

# High-Speed and Multi-Bitrate Clock and Data Recovery System Based on Half-Rate Clocking

Sung-Hwan (David) Hong



Department of Electrical & Computer Engineering  
McGill University  
Montreal (Québec)  
September 2005

A thesis submitted to the Faculty of Graduate Studies and  
Research in partial fulfillment of the requirements of the  
degree of Masters of Engineering.

© Sung-Hwan (David) Hong, 2005



Library and  
Archives Canada

Bibliothèque et  
Archives Canada

Published Heritage  
Branch

Direction du  
Patrimoine de l'édition

395 Wellington Street  
Ottawa ON K1A 0N4  
Canada

395, rue Wellington  
Ottawa ON K1A 0N4  
Canada

*Your file    Votre référence*

*ISBN: 978-0-494-22648-3*

*Our file    Notre référence*

*ISBN: 978-0-494-22648-3*

#### NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

#### AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

---

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

  
**Canada**

---

# Acknowledgments

---

I would like to acknowledge the Canadian Microsystems Corporation (CMC), the Regroupement Stratégique en Microélectronique du Québec (ReSMiQ), Micronet, and McGill University for providing financial support as well as the necessary resources for my research project. I would like to thank my research supervisor, Dr. Mourad N. El-Gamal, for the great opportunity and experience. His guidance, support, and advice will not be forgotten.

I would like to thank the following people for giving me inspiration and directly contributing to this work: Alex Marsolais, Francis Beaudoin, Rola Baki, and Maher Assaad. I would like to thank my numerous friends – which I will not attempt to name lest I forget one – for being there with me during my successful and desperate moments. They have been my emotional support as well as my reasoning in difficult times; my life would have not been the same without them. Thank you sincerely.

Last but not least, I would like to sincerely thank my family who are always there for me in good and bad times. They will always be my dependable source of support and love. They will always be my source of joy. I would not have been half the man I am without them.

This achievement was a wonderful personal journey. I have learned many lessons; above all, I have grown as a person. I thank my God for that.

---

# **Abstract**

# **Résumé**

---

---

## R É S U M É

---

Les infrastructures de communication d'aujourd'hui ainsi que les produits électroniques de consommation dépendent de plus en plus sur la communication de données en série. Cette forme de communication en transport de données est grandement usitée; des méthodes de conceptions de systèmes plus rapides et rentables sont alors recherchées ([1]-[8]). Cette étude se concentre sur l'écoulement complet de conception de system intégré de rétablissement d'horloge et de données (Clock and data recovery – CDR) en technologie de CMOS, répondant au besoin de solutions à grande vitesse et rentables en circuits intégrés.

Cette thèse fait un rapport sur l'écoulement de conception ascendante du system CDR. Le défi principal s'agit d'exploiter une technologie accessible et courante, tel que la technologie CMOS 0.18-micron, pour mettre en application un CDR qui fonctionne à 6-Gbps pour applications en communication de fil. La structure de conception traditionnelle en CDR s'agit de l'exécution en « full rate ». Malheureusement, cette méthode d'exécution s'avère irréalisable avec cette technologie; ce problème est donc allégé en exploitant le concept et les architectures « half rate ».

D'abord, les blocs de base ont été conçus en utilisant la technologie CMOS 0.18-micron de la compagnie TSMC. Ces blocs de circuits ont été alors assemblés afin d'engendrer de plus grands blocs tels qu'une bascule électronique déclenchée par front d'impulsion, un une bascule électronique par double front d'impulsion, un détecteur de phase, un détecteur de phase et de fréquence, etc. Toute composante a été caractérisée et vérifiée. Ainsi, le système complet et assemblé a été simulé, vérifié, et envoyé pour la fabrication. En conclusion, le système était testé et vérifié pour ensuite faire l'analyse de résultats.

Des images de diagrammes d'oeil ont été extraites à partir du circuit prototype CDR. Les résultats montrent une ouverture d'oeil prononcée, ce qui signifie que les données sont correctement récupérées. Le système CDR est muni de vitesse de récupération ajustable et il a été examiné avec succès. Une méthode de haut en bas de vérification a été alors exécutée afin de comparer les résultats examinés et de simulation.

La contribution principale de cette recherche s'avère le rapport bien documenté et détaillé de l'écoulement de conception d'un système de CDR en utilisant une méthode d'exécution ascendante avec la vérification de haut en bas. Des méthodes de vérifications ont été conçues et ensuite appliquées afin d'étudier quelques anomalies dans les résultats.

---

## A B S T R A C T

---

Today's telecommunications infrastructures and consumer electronics rely largely on serial communications. Serial communications is a major form of transmitting information; therefore, faster and more cost effective methods of designing receivers are becoming interesting research topics ([1]-[8]). This work focuses on the full design flow of an integrated clock and data recovery (CDR) system in CMOS technology, responding to this need of high-speed and cost effective circuit solutions.

This thesis reports a full bottom-up design flow of a CDR system. The main challenge is to use an affordable and mainstream technology, such as the CMOS 0.18-micron technology, to implement a 6-Gbps serial receiver for land line applications. Unfortunately, the traditional full rate implementation is impossible with this CMOS technology; therefore, this problem is alleviated by exploring half-rate architectures.

The basic building blocks were first designed using TSMC's CMOS 0.18-micron technology. These gates were then assembled to create larger circuit blocks such as an edge-triggered flip flop, a double edge-triggered flip flop, a phase detectors, a phase/frequency detector, etc. Each component was characterized and then verified. The completely assembled system was simulated, verified, and then sent for fabrication. Finally, the system was tested and verified for analysis.

Eye diagram patterns were extracted from the CDR prototype circuit. Results show an opening in the eye, which suggests that the data pattern are properly recovered. The system was also successfully tested for lower bit rates. A top-down verification method was then executed in order to compare the experimental and simulation results.

The main contribution of this work is the full detailed documentation of the design flow of a CDR system using a bottom-up implementation with a top-down verification methodology. Verification methods were designed and applied in order to investigate some discrepancies in the results.



<b>CHAPTER I</b>	<b>INTRODUCTION</b>	<b>1</b>
1.1	Motivation.....	2
1.1.1	Global Village.....	2
1.1.2	Serial Communications as a Solution .....	2
1.2	Contributions & Summary of this Thesis .....	3
<b>CHAPTER II</b>	<b>INTRODUCTORY CONCEPTS FOR CLOCK AND DATA RECOVERY CIRCUITS</b>	<b>5</b>
2.1	The Role of a Clock and Data Recovery Circuit .....	6
2.2	Properties of Random Binary Data .....	7
2.2.1	Non-Return-to-Zero Random Data & Spectrum .....	7
2.2.2	Generating a Pseudo-Random Binary Sequence .....	8
2.3	Effect of Noise on Random Data .....	11
2.3.1	Relation Between the Signal-to-Noise Ratio (SNR) & the Bit-Error Rate (BER) .....	11
2.4	Phase Noise & Jitter.....	13
2.4.1	Phase Noise.....	13
2.4.2	Jitter .....	14
2.4.3	Mathematical Relationship Between Phase Noise and Jitter.....	16
2.4.4	The Effect of Additive Noise.....	16
2.5	Jitter in CDR Circuits .....	17
2.5.1	Jitter Transfer.....	18
2.5.2	Jitter Generation.....	19
2.5.3	Jitter Tolerance .....	21
2.6	Summary .....	23
<b>CHAPTER III</b>	<b>HIGH-SPEED CURRENT MODE LOGIC CIRCUIT BUILDING BLOCKS</b>	<b>24</b>
3.1	Introduction to Current-Mode Logic .....	25

3.1.1	Concept of Current Switching .....	25
3.1.2	Advantages of the CML Style .....	25
3.1.3	Performance Comparison .....	27
3.2	CML Speed Optimization for A CMOS 0.18-micron Technology .....	28
3.2.1	Design Parameters and Performance Requirements .....	28
3.2.2	Slew Rate & Frequency Response Relationship .....	29
3.2.3	Eye-Diagram and its Relationship to Speed .....	31
3.2.4	CML Design Challenges & Design Flow .....	33
3.3	Logic CML Circuit Blocks .....	35
3.3.1	Inverter or Small Buffer .....	35
3.3.2	Two-Port Multiplexer .....	35
3.3.3	D-Latch or Level-Sensitive Latch .....	38
3.3.4	Further Design of Complex Circuit Blocks .....	39
3.4	Summary .....	40

---

<b>CHAPTER IV</b>	<b>BUILDING BLOCKS OF A PLL BASED CLOCK AND DATA RECOVERY CIRCUIT</b>	<b>42</b>
-------------------	---	-----------

---

4.1	Introduction .....	43
4.2	Basics of Phase & Frequency Detection for Random Data .....	43
4.2.1	Basics of Phase & Frequency Detection .....	43
4.2.2	Basic Operating Principal of the Hogge Phase Detector .....	45
4.2.3	Basic Principals of the Early-Late Alexander Phase Detector .....	45
4.2.4	Implementation of a Half-Rate Phase Detector – Data Transition Tracking Loop .....	48
4.2.5	Frequency Detector .....	51
4.2.6	The DTTL Phase & Frequency Detector .....	53
4.3	Charge Pump .....	55
4.4	Current-Starved Ring Oscillator .....	56
4.5	Four-Phase Frequency Divider .....	60
4.6	Decision Circuit for BER Optimization .....	63
4.6.1	Eye Diagram .....	63

4.6.2	BER Optimization.....	63
4.6.3	Decision Circuit Bank.....	65
4.7	Output Drivers .....	65
4.8	Summary.....	68

---

<b>CHAPTER V</b>	<b>SYSTEM VERIFICATION &amp; TESTING</b>	<b>69</b>
------------------	--	-----------

---

5.1	Implementation & Description .....	70
5.1.1	Prototype Design.....	70
5.1.2	Layout Considerations .....	71
5.1.2.1	Minimizing Trace Distances.....	71
5.1.2.2	Supply & Ground Wiring .....	72
5.1.2.3	Input & Output Impedance Matching.....	74
5.1.3	Prototype PCB for Testing.....	74
5.2	Testing setup .....	76
5.2.1	Test Equipment.....	76
5.2.2	Probing Station .....	77
5.3	Verification & Results .....	77
5.3.1	Test Bench for Simulations.....	77
5.3.2	Simulation Results of the Half-Rate Phase/Frequency Detector .....	78
5.3.3	Current-Starved Ring Oscillator .....	78
5.3.3.1	Frequency Transfer Characteristic & Gain .....	78
5.3.3.2	Phase Noise and Jitter .....	83
5.3.4	Closed-Loop Simulations and Measurements .....	85
5.3.4.1	System Setup for Simulation .....	85
5.3.4.2	Modeling the Ring Oscillator to Fit Measurement Data.....	86
5.3.4.3	Mixed-Mode Simulated Jitter Generation .....	88
5.3.4.4	Measurement of the Recovered Data.....	93
5.3.5	Analysis of Results .....	95
5.3.5.1	BERT & Eye-Diagram.....	96
5.4	Summary.....	97

<b>CHAPTER VI</b>	<b>SUMMARY &amp; CONCLUSION</b>	<b>100</b>
6.1	Modeling & Simulation .....	101
6.1.1	Bottom-Up Methodology of Design .....	101
6.1.2	Summary of Results and Issues .....	102
6.2	Testing .....	102
6.2.1	On-Chip Probing Versus Off-Chip Ports.....	102
6.2.2	Demultiplexing and Multiplexing Input/Output .....	103
6.3	Suggested MODifications for Future Design .....	103
6.3.1	Low-Risk Design Approach .....	103
6.3.2	Layout Techniques.....	104
6.3.3	Alternate VCO Design.....	105
6.3.4	Modification to the 4-Phase Clock Divider .....	105
6.3.5	Use of Foundry Technology .....	105
6.4	Summary.....	107
<b>VII</b>	<b>APPENDIX</b>	<b>108</b>
7.1	Abbreviations.....	109
7.2	Jitter & Eye Diagram Measurements.....	110
7.2.1	Microsoft Excel.....	110
7.2.2	MATLAB Eye Diagram .....	111
7.3	PRBS Generation in MATLAB.....	112
7.4	Verilog-A codes for the Ring Oscillator Components.....	113
7.4.1	Ideal Continuous-Time (Analog) Delay Block.....	113
7.4.2	Ideal Single-Phase Voltage Controlled Oscillator (VCO).....	115
7.5	Ring Oscillator Supplementary Results.....	115
<b>VIII</b>	<b>REFERENCES</b>	<b>118</b>

---

# Chapter 1

Introduction

---

## 1 MOTIVATION

### 1.1 Global Village

In the 1970s, Marshall McLuhan envisioned that our world would become a “global village” with the advent of communications technologies. The world-wide proliferation of the access to communications is definitely evidence to this “global village” concept. Figure 1-1 shows the growth of internet users and the percentage of population who have access to the internet [9]. Today, the discovery of a low-loss medium, the fibre optic, has congregated all forms of communications (TV, radio, telephone, internet, etc.) to one platform, the optical carrier network. Today, optical networks become the framework of endless telecommunication possibilities.

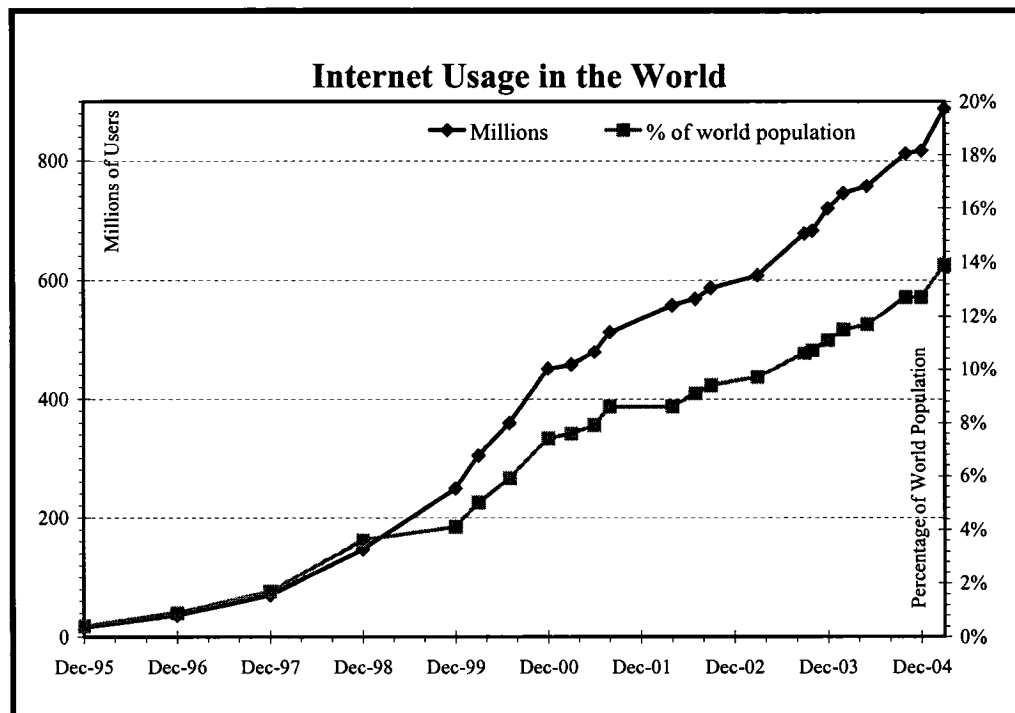


Figure 1-1 The growth chart of internet usage in the world.

### 1.2 Serial Communications as a Solution

Throughout history, there has been a shift from serial to parallel communications. Due to the rapid growth of the internet, serial land line communications infrastructures are installed all over the world, in order to meet the growth rate and the bandwidth growth of

the internet [10]. Recently, even consumer electronics are moving towards serial communications, such as the USB (Universal Serial BUS), SATA (Serial ATA), and serial computer memory. Serial communications has become the solution to higher and more efficient data transmission in order to meet the demands and trends of the information technology.

- A serial connection occupies less physical space. The gained space can be used to isolate it better from its surroundings, e.g. fibre optics, well isolated microstrips on a PCB, coaxial cables, etc.
- The presence of multiple conductors in parallel and in close proximity implies more crosstalk at higher frequencies. With serial connections, the cross-talk can be minimized.
- Given a good design of the receiver, the issue of clock skew in the medium is relaxed.
- The cost of the transmission medium is significantly reduced.

All the advantages above imply that the clock speeds can be increased, and therefore possibly condense more data bandwidth on a single connection, rather than using multiple connections in parallel. With the advent of affordable high-speed technologies such as CMOS and SiGe, it is now possible to integrate serial communications in everyday products. All serial-data devices need a serial receiver, such as clock and data recovery (CDR) circuits, whether it is for land line communications (optical, coaxial, LAN, etc.) or for consumer products (USB, serial memory, SATA drivers, etc.).

---

## **2      CONTRIBUTIONS & SUMMARY OF THIS THESIS**

---

One of the main contributions of this research is the full documentation of the bottom-up approach in designing a system as complex and as complete as a CDR. Concepts of CDRs and PLLs, as well as testing methods are discussed and documented.

Chapter II presents the introductory concepts of CDRs, and their theory of operation based on the type II PLL architecture. Since the CDR designed here is a PLL-based sys-

tem, it is important to understand some of the similarities and differences when it comes to theory. As well, some performance metrics of CDRs are described for proper characterization.

Chapter III is the main contribution of this thesis. This chapter addresses the design methods of CML gates for high-speed switching. It also compares it briefly to other logic styles, such as the CMOS logic styles. An extra novelty has been added to CML circuits to eliminate feedthrough of unwanted signals. This is the major contribution in terms of novelty.

Chapter IV presents some popular designs of phase/frequency detector (PFD) circuits. The main focus of this chapter is on characterizing the half-rate binary PFD presented by Savoj & Razavi [11], and on verifying the performance and function of this circuitry. The complete block-by-block system is designed in this section, and each block is described and documented in details.

Chapter V starts by describing the layout and implementation of the testing board used to characterize the system built here. The testing procedures, verifications, as well as results are reported in this chapter, in order to demonstrate the validity of the modeling through measurements. Furthermore, this chapter addresses some challenges in setting up mixed-mode simulation with Verilog-A, in order to reduce the simulation time with a reasonable compromise in the accuracy of the results. This section completes the bottom-up implementation with a top-down verification design flow:

- D. Hong and M. N. El-Gamal, "A Concise Design and Verification Workflow for Evaluating the Proper Operation of a Clock and Data Recovery Systems." accepted for presentation at the 48th IEEE Int'l Midwest Symposium on Circuits & Systems, Cincinnati, Ohio, Aug. 7 - 10, 2005.

Chapter VI concludes the research with discussions and possible improvements for future research work in the same area.



---

# Chapter 2

Introductory Concepts for Clock and  
Data Recovery Circuits

---

## 1 THE ROLE OF A CLOCK AND DATA RECOVERY CIRCUIT

Clock and Data Recover (CDR) circuits are pertinent to serial and asynchronous communication. Figure 2-1 is a simple illustration of the main role of a CDR system. The input is fed serially into the data input port of the CDR. In practical systems, this signal is usually corrupted by noise and hence can manifest itself in the form of unclean and jittery input data. The main role of the CDR system is to do the following:

- Reproduce the input serial binary data with a cleaner signal, i.e. without jitter and noise. This function is carried on by the data recovery unit of the CDR.
- Recover the clock that corresponds to the input data bit rate. This clock is often needed for circuits placed after the CDR, such as for demultiplexing the data. This role is carried on by the clock recovery unit of the CDR.

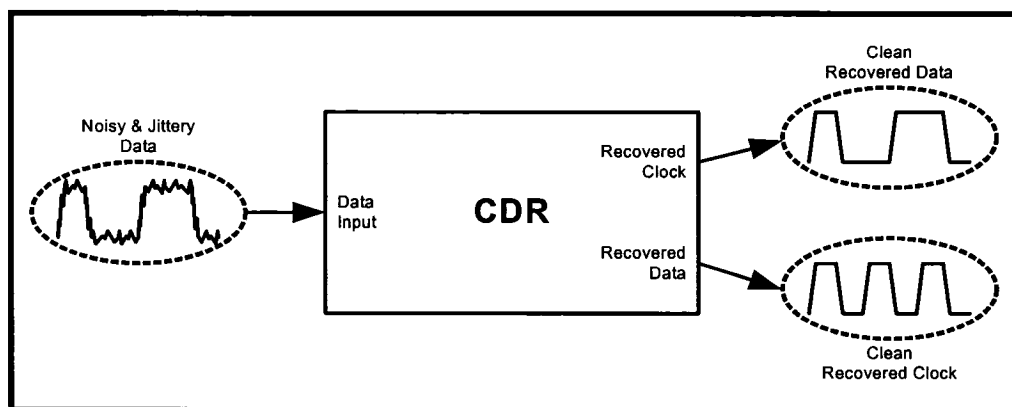


Figure 2-1 Illustration of the role of a clock and data recovery (CDR) system.

In long-haul optical communication systems, the role of the CDR is quite apparent in optical repeaters. Because land line media are not ideal, the signal quality degrades over long distances and is corrupted by noise and jitter. If the data is transported over very long distances through this realistic medium, there comes a point where the noise and jitter become too severe for the receiver to properly read the data signal. Therefore, these repeater nodes simply regenerate and clean up the data using a CDR. Basically, repeater nodes are placed between nodes that are far apart in order to ensure minimum error in the transport of the data.

Another application of the CDR is apparent in the context of demultiplexing serial

lines. Serial communication lines often transport multiplexed data. Let us suppose that 16 channels of 625Mbps data are multiplexed and sent through a 10-Gbps ( $16 \times 625\text{Mbps}$ ) high-speed or long-haul line. In order to recover the information in each channel at the receiving end, the receiver must be able to demultiplex the high-speed serial line into the corresponding 16 channels. However, in order to properly demultiplex the incoming data, a clock signal corresponding to the incoming data signal is needed. This clock can be obtained by a CDR circuit. In fact, this clock can be used for any synchronous data applications as well.

## **2 PROPERTIES OF RANDOM BINARY DATA**

---

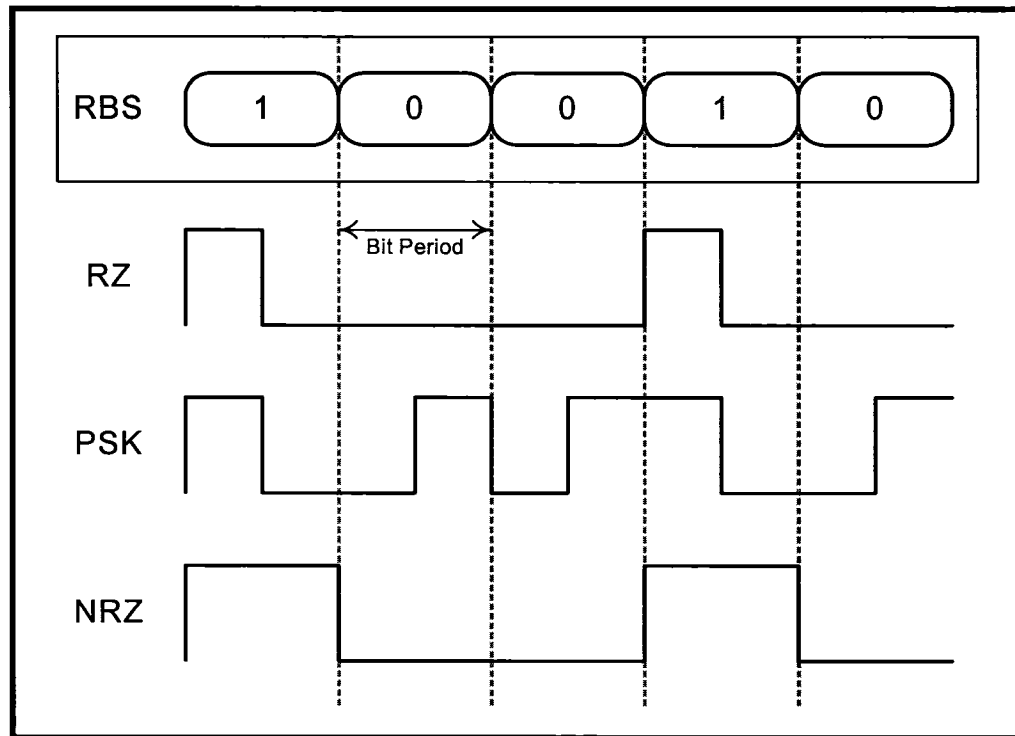
### **2.1 Non-Return-to-Zero Random Data & Spectrum**

There are several serial data formats for random binary or bit sequence (RBS). The three main formats are the return-to-zero (RZ) sequence, the non-return-to-zero (NRZ) sequence, and the phase-shift-keying (PSK) sequence. Figure 2-2 is an illustration of the three different types of RBS formats.

The PSK data sequence is worthy of notice due to its high data transition density. No matter the binary sequence, this data format guarantees a transition edge, i.e. from 0 to 1 or vice versa, spaced at one bit period. Therefore, among all data formats, this format contains the most information about its carrier clock. This is quite advantageous when trying to minimize jitter and locking time. On the other hand, the NRZ sequence is the most practical format for land-line applications, given the proper transition or binary density; in other words the NRZ sequence must be coded in such way that a certain density of transitions is present in order for the receiver to read the information properly.

The spectral information of a NRZ sequence is concentrated in a smaller bandwidth than the RZ or PSK coding sequence: the main spectrum lobe is confined within a smaller bandwidth. Figure 2-3 illustrates the spectral energy for the NRZ and RZ formats.

In the case of optical carriers, it is simpler to generate light pulses with NRZ coding, rather than RZ or PSK coding. Consequently, the circuit converting light to an electrical



**Figure 2-2** Time-domain illustration of different formats of random bit sequences: Return-to-zero (RZ), Phase-shift-keying (PSK), and non-return-to-zero (NRZ).

signal can be implemented with a slower technology, than with the NRZ or PSK formats.

## 2.2 Generating a Pseudo-Random Binary Sequence

Generating a RBS signal is used for testing and verification of CDR designs. However, these signals are sometimes not readily available in simulation tools or test equipment. Therefore it is important to address the synthesis of such signals.

First of all, a RBS is an infinitely long sequence that has no periodicity in its pattern. It can therefore be expressed as a long or infinite summation of shifted bipolar pulses (taking two polarities), as described by

$$x(t) = \sum_k b_k p(t - kT_b). \quad (2-1)$$

The signal  $x(t)$  is a binary weighted summation of the pulse  $p(t)$  with a  $T_b$  pulse width. The binary random sequence is represented by  $b_k$ , which can take values of either 1 or -1.

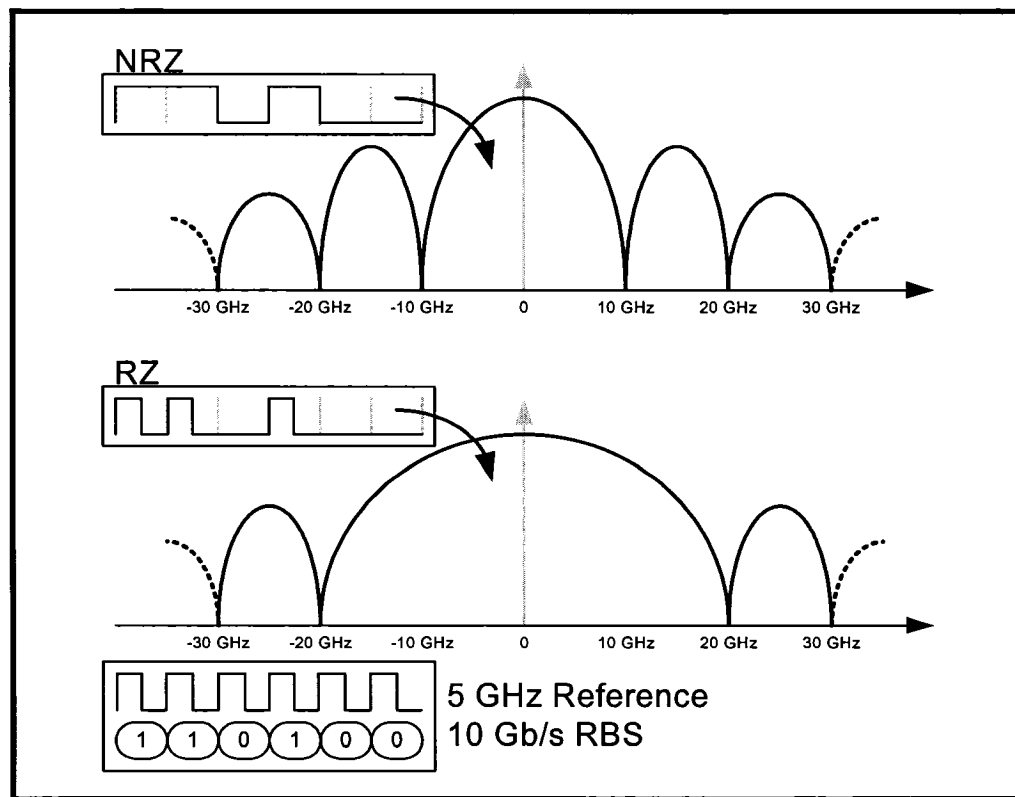


Figure 2-3 Spectral energy of NRZ and RZ sequences for a 10Gbps RBS along with their corresponding RBS in the time domain.

This bipolar sequence  $[b_0, b_1, b_2, \dots, b_k]$  is then multiplied by a pulse train  $p(t)$  spaced by the bit period  $T_b$ . Figure 2-4 illustrates the assembly of a RBS according to Equation 2-1.

Purely RBS signals are not practical since there is an eventual occurrence of long runs of trailing ones or zeros. Although very long runs occur with a small probability, this occurrence can cause serious problems in CDR receivers. CDRs have limitations in detecting long runs due to lack of transition edges: this causes jitter in the recovered clock and, at the very worst case, the VCO starts to drift slowly until it becomes out-of-lock. In order to avoid this inherent property of a RBS, there exists coding techniques, such as the 8B/10B and the 64B/66B codings, which limit the maximum length of trailing ones and zeros [12] & [13]. In other words, these techniques encode the data in order to guarantee a minimum density of transitions and a more balanced density of ones and zeros.

Another method consists of the generation of a pseudo-random binary sequence (PRBS). A PRBS is periodic over a defined sequence length, hence not entirely random.

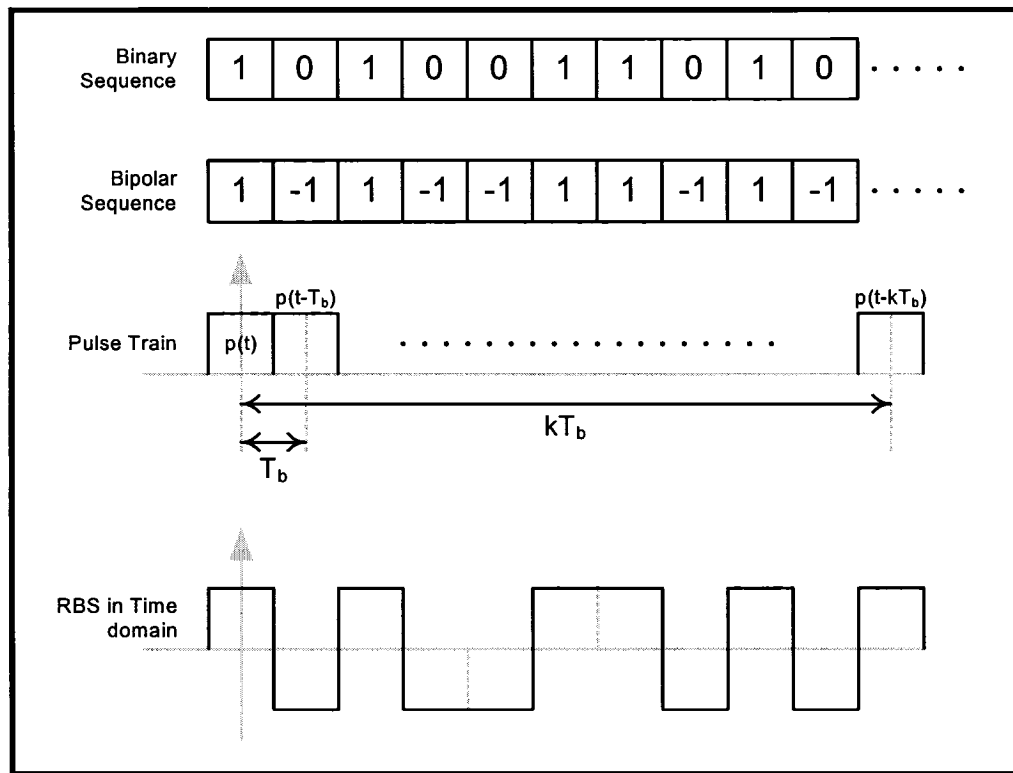


Figure 2-4 Illustration of an RBS synthesis with binary sequence, bipolar sequence, and pulses.

However, within a reasonable observation length of this periodic sequence, it can be considered random. Due to the long periodicity of a binary sequence, the maximum run length of ones and zeros of a PRBS can be controlled. PRBS signals are expressed in the following form:  $2^m - 1$ . The variable  $m$  indicates the maximum run length of either ones or zeros, and the value  $2^m - 1$  indicates the periodicity of the binary sequence. For instance, an aggressive type of PRBS for testing high-speed and high-performance CDRs is the  $2^{23} - 1$ . This PRBS generates a sequence of at most 23 consecutive ones and zeros, and has a periodicity of 8,388,608 bits. Therefore, within the observation length sequence of few million bits, this signal is considered random.

