DESIGN OF DELAY-LOCKED LOOP IN 0.18-µm CMOS TECHNOLOGY

Thesis submitted towards the partial fulfillment of the requirements for the award of the degree of

Master of Technology (VLSI Design & CAD)

Submitted by

SHIRISH TRIPATHI

Regn. No. 600861015

Under the supervision of

Mr. Sanjay Kumar

Assistant Professor

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

THAPAR UNIVERSITY,

PATIALA – 147004,

INDIA

2010

CERTIFICATE

I hereby declare that the work which is being presented in the thesis entitled **"DESIGN OF DELAY-LOCKED LOOP IN 0.18-µm CMOS TECHNOLOGY"** in the partial fulfillment for the award of **Master of Technology in VLSI Design & CAD** from **Thapar University** is an authentic record of my own work carried under the supervision and guidance of **Mr. Sanjay Kumar**, Assistant Professor, Department of Electronics & Communication Engineering.

Shirish

Dated: 23.06.10

(SHIRISH TRIPATHI)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge and belief.

Kumar

Mr. SANJAY KUMAR
Assistant Professor,
Thesis Supervisor,
Thapar University,
Patiala.

Counter Signed by:

A.K.Chatterjee 2.7.10

Dr. A. K. CHATTERJEE
Professor & Head,
Department of Electronics & Communication Engg.,
Thapar University, Patiala-147004

R.K.Sharma 5/7/10

Dr. R. K. SHARMA
Dean of Academic Affairs,
Thapar University,
Patiala-147004.

ACKNOWLEDGEMENTS

It has been a great privilege to be a post-graduate student in the ECE department at the Thapar University, Patiala, and certainly a grand finale for all the years that I have been in school. My experience here is full of opportunities to learn not only from faculties and students that are experts in the field, but also from the interactions with sponsoring companies and industrial fellows with expertise in the marketplace. In addition, family and friends outside of the department have also been an important part of my life. Therefore, I would like to directly thank those people who are particularly instrumental in contributing to my experience at Thapar University. It has certainly been a unique privilege to be a member of the VLSI group, and I hope to carry on the tradition of excellence in the future work place.

First and foremost, I would like to thank my advisor and mentor, *Assistant Professor Sanjay Kumar*, for his invaluable guidance throughout my post-graduate career. His wealth of knowledge in the integrated circuit and system designs specially in low power techniques have assisted me in identifying the critical and interesting issues of research. His attitude toward work and professionalism has inspired me to continuously challenge myself to reach new levels. I look up to him not only as an excellent research advisor, but also a great human being. I often reflect upon the past 1 year of study under Sanjay sir's guidance and realize again and again how blessed I have been that good fortune always manages to find me even when I am looking elsewhere. No one could have asked for a better advisor. I am loving and enjoying every minute of my work because sir showed me how wonderful research can be. He motivated me by working harder and yet exuberating more energy and optimism than anybody I've ever seen. The example and the standard he sets will be with me always.

Next, I wish to express my gratitude to *Professor (Dr.) A. K. Chatterjee*, Head, Department of Electronics and Communication Engineering and *Ms. Alpana Agarwal*, P.G.Coordinator, for their excellent guidance and encouragement right from the beginning of this course.

Next, I'd like to thank Mr. B. K. Hemant (*Anna Sir*) for all the good times at the lab. I specially thank to my friends Pankaj (*Bihari*), Saurabh (*Beri*), Himanshu and Vishal for their help, criticisms, suggestions, and friendship which makes everyday a pleasant one. Thanks so much to all of you for the fun and great memories in my life.

Finally, and above everyone else, I would like to thank *My Family* for standing by me through all the joys and sorrows that life had to offer. I really appreciate my parents, who have patiently listened to my complains during the tough times and cheered with me during the good times. *You both* are the *Best and Most Loving Parents* that anyone can hope to have in this entire universe. I love you so much. I would also like to express my heartily gratitude to my caring sister Divya (*Ro-Li*) for boosting my moral.

I would like to thank all of them, whose names remain unmentioned but who have never retraced back from helping me whenever the need arose.

My acknowledgements would not be complete without expressing my gratitude towards God. I feel very fortunate to come to know Him during my post-graduate years at Thapar University and have continually been blessed by His endless love ever since. He is the true shepherd of my life.

SHIRISH TRIPATHI

ABSTRACT

The main objective of this thesis is to align the controlling clock signals for high-speed synchronous interface circuits by decreasing the static phase error. A dynamic de-skew circuit can be used to ensure good clock alignment across variations in process, voltage, and temperature variations (PVT). The Delay-locked Loop (DLL) is such a circuit, using a first-order closed-loop architecture that dynamically aligns its output clock signal with a reference clock signal.

Two basic types of DLL architectures are currently used: analog and digital. The analog DLL uses a continuously variable delay line to remove the skew between the output clock and the reference clock. A digital delay line uses digital elements, making the design more simple and portable, with quantized steps in the delay time.

The Delay-locked Loop is designed and simulated in Cadence Schematic Composer and Spectre respectively using UMC 0.18 μm technology. This thesis presents a DLL design with several new techniques to achieve reduced static phase error, wide lock range, and short locking time at the circuit and architectural level. All the circuit simulations has been done using various schematics of the structures and post-layout simulations are also being done after they all have been laid-out by considering all the basic design rules and by running the LVS program. The lock range is 25-125 MHz, with static phase error of 440 ps.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

F_{REF}	Reference Frequency
F_O	Output Frequency
ΔPH	Phase Difference
CK_{ref}	Reference Clock
T_{ref}	Period of input reference clock
ω_N	Close-loop bandwidth
V_{CTRL}	Control voltage
C	Capacitance
I_{CP}	Charge pump current
I_O	Output current
I_{REF}	Reference current
K_{PD}	Phase detector gain
K_{VCDL}	VCDL gain
$D_O(s)$	Output delay
$D_I(s)$	Input delay
W/L	Aspect Ratio
V_{DD}	Supply Voltage
CLK_IN	Input Clock
CLK_OUT	Output Clock

NOMENCLATURE

BIST	Built In Self Test
CP	Charge Pump
CMOS	Complementary Metal Oxide Semiconductor
DLL	Delay Locked Loop
DHDL	Digitally Controlled Half-Replica Delay Line
FSM	Finite State Machine
NMOS	n-channel Metal Oxide Semiconductor
PMOS	p-channel Metal Oxide Semiconductor
PLL	Phase Locked Loop
PD	Phase Detector
VLSI	Very Large Scale Integration
VCDL	Voltage Controlled Delay Line
VCO	Voltage Controlled Oscillator

CHAPTER 1 INTRODUCTION

Delay-locked loop (DLL) is a critical circuit component widely used in many timing applications. In this thesis, we wish to present a DLL design which can be used for a variety of applications. Specifically, we wish to build a DLL which is used in Very Large Scale Integrated (VLSI) circuits to decrease clock skew in the clock networks with short locking time and wide lock range. In this chapter, we will introduce the motivation for our proposed study, formally state the problem we are facing, and present the organization of this study.

The reduction of clock skew is one of the important problems in the VLSI design. Passive techniques such as clock network optimization techniques cannot completely reduce the clock skew. Phase-locked loops and delay-locked loops (DLL) are extensively used in VLSI circuits in order to decrease clock skew in the clock networks. DLL is a first order loop that compares it's input with a reference signal, than delay its output so that it can synchronize with the reference signal in a feedback fashion [1].

1.1 MOTIVATION

To reduce the clock skew, both phase-locked loops (PLLs) and DLLs are used in many timing circuits. When either a DLL or a PLL can be used, a DLL is always preferred in many cases because of its better stability and faster locking time compared to a PLL. Therefore, in our research, we have identified that the best circuit to generate stable time delays on chip is a DLL. A DLL is widely used as a timing circuit in many systems for the purpose of clock generation, signal synchronization, and others. For example, a DLL

is able to provide multiple clock signals which are separated from each other by a well-controlled phase shift (delay). When appropriate logic, such as edge combining is used, a new clock signal which is of a different frequency can be generated by the DLL. Such an application of a DLL has been reported in [2] for personal communication services (PCS). In addition, a DLL can also be used for signal synchronization, for example, between a CPU and a co-processor so that they can share the same data bus. This application was reported in [3].

Due to the wide range of applications of DLLs, the motivation of our research has evolved to designing a high performance DLL which can be used for a variety of applications, including decrease of clock skew in the clock networks, delay generation for testing purposes as well as for other timing applications, such as signal synchronization. Fortunately, many of the above applications share similar requirements for the DLL, such as a short locking time, better stability, and a wide locking range. These common requirements have become our design goals.

1.2 PROBLEM STATEMENT

The primary goal of the proposed thesis is to design a high performance DLL that could serve a variety of purposes. In this thesis, we need to review and evaluate various existing DLL design techniques and architectures. A DLL typically basically consists of three basic blocks: a phase detector (PD), a charge pump (CP), and a voltage-controlled delay line (VCDL). A simplified DLL block diagram is shown in the following Figure 1.1.

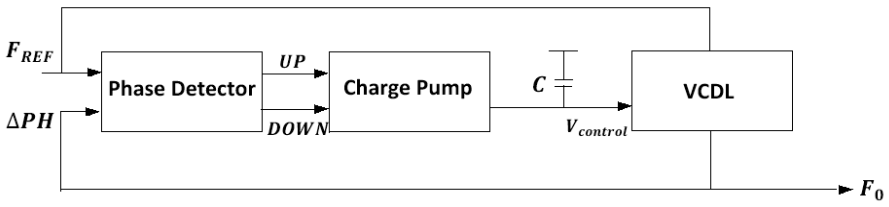


Figure 1.1 Block diagram of the Delay Locked Loop.

For a DLL, locking time, lock range and static phase error are the most important metrics. Locking time refers to the time interval a DLL takes to achieve a stable locking state from an initial state. Generally, locking time is related to the speed of the PD, the magnitude of

the charging or discharging current in the CP, and the overall delay loop bandwidth. Lock range refers to the maximum and minimum delays of the VCDL, which set the range in which the delay of the VCDL can be varied. A DLL is able to achieve lock only in this range. Lock range directly affects the operating frequency range of a DLL. Static phase error refers to the phase difference between the output signal of the last stage of the VCDL and the input reference signal. In the ideal case, after a DLL is locked, the phases of these two signals should be perfectly matched.

1.3 THESIS ORGANIZATION

The primary goal of this thesis is to demonstrate a circuit level design approach for a Delay Locked Loop which is used to reduce clock skew.

This thesis is organized into seven chapters as follows:

CHAPTER 1: INTRODUCTION. This chapter introduces the motivation of our research, outlines the problems addressed in this thesis, and presents the overall organization of this thesis.

CHAPTER 2: DLL OVERVIEW. This chapter explains the different types and architectures of DLL. Previous work in designing of DLL is also explained in a brief manner.

CHAPTER 3: PHASE DETECTOR. This chapter briefly introduces the concept of phase detector, one of the most important building blocks in the design of DLL. Conventional PD designed with flip-flop and Low-Glitch PD both is compared.

CHAPTER 4: CHARGE PUMP. This chapter introduces the concept of different current mirrors for the implementation of charge pump. Then finally a charge pump is introduced with a low voltage current mirror.

CHAPTER 5: VOLTAGE CONTROLLED DELAY LINE. This chapter explains the various types of delay elements and finally a voltage controlled delay line is designed.

CHAPTER 6: PERFORMANCE EVALUATION. This chapter discusses all about performance summary at different PVT variations. It also discusses the designs of different layouts for all the proposed structures, which are designed in Cadence Virtuoso UMC 0.18 micron Technology and the Layout versus Schematic (LVS) program was executed to perform a comparison of the schematic to the physical layout.

CHAPTER 7: CONCLUSIONS AND FUTURE WORK. This chapter summarizes the major accomplishments of this thesis and presents the scope for future and further research.

CHAPTER 2

DLL OVERVIEW

2.1 INTRODUCTION

A delay-locked loop (DLL) is a digital circuit similar to a phase-locked loop (PLL), with the main difference being the presence of a voltage controlled delay line in place of voltage controlled oscillator. The primary function of a DLL is to achieve phase alignment between the input clock and the output clock from the final stage of the VCDL. When the phase alignment is achieved, each VCDL delay stage is able to provide a stable clock signal which is phase shifted from the input clock. However, the rising clock speeds and integration levels of digital circuits have made the phase alignment task increasingly difficult [4].

The application areas of DLL include clock distribution, clock synthesis, memory, microprocessors, clock and data recovery, and communication ICs in order to reduce on chip clock buffering delays and improve I/O timing margins. They can also be used to generate multiple clock signals on chip for applications such as for Built In Self Test (BIST) circuits. Thus, a thorough study of previous work in DLL design and analysis is required to accomplish the design goals of wide lock range, less static error, and fast locking. In the subsequent sections of this chapter different architecture of DLL design, operating principle, closed loop dynamics and analysis is shown in an elaborated manner.

2.2 CLOCK SKEW

As silicon fabrication technology develops, more logic can be packed on a die and as a result the chip size gets progressively bigger. The number of logic gates and chip operating frequencies increase, and the clock skew becomes increasingly more important in ensuring the proper functioning of VLSI chips. With a synchronous communication protocol on and off the chip, it is impractical to increase the communication clock speed further without reducing the clock skew on the chip [5].

In circuit designs, clock skew is a phenomenon in synchronous circuits in which the clock signal (sent from the clock circuit) arrives at different components at different times. This can be caused by many different things, such as wire-interconnect length, temperature variations, variation in intermediate devices, capacitive coupling, material imperfections, and differences in input capacitance on the clock inputs of devices using the clock. As the clock rate of a circuit increases, timing becomes more critical and less variation can be tolerated if the circuit is to function properly. Figure 2.1 clearly shows the concept of clock skew.

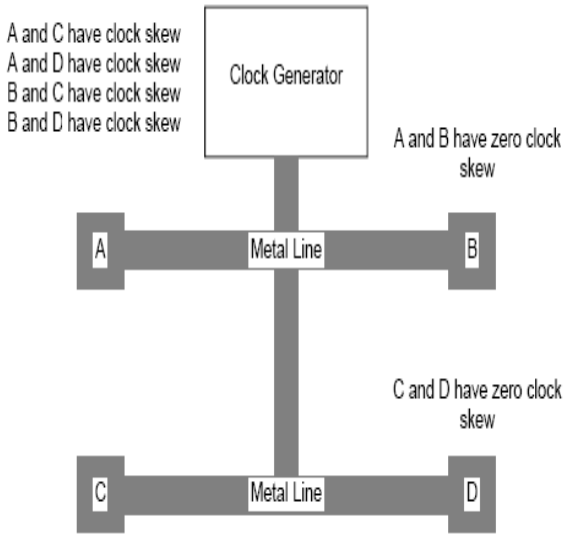


Figure 2.1 Clock Skew Concepts.

There are two types of clock skew: negative skew and positive skew. Positive skew occurs when the sending register receives the clock earlier than the receiving register while negative skew occurs when the receiving register gets the clock earlier than the sending register.

2.3 DLL OPERATING PRINCIPLE

A DLL is essentially a nonlinear negative feedback system. In a DLL, the input clock signal propagates through the VCDL and develops time delay at every delay stage of the VCDL. The phase shift of each delay stage is controlled by the voltage of a loop filter. The output is taken from one of the delay stages. The phase of the output signal is compared with the phase of the input clock in the PD as shown in Figure 2.2.

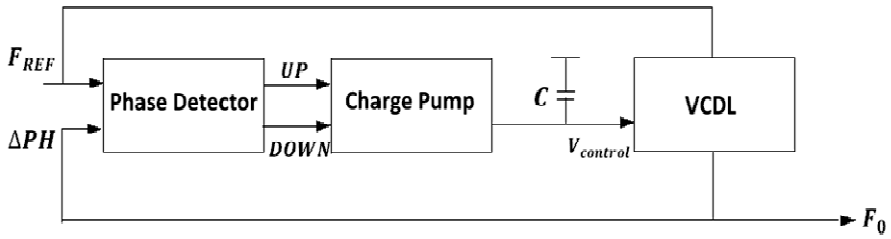


Figure 2.2 Block Diagram of DLL.

The phase error information generated by the PD (usually in the form of a voltage or a current) is then transferred to the CP. The CP uses the phase error information to adjust the voltage of the loop filter and thus to change the delay of the VCDL. Owing to such a negative feedback mechanism, the phase error is gradually reduced until it finally becomes zero. At that time, the delay of the whole VCDL line becomes equal to one clock period, and the voltage of the loop filter is stabilized, which indicates that a locked state has been established [6].

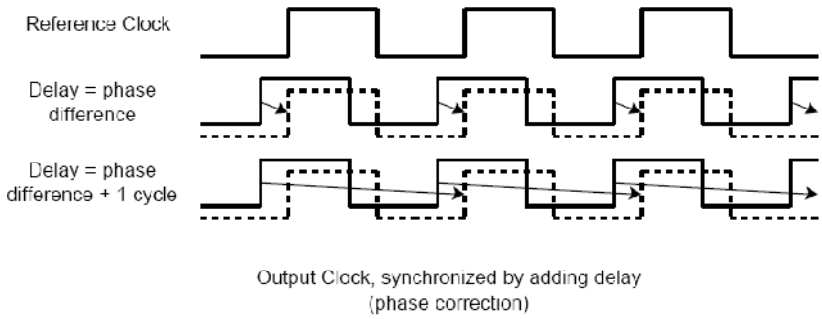


Figure 2.3 Basic DLL Timing.

The DLL tries to remove the clock skew in between two clock signals by adding additional delay. There are two options available for delay addition, either the delay to be added is any multiplication of clock period plus the phase error or it is just equal to the phase error in between the clock signals as shown in Figure 2.3.

2.4 DLL ARCHITECTURES

There are two types of DLL's with different architectures. In the first type, the input reference clock CK_{ref} is compared with a delayed version of itself in the PD. This DLL structure is often used for frequency synthesis, clock generation, and signal synchronization [7].

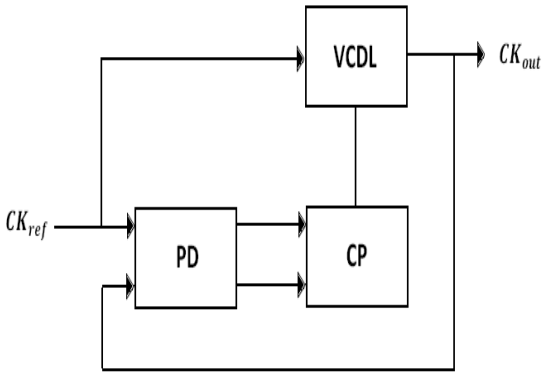


Figure 2.4 The first type of DLL.

There is another type of DLL in which the reference signal CK_{ref} is compared with a delayed version of another uncorrelated signal CK_{src} [7], as shown in Figure 2.5.

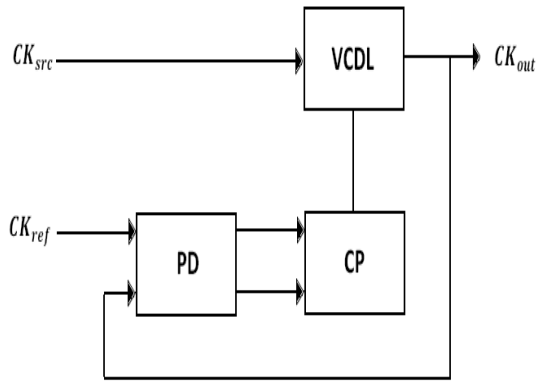


Figure 2.5 The second type of DLL.

This type of DLL requires two input signals (CK_{ref} and CK_{src}) and it is often used in some clock recovery circuits. Since it is the first type of DLL that will be used in the proposed research, our focus will be placed on this type of DLL.

2.5 DLL CLOSED LOOP RESPONSE

The frequency response of the DLL is analyzed here. The response is formulated in terms of input delay and output delay as clearly shown in Figure 2.6. The output delay is the delay between the reference input and the DLL output or, equivalently, the delay established by the VCDL. The input delay is the delay to which the phase detector compares the output delay or, equivalently, the phase difference for which the phase detector and charge pump generate no error signal.

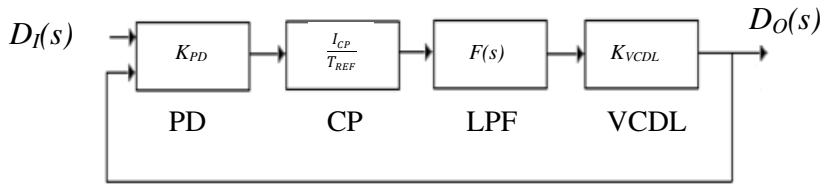


Figure 2.6 Closed Loop Dynamics of the DLL.

The parameters in the above figure include the phase detector gain K_{PD} (*second/rad*), the charge pump current I_{CP} (*Amp*), the loop filter transfer function *F*(*s*), and the VCDL gain K_{VCDL} (*rad/V*) which is proportional to the number of delay cells. The input and output delays are denoted by $D_o(s)$ and $D_I(s)$, respectively. The period of the input reference clock is T_{REF} . In most cases, the loop filter consists of only a single capacitor. Thus, the transfer function *F*(*s*) can be written as

$$F(s) = \frac{1}{sC_1} \tag{2.1}$$

The output delay, $D_o(s)$, is related to the input delay, $D_I(s)$, by

$$D_O(s) = \big(D_I(s) - D_O(s)\big) \cdot F_{REF} \cdot \frac{I_{CP}}{sC_1} \cdot K_{VCDL}K_{PD} \tag{2.2}$$

Hence, in the steady locked state, the close-loop behavior of a DLL can be characterized by a first-order transfer function

$$\frac{D_O(s)}{D_I(s)} = \frac{1}{1+s/\omega_N} \tag{2.3}$$

where ω_N, defined as the loop bandwidth (rad/s), is given by

$$\omega_N = \omega_{REF} \cdot \frac{I_{CP}}{2\pi C_1} \cdot K_{VCDL} \cdot K_{PD} \tag{2.4}$$

If the charge pump current I_{CP} , PD gain K_{PD} and the VCDL gain K_{VCDL} are constant, the loop bandwidth ω_N will track the operating frequency ω_{REF}. However, the parameters I_{CP}, K_{VCDL}, and C_1 are process technology dependent and will cause the loop bandwidth to vary around the design target.

It can be seen from (2.3) and (2.4) that a conventional DLL is a single-pole system and therefore is unconditionally stable. The loop bandwidth ω_N can track the operation frequency ω_{REF} as long as the term $K_{PD}K_{VCDL}I_{CP}/C_1$ is kept constant as given by (2.4). The effect of such tracking is two-folded. On one hand, the tracking is advantageous because as the operation frequency ω_{REF} increases, the loop bandwidth ω_N also increases which results in a faster acquisition speed. On the other hand, such tracking may not be desirable since a narrower loop bandwidth can help attenuate more high-frequency input noise, since the input jitter would be attenuated at 20dB/decade outside the loop bandwidth [6].

2.6 DLL CATEGORIES

There are essentially two types of DLL designs: analog and digital. The design choice is determined by several factors, including design complexity, layout size, system noise levels, necessity of process portability, and required accuracy. Generally speaking, a digital DLL has the advantage of being more robust, enabling better process portability, requiring lower supply voltage, and being simpler design.

2.6.1 DIGITAL DLL DESIGNS

A digital delay-locked loop uses digital devices to implement the variable delay-line as shown in Figure 2.7. This means that the minimum change in the delay is some quantized

step. A digital delay cell makes up the minimum delay, and many of these delay cells are used to create the delay line.

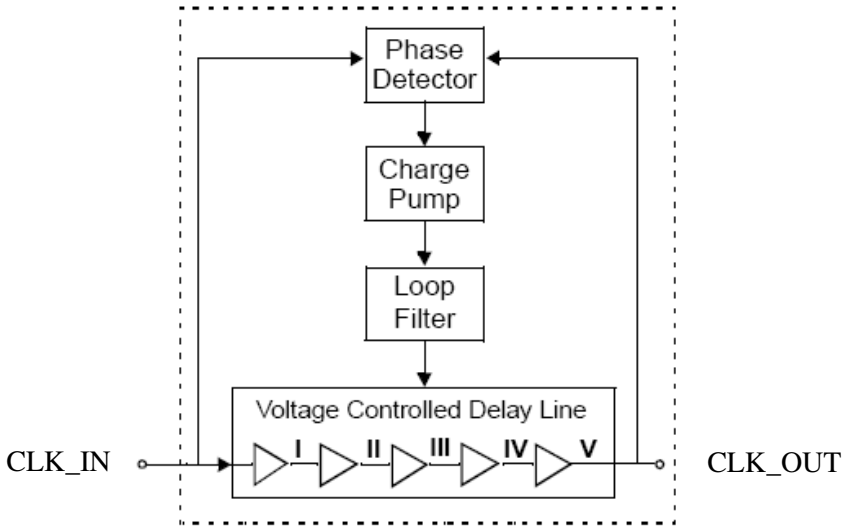


Figure 2.7 A Digital DLL.

However, due to the reliable nature of digital circuit performance across varying processes, a digital design may still be desirable [8].

There are many flavors of digital DLL's. The delay line may be implemented using pairs of inverting CMOS gates (to avoid an unwanted phase inversion), with the pairs ranging from NAND-invert to NOR-NAND combinations. This choice is influenced by the required duty cycle, need for symmetry in t_{PHL} and t_{PLH} , as well as the method used to inject or extract the clock into or out of the delay line [9].

The PD in a digital DLL is also made from digital elements such as logic gates and flip-flops. Using more digital blocks in a circuit is very much beneficial. It increases the first-run success rate and also improves the design portability across different processes. The digital DLL designs are more adjustable to lower supply voltages than analog DLL's because in analog DLL designs, the voltage headroom problem will occur as supply voltage drops, whereas digital DLL's can still function correctly as long as the supply voltage is sufficient to maintain proper noise margins. Moreover, when the supply voltage drops, the power consumption of digital circuits will decrease proportional to V^2 , while the power consumption of analog circuits only goes down roughly proportional to V [8].

Conventional digital DLL's suffer from several drawbacks [8]. Firstly, the phase resolution of digital DLL's using basic inverter delay lines is limited to two inverter delays, which is not fine enough in many applications. Secondly, since the desired signal is not necessarily from the last tap, the delay line needs to provide a large phase coverage which inevitably requires more delay cells. Thirdly, basic inverters have poor power supply rejection ratio (PSRR), which leads to more jitter in the output signal. Therefore, digital DLL's employing basic inverters in the VCDL are not suitable for applications with stringent timing requirements.

