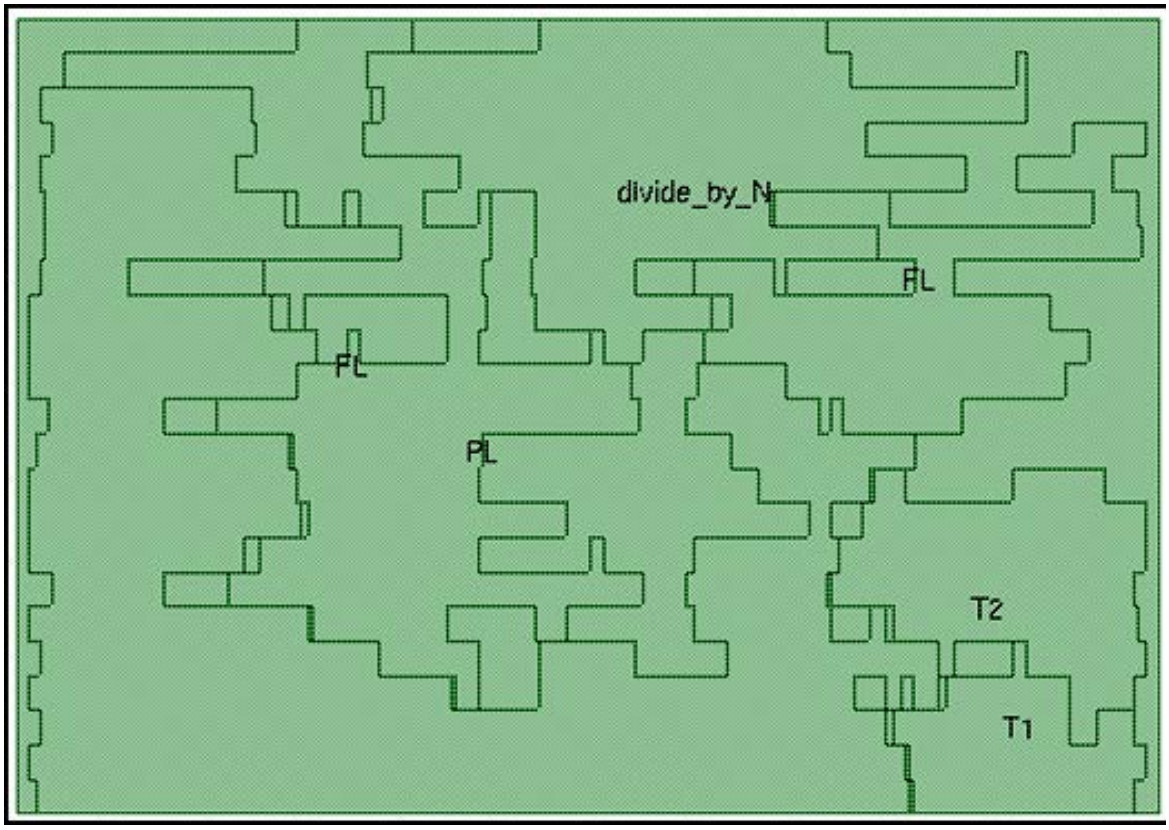


Figure 3.30 ADPLL digital blocks floor plan snapshot from SoC encounter

The Verilog codes of the digital blocks were synthesized using Synopsys. After that, post-synthesis simulations and verification were done using Modelsim. SoC encounter was used to perform a place and route for the digital blocks. A snapshot of the digital blocks floorplan is shown in Figure 3.30. The snapshot shows the placement of different ADPLL modules; PL stands for the phase loop and it consists of a BPD, a DLF, and a ΣΔ. FL stands for frequency loop. T1 is the coarse bits thermometer encoder and T2 is the fine bits thermometer encoder. Divide_by_N is the programmable frequency divider.