A PRBS can be generated with linear feedback shift registers. An example of a 1-tap 3 D-flip-flop linear feedback shift register (LFSR) is illustrated in Figure 2-5. This same digital circuit can be extended for larger PRBS sequences. In order to obtain maximal length sequences, it is necessary to have the proper taps. This configuration is further described in the Appendix (please refer to the section entitled “PRBS Generation in MAT-

LAB” on page 112). On the other hand, LFSRs can be manipulated through software, i.e. MATLAB. This is quite useful, especially for generating PRBS for simulation. An example of a MATLAB script is provided in Appendix (please refer to the section entitled “PRBS Generation in MATLAB” on page 112).

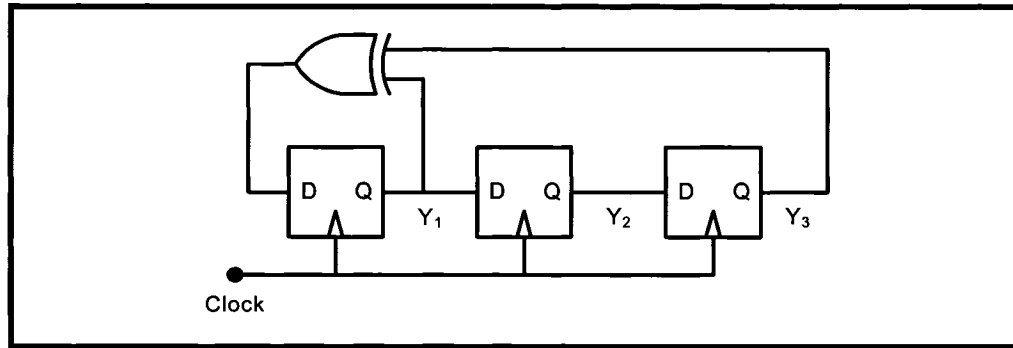


Figure 2-5 Illustration of a  $m=3$  1-tap LFSR with 3 D-flip-flops and an XOR gate.

### 3 EFFECT OF NOISE ON RANDOM DATA

#### 3.1 Relation Between the Signal-to-Noise Ratio (SNR) & the Bit-Error Rate (BER)

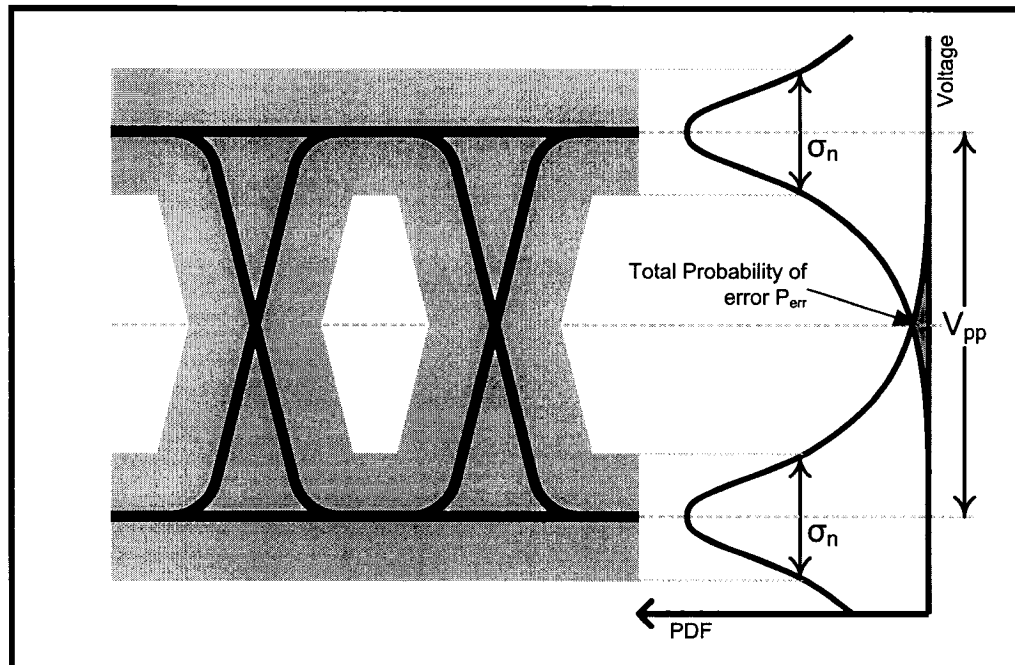
The bit error rate (BER) is a performance metric that indicates the reliability of a CDR for recovering data. As it will be explained further, the BER is closely related to the signal-to-noise ratio (SNR). The total probability of error (i.e. the BER) can be expressed in a simple form as in Equation 2-2 below, where  $Q$  is an approximation of the Gaussian integral function in Equation 2-3. This function is a valid approximation for SNR values greater than 6. Therefore, keeping a large ratio of peak-to-peak voltage  $V_{pp}$  and RMS noise  $\sigma_n$ , the BER remains small.

$$P_{error} = Q\left(\frac{V_{pp}}{2\sigma_n}\right) = Q\left(\frac{SNR}{2}\right) \quad (2-2)$$

$$Q(x) \approx \frac{1}{x\sqrt{2}} e^{\frac{-x^2}{2}}. \quad (2-3)$$

For a more intuitive explanation of the relationship between the SNR and the BER, a

graphical representation of the probability of error is illustrated in Figure 2-6. The two Gaussian probability density functions (PDF) representing the noise distribution on the voltage scale create an overlap. This overlapped region represents the total probability of error:  $P_{\text{err}} = P_{0 \rightarrow 1} + P_{1 \rightarrow 0}$ , the sum of the probability that a 0 is detected as a 1 and vice versa. Corroborating with Equation 2-2, the graph in Figure 2-7 demonstrates that the BER falls exponentially with the increase in SNR.



**Figure 2-6** Graphical representation of the Gaussian noise distribution, the peak-to-peak voltage ( $V_{pp}$ ), and the probability of error.

For instance, the peak-to-peak voltage cannot be increased boundlessly, especially for low-voltage applications. At this point, noise becomes the only variable that can significantly affect the BER. By limiting the bandwidth to approximately 0.7 times the bit rate, e.g. for 10Gbps operation, a 7-GHz 3-dB bandwidth is used, the noise power is limited without compromising the proper operation of the circuit ([14], [15]). Other sources of noise such as the supply/substrate noise and device noise should be reduced to improve the BER.



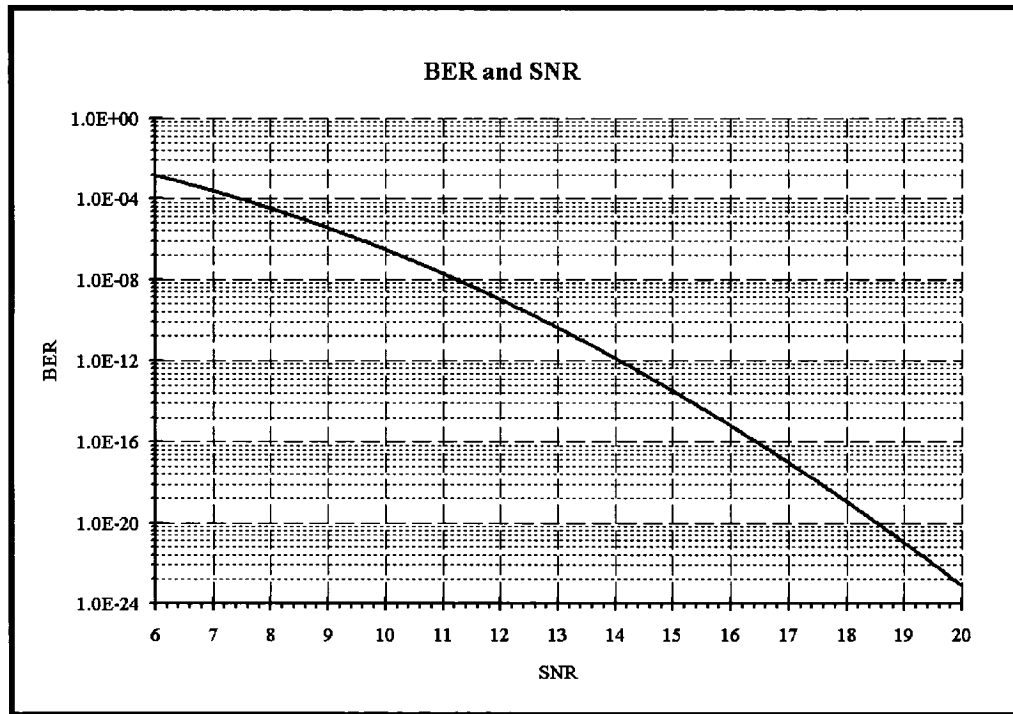


Figure 2-7 Graphical representation of the exponential relationship of the approximated BER for SNR values greater than 6.

## 4 PHASE NOISE & JITTER

For periodic waveforms, jitter and phase noise are interchangeable terms that express the “purity” or the “quality” of a digital or continuous-time synchronous signal. The measure of jitter and phase noise can be computed from one another given the proper conditions and information about types of jitter or phase noise.

### 4.1 Phase Noise

The mathematical expression for phase noise is included in the following equation as  $\phi_n(t)$ , being a small random signal

$$V_{out} = V_0 \cos[\omega_0 t + \phi_n(t)]. \quad (2-4)$$

This expression can be rewritten in order to clearly see the contribution of the phase noise term, as follows.

$$V_{out} = V_0 \cos(\omega_0 t) \cos(\phi_n(t)) - \sin(\omega_0 t) \sin \phi_n(t). \quad (2-5)$$

Note that  $\phi_n(t)$  is very small, therefore  $V_{out}$  can be approximated to

$$V_{out} \approx V_0 [\cos(\omega_0 t) - \phi_n(t) \sin(\omega_0 t)]. \quad (2-6)$$

In the frequency domain, the phase noise term  $\phi_n(t) \sin(\omega_0 t)$  has the spectral shape shown in Figure 2-8. Ideally, this skirt-shaped spectrum  $S_{\phi_n}(\omega)$  would fall at the rate of  $1/(\Delta\omega)^2$  about the frequency of oscillation  $\omega_0$ . Relative phase noise is measured according to Equation 2-7, where  $\Delta\omega$  is the frequency offset from the oscillation frequency  $\omega_0$ ,  $P_c$  is the power measured at frequency  $\omega_0$ , and  $P_{1Hz}|_{\Delta\omega}$  is the power measured at  $\Delta\omega$  away from  $\omega_0$  within a 1-Hz bandwidth resolution.

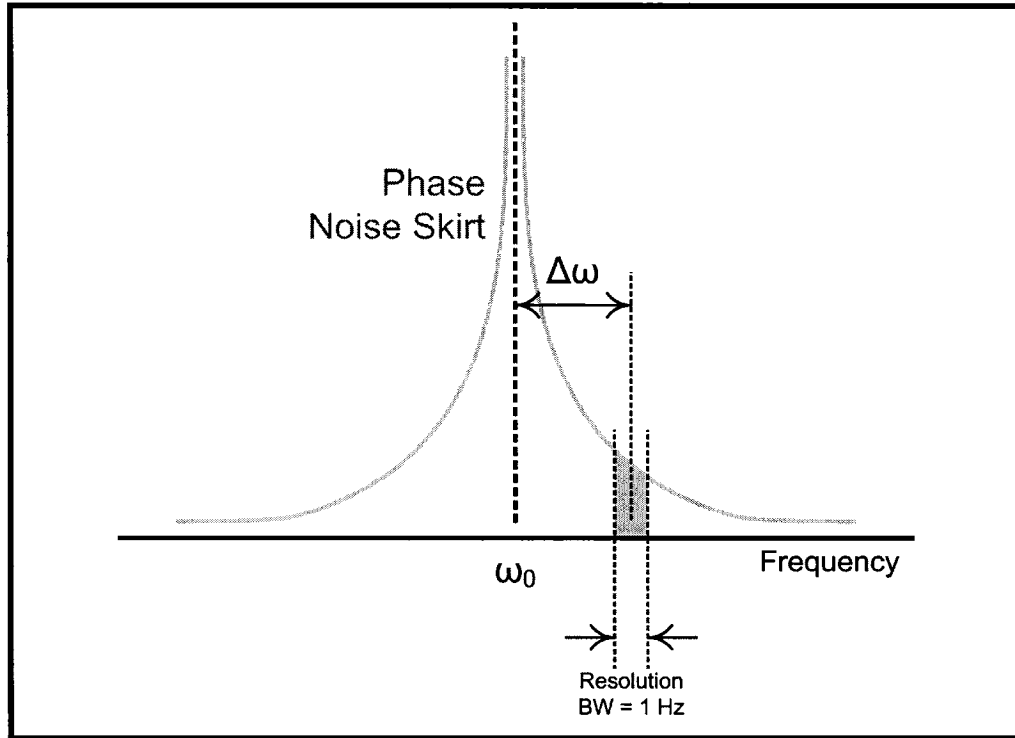


Figure 2-8 Spectral shape of phase noise.

$$\text{Relative Phase Noise}|_{\Delta\omega} = 10 \cdot \log \frac{P_{xHz}|_{\Delta\omega}}{P_c}. \quad (2-7)$$

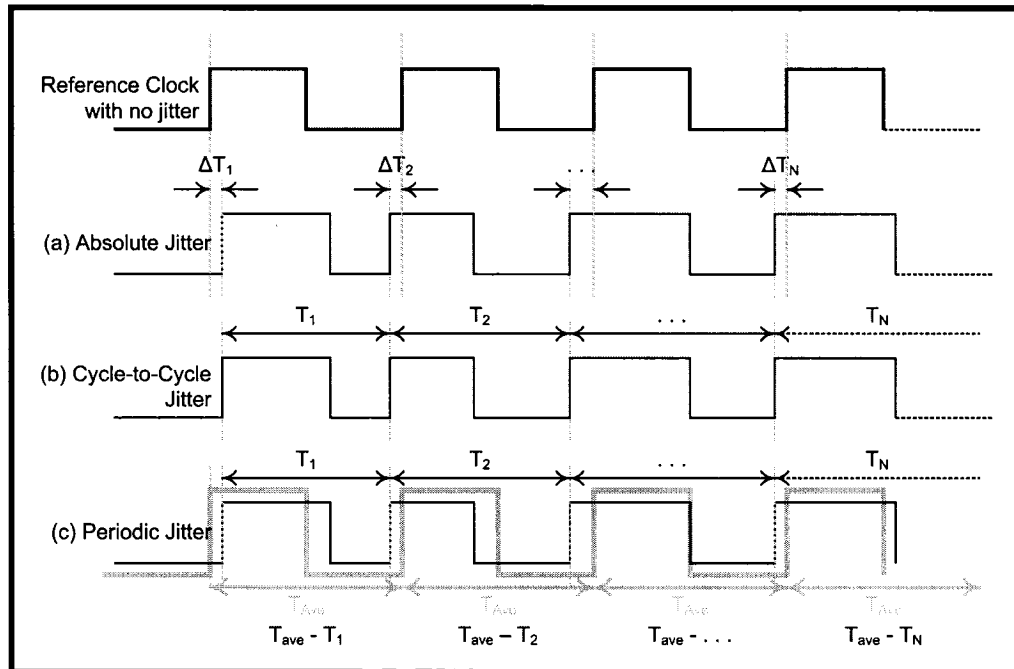
## 4.2 Jitter

Jitter is also a measure of the purity and quality of periodic signals. The jitter informa-

tion is given in the time-domain. This is more relevant to clock signals or periodic signals with sharp edges. The random phase noise component  $\phi_n(t)$  (Equation 2-4) can be expressed in terms of the zero-crossings represented by  $t_z$  as follows.

$$t_z = \frac{k\frac{\pi}{2} - \phi_n(t)}{\omega_0}, \text{ where } k = 1, 3, 5, \dots \quad (2-8)$$

These zero-crossings can be compiled and analyzed in three different ways as shown in Figure 2-9. The absolute RMS jitter  $\Delta T_{\text{ABS, rms}}$ , cycle-to-cycle RMS jitter  $\Delta T_{\text{CTC, rms}}$ , and periodic RMS jitter  $\Delta T_{\text{P, rms}}$  can be calculated, as shown in Equation 2-9, Equation 2-10, and Equation 2-11, respectively.



**Figure 2-9** Compilation and analysis of (a) absolute jitter, (b) cycle-to-cycle jitter, and (c) periodic jitter.

$$\Delta T_{\text{ABS, rms}} = \lim_{N \rightarrow \infty} \frac{1}{N} \sqrt{\Delta T_1^2 + \Delta T_2^2 + \dots + \Delta T_N^2} \quad (2-9)$$

$$\Delta T_{\text{CTC, rms}} = \lim_{N \rightarrow \infty} \frac{1}{N} \sqrt{(T_2 - T_1)^2 + (T_3 - T_2)^2 + \dots + (T_N - T_{N-1})^2} \quad (2-10)$$

$$\Delta T_{P, \text{rms}} = \lim_{N \rightarrow \infty} \frac{1}{N} \sqrt{(\bar{T} - T_1)^2 + (\bar{T} - T_2)^2 + \dots + (\bar{T} - T_N)^2}. \quad (2-11)$$

### 4.3 Mathematical Relationship Between Phase Noise and Jitter

In practice and in simulation, phase noise is more readily measurable than jitter. Equation 2-12 relates the absolute RMS jitter  $\Delta T_{\text{ABS, rms}}$  to phase noise over a large span of frequencies, given that the absolute jitter is represented by the following relation  $\Delta T_j = (2\pi / T_0) \cdot \phi_{n,j}$  [14].

$$\Delta T_{\text{abs, rms}}^2 = \left( \frac{2\pi}{T_0} \right)^2 \cdot \int_{-\infty}^{\infty} S_{\phi_n}(f) df. \quad (2-12)$$

CTC jitter (Cycle-to-Cycle) can be expressed as well through Equation 2-13, where  $\omega_0$  is the oscillation frequency and  $S_{\phi_n}(\Delta\omega)$  is the relative phase noise power at an offset frequency of  $\Delta\omega_0$  (refer to Figure 2-8).

$$\Delta T_{\text{CTC, rms}}^2 \approx \frac{4\pi}{\omega_0} \cdot S_{\phi_n}(\Delta\omega) \Delta\omega^2. \quad (2-13)$$

### 4.4 The Effect of Additive Noise

Additive noise degrades the jitter performance: Figure 2-10 shows an example of noise that can add jitter at the zero-crossing point. Sharper transitions allow to reduce the effect of additive noise on jitter, according to Equation 2-14, where  $\Delta T_0$  is the jitter due to additive noise at the zero-crossing point  $t_0$ ,  $n(t_0)$  is the instantaneous noise at  $t_0$ , and  $S$  is the slope or the slew rate of the waveform

$$\Delta T_0 = \frac{n(t_0)}{S}. \quad (2-14)$$

An approach to reduce noise is to minimize the  $n(t_0)$  term. This can be achieved by either reducing the available bandwidth to the optimal bandwidth, or by optimizing the circuits for better noise performance. The bandwidth reduction takes into account the BER

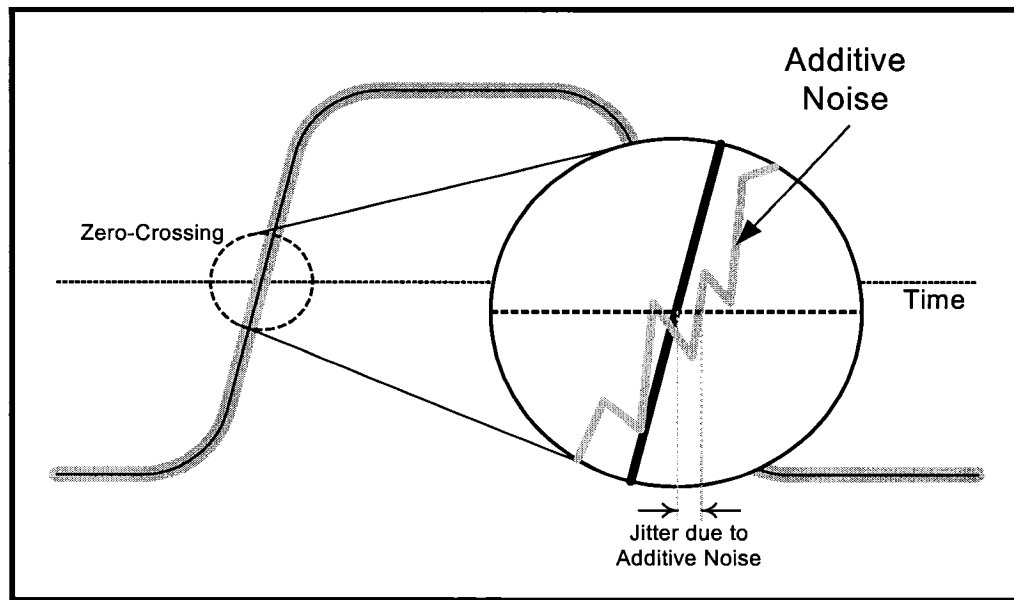


Figure 2-10 Jitter due to additive noise at the zero-crossing points.

and the speed of the system. The author in [15] shows that the optimum bandwidth is obtained by considering the following two constraints:

- The NRZ spectrum has an infinite span spectrum, as shown previously in Figure 2-3. Thus, the system needs a large bandwidth to avoid inter-symbol interference (ISI).
- The noise power is proportional to bandwidth. Thus, it is preferable to have the smallest bandwidth in order to minimize additive noise. Lower additive noise simply implies better jitter results.

For instance, a  $x$ -Gbps system demands an optimum bandwidth of  $0.75x$ -GHz. Hence this constraint will set a lower bound value for the noise term  $n(t)$  in Equation 2-14.

## 5 JITTER IN CDR CIRCUITS

Jitter is an important specification when designing CDRs. Depending on the application, the jitter requirement can be quite different. For instance, optical network applications demand more stringent jitter performance. However, it is important to note that CDRs are very much analyzed like PLLs, like the one described in Figure 2-11. Note that capacitor  $C_p$  is typically kept at least 5 times larger than  $C_L$  in order to minimize the effect

of the damping factor.

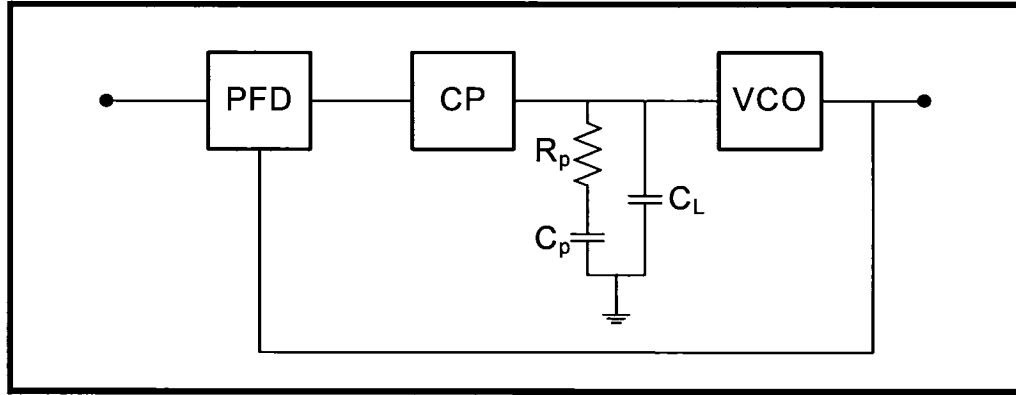


Figure 2-11 A block diagram of a CDR modelled as a PLL type II, where the PFD is the phase/frequency detector, CP is the charge pump, and VCO is the voltage controlled oscillator.

### 5.1 Jitter Transfer

Jitter transfer is simply a transfer response that measures the output jitter given an input jitter at a specific jitter modulation frequency. This transfer function also coincides with the phase transfer response  $H(s) = \phi_{out}/\phi_{in}$ , expressed in Equation 2-15 below. Note that Equation 2-15 assumes a linear type PFD. Although the linear and the bang-bang (or early/late) type PLLs do not behave in the same way, this expression can be used as an approximation for analyzing the early/late type PFD in a closed loop.

The natural frequency  $\omega_n$  and the dampening factor  $\zeta$  are given in Equation 2-16 and in Equation 2-17, respectively, where  $R_p$  is the loop filter resistance,  $C_p$  is the loop filter capacitor,  $D_T$  is the density of the incoming data transitions taking value between 0 and 1 exclusively (by default 0.5 for DC-balanced PRBS),  $I_p$  is the push and pull current for the charge pump, and  $K_{VCO}$  is the VCO gain in rad/volts. An example of a typical jitter transfer function is illustrated in Figure 2-12.

$$|H(s)| = \left| \frac{2\zeta\omega_n s + \omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \right| = \left| \frac{J_{out}}{J_{in}} \right| = \left| \frac{\phi_{out}}{\phi_{in}} \right| \quad (2-15)$$

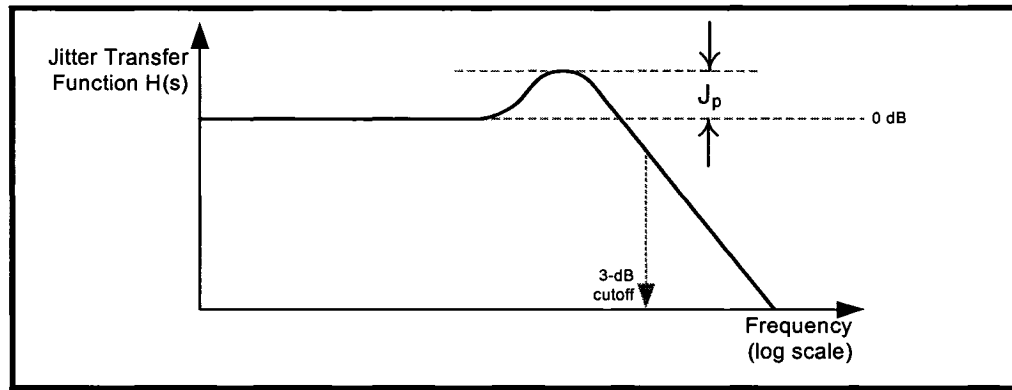


Figure 2-12 Graph of a typical jitter transfer function showing jitter peaking. Note that this graph is also applicable to the phase transfer function.

$$\zeta = \frac{R_p}{2} \sqrt{\frac{I_p C_p D_T K_{VCO}}{2\pi}} \quad (2-16)$$

$$\omega_n = \sqrt{\frac{I_p D_T K_{VCO}}{2\pi C_p}}. \quad (2-17)$$

The jitter peaking amplitude can be approximated as in Equation 2-18, and the 3-dB bandwidth is expressed in Equation 2-19.

$$J_p \approx 1 + \frac{1}{4\zeta^2} \quad (2-18)$$

$$\omega_{3\text{-dB}} = \frac{R_p I_p D_T K_{VCO}}{2\pi} \quad (2-19)$$

## 5.2 Jitter Generation

Jitter generation, as the name suggests, is the jitter that the system itself generates. The way to measure jitter generation is to introduce a jitterless RBS data signal at the input and to measure the jitter at the recovered clock. However, jitter generation may vary depending on the coding or the type of RBS at the input (refer to Chapter 2 Section 2.2). It is important to know the sources of jitter. It can result from the following:

- VCO phase noise [16]

- Noise at the VCO control voltage or in the loop filter [14] & [16]
- Unwanted cross-talk between system blocks, i.e. the PFD, the charge pump, etc.
- Supply and substrate noise [16]

The closed-loop jitter of a PLL is expressed in Equation 2-21, where  $f_u$  is the loop bandwidth. In fact, in a closed loop system, it was shown that jitter accumulates proportionally to the square-root of time, as shown in Equation 2-20 [16]. By substituting this equation (with  $\Delta T_{CTC}$  obtained from Equation 2-13), the PLL's RMS jitter is obtained in Equation 2-21

$$\Delta T_{ABS} = \sqrt{\frac{f_0}{2}} \Delta T_{CTC} \sqrt{t} \quad (2-20)$$

$$\Delta T_{PLL} = \frac{1}{\sqrt{2\pi f_u}} \sqrt{S_{\phi_n}(\Delta\omega)} \frac{\Delta\omega}{\omega_0}. \quad (2-21)$$

Noise at the control port of the VCO can generate jitter in the system. The cycle-to-cycle jitter resulting from noise at this port is expressed in Equation 2-22, where  $V_m$  is the modulating voltage at the control port of the VCO,  $K_{VCO}$  is the VCO gain,  $f_0$  is the oscillation frequency, and  $\omega_m$  is the modulation frequency of the signal at the VCO control port. Although this equation only relates jitter to a sinusoidal noise type at the control port, it can give some insight on how to reduce jitter generation. For modulation frequencies  $f_m$  much less than the oscillation frequency  $f_0$ , the cycle-to-cycle jitter can be approximated by Equation 2-23.

$$\Delta T_{CTC} = \frac{V_m K_{VCO}}{f_0^2} \sqrt{1 - \cos \frac{\omega_m}{f_0}} \quad (2-22)$$

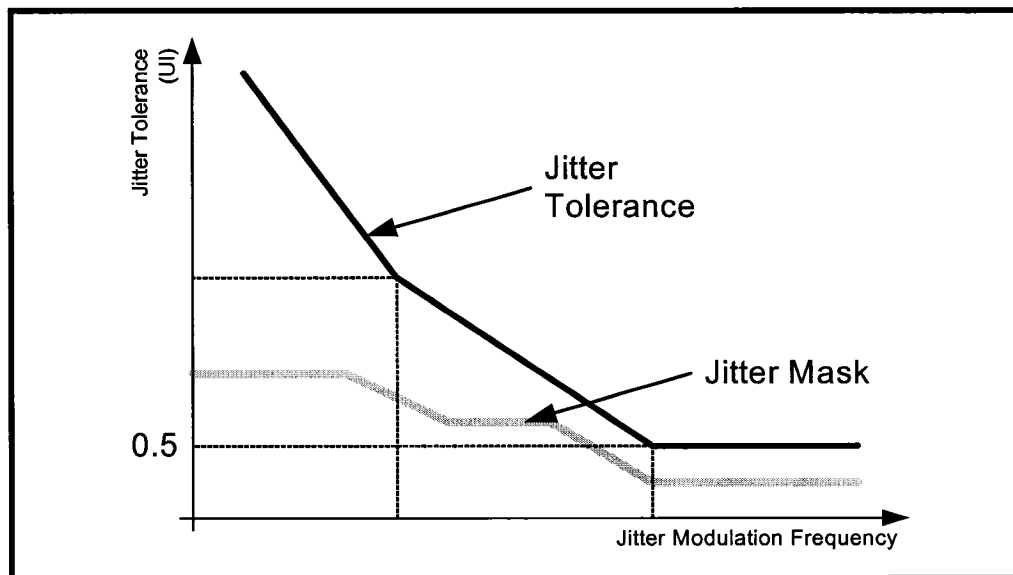
$$\Delta T_{CTC} \approx \frac{V_m \omega_m K_{VCO}}{\sqrt{2} f_0^3}. \quad (2-23)$$



### 5.3 Jitter Tolerance

Jitter tolerance measures the allowable input jitter modulation such that the recovered data does not increase in BER. The allowable input jitter amplitude is given in unit intervals (UI), which is equivalent to a 1 bit period.

Intuitively, slow jitter modulations at the input data would be tracked by the CDR, hence the data would be recovered without additional BER. As the jitter modulation frequency increases, the CDR starts having more difficulty tracking the changes, and therefore cannot allow large jitter at the input for it to maintain the same BER. In short, the CDR can tolerate larger jitter at low modulation frequencies and less as the modulation frequency increases, as illustrated in Figure 2-13. Note that a jitter tolerance graph is often accompanied by a jitter mask. This mask depends on the specification of the CDR or on the application context.



**Figure 2-13** An example of a CDR's jitter tolerance graph indicating the allowable input jitter versus modulation frequency, for a sustained BER at the recovered data.

Mathematically, the jitter tolerance plot can be obtained given the jitter transfer function (e.g. Equation 2-15). The constraint for setting up the jitter tolerance is to maintain the input and output phase,  $\phi_{in}$  and  $\phi_{out}$  respectively, less than half a bit period or  $1/2\text{UI}$  apart, hence the boundary set in Equation 2-24 below. By substituting this constraint in

Equation 2-15, the jitter transfer function  $G_{JT}(s)$  is obtained in Equation 2-25

$$\phi_{in} - \phi_{out} = \frac{1}{2}UI \quad (2-24)$$

$$G_{JT}(s) = \frac{s^2 + 2\zeta\omega_n s + \omega_n^2}{2s^2} = \frac{\phi(s)}{UI}. \quad (2-25)$$

Notice that Equation 2-25's two zeros coincide with the poles of  $H(s)$  in Equation 2-15, as shown in Equation 2-26 and Equation 2-27

$$\omega_{z1} = (-\zeta + \sqrt{\zeta^2 - 1})\omega_n \quad (2-26)$$

$$\omega_{z2} = (-\zeta - \sqrt{\zeta^2 - 1})\omega_n. \quad (2-27)$$

Knowing the location of these zeros, the jitter transfer function can be shaped in order to meet the desired jitter mask specifications. The loop bandwidth  $\omega_n$  (Equation 2-17), as well as the damping factor  $\zeta$  (Equation 2-16), can be modified in order to change the locations of the zeros in the jitter tolerance function  $G_{JT}(s)$ , as illustrated in Figure 2-14.