To overcome the limitations of conventional digital DLL's, several techniques have been proposed in the literature to improve the DLL's performance. A portable digital DLL was proposed in [8] for clock alignment in the interface cells of a high-speed memory system. Two complementary delay lines are used in the digital DLL to improve the phase resolution by a factor of two. An end-of-cycle (EOC) detector is also employed to facilitate the switching between the operations of the two delay lines. In addition, phase blenders are used to further increase the phase resolution of the DLL.

Another improved digital DLL was proposed in [10]. All the components in this DLL are built from digital parts. The DLL uses a 128 tap delay line. A 128-to-1 selector is employed to select the appropriate output signal. An attractive component of this DLL is a digital lead-lag PD which can achieve zero jitter in the locked state.

2.6.2 OTHER DLL STRUCTURES

Instead of just focusing on redesigning the building blocks of a conventional DLL, many novel DLL architectures have also been proposed in the literature. Many of these DLL's use both analog and digital components to achieve fast speed, less lock time, and wide lock range.

The first DLL design introduced here is a low jitter DLL proposed in [11]. For a conventional DLL, the jitter associated with the input signal can be suppressed by decreasing the loop bandwidth. However, this method can not reduce the jitter originating from the VCDL. To solve this problem, a Butterworth DLL is proposed which can reduce the jitter due to both the input noise and the VCDL noise. The basic idea is to divide the

VCDL into several shorter and identical segments so that the VCDL noise does not propagate through the whole VCDL. For example, in a second order Butterworth DLL, the VCDL is divided into two segments, namely, VCDL1 and VCDL2. Theoretical noise analysis shows that the proposed second order and third order DLL can reduce the input noise by a factor of $\sqrt[4]{2}$ and $\sqrt{1.5}$, respectively. The proposed DLL can also reduce VCDL noise by a factor of $\sqrt{2}$ and 2, respectively.

Next, several dual-loop DLL structures will be introduced which have been proven to have a number of advantages over conventional DLL's at the expense of additional power and chip area consumption. The dual-loop DLL in [12] is based on a cascade of a core loop and a peripheral loop. The VCDL in the core loop consists of six delay stages and produces six clocks which are evenly spaced by 30º. The peripheral loop includes a phase selection and inversion circuit, a phase interpolator, a PD, and a digital FSM. The six clocks from the core VCDL are selectively interpolated in the peripheral loop to generate the desired output clock under the control of a FSM. This dual-loop DLL has unlimited phase shift which is largely due to the effective phase capture algorithm implemented by the FSM.

Another dual-loop DLL with multiple VCDL's was proposed to extend the DLL's delay range [13]. The proposed DLL consists of a main loop and a reference loop. The reference loop generates four quadrature clocks, which are delayed by four VCDLs and then multiplexed to generate the desired output clock. This output clock is sent to the main loop for phase comparison and the phase error signal is used to adjust the delays of the four VCDL's. This dual-loop structure allows unlimited delay range.

Finally, a fast-locking and low-jitter DLL using a digitally controlled half-replica delay line (DHDL) was proposed in [14]. In conventional DLL's, some non-ideal effects such as leakage current may disturb the control voltage $V_{CONTROL}$, which further leads to jitter in the VCDL output signal. To solve this problem, the proposed DLL incorporates a DHDL structure to provide a stable bias voltage for the CP so that the control voltage is less affected by these non-ideal effects. As a result, the DLL achieves low jitter operation and at the same time maintains the property of bandwidth tracking. In summary, there are many techniques and architectures currently available for the design of high-performance DLL's. These methods have been proven to be effective in their respective applications.

2.7 DLL VS. PLL

When it comes to choosing between a DLL and PLL for a particular application, differences in their architecture need to be understood. The oscillator used in PLL inherently introduces instability and accumulation of phase errors. This in turn degrades the performance of the PLL when compensating for the delay of the clock distribution network. On the other hand, the unconditionally stable DLL architecture does not accumulate phase errors. For this reason, the DLL architecture is widely used for delay compensation and clock conditioning.

The DLL's closed loop transfer function has only one pole [15]. Therefore, it is naturally a stable system. On the other hand, a PLL's closed loop transfer function has two or three poles. Therefore, stability is a major issue and needs to be addressed during designing.

The main disadvantage of a conventional DLL to a PLL is its limited phase capture range. At a given operating clock frequency a DLL can delay its input clock by an amount bounded by a minimum and maximum delay. As a consequence extra care is needed to prevent the loop from trying to lock to a delay outside these limits.

2.8 PERFORMANCE PARAMETERS OF DLL

The DLL performance basically depends on some of the parameters evaluation and these parameters are explained below in brief.

2.8.1 STATIC PHASE ERROR

Static phase error refers to the phase difference between the output signal of the last stage of the VCDL and the input reference signal. In the ideal case, after a DLL is locked, the phases of these two signals should be perfectly matched. However, due to the limited resolution of the PD and CP some static phase error may exist.

2.8.2 LOCK TIME

Locking time refers to the time interval a DLL takes to achieve a stable locking state from an initial state. Generally, locking time is related to the speed of the PD, the magnitude of the charging or discharging current in the CP, and the overall delay loop bandwidth.

2.8.3 LOCK RANGE

Lock range refers to the frequency range in which a DLL is able to achieve lock. Although it is often desired to have lock range as wide as possible, the lock range of a particular DLL is usually designed for a specific application.

CHAPTER 3 PHASE DETECTOR

3.1 INTRODUCTION

The primary objective of the Phase Detector is to measure the phase difference in between the reference and feedback signals. If there is a phase difference between the two signals, it generates error signal UP or DOWN synchronized to the charge pump/low pass filter. The PD will generate UP signal as output when CLK_IN leads CLK_OUT and DOWN signal when CLK_IN lags CLK_OUT. If the error signal from the PD is an UP signal, then the charge pump pumps charge onto the low pass filter (LPF) capacitor which increases the control voltage V_{CTRL}. On the contrary, if the error signal from the PD is a DOWN signal, the charge pump removes charge from the LPF capacitor, which decreases V_{CTRL}.

The phase detector is basically of three types:

(1) XOR gate based PD
(2) Flip-Flop based PD
(3) Dynamic PD

An example of a basic phase detector is the XOR gate. It produces error pulses on both falling and rising edges [16]. A detailed analysis of the XOR PD when the reference and feedback signals are out of phase by zero, π/2, and π respectively is shown below.

Figure 3.1 XOR Phase Detector.

In Figure 3.2, the phase difference between the two signals is zero-locked phase. The average output V_{AVG}, from the XOR gate is zero for this case.

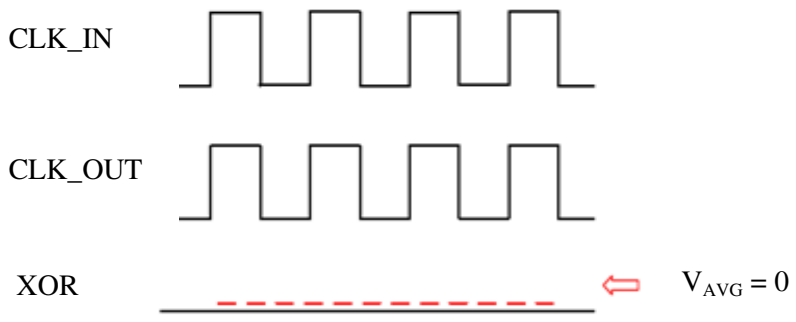


Figure 3.2 XOR Phase Detector with Phase Difference = 0.

The XOR input/output characteristic graph is a plot of V_{AVG} versus the phase difference. Figures 3.3 and 3.4 represent the phase difference for $\pi/2$, and π respectively. In an XOR

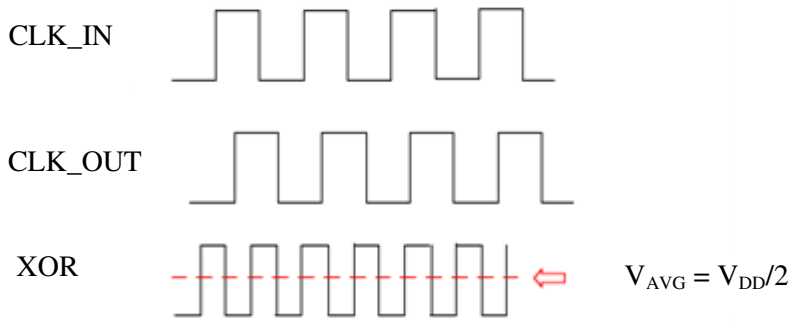


Figure 3.3 XOR Phase Detector with Phase Difference = $\pi/2$.

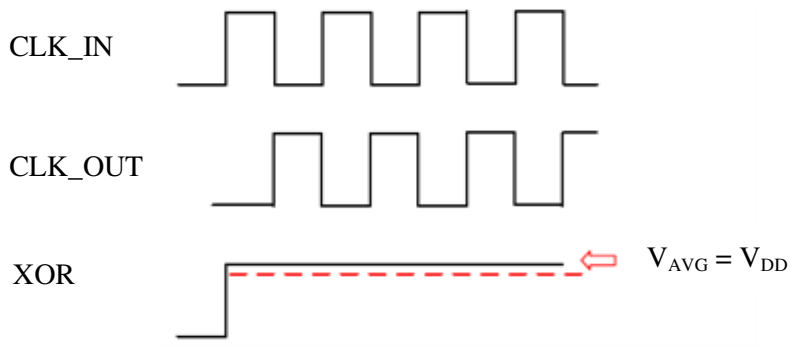


Figure 3.4 XOR Phase Detector with Phase Difference = π.

PD, the dc value of the PD output signal is linearly proportional to the difference of the two input signals. The XOR PD characteristic plot is shown in Figure 3.5.

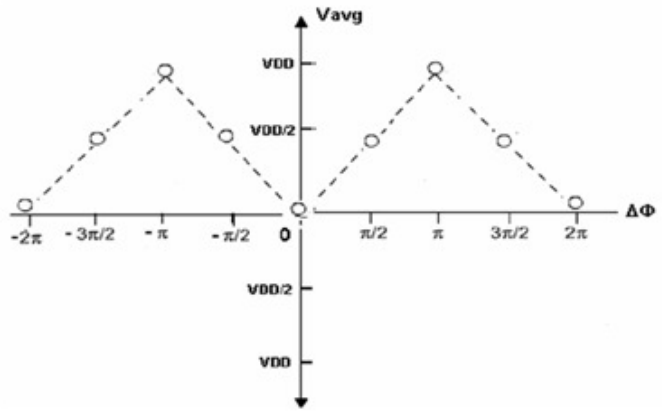


Figure 3.5 PD Characteristics Graph of Phase Difference ranging from 0 to 2π [16].

The simple XOR PD suffers from several drawbacks. Firstly, there is only one output from the XOR PD, which makes it difficult to interface with the subsequent circuit. Secondly, an XOR gate is essentially a signal level detector, i.e., the output of a gate is dependent upon the duty cycle of the two input signals. Consequently, XOR PD may generate incorrect phase error information unless duty cycle correction circuits are used.

An improved PD is based on flip-flops. Since flip-flops offer edge detection, the duty cycle dependence problem with the XOR gate PD can be avoided. Figure 3.6 shows a commonly used flip-flop based linear PD, which consists of two flip-flops and one NAND gate [17].

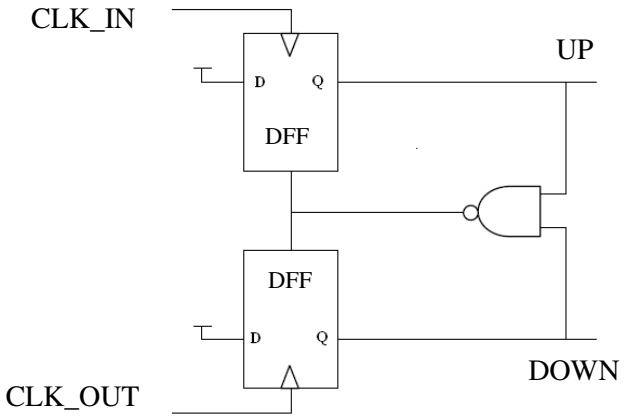


Figure 3.6 Phase Detector based on D Flip-Flop.

This Phase Detector consists of two edge-triggered, resettable D flip-flops with their D input held to "1". The inputs CLK_IN and CLK_OUT serve as the clocks for the flip-flops. The outputs of the flip-flops are denoted as UP and DOWN.

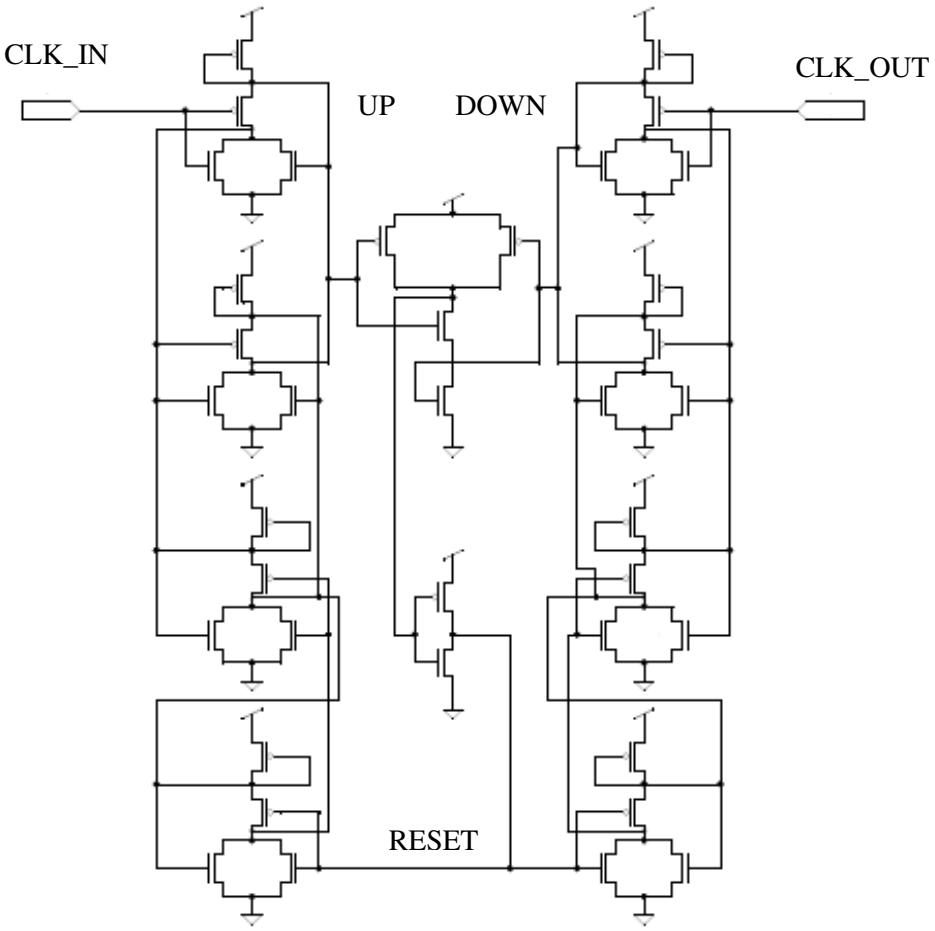


Figure 3.7 Gate Level Implementation of Flip-Flop based Phase Detector.

At any point of time, the phase detector can be in one of the four states:

- UP = 0 and DOWN = 0.
- UP = 1 and DOWN = 0.
- UP = 0 and DOWN = 1.
- UP = 1 and DOWN = 1.

However, when both UP = 1 and DOWN = 1 occurs, the AND gate output becomes "1" and both the flips are resetted. Thus, UP and DOWN are simultaneously high for a short duration of time but the difference between their average values represents the input phase difference correctly.

The flip-flop based PD is capable of detecting both the phase and the frequency difference, which helps to increase the acquisition range and the locking speed. The disadvantage about the flip-flop based PD is that the reset time may limit the speed of the PD, which may further limit the operating frequency and the acquisition speed of the DLL. It has the drawbacks of dead zone, high power consumption when operating at high frequency as the internal nodes are not completely pulled up or pulled down, limited speed as the maximum operation frequency is inversely proportional to the reset pulse width of the circuit, and large area for large number of transistors.

A third type of PD, which is widely used recently in high-speed DLL designs, is a dynamic PD shown in Figure 3.8. The basic structure of a dynamic PD includes two blocks, which are used to generate the UP signal and the DOWN signal, respectively. The two blocks have exactly the same design, except that the two input signals are switched in position. The dynamic PD eliminates flip-flops and has the advantages of simple structure and a fast transition time.

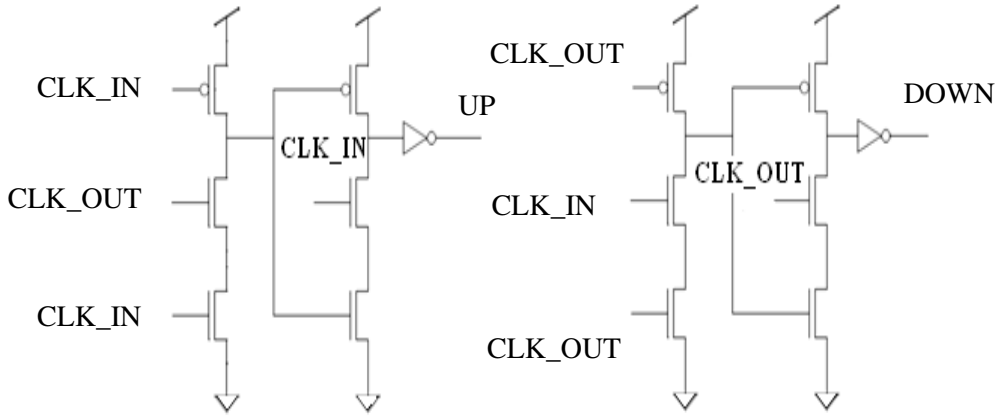


Figure 3.8 A Dynamic Phase Detector.

Each block consists of two cascaded stages with a precharge PMOS in each stage. The precharge activity of the second stage is often controlled by the output of the first stage, as shown in Figure 3.8. The dynamic PD eliminates flip-flops and has the advantages of simple structure and a fast transition time. However, dynamic PD needs to be carefully designed in order to minimize the dead zone.

3.2 LOW GLITCH DYNAMIC PHASE DETECTOR

The phase detector based on flip-flop has high glitch as compared to dynamic PD. If the glitch is more at the output of PD, the reset time of the flip-flop is also more. Effectively the duration of glitch is a factor which determines the speed of the PD. In order to reduce the glitch as well as increase the speed of PD, a dynamic PD with low glitch is introduced here. The simulation results clearly show that dynamic PD ensures a low glitch in the output as compared to the PD based on flip-flops. The schematic view of low glitch dynamic phase detector is shown below in Figure 3.9.

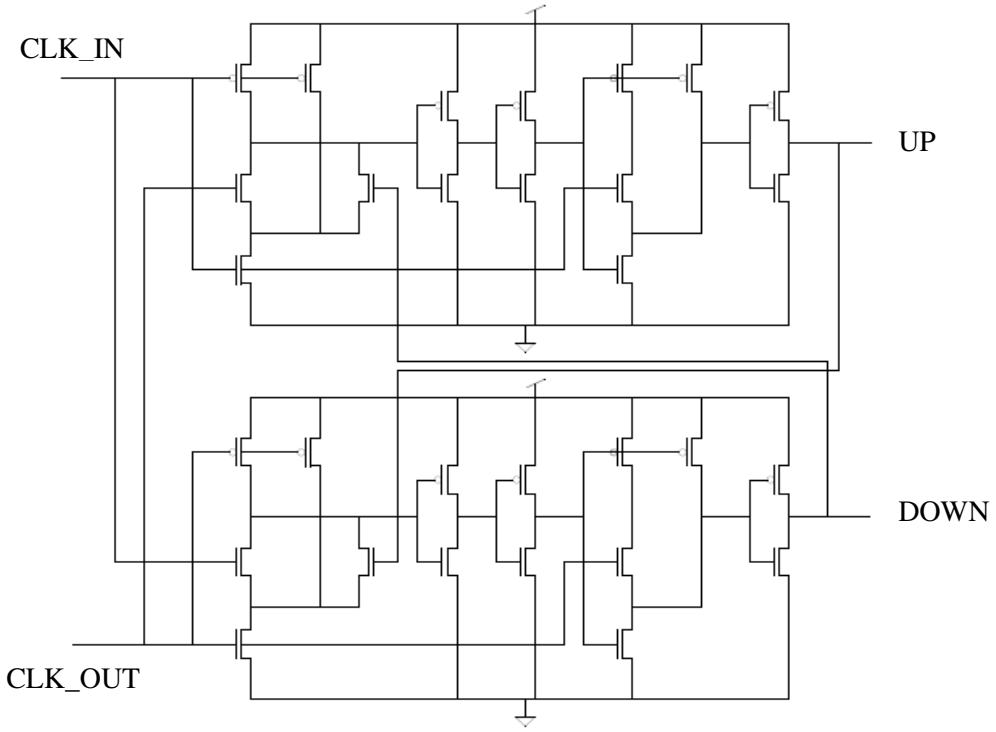


Figure 3.9 Phase Detector Schematic.

The implemented phase detector circuit can be analyzed in two different ways. One way in which CLK_IN leads CLK_OUT, and the other in which CLK_IN leads CLK_OUT.

The UP pulse is the difference between the phases of the two clock signals. This UP pulse indicates to the rest of the circuit that the feedback signal needs to speed-up or catch-up with the reference signal.

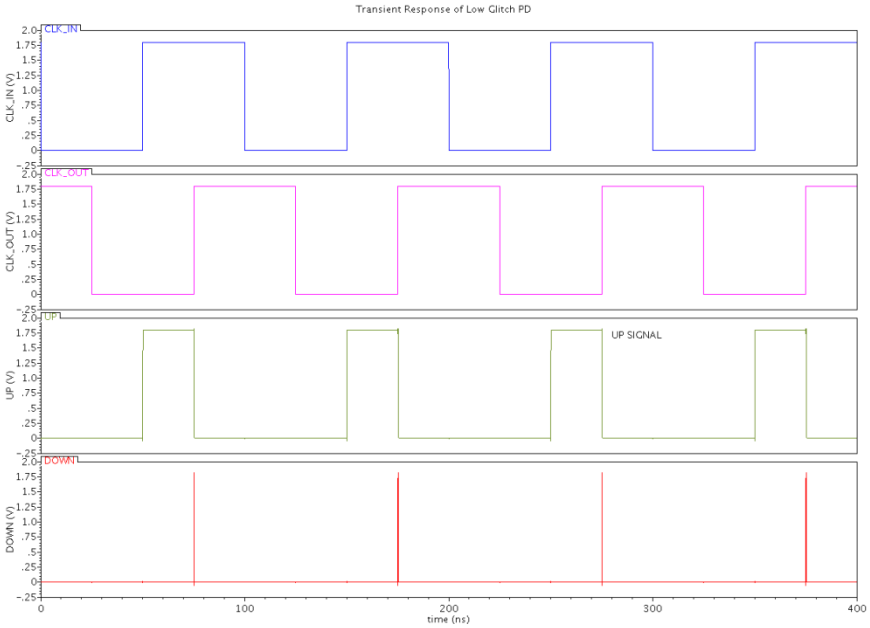


Figure 3.10 PD Simulation I (CLK_IN leads CLK_OUT).

In the second case CLK_OUT is leading CLK_IN. In this DOWN pulse represents the difference between the phases of the two clock signals.

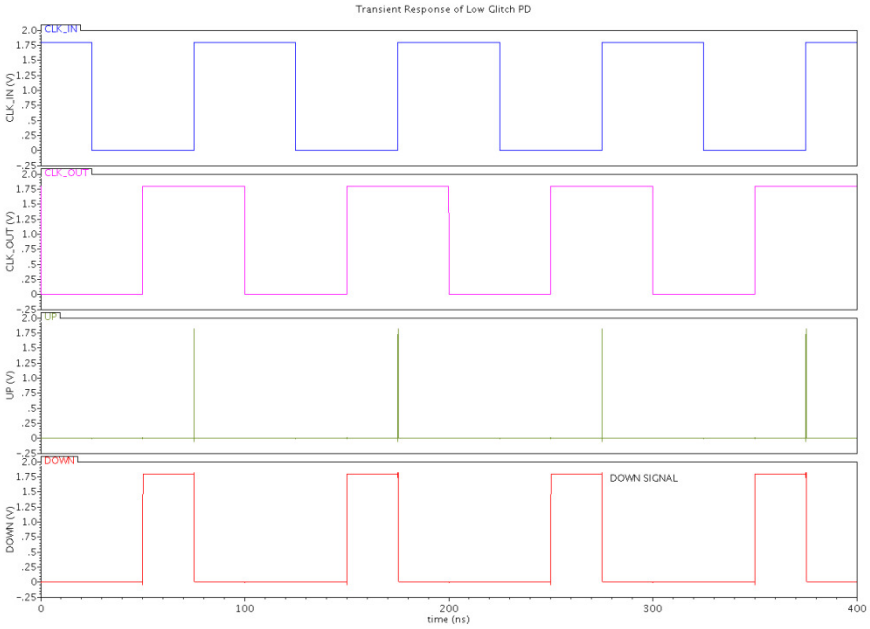


Figure 3.11 PD Simulation II (CLK_IN lags CLK_OUT).

The inputs to the PD are two signals which are having a period of 100 ns and with a phase difference of 90º (25 ns) in 180 nm technology with V_{DD} = 1.8V. On simulation, the flip-flop based PD has the duration of glitch as 590 ps [75.00 ns to 75.59 ns] which

is quite high as compared to introduced low-glitch PD. The zoomed-view of glitch in flip-flop based PD is clearly shown in Figure 3.12.