3.5.2.2. Layout

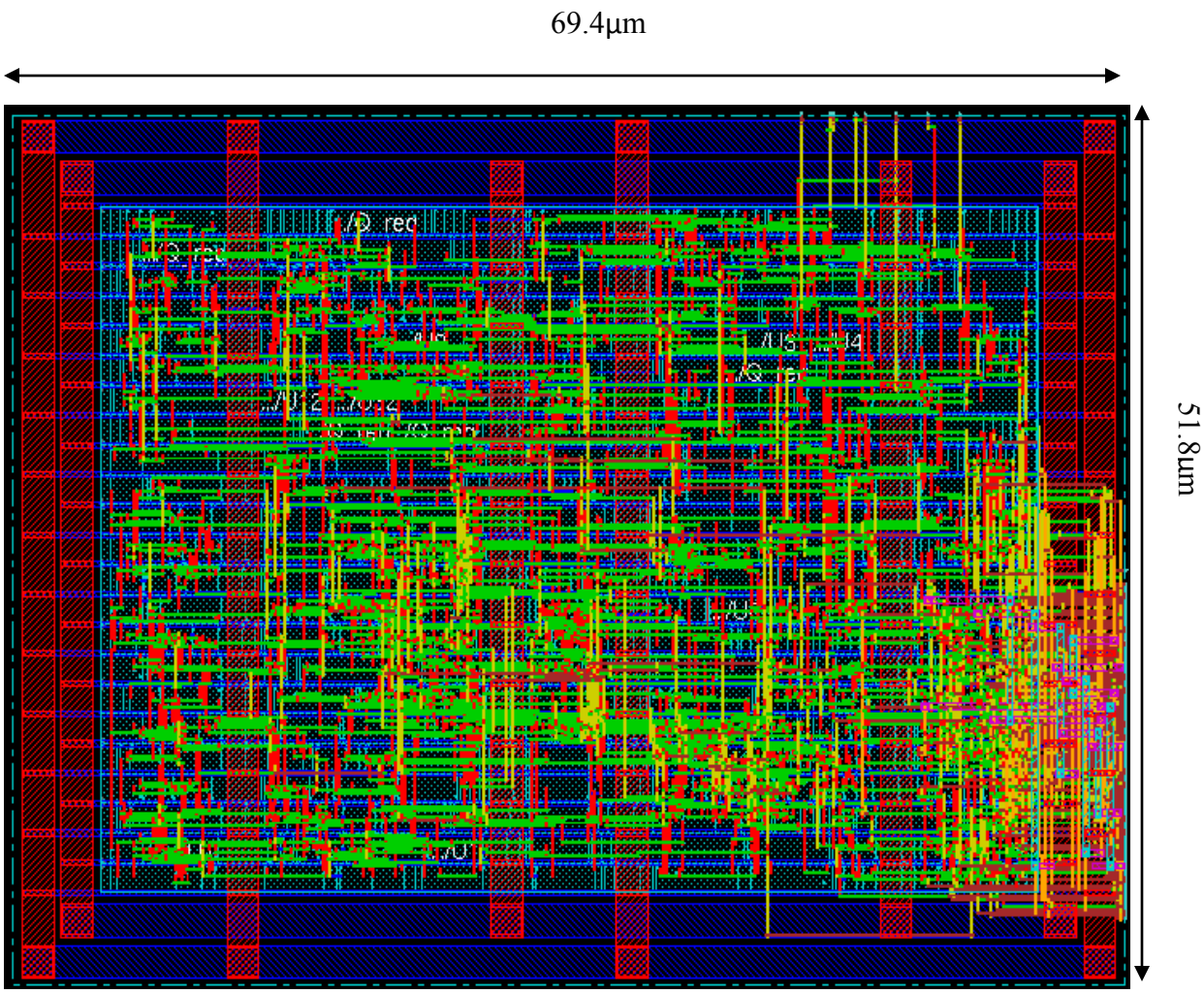


Figure 3.31 ADPLL digital blocks Layout

Figure 3.31 shows the layout of the synthesized digital blocks. The DCO control pins location is chosen to be on the right for the ease of integration with the analog DCO layout. The total power consumed in the ADPLL digital blocks was found to be 1.3mW, and the area of the digital blocks equals to 69.4µm × 51.8µm.