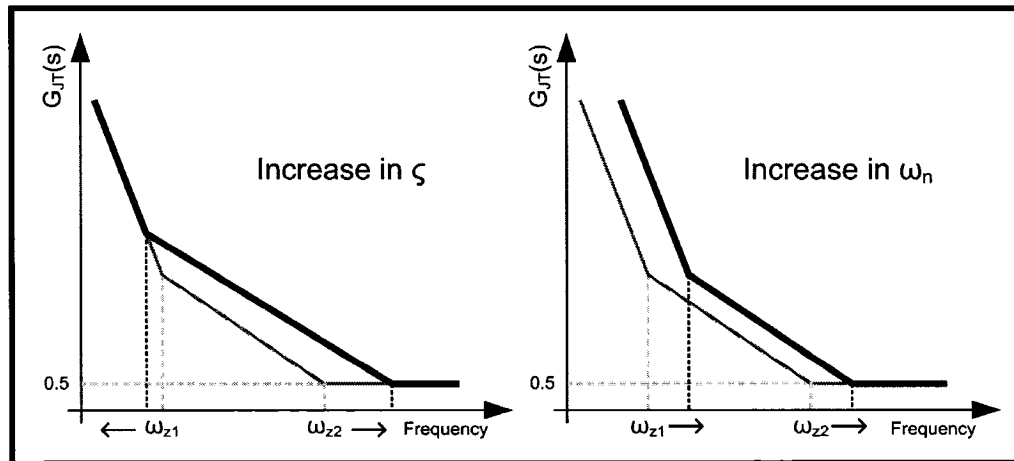


Figure 2-14 The corresponding changes in the jitter tolerance  $G_{JT}(s)$  graph given a change in the damping factor  $\zeta$  (left) or the loop bandwidth  $\omega_n$  (right).

## 6 SUMMARY

---

This chapter presented how a CDR circuit works and how it is useful. It also presented the introductory concepts of a CDR, and their theory of operation based on the type II PLL architecture. Since the CDR designed here is a PLL-based system, it was important to understand some of the similarities and differences when it comes to theory. As well, some performance metrics of CDRs are described for proper characterization.

---

# Chapter 3

High-Speed Current Mode Logic  
Circuit Building Blocks

---

## 1 INTRODUCTION TO CURRENT-MODE LOGIC

### 1.1 Concept of Current Switching

Current-mode logic (CML) circuits use the concept of current switching for representing the digital binary states. This type of circuit defines the binary states, high (one) or low (zero), by the presence or the absence of current at the output branches. Figure 3-1 is a simple illustration of the current switching concept for logical gates.

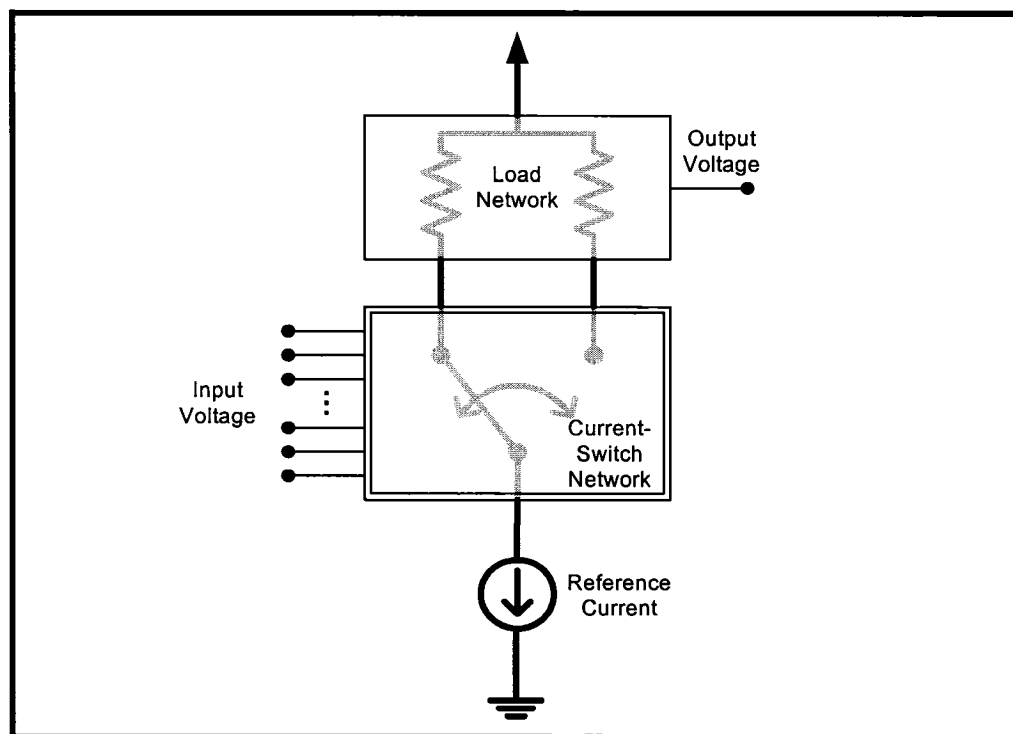


Figure 3-1 Simple schematic diagram illustrating the concept of current switching in current-mode logic – CML – gates.

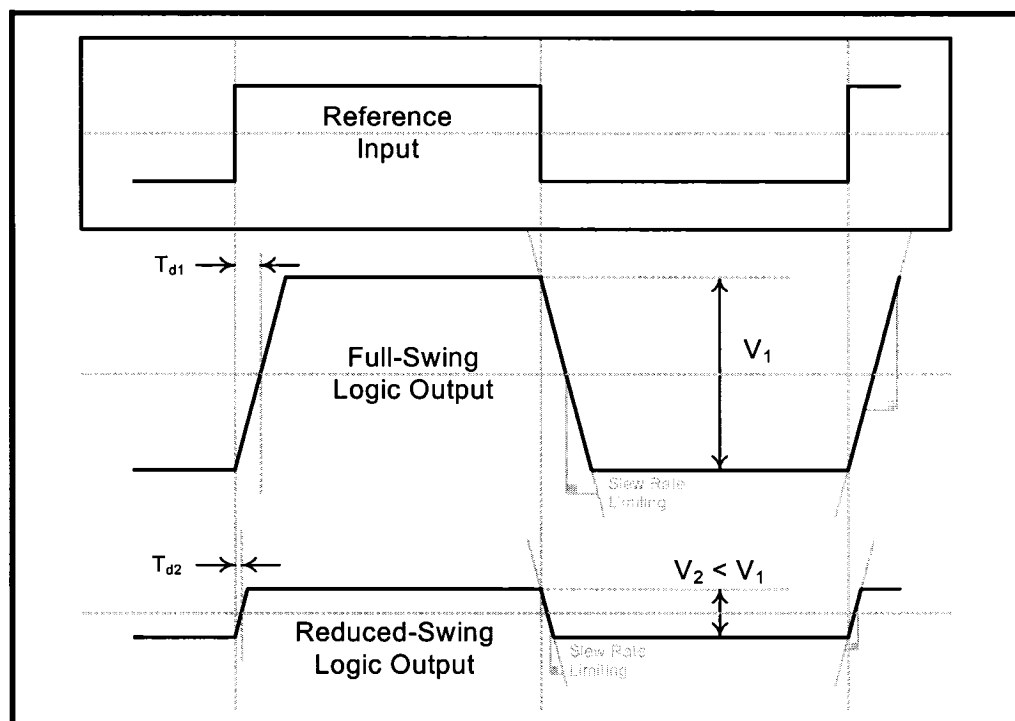
### 1.2 Advantages of the CML Style

CML gate designs are more practical than CMOS logic style for several reasons. First of all, CML circuits operate differentially, therefore any common-mode disturbance is suppressed by the common-mode rejection ratio (CMRR). Also, CML gates are known to have less substrate noise than their CMOS logic style counter part [17]. Due to its reduced logical voltage swing, propagation delays are shorter [18], which translates to faster switching circuits. Well designed CML gates can consume less power than the CMOS

logic style at higher-frequencies of operation [19]. Especially, CML gates reduce current spikes and peak currents during logical transitions at the supply, which contributes to lessening the effects of supply and ground bouncing.

CML circuit blocks are mainly designed for low-power and high-speed applications, due to their capability of switching at giga-Hertz speeds. For practical reasons, they are usually designed with resistive loads:

- These loads convert the current switching signals to voltages.
- These loads are designed to constrain the output voltage swings: by reducing the logical voltage swing, the propagation delays are reduced significantly. Figure 3-2 is an illustration of reduced versus full logic swings, and the corresponding propagation delays.



**Figure 3-2** Reduced swing versus full swing logic, and the corresponding change in propagation delay. This illustration is also a good comparison between the voltage swing of CMOS and MCML logic styles.

CML circuit blocks can be implemented with bipolar or MOSFET technologies. When implemented in bipolar, it is called ECL or emitter-coupled logic. The large transconduct-

tance  $G_m$  and the large transition frequency  $f_t$  of the bipolar technology allow ECL circuits to operate at extremely high frequencies compared to MOSFET transistors [20].

### 1.3 Performance Comparison

There are many circuit implementation techniques that are used with the CMOS technology for logic circuits. To name a few, circuits can be designed in complementary MOS logic (CMOS) along with their variations such as dynamic threshold CMOS (DTMOS), reduced supply bounce CMOS (RSBMOS), etc. Other implementing styles are the MOS current mode logic (MCML), folded source mode logic (FSCL), domino logic, complementary pass logic (CPL), differential cascode voltage logic swing (DCVSL), etc. The most popular styles are based on the CMOS logic for digital circuits. The CMOS logic style is known for being robust to fabrication tolerances, hence producing good yield. As opposed to the MCML style, these circuits are sensitive to resistors' process variation and mismatches. For a CMOS technology, resistors are known to vary up to 30%, which can affect the production yield. Moreover, CMOS logic circuit components are very simple to synthesize. Due to the popularity of the CMOS logic style, the MCML's characteristics will be compared mostly to it.

The main performance metric of digital gates are the power consumption and delay. The table below compares the delay, power, and power-delay products.

**Table 3-1** Table of comparison of MCML and CMOS circuit styles, where  $C$  is the total capacitive load, and  $I_{REF}$  is the bias current of the MCML [17].

	MCML	CMOS Logic
<b>Delay</b>	$\frac{C \cdot \Delta V}{I_{REF}}$	$\frac{C \cdot VDD}{k/2 \cdot (VDD - V_t)^\alpha}$
<b>Power</b>	$I_{REF} \cdot VDD$	$\frac{C \cdot VDD^2}{t_{delay}}$
<b>Power-Delay Product</b>	$C \cdot \Delta V \cdot VDD$	$C \cdot VDD^2$

where  $k$  is a process dependent parameter for MOSFETs, and  $\alpha$  is the technology size dependent variable having value between 1 and 2. The power-delay product expression represents the dynamic energy consumption due to non-ideal switching, where switching is defined as a transition from a logical high to low state or vice-versa. Clearly, Table 3-1 shows that the dynamic energy consumption of a single inverter is much higher in the CMOS logic style.

Another point of comparison is the cross-talk through the common voltage supply in high-speed digital circuits. Figure 3-3 is a diagram comparing the supplied currents for the CML and CMOS circuit styles for a simple buffer and inverter. The supply's dynamic current is quite stable for the CML style. In fact, when IC designs are packaged, the combination of the effective inductance due to the presence of bonding wires and large current spikes at the supply cause ground and supply bouncing [14]. When high-speed circuits are implemented in CMOS logic style, they can cause large current fluctuations during logical transitions. On the other hand, the MCML style has a somewhat stable supply current, therefore it is not so much affected by ground and supply bouncing.

## **2 CML SPEED OPTIMIZATION FOR A CMOS 0.18-MICRON TECHNOLOGY**

---

### **2.1 Design Parameters and Performance Requirements**

The correct set of performance requirements need to be set in order to ensure the proper implementation of a CML gate working at 10Gbps. The proper design of these gates has ultimately an effect on lowering the BER. The opening of the eye-diagram qualitatively indicates the performance of these gates.

As an example, the voltage gain is a very important design requirement or specification; it ensures that a chain of MCML gates can propagate the signal without degenerating or attenuating. It also ensures the rectification of slow ramping signals to sharp edges. In some sense, these gates must behave like comparators. Furthermore, the design of MCML gates requires the consideration of several parameters such as the unity-gain frequency, voltage gain, and voltage swing.



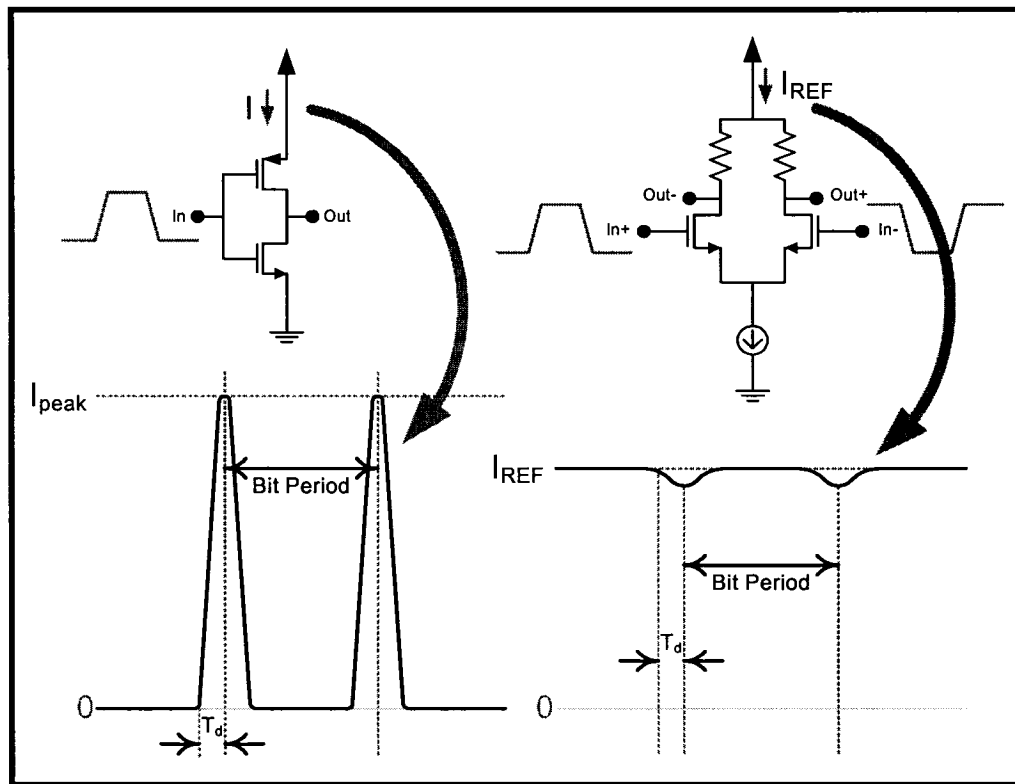


Figure 3-3 Supply current versus time for a switching inverter in CMOS style (left) and CML style (right) circuits, where  $T_d$  is the propagation delay and  $I_{REF}$  is the CML reference current.

## 2.2 Slew Rate & Frequency Response Relationship

One of the simplest approaches to qualitatively assess the speed of a circuit is to look at its frequency response. A higher bandwidth indicates that higher bit rate signals can be correctly regenerated. Also, the gain bandwidth product (GBWP) of the circuit truly characterizes its speed limitations. There exists a simple and direct correspondence between the slew rate and the frequency response.

The switching speed of a digital circuit is either described by the transition time or the propagation delay. Similarly, the output transitions can also be characterized by the slew rate, which is the maximum rate of change a gate can produce. Suppose an input signal is described by  $v_{in}(t) = A \cdot \cos(\omega t)$ , the maximum rate of change of this input signal is  $A \cdot \omega$ . At low frequencies, the small signal voltage gain is equal to the gain of the maximum rate of changes developed in Equation 3-1

$$\frac{\max\left(\frac{d}{dt}v_{out}(t)\right)}{\max\left(\frac{d}{dt}v_{in}(t)\right)} = \frac{\max\left(\frac{d}{dt}A_2\sin(\omega t)\right)}{\max\left(\frac{d}{dt}A_1\sin(\omega t)\right)} = \frac{A_2\omega}{A_1\omega} = \frac{A_2}{A_1} = \text{Gain}. \quad (3-1)$$

When the input signal pushes the gate to slew rate limiting, the output rate will be defined as  $SR_{MAX}$ . In other words, for frequencies beyond the 3-dB cutoff, the rate gain proportionally decreases with frequency, as shown in Equation 3-2.

$$\begin{aligned} \left. \frac{\max\left(\frac{d}{dt}v_{out}(t)\right)}{\max\left(\frac{d}{dt}v_{in}(t)\right)} \right|_{\omega > \omega_{3dB}} &= \left. \frac{SR_{MAX}}{\max\left(\frac{d}{dt}A_1\sin(\omega t)\right)} \right|_{\omega > \omega_{3dB}} \\ &= \left. \frac{SR_{MAX}}{A_1\omega} \right|_{\omega > \omega_{3dB}} = \text{Gain}(\omega). \end{aligned} \quad (3-2)$$

For the case of a simple MCML inverter, let us assume that the inverter switches currents instantaneously. Thus, the MCML gate can be viewed and analyzed as a cascode MOS voltage amplifier, as illustrated in Figure 3-4. According to the GBWP, the following expression can be derived for the simple MCML inverter stage in Figure 3-4.

$$GBWP = (G_M \cdot R) \times \omega_p = (G_M \cdot R) \times \frac{1}{RC_L} = \frac{g_{m1} \cdot g_{m2}}{g_{m1} + g_{m2}} \times \frac{1}{C_L}. \quad (3-3)$$

The size of transistor  $M1$  is assumed to be much larger than that of transistor  $M2$ . In other words,  $g_{m2}$  is assumed to be larger than  $g_{m1}$ . This allows the approximation of the overall transconductance  $G_M$  to be equivalent to  $g_{m1}$ . Relating the GBWP to the slew-rate limited gain, the following relation can be written

$$SR = \frac{G_M}{C_L} \cdot A = \omega_t \cdot A. \quad (3-4)$$

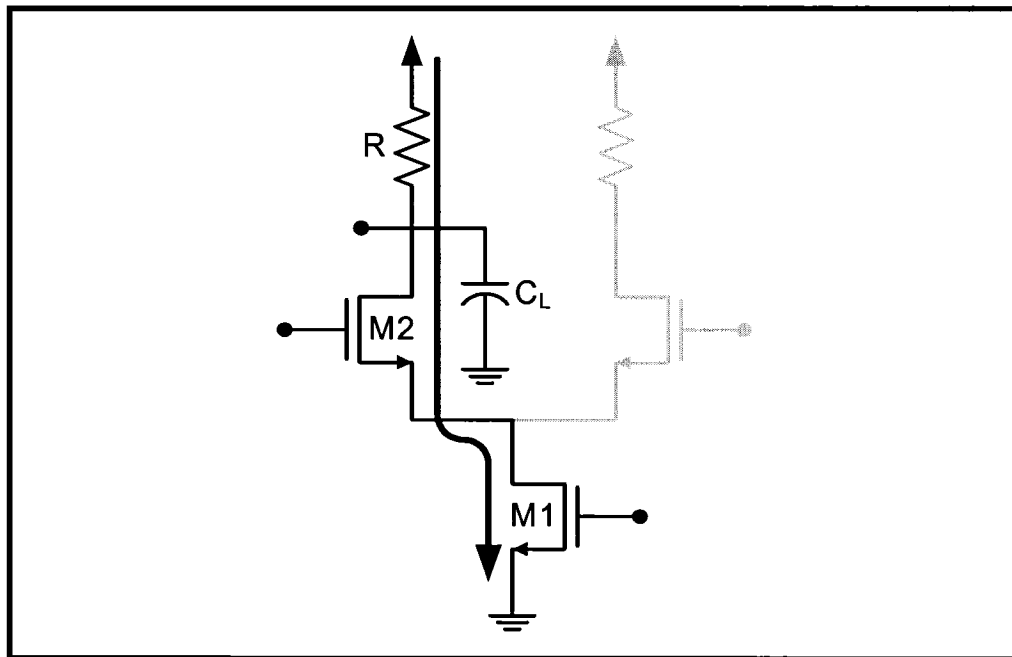


Figure 3-4 Approximate analysis of the small signal gain when switching in an MCML inverter. The thick arrow indicates the current path during the logic switch.

This relation indicates that the output slew rate limiting of a gate depends on the average transconductance  $G_M$ , the total capacitance at the load  $C_L$ , and the defined logical voltage swing amplitude  $A$ . This expression can also be written in terms of  $\omega_t$ , the unity gain bandwidths of the gate. Figure 3-5 illustrates graphically the change in the frequency response with respect to the change in the load capacitance  $C_L$  and the transconductance  $G_M$ . In order to obtain large rate of changes, the gates must operate at high speeds or have large bandwidths. More specifically, the CML gates must have larger unity-gain bandwidths (UGBW) or frequencies  $\omega_t$ .

### 2.3 Eye-Diagram and its Relationship to Speed

Figure 3-5 shows that the change in the overall transconductance  $G_M$  both increases the unity-gain bandwidth and the gain of the circuit. In order to increase the transconductance  $G_M$ , either the size of the transistor M2 (Figure 3-4) and/or the tail current must be increased. An increase in transistor size implies an increase in the overall capacitance  $C_L$ . On the other hand, an increase in tail current increases the voltage output swing. In effect, the latter solution slows down the response by increasing the propagation delay. Thus, gaining speed in a circuit is not straightforward and needs several iterations in order to

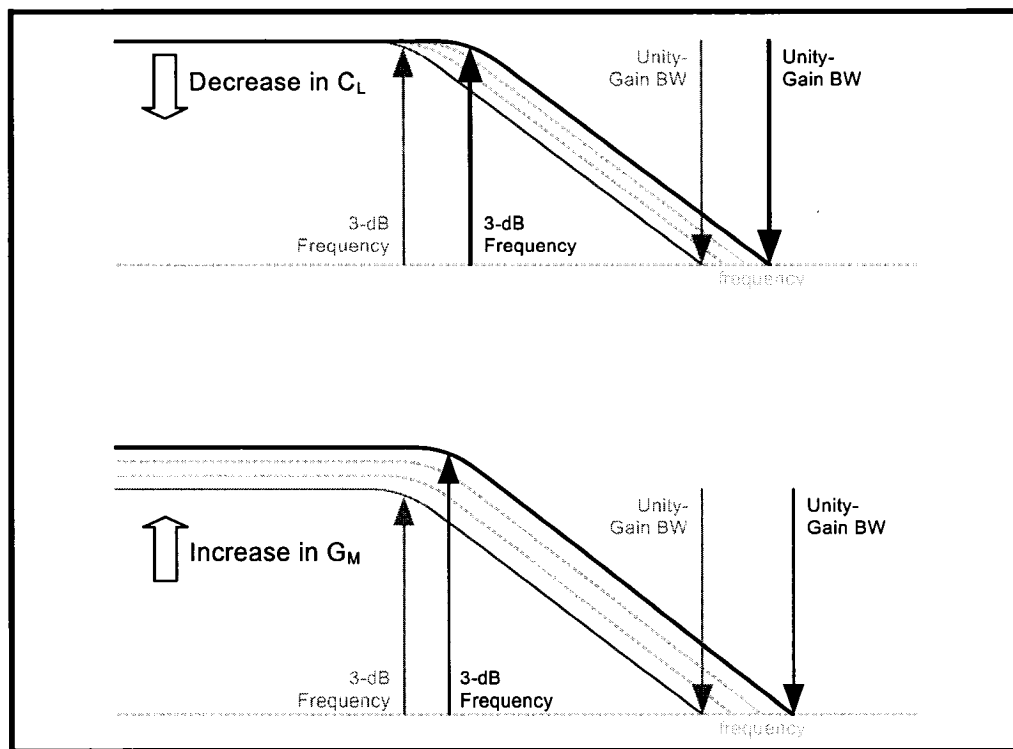
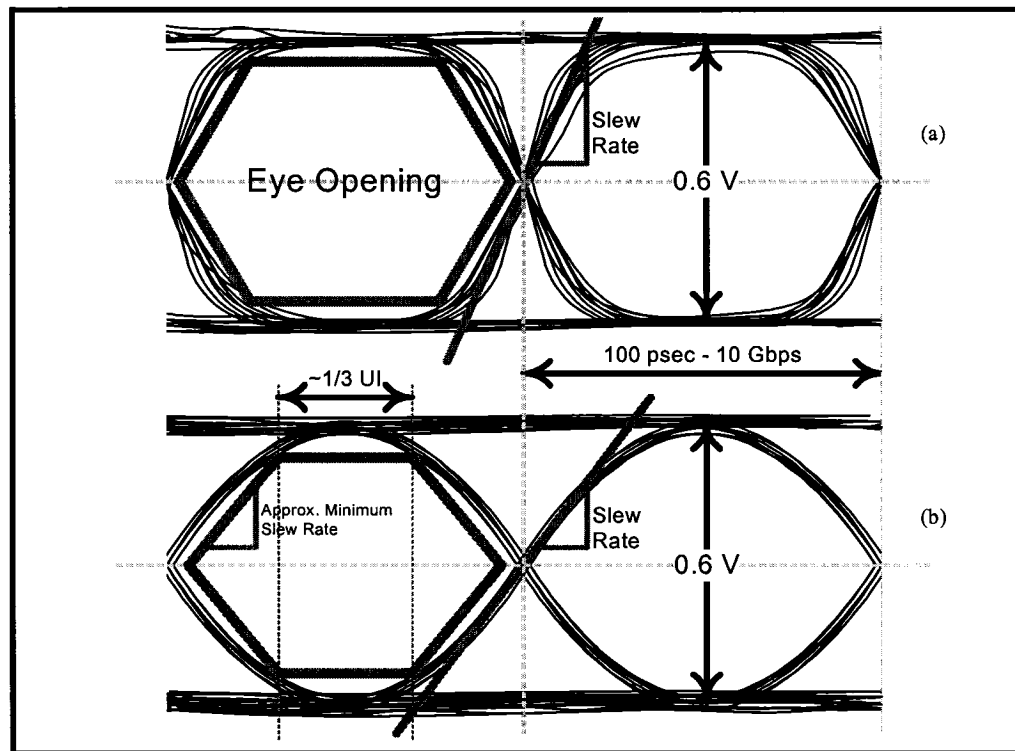


Figure 3-5 The effect of  $G_M$  and  $C_L$  on the frequency response of a CML Inverter.

optimize the performance.

In characterizing a circuit, two important parameters must be considered: the logical voltage swings and the corresponding speed. As explained earlier, the rate of change or the slew rate can give an indication on the speed of the circuit. Referring to the eye-diagram in Figure 3-6, the opening of the eyes is larger for higher slew rates. Larger eye opening implies smaller BER. Therefore, a minimum slew-rate specification can be set in order to produce a defined eye-opening. According to Figure 3-6 (b), the output voltage must rise by 0.3 V within 33 psec time run, which is equivalent to one third bit period, or unit interval (UI). Thus, a minimum theoretical slew-rate of 10 V/nsec is necessary, according to the minimum slew-rate polygon in Figure 3-6 (b). Nevertheless, in order to ensure quality and reliability, a minimum rate of change of approximately 15 V/nsec is necessary. Note that these numerical figures can be graphically inferred and are good approximations.

Finally, for the standard 0.18-micron CMOS available to us, we can now analyze the feasibility of running the MCML gates at 10 Gbps. Given that a slew rate of at least



**Figure 3-6** Quality of the eye diagram and its relation to the slew rate. The top diagram has a larger slew-rate compared to the lower. The bold hexagon shows the minimum slew-rate limiting for proper eye-opening.

15 V/nsec is needed, the unity-gain frequency must be in the order of at least 8 GHz, for a voltage amplitude of 0.3 V. However, according to the bandwidth optimization issues explained in Chapter 2 Section 3.1, it has been argued that the 3-dB bandwidth must approximately be equal to 7 GHz for 10 Gbps rates.

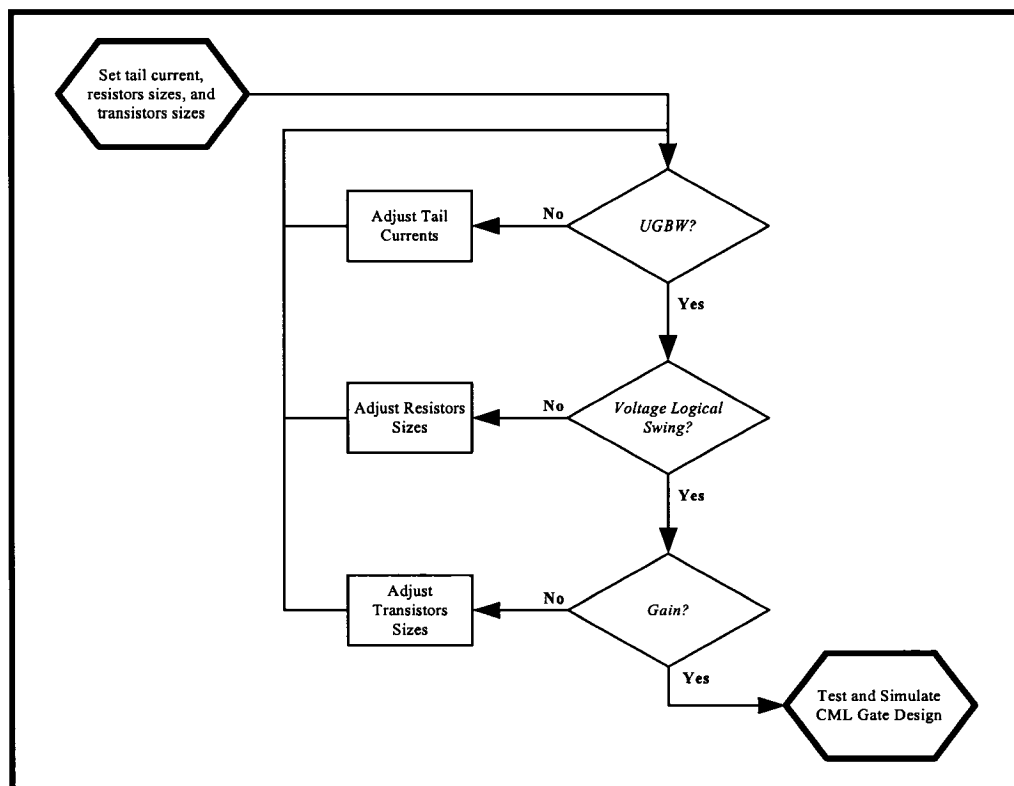
As for the gain specification, in order to regenerate and rectify the signals, the gain must be at least 0 dB. Given that the voltage gain strongly depends on the tolerances of the resistors, the gain must at the least 10 dB. Thus, the UGBW of the gate must be much larger than 7 GHz with approximately at least 10 dB of gain. Assuming a single-pole response with a 20 dB/dec roll-off, the UGBW of the gate should be closer to 20 GHz.

## 2.4 CML Design Challenges & Design Flow

Designing optimal CML gates can be challenging and time consuming. In order to simplify this task, some researchers prefer creating scripts to find the optimum design parameters for CML gates [22]. In this specific publication, the optimization problem def-

inition included very precise and specific set of constraints. In fact, such optimization programs are very practical, especially when migrating circuit designs between different technologies [21].

The most basic block, the CML buffer or inverter, is often used as an example for demonstrating the process of optimizing the design parameters: transistors sizes, tail currents, resistor sizes, etc. Although in this research no automated optimization or formal methods were used, the process of maximizing the blocks' gain and speed remains similar. A simplified design flow for the CML buffer is shown in Figure 3-7.



**Figure 3-7** A simplified design flow for CML gates in a CMOS technology. In this design flow, there are three variables and three criteria or requirements.

The design flow has three main criteria and three variables. The variables are listed in Table 3-2, with their corresponding effects on the design criteria. The criteria are formulated based on the discussion and analysis elaborated on earlier in this chapter. Table 3-2 summarizes the impact of parameters on a resistive load MCML circuit.

This optimization method is quite simple and can be done without the use of a script or

**Table 3-2 The corresponding effects of changing the main variables for an MCML buffer design: tail current, transistor size, and resistor size.**

Variable	Increase	Decrease
Tail Current	Increases in $f_T$ , voltage swing, and gain. Too large an increase may drive the switching transistors into triode.	Decrease in $f_T$ , voltage swing, and gain. Too small a voltage swing is sensitive to noise and increases the BER.
Transistors Sizes	Increase in $G_M$ , therefore more voltage gain. A larger transistor increases the load capacitance $C_L$ , therefore $f_T$ decreases.	Lower $C_L$ therefore increase in $f_T$ . Gain decreases due to decreasing $G_M$ , signal can degenerate.
Resistors Sizes	Increase in voltage gain, increase voltage logical swing.	Reduced logical voltage swing, decrease in voltage gain.

a program. There are no strict set of constraints on power dissipation, common mode rejection ratio (CMRR) or transistors sizes. Although not specified, it goes without saying that power dissipation is kept at a minimum, and that transistor sizing is kept reasonably small. As for obtaining good CMRR, several layout techniques can be applied to minimize mismatches.