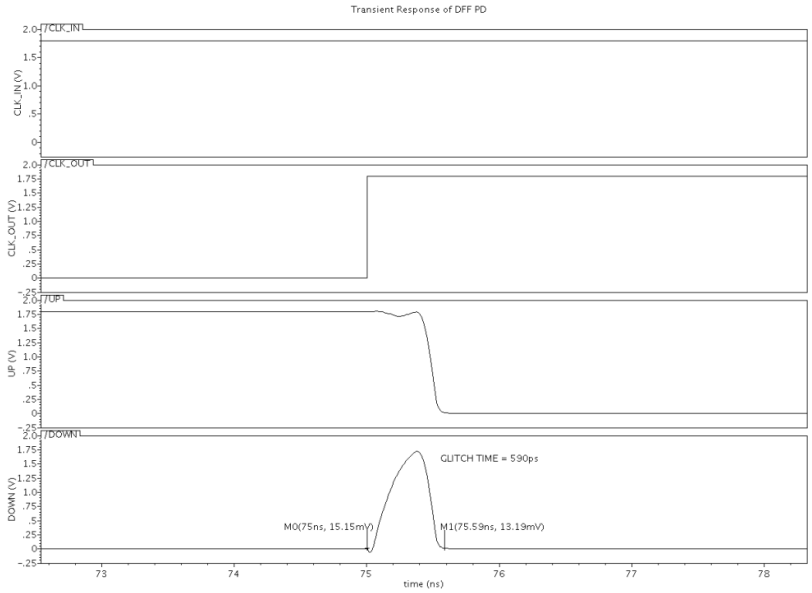


Figure 3.12 Zoomed view of PD based on FF [Glitch is from 75 ns to 75.59 ns].

The inputs to the low glitch PD are the same signals that applied to the prior circuit. The outputs of the two PD's are similar without zooming. The zoomed view establishes the advantage of the low glitch PD over the prior circuit. The glitch at the output of low glitch PD is much less as compared to PD based on flip flops. The duration of glitch in the PD is 220 ps [75.00 ns to 75.22 ns] as shown in Figure 3.13.

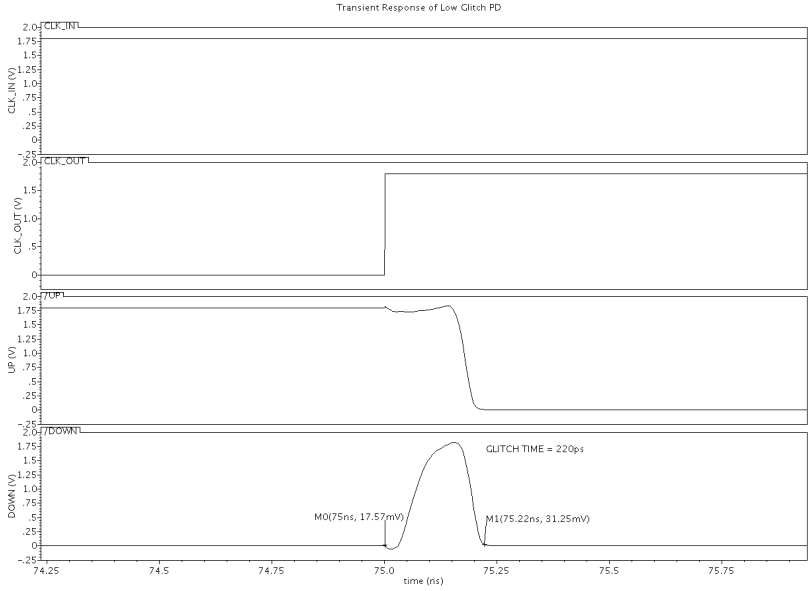


Figure 3.13 Zoomed view of Low Glitch PD [Glitch is from 75 ns to 75.22 ns].

3.3 CHARACTERISTICS OF LOW GLITCH PHASE DETECTOR

Phase sensitivity, maximum operating frequency, dead zone region and phase noise are the characteristics of any PD. Sensitivity of PD means the smallest difference the PD can detect and produce corresponding correct output signals, this leads to a conclusion that the higher the sensitivity, the better the PD. The low glitch PD is sensitive to even very small phase differences upto1 ns as shown below in Figure 3.14.

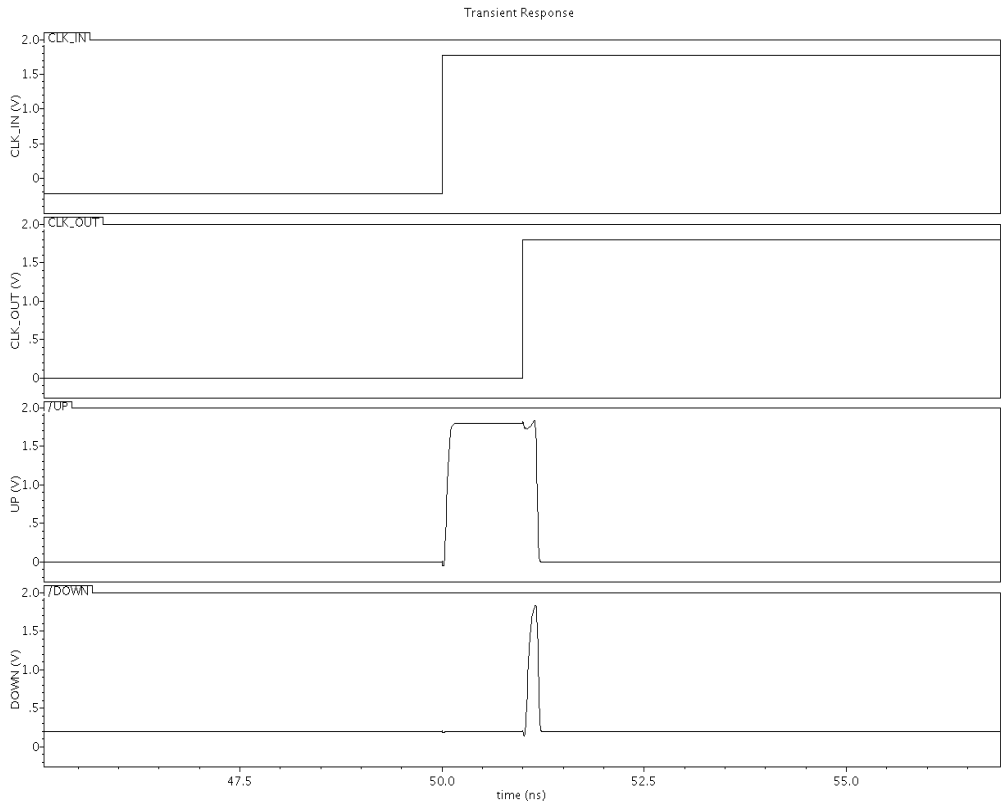


Figure 3.14 Simulation of Low Glitch PD with 1 ns Phase Difference.

The definition of maximum operating frequency is defined as one over the shortest period with correct PD output signals when the inputs have the same frequency and 90º phase difference. The maximum operating frequency is inversely proportional to the glitch width of the circuit. Since glitch is very less in the proposed low glitch PD, it can be operated over a higher range than the prior PD based on flip flop. The low glitch PD operates up to 0.5 GHz as shown in Figure 3.15.

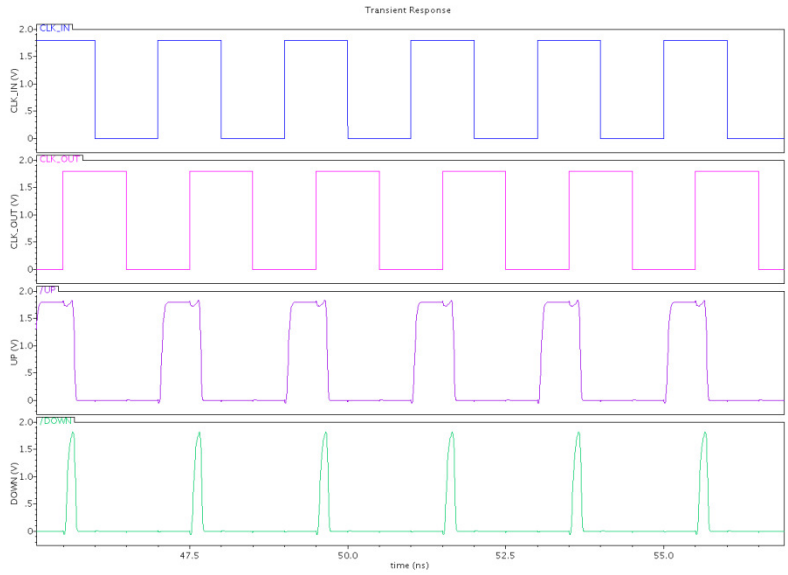


Figure 3.15 Low Glitch PD's 0.5 GHz operation.

A hypothetical PD produces no pulses for a zero input phase difference. Whenever a small phase error comes, due to finite rise time and fall time, the output of PD pulse may not find enough time to reach a logical high level. When the phase error is within the dead zone, the control system does not change the control voltage. In other words, if the input phase difference falls below a certain value, the output voltage of the PD is no longer a function of phase difference. The low glitch PD generates narrow, coincident pulses on both UP and DOWN even when the phase difference is zero. Effectively the coincident pulses on UP and DOWN can eliminate the dead zone.

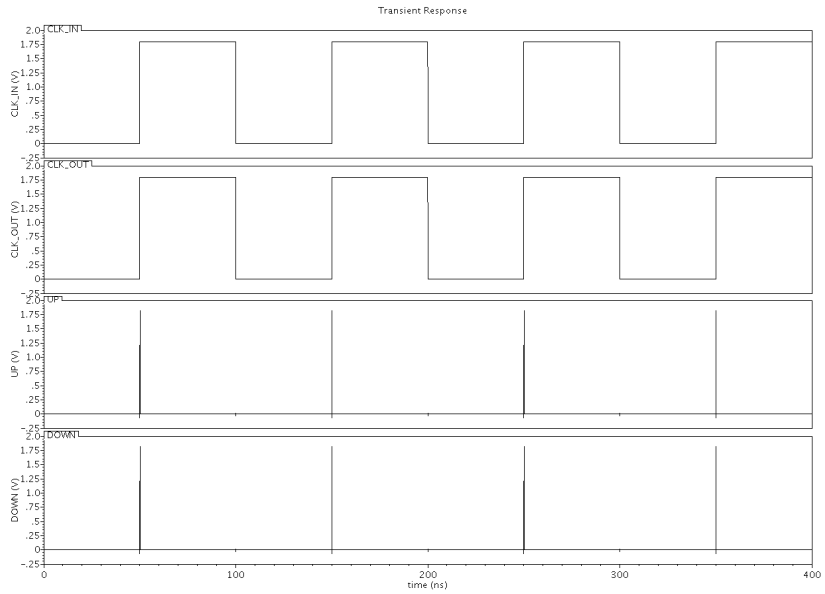


Figure 3.16 Generation of Coincident Pulses when the Phase Difference is Zero.

Phase noise simulation of Low Glitch PD is shown in Figure 3.17. CLK_IN and CLK_REF are 50 MHz and have the same phase approximately. Phase noise computes the total noise contribution of the input signal to the circuit. Noise is contributed to PD from different sources such as dead zone and transistors. The phase noise of a DLL based system is affected by two main factors namely the phase noise of the input frequency by the crystal oscillator and the phase noise contribution by the delay chain. The phase noise of Low Glitch PD and flip-flop based PD is almost same. Low Glitch PD has phase noise of -156 dBc/Hz while flip-flop based PD has -156.3 dBc/Hz at 1 MHz offset as shown below in Figures 3.17 and 3.18 respectively.

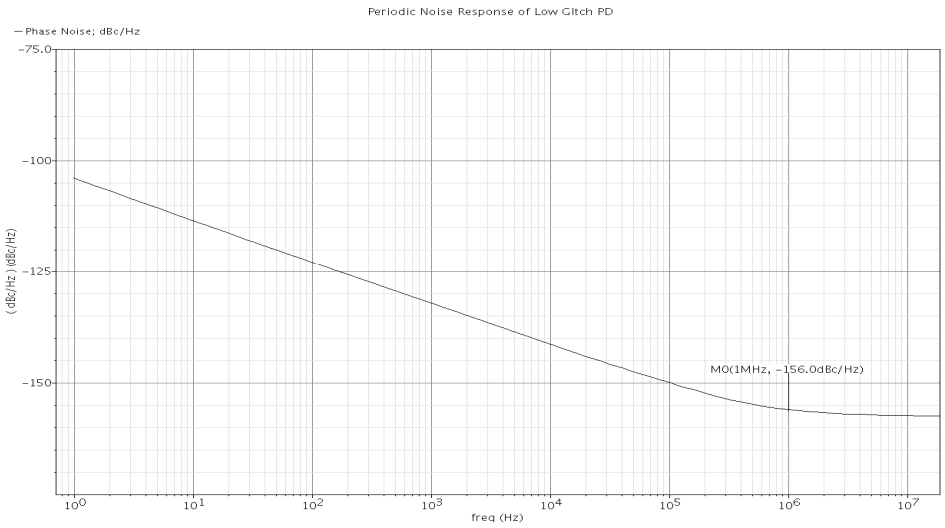


Figure 3.17 Phase Noise of Low Glitch PD.

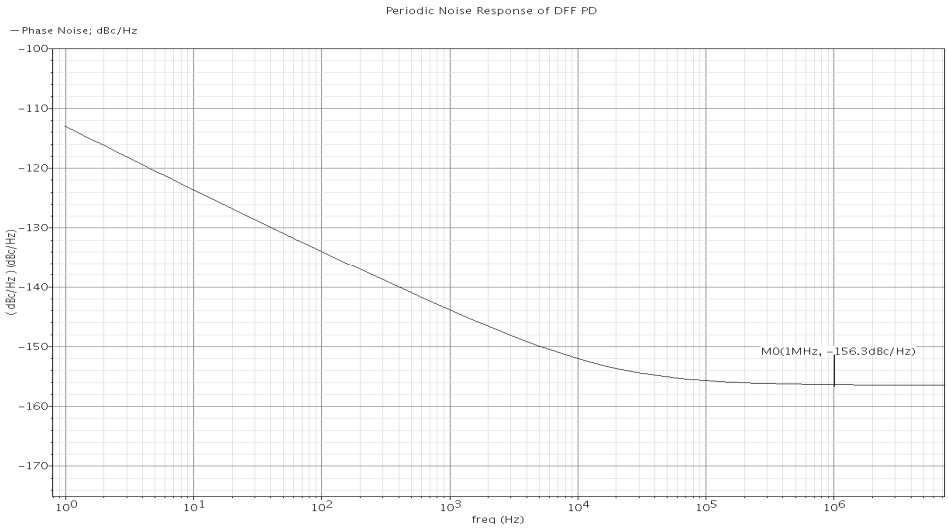


Figure 3.18 Phase Noise of Flip-Flop based PD.

3.4 COMPARISON BETWEEN LOW GLITCH PD AND D FLIP-FLOP BASED PD

A high speed, low glitch phase detector is proposed in 180 nm technology with V_{DD}=1.8V in Cadence Schematic Composer for schematic capture and analog artist (Spectre) Tool for simulations. The proposed PD is having a better phase sensitivity, no dead zone and a higher frequency of operation. Simulation results show that the proposed PD has low glitch as compared to conventional PD based on flip-flop. So, the speed of the proposed PD is also high.

TABLE 3.1

COMPARISON OF LOW GLITCH PD & DFF PD

Comparison	**D Flip-Flop based PD**	**Proposed Low Glitch PD**
Technology	180 nm	180 nm
V_{DD}	1.8 V	1.8 V
Dead Zone	Zero	Zero
Glitch Time	590 ps	220 ps
Glitch Period	75.00 ns to 75.59 ns	75.00 ns to 75.22 ns
Power Dissipation	221.686 μW	661.262 pW
Transistors Counts	38	30

CHAPTER 4 CHARGE PUMP

4.1 INTRODUCTION

In a Delay-locked Loop, the phase error between the input reference clock and the VCDL output clock is detected by the Phase Detector and transferred to the Charge Pump in the form of voltage pulses or current pulses. The CP performs the function of adjusting the voltage of the loop filter and thereby altering the VCDL delay according to the phase error information from the PD. In principle, the CP simply consists of two controlled switches, one current source, and one current sink, as shown in Figure 4.1.

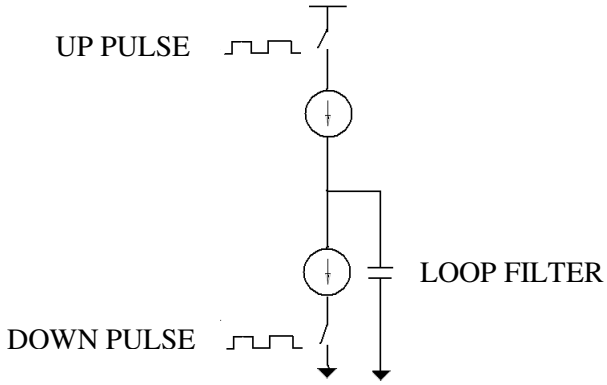


Figure 4.1 A Simplified Charge Pump.

The two switches are controlled by the UP pulses and the DOWN pulses, respectively. Once the switch is closed, the current source or sink will start adding charge onto or removing charge from the loop filter (capacitor). This charging or discharging process will continue until lock is achieved. In the locked state, the voltage (charge) of the loop filter is kept constant. However, the charging and discharging currents must be identical as well as very narrow so that the voltage of the loop filter will not be disturbed.

Both a single-ended topology and a differential topology CP exist in practical implementations. A single-ended topology has the advantages of smaller area and less power dissipation, but is more vulnerable to supply and substrate noise compared to a differential topology. Nevertheless, a single-ended topology is still popular in CP designs, because a single-ended CP does not need an additional loop filter and its power consumption is lower.

4.2 CURRENT MIRROR CONFIGURATIONS

A current mirror is a circuit designed to copy a current through one active device by controlling the current in another active device of a circuit, keeping the output current constant regardless of loading. The current being 'copied' can be, and sometimes is, a varying signal current. The current mirror is used to provide bias currents and active loads to circuits. There are three main specifications that characterize a current mirror. The first is the current level it produces. The second is its output resistance, which determines how much the output current varies with the voltage applied to the mirror. The third specification is the minimum voltage drop across the mirror necessary to make it work properly. This minimum voltage is dictated by the need to keep the output transistor of the mirror in active mode. The range of voltages where the mirror works is called the compliance range.

4.2.1 CONCEPTUAL MEANS OF COPYING CURRENTS

The basic copying principle used in current mirrors is described here in this section. For a MOSFET, if $I_D = f(V_{GS})$, where $f(\cdot)$ denotes the functionality of I_D versus V_{GS}, then $V_{GS} = f^{-1}(I_D)$. That is, if a transistor is biased at I_{REF}, then it produces $V_{GS} = f^{-1}(I_{REF})$ as shown in Figure 4.2 [16]. Thus if this voltage is applied to the gate and source terminals of a second MOSFET, the resulting current is

$$I_{OUT} = ff^{-1}(I_{REF}) = I_{REF} \tag{4.1}$$

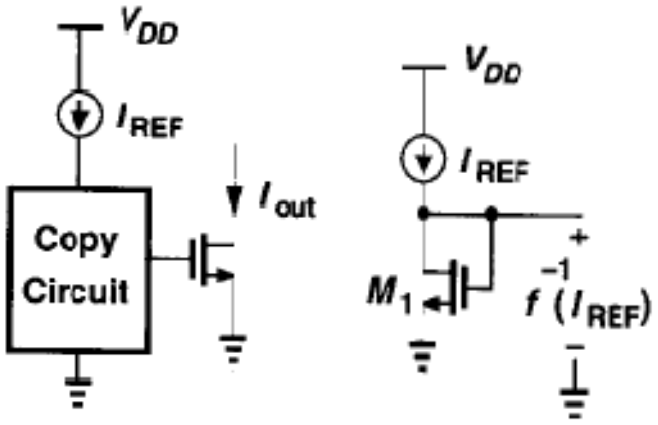


Figure 4.2 Ideas of Current Copying [16].

The proposed charge pump is based on new high output voltage compliance, high output resistance current mirror inspired from the high-swing super-Wilson current mirror. First, the Super-Wilson and the high-swing Super-Wilson current mirrors will be reviewed, and then the current mirror architecture used in the proposed charge pump will be presented.

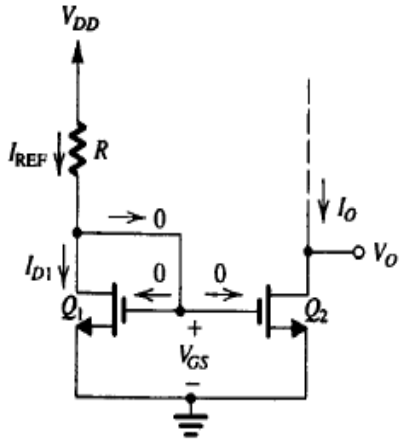


Figure 4.3 Circuit for a basic constant current source [18].

The operation of a basic current mirror circuit is explained with reference to Figure 4.3 which gives a basic current source using NMOS transistors. The heart of the circuit is transistor Q_1 whose drain is shorted to its gate and thus is operating in the saturation region, such that

$$I_{D1} = \frac{1}{2} k_n' \left(\frac{W}{L}\right) (V_{GS} - V_t)^2 \qquad (4.2)$$

where the channel length modulation is neglected. Here k_n' is the process transconductance of a NMOS transistor, W and L are the width and length of transistor respectively.

The drain current of Q_1, is supplied by V_{DD} through resistor R. Since the gate currents are zero,

$$I_{D1} = I_{REF} - \frac{V_{DD} - V_{GS}}{R} \tag{4.3}$$

where the current through R is considered to be the reference current of the current source and is denoted I_{REF}.

Transistor Q_2 has the same V_{GS} as Q_1. Thus, if it is assumed that it is operating in saturation, its drain current, which is the output current I_O of the current source, will be

$$I_o = I_{D2} = \frac{1}{2} k_n' \left(\frac{W}{L}\right)_2 (V_{GS} - V_t)^2 \tag{4.4}$$

From above equations, the output current I_O can be related to the reference current I_{REF},

$$\frac{I_O}{I_{REF}} = \frac{(W/L)_2}{(W/L)_1} \tag{4.5}$$

Thus, the relationship between I_O and I_{REF} is solely determined by the geometry of the transistors. When the transistors are of the same size, then $I_O = I_{REF}$, and the circuit simply mirrors the reference current in the output terminal, and is hence called a current mirror circuit. The operation of a PMOS current is quite similar to that of a NMOS current mirror, except that a PMOS current source sources the current from the power supply, and the NMOS current source sinks the current to the ground [18].

4.2.2 SUPER-WILSON CURRENT MIRROR

The Super-Wilson current mirror, shown in Figure 4.4, achieves high output resistance by using negative feedback. The current mirror formed by M2–M4 samples the output current I_{OUT} and its value is compared with that of the input current source I_{IN}. As a result, the gate voltage of M3 is adjusted so it sinks a current equal to I_{IN}. The output

resistance is directly proportional to the magnitude of the loop-gain of the feedback action from the output current to the gate of the output transistor M3. This loop-gain can be attributed to transistor M2 which, in combination with current source load I_{IN}, form a common-source amplifier used to maintain the gate voltage of output transistor M3 such that I_{OUT} is equal to I_{IN}. Due to channel-length modulation, the output current will be equal to I_{IN} only if the drain-source voltages of M2 and M4 are equal, otherwise an undesired current offset will result. In order to overcome this problem, the Super-Wilson current mirror uses a diode-connected transistor M1 in series with current source I_{IN} to lower the drain voltage of M2 and make it equal to that of M4 [19].

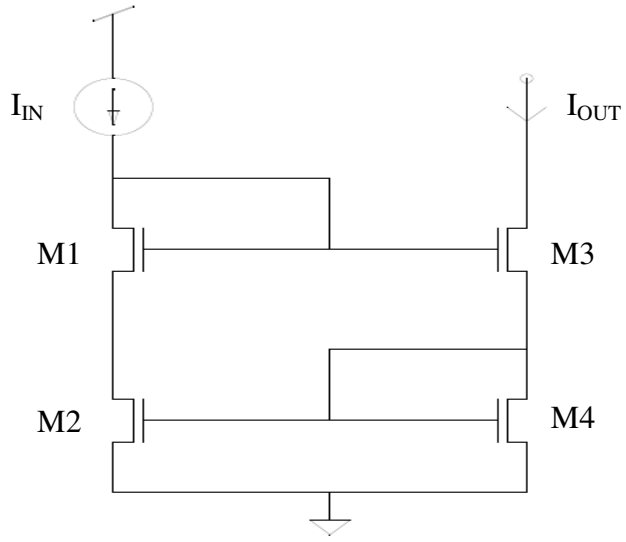


Figure 4.4 CMOS Implementation of the Super-Wilson Current Mirror.

The output impedance of the Super-Wilson current mirror of Figure 4.4 can be found using a test voltage source v_x at the output and finding the resulting current i_x flowing into the output port. If we assume that the output resistance of the current source I_{IN} is infinite, the effect of diode-connected transistor M1 is eliminated and the small-signal incremental output resistance $r_{out} = v_x/i_x$ is given as

$$r_{out} = r_{o3}\left(1 + \frac{g_{m3}(1+g_{m2}r_{o2})+g_{o3}}{g_{m4}+g_{o4}}\right) \approx g_{m2}r_{o2}r_{o3} \tag{4.6}$$

where $g_{m,}$ and r_o are, respectively, the transconductance and incremental output resistance of the transistors, and $g_o$is their output conductance. In this equation, body

effect, drain-induced barrier lowering (DIBL), and hot carriers effects are neglected for the sake of simplicity, since the goal is to give an idea on the order of magnitude of r_{out} rather than finding an exact value.

The Super-Wilson current mirror has a very-high output resistance, but it requires an output voltage larger than a V_{GS} plus a saturation voltage due to the simple current sampling mirror connected in series with the output. As a result, the Super-Wilson current mirror is unusable in low-supply voltage applications.

4.2.3 HIGH-SWING SUPER-WILSON CURRENT MIRROR

The high-swing Super-Wilson current mirror, shown in Figure 4.5 solves the above problem by sensing the output current with a very low input voltage current mirror that directly replaces the simple current mirror used in the Super-Wilson current mirror [19]. The high-swing Super-Wilson current mirror has an output compliance voltage of about two saturation voltages, and its incremental output resistance can also be approximated by (4.5).