3.6. Design Comparison

Table 3.6 compares between the results of this work and other published work in [52], [53], [54], and [55]. Although, the TDC based architecture has proven a better jitter performance compared to the proposed bang-bang based architecture, this work has proven a significant reduction in power and area. The reduction in power and area results from the proposed simple DCO, and eliminating the area and power hungry TDC block.

Table 3.6 Design comparison of the proposed BBADPLL with other work in published literature

	[52]	[53]	[54]	[55]	**This Work**	
Type	Bang-Bang	Linear Phase Quantization	Linear TDC	Linear TDC	Bang-Bang	
Reported Frequency (GHz)	4	3	3.6	3	7.5	10
Technology (nm)	65	65	130	65	65	
Freq Range (GHz)	0.5 – 8	0.19 – 4.27	3.2 – 4.2	3	5 – 10	
Rms jitter (ps)	0.7	1.4	0.2	-	0.58	0.19
Phase Noise (dBc/Hz)	-112 (at 10 MHz)	–	-108 (at 400 KHz)	-120 (at 1 MHz)	–	–
Power (mW)	32	10.23	46.7	9.5	2.3	2.7
Area (μm^2)	200 × 150	190 × 200	1400 × 1400	1600 × 1100	84 × 52	
Lock Time (μm)	17.4	–	20	–	1.6	2.4
Reference Frequency (MHz)	500	25	50	25	100	

Chapter 4

Summary and Future Work

4.1. Summary

This thesis presented and investigated three different SerDes architectures for a three different signaling schemes. Scheme 1, used the conventional two-level signaling schemes, and a SerDes architecture with a half-rate all digital clock and data recovery (ADCDR) at the receiver. The digital implementation of the proposed half-rate ADCDR offered many advantages, as compared to the conventional ones. First, it reduced the sensitivity of the supply to noise. Second, the digital tuning voltage control of the DCO was less sensitive to noise, which made the design more robust to process, voltage and temperature variations. Third, since most of the building blocks of the ADCDR were made of synthesized digital circuits, they can be easily ported from one technology to another. Fourth, the Digital Loop Filter (DLF) occupied a small area as compared to the analog loop filter, because it didn't contain any passive elements. Finally, the absence of analog blocks and passive elements reduced the area and power significantly. Moreover, implementing a half-rate CDR architecture was shown to be a very attractive alternative for high data rate transmission, because of the lower frequency of operation that reduced the power and reduced constrains on the CDR's digital blocks. In addition, as the clock frequency was reduced, the design of DCO was relaxed. Thus, a good jitter performance and stability was easily achieved.

Schemes 2 and 3 are the two new schemes that proposed a new three-level code that multiplexed the clock with the data. The new three-level code enabled recovering the clock from

the data at the receiving side with simple circuitry, and eliminated the need of sending the clock using an extra wire, or using a complex conventional CDR at the receiver side. In addition, the proposed three-level signaling schemes were insensitive to jitter accumulated during signal transmission; hence, there were no strict constraints on the jitter performance of the PLL at the transmitter. However, the signaling schemes in scheme 2 and scheme 3 were not the same. Scheme 2 is three-level signaling offered a *DC* level preserving advantages, that were attractive for applying the source matching technique, and benefited from its properties. The source matching technique that was applied on scheme 2 doubled the signal amplitude at the transmitter. Hence, the signal attenuation along the transmission line was compensated by doubling the signal amplitude. This resulted in a smaller driver size at the transmitter, and reduced power, and area overhead.

There were many problems found in signaling scheme 2, and signaling scheme 3 signaling was proposed to overcome these problems. First, scheme two signaling scheme led to losing half of the achievable data rate, because each bit was represented in two clock cycles. Second, scheme 2 put limitations on the minimum frequency of the transmission line, which made it difficult to operate properly in slow corners. Third, sending a differential three-level signal had no advantages in terms of noise immunity, because at the receiver, there were two separate low switching inverters used to detect the signals. Therefore, there was a great probability of mismatch at the receiver end, which made extracting the clock, and recovering the data difficult. Scheme 3 proposed a resistive termination technique for the transmission line; which removed the limitation on the minimum frequency found in scheme 2. Furthermore, the data transmitted on a single link saved routing resources and avoid the mismatch problems at the receiver.