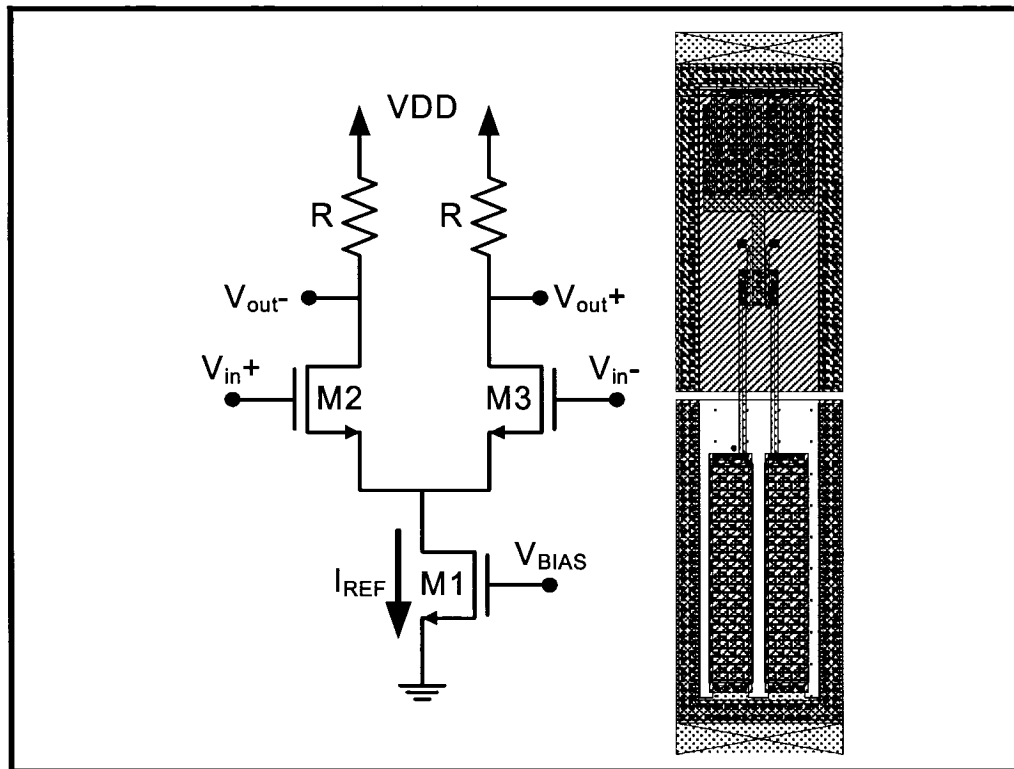
### 3 LOGIC CML CIRCUIT BLOCKS

#### 3.1 Inverter or Small Buffer

The MCML buffer amplifier is optimized to switch with a 10-Gbps PRBS. Using several iterations of the design flow in Figure 3-7, the implemented schematic and layout design specifications are listed in Table 3-3. This MCML buffer meets the performance requirements with a fanout of 2 to 3 other buffers, which represents worst-case loading conditions.

#### 3.2 Two-Port Multiplexer

The ideal-switch two-port multiplexer and the corresponding implementation in CMOS technology is shown in Figure 3-9. A common problem in designing multiplexers is the issue of unwanted signal feedthrough. The signal activity in the unwanted input ports can feedthrough to the output. This effect is called signal feedthrough, and is illustrated in Figure 3-10.



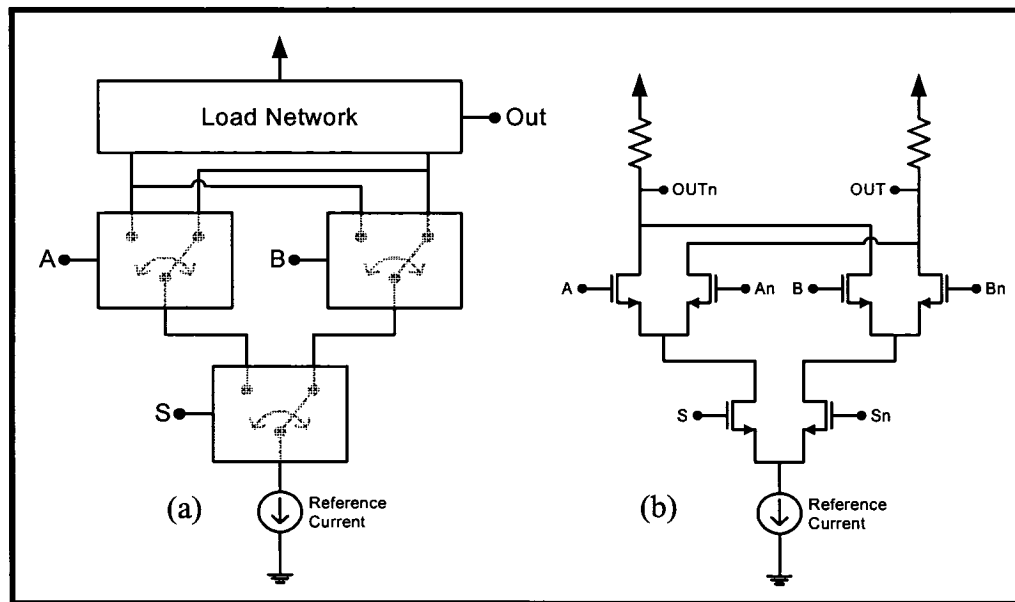
**Figure 3-8** Schematic diagram of the MCML buffer and the corresponding layout in a CMOS 0.18-micron technology.

**Table 3-3** Design parameters of the MCML buffer gate in a CMOS 0.18-micron technology.

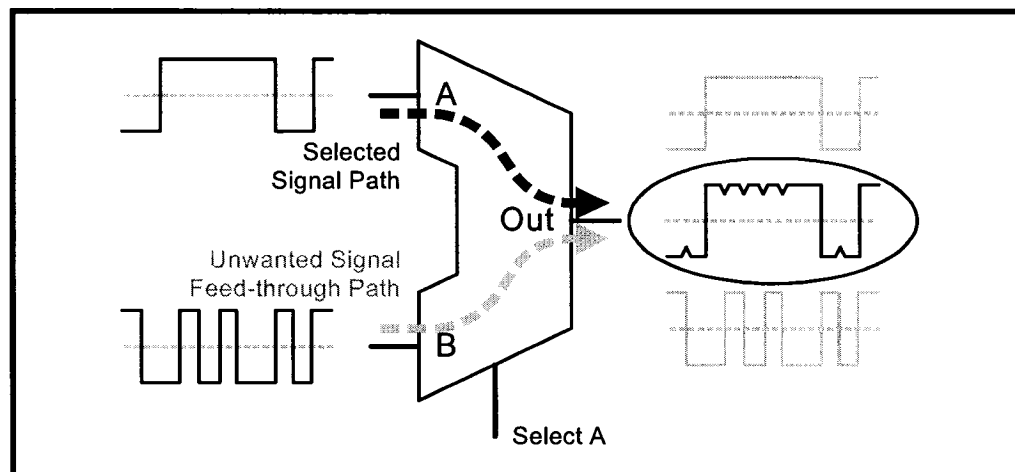
Sizing & Parameter	Value
Transistor Length	180 nm
Transistor Widths M2 & M3	2micron
Transistor Width M1	20micron
Resistor Size	1.785kOhm
Tail Current – $I_{REF}$	0.300 mA
VDD	1.5 V to 1.8 V

A simple and quick solution to this problem is to add buffers that regenerate the output. This will eliminate the glitches, given that the feedthrough is not severe. However, this solution implies the addition of extra gates, hence more power consumption and an increase in delay.



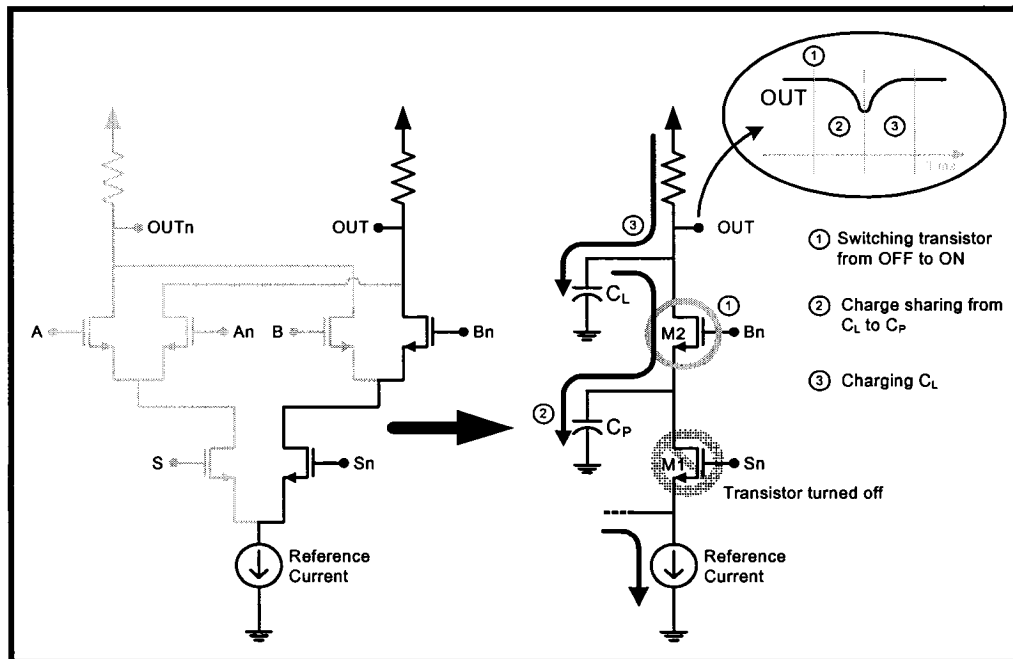


**Figure 3-9** (a) Schematic of a two-port (Port A & Port B) multiplexer circuit using ideal current switches and (b) the corresponding CMOS implementation with resistive loads.



**Figure 3-10** Signals in an MCML two-port multiplexer, showing the effect of feedthrough.

Another method consists of analyzing at the circuit level and identifying the sensitive nodes at which signal feedthrough is likely to happen. A novel method has been implemented in this design in order to eliminate this specific problem. This method is an adaptation of the bleeder transistor used for dynamic logic CMOS circuits [20]. As the name suggests, the bleeder drives small but constant current to a node that is susceptible to charge sharing. Figure 3-11 illustrates the feedthrough problem caused by charge sharing



**Figure 3-11** Illustration explaining the source of the signal feedthrough due to charge sharing in a two-port MCML multiplexer.

at specific nodes in the two-port MCML multiplexer circuit. The weak point of the circuit is capacitor  $C_P$  which is initially uncharged, then starts to share charges with  $C_L$  when transistor M2 is turned on. In order to mitigate this, a bleeder is connected to  $C_P$  in order to keep it charged. Note that this does not affect the function of the circuit, which is exactly what is desired. The bleeder's current is usually much less than the reference and operating currents. Too large of a current would prevent the top transistors from switching properly, as for a too small current would not replenish the capacitor  $C_P$  fast enough, hence leading to charge sharing.

The overall multiplexer circuit after adding the bleeder transistors is shown in Figure 3-13. In fact, any circuit with at least two stacks of differential switches will experience charge sharing. Therefore, the solution proposed here can be generalized for any MCML gates with cascoded switches.

### 3.3 D-Latch or Level-Sensitive Latch

The same design techniques presented earlier can be used to set the design parameters of the D-latch or the level-sensitive latch.

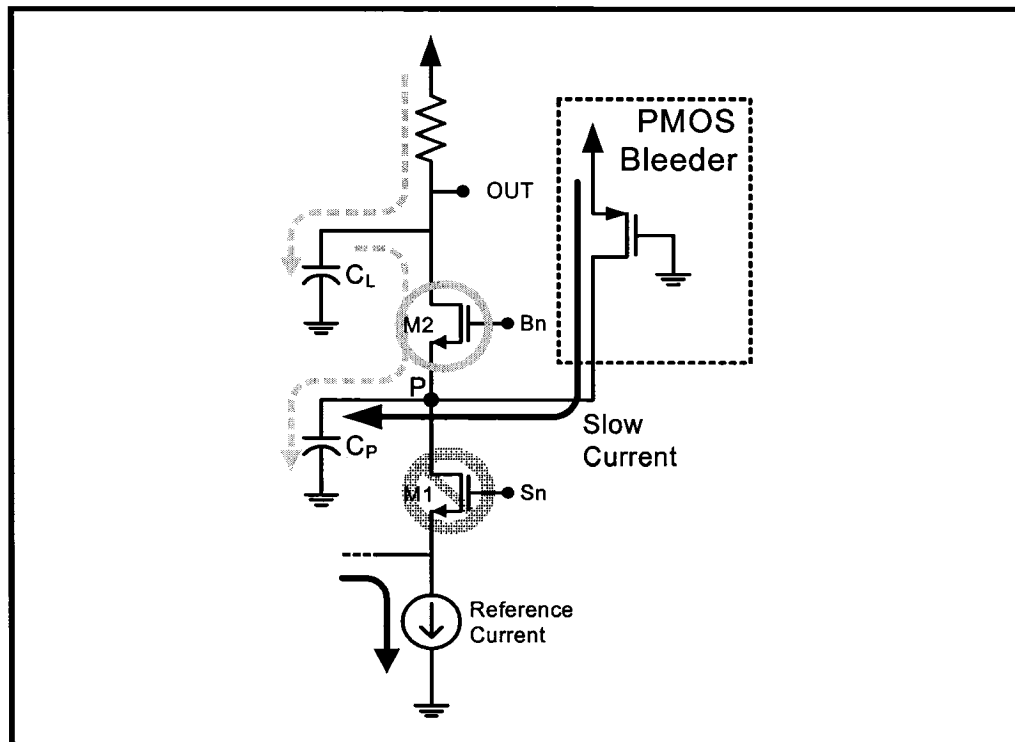


Figure 3-12 Solution to mitigate charge sharing at the node P. This bleeder pre-charges capacitor  $C_P$ , hence resulting in unwanted signal feedthrough.

The D-latch, also known as the level-sensitive latch, suffers from the same limitation as the two-port multiplexer, i.e. signal feedthrough. The unwanted feedthrough occurs when the latch is in the memory state. Placing a bleeder at the appropriate sensitive nodes of the circuit reduces the effect of feedthrough. The layout as well as the schematic diagram of the D-Latch gate are shown in Figure 3-14.

### 3.4 Further Design of Complex Circuit Blocks

The two-port multiplexer and the D-Latch gates are examples of complex circuit blocks. Many different gates can be designed in MCML given the right description of the function and the proper design method. However, the designer must take into consideration the limitations of the CMOS technology used. Not only do the MCML gates need to meet the requirements of speed, gain, and voltage logic swing; the system is bound by the supply voltage of 1.5V to 1.8V. This extra restriction limits the practical number of stacked differential switches to at most 2 levels. Any additional levels would either limit the logical voltage swing or make it difficult to meet other crucial performance require-

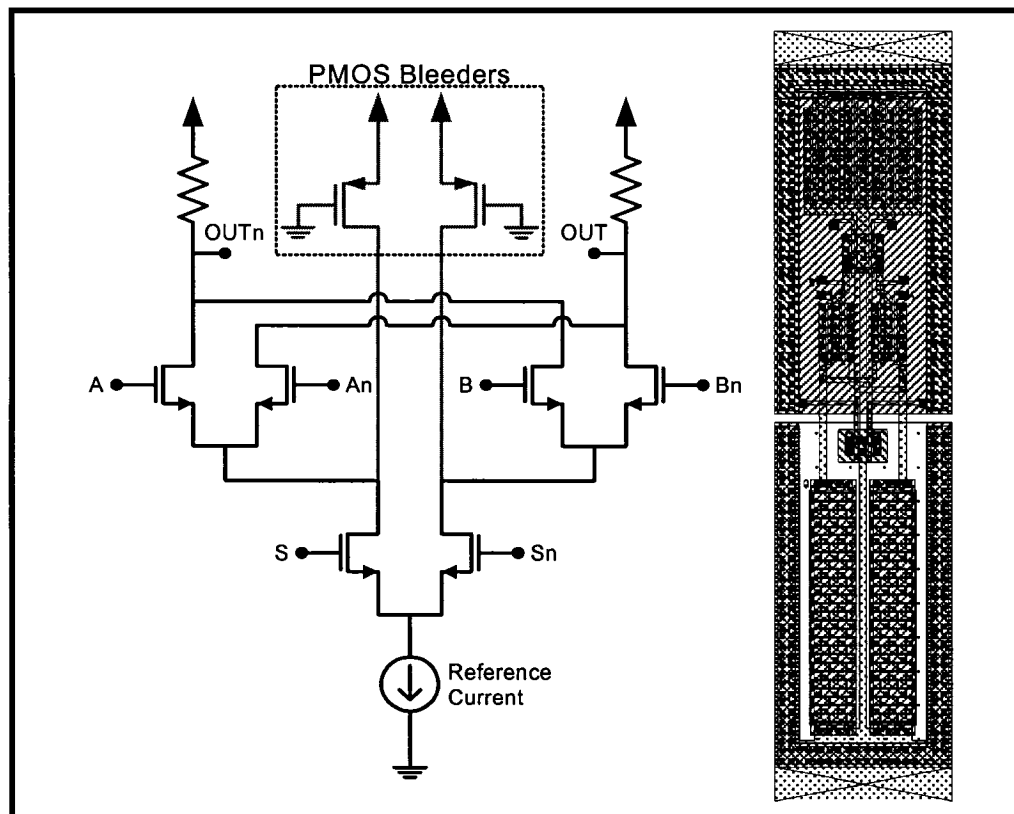


Figure 3-13 Schematic diagram of the MCML two-port multiplexer with no signal feedthrough and its corresponding layout in a CMOS 0.18-micron technology.

ments.

More complex systems of circuit blocks can be designed by combining the D-latch and the multiplexer, such as the edge-triggered D-flip-flop (ETDFF), the double edge-triggered D-flip-flop (DETFF), etc. The assembly of these system of gates are presented in detail in Chapters 4 and 5.

#### 4 SUMMARY

This chapter is the main contribution of this thesis. It addressed the analysis, design methods, and the design flow of CML gates for high-speed switching. It also compares it briefly to other logic styles, such as the CMOS logic styles. Finally, an extra novelty has been added to CML circuits to eliminate feedthrough of unwanted signals. This is the major contribution in terms of novelty.

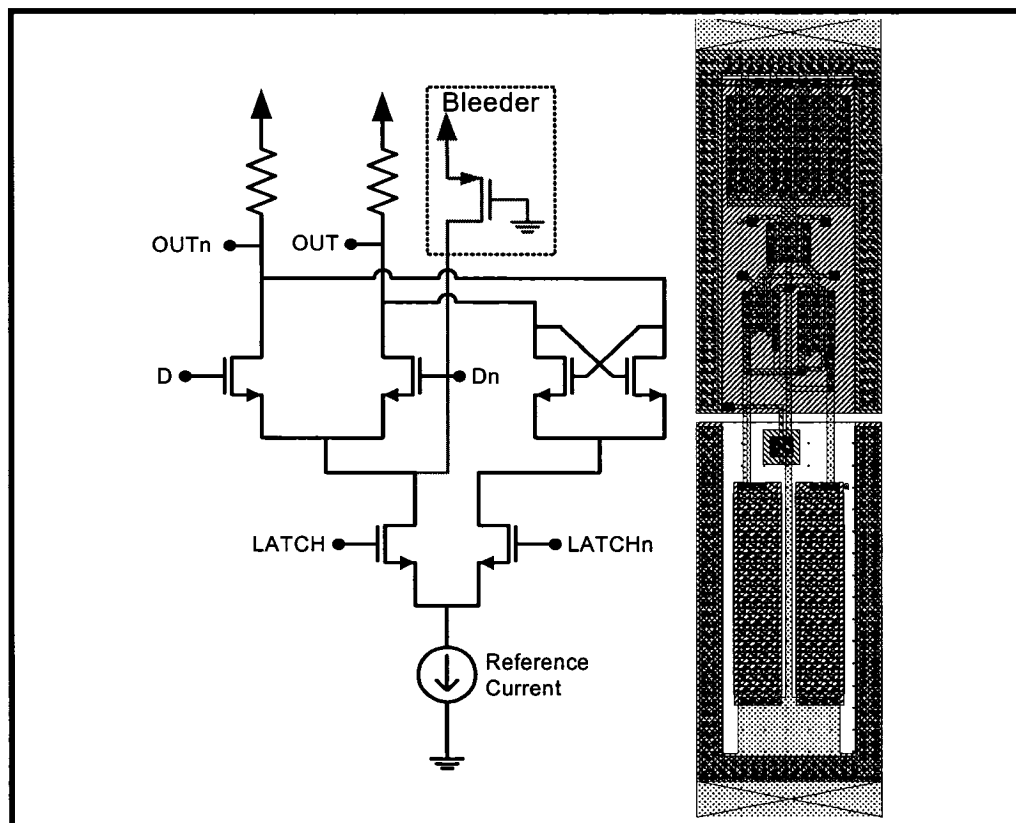


Figure 3-14 Schematic diagram of an MCML D-Latch (level sensitive latch) and its corresponding layout in a CMOS 0.18-micron technology.

---

# Chapter 4

Building Blocks of a PLL Based Clock  
and Data Recovery Circuit

---

## 1 INTRODUCTION

This chapter is a compilation of the assembly and the descriptions of all the circuit blocks of the CDR system presented in Figure 4-1.

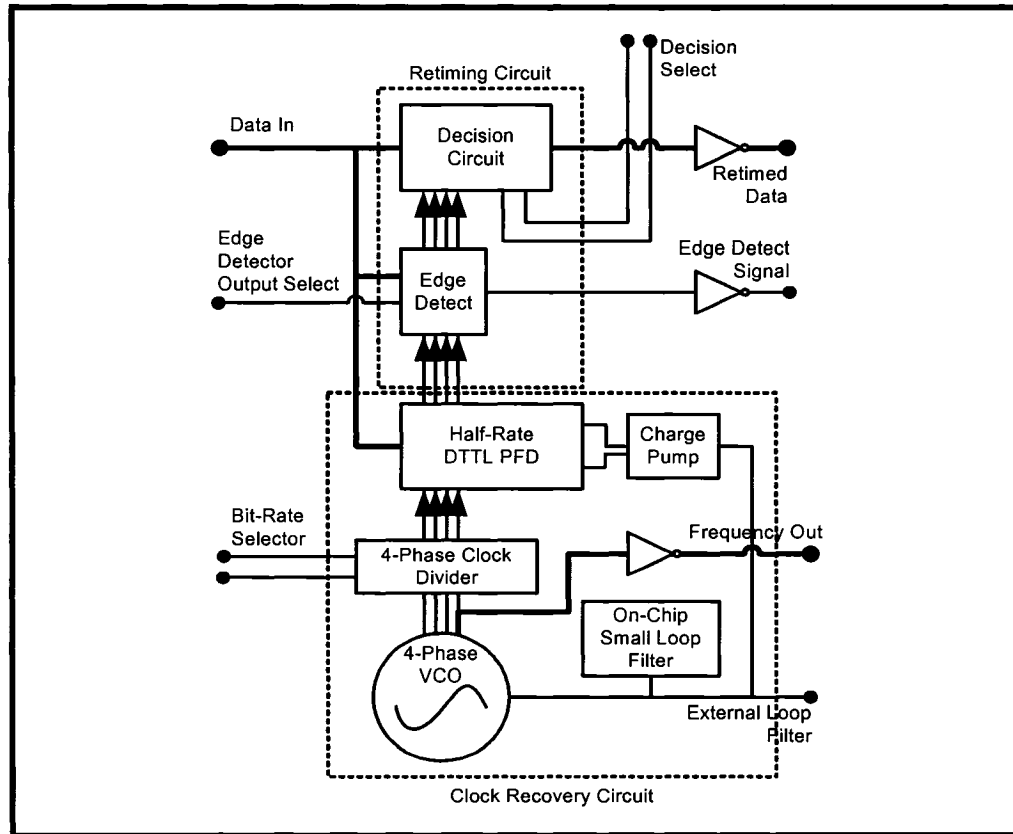


Figure 4-1 The system block diagram of the half-rate CDR with a multi bit rate selector.

## 2 BASICS OF PHASE & FREQUENCY DETECTION FOR RANDOM DATA

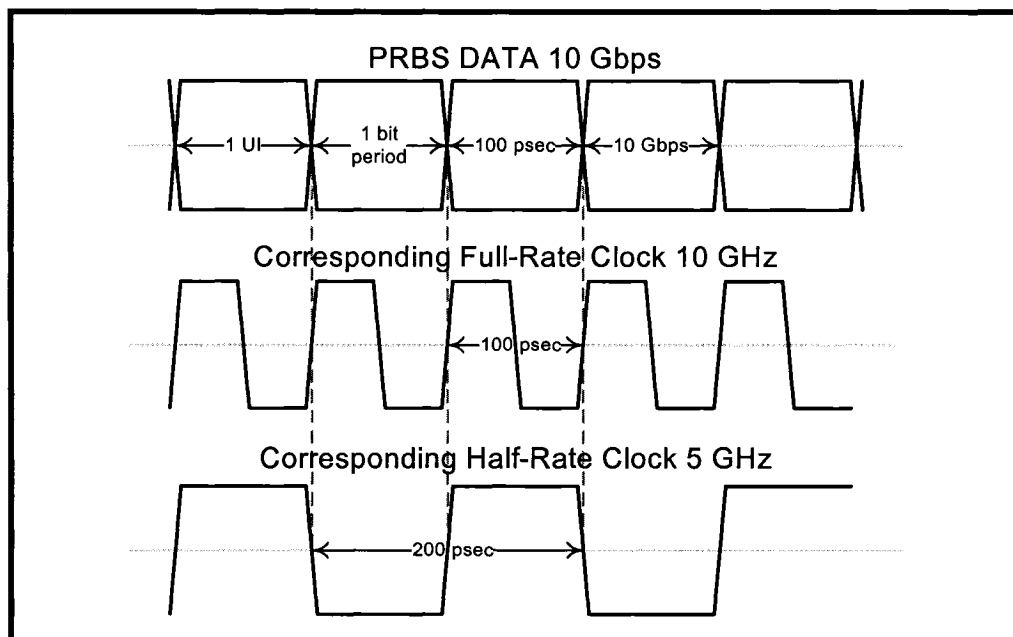
### 2.1 Basics of Phase & Frequency Detection

There is a noticeable preference toward the use of bang-bang, also named early-late, phase detectors compared to linear ones [2]. At high bit rates (above  $\sim 1$  Gbps), bang-bang PDs are typically used. These type of detectors are more robust to many types of variations, such as process variation, non-linear effects, etc. However these detectors are known to introduce more jitter than the linear ones.

CDRs and PLLs are similar at the system level: they are mainly composed of a

phase/frequency detector, a loop filter, a charge pump, and a VCO. However, the details of each block are different, in order to ensure proper operation with a pseudo-random bit sequence (PRBS) input. Several designs that can track the phase and/or frequency of a random bit sequence have already been proposed and documented, such as the Hogge detector [23], the Alexandre detector [24], the half-rate detectors [26]-[30], etc.

Ordinary phase detectors (PD) and phase/frequency detectors (PFD) designed for PLLs fail to work with CDRs because the input is no longer a coherent and periodic signal. The spectrum of a PRBS signal has no tones at its corresponding half-rate frequency and even less so at its full-rate frequency. For example, a 10-Gbps signal has an equivalent bit period to that of a 5-GHz half-rate clock, and a clock period of a 10-GHz full-rate clock, as shown in Figure 4-2. Furthermore, some form of rectification and edge detection must be used in order to track a PRBS signal. The two well-known PD designs are the Hogge and Alexander PDs. These full-rate PDs provide some form of rectification in order to extract the correct phase/frequency information from a PRBS.



**Figure 4-2** Illustration of a 10-Gbps PRBS and its corresponding full-rate and half-rate clock signals. Note the equivalent expressions representing a one bit period.



## 2.2 Basic Operating Principal of the Hogge Phase Detector

The Hogge phase detector is considered linear because it produces periodic pulses proportional to the phase delay between the data and the clock. In addition to these pulses, the output is accompanied by “reference pulses” which represent unique phase differences for various data patterns. Without these reference pulses, the transitions density can skew the phase delay results. Figure 4-3 is a simple schematic block illustration of the Hogge phase detector and its corresponding waveforms. Suppose an input data pattern has a given delay with respect to the clock. This will generate a pulse density at the signal  $X$  corresponding to this phase delay (Figure 4-3). Now, suppose a different data pattern with half the transition density, yet maintaining the same phase delay as the previous data pattern. Although both signals have the same phase delay, the output signal  $X$  corresponding to this new input will pulsate to a different average. Therefore, not all phase delays can be uniquely represented by the same pulse density at the output signal. As a solution, unique phase information is obtained by subtracting  $X$  – the phase difference – from  $Y$  – the reference pulse (Figure 4-3). Under the phase-locked condition, the net phase difference is zero: the area under both  $X$  and  $Y$  signals are equal,  $(X(t) - Y(t) = 0)$ .

The  $X$  and  $Y$  signals, identified as the Down and Up signals respectively, can be fed into an up/down V/I converter. The V/I convert performs the subtraction of these signals and outputs the corresponding push or pull current signal into the loop filter. In closed-loop operation, the PD experiences little activity when the CDR is in the locked condition. The bottleneck occurs at the V/I converter or the charge pump. The pushing and pulling of current must happen fast and continuously in order to avoid a noticeable deadzone in the phase response when  $X(t) - Y(t) \approx 0$ .

## 2.3 Basic Principals of the Early-Late Alexander Phase Detector

The Alexander phase detector is of the bang-bang or early-late type. Therefore, unlike the Hogge PD, this design relies less on the performance of the V/I converter, since the minimum up/down pulses are no smaller than a bit period. On the other hand, the Hogge PD generates pulses at signal  $X$  that take any width less than a bit period. Therefore, the constraints can be relaxed on the charge pump. In essence, this design only determines

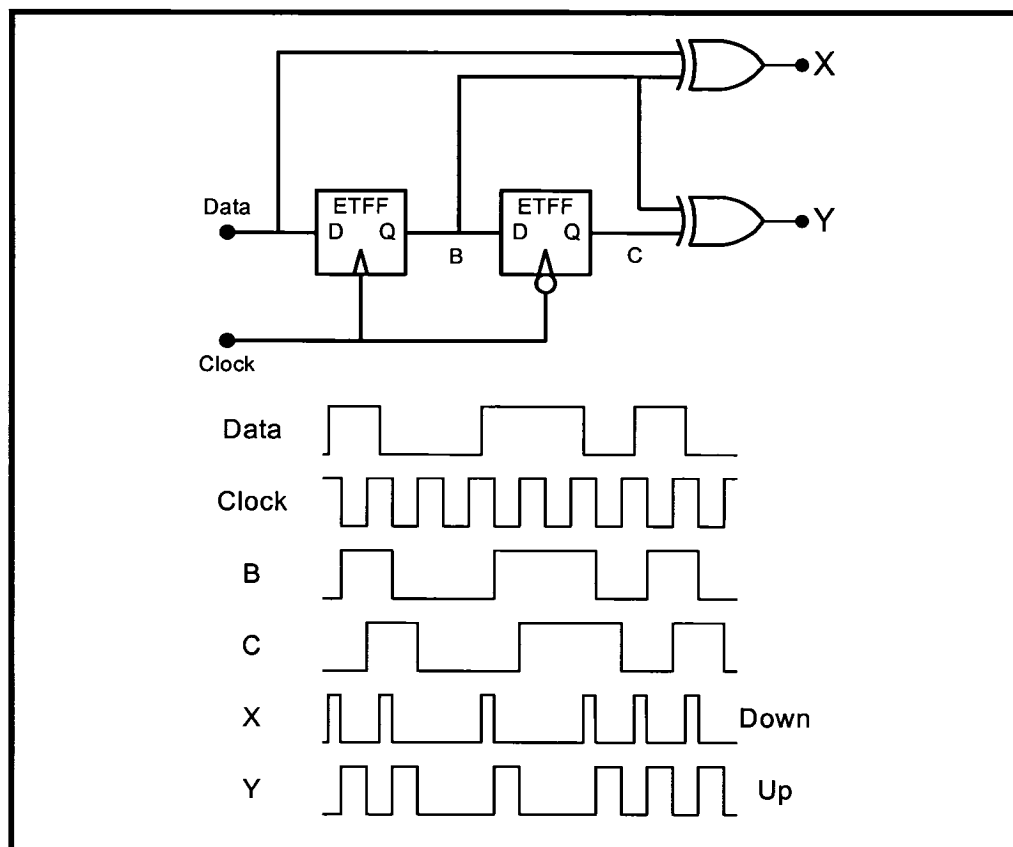


Figure 4-3 Full-rate Hogge phase detector and its corresponding timing diagram for a late clock signal – ETFF stands for Edge-Triggered Flip-Flop.

whether the clock is early or late. Illustrated in Figure 4-4 is the structure of the Alexander's early-late phase detector, along with a simple timing diagram.