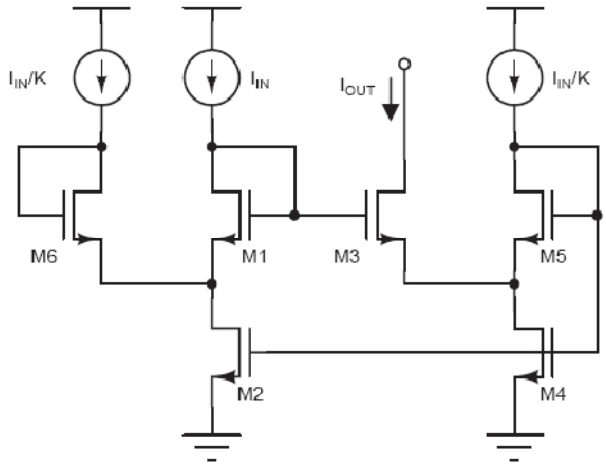


Figure 4.5 CMOS Implementation of the High-Swing Super-Wilson Current Mirror [19].

In the Super-Wilson and high-swing Super-Wilson current mirrors presented above, the role of the diode-connected transistor M1 is to reduce unwanted offset in the output current by setting the drain of grounded transistors M2–M4 approximately to the same voltage. However, this diode-connected transistor along with the current source I_{IN} forms

a voltage divider from the drain of M2 to the gate of M3. Therefore, the gain of the common-source amplifier formed by M2 and I_{IN} is reduced when a non-ideal current source is used. The gain of the feedback path common-source amplifier would be significantly higher if one replaced the diode connection of transistor M1 with a cascode one, as is proposed in Figure 4.5. By doing so, the loop-gain is increased by a factor $g_m r_o$, and so is the closed-loop output resistance of the current mirror [19]. Therefore, assuming that the output resistance of the current sources I_{CP}/K and I_{CP} is infinite, the incremental output resistance of the proposed current mirror is given by

$$r_{out} = r_{o3}\left(1 + \frac{g_{m3}(1+g_{m1}g_{m2}r_{o1}r_{o2})+g_{o3}}{g_{m4}g_{o4}}\right) \approx g_{m1}g_{m2}r_{o1}r_{o2}r_{o3} \qquad (4.7)$$

Comparing this result with (4.6), we see that the proposed implementation yields to an increase of the output resistance by a factor of $g_{m1}r_{o1}$ over that of the high-swing Super-Wilson current mirror.

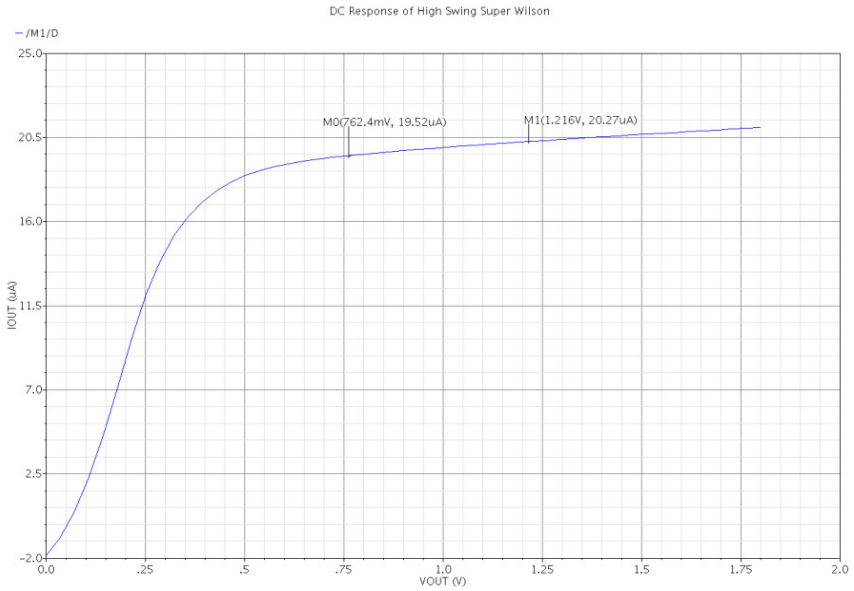


Figure 4.6 Output Current Variation as a function of the Output Voltage for the High-Swing Super-Wilson Current Mirror.

The output current variation with respect to output voltage ranging from 0 to 1.8*V* clearly shows that the compliance range of high-swing Super-Wilson current mirror is very less. About 4% of rise occurs in the output current just in the range of 0.762*V*-1.216*V* of output voltage as shown in Figure 4.6.

4.2.4 ENHANCED LOW-VOLTAGE CURRENT MIRROR

With transistor M1 cascoded instead of diode-connected, the problem of matching the drain-source voltages of grounded transistors M2 and M4 comes up again. The key design consideration in matching these drain-source voltages is to make sure that M2 and M4 are operating at the exact same saturation level. Therefore, we need to bias cascode transistor M1 with a voltage V_{bn_cascd} that will place bottom transistor M2 in the exact same saturation level as that of M4.

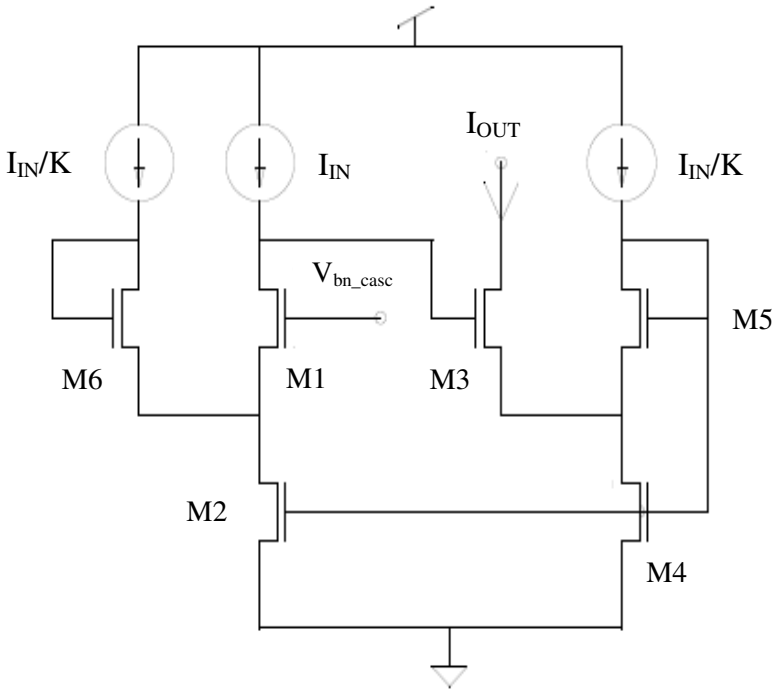


Figure 4.7 Enhanced Low-Voltage Current Mirror Schematic.

The output current variation with respect to output voltage ranging from 0 to 1.8*V* clearly shows that the compliance range of proposed enhanced low-voltage current mirror is quite impressive as compared with high-swing Super-Wilson current mirror as shown in Figure 4.8.

In order to illustrate the increase in output resistance resulting from cascading transistor M1, a comparison of the output current as a function of the output voltage for the high-swing Super-Wilson current mirror and the proposed current mirror is shown in Figure 4.9.

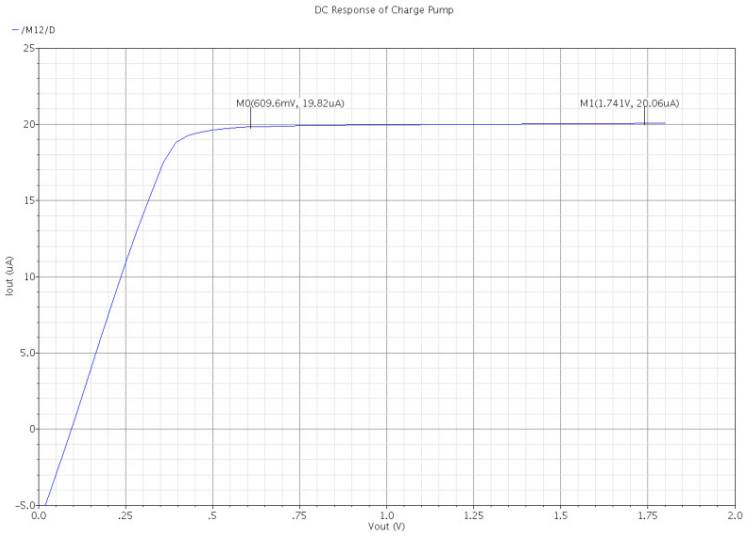


Figure 4.8 Output Current Variation as a function of the Output Voltage for Proposed Current Mirror.

The improvement in the output impedance resulting from the cascade connection of transistor M1 in the proposed current mirror is significant; the output current increases by 1.25% when the output voltage goes from 0.596*V* to 1.76*V*, whereas that of the high-swing Super-Wilson increases by 9.38% for the same output voltage range. That is, the cascade connection of transistor M1 increases the output resistance by a factor of 7.5 as shown in Figure 4.9 below.

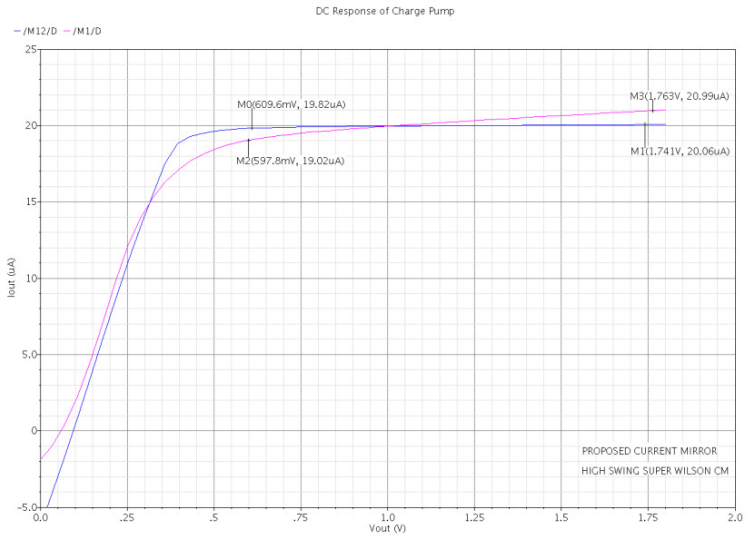


Figure 4.9 Output Current variations as a function of the output voltage for the high-swing Super-Wilson current mirror and the enhanced low voltage current mirror.

4.3 IMPLEMENTATION OF CHARGE PUMP

The enhanced current mirror described in the previous section was used to implement a charge pump with accurate UP/DOWN current matching and a very-high output resistance over a wide output voltage range. The resulting charge pump circuit is shown in Figure 4.10. The lower part of the charge pump, used to sink current is basically the current mirror of Figure 4.7 to which control switches S1–S2 were added in order to enable or disable the current mirror. The upper half of the charge pump is used to source a current I_{CP} and is implemented using complementary devices.

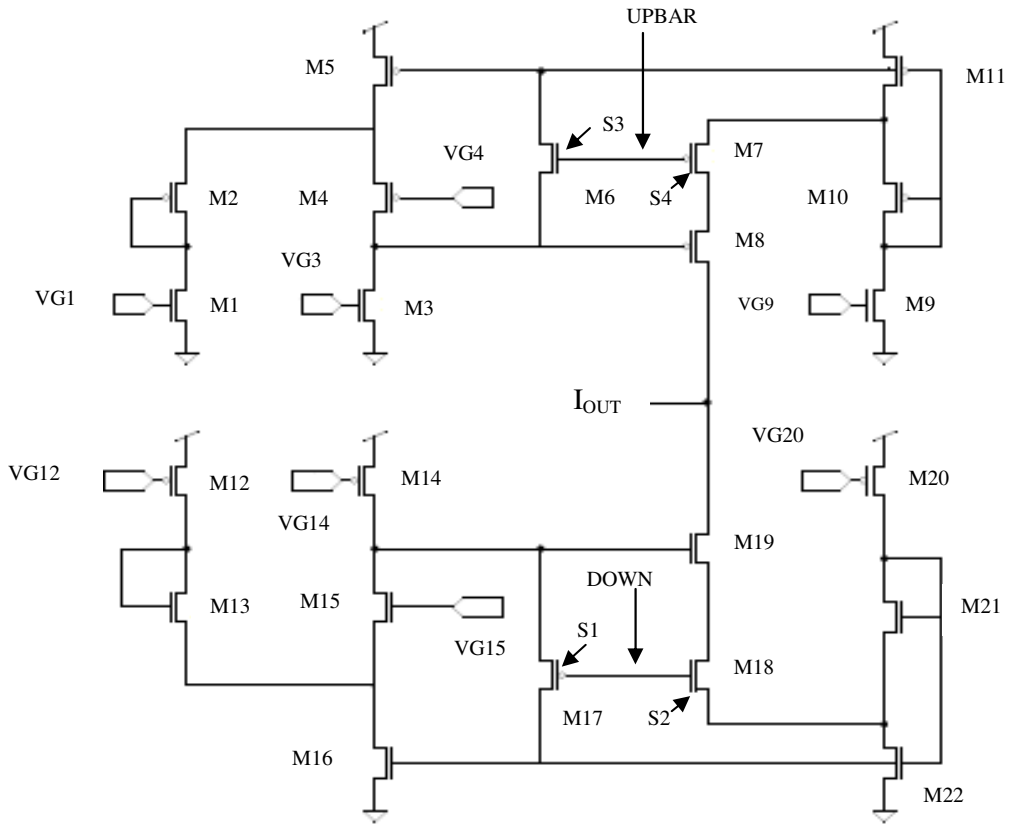


Figure 4.10 Simplified schematic of the Proposed Charge Pump.

Let us consider the lower half circuit of the charge pump, used to sink a current I_{CP}. When the signal DOWN goes high, the NMOS switch S2 closes the feedback loop whereas the PMOS switch S1 goes off. The feedback action forces I_{OUT} to be precisely equal to I_{CP}. As will now be explained, PMOS switch S1 is used to minimize glitches in the output current. If switch S1 was absent and DOWN was low, the output current I_{CP} would not flow in M4 and would not be mirrored into M2, thus pulling the drain of M1 towards V_{DD}.

This would result in a large current glitch at the next transition of DOWN going high. Switch S1 solves this problem by connecting the gates of mirror transistors M2, M4 and M5 to the output of the reference current source I_{CP} and DOWN is low. As a consequence, the current sourced by I_{CP} keeps flowing in the feedback branch even when the loop is open.

The value of the auxiliary bias current sources I_{CP}/K influences the output voltage range and output resistance. The value of these auxiliary current sources also has an impact on the settling time and maximum speed of the charge pump.

4.4 LOOP FILTER

The signal coming from the phase detector and charge pump combination has a large periodic content due to the recurring phase detection. To remove the high frequency component and average the charge pump output, a loop filter is applied at the output of the charge pump. This filter output is then used as the control voltage for the delay chain.

In DLL's there is no need to include a low pass filter, instead a capacitor is used in order to integrate the phase error mainly, and thus increasing control voltage appropriately. So filter block is simply a capacitor, and a 500 *fF* capacitor was used. Increasing capacitor increases the lock time while decreases the bandwidth and the ripples on the control voltage, and decreasing it decreases lock time and increases the ripples.

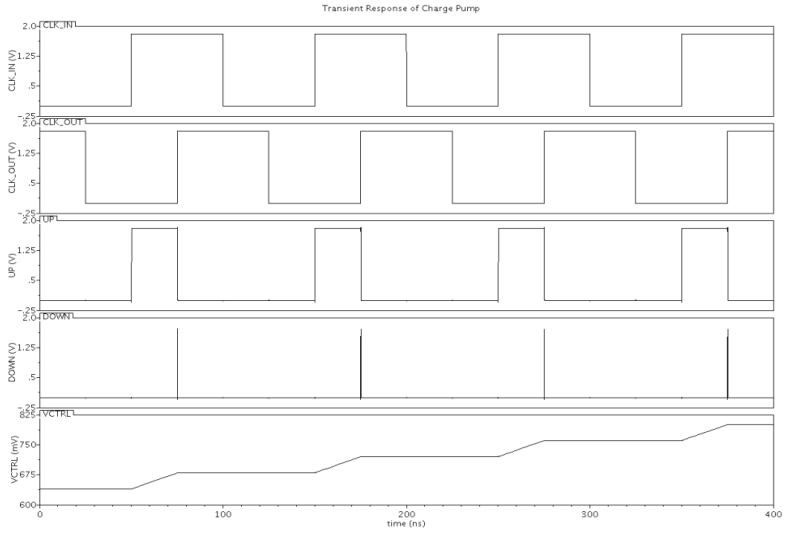


Figure 4.11 Charge Pump Simulation with UP signal high.

If for example, CLK_IN leads CLK_OUT, then UP continues to produce pulses and charge pump output rises steadily. Conversely CLK_OUT leads CLK_IN then DOWN continues to produce pulses and output falls steadily. The currents through the PMOS and NMOS branch are nominally equal.

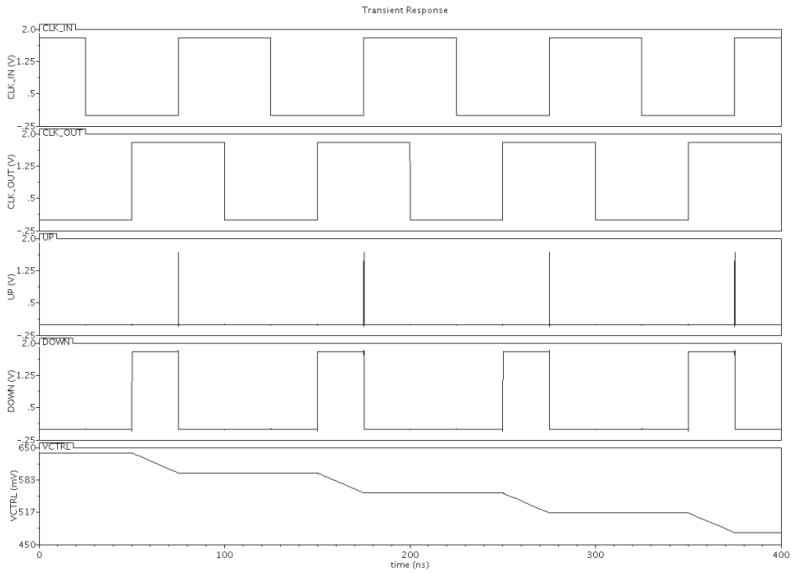


Figure 4.12 Charge Pump Simulation with DOWN signal high.

CHAPTER 5 VOLTAGE CONTROLLED DELAY LINE

5.1 INTRODUCTION

The Voltage Controlled Delay Line is one of the most critical blocks within a DLL as the output signal of the DLL is directly taken from the VCDL. The VCDL's performance considerably affects the lock range and the stability of the DLL. A VCDL typically consists of a number of delay stages (cells) which are connected in series. A VCDL is an open loop configuration by itself, so it does not oscillate and thereby is different from the voltage-controlled oscillator (VCO) in a PLL.

Figure 5.1 A Typical VCDL Configuration.

Basically, all the delay stages in the VCDL are designed to be identical. Therefore, each delay stage contributes a time delay of T_{REF}/n for an n stage VCDL. Using more stages increases the phase resolution, but also increases the minimum VCDL delay. The delay elements of VCDL are controlled by the control voltage generated by the Charge Pump block. This control voltage determines the current through the current-mirror arrangement that derives the delay-element control currents from a single control current.

If the final output phase lags the reference clock i.e. it has a lower time period compared to the reference clock, then the control voltage decreases thus decreasing the current through the delay cells and thereby increasing the delay of each delay element. Thus, finally the overall time period of the clock phases is increased to match the reference clock period. Once this lock is achieved, the control voltage remains stable and the delay cells maintain the delay locked to the clock time period.

Exactly opposite thing happens when the output clock phase has higher time period compared to the reference clock. The control voltage increases in this case, thus decreasing the delay offered by each cell, till the time period of each clock phase locks to the reference clock period.

The function of Voltage Controlled Delay Line (VCDL) is to delay the reference signal so that there is no skew between the output clock and reference signal. Its transfer function is simply,

$$t_d = k_{VCDL} V_{CTRL} \tag{5.1}$$

where V_{CTRL} is the control input to the VCDL from charge pump, t_d is the delay and k_{VCDL} is the gain of VCDL.

5.2 VARIABLE DELAY ELEMENTS

There are several different methods for implementing a delay element. Each of these methods has its advantages and drawbacks. There are two popular techniques for designing a variable delay element. These are known as: shunt capacitor technique and current starved technique.

The delay range of Current-Starved inverter circuit is higher than Shunt-Capacitor circuit, however Shunt-Capacitor transfer function is more linear and has lower gain than that of Current-Starved one as shown in Figure 5.2 [20]. The delay range is more important as it sets the operating frequency of the DLL.

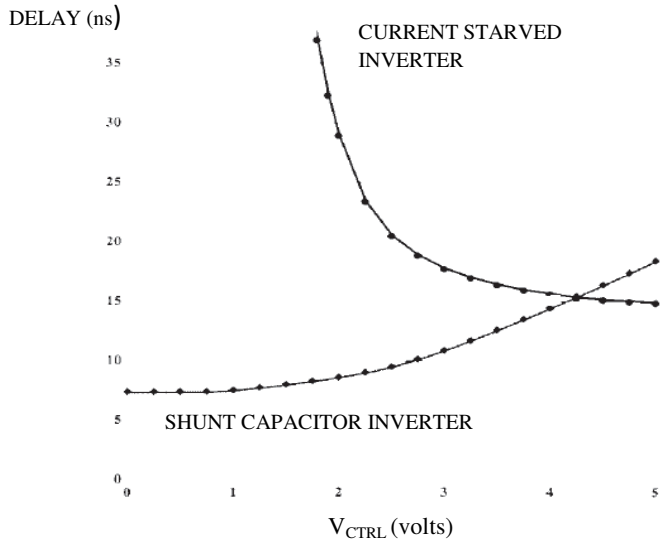
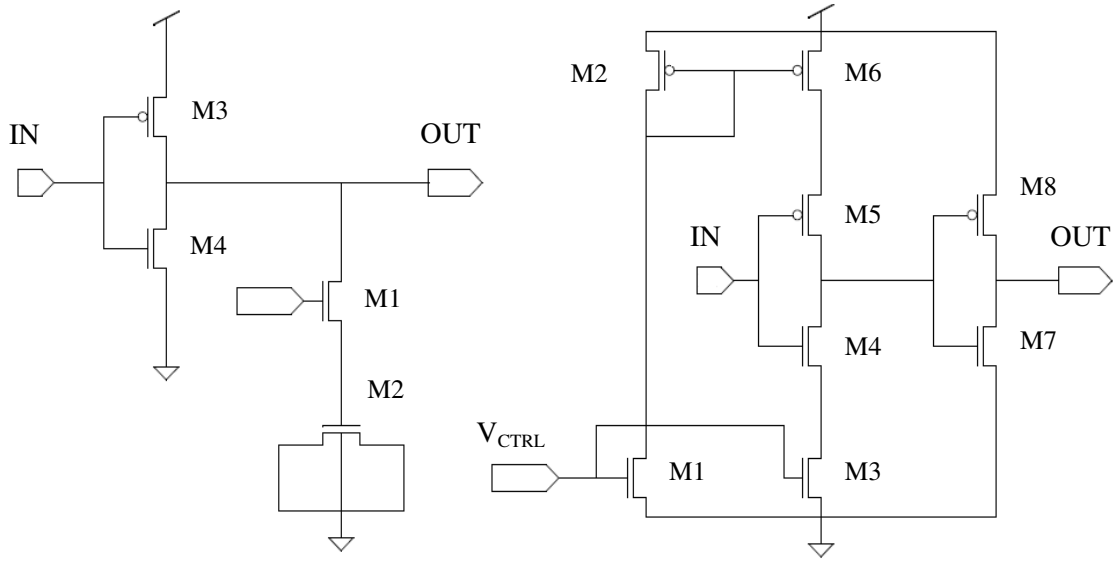


Figure 5.2 Delay Time Versus Control Voltage [20].

Figure 5.3(a) shows the basic circuit of using a shunt capacitor. In this circuit, M2 acts as a capacitor. Transistor M1 controls the charging and discharging current to the M2 from the NOR gate. The M1 gate voltage, V_{CTRL} , controls the (dis)charge current.



(a) (b)

Figure 5.3 (a) Shunt-Capacitor Delay Stage (b) Current-Starved Delay Stage.

Figure 5.3(b) illustrates the basic building block of a current starved delay element. As can be seen in this figure, there are two inverters between input and output of this circuit. The charging and discharging currents of the output capacitance of the first inverter,

composed of M4 and M5, are controlled by two MOS transistors, M3 and M6. Charging and discharging currents depend on the gate voltage of M6 and M3 transistors, respectively. M1 and M2 constitute a current mirror for controlling the gate voltage of M6. The second stage inverter (composed of M7 and M8) is for improving the rise and fall times of the circuit. Sometimes, multiple cascaded inverters are used for this purpose.

5.3 THE PROPOSED VCDL

Figure 5.4 shows the architecture of the delay element used in voltage controlled delay line. As can be seen in this figure, a current starved buffer, M3–M7, is the main element. The controlling current through this buffer is controlled by a current mirror circuit composed of transistors M2–M5. An appropriate current through M5 can be adjusted by turning-on transistor M1.

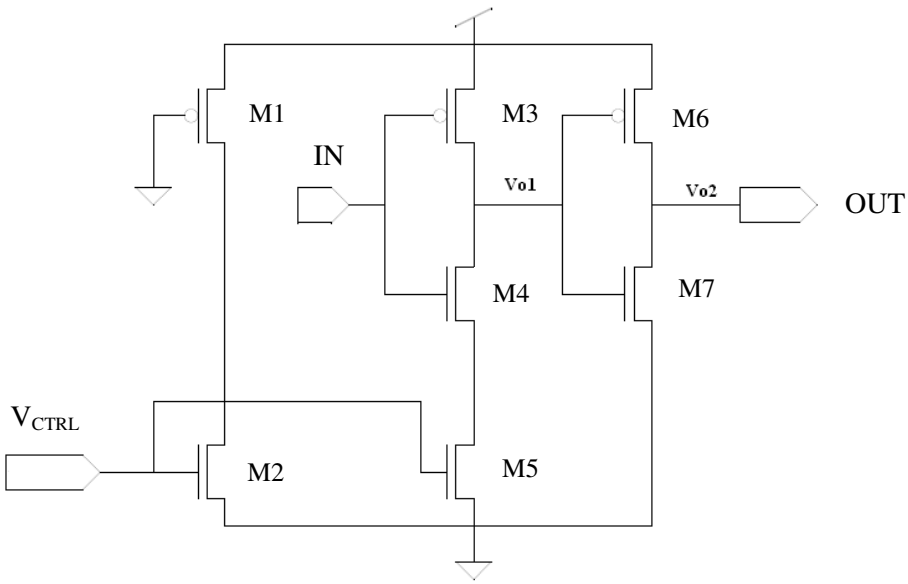


Figure 5.4 Proposed Delay Element.