However, since source matching was not used in scheme 3, the driver size, power, and area were larger as compared to scheme 2.

A calibration technique was proposed to calibrate the switching threshold of the inverter and overcome the problems of the process variation. However, temperature and voltage variations can still affect the inverter switching threshold. Designing the switching threshold to be close to *VDD/2* was done to guarantee a temporary solution to temperature, and voltage variations, but a permanent solution of continuous calibration should be investigated in future work.

The SerDes transceiver was implemented for a 3mm long lossy on-chip differential transmission line in a 65nm TSMC CMOS technology. The total power consumed in the TX/RX pair with the transmission line in scheme 2 and scheme 3 were only 15.5mWatt and 18.1mWatt, respectively. These power dissipation values are much lower compared to similar published signaling architectures.

This thesis also presented the design and the implementation of a digital bang-bang all digital frequency synthesizer for serial links. A new design methodology for the bang-bang all digital phase locked loop (BBADPLL) was presented. This BBADPLL based on the programmable coefficients of the digital loop filter (DLF) that managed the tradeoffs between stability and the jitter of the closed loop. The proposed design showed a reduction in the area and power, compared to other time-to-digital converter (TDC) based ADPLL architectures. This resulted from eliminating the need for complex, power, and area consuming TDC blocks. A proposed simple three stages ring oscillator based digital controlled oscillator (DCO), reduced the total power and area significantly, and provided a wide frequency range.

The proposed BBADPLL was implemented on a TSMC CMOS 65nm technology with a frequency range of 5−10GHz and a frequency resolution of 500MHz. The total power consumed at 10GHz was only 2.7mWatt. The peak-to-peak period jitter and the RMS jitter at 10GHz was 1.49ps and 0.19ps, respectively. The total area of the proposed ADPLL was only 4372μm^2, which is much lower than other published architectures.

4.2. Conclusion

The goal of this thesis was to investigate a new self-timed signaling scheme for a reliable, low-power on-chip serial link. The encoder at the transmitter multiplexed the clock with the data in three-level code. The new three-level encoding technique enabled recovering the clock from the data, which eliminated any need for sending the clock through an extra wire or using power hungry complex blocks, such as PLLs and CDRs. This leads to power and area savings, in addition to reduced routing complexity. Moreover, this technique was insensitive to jitter accumulated during signal propagation or at the receiver input, because the clock signal was extracted from the multiplexed data stream. The proposed SerDes architecture was implemented using simple full custom digital CMOS circuits. The proposed architecture led to significant area and power reduction compared to the conventional SerDes transceiver.

4.3. Future Work

PVT variations affect the maximum data rate of serial links significantly. Further investigations should be done to overcome the impact of temperature and voltage variations on the switching threshold of the detector at the receiver.

Moreover, this work should be extended to the serialization of vertical TSV interconnects in 3D ICs to address the challenges of the high power densities and routing congestion due to TSV

pads. The use of three-level signaling schemes in an off-chip SerDes transceiver should be investigated as well.

The programmable digital loop filter coefficients can be programmed after lock to improve jitter performance. The digital loop filter coefficients can also be programmed to overcome any temperature and voltage variations that may lead to instability after lock.

References

[1] International Technology Roadmap for Semiconductors (ITRS), 2005.

[2] Michael P. Flynn and Joshua Jaeyoung Kang, "Global signaling over lossy transmission lines," *International Conference in Computer Aided Design ICCAD*, pp 985-992, November 2005.

[3] A. ITsuchiya, Y. Gotoh, M. Hashimoto, and H. Onodera, "Performance limitation of on-chip global interconnects for high-speed signaling," *Custom Integrated Circuits Conference (CICC),* September 2004.

[4] E. E. Hussein and Y. I. Ismail, "A novel variation insensitive clock distribution methodology," *International Symposium on Circuits and Systems (ISCAS)*, pp.1743-1746, May 2010.