The Alexander PD samples three points about a data edge and determines whether the clock is early or late. Under near phase-locked conditions, the sample point *C* is located about the edge of a data transition, changing rapidly between early and late state. For a late clock, the sampling edge *C* is located after the data edge, and before the data edge for an early clock (represented by *C'* in Figure 4-4). Moreover, the interesting feature of this circuit is that it remains idle during absence of data transitions. Therefore, this design shows a marginally better jitter performance, compared to the ordinary binary-state PD. This same feature accommodates the occurrence of trailing ones and zeros.

In summary, this design compares the three samples about a data transition and uses an XOR logic to determine whether the clock is early or late. A truth table (Table 4-1) can

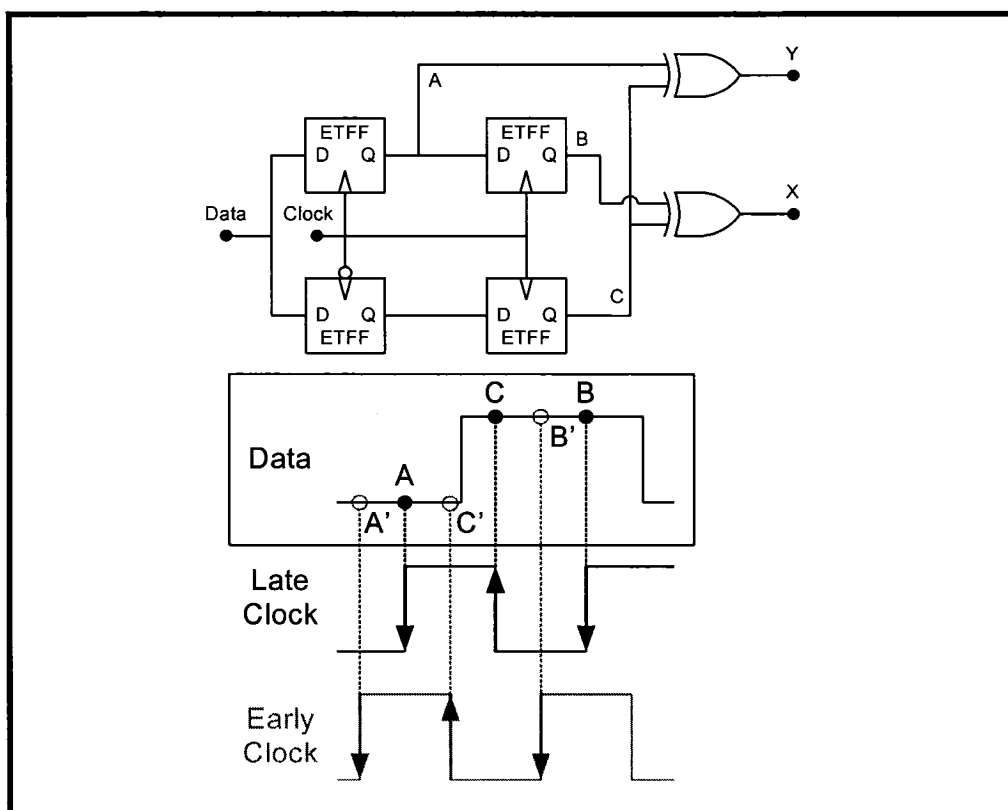


Figure 4-4 Full-rate Alexander phase detector and the early and late waveforms – ETFF stands for Edge-Triggered Flip-Flop.

therefore be generated. The  $X$  and  $Y$  signals are then fed to a V/I converter or charge pump to indicate a speed up or slow down signal to the VCO.

Table 4-1 Truth table representing all states of the full-rate Alexander phase detector.

Data about edge $C$	A	B	C	$Y = A \oplus C$	$X = B \oplus C$	Detection
	0	0	0	0	0	Idle
	0	0	1	0	1	Early
	0	1	0	1	1	—
	0	1	1	1	0	Late
	1	0	0	1	0	Late
	1	0	1	1	1	—
	1	1	0	0	1	Early
	1	1	1	0	0	Idle

It is worthy to note that the concept of the Alexandre early-late PD can be extended for half-rate operation. It is a matter of modifying the full-rate Alexandre PD circuit in Figure 4-4 using quadrature half-rate clock signals. Table 4-1 would still be applicable for half-rate operation.

## 2.4 Implementation of a Half-Rate Phase Detector – Data Transition Tracking Loop

Half-rate phase detectors are of particular interest since they relax the speed constraints of the logic gates. The purpose of this approach is to provide a valid phase detection while sensing a PRBS with a half-rate clock, i.e. a  $x$ Gbps data is clocked at an  $x/2$ GHz instead of an  $x$ GHz.

The full rate designs described in Chapter 4 Section 2.2 and Chapter 4 Section 2.3 are shown only for full-rate operation. However, they can be modified to accommodate half-rate clocks. For instance, the Alexander phase detector can be altered to sample at quadrature phases in order to function at half rate. Similar modifications can be made to the Hogge phase detector for half-rate operation.

More complex structures of half-rate phase detectors are available for study ([26] & [27]). It is shown in the literature that both linear and bang-bang phase detectors are feasible at half-rate ([11], [14] & [27]). The one of particular interest is the binary phase detector published by J. Savoj & B. Razavi [11]. This half-rate PD avoids the use of XOR gates, which usually constitute the main performance and speed bottleneck. Like the Hogge and Alexandre PDs, this half-rate detector provides information on transitions only; however it has the weakness of remaining in a memory state during the absence of data transitions. Other PDs, like the Alexander type, are designed with an idle state (tri-state) during the absence of transitions [31].

A binary PD is implemented because it can be easily extended to be a phase and frequency detector, unlike the Alexander and Hogge phase detectors which do not have a simple aided frequency acquisition implementation. A binary PD works as a data transition tracking loop (DTTL). As the name suggests, it tracks the data transitions in order to obtain information on its phase relative to the clock. This phase detector is constructed of

double edge-triggered flip-flop (DETFF), an edge-triggered D-flip-flop (ETFF), and of a 2-to-1 multiplexer (MUX), as shown in Figure 4-5.

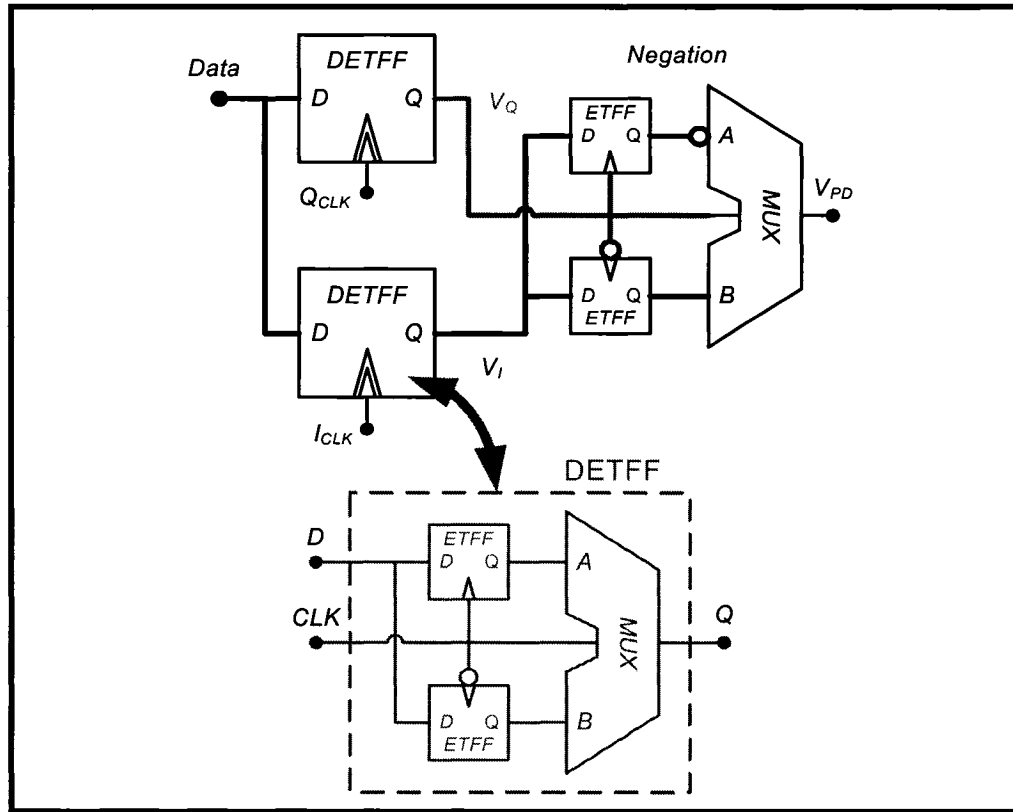


Figure 4-5 Schematic block diagram of the implemented half-rate phase detector by J. Savoj and B. Razavi – DETFF stands for double edge-triggered D-flip-flop; MUX stands for multiplexer.

Although not indicated on Figure 4-5, this system is designed entirely with fully differential signals. Furthermore, all gates such as the D-flip-flops, MUX, buffers, etc. are constructed in MOS current-mode logic (MCML) in order to significantly increase the switching speeds compared to traditional CMOS logic designs. In addition, differential signals mitigate the need for adding inverters – inverters often introduce clock synchronization problems in the form of clock skews and delays, an issue especially critical for high-speed applications.

As shown in the timing diagrams in Figure 4-6 and Figure 4-7, this PD takes samples with a DETFF at  $S_{rise}$  and  $S_{fall}$  about the data edges. Using quadrature phase to sample the PRBS, the  $V_I$  and  $V_Q$  replicate the input signal at  $90^\circ$  apart. When the clock is early, the

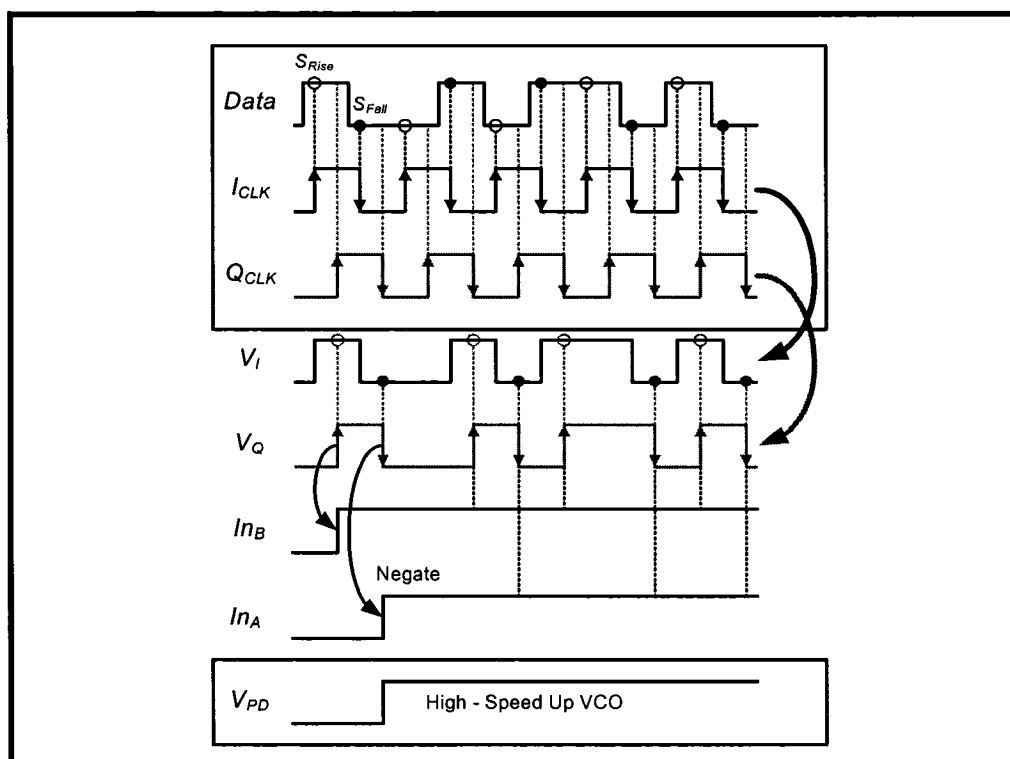


Figure 4-6 Timing diagram of the half-rate DTTL phase detector for a late clock.

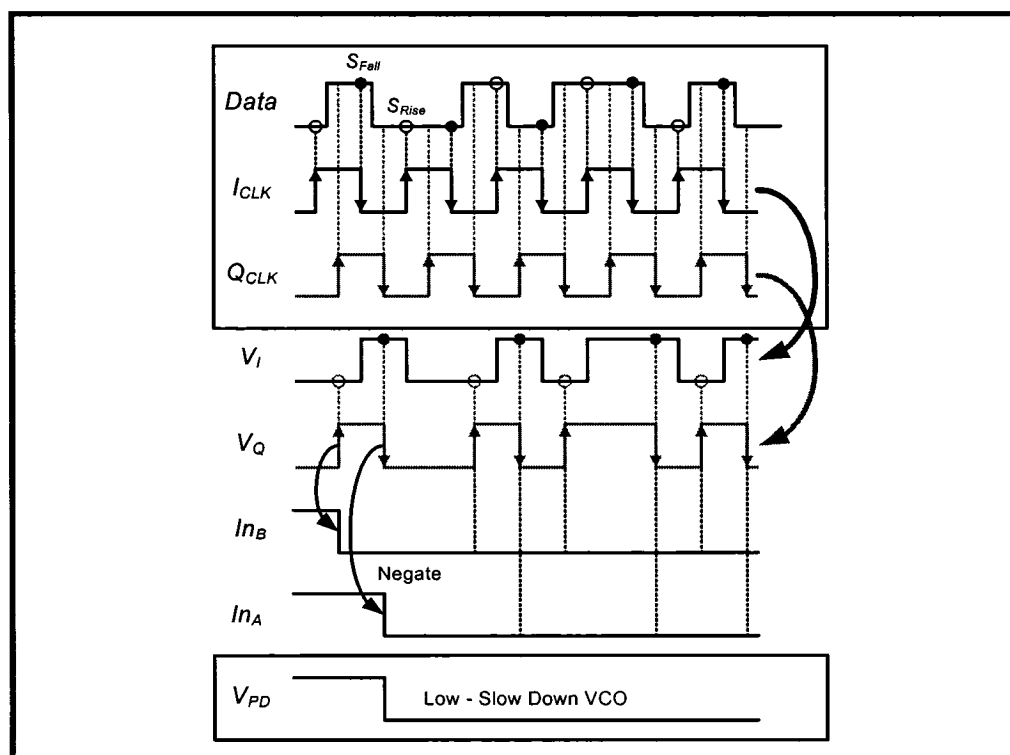


Figure 4-7 Timing diagram of the Half-Rate DTTL phase detector for an early clock.

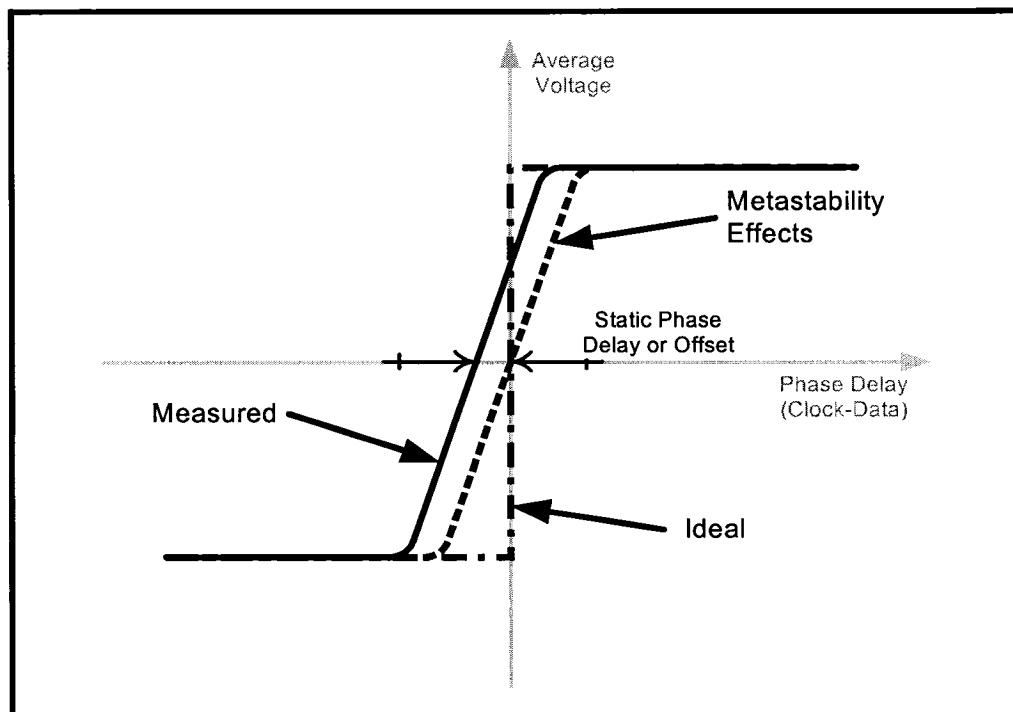
pattern at  $V_Q$  leads  $V_I$ 's. Conversely, the pattern at  $V_I$  leads  $V_Q$ 's when the clock is late. In order to represent the early/late detection by a binary high/low signal, the  $V_I$  and  $V_Q$  signals are fed into a circuit composed of ETFF and MUX. This part of the circuit determines which pattern (at  $V_I$  or  $V_Q$ ) leads the other. As shown in the timing diagram, the early/late detections are uniquely identified by a high or low signal that controls the VCO to slow down or to speed up.

The half-rate binary PD phase versus voltage characteristic is shown in Figure 4-8. Ideally, the PD should have a sharp step at the zero phase difference point. However, in reality, non-linear effects come into play at phase differences near zero. At near phased-locked conditions, the PD is in a metastable state, therefore producing a pulsating early-late beat signal. The V/I converter, as well as the loop, filter averages the early-late signal and follows a linear transfer characteristic near the phase-locked region of the curve. In this small region of the transfer curve, the PD can be modeled and analyzed as a linear PD for simplicity. The presence of a static phase shift is caused by the path difference delay between the input and the clock. For clock recovery purposes, the static phase shift has no effect on performance.

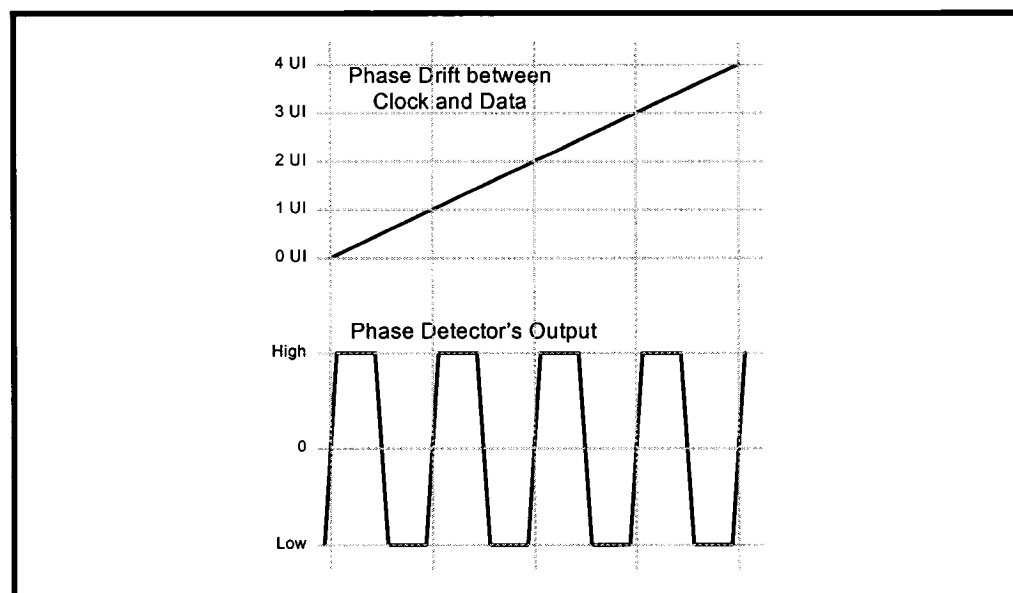
When not in frequency lock, i.e.  $T_{clk} \neq T_{data}$ , the output looks like a periodic square wave. This can be explained by the constant phase drifting between clock and data. Figure 4-9 illustrates the output phase behavior when out of frequency lock. The constant drifting of the clock produces a beat signal at the output of the PD, half the time late or early with respect to the data input stream. This signal can be used later on for frequency detection.

## 2.5 Frequency Detector

As mentioned in the previous sections, an ordinary frequency detector (FD) cannot lock onto a PRBS signal due to the nature of its spectrum. Depending on the pseudo-random type, the peak frequency can be located near DC or away from its actual frequency. Mathematically, a PRBS can be expressed as a sum of periodic pulses with very long periods. Therefore, by linear superposition, a strong near-DC component is expected to be present in the PRBS spectrum. Hence, an ordinary PLL-based FD or a delay-locked loop



**Figure 4-8** A comparison of a non-idealized and ideal transfer function of a half-rate early-late phase detector.



**Figure 4-9** Beat signal at the output of the half-rate phase detector when the frequency is not locked due to the constant phase drifting between clock and data. One unit interval (UI) is equivalent to a 1 bit period.

(DLL) will have the tendency of tracking a data rate lower than that of the actual input bit-stream (Refer to Chapter 2 Section 2).



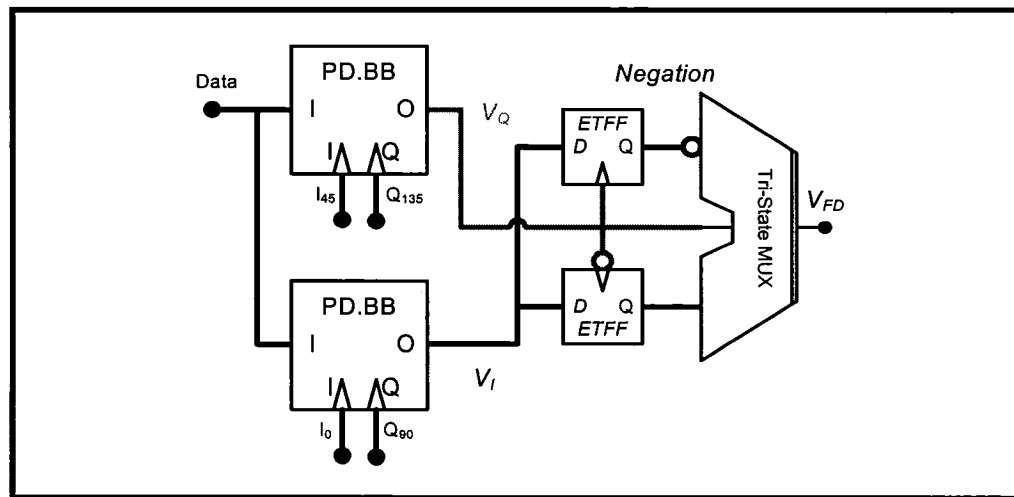


Figure 4-10 A block schematic diagram of the half-rate frequency detector. the PD.BB block is the half-rate early-late (or bang-bang) phase detector.

The binary half-rate FD is based on the digital implementation of quadri-correlation [32]. The principle of quadri-correlation is to make use of the beat waveform of the PD when the data rate is different from the clock frequency. This beat signal can be generated with quadrature and half quadrature clock signals. The fact that one quadrature beat leads the other determines whether the rate is faster or slower with respect to the clock frequency. As we recall from Figure 4-8, the half-rate PD produces a beat frequency proportional to the rate difference.

The half-rate FD is implemented using two PDs in half-quadrature in order to compare the frequencies. As illustrated in Figure 4-11 and Figure 4-12, by determining which PD output leads or lags the other, the FD can determine whether the clock is fast or slow. However, this requires that the clock generates 4 phases at half-quadrature, i.e.  $45^\circ$  apart.

## 2.6 The DTTL Phase & Frequency Detector

The following challenges must be addressed in order to efficiently design a phase & frequency detector:

- The operation of the frequency detector must not interfere with the phase detector.
- The hand over of the FD to the PD, and vice versa, must be done without interruption.

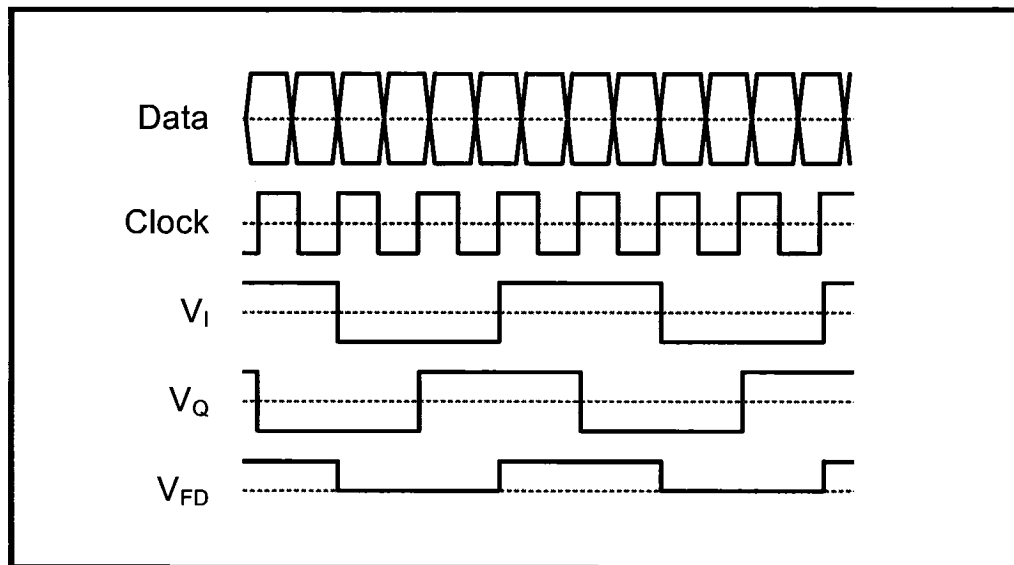


Figure 4-11 Timing diagram of the half-rate DTTL frequency detector for a fast clock in the presence of a coherent input data stream.

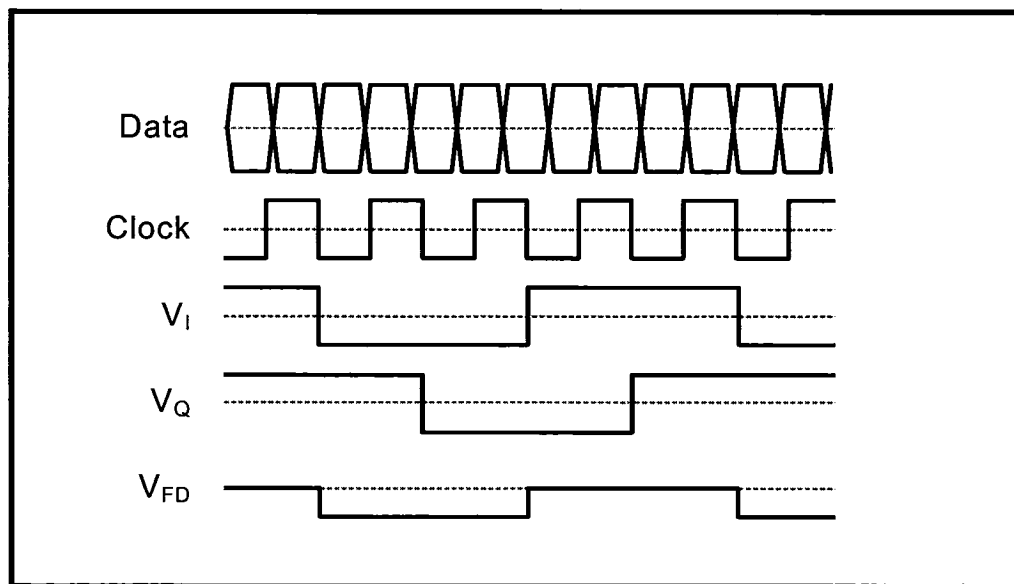


Figure 4-12 Timing diagram of the half-rate DTTL frequency detector for a slow clock in the presence of a coherent input data stream

The advantage of using a binary early-late PD is that it can be seamlessly combined with the FD. Unlike most CDR designs, this system does not require an external reference frequency. Both the PD and the FD can be combined in a dual-loop with shared VCO control, as shown in Figure 4-13. In this case, the subtraction of  $V_{FD}$  and  $V_{PD}$  contains the phase detection signal with the aided acquisition from the frequency detector. Hence the signal  $V_{FD} - V_{PD}$  uniquely identifies whether the clock ought to speed up or slow

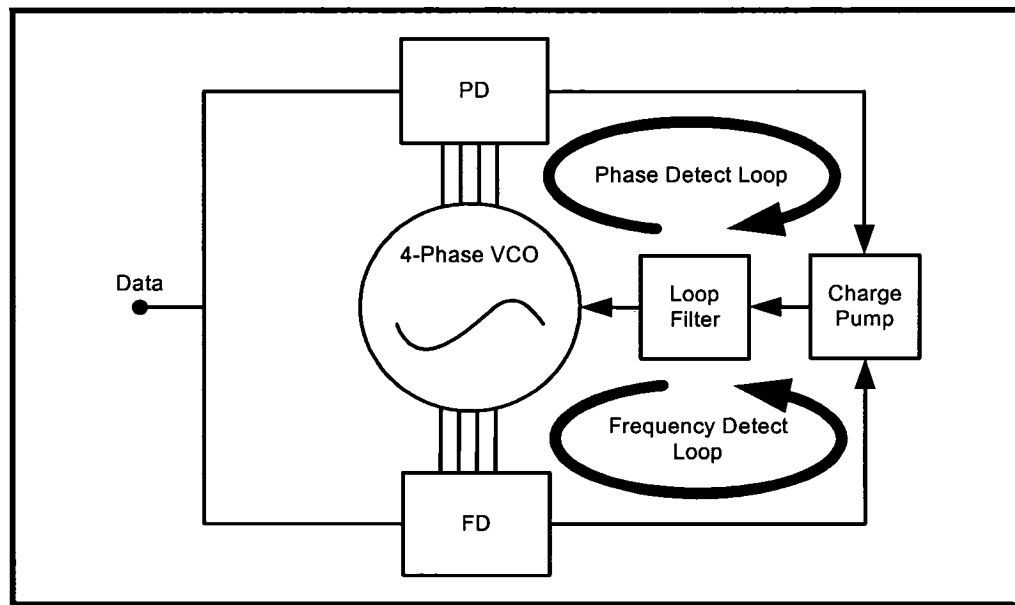


Figure 4-13 System block diagram of the dual-loop with shared VCO control architecture.

down.

### 3 CHARGE PUMP

The charge pump consists a voltage-to-current converter, taking the PFD signal differentially and outputting a single-ended push-pull current. This circuit generates control signals to the VCO which are dampened or filtered out first by a loop-filter.

The charge pump or V/I converter subtracts two differential signals, namely  $VPD$  and  $VFD$  (the phase and frequency detection signal respectively). The possible output states of the V/I converter is to push current, to pull current, or to maintain idle (quiescent) current. One of the challenges in designing this converter is to maintain a near zero quiescent current for the entire dynamic range. This in effect will require special attention for the design of the VCO's tuning circuitry. The tail current of the charge pump can be adjusted externally in order to fine tune the performance and stability of the closed-loop CDR. Figure 4-14 shows a simplified circuit-level schematic of the charge pump.

The dynamic range of this circuit is limited, especially at the desired biased supply voltage of 1.5V. One method of measuring the dynamic range is to observe the leaking

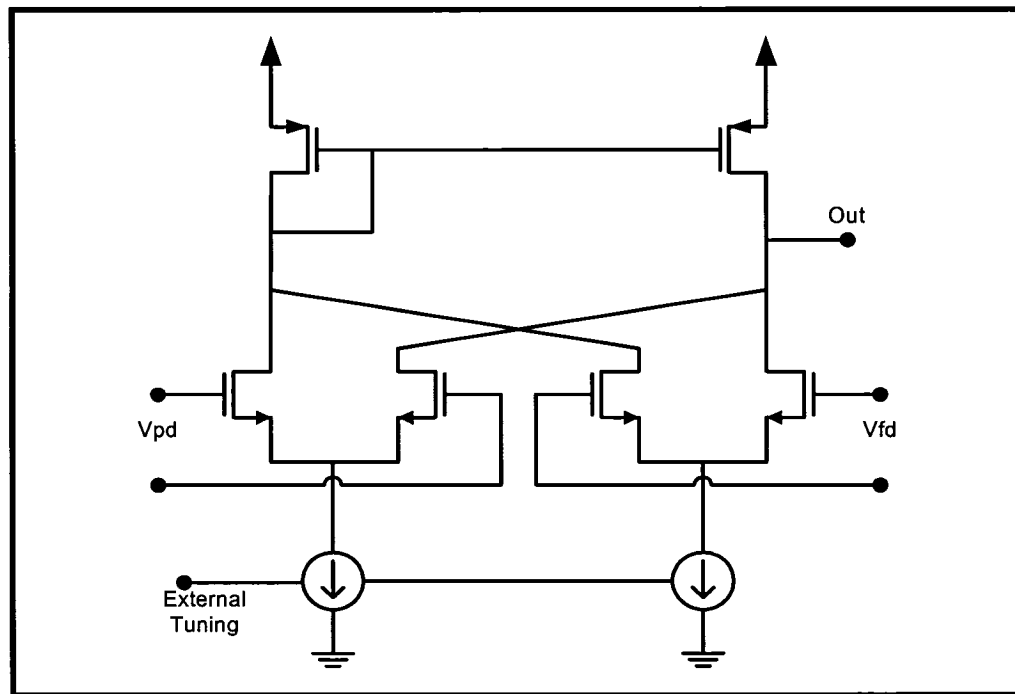


Figure 4-14 Tri-state charge pump schematic diagram with a fully differential input.

current versus the output DC voltage. An allowable leakage current of  $25\ \mu\text{A}$  is chosen; as shown in Figure 4-15, the corresponding dynamic range of the charge pump is limited. It is important to note that the optimal quiescent current is located at approximately  $0.9\text{V}$ , with a tail current of  $125\ \mu\text{A}$ . The VCO's control voltage must coincide with the near-zero quiescent current when in acquisition mode. Otherwise, the leaking current may cause the VCO or the ring oscillator to drift and introduce more jitter at the recovered clock and data.