At the instance when M4 turns on, the capacitor at its output node starts to discharge. The discharging current is controlled by transistor M5 acting as a current source. The passing current through this transistor is determined by the gate voltage of M2. The gate voltage of M2, in turn, is determined by the current passing through its drain.

5.4 MATHEMATICAL MODEL OF DELAY ELEMENT

Figure 5.4 shows part of the delay element. In order to have a better controllability, the W/L ratio of transistor M8 should be much bigger than that of M7. In such an arrangement the current is controlled by M7.

In order to find a relationship between V_g (the gate voltage of transistor M7 and/or M6), and the delay of the circuit (t_d), we should calculate the current passing through transistor M7. Once this current is known, one can find the output voltage. Transistor M7 is a relatively small transistor with a channel length of 0.18 μm. Hence, we can consider the following for the drain current of this transistor:

$$i_d = \frac{k_n W_7}{2L_7}\left(V_g - V_{T7}\right)(1 + \lambda_7 V_{DS7}) \tag{5.2}$$

Equation (1) is valid as long as the transistor is in the saturation region. This is true for most of the transition time because the gate voltage of M7 is not much bigger than its threshold voltage. Moreover, we assume that the voltage drop across M8 is very small so that $V_{DS7} \cong V_{o1}$. The output voltage V_{o1} can be found from the following equations:

$$-C_{L1}\frac{dV_{o1}}{dt} = \frac{k_n W_7}{2L_7}\left(V_g - V_{T7}\right)(1 + \lambda_7 V_{DS7}) \tag{5.3}$$

$$-C_{L1}\frac{dV_{o1}}{dt} = K_1 + K_1\lambda_7 V_{o1} \tag{5.4}$$

where C_{L1} represents the overall capacitance at node V_{o1} and

$$K_1 = \frac{k_n W_7}{2L_7}\left(V_g - V_{T7}\right) \tag{5.5}$$

Solving the above differential equation with initial condition of $V_{o1} = V_{DD}$ at $t = 0$ results to the following for V_{o1}

$$V_{o1} = (V_{DD} + 1/\lambda_7)e^{-t/\tau 1} - 1/\lambda_7 \tag{5.6}$$

where $\tau_1 = C_{L1}/K_1\lambda_7$. At $t = t_{d1}$ (inverter delay from *IN* to V_{o1}) $V_{o1} = V_{DD}/2$. Hence

$$t_{d1} = \tau_1 Ln\frac{1+\lambda_7 V_{DD}}{1+\lambda_7 V_{DD}/2} \tag{5.7}$$

To compute the circuit delay of this delay element, we should find V_{o2} as a function of time. At the instance when the input voltage (V_{in}) goes high, V_{o1} starts to fall and M10 starts to turn off. When V_{o1} becomes less than $V_{DD} - |V_{T6}|$, transistor M6 starts to conduct while transistor M7 starts to turn off. Hence, for a period of time, both M7 and M6 transistors are on. Owing to the current starved nature of the first inverter, the fall time of V_{o1} is not very small [21].

Therefore, the direct current passing through transistors M7 and M6 is not negligible. It is necessary to consider the current in both of these two transistors in order to find V_{o2}. However, this complicates the equations and defeats the purpose of a simple analytical model. We assume that the direct path current is negligible and can be ignored in these calculations. Moreover, ignoring the channel length modulation effect of M6, we can write

$$i_{d6} = \frac{k_p W_{11}}{2L_{11}}\left(V_{gs6} - V_{T6}\right)^2 \tag{5.8}$$

$$i_{d6} = C_L\frac{dV_{o2}}{dt} \tag{5.9}$$

The initial condition for the above differential equation is $V_{o2} = 0$ at $t = 0$. We can substitute V_{gs6} in the above equation by $V_{DD} - V_{o1}(t + t_p)$ where t_p is the time when V_{o1} reaches $V_{DD} - |V_{T6}|$ that is

$$t_p = \tau_1 Ln\frac{1+\lambda_7 V_{DD}}{1+\lambda_7(V_{DD}-|V_{T6}|)} \tag{5.10}$$

Combining (6)-(8) and solving the resulting equation, V_{o2} can be found as the following

$$V_{o2} = K_3 K_2^2 \tau_1\left(\frac{t}{\tau_1} + 2e^{-t/\tau_1} - \frac{1}{2}e^{-2t/\tau_1} - 1.5\right) \tag{5.11}$$

where

$$K_2 = V_{DD} + \frac{1}{\lambda_7} - |V_{T6}|$$

$$K_3 = k_p W_6 / 2L_6 C_L$$

The delay time of the circuit can be computed with (5.11).

Here I am using four delay elements in a chain to form a voltage controlled delay line. All the four clock outputs are separated by an equal delay with respect to each other starting from CLK_IN as clearly shown in figure below. The complete schematic of voltage controlled delay line used in DLL is shown below in Figure 5.5.

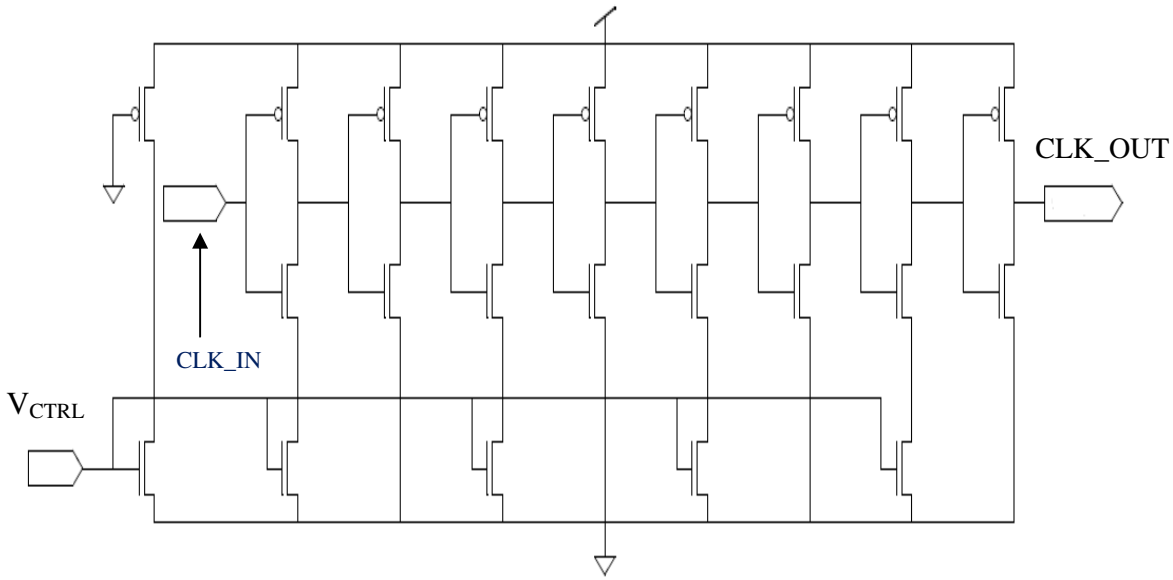


Figure 5.5 Voltage Controlled Delay Line.

The circuit delay of the voltage controlled delay line is a function of V_{CNTRL}. Table 5.1 below shows the functionality in between delay and V_{CNTRL}. Then based on the readings from the Table 5.1 a graph is drawn between circuit delay and V_{CNTRL} which surely justifies the relation shown in Figure 5.2.

TABLE 5.1

DELAY VERSUS CONTROL VOLTAGE

Control Voltage (mV)	Delay (ns)
550	4.90
600	2.91
650	2.06
700	1.55
750	1.24
800	1.04

Figure 5.6 clearly shows the relationship between the circuit delay and control voltage.

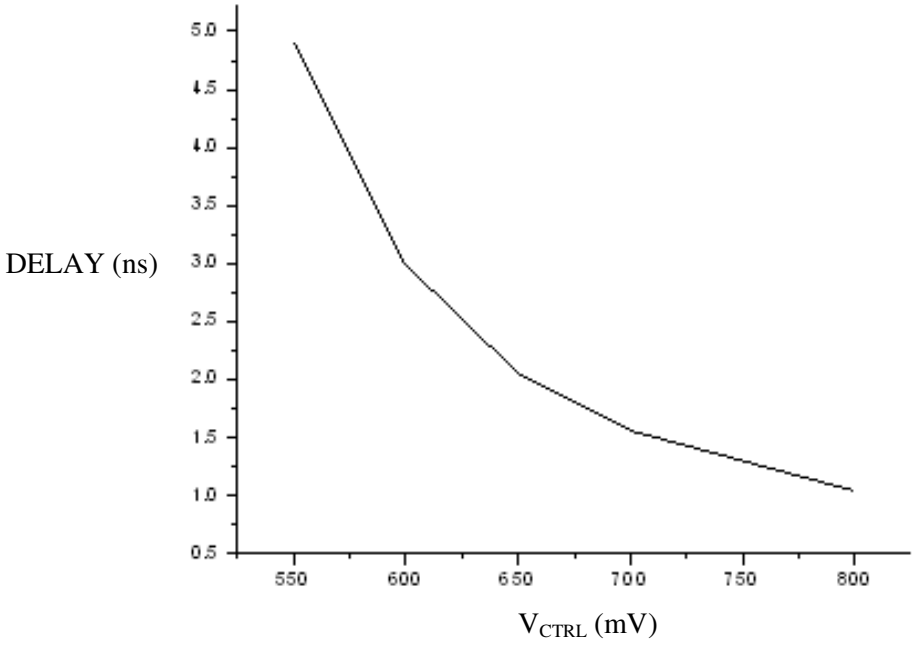


Figure 5.6 VCDL's Transfer Charachteristics.

The overall output waveform of a VCDL is shown below in Figure 5.7 below with CK1, CK2, and CK3 as intermediate output clocks.

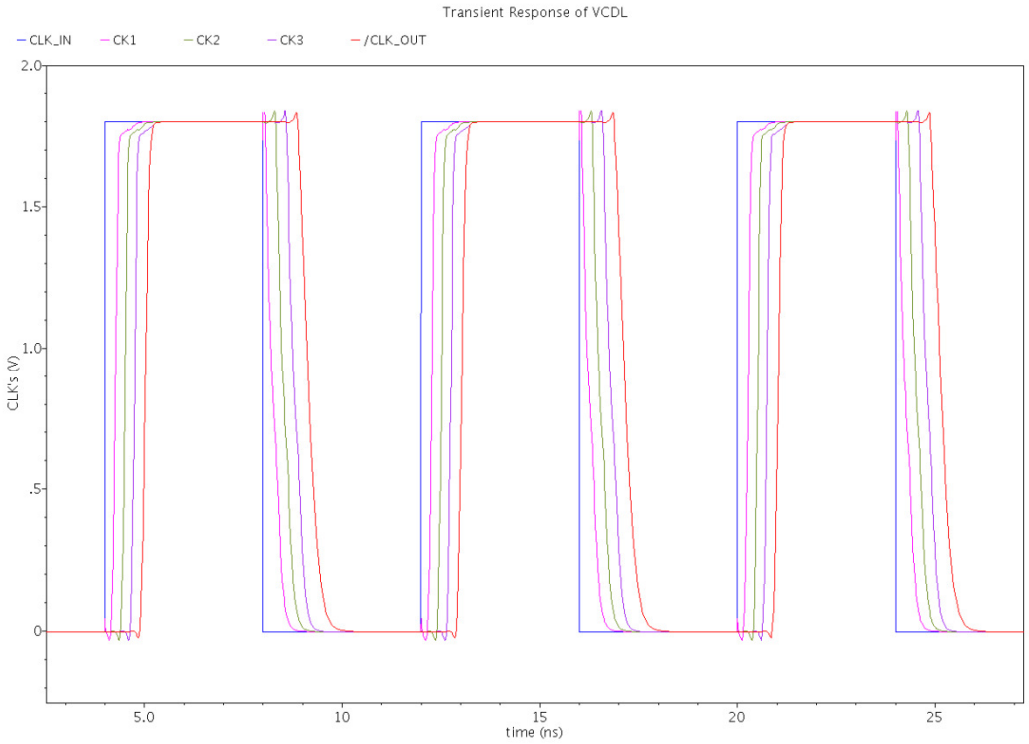


Figure 5.7 Simulation result of VCDL.

CHAPTER 6 PERFORMANCE EVALUATION

6.1 PROCESS VARIATIONS

In the semiconductor fabrication process, the characteristics of a MOS transistor can vary substantially from wafer to wafer and even from location to location on the same wafer. This variation is primarily due to the non-uniformity of the processes such as impurity concentration density, oxide thickness, and diffusion depth. Specifically, the following transistor parameters can vary with processes.

6.1.1 THE THRESHOLD VOLTAGE V_{tho}

The threshold voltage of an NMOS transistor is given as

$$V_{th0} = \emptyset_{MS} + 2\emptyset_F + \frac{Q_{dep}}{C_{ox}} \tag{6.1}$$

Where $\emptyset_{MS}$ is the difference between the work functions of the polysilicon gate and the silicon substrate, $\emptyset_F = (kT/q)ln(N_{sub}/n_i)$, Q_{dep} is the charge in the depletion region, and C_{ox} is gate oxide capacitance per unit area. As can be seen from equation 6-1, the threshold voltage V_{th0} may vary with the changes in the silicon substrate, polysilicon gate, doping concentration, surface charge, or oxide thickness. Similarly, the threshold voltage of PMOS transistor also varies as a function of the above parameters.

6.1.2 PROCESS TRANSCONDUCTANCE K'

Process transconductance K' can be expressed as

$$K' = \mu C_{ox}\,, \tag{6.2}$$

where μ is the mobility of the carrier. The main cause for variations in process transconductance is the change in oxide thickness t_{ox} .

6.1.3 DIMENSIONS W AND L

Limited resolution of lithographic processes can cause variations in the dimension of transistors (W and L). As a result, the W/L ratio of transistors may be affected.

The circuit is first simulated using the transistor models at corner conditions provided by UMC, namely, TT (typical NMOS, typical PMOS), FF (fast NMOS, fast PMOS), SS (slow NMOS, slow PMOS), FS (fast NMOS, slow PMOS), and SF (slow NMOS, fast PMOS). These transistor models represent the different characteristics of the transistors at extreme process corners.

6.2 SIMULATION TEST CIRCUIT

With all of the blocks designed, the DLL can finally be run as a system. Figure 6.1 is a final symbol view of DLL consisting of all the individual blocks. The given test circuit for DLL is simulated for different frequencies with process, temperature, and power supply variations. A schematic view of the circuit is created using the Cadence *Composer Schematic Editor*. After the transistor-level description of a circuit is completed using the Schematic Editor, the electrical performance and the functionality of the circuit must be verified using a Simulation tool. So the circuit is simulated using the Cadence *Spectre* tool.

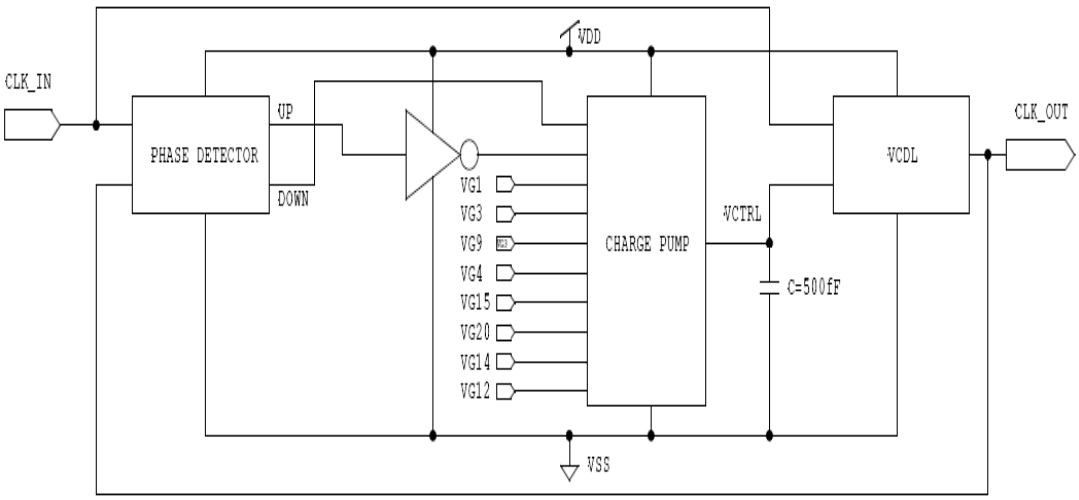


Figure 6.1 Symbol View of DLL.

The reference signal CLK_IN was a 50 MHz clock signal. VG1, VG9, VG12, and VG20 are the input ports consist of DC biasing to analyze the ideal current source of 2 μA whereas VG3 and VG14 are the input ports, with DC biasing; those are used to analyze the charge pump current of 20 μA. VG4 and VG15 are used to switch on the PMOS and NMOS transistors of the charge pump. A 500 fF capacitor is used in between Charge Pump and Voltage Controlled Delay Line as a Low Pass Filter.

After giving all the biasing and supply voltages, the DLL is analyzed at different process corners with variations in power supply and temperature. The overall DLL output waveforms are shown in the next section.

6.3 DLL OUTPUT WAVEFORMS

In this section, the DLL output waveforms are analyzed at different variations of process, temperature, and power supply. The proposed DLL functions correctly under TT, FF, SS, FS, and SF process corners. The phase alignment of CLK_IN with CLK_OUT is shown for each condition.

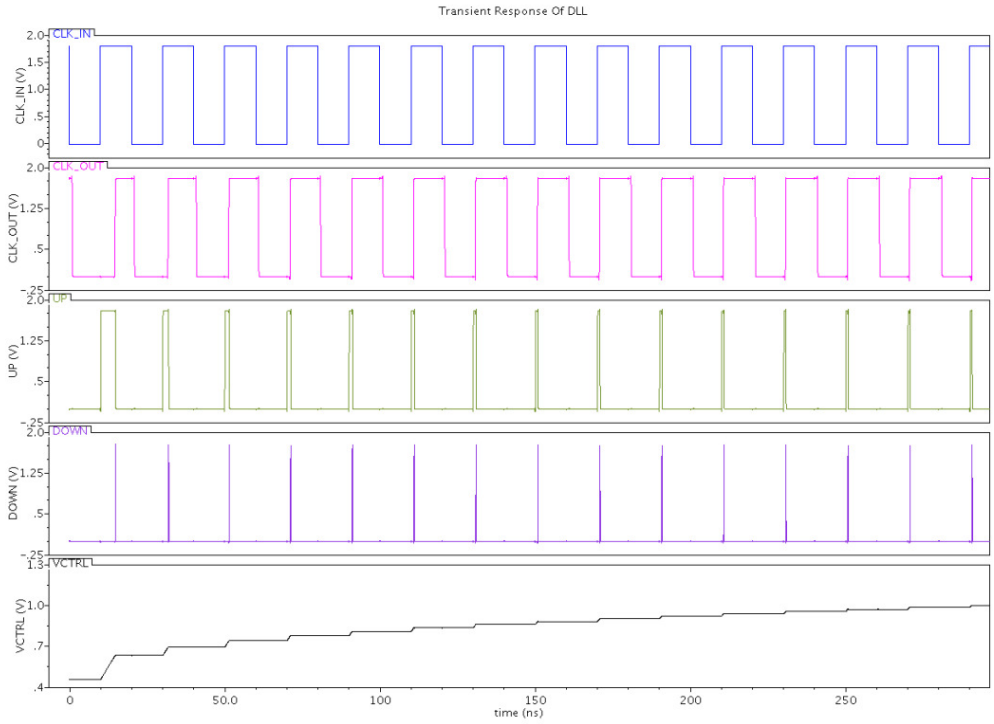


Figure 6.2 CLK_IN, CLK_OUT, UP, DOWN & VCTRL waveforms of DLL in unlocked state.

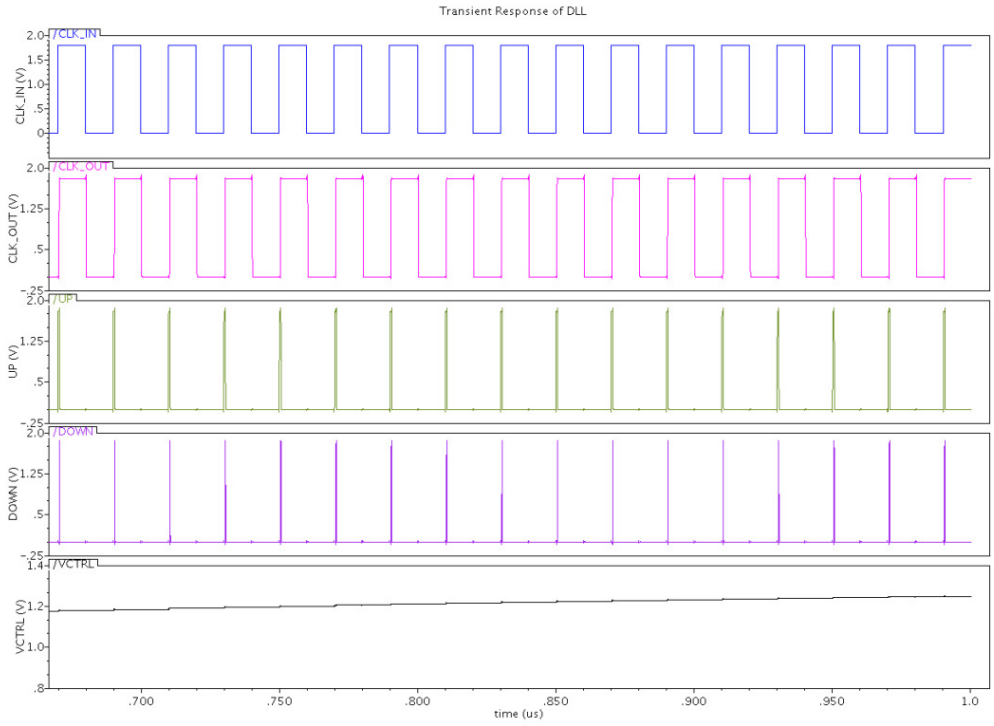


Figure 6.3 CLK_IN, CLK_OUT, UP, DOWN & VCTRL waveforms of DLL in locked state.

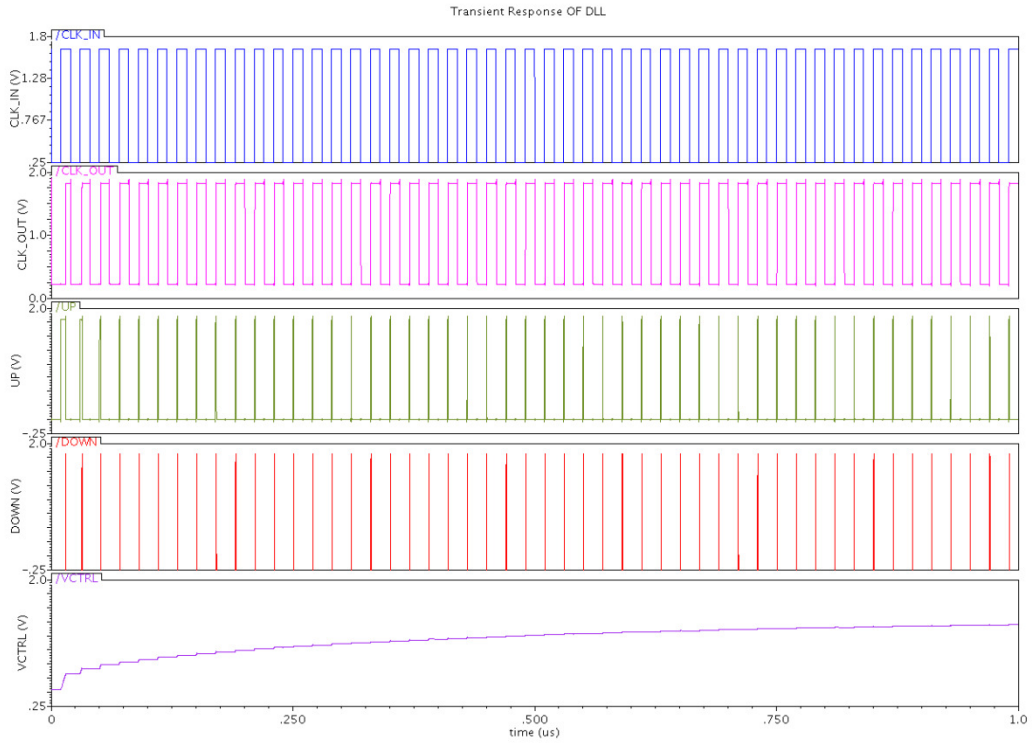


Figure 6.4 CLK_IN, CLK_OUT, UP, DOWN & VCTRL waveforms of DLL in a typical process @ 50MHz.