[5] Anant Agarwal, and Markus Levy, "Going multicore presents challenges and opportunities," *EE Times, design article*, October 2007.

[6] International Technology Roadmap for Semiconductors (ITRS), "The next step in assembly and packaging: system level integration in the package (SiP)," 2009.

[7] S. Pasricha, "Exploring serial vertical interconnects for 3D ICs," *Design Automation Conference (DAC)*, pp.581-586, July 2009.

[8] A. W. Topol et al., "Three-dimensional integrated circuits," *IBM J. Res. & Dev.*, Vol. 50 No. 4/5, Jul/Sep 2006.

[9] K. Bernstein, et al., "Interconnects in the third dimension: design challenges for 3D ICs," *Design Automation Conference (DAC)*, pp.562-567, July 2007.

[10] R.S.Patti, "Three-dimensional integrated circuits and the future of System-on-Chip Designs," *Proceedings of the IEEE*, vol 94, no. 6, Jun 2006.

[11] A. ITsuchiya, Y. Gotoh, M. Hashimoto, and H. Onodera, "Performance limitation of on-chip global interconnects for high-speed signaling," *Custom Integrated Circuits Conference (CICC)*, September 2004.

[12] M. Sorna *et al.*, "A 6.4Gb/s CMOS SerDes core with feed-forward and decision-feedback equalization," *IEEE International Solid-State Circuits Conference (ISSCC)*, pp. 2-63, February 2005.

[13] T.Geurts, W. Rens, J. Crols, S. Kashiwakura, and Y. Segawa, "A 2.5 Gbps - 3.125 Gbps multi-core serial-link transceiver in 0.13 μm CMOS," *Proceeding of the 30th European Solid-State Circuits Conference (ESSCIRC)* , no., pp. 487- 490, 21-23, September 2004.

[14] J.Young, J. Kang, S. Park, and M. Flynn "A 9Gbit/s serial Transceiver for on-chip global signaling over lossy transmission lines," *Custom Intergrated Circuits Conference (CICC)*, pp. 347-350, September 2008.

[15] Se-Joong Lee, Kwanho Kim, Hyejung Kim, Namjun Cho, Hoi-Jun Yoo , "Adaptive network-on-chip with wave-front train serialization scheme," *Digest of Technical Papers Symposium on VLSI Circuits, 2005*, pp. 104- 107, June 2005.

[16] R. A. Philpott, J. S. Humble, R. A. Kertis, K. E. Fritz, B. K. Gilbert and E. S. Daniel, "A 20Gb/s SerDes transmitter with adjustable source Impedance and 4-tap feed-forward equalization in 65nm bulk CMOS," *IEEE Custom Intergrated Circuits Conference (CICC)*, pp. 623-626, September 2008.

[17] Y. Suzuki, K. Odagawa, and T. Abe, "Clocked CMOS calculator circuitry, " *IEEE Journal of Solid State Circuits*, vol. SC-8, pp. 462-469, December 1973.

[18] Hae-Chang Lee, "An Estimation Approach to Clock and Data Recovery," Stanford University, dissertation, Nov 2006.

[19] H. Song, D. -S. Kim, D. -H. Oh, S. Kim, D. -K. Jeong, "A 1.0–4.0-Gb/s All-Digital CDR With 1.0-ps Period Resolution DCO and Adaptive Proportional Gain Control," *IEEE Journal of Solid State Circuits*, vol.46, no.2, pp.424-434, February 2011

[20] P. K. Hanumolu, G. -Y. Wei, U. -K. Moon, and K. Mayaram, "Digitally enhanced phase-locking circuits," *IEEE Custom Intergrated Circuits Conference (CICC)*, pp. 361–368, September 2007.

[21] J. Savoj, B. Razavi, "Design of half-rate clock and data recovery circuits for optical communication systems," *Design Automation Conference (DAC),* pp. 121- 126, July 2001.