#### 4 CURRENT-STARVED RING OSCILLATOR

For this half-rate system, a 5-GHz fully differential half-quadrature voltage-controlled ring oscillator is designed. In order to properly operate with the DTTL PFD, the VCO must produce 4 phases of half-quadrature clock signals. With the available CMOS  $0.18\text{-}\mu\text{m}$  technology, the gates are fast enough to oscillate at the desired frequency using 4 inverter stages. Figure 4-16 shows the schematics details of the current-starved ring oscillator and its voltage control circuitry.

High-quality capacitors at the output of each node are added in order to slow down the

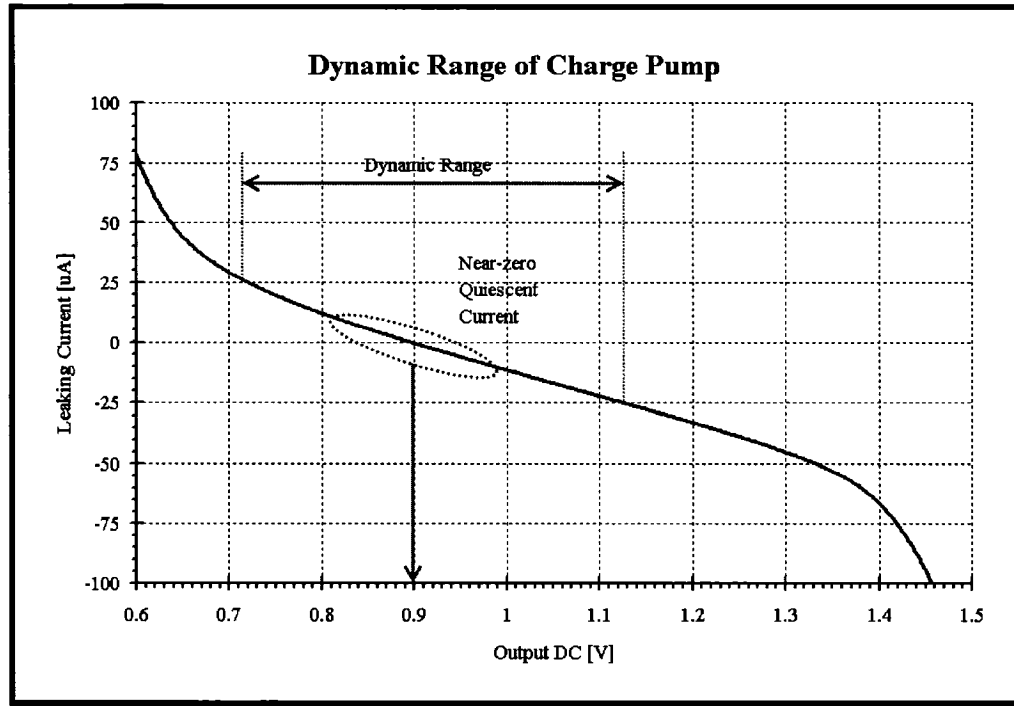


Figure 4-15 A plot of leaking current vs. the output DC voltage with a near-zero quiescent current and the corresponding dynamic range of the differential charge pump for a  $\pm 25 \mu\text{A}$  allowable leakage current.

oscillations within the 5-GHz frequency range. By inserting these fixed capacitors, the effect of the temperature dependent parasitic capacitances of the MOSFET can be minimized. Furthermore, recent publications indicate that the number of stages negligibly affect the overall jitter [33]. Equation 4-1 below shows an inverse proportionality relationship between figure of merit  $\kappa$  and the with tail current  $I_{bias}$  and  $R_D$  ([33] & [25]).

$$\kappa_{total} \propto \sqrt{\frac{2}{\ln(2)} \frac{KT}{I_{bias}^2 \cdot R_D} + \frac{1}{2\ln(2)} \frac{i_{noisetail}^2}{I_{bias}^2} + \frac{1}{2} \left( \frac{K_{VCO}}{\omega_0} n(VCO) \right)^2} \quad (4-1)$$

$$\kappa_{total} \cdot \sqrt{T_d} = \sigma_t. \quad (4-2)$$

The total figure of merit  $\kappa_{total}$  is expressed as the root of squared sum of the following effects respectively: the thermal noise, the tail current  $I_{bias}$ , and the noise at the loop filter (tuning port), where  $I_{bias}$  is the DC tail current,  $R_D$  is the drain resistance,  $i_{noisetail}$  is the total equivalent noise at the tail current,  $K_{VCO}$  is the gain of the VCO in rad/V, and

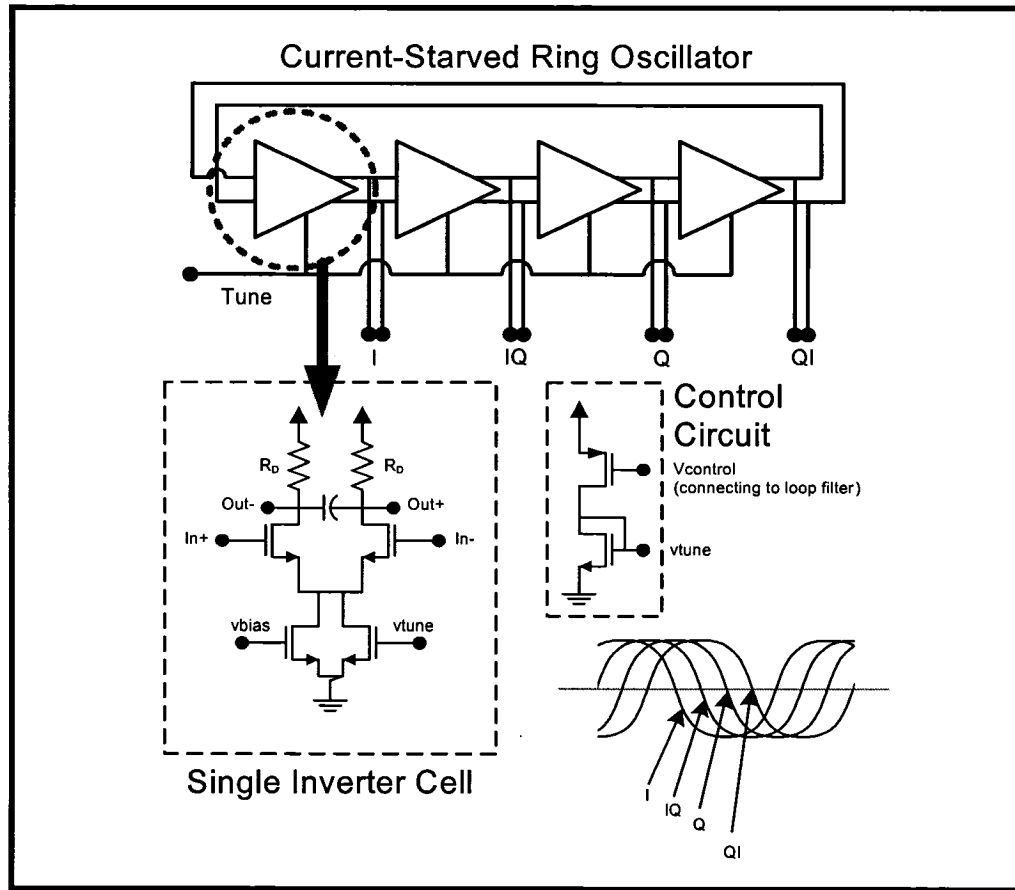


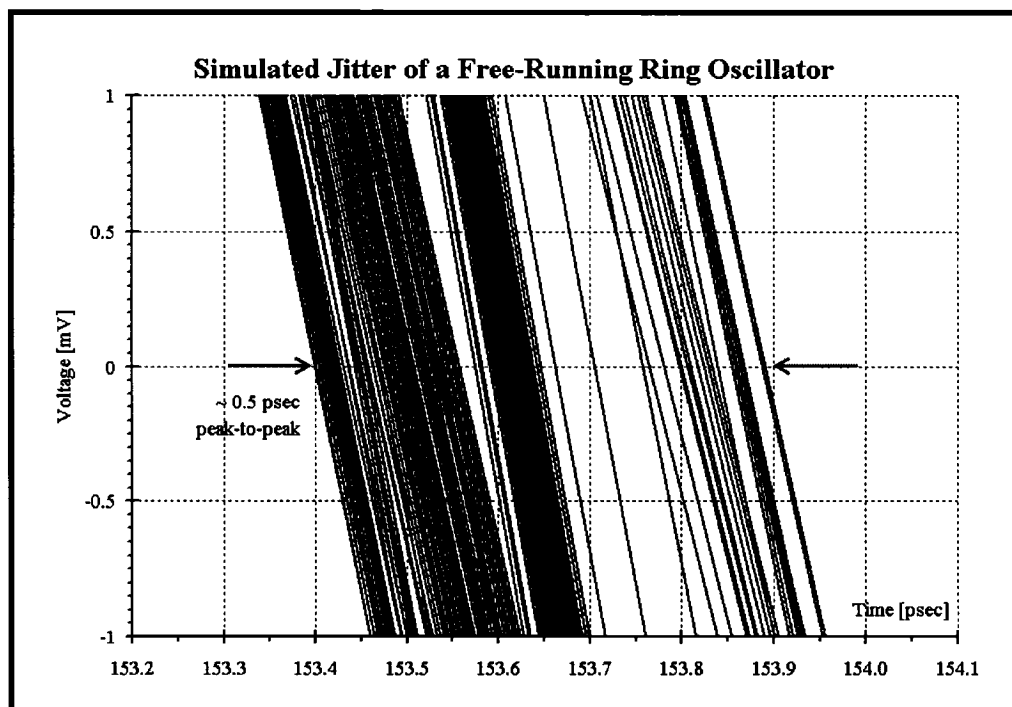
Figure 4-16 Structure of a fully differential current-starved ring oscillator and the voltage control circuit.

$n(VCO)$  is the total injected noise at the VCO control port. A more tangible result is the root mean squared jitter  $\sigma_t$  expressed in Equation 4-2, also referred to as the standard deviation jitter. This measure is expressed in terms of the figure of merit  $\kappa$  and measurement interval  $T_d$ . Clearly,  $\kappa_{total}$  can be minimized by the following methods:

- The use of large tail currents and large drain resistors in order to reduce the thermal noise contribution to jitter: this in effect translates to large gain and peak-to-peak voltage swings (first RSS term in Equation 4-1).
- Reducing the tail current noise (mostly flicker noise) in order to minimize the second term in Equation 4-1: large transistors and high bias currents help reduce flicker noise in CMOS transistors [34]. Flicker noise is usually the dominant source of noise in CMOS devices. Significant impact can be made by reducing this source of noise in CMOS circuits as expressed in Equation 4-1.

- Use of reasonably low VCO gain and a small loop bandwidth in order to minimize the overall noise (assuming white noise at the tuning port). This will in effect minimize the third term of Equation 4-1.

The simulated open-loop jitter results are compiled in Microsoft Excel and displayed in Figure 4-17 (please refer to the section entitled “Abbreviations” on page 109 in the Appendix for the procedure on how to create such a graph). In order to take into account the worst-case jitter, the output of the oscillator was fed to the input of a divider. This takes into account the jitter introduced by the divider circuit as well. The figure is generated from the free-running VCO operating at approximately 2.475 GHz, halved from the 5 GHz clock signal generated by the VCO.



**Figure 4-17 Simulated jitter of the free-running voltage controlled ring oscillator about a zero-crossing voltage level.**

A typical VCO has a linear frequency versus control voltage within the frequency range of interest. Since the charge pump's dynamic range is smaller than that of the VCO, the VCO must be designed in such a way that the desired frequency is within the linear control voltage range of the charge pump. By changing the tail current and the capacitance, the curve can be shifted side ways as well as up or down. This technique, as illus-

trated in Figure 4-18, allows the positioning of the VCO's tuning range to coincide with the dynamic range of the charge pump.

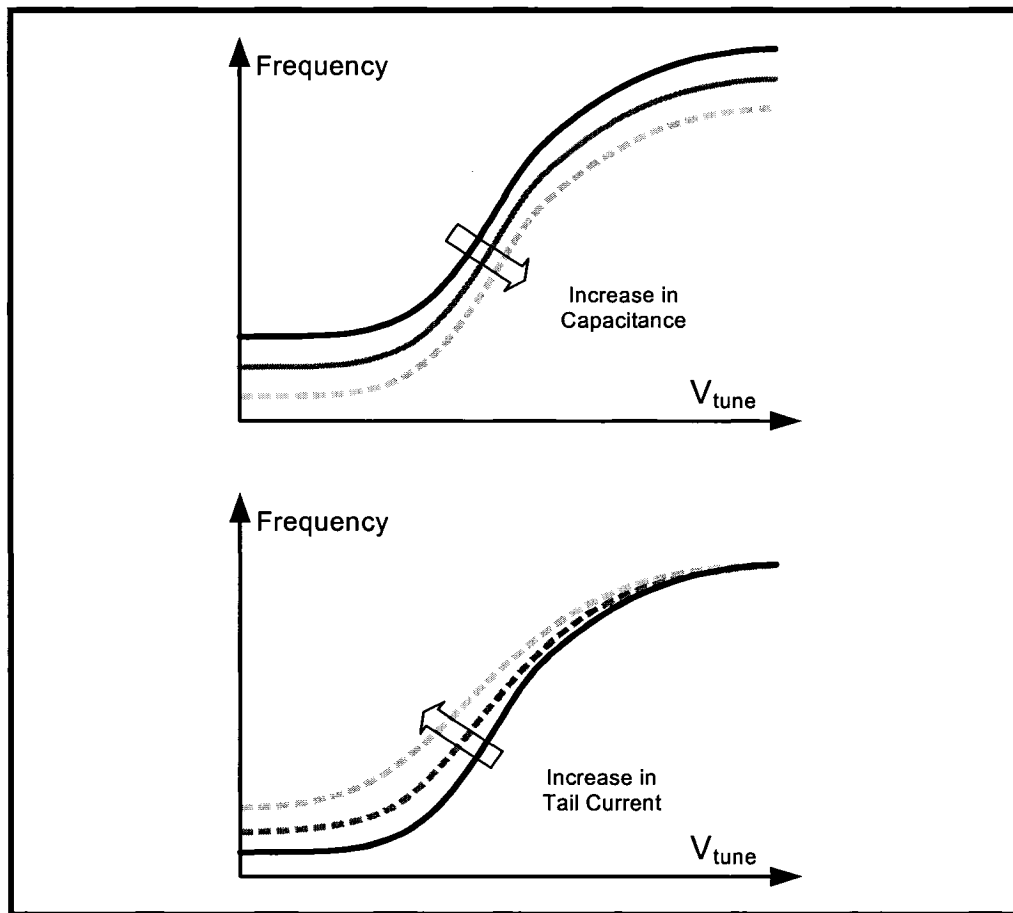


Figure 4-18 Effects of tail current and total capacitance on the VCO transfer characteristic

The extracted simulation of the ring oscillator's frequency transfer characteristic is shown in Figure 4-19. Each curve represents the transfer curve of the ring oscillator for a specified reference tail current. Although the measurement results may be different from simulation, the oscillator may be tuned externally to obtain the optimum range.

## 5 FOUR-PHASE FREQUENCY DIVIDER

In order to design a fail-safe prototype, a 4-phase frequency divider was implemented to run at lower bit rates. Hence, rates of approximately 10Gbps, 5Gbps, and 2.5Gbps can be accommodated by simply changing the rates through external switches.



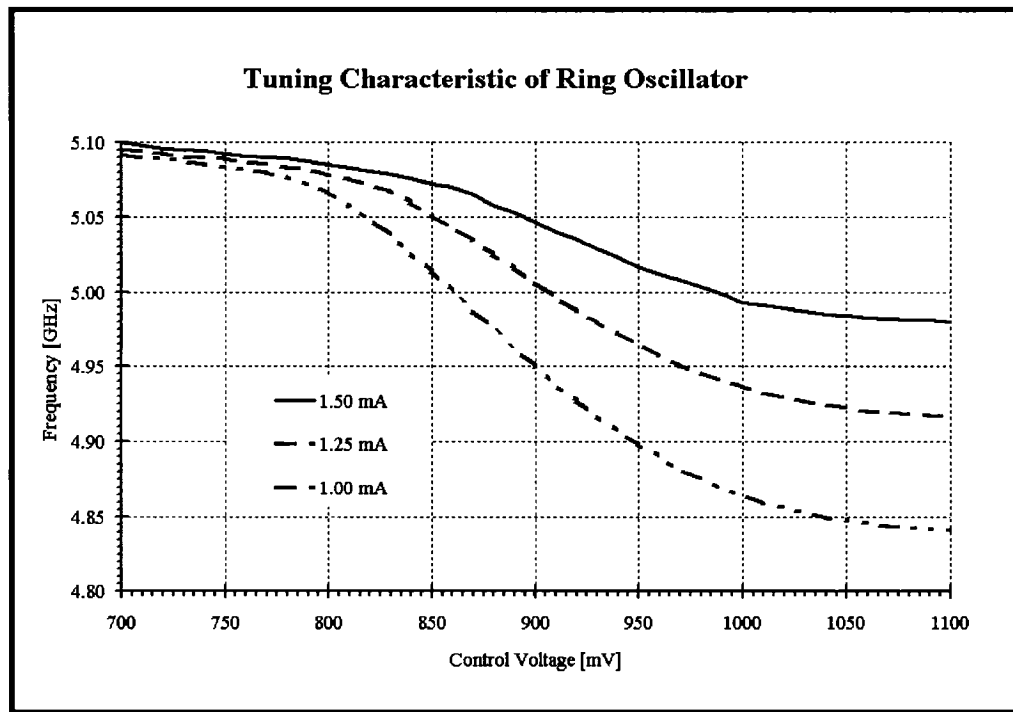


Figure 4-19 Extracted simulations of frequency versus control voltage of the implemented voltage controlled ring oscillator at 35°C for various tail currents.

There are two major concerns when designing a high-speed frequency divider.

- Maintaining equal input and output loading for all phases in order to avoid unnecessary phase-to-phase skew or delay.
- Generating 4 phases, 45° apart, for all 3 possible bit rates.

The generation of all 4 half-quadrature signals (DIV2(I, IQ, Q, QI)) requires the full-rate quadrature signals only, as shown in Figure 4-20. Effectively, the bit rate can be repeatedly halved by cascading these blocks. A network of multiplexed dividers is implemented for prototyping purposes. As shown in Figure 4-21, this half-quadrature divider network can provide half and quarter bit rate with a selectable MUX switch.

By inserting another MUX at the input, the available rates are 5 GHz (10Gbps), 2.5 GHz (5Gbps), and 1.25 GHz (2.5Gbps). Although this prototype can operate at multiple bit rates, it is only optimized for 10Gbps.

It is worthy to note that the memory issues of the divider blocks does not affect the functioning of the CDR. In other words, depending of its initial state, some output clocks

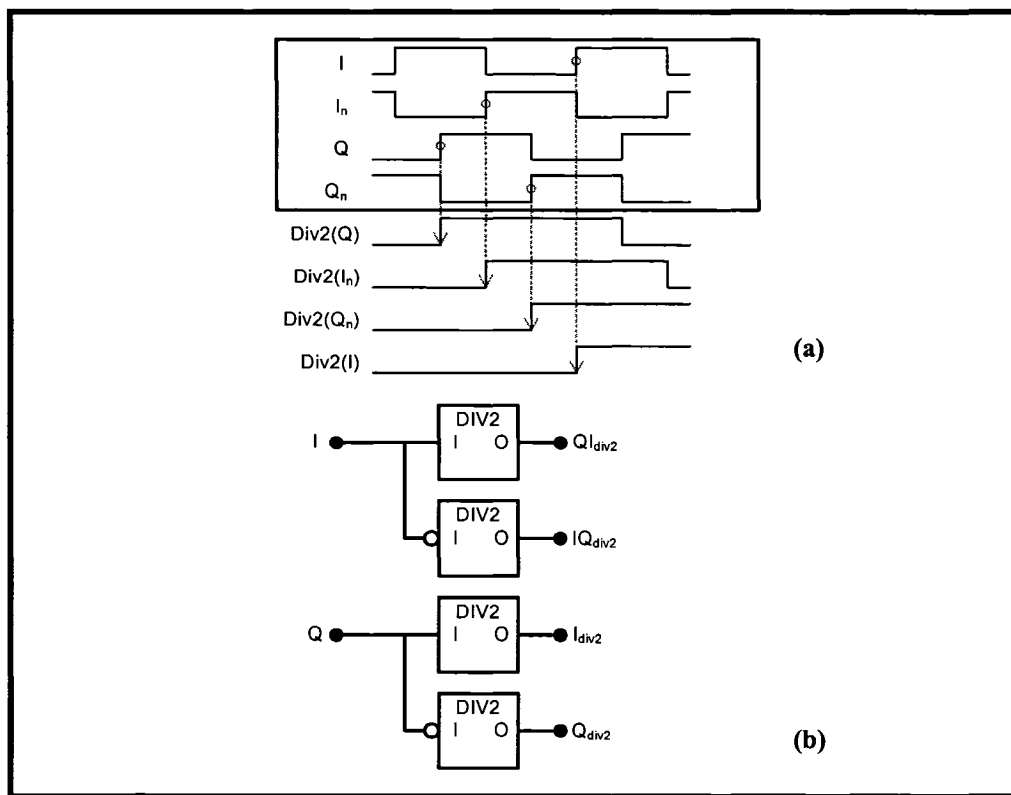


Figure 4-20 (a) Half-quadrature frequency divider and its corresponding waveform. (b) The corresponding schematic block diagram with divide-by-2 blocks  $\text{DIV2}$ .

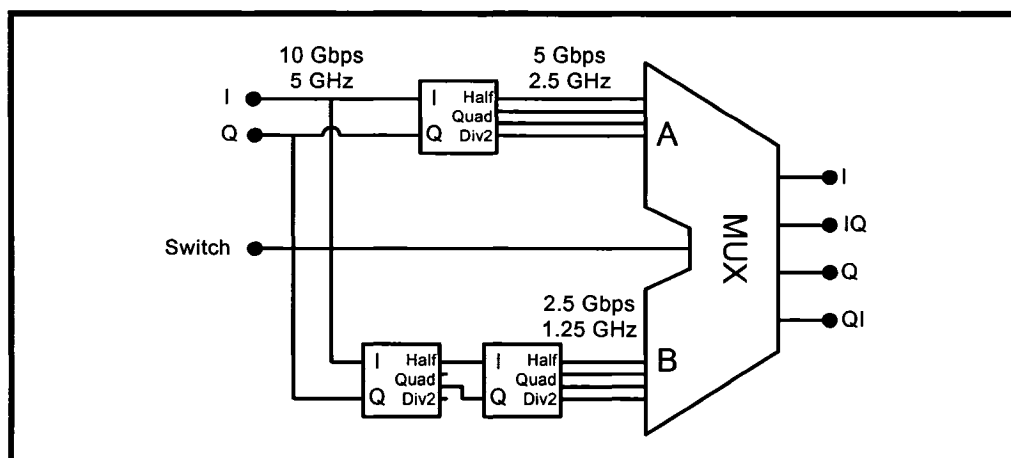


Figure 4-21 Half-quadrature divide-by-2 and divide-by-4 switching circuit.

may be inverted. Because the system is half-rate, regardless of this memory problem, the circuit will still operate on both edges whether inverted or not. Although the divided half-quadrature signals may not appear to look like in Figure 4-20, the edges of the divided signals will be half-quadrature apart. Figure 4-22 shows a more probable outcome of the 4-

phase divide-by-2 network due to memory issues.

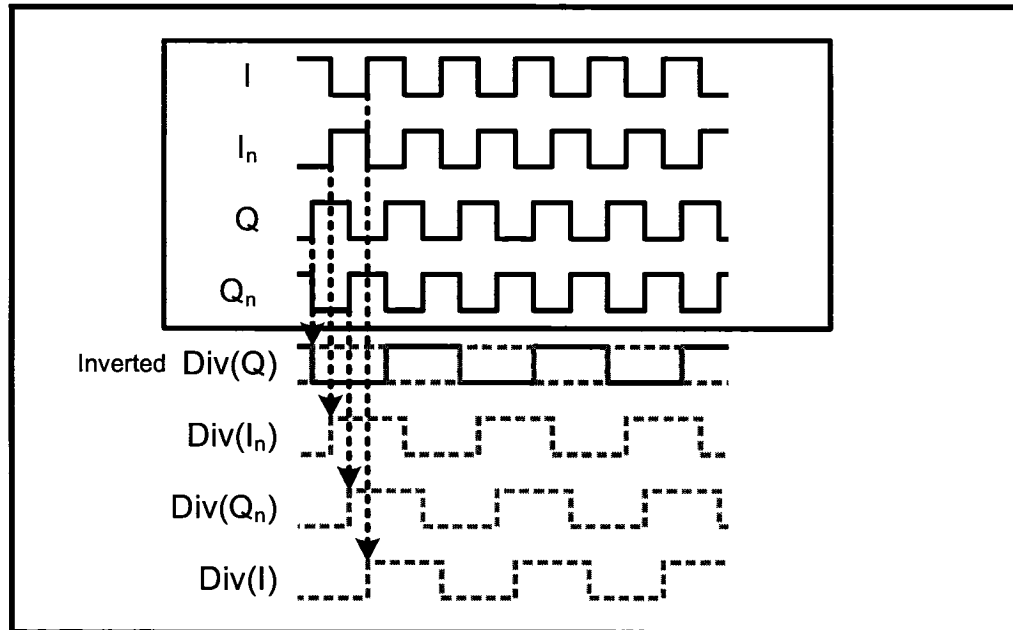


Figure 4-22 Possible output of the 4-phase half quadrature divide-by-2 network due to memory issues of the flip-flops. Each phase can be arbitrarily inverted due to this effect.

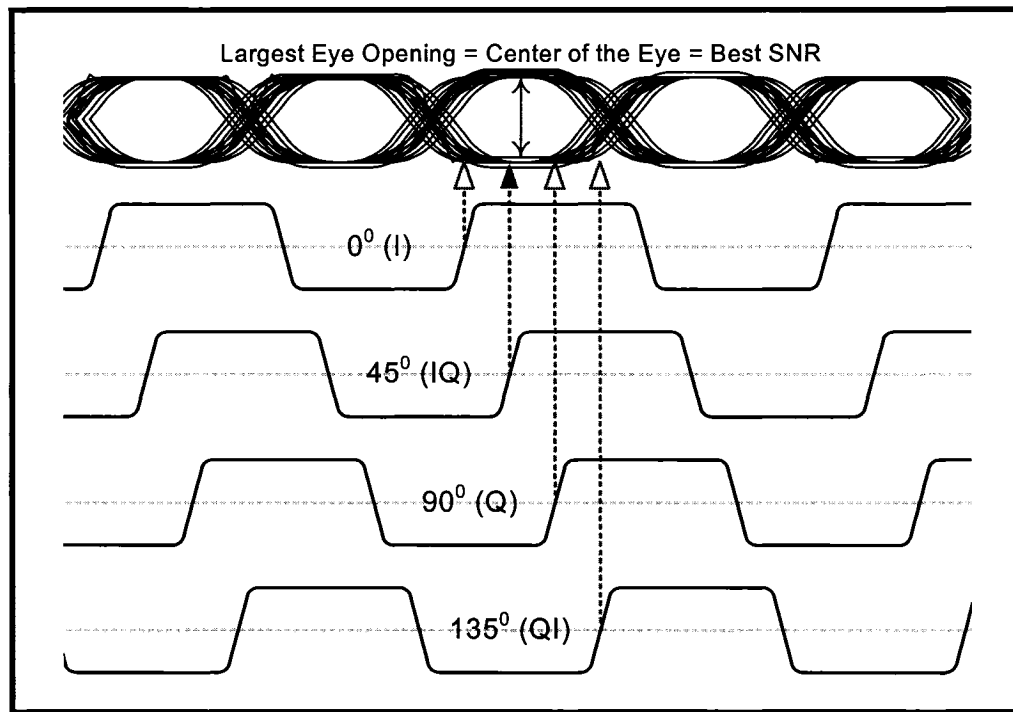
## 6 DECISION CIRCUIT FOR BER OPTIMIZATION

### 6.1 Eye Diagram

The eye diagram is an important plot that shows the quality of the PRBS data. From this figure, one can qualitatively assess noise, jitter, bit error rate (BER), eye opening, etc. The BER is a probability measure expressing a value between zero and one, in which the smaller the number, the lower the error occurrence in the detection of the retimed data. As shown in Figure 4-23, the smallest BER can be obtained by sampling at the largest opening, i.e. the middle of the eye.

### 6.2 BER Optimization

First of all, the noise power must be minimized which is obtained by optimizing the bandwidth of the high-speed gates. By ensuring that most gates operate at a bandwidth that is approximately equivalent to 0.7 times the bit rate, i.e. a 7-GHz bandwidth for a 10-Gbps operation, this method will ensure to obtain the optimal SNR value at the middle of



**Figure 4-23** Simplified drawing of an eye diagram opening, showing the half-quadrature sampling points.

the eye-diagram. ([14] & [15])

A bank of 2 edge detectors is setup in order to determine the relative phase of the data, as shown in Figure 4-24. Making use of a DTTL phase detector, a relation is determined for the best sampling point within  $\pm 1/8$  of a unit interval (UI) or bit period, for instance at a  $\pm 12.5$  psec resolution for a 10-Gbps signal. In order to avoid glitches at the output of the edge detectors, capacitors are added as loads. The effective capacitance can be doubled by placing the capacitor between the two differential output. A 5-pF to 10-pF capacitor suffices to discriminate any sudden change or glitch in the edge detectors.

As shown in Figure 4-24, the same data is fed into the DTTL phase detectors at half-quadrature clock phases apart. At the output, the phase transfer characteristic shows four distinct lettered regions of phase delays – note that a constant phase offset is present between the clock and data due to differences in signal paths. The corresponding lettered regions are identified in Figure 4-25. For each lettered phase delay, the thick double-arrow indicates the nearest clock edge that samples the centre of the eye opening. Given the delays are well conserved, Table 4-2 tabulates the information in Figure 4-25.