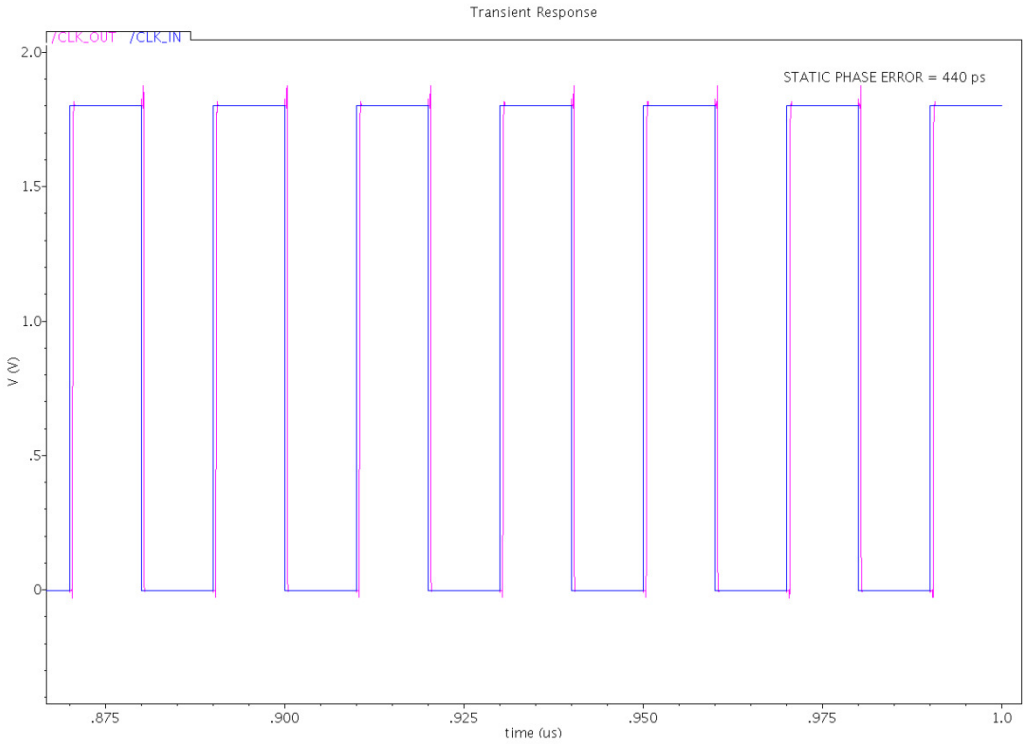


Figure 6.5 DLL phase alignment in a typical process @ 50 MHz.

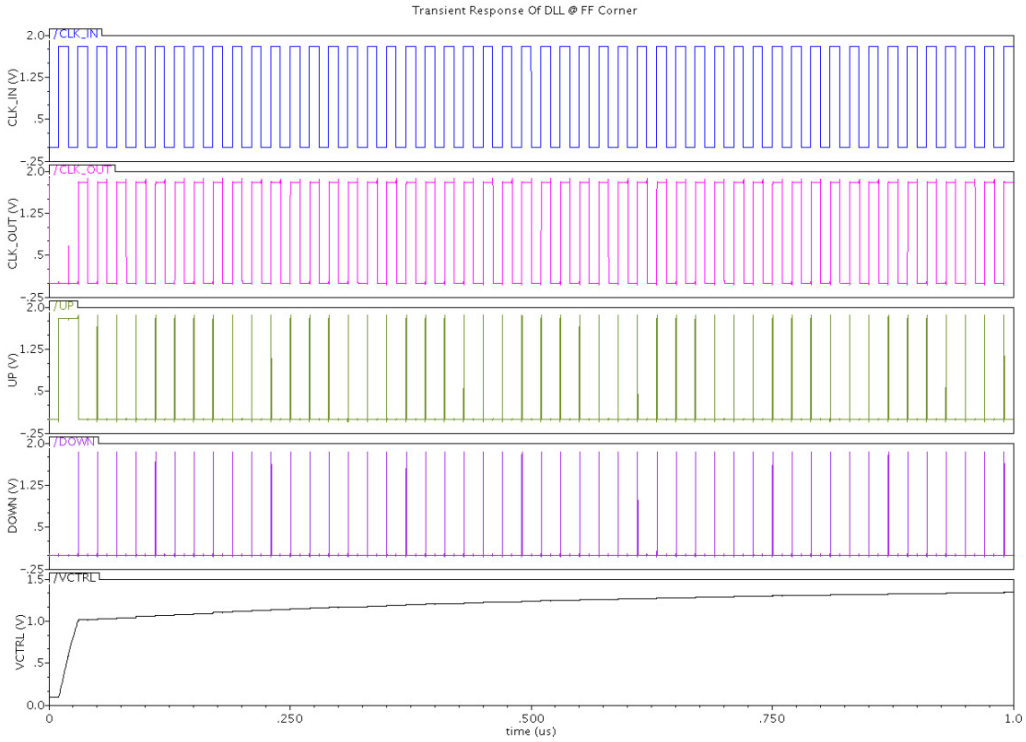


Figure 6.6 DLL performance in Fast corner.

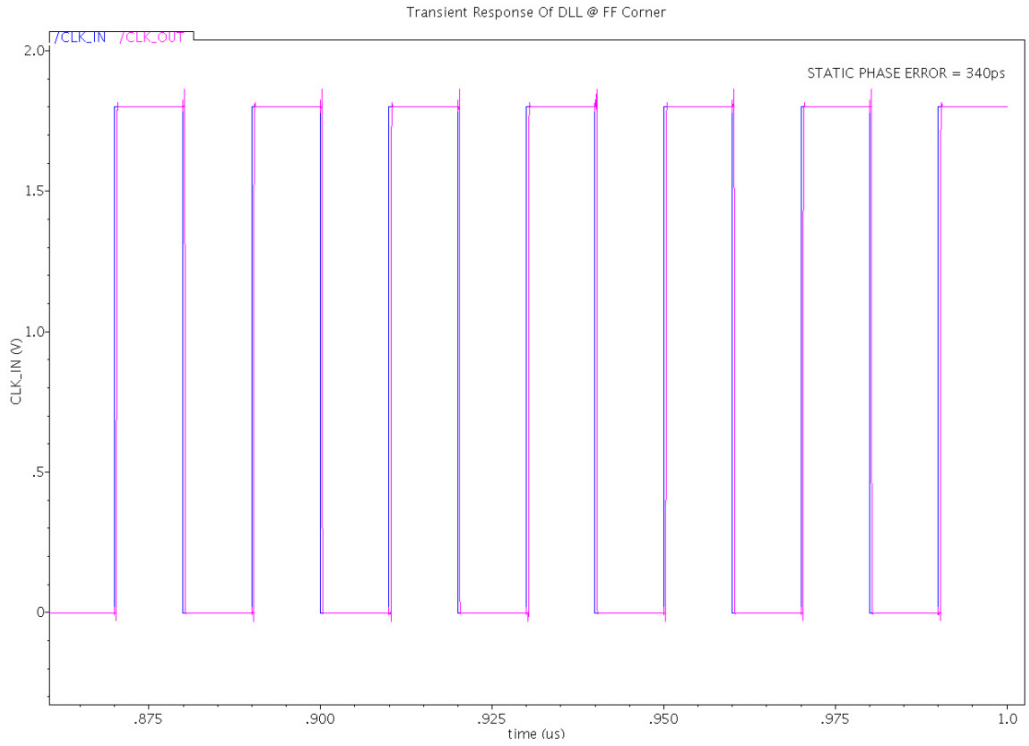


Figure 6.7 DLL phase alignment in a FF process corner @ 50 MHz.

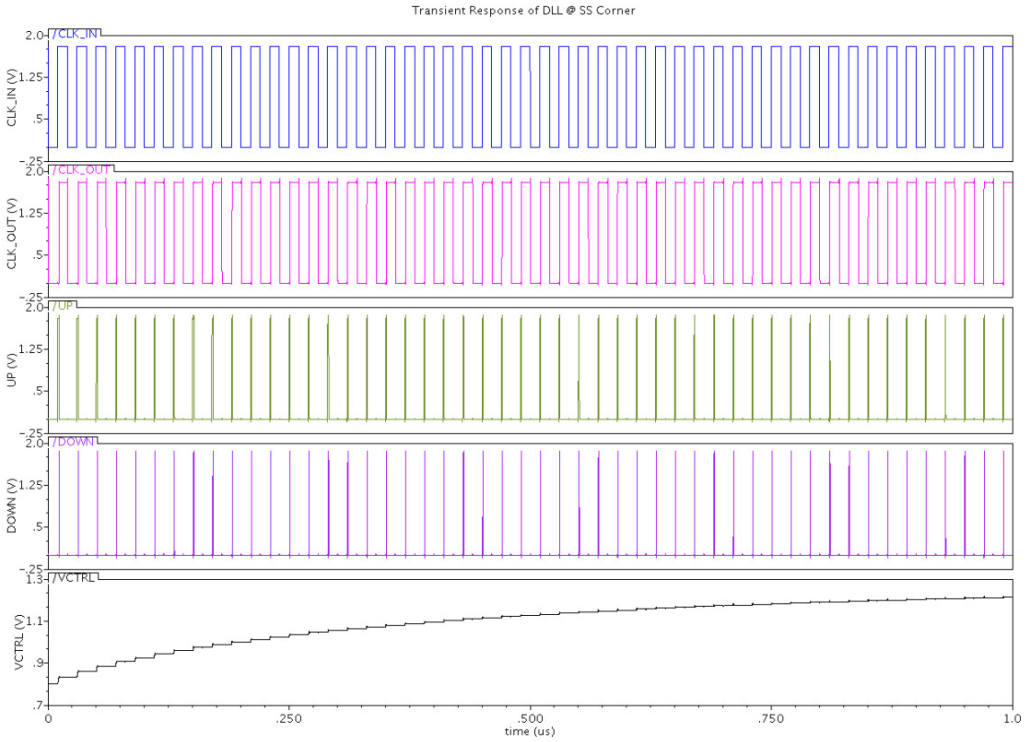


Figure 6.8 DLL performance in Slow corner.

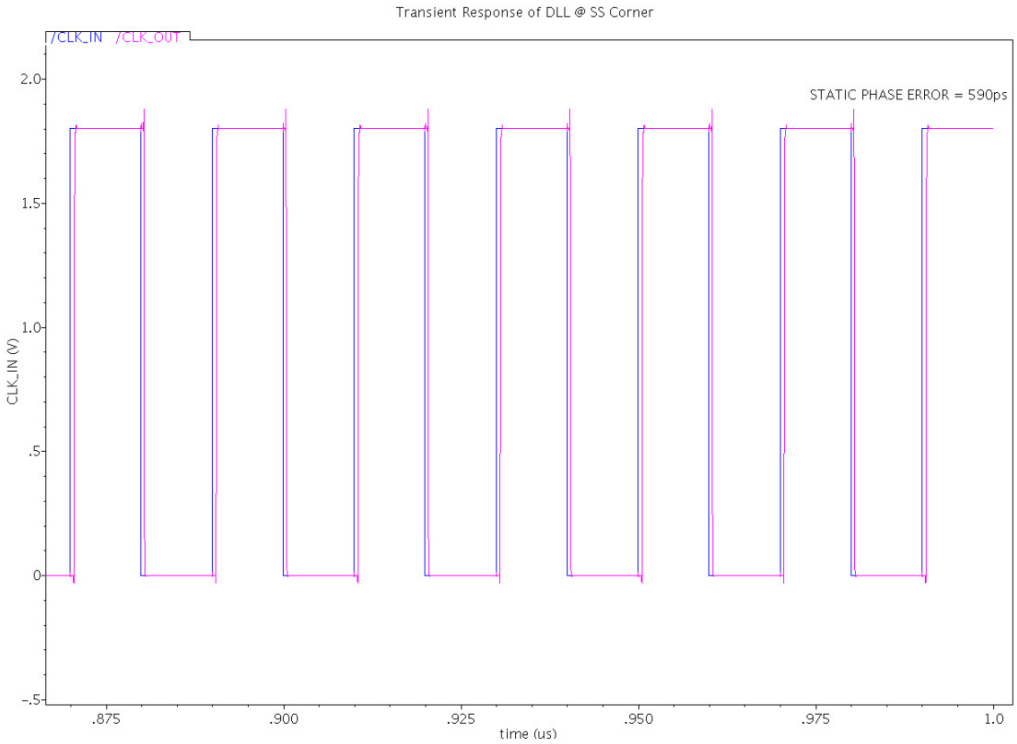


Figure 6.9 DLL phase alignment in a SS process corner @ 50 MHz.

6.3.1 TEMPERATURE VARIATIONS

Many parameters of MOS transistors vary with temperature. Generally, the following parameters are considered to be temperature-dependent: threshold voltage V_{th0}, built-in potential of S/D junctions, carrier concentration of silicon n_i, and the carrier mobility μ. The DLL was designed to operate in the standard commercial temperature range (0 °C to 70 °C). Simulations have been done at the two temperature extremities to check the functionality of the design in this temperature range as shown in Figures 6.10 and 6.11.

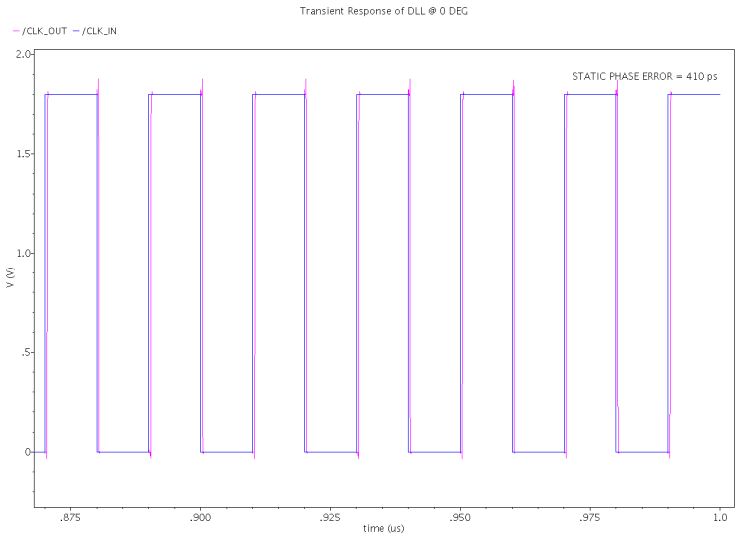


Figure 6.10 DLL phase alignment in a typical process and 0 ºC @ 50 MHz.

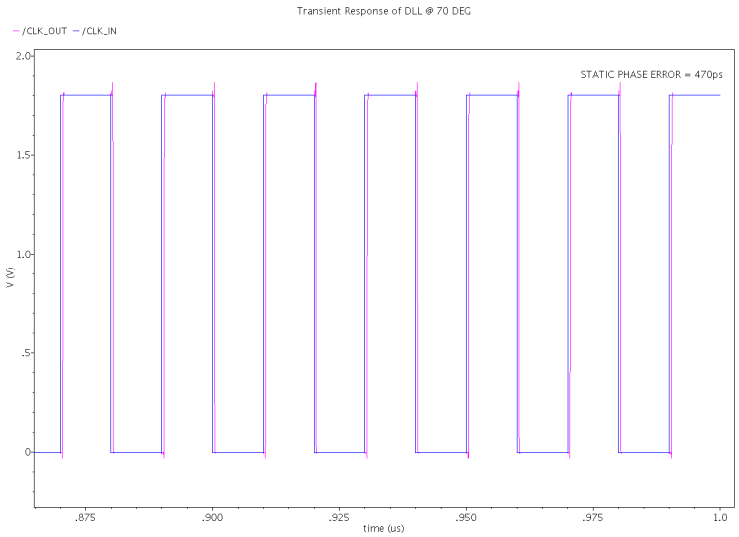


Figure 6.11 DLL phase alignment in a typical process and 70 ºC @ 50 MHz.

6.3.2 POWER SUPPLY VARIATIONS

Power supply affects the performance of the Delay Locked Loop. The performance is analyzed at ±10% of the V_{DD}, so the outputs has to be taken at the V_{DD} = 1.62V and V_{DD}=1.98V. The static phase error is 510 ps when the power supply is V_{DD} = 1.62V and is 380 ps when the power supply is V_{DD} = 1.98V as shown in Figures 6.12 and 6.13 respectively

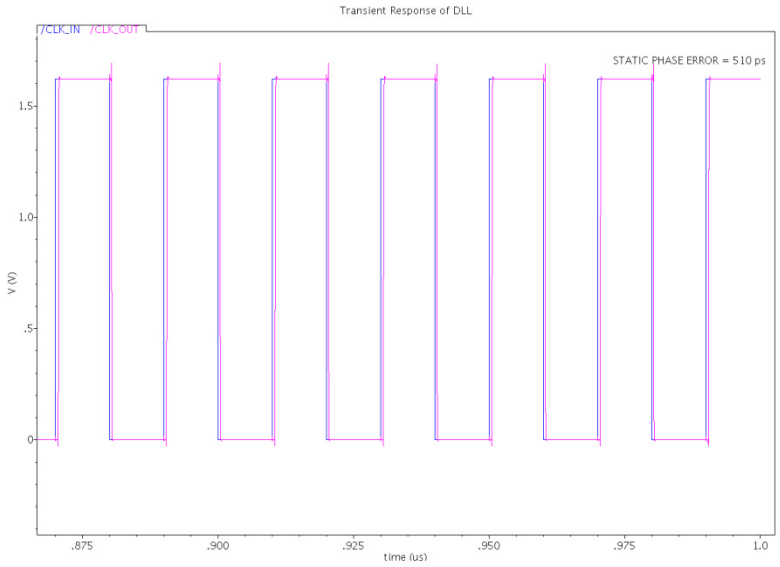


Figure 6.12 DLL phase alignment in a typical process and V_{DD} = 1.62V @ 50 MHz.

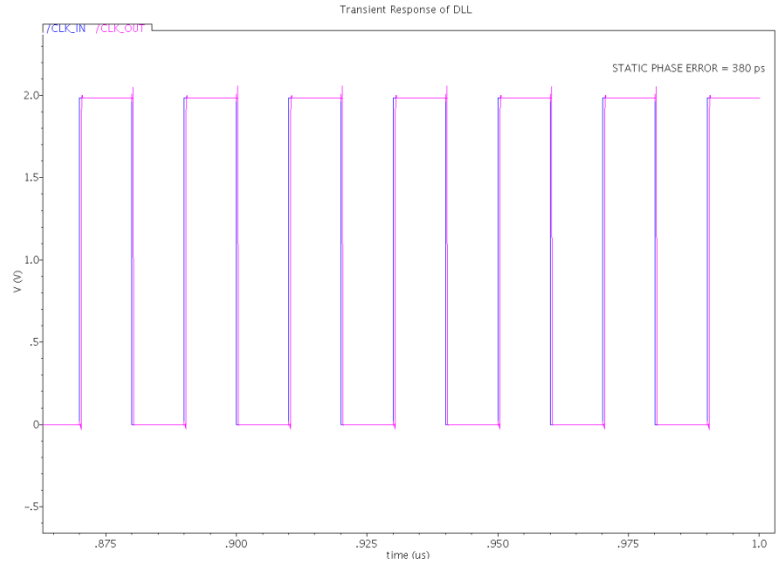


Figure 6.13 DLL phase alignment in a typical process and V_{DD} = 1.98V @ 50 MHz.

6.4 PERFORMANCE PARAMETERS OF DLL

The DLL performance basically depends on some of the parameters evaluation and these parameters are explained below in brief. Process, temperature, and power supply all affect the performance of the DLL. In the following sections, their influence on static phase error, lock range, and lock time are discussed in detail.

6.4.1 STATIC PHASE ERROR

Static phase error refers to the phase difference between the output signal of the last stage of the VCDL and the input reference signal. In the ideal case, after a DLL is locked, the phases of these two signals should be perfectly matched. However, due to the limited phase resolution of the PD and CP, some static phase error may exist. Table 6.1 shows the static phase errors of the DLL.

TABLE 6.1

STATIC PHASE ERRORS @ 50MHz (in *ps*)

TEMPERATURE & V_{SUPPLY}	**ROOM & 1.8V**	**0 ºC & 1.62V**	**0 ºC & 1.98V**	**70 ºC & 1.62V**	**70 ºC & 1.98V**
TT	440	511	360	574	410
FF	340	380	300	440	340
SS	590	710	460	780	430
FS	454	530	370	610	420
SF	420	570	350	550	540

6.4.2 LOCK RANGE

Lock range refers to the frequency range in which a DLL is able to achieve lock. Although it is often desired to have lock range as wide as possible, the lock range of a

particular DLL is usually designed for a specific application. The lock range of our design is shown in Table 6.2.

TABLE 6.2

LOCK RANGE (in MHz)

CORNERS	LOCK RANGE
TT	25-125
FF	35-140
SS	20-110
FS	25-120
SF	35-135

6.4.3 LOCK TIME

Locking time refers to the time interval a DLL takes to achieve a stable locking state from an initial state. Generally, locking time is related to the speed of the PD, the magnitude of the charging or discharging current in the CP. Table 6.3 lists the lock times under different process variations.

TABLE 6.3

LOCK TIME AT DIFFERENT CORNERS (in μs)

CORNERS	LOCK TIME
TT	1
FF	1.4
SS	1.4
FS	1
SF	1

6.5 PHYSICAL LAYOUT DESIGN

The layout of an integrated circuit defines the geometries that appear on the masks used in fabrication. Careful layout helps to achieve the design goals. The creation of the mask layout is one of the most important steps in the full-custom (bottom-up) design flow, where the designer describes the detailed geometry and the relative positioning of each mask layer to be used in actual fabrication, using a Layout Editor. Physical layout design is very tightly linked to overall circuit performance (area, speed and power dissipation) since the physical structure determines the transconductances of the transistors, the parasitic capacitances and resistances, and obviously, the silicon area that is used to realize a certain function.

6.5.1 THE ROLE OF LAYOUT IN THE DESIGN PROCESS

From a computer scientist's point of view, the layout process seems familiar enough. We are given a piece of *source code*, this time usually in terms of a circuit diagram, and we want to compile it to an *object code*, the physical layout of the circuit.

The layout step is the last major step in the design process before testing and fabrication; it is the step which reveals to the designer all the subtle electrical characteristics of the clean and logical digital systems.

6.5.2 TOLERANCES AND DESIGN RULES

The layout must pass a series of checks in a process known as Verification. The two most common checks in the verification process are Design Rule Checking (DRC), and Layout Versus Schematic (LVS). When all verification is complete, the data is translated into an industry standard format, typically GDSII, and sent to a semiconductor foundry. The process of sending this data to the foundry is called tapeout, due to the fact the data used to be shipped out on a magnetic tape. The foundry converts the data into another format and uses it to generate the photo masks used in a photolithographic process of semiconductor device fabrication.

6.5.3 DESIGN RULE CHECKING

Design Rule Checking of Check(s) (DRC) is the area of Electronic Design Automation that determines whether a particular chip layout satisfies a series of recommended parameters called Design Rules. Design Rule Checking is a major step during Physical Verification of the design, which also involves LVS (Layout Versus Schematic) Check, XOR Checks, ERC (Electrical Rule Check).

Design rules are a set of parameters provided by the semiconductor manufacturer that enable the designer to verify the correctness of the mask set. Design rules are specific to a particular semiconductor manufacturing process. A design rule set specifies a minimum size or spacing requirements between the layers of the same type or of different types.

6.5.4 DESIGN RULE CHECKING SOFTWARE AND RULES

The main objective of design rule checking (DRC) is to achieve a high overall yield and reliability for the design. If the design rules are violated the design may not be functional. While design rule checks do not validate that the design will operate correctly, they are constructed to verify that the structure meets the process constraints for a given design type and process technology.

DRC software usually takes a layout as input in the GDSII standard format and a list of rules specific to the semiconductor process chosen for fabrication. DIVA software is used in Cadence for all types of verification tools. From these it produces a report of design rule violations that the designer may or may not choose to correct.

Basic Design Rules are

(1) Size Rules.

(2) Separation Rules.

(3) Overlap Rules.

The most important design rules are summarized below (all distances are *minimum*):

Metal 1 to metal 1 spacing	0.24 µm

Poly width	0.18 µm
Metal 1 width	0.24 µm
Poly to poly spacing	0.24 µm
Poly to metal spacing	0.28/0.00 µm
Minimum contact size	0.24 µm *0.24 µm
Minimum metal area	0.1764 µm * µm
Minimum metal2 width	0.28 µm
Minimum contact spacing	0.26 µm
N well overlap p+ diffusion	0.43 µm
Poly extension beyond active	0.22 µm
Contact overlap to p+ diffusion	0.1 µm
Diffusion contact to poly spacing	0.15 µm
Gate poly spacing over diffusion	0.34 µm
Metal1 and metal2 overlap over via	0.08 µm
Minimum p+ implant overlap p+ diffusion	0.22 µm

6.5.5 LAYOUT VERSUS SCHEMATIC (LVS)

The Layout Versus Schematic (LVS) is the class of electronic design automation (EDA) verification software that determines whether a particular integrated circuit layout corresponds to the original schematic of circuit diagram of the design.

A successful Design rule check (DRC) ensures that the layout conforms to the rules designed / required for faultless fabrication. However, it does not guarantee if it really represents the circuit you desire to fabricate. This is where an LVS check is used. LVS checking software recognizes the drawn shapes of the layout that represent the electrical components of the circuit, as well as the connections between them. The software then compares them with the schematic or circuit diagram. In most cases the layout will not pass LVS the first time requiring the layout engineer to examine the LVS software's reports and make changes to the layout.

In the next sections layout and post-layout simulations are shown for different blocks of DLL.

6.5.6 LAYOUT DESIGN OF LOW GLITCH PHASE DETECTOR

The layout and extracted view of a Low Glitch Phase Detector are shown below in Figures 6.14 and 6.15 respectively.

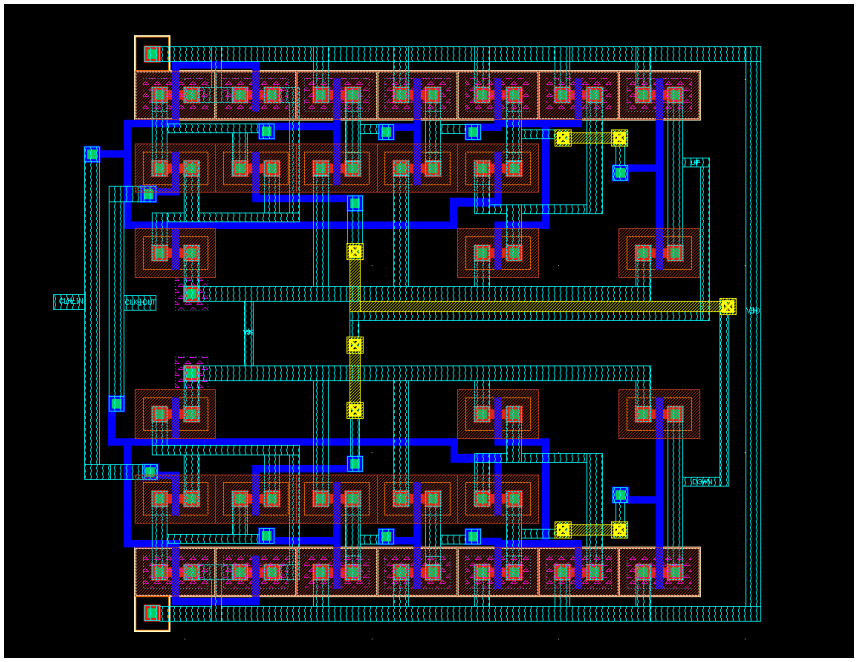


Figure 6.14 Layout of Low Glitch Phase Detector.