[22] Do-Hwan Oh, Deok-Soo Kim, Suhwan Kim, Deog-Kyoon Jeong, Wonchan Kim, "A 2.8Gb/s All-Digital CDR with a 10b Monotonic DCO," *IEEE International of Solid-State Circuits Conference (ISSCC),* pp.222-598, 11-15 February 2007

[23] J. D. H. Alexander, "Clock Recovery from Random Binary Signals," *Electronic Letters*, vol. 11, pp. 541-542, October 1975.

[24] David J. Rennie, "Analysis and design of robust multi-Gb/s clock and data recovery circuits," University of Waterloo, thesis, 2007.

[25] Xiaodong Liu, Yanyan Shi, Meng Wang, and Jiaojiao Deng, "Direct digital frequency synthesizer based on curve approximation," *IEEE International Conference on Industrial Technology (ICIT)*, pp.1-4, 21-24 April 2008.

[26] A. Bellaouar, M. S. O'brecht, A. M. Fahim, M. I. Elmasry, "Low-power direct digital frequency synthesis for wireless communications," *IEEE Journal of Solid-State Circuits*, vol. 35, pp.385 -390, March 2000.

[27] Ramin Farjad-Rad, William Dally, Hiok-Tiaq Ng, Ramesh Senthinathan, M.-J. Edward Lee, Rohit Rathi and John Poulton, "A Low-Power Multiplying DLL for Low-Jitter

Multigigahertz Clock Generation in Highly Integrated Digital Chips", *IEEE Journal of Solid-State Circuits*, vol. 37, no.12, December 2002.

[28] Kuo-Hsing Cheng and Yu-Lung Lo, "A Fast-Lock Wide-Range Delay-Locked Loop Using Frequency-Range Selector for Multiphase Clock Generator," *IEEE Transactions on Circuits and Systems II* , vol.54, no.7, pp.561-565, July 2007.

[29] Behzad Razavi, "Design Of Analog CMOS Integrated Circuits," International Edition 2000.

[30] C. S. Vaucher, "Architectures for RF Frequency Synthesizers," Boston: Kluwer Academic Publishers, 2002.

[31] B. Razavi, "Design of Monolithic PLLs and Clock Recovery Circuits – A Tutorial," Monolithic Phase Locked Loops and Clock Recovery Circuits: Theory and Design, New York: IEEE Press, 1996.

[32] F. M. Gardner, "Charge-Pump Phase-Lock Loops," *IEEE Trans. on Communications*, vol. 28, pp. 1849-1858, November 1980.

[33] Xiaoliang Xu, Huihua Liu, Weifeng Tan, "Parameters design of 1.25GHz low jitter charge pump PLL," *International Conference on Electric Information and Control Engineering (ICEICE)* , pp.3418-3421, April 2011.

[34] Song Ying, Wang Yuan, Jia Song, Zhao Baoying, Ji Lijiu, "Design of low jitter adaptive-bandwidth charge pump PLL with passive filter," *7th International Conference on ASIC* , pp.319-322, October 2007.

[35] A.L.S Loke, R.K Barnes, T.T Wee, M.M. Oshima, C.E. Moore, R.R Kennedy, M.J Gilsdorf, "A Versatile 90-nm CMOS Charge-Pump PLL for SerDes Transmitter Clocking," *IEEE Journal of Solid-State Circuits*, vol.41, no.8, pp.1894-1907, Aug. 2006.

[36] V. Kratyuk, P.Hanumolu, Un-Ku Moon, and K. Mayaram, "A Design Procedure for All-Digital Phase-Locked Loops Based on a Charge-Pump Phase-Locked-Loop Analogy," *IEEE Trans. on circuits and systems*, vol. 54, pp. 247–251, March 2007.

[37] K. Choi, J. Shin, J. Sim, and H. Park, "An Interpolating Digitally Controlled Oscillator for a Wide-Range All-Digital PLL," *IEEE Trans. on circuits and systems*, vol.56, pp. 2055-2063, September 2009.

[38] R. Staszewski, J. Wallberg, and S. Rezeq, "All Digital PLL and Transmitter for Mobile Phones, " *IEEE Journal of Solid-State Circuits*, vol. 40, pp. 2469-2482, December 2005.