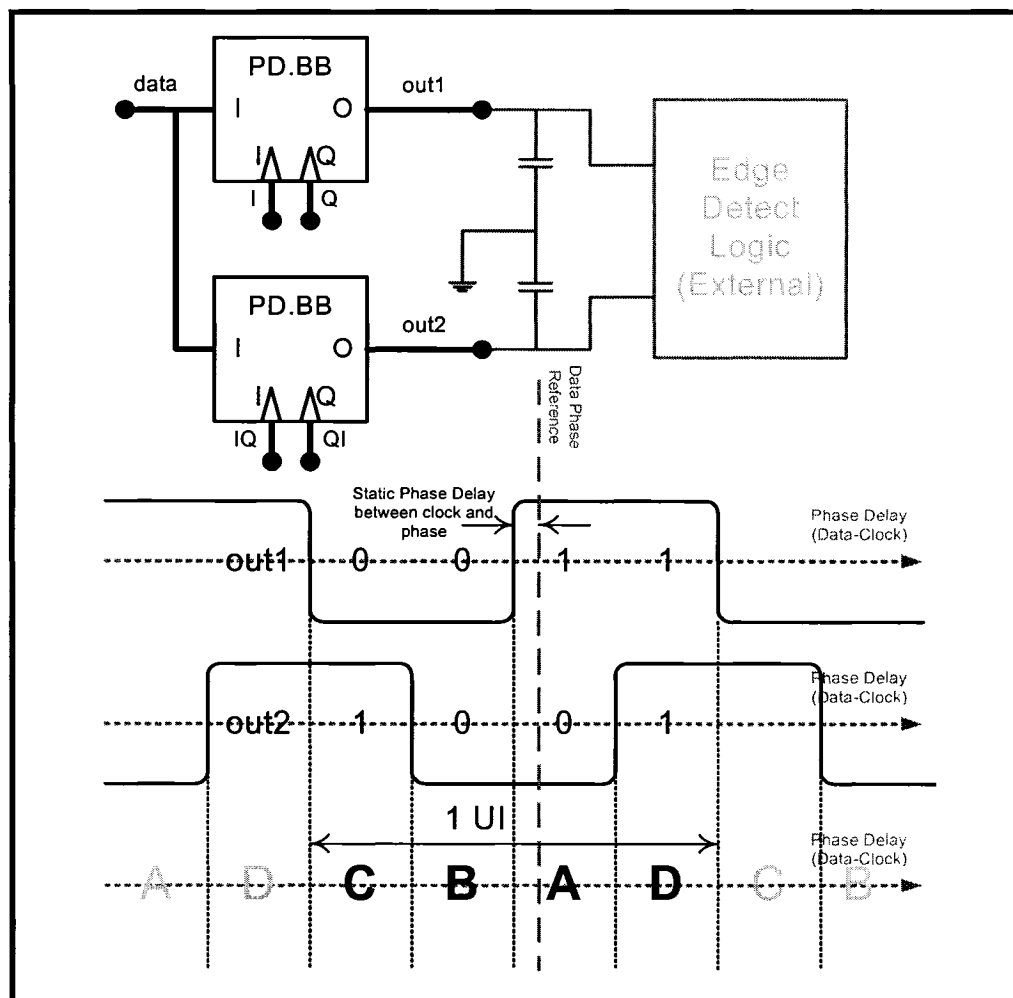


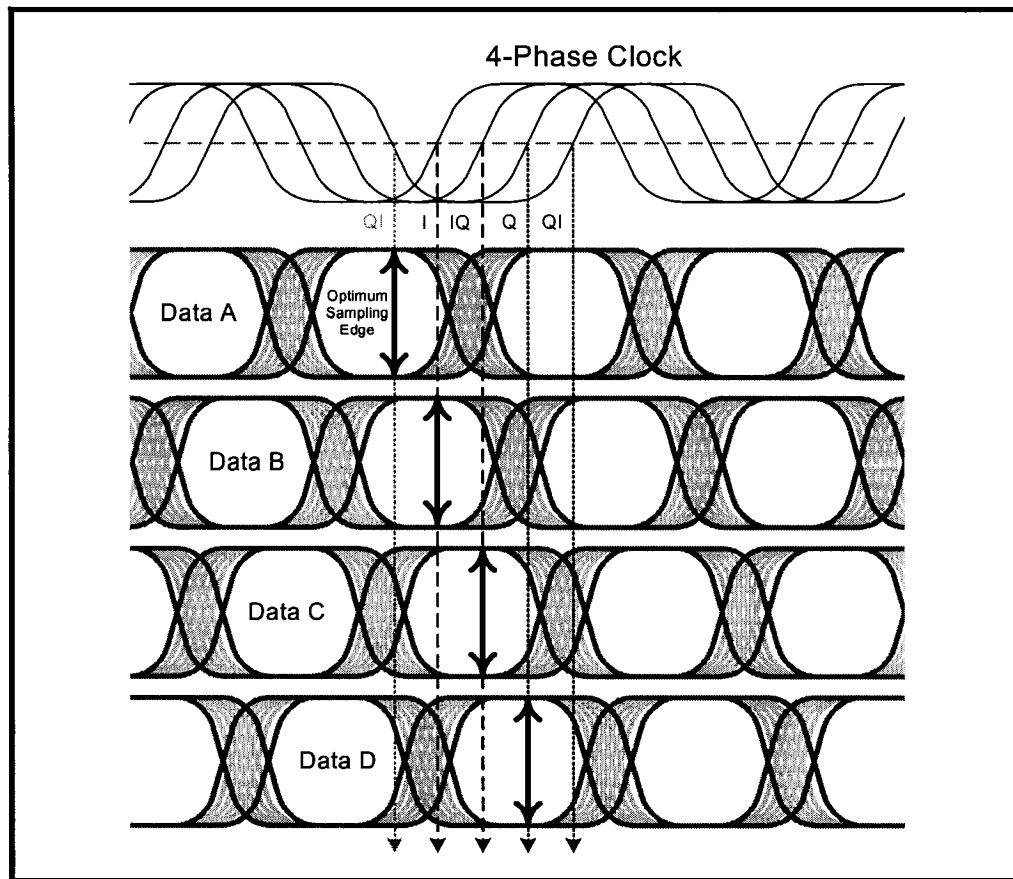
Figure 4-24 Edge detector block diagram and its corresponding voltage-phase transfer characteristic, with 4 distinct phase regions (A, B, C, and D) at 10Gbps.

### 6.3 Decision Circuit Bank

The retiming/decision circuit is mainly a bank of DETFF with selectable outputs. Figure 4-26 shows the schematic block diagram of the retiming circuit. As a first hand observation, by connecting  $S0$  to  $out1$  (from Figure 4-24) and  $S1$  to  $out2$  (from Figure 4-24), respectively, the system automatically selects the best-BER retimed data branch to the output. However, an edge-detect logic circuit is recommended to further stabilize the operation. Although not indicated on the figure, the entire circuit is fully-differential.

## 7 OUTPUT DRIVERS

Differential output buffers are required to drive signals to a 50- $\Omega$  probe. At the



**Figure 4-25** Positioning of the data eye diagrams A, B, C, and D with respect to the output of the edge detector logic circuit, and the corresponding best-BER sampling phase.

**Table 4-2** Data and optimal sampling point for the output of the edge detector bank (Figure 4-24 & Figure 4-25)

PD ( $I_{45}$ $Q_{135}$ )	PD ( $I_0$ $Q_{90}$ )	Corresponding Lettered Region	Phase Corresponding to Optimal Sampling or Center of the Eye
0	0	B	$I(0^\circ)$
0	1	C	$IQ(45^\circ)$
1	0	A	$QI(135^\circ)$
1	1	D	$Q(90^\circ)$

moment of testing, a large decoupling capacitor is placed between the load and the output of the buffer in order not to disturb the DC operating points. Progressive sizing of each

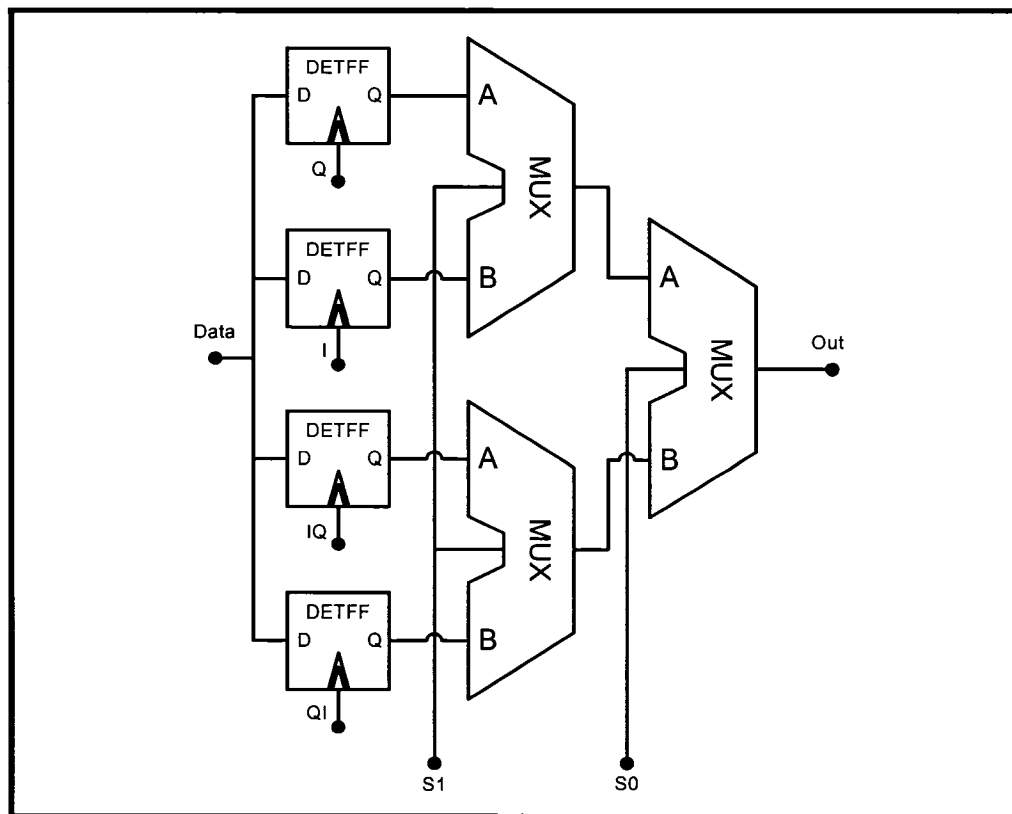


Figure 4-26 A schematic block diagram of the retiming circuit bank with half-quadrature selectable edge triggering.

differential pair stages is required to optimize bandwidth. The gain of the buffer need not be large: approximately 0-dB gain is sufficient to extract the signal without excess noise corruption and to drive the instrumentation devices. Figure 4-27 is the general schematic diagram of the output buffer connected to the external DC decoupled probe.

Several techniques were explored to extend the bandwidth of the output buffer. For instance, inductors were placed in series with the resistors at the drain [35]. Unfortunately, this solution is not always viable due to its large chip area and cost. Other techniques were also considered such as cascoding and  $f_t$  doubling, however they did not increase the overall speed of the circuit. These approaches both attempt at minimizing the capacitive Miller effect, but in fact actually reduce the overall speed due to their added parasitic capacitance and the overall low signal gain.

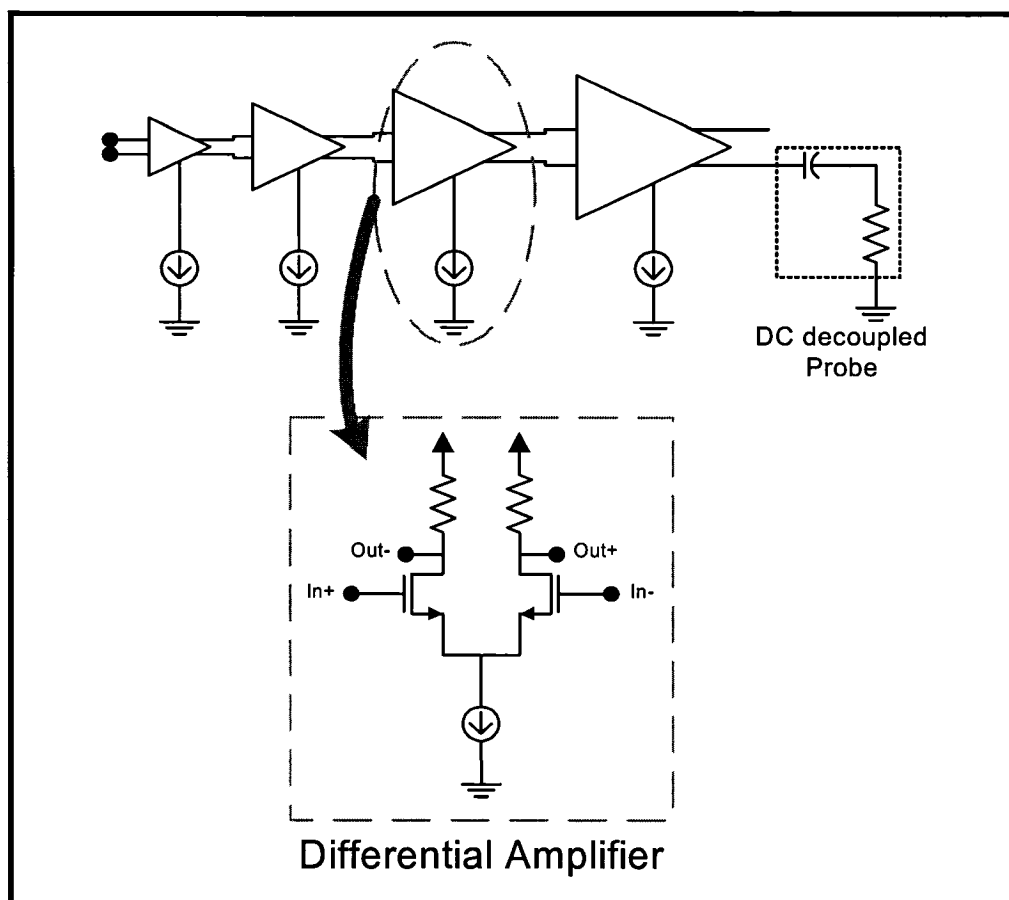


Figure 4-27 Schematic of the high-speed output buffer with an external DC decoupled 50-Ω probe termination.

## 8 SUMMARY

This chapter presented some popular designs of phase/frequency detectors (PFD) circuits. The main focus of this chapter was on characterizing the half-rate binary PFD presented by Savoj & Razavi [11], and on verifying the performance and function of this circuitry. All the essential building blocks of a half-rate binary phase detector CDR were presented and their important design issues discussed in this chapter. The overall system was carefully assembled and is ready for testing and verification. A detailed block diagram of the assembled system was shown at the beginning of this chapter in Figure 4-1.



---

# Chapter 5

System Verification & Testing

---

## 1 IMPLEMENTATION & DESCRIPTION

### 1.1 Prototype Design

Figure 5-1 is a microphotograph of the prototype implemented in TSMC's standard CMOS 0.18-micron technology. The respective blocks of a PLL-based CDR circuit are shown, such as the VCO and the divider network, the phase/frequency detector, the edge detector & the data retimer network, the on-chip loop filter capacitor, and the output drivers. The schematic block diagram of the system was shown in Figure 4-1.

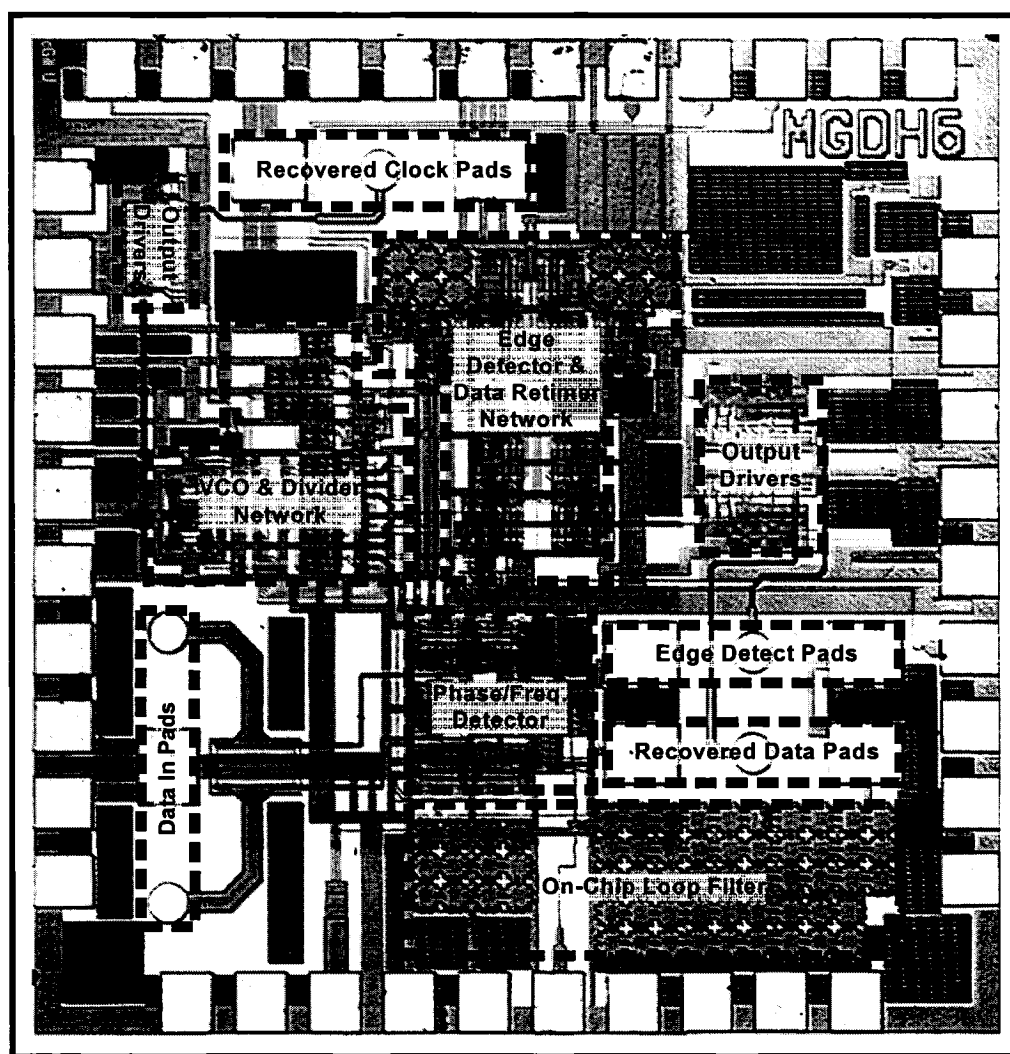
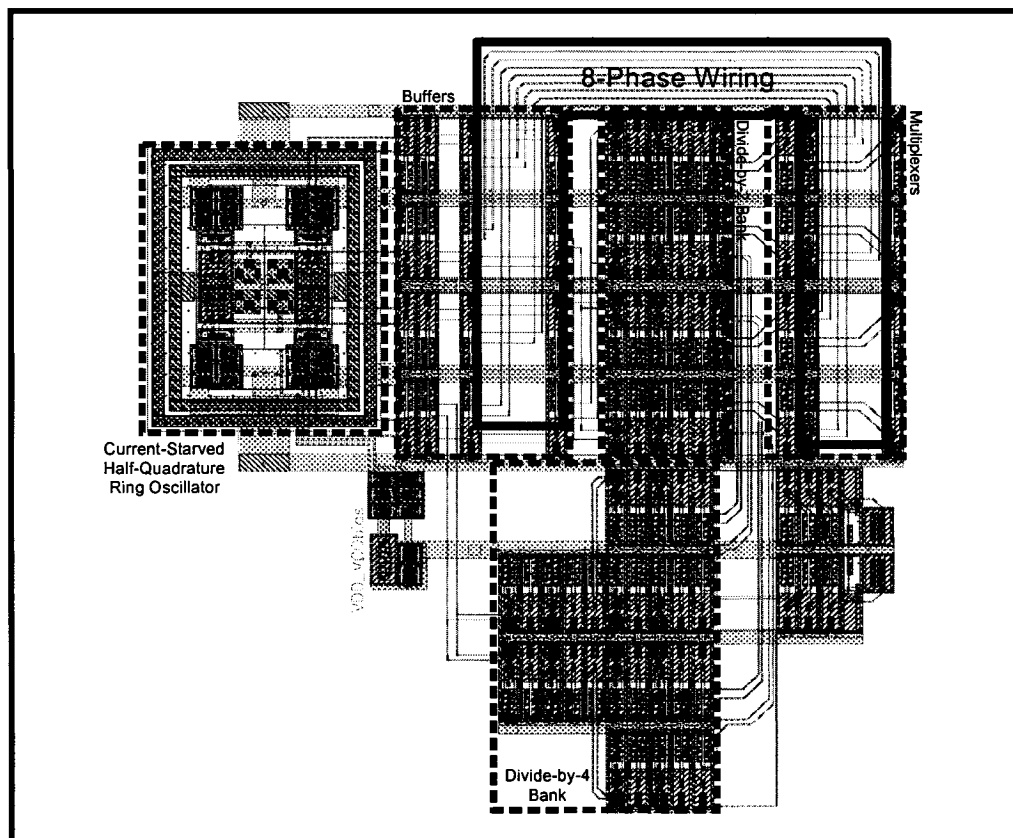


Figure 5-1 Chip microphotograph of a loose die of the implemented CDR system in TSMC's CMOS 0.18-micron technology.

## 1.2 Layout Considerations

### 1.2.1 Minimizing Trace Distances

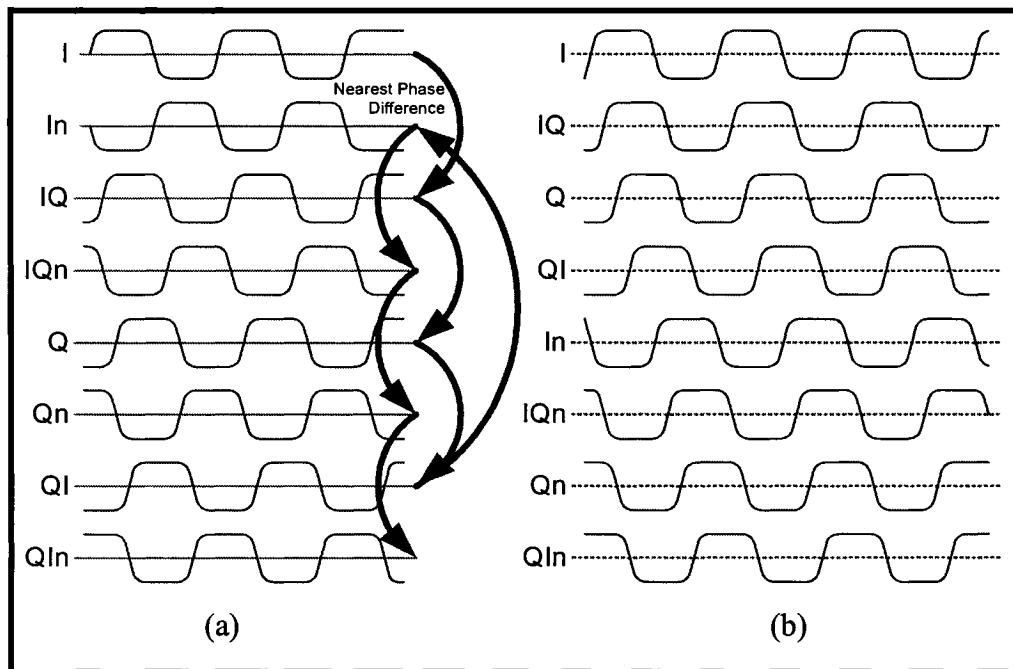
The system implemented runs on the half-quadrature differential clock. Therefore, there are 8 signal wires (4 half-quadrature phases & their compliments) that clock the phase/frequency detector, edge detector, and the data retimer circuit blocks. The clock wires have been laid out carefully by considering the following points (Figure 5-2 is an example of routing the clock signals following the guidelines suggested below):



**Figure 5-2** The half-quadrature ring oscillator with the divider network. This layout required efficient routing of 8 clock signals across the circuit.

- Separate the parallel traces by distancing them apart as much as possible. This will minimize capacitive and inductive coupling. However, far apart traces increase the challenges of making each path of equal length, hence causing clock skews.
- Ensure that each trace travels through the same metal layers and uses the same number of contacts in order to preserve the delay and minimize clock skews.

- Use wider signal strips: although wider striplines may be more capacitive, a wide layout will reduce the inductive effects of long wires. Moreover, transmission lines coupling and inductive effects are often not readily available in circuit simulators, therefore it is preferable to make the wires more capacitive rather than inductive.
- Minimize crossing over other metal layers, and use the upper layers for critically long wires. For instance, in TSMC's CMOS 0.18-micron technology process, metal layer 6 has better conductivity and less substrate coupling.
- Although not implemented in Figure 5-2, the clock signals can be placed adjacent to the nearest phase difference in order to minimize the Miller capacitance effect between differential lines. Because the 4-phase clock wires are fully differential, pairing the clock wires with their compliments will increase the effective capacitance between them. Therefore the wires should be placed adjacent to the nearest phase difference, as illustrated in Figure 5-3.

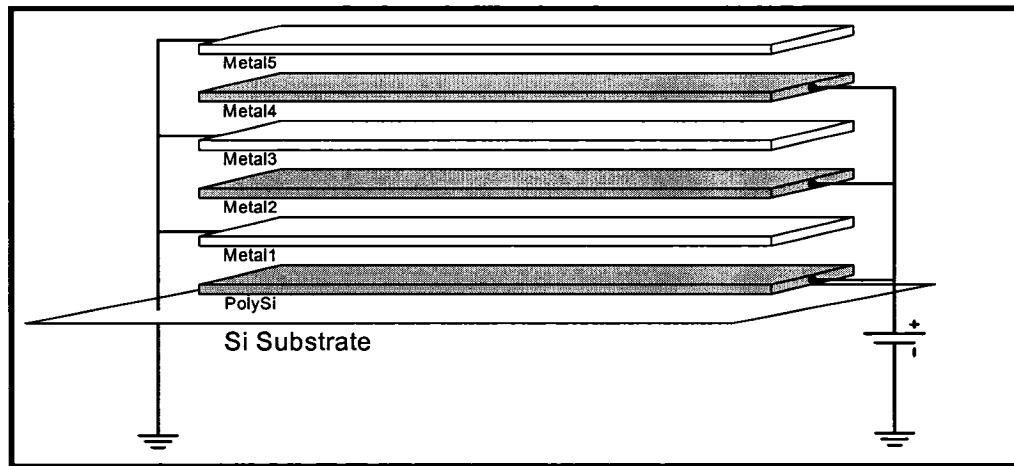


**Figure 5-3** Timing diagrams illustrating the ordering of the 4-phase differential clock wires: (a) ordered by compliment signal pairing and (b) by nearest phase difference. The arrows indicate the nearest phase starting from phase “I.”

### 1.2.2 Supply & Ground Wiring

Current spiking is a common problem, especially in high-speed switching circuits. In

order to mitigate the effect of current spikes, large capacitors can be placed between the voltage supply and ground. This will ensure that the voltage at the supply is more quiet. Instead of adding capacitors to the supply bus, the supply and ground wires can be interdigitated, as shown in Figure 5-4, in order to maximize the capacitance obtained per unit area. This approach has been implemented in the chip shown in Figure 5-1 around the perimeter of the circuit (the supply and ground bus must be able to carry large currents, in order to avoid electromigration). By interdigitating the supply and ground wires, their effective widths are 3 times larger for the same area occupied on the chip.



**Figure 5-4** Interdigitated layered bus for maximum capacitive coupling between the voltage supply and ground. This approach also increases the effective width of the busses by three times, for the same area occupied on chip.

As the design is housed in a package, bonding wires add parasitic inductive effects. The inductive effects of bonding wires must be taken into account for accurate simulation results. Large current spikes  $\Delta i$  can generate ground and supply bouncing  $\Delta v$ , as summarized in Equation 5-1, where  $L_{eff}$  is the effective inductance due to the bonding wires.

$$\Delta v = \Delta i \cdot L_{eff} \quad (5-1)$$

The use of parallel pads for supply and ground reduces the effect of the effective inductance  $L_{eff}$ . However, certain circuit blocks generate larger current spikes than others: for example, the output drivers consume almost one third of the entire power of the system. Certain circuit blocks are more sensitive to supply and ground bouncing than others;

it is therefore important to identify them, to separate the supplies from each other, and to use redundant DC pads in order selectively stabilize the voltage at the supplies.

### 1.2.3 Input & Output Impedance Matching

All instruments are terminated with the nominal  $50\text{-}\Omega$  impedance. In order to record measurements with minimum reflections, maximum power must be transferred to the measurement instruments. Referring to Chapter 4 Section 7, the output impedances of the output buffers are designed for  $50\text{-}\Omega$  loads. Similarly, the inputs to the CDR are also terminated with  $50\text{-}\Omega$  impedances, as illustrated in Figure 5-5. Although resistive load matching may introduce more noise to the system, this method is actually very appropriate and simple for very wide band matching: the input and output signals are PRBS in nature and therefore require wide band matching.

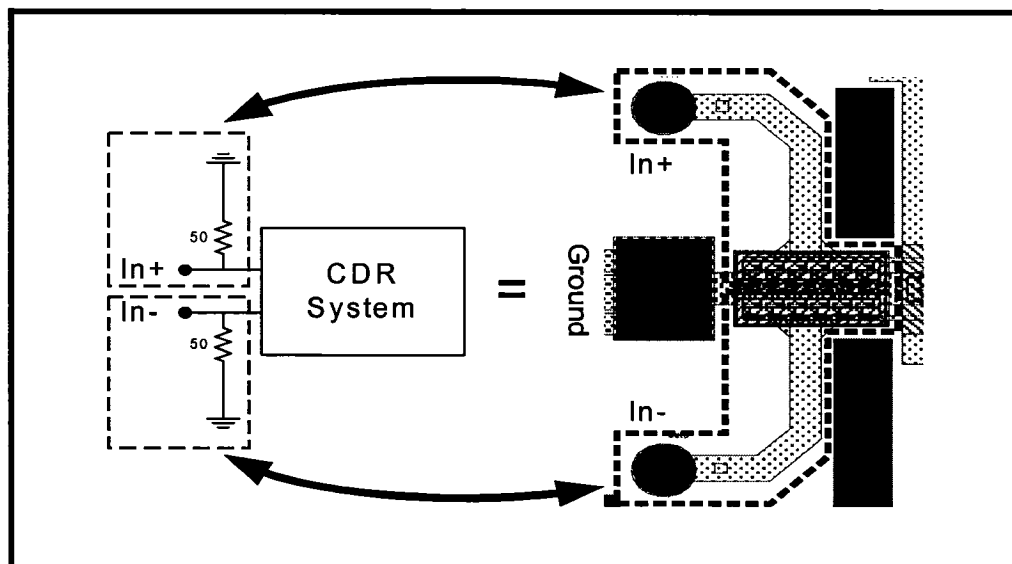
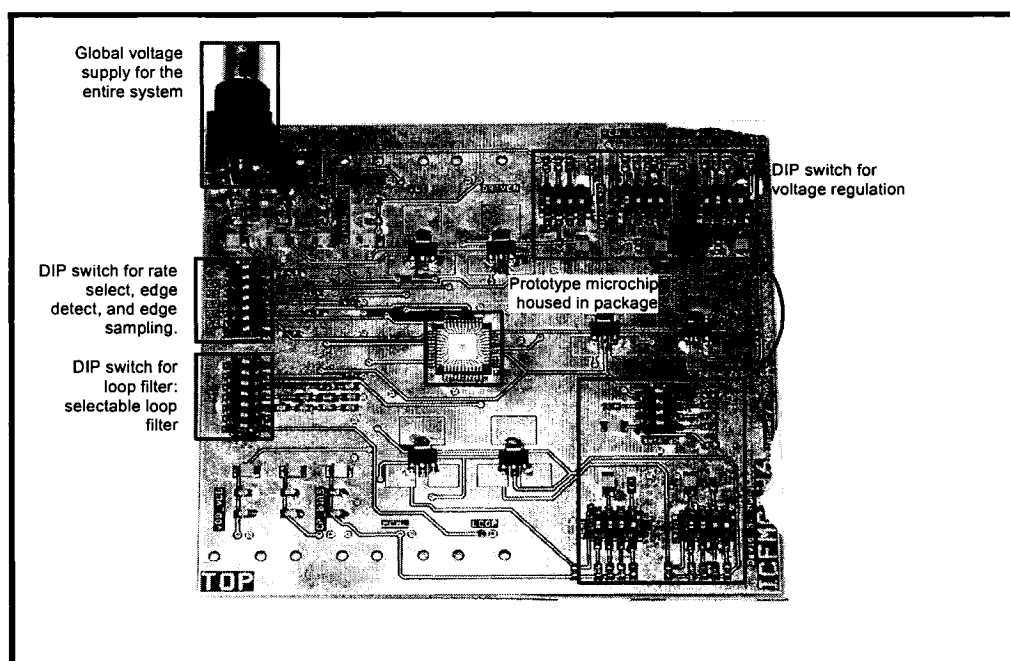


Figure 5-5 Input termination of  $50\text{ }\Omega$ : on the left is the schematic equivalent of the layout diagram (on the right) that shows the input pads for the SGS probe.

### 1.3 Prototype PCB for Testing

A printed circuit board was designed and fabricated for testing the prototype IC, as shown in Figure 5-6. The size of this board is approximately 5 in. by 4.5 in. The system must be accompanied by an appropriate testing PCB in order to interface with the testing equipment. This PCB was custom designed in order to have as much flexibility in pin assignment as possible.



**Figure 5-6 Custom PCB with programmable biasing control, selectable loop filters, and selectable rates.**

This PCB includes elements such as DIP switches and voltage regulators, in order to generate multiple biasing voltages from one supply. Other DIP switches are used for controlling clock rates, data resampling edges, and the loop filter bandwidth. All of these settings are memorized on this PCB, which minimizes the overhead for testing and setting up. This CDR system has 5 external switches routed to a DIP-7 switch located on the top left corner of the PCB in Figure 5-6.

The PCB also provides easy adjustment of biasing voltages. There are 6 biasing voltages that need to be set. In summary, the features of this PCB are described below:

- Instead of having multiple ports for biasing, the programmable voltage regulators with DIP switches keep the bias voltage settings in memory, sharing only one supply.
- All control signals ( $S0$  to  $S4$ ) and loop filter settings are controlled by DIP switches.

- As a fail-safe approach, the PCB also accommodates the use of external tuning for all biasing points, to account for cases when the ranges of the voltage regulators are either too limited or too coarse.

The programmable voltage regulator was designed to supply voltages of 1.4V to 1.8V at 100mV increments through simple thermometer coded DIP switches. This circuit can handle currents as large as 1A. Furthermore, this voltage regulator has very good supply rejection, and thus injects less noise in the circuit [36].