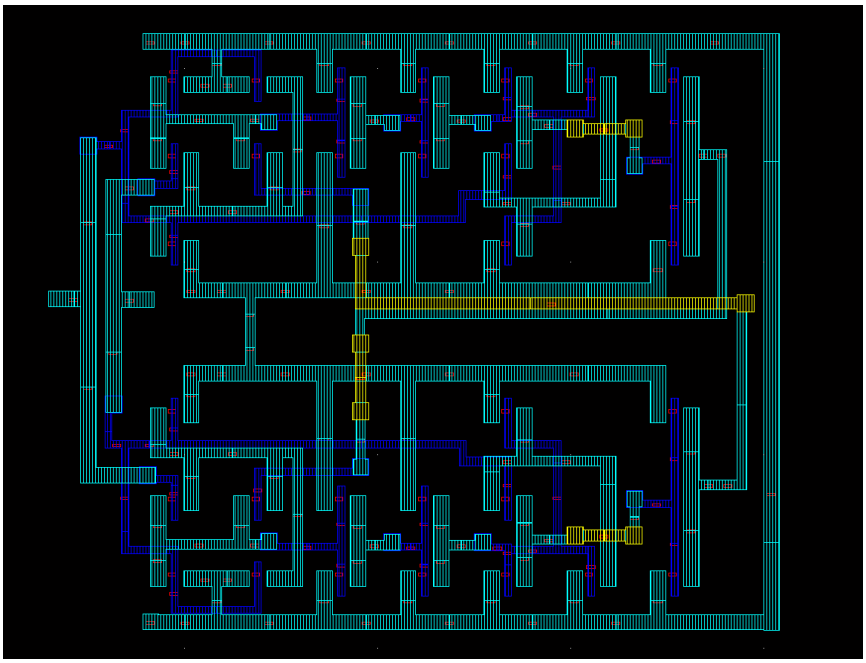


Figure 6.15 An Extracted View Of Low Glitch Phase Detector.

6.5.7 LAYOUT DESIGN OF CHARGE PUMP

The layout and extracted view of a Charge Pump are shown below in Figures 6.16 and 6.17 respectively.

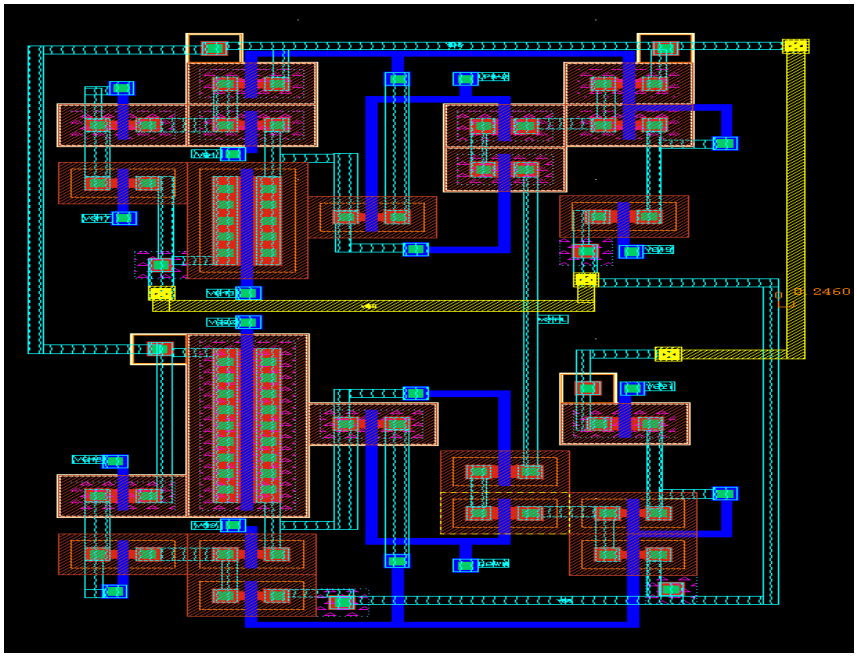


Figure 6.16 Layout Of Charge Pump.

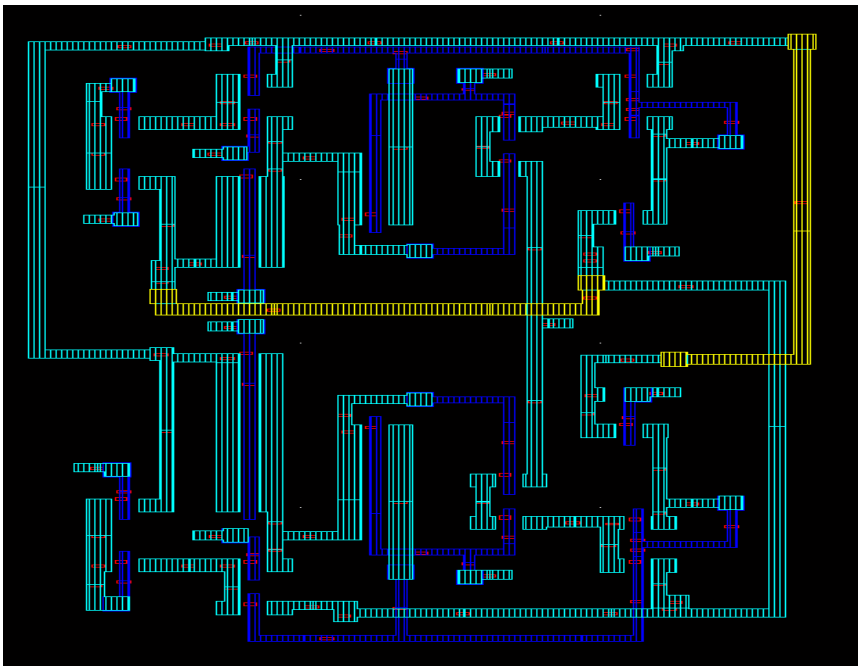


Figure 6.17 An Extracted View Of Charge Pump.

6.5.8 LAYOUT DESIGN OF LOW PASS FILTER

The layout and extracted view of a Low Pass Filter are shown below in Figures 6.18 and 6.19 respectively.

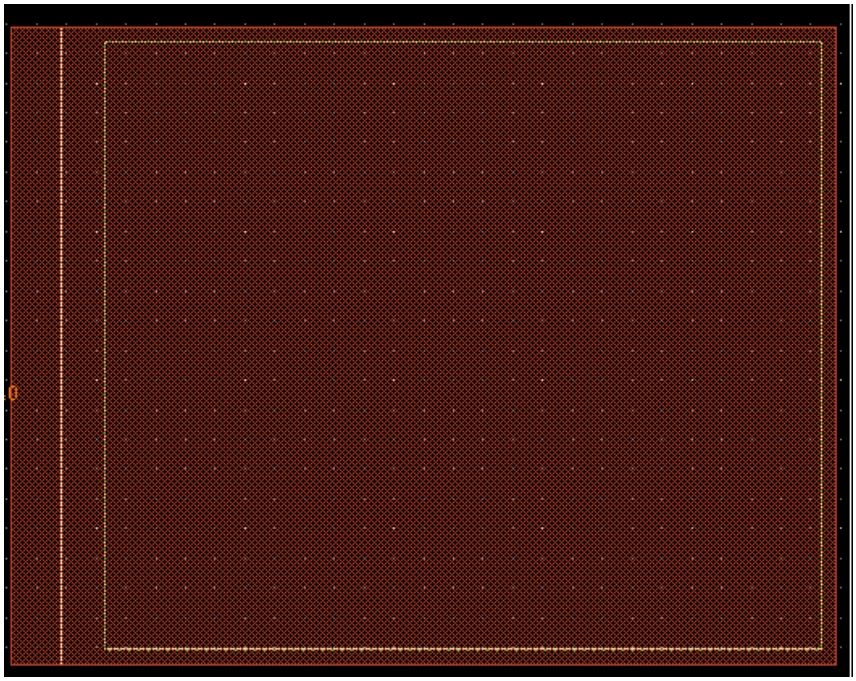


Figure 6.18 Layout Of 500 fF Capacitor.

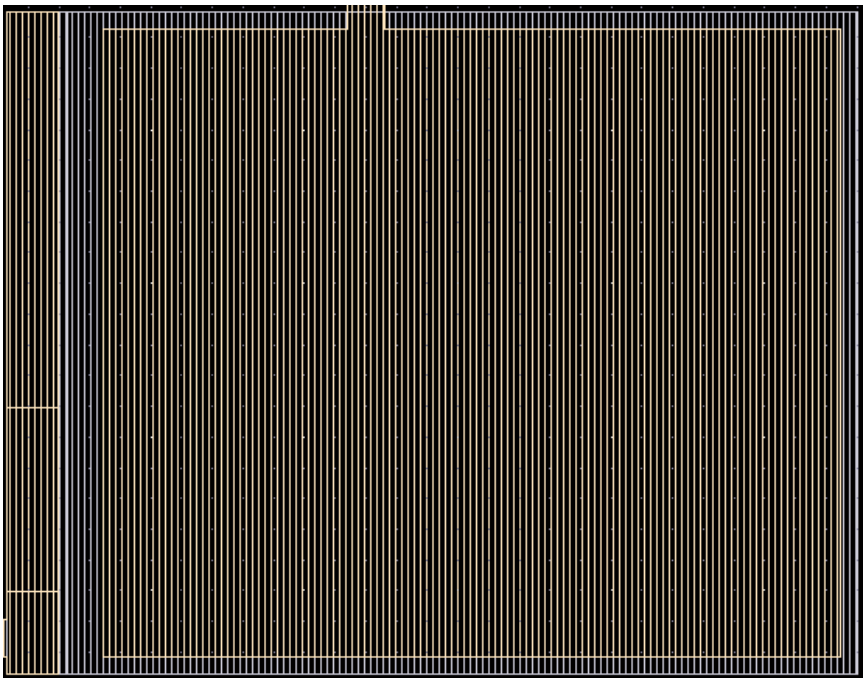


Figure 6.19 An Extracted View of 500 fF Capacitor.

6.5.9 LAYOUT DESIGN OF VOLTAGE CONTROLLED DELAY LINE

The layout and extracted view of a Voltage Controlled Delay Line are shown below in Figures 6.20 and 6.21 respectively.

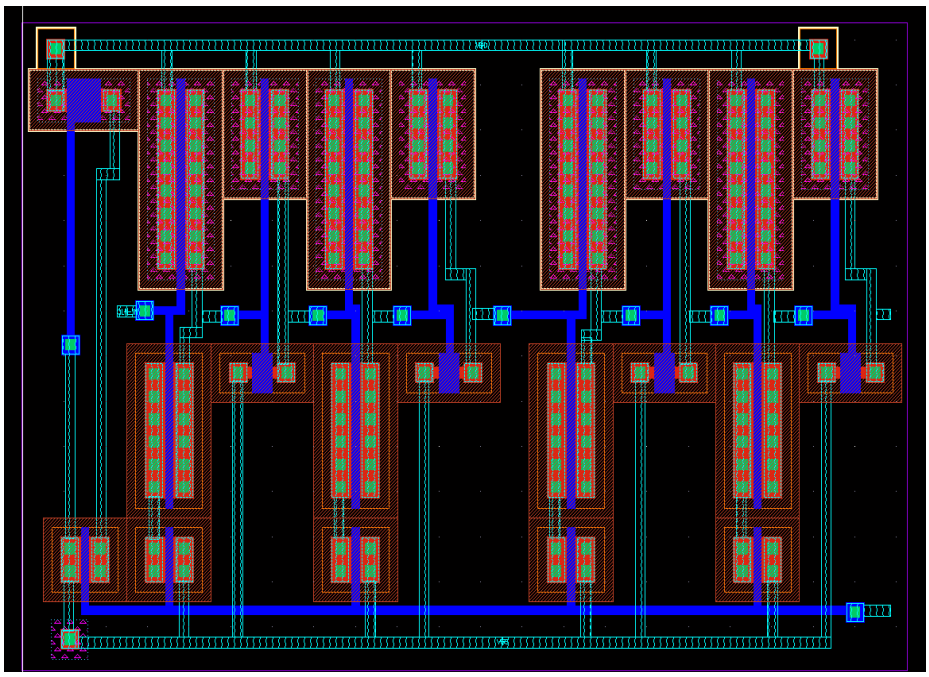


Figure 6.20 Layout Of Voltage Controlled Delay Line.

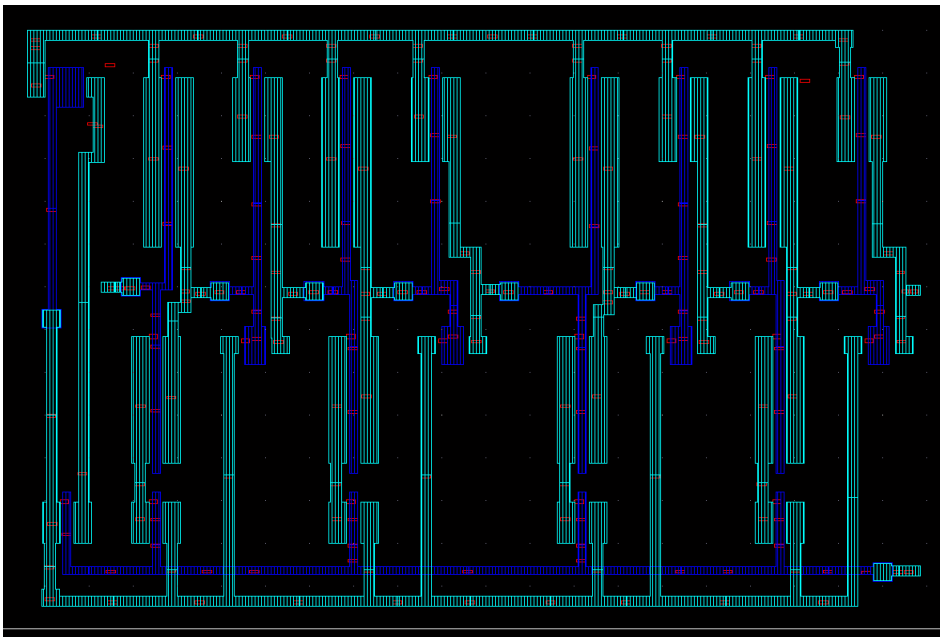


Figure 6.21 An extracted view of VCDL.

6.5.10 LAYOUT DESIGN OF DELAY-LOCKED LOOP

The layout and extracted view of a complete Delay Locked Loop are shown below in Figures 6.22 and 6.23 respectively.

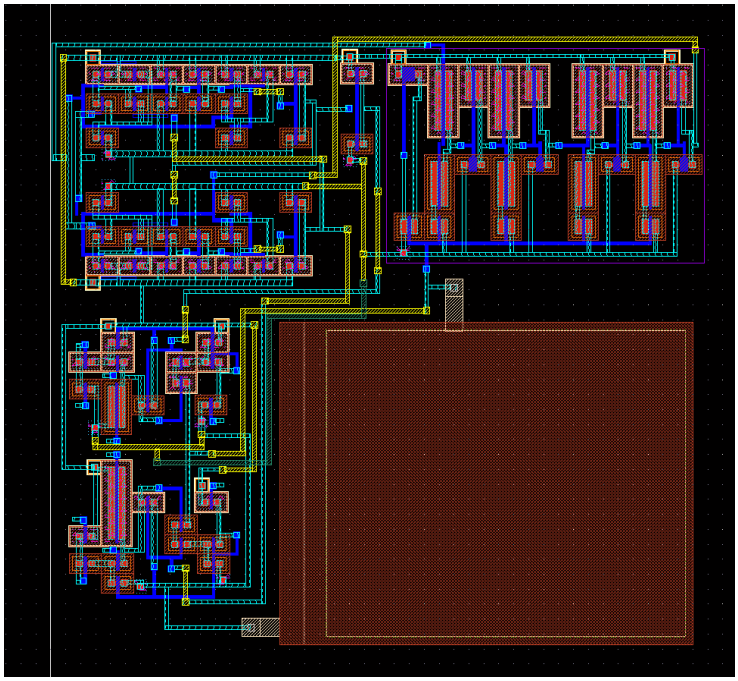


Figure 6.22 Layout of Delay Locked Loop.

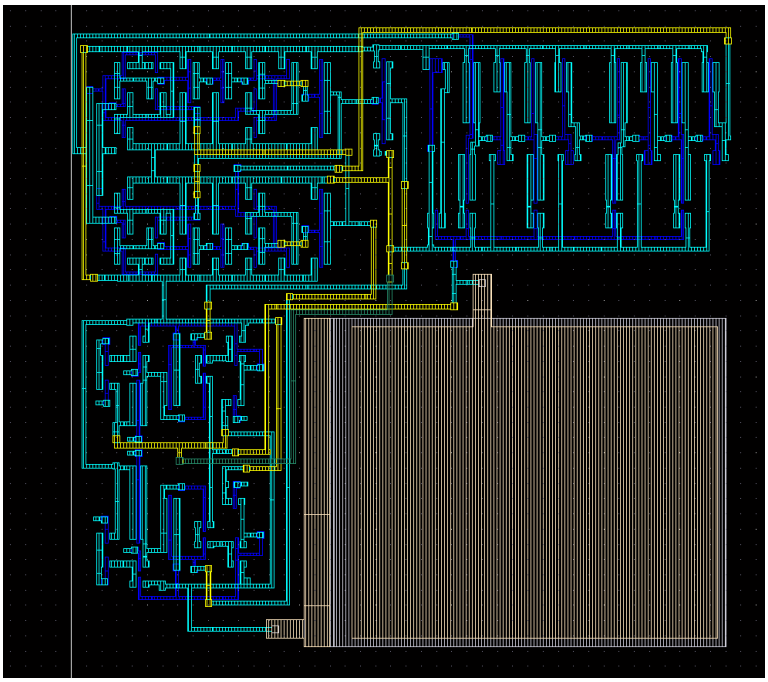


Figure 6.23 An extracted view of Delay Locked Loop

6.6 POST-LAYOUT SIMULATIONS

After performing the Design Rule Check and Layout vs. Schematic Check on each of the blocks and the entire DLL, we proceed to the RC extraction and thereby create a detailed netlist for the entire circuit. Then we need to evaluate the post-layout performance of this DLL design to verify that it satisfies the required performance criteria. The post-layout simulation results of phase detector are shown in Figure 6.24 and 6.25.

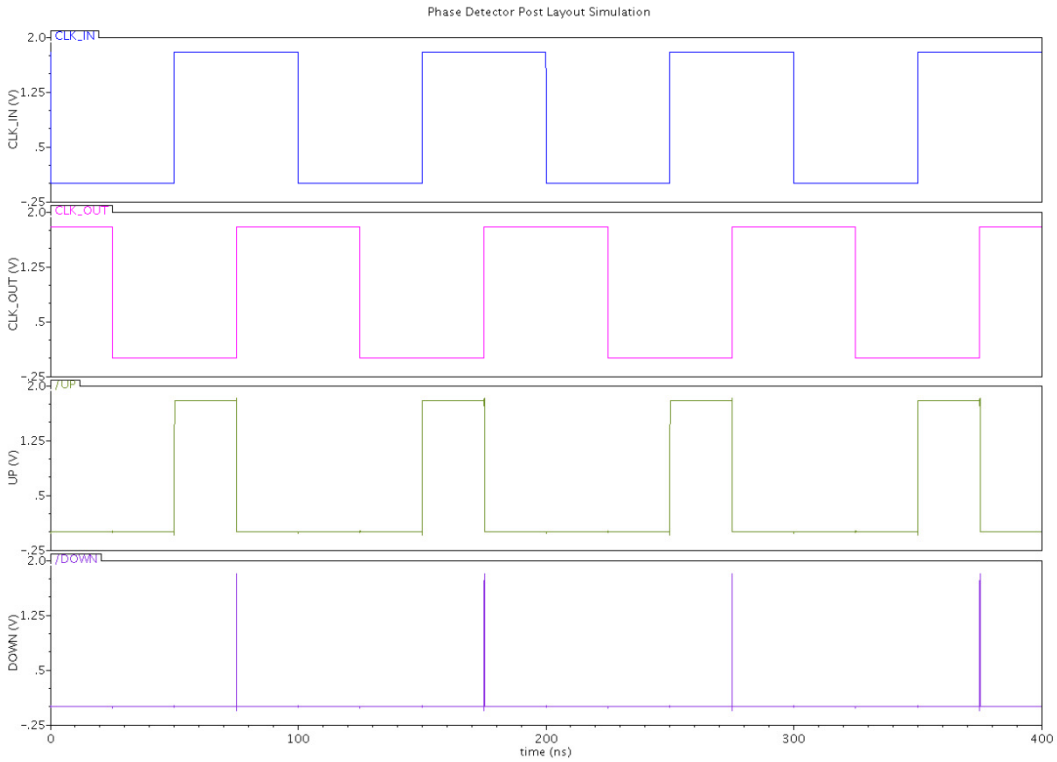


Figure 6.24 Post-Layout simulation of Phase Detector.

The inputs to the low glitch PD are the same signals that applied to the schematic circuit. The outputs of the PD with and without parasitic are quite similar. The zoomed view shows the change in duration of glitch. The glitch at the output of low glitch PD with parasitic is more as compared to PD without parasitic.

The duration of glitch in the post-layout simulation of PD is 240 ps [75.00 ns to 75.24 ns] as shown in Figure 6.25.

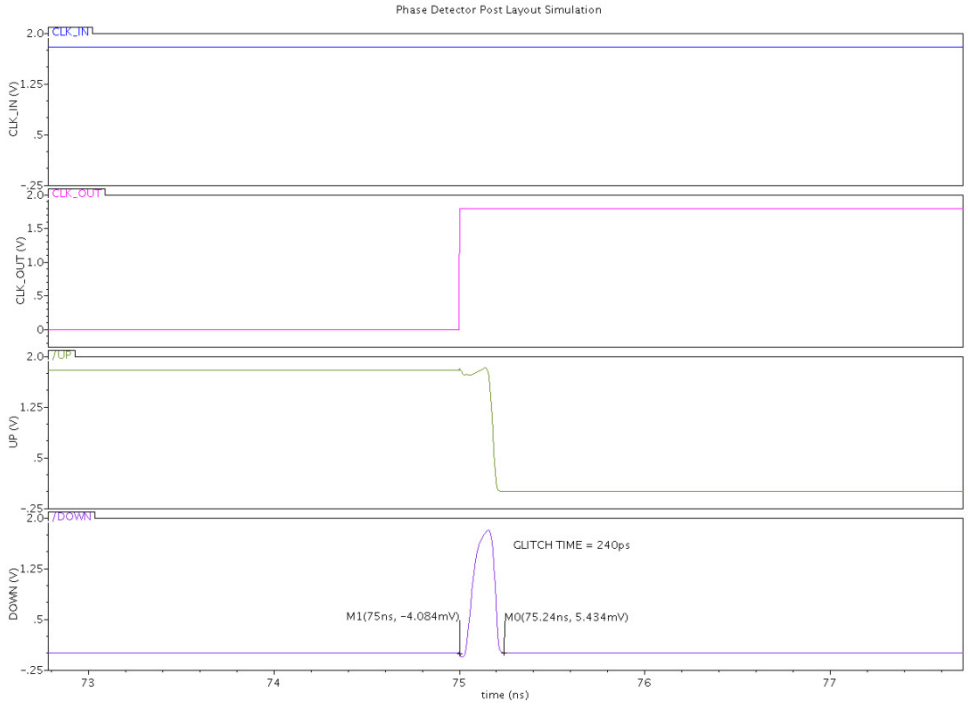


Figure 6.25 Zoomed view of PD post-layout simulation.

The post-layout simulation of the voltage controlled delay line is shown in Figure 6.26 below.

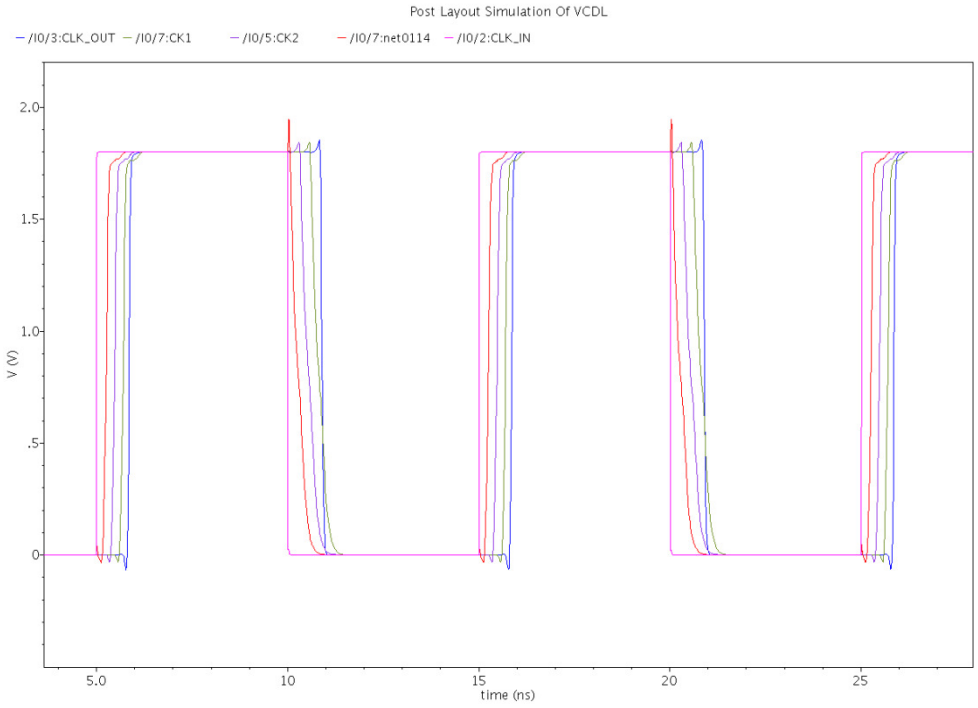


Figure 6.26 Post-Layout Simulation of the VCDL.