[39] C. Chung and C. Lee, "An All-Digital Phase-Locked Loop for High-Speed Clock Generation," *IEEE J. Solid-State Circuits*, vol. 38, pp. 347–351, February 2003.

[40] B. Razavi, "Devices and circuits for phase-locked systems," in *Phase-Locking in High Performance Systems*, B. Razavi, Ed. New York: IEEE Press, 2003.

[41] S. M. Rezaul Hasan, "A Novel 16-bit CMOS Digitally Controlled Oscillator, " *IEEE Asia Pacific Conference on circuits and systems*, pp. 526-529, December 2006.

[42] M. Saint-Laurent and M. Swaminathan, "A Digitally Adjustable Resistor for Path Delay Characterization in High-Frequency Microprocessors," Southwest Symposium on Mixed Signal Design, pp. 61-64, 2001.

[43] M. Saint-Laurent et al, "A Digitally Controlled Oscillator Constructed Using Adjustable Resistor, " IEEE Southwest Symposium on Mixed-Signal Design, pp. 80-82, 2001.

[44] T. Kuendiger, Fang Chen, L. Mac Eachern, S. Mahmoud, "A Novel Digitally Controlled Low Noise Ring Oscillator, " *IEEE International Symposium on Circuits and Systems*, pp. 1016-1019, May 2008.

[45] J. Dunning, G. Garcia, J. Lundberg, and E. Nuckolls, "An all-digital phase-locked loop with 50-cycle lock time suitable for high-performance microprocessors," *IEEE J. Solid-State Circuits*, vol. 30, pp. 412–422, April 1995.

[46] Chia-Tsun Wu, Wen-Chung Shen, Wei Wang, and An-Yeu Wu; "A Two Cycle Lock time ADPLL Design Based on a Frequency Estimation Algorithm, " *IEEE Trans. on circuits and systems*, vol.56, pp. 430-434, June 2010.

[47] T. -Y. Hsu, C. -C. Wang, and C. -Y. Lee, "Design and analysis of portable high-speed clock generator, " *IEEE Trans. Circuits Syst. II*, vlo. 48, pp. 367-375, Apr. 2001.

[48] John Willy, "All digital frequency synthesizer in deep submicron CMOS," 2006.

[49] John Rogers, "Integrated Circuit Design for High-Speed Frequency Synthesis," 2006.

[50] Jri Lee, K. S. Kundert, B. Razavi , "Modeling of jitter in bang-bang clock and data recovery circuits," *Proceedings of the IEEE Custom Integrated Circuits Conference 2003* , pp. 711- 714, Sept. 2003.

[51] N. Da Dalt, "Markov Chains-Based Derivation of the Phase Detector Gain in Bang-Bang PLLs," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol.53, no.11, pp.1195-1199, Nov. 2006.

[52] JA Tierno, AV Rylyakov, DJ Friedman, "A wide power supply range, wide tuning range, all static CMOS all digital PLL in 65 nm SOI," *IEEE J. Solid-State Circuits*, vol. 43, no. 1, pp. 42-51, Jan. 2008.

[53] W. Grollitsch, R. Nonis, and N. Da Dalt, "A 1.4psrms-period-jitter TDC-less fractional-N digital PLL with digitally controlled ring oscillator in 65nm CMOS," *IEEE International Solid-State Circuits Conference (ISSCC)*, pp. 478-479, Feb. 2010.

[54] C. Hsu, MZ Straayer, MH Perrott, "A low-noise, wide-BW 3.6GHz digital ΔΣ fractional-N frequency synthesizer with a noise-shaping time-to-digital converter and quantization noise cancellation," *IEEE International Solid-State Circuits Conference (ISSCC)*, pp. 340-341, February 2008.

[55] C.Weltin-Wu, E. Temporiti, D. Baldi, F. Svelto, "A 3GHz fractional-N all-digital PLL with Precise time-to-digital converter calibration and mismatch correction," *IEEE International Solid-State Circuits Conference (ISSCC)*, pp.344-618, February 2008.