The post-layout simulation of entire Delay-Locked Loop is shown in Figure 6.27 below. The phase alignment in between CLK_IN and CLK_OUT after post-layout is shown in Figure 6.28.

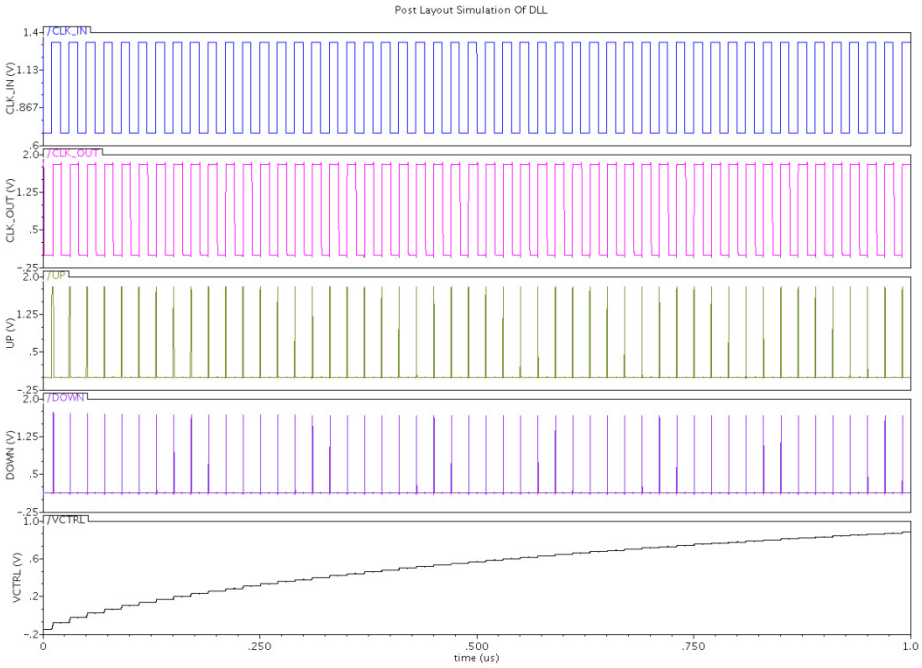


Figure 6.27 Post-Layout Simulation of the DLL.

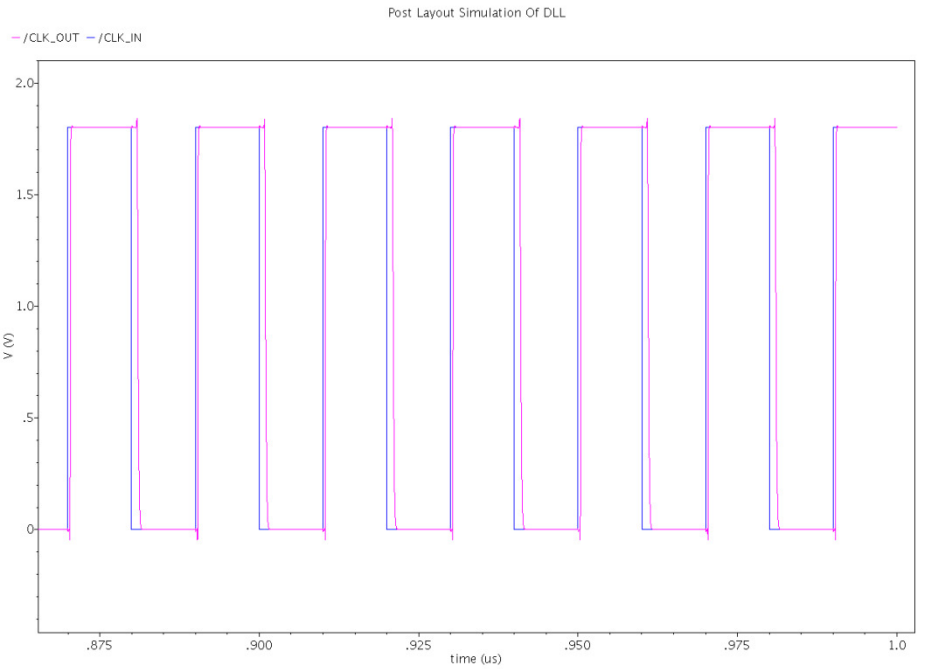


Figure 6.28 DLL phase alignment after post-layout simulation.

The following table summarizes the performance of the Delay-Locked Loop.

TABLE 6.4

DLL PERFORMANCE SUMMARY

Technology	UMC 0.18 μm
Power Supply	1.8 V
Operating Frequency	25MHz-125MHz
Charge Pump Current	20 μA
Lock-in Time	1 μs @ 50MHz
Power Dissipation	130 μW

CHAPTER 7 CONCLUSIONS AND FUTURE WORK

7.1 CONCLUSIONS

The Delay-Locked Loop remained an interesting topic for the research, as it covered many applications such as clock-synchronization, generation of multiple clock phases, clock and data recovery. Analog and digital DLL implementations were discussed in detail, paying special attention to the effects of a large delay model in the feedback path. Design considerations were discussed and supported with simulation results.

To measure the performance of the DLL, the circuit was simulated across PVT corners and over the full operating range. The DLL takes exactly $1\mu s$ to lock with a static phase error of 440 *ps* dissipating 130 μW of power.

The specific research contributions of this work include (1) proposing a low glitch phase detector in which glitch time is just 220 *ps* as compared to 590 *ps* glitch time of phase detector based on flip-flop, (2) a charge pump design with I_{CP} = 20 μA, in which the output current of proposed current mirror increases by 1.25% when the output voltage goes from 0.596 *V* to 1.76 *V* as compared to 9.38% increase in high-swing Super-Wilson current mirror.

7.2 FUTURE WORK

The subject of Delay-Locked Loop is wide and diverse. There are many aspects that can be combined in the design to achieve better performance. In many applications it is very desirable to be able to test whether a DLL has been functioning correctly. A DLL arrives at a stable or locked state after the phase of the output signal CLK_OUT is aligned with the phase of the input signal CLK_IN. Thus, a fault-free DLL would have a zero phase error between CLK_IN and CLK_OUT in the stable state, and any substantial phase error in the stable state would indicate some faults in the DLL. To fulfill this need, I would like to design a BUILT IN SELF TEST CIRCUIT (BIST) to test the functionality of a DLL.

The other aspect that was not included in this DLL design was the NOISE ANALYSIS. Since noise is an important parameter which affects the performance of a design mostly in non-linear fashion, is necessary to accurately measure the performance of the design. Including noise consideration provide more details about the sensitive point and parameters of a design. I would like to study the sources of noise in DLL design and its effect on the performance as my future work.

REFERENCES

[1] Y. SINAN. HANAY, "Delay Locked Loop Design", ECE 658 Project, December 2007.

[2] G. CHIEN AND P. R. GRAY, "A 900-MHz Local Oscillator using a DLL-based Frequency Multiplier Technique for PCS Applications," IEEE J. Solid-State Circuits, vol. 35, pp. 1996-1999, Dec. 2000.

[3] M. G. JOHNSON AND E. L. HUDSON, "A Variable Delay Line PLL for CPU-Coprocessor Synchronization," IEEE J. Solid-State Circuits, vol. 23, pp. 1218-1223, Oct. 1988.

[4] M.-J. E. LEE, *et al.*, "Jitter Transfer Characteristics of Delay-Locked Loops, Theories and Design Techniques," IEEE J. Solid-State Circuits, vol. 38, pp. 614-621, April 2003.

[5] A. SAGHAFI, "A Novel High Resolution Delay Locked Loop," Graduate Thesis, The University Of Science & Technology, Tehran, July 2005.

[6] C. JIA, "A Delay-Locked Loop for Multiple Clock Phases/Delays Generation, Doctoral Thesis, Georgia Institute of Technology, 2005.

[7] A. CHANDRAKASAN, W. J. BOWHILL, AND F. FOX, "*Design of High-Performance Microprocessor Circuit*," New York: IEEE Press, 2001.

[8] B.W. GARLEPP, K. S. DONNELLY, J. KIM, P. S. CHAU, J. L. ZERBE, C. HUANG, C. V. TRAN, C. L. PORTMANN, D. STARK, Y. F. CHAN, T. H. LEEN, AND M. A. HOROWITZ, "A Portable Digital DLL for High-Speed

CMOS Interface Circuits," IEEE J. Solid-State Circuits, vol. 34, pp. 632-644, May 1999.

[9] Y. OKAJIMA, M. TAGUCHI, M. YANAGAWA, K. NISHIMURA, AND O. HAMADA, "Digital Delay-locked Loop and Design Technique for High-Speed Synchronous Interface," IEICE Trans. Electron, vol. E79-C, pp. 798-807, June 1996.

[10] A. EFENDOVICH, et al., "Multifrequency Zero-Jitter Delay-Locked Loop", IEEE J. Solid-State Circuits, vol. 29, pp. 67-70, Jan. 1994.

[11] H. -H. CHANG, C. -H. SUN, AND S. -I. LIU, "Low Jitter Butterworth Delay-Locked Loops," Int. Symp. on VLSI Circuits Dig. Tech. Papers, pp. 177-180, June 2003.

[12] S. SIDIROPOULOS, M. A. HOROWITZ, "A Semidigital Dual Delay-Locked Loop," IEEE J. Solid-State Circuits, vol. 32, pp. 1683-1692, November 1997.

[13] Y.-J. JUNG, S. -W. LEE, D. SHIM, W. KIM, C. KIM, S. -I. CHO, "A Dual-Loop Delay-Locked Loop Using Multiple Voltage-Controlled Delay Lines," IEEE J. Solid-State Circuits, vol. 36, pp. 784-791, May 2001.

[14] H. -H. CHANG, J. -W. LIN, AND S. -I. LIU, "A Fast Locking and Low Jitter Delay-Locked Loop Using DHDL," IEEE J. Solid-State Circuits, vol. 38, pp. 343-346, Feb.2003.

[15] R. L. AGUIAR, D. M. SANTOS, "Simulation and Modeling of Digital Delay Locked Loops," ISCAS 1999, 42nd Midwest Symposium on, vol. 2, pp. 843-846, August 1999.

[16] B. RAZAVI, "*Design of Analog CMOS Integrated Circuits,"* McGraw-Hill, New York, NY, 2001.

[17] M. MANSURI, D. LIU AND C. K. K. YANG, "Fast Frequency Acquisition Phase-Frequency Detectors for GSamples/s Phase-Locked Loops," IEEE J. Solid-State Circuits, vol. 37, pp. 1331-1334, Oct. 2002.

[18] ADEL S. SEDRA AND KENNETH C. SMITH, "*Microelectronic Circuits.* Fourth Edition," New York: Oxford University Press, 1997.

[19] L. F. TANGUAY, M. SAWAN, Y. SAVARIA, "A Very-High Output Impedance Charge Pump for Low-Voltage Low-Power PLLs," ELSEVIER Microelectronics Journal, pp. 1026-1031, April 2009.

[20] M. G. JOHNSON AND M. E. HUDSON, "A Variable Delay Line PLL for CPU Coprocessor Synchronization," IEEE J. Solid-State Circuits, vol. 23, pp. 1218-1223, Oct. 1988.

[21] M. MAYMANDI-NEJAD AND M. SACHDEV, "A Digitally Programmed Delay Element: Design & Analysis," IEEE J. Solid-State Circuits, vol. 11, Oct. 2003.

APPENDIX A

LVS REPORTS

LVS REPORT OF PHASE DETECTOR

****

PHASE_DETECTOR schematic shirish <vs> PHASE_DETECTOR layout shirish

****

Filter/Reduce statistics only. Network matching was OK.

Pre-expand Statistics

==========	Original	
Cell/Device	schematic	layout
(P_18_MM) MOS	14	14
(N_18_MM) MOS	16	16
	------	------
Total	30	30

Filter Statistics

==========	Original		Filtered	
Cell/Device	schematic	layout	schematic	layout
(N_18_MM) MOS	16	16	16	16
(P_18_MM) MOS	14	14	14	14

Reduce Statistics

==========	Filtered		Reduced	
Cell/Device	schematic	layout	schematic	layout
(N_18_MM) MOS	16	16	12	12
(P_18_MM) MOS	14	14	14	14
(N_18_MM:ParMos2#1) MosBlk	-	-	2	2
	------	------	------	------
Total	30	30	28	28

Schematic and Layout Match

LVS REPORT OF CHARGE PUMP

****

CHARGE_PUMP_FINAL schematic shirish <vs> CHARGE_PUMP_FINAL layout shirish

****

Filter/Reduce statistics only. Network matching was OK.

Pre-expand Statistics

==========	Original	
Cell/Device	schematic	layout
(P_18_MM) MOS	11	11
(N_18_MM) MOS	11	11
	------	------
Total	22	22

Filter Statistics

==========	Original		Filtered	
Cell/Device	schematic	layout	schematic	layout
(N_18_MM) MOS	11	11	11	11
(P_18_MM) MOS	11	11	11	11

Reduce Statistics

==========	Filtered		Reduced	
Cell/Device	schematic	layout	schematic	layout
(N_18_MM) MOS	11	11	9	9
(P_18_MM) MOS	11	11	9	9
(N_18_MM:SerMos2#1) MosBlk	-	-	1	1
(P_18_MM:SerMos2#1) MosBlk	-	-	1	1
	------	------	------	------
Total	22	22	20	20

Schematic and Layout Match

LVS REPORT OF VOLTAGE CONTROLLED DELAY LINE

****

VCDL_DLL_FINAL schematic shirish <vs> VCDL_DLL_FINAL layout shirish

****

Filter/Reduce statistics only. Network matching was OK.

Pre-expand Statistics

==========	Original	
Cell/Device	schematic	layout
(N_18_MM) MOS	13	13
(P_18_MM) MOS	9	9
	------	------
Total	22	22

Filter Statistics

==========	Original		Filtered	
Cell/Device	schematic	layout	schematic	layout
(N_18_MM) MOS	13	13	13	13
(P_18_MM) MOS	9	9	9	9

Reduce Statistics

==========	Filtered		Reduced	
Cell/Device	schematic	layout	schematic	layout
(N_18_MM) MOS	13	13	5	5
(P_18_MM) MOS	9	9	9	9
(N_18_MM:SerMos2#1) MosBlk	-	-	4	4
	------	------	------	------
Total	22	22	18	18

Schematic and Layout Match

LVS REPORT OF ENTIRE DELAY-LOCKED LOOP DESIGN

****

DELAY_LOCKED_LOOP_COMPACT schematic shirish
<vs>
DELAY_LOCKED_LOOP_COMPACT layout shirish

****

Filter/Reduce statistics only. Network matching was OK.

Pre-expand Statistics

==========	Original	
Cell/Device	schematic	layout
(PDDLL schematic shirish, PDDLL) Cell	1	1
(CPDLL schematic shirish, CPDLL) Cell	1	1
(VCDLDLL schematic shirish, VCDLDLL) Cell	1	1
(Inverter schematic shirish, Inve...) Cell	1	1
(MIMCAPS_MM) CAP	1	1
	------	------
Total	5	5

Filter Statistics

==========	Original		Filtered	
Cell/Device	schematic	layout	schematic	layout
(CPDLL) Cell	1	1	1	1
(Inverter) Cell	1	1	1	1
(PDDLL) Cell	1	1	1	1
(VCDLDLL) Cell	1	1	1	1
(MIMCAPS_MM) CAP	1	1	1	1

Reduce Statistics

==========	Filtered		Reduced	
Cell/Device	schematic	layout	schematic	layout
(CPDLL) Cell	1	1	1	1
(Inverter) Cell	1	1	1	1
(PDDLL) Cell	1	1	1	1
(VCDLDLL) Cell	1	1	1	1
(MIMCAPS_MM) CAP	1	1	1	1
	---	---	---	---
Total	5	5	5	5

Schematic and Layout Match

LVS COMPARE SUMMARY OF ENTIRE DLL DESIGN

Schematic	Layout	Status
PDDLL schematic shirish	PDDLL layout shirish	matched
CPDLL schematic shirish	CPDLL layout shirish	matched
VCDLDLL schematic shirish	VCDLDLL layout shirish	matched
Inverter schematic shirish	Inverter layout shirish	matched
DELAY_LOCKED_LOOP_COMPACT schematic shirish		
DELAY_LOCKED_LOOP_COMPACT layout shirish		matched

Schematic and Layout Match.

APPENDIX B

SPICE BSIM3v3 VERSION 3.2 MOS MODEL PARAMETERS

DIODES & NMOSFET SPICE TYPICAL PARAMETERS

```
model bsim_diode_area diode
cj0=0.00103*(1+dcj_n_18_rf )        vj=0.813        m=0.443
is=1e-06                            n=1

model bsim_diode_perim diode
cj0=1.34e-10*(1+dcjsw_n_18_rf )     vj=0.88         m=0.33
is=7e-11                            n=1

model bsim_diode_swg diode
cj0=5e-10 *(1+ dcjgate_n_18_rf)     vj=0.88         m=0.33
is=7e-11            n=1

model bsim_mos_transistor_mos bsim3v3 type=n
version=3.2                         binunit=1       mobmod=1
capmod=2                            nqsmod=0        noimod=2
tox=4.2e-09 + dtox_n_18_rf          toxm=4.2e-09    xj=1.6e-07
nch=3.745e+17                       rsh=8           ngate=1e+23
vth0=0.3075 + dvth0_n_18_rf         k1=0.4578       k2=-0.02638
k3=-10.88                           k3b=0.2379      w0=-8.813e-08
nlx=4.279e-07                       dvt0=0.4042     dvt1=0.3237
dvt2=-0.8602                        dvt0w=0.383     dvt1w=6e+05
dvt2w=-0.025                        lint=1.587e-08  wint=1.022e-08
dwg=-3.396e-09                      dwb=1.346e-09   u0=332.1
du0_n_18_rf
ua=-1.17e-09                        ub=2.407e-18    uc=4.355e-11
vsat=8.1e+04                        a0=1.93         ags=0.5072
b0=1.486e-06                        b1=9.064e-06    keta=0.01752
a1=0                                a2=1            voff=-0.1208
nfactor=1.038                       cit=-0.001511   cdsc=0.002175
cdscd=0                             cdscb=0.0008241 eta0=0.005504
```

etab=-0.001459 dsub=0.001592 pclm=0.741
pdiblc1=0.005061 pdiblc2=0.006001 pdiblcb=0
drout=0.001592 pscbe1=4.866e+08 pscbe2=3e-08
pvag=-0.2958 rdsw=9.905 prwg=1.1
prwb=0 wr=p_wr alpha0=0
alpha1=0 beta0=30 xpart=1
cgso=1.55e-10 *(1+dcgso_n_18_rf) cgdo=1.55e-10 *(1+dcgdo_n_18_rf)
cgbo=0
cgsl=3e-11*(1+dcgsl_n_18_rf) cgdl=3e-11 *(1+dcgdl_n_18_rf)
ckappa=0.6
cf=2.33e-11 *(1+dcf_n_18_rf) clc=1e-07 cle=0.6
dlc=4e-08 dwc=0 vfbcv=-1
noff=1 voffcv=0 acde=1
moin=15 noia=1.31826e+19 noib=1.44544e+05
noic=-1.24516e-12 em=4.1e+07 af=1
ef=0.92 kf=0 lmin=1.8e-07
lmax=1.805e-07 wmin=5e-06 wmax=1.05e-04
xl=-1.05e-08 + dxl_n_18_rf xw=0 + dxw_n_18_rf js=1e-06
jsw=7e-11 cj=0.00103 *(1+dcj_n_18_rf) mj=0.443
pb=0.813 cjsw=1.34e-10 *(1+dcjsw_n_18_rf) mjsw=0.33
tnom=25 ute=-1.286 kt1=-0.2255
kt1l=-4.175e-09 kt2=-0.02527 ua1=2.153e-09
ub1=-2.673e-18 uc1=-3.832e-11 at=1.449e+04
prt=-46.18 xti=3 wl=0
wln=1 ww=7.262e-16 wwn=1
wwl=0 ll=-1.062e-15 lln=1
lw=2.996e-15 lwn=1 lwl=0
llc=-6.64e-15 lwc=0 lwlc=0
wlc=0 wwc=0 wwlc=0
lvth0=-0.0001 + dlvth0_n_18_rf wvth0=0.06027 + dwvth0_n_18_rf pvth0=0
dpvth0_n_18_rf
lnlx=-2.854e-08 wnlx=0 pnlx=0
lnfactor=0.032 wua=-1.88e-11 wu0=0.54
pub=3.8e-20 pw0=1.3e-09 lua=1.5e-11
lub=9.76e-20 wrdsw=0 weta0=0
wetab=0 leta0=0.001574 letab=0
peta0=0 petab=0 wpclm=0
wvoff=-0.0004078 lvoff=-0.004208 pvoff=-0.0003788
wa0=-0.04731 la0=-0.4667 pa0=-0.02649
wags=0.004242 lags=0.3028 pags=0
wketa=0 lketa=-0.01942 pketa=0
wute=0.06373 lute=0 pute=0
wvsat=5066 lvsat=0 pvsat=0
dpvsat_n_18_rf
lpdiblc2=-0.004752 wat=7067 wprt=0
ldif=8e-08 hdif=2.6e-07 n=1
pbsw=0.88 cjswg=5e-10 *(1+dcjgate_n_18_rf) ctp=0.000914
ptp=0.000924 cta=0.000919 pta=0.00158
elm=5 tlevc=1

DIODES & PMOSFET SPICE TYPICAL PARAMETERS

```
model bsim_diode_area diode
cj0=0.00114 *(1+ dcj_p_18_rf)          vj=0.762          m=0.395
is=3e-06                               n=1

model bsim_diode_perim diode
cj0=1.74e-10 *(1+ dcjsw_p_18_rf)       vj=0.665          m=0.324
is=4.12e-11                  n=1

model bsim_diode_swg diode
cj0=4.2e-10 *(1+ dcjgate_p_18_rf)      vj=0.665          m=0.324
is=4.12e-11                  n=1

model bsim_mos_transistor_mos bsim3v3 type=p
mobmod=3                               version=3.2       capmod=2
binunit=1                              nqsmod=0          noimod=2
tox=4.2e-09 + dtox_p_18_rf             toxm=4.2e-09      xj=1e-07
nch=6.131e+17                          ngate=1e+23       vth0=-0.4325
dvth0_p_18_rf
k1=0.5704                              k2=0.006973       k3=-2.833
k3b=1.326                              w0=-1.943e-07     nlx=2.56e-07
dvt0=0.4885                            dvt1=0.09578      dvt2=0.1287
dvt0w=-0.1261                          dvt1w=2.479e+04   dvt2w=0.6915
lint=-1.041e-08                        wint=-1.525e-07   dwg=-1.151e-07
dwb=-1.039e-07                         u0=90+du0_p_18_rf ua=1.49e-09
ub=4.646e-19                           uc=-0.09587       vsat=4.75e+04
a0=1.35                                ags=0.3818        b0=-3.088e-07
b1=0                                   keta=0.01044      a1=0
a2=1                                   voff=-0.1073      nfactor=0.984
cit=-0.001067                          cdsc=0.0007578    cdscd=0
cdscb=0.0001                           eta0=1.071        etab=-0.9291
dsub=1.919                             pclm=0.553        pdiblc1=0.007
pdiblc2=0.008005                       pdiblcb=0         drout=0.157
pscbe1=4.866e+08                       pscbe2=2.8e-07    pvag=-0.888
rdsw=202.1                             prwg=1.2          prwb=0
wr=p_wr                                alpha0=0          alpha1=0
beta0=30          cgdo=1.254e-10 *(1+dcgdo_p_18_rf )     cgbo=0
cgso=1.254e-10 *(1+dcgso_p_18_rf )     xpart=0           cf=1.533e-10*(1+dcf_p_18_rf )
dlc=7.01e-08      cgsl=2e-11*(1+dcgsl_p_18_rf )          cgdl=2e-11*(1+dcgdl_p_18_rf )
ckappa=0.6                             clc=1e-07         cle=0.6
dwc=2.3e-07                            vfbcv=-1          noff=1
voffcv=0                               acde=1            moin=15
noia=3.57456993317604e+18              noib=2.5e+03      noic=2.6126e-11
em=4.1e+07                             af=1              ef=1.1388
kf=0                                   lmin=1.8e-07      lmax=1.805e-07
wmin=5e-06                             wmax=1.05e-04     xl=-2e-09
dxl_p_18_rf
xw=0  + dxw_p_18_rf                    js=3e-06          jsw=4.12e-11
```

cj=0.00114*(1+dcj_p_18_rf)
cjsw=1.74e-10 *(1+dcjsw_p_18_rf)
ute=-0.4484
kt2=-0.009487
uc1=-0.0985
xti=3
ll=6.635e-15
wwn=1
lwn=1
lwc=0
wwc=0
dlvth0_p_18_rf
wvth0=-0.0148 + dwvth0_p_18_rf
pvth0=0.0031+ dpvth0_p_18_rf
weta0=0
wua=2.7e-09
wub=0
wuc=0
wvoff=-0.009816
wa0=-0.04807
wags=-0.04177
wketa=0
wute=-0.2682
wvsat=-1.42e+04
dpvsat_p_18_rf
lpdiblc2=0.003012
n=1
ctp=0.000753
ldif=8e-08
rsc=0
rs=0

mj=0.395
mjsw=0.324
kt1=-0.2194
ua1=4.571e-09
at=1.203e+04
ww=1.236e-14
wl=0
wwl=0
lwl=0
lwlc=0
wwlc=0

lu0=-3
lnlx=-1.584e-08
wetab=0
lua=-2.37e-10
lub=0
luc=0
lvoff=-0.0009871
la0=-0.281
lags=0.04454
lketa=-0.012
lute=0
lvsat=0

wat=-6405
pbsw=0.665
pta=0.00155
rsh=8
rdc=0

pb=0.762
tnom=25
kt1l=-8.204e-09
ub1=-6.026e-18
prt=0
lw=-2.873e-16
wln=1
lln=1
llc=-1.31e-14
wlc=0
lvth0=0.0057

lnfactor=0.03
wrdsw=10.07
wpclm=0
pua=5.855e-11
pub=0
puc=0
pvoff=-9.833e-05
pa0=0.08661
pags=-0.04076
pketa=0
pute=0
pvsat=-350

wprt=216.6
cta=0.001
ptp=0.00124
rd=0
hdif=2.6e-07