## 2 TESTING SETUP

### 2.1 Test Equipment

Figure 5-7 is a photograph of the test setup used for generating the eye-diagram and to measure the bit-error-rate (BER). There are four essential measurement equipment for testing a CDR:

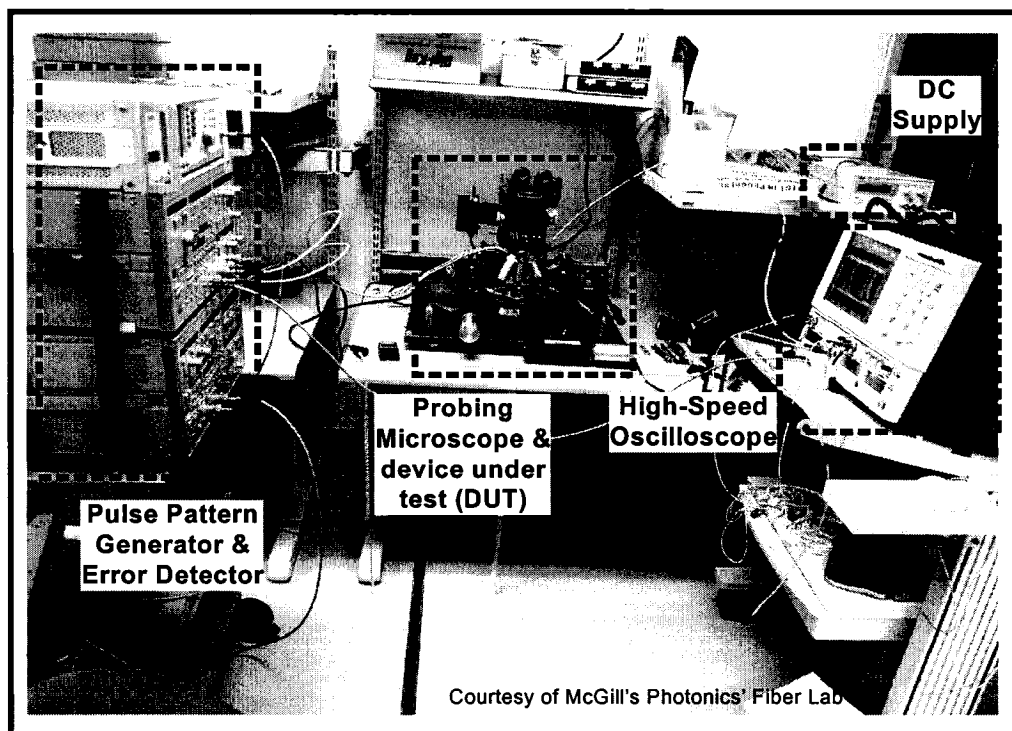


Figure 5-7 Test setup for the measurement of the eye diagram and the BER in McGill's Photonics' System Group's Fiber Lab.



- DC source – Agilent E3630A & E3646A – this equipment is used to provide DC power to the board, which will then distribute the biasing and supply voltages through voltage regulators on the PCB.
- Power spectrum analyzer (PSA) – Agilent E4440A – this equipment is used to measure the frequency power spectrum of signals. This instrument can be especially useful since it does not require a high-speed trigger signal, unlike the high-speed digital oscilloscope. Some PSAs come with phase noise measurement modules, and therefore the jitter and the quality of the oscillation can be measured as well.
- Pulse pattern generator & Error detector – Anritsu MP1763B & MP1764A – This set of instruments allow the generation of the PRBS patterns ( $2^7-1$  to  $2^{32}-1$ ) and the detection of the corresponding error in order to assess the BER.
- High-speed digital oscilloscope – Tektronix TDS 8000B with the 80E03 sampling module – This is a high-speed digital oscilloscope that allows the generation of the eye diagrams and the measurement of the opening of the eye, the jitter, etc.

## 2.2 Probing Station

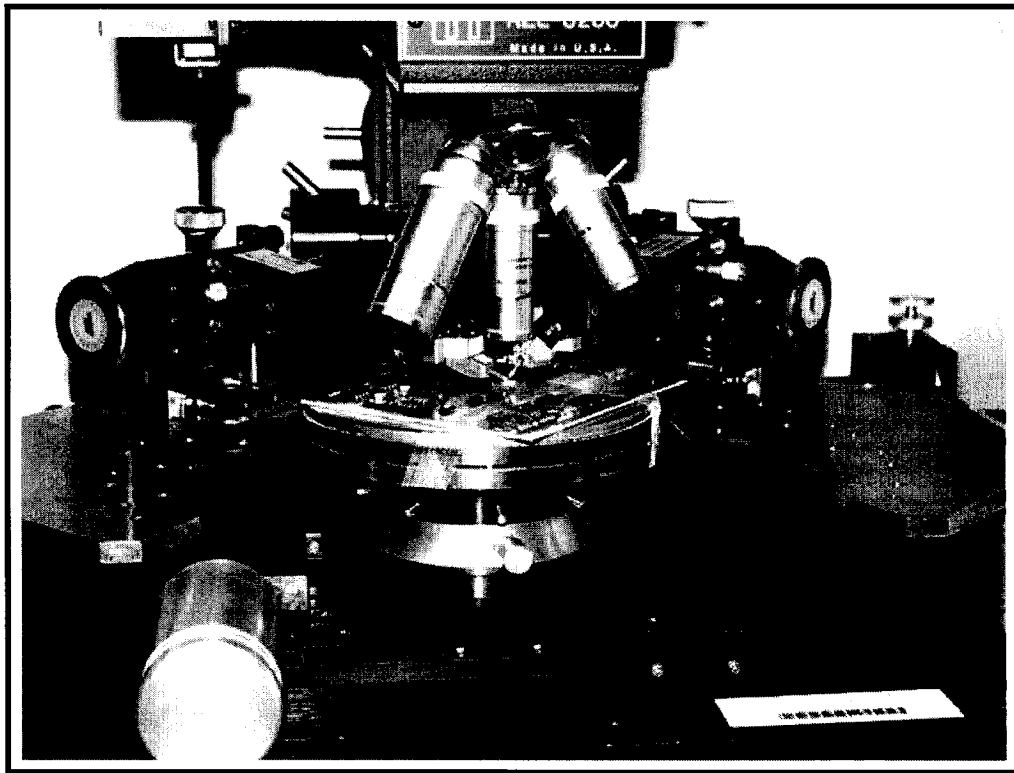
Figure 5-8 is a photograph of the probe station setup: the prototype PCB and the ground-signal-ground (GSG) and signal-ground-signal (SGS) probes are included in this picture. As described earlier, all high-speed or frequency sensitive ports were probed on-chip. Figure 5-9 shows a close up image of the delicacy of probing (note the proximity of the probes to the bonding wires).

## 3 VERIFICATION & RESULTS

---

### 3.1 Test Bench for Simulations

During the design phase, a low-level simulation was performed using Cadence's Analog Environment. In order to shorten the simulation times, a periodic signal was used for plotting the graphs in Figure 5-10. Although a PRBS could have been used in order to really assess the capability of the system, this approach was used to simply observe the trend and the best-case performances of the circuit blocks of the CDR system. At later



**Figure 5-8** Probing station microscope with the probe arms holding the SGS and the GSG probes.

stages, a PRBS simulation was performed on this system, and the trends corroborate to the ideal response in Figure 5-10.

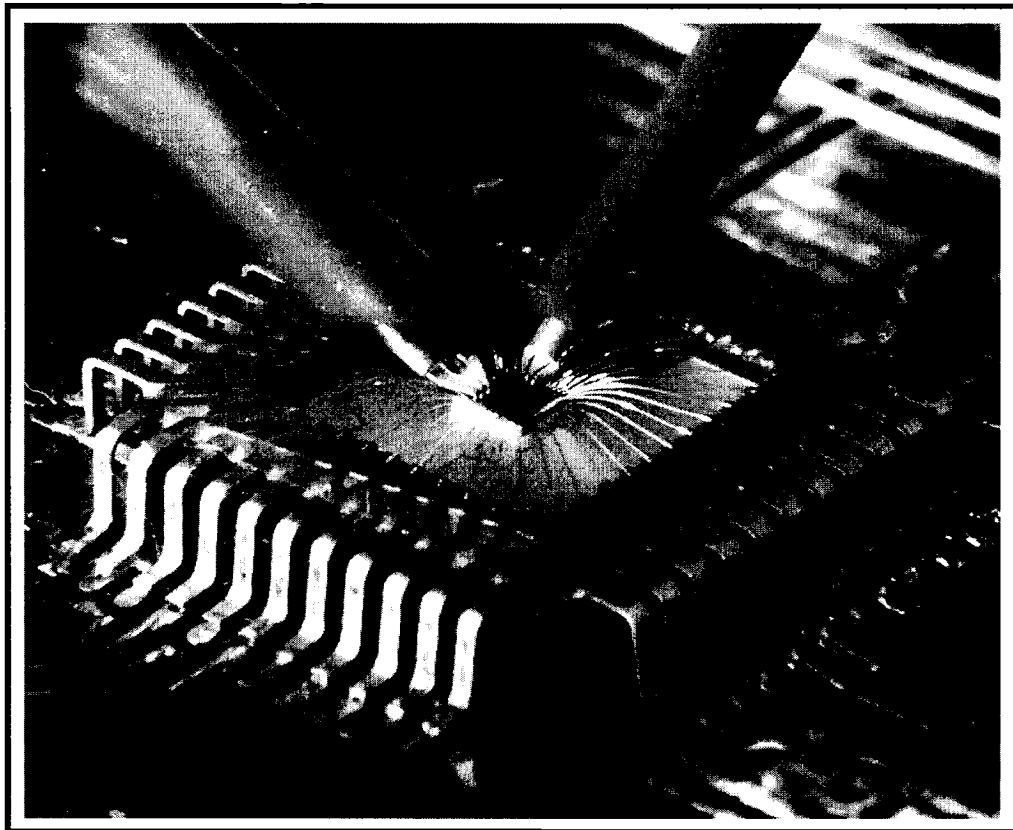
### **3.2 Simulation Results of the Half-Rate Phase/Frequency Detector**

In order to measure the characteristics of the different building blocks of the system, the system must be tested in open-loop. Performing a fast/slow and early/late clock simulations, the PFD results are plotted in Figure 5-10 and Figure 5-11. Figure 5-10 clearly shows that the subtraction of the PD and FD signals indicates a corresponding speed up or a speed down of the output.

### **3.3 Current-Starved Ring Oscillator**

#### **3.3.1 Frequency Transfer Characteristic & Gain**

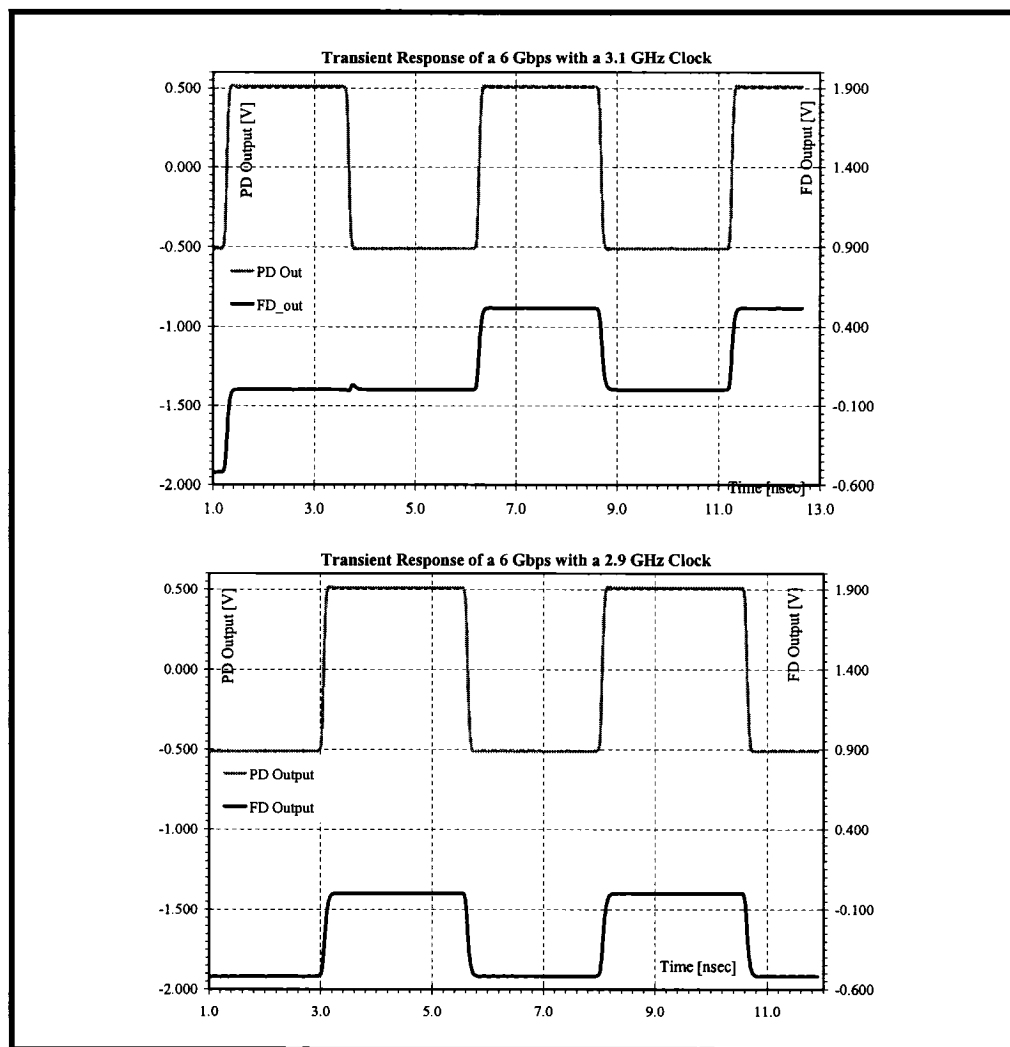
The simulation of the ring oscillator was performed on a fully extracted layout of the CDR system, with the parasitic capacitances calculated by Cadence's Diva Parasitic



**Figure 5-9** Photo of the microchip in the 44-pin package. The SGS probe is gently touching the clock port for VCO characterization.

Extractor. Furthermore, the slow-slow corner of the CMOS models were used to generate the simulation results, as shown in Figure 5-12. This results represents the worst case performance to be expected.

The measured results of the implemented chip are found in Figure 5-13. Notice that the measured frequency of oscillation is lower than the simulated one. Initially, this measurement was thought to be unexpectedly low due to poor fabrication yield. However, a second chip of the same fabrication run was tested and consistent results as those in Figure 5-13 were measured. Furthermore, the tuning ranges of both chips were quite similar, and so were their frequencies of operation. This kind of discrepancy suggests that there is a modeling issue with the extractor. As analyzed in Figure 4-18 (please refer to the section entitled “Current-Starved Ring Oscillator” on page 56), an increase in the total capacitance at the output of each stage would cause such a shift in the oscillation frequency, without affecting the gain or the tuning span. The results are summarized in



**Figure 5-10** Simulation of the open-loop CDR. The output voltage of the PD and FD is simulated against different clock speeds for an input rate of approximately 6Gbps.

Table 5-1.

As shown in Figure 5-14, the array of capacitors used to slow down the ring oscillator creates a complicated network of capacitances and mutual capacitances. It also shows the possible capacitance paths from one edge of the capacitor to the 7 other adjacent edges. The Diva parasitic extractor only takes into account 3 fringing paths out of the 7 possible ones. In fact, this extractor program only generates fringing capacitances between metals on the same layer and to non-diagonal objects indicated by dark arrows on Figure 5-14. These capacitances may seem negligible compared to the capacitor array, however elec-

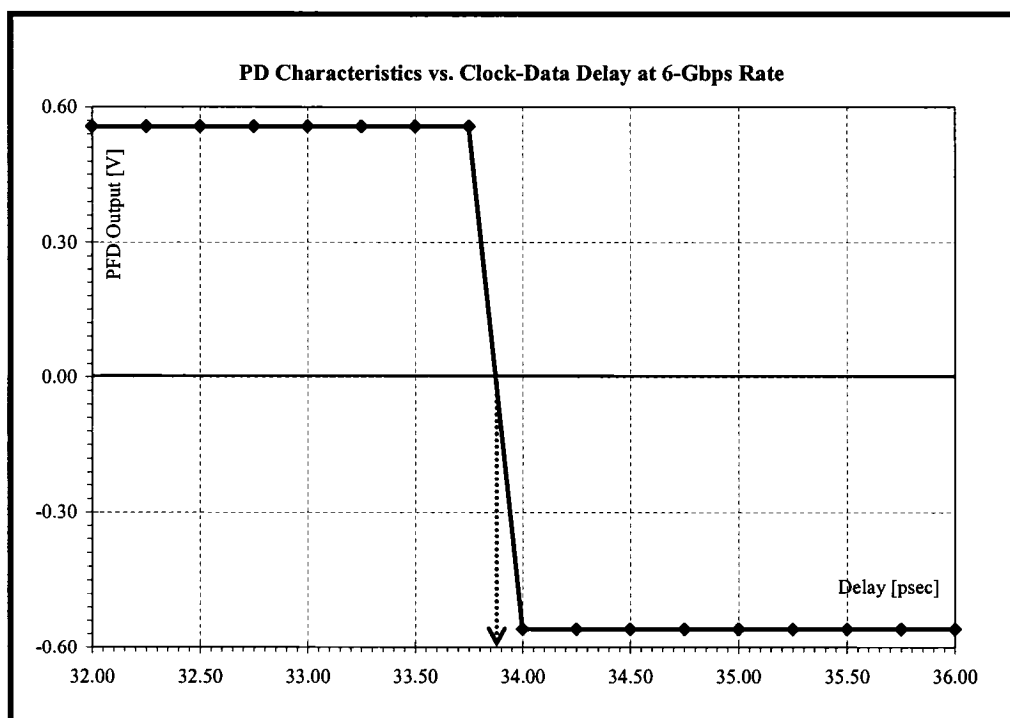


Figure 5-11 Simulation of the open-loop CDR for the PD characteristics running at a 6-Gbps data rate.

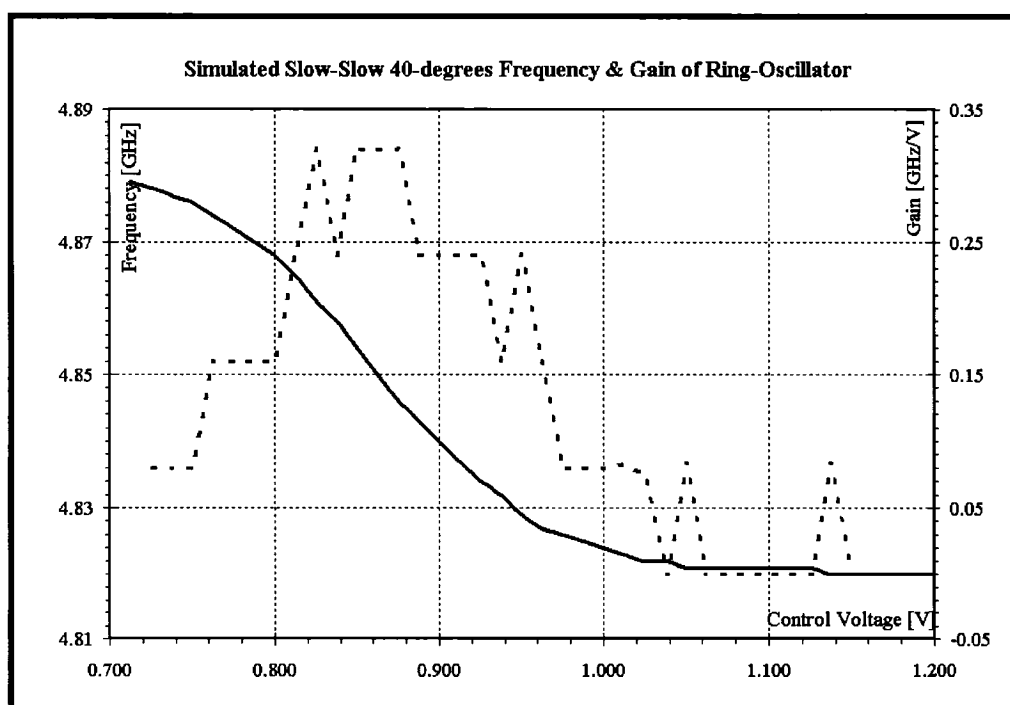
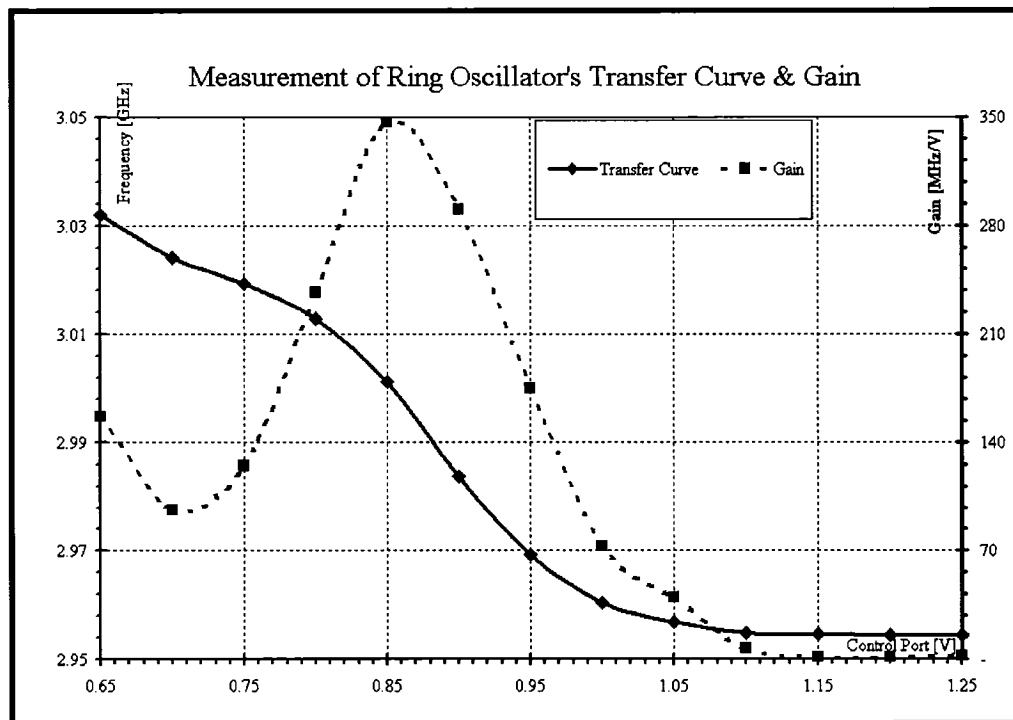


Figure 5-12 Full parasitic extraction simulation result for the VCO's frequency versus the control voltage using the slow-slow corner models of TSMC's CMOS 0.18-micron at a 40-degree Celsius operating temperature.



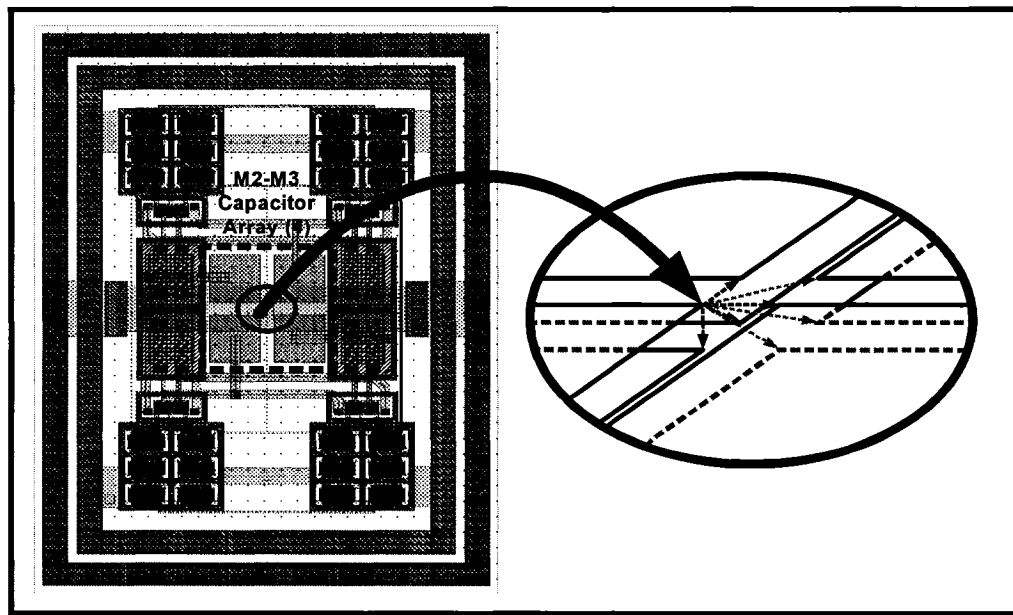
**Figure 5-13** Measurement results: frequency vs. the control voltage of the ring oscillator (dashed line) and its corresponding frequency gain (solid line).

**Table 5-1** Comparison between measurements and simulation results of the ring oscillator

	Measured Results	Simulated Results
Frequency at 0.9V	2.98GHz	4.84GHz
Approximate Frequency Span	60MHz	60MHz
Max. Frequency Gain	350MHz	330MHz

tro-crowding and Miller effects may increase the overall effective capacitance at each node. This hypothesis may explain the shift in the frequency without affecting the tuning range or the frequency gain.

Another possible explanation for this discrepancy may come from the long stripline wires that connect the inverter stages in the ring oscillator. These relatively long wires may introduce parasitic inductances over long lengths. However, simulations showed that a significant increase in these parasitic inductors negligibly affected the oscillation frequency. Although the wires are somewhat long, they are also very wide. Therefore, the effective inductance of these wires are most probably not the main cause of this significant



**Figure 5-14** Capacitor array located on the layout of the ring oscillator. Each capacitor is connected differentially at the output of one stage. The positioning of the array may cause complex fringing patterns, hence generating many unaccounted for parasitic capacitances.

discrepancy in frequency of oscillation.

There are several solutions to remedy this discrepancy in the future: one is to use more accurate extractor tools such as Mentor Graphics Calibre XRC, or use FEM solvers in order to treat the capacitor array as an 8-port network. Another method is to move the capacitors further apart in order to minimize complex coupling and fields, and hence to allow the extractor program to model more accurately the parasitic capacitances.

### 3.3.2 Phase Noise and Jitter

The output of the half-quadrature ring oscillator was measured with the Agilent E4440A power spectrum analyzer (PSA) with a phase noise personality. The phase noise and power spectrum are measured and shown in Figure 5-15 and Figure 5-16, respectively. These results are from a free-running VCO.

According to the phase noise measurement results, the approximated jitter can be calculated using Equation 2-12 or Equation 2-13 (Chapter 2 Section 4). Taking the phase noise measurement at the first three points in Figure 5-15, the cycle-to-cycle jitter can be estimated as tabulated in Table 5-2. The last point has been discarded since it seemed to

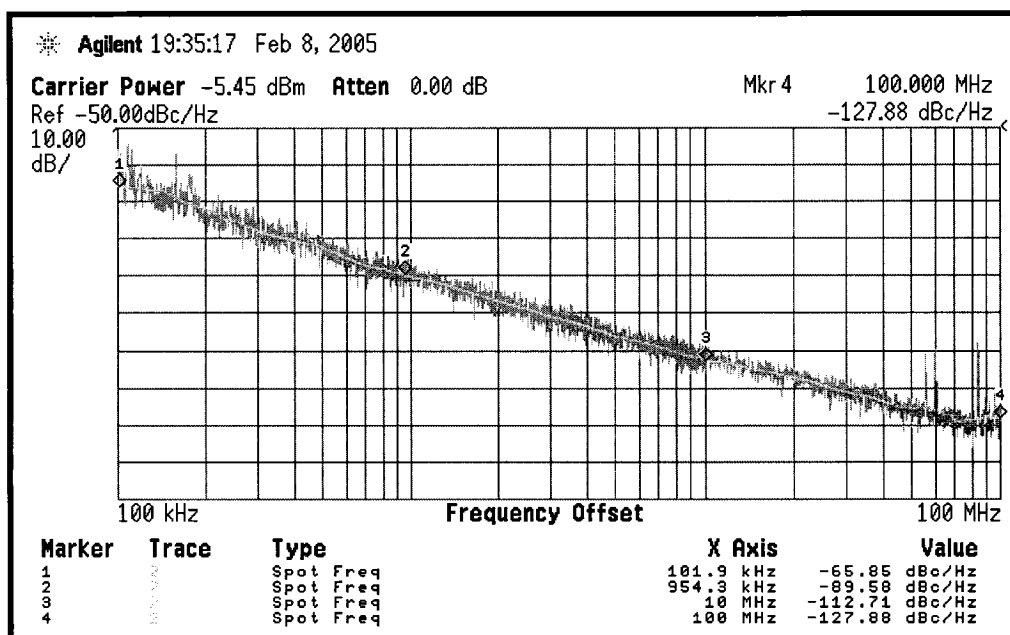


Figure 5-15 Phase noise measurement of the half-quadrature ring oscillator, oscillating at approximately 3.0 GHz.

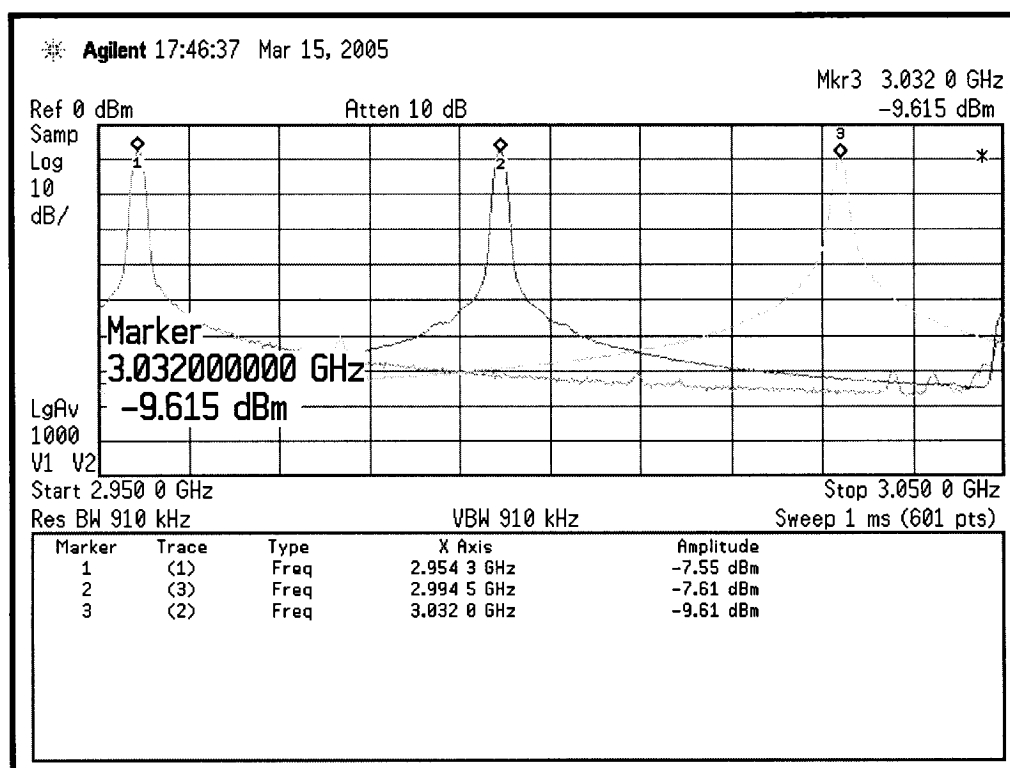


Figure 5-16 Power spectrum measurements of the half-quadrature ring oscillator at the highest, lowest, and middle frequencies.

have reached the noise floor of the PSA. Table 5-2 shows that, taking the worst case, the



RMS cycle-to-cycle jitter is approximately 0.45 psec. Since it is expected that the peak-to-peak jitter is larger than the RMS jitter, these measurements confirm the jitter simulation of approximately 0.5 psec peak-to-peak jitter of Figure 4-17. This jitter value corresponds also to the lower bound jitter generation value for the entire CDR system.

**Table 5-2 Relative phase noise, and corresponding approximate cycle-to-cycle RMS jitter for a center oscillating frequency of 3 GHz.**

Relative Phase Noise [dBc/Hz]	Frequency Offset [MHz]	Approx. CTC RMS Jitter [psec]
-65.85	0.1019	0.45
-89.59	0.9543	0.27
-112.71	10.000	0.20

### 3.4 Closed-Loop Simulations and Measurements

#### 3.4.1 System Setup for Simulation

Full parasitic extracted simulations are seldom performed in systems as large and complex as CDRs, due to the demanding computational and time resources required. However, other higher level simulations can be done in order to get a good understanding of the behavior and the operation of the designed system. In this work, a schematic circuit-level simulation was performed using a combination of Spectres' Verilog-A model which described the fully differential half-quadrature ring oscillator. The generated Verilog-A codes were obtained through the ModelBuilder program included in the Cadence package. The Verilog-A codes are found in the Appendix (please refer to the section entitled "Verilog-A codes for the Ring Oscillator Components" on page 113).

This mixed-mode simulation (behavioral and circuit level) provides a very good compromise in verification accuracy & computational speed. The appropriate inductances were added in order to take into account the effects of the bonding wires. For instance, at the loop filter port, an inductor of 2 nH is added, and the supplies 150-pH inductors are added. Fortunately, all critical and high-speed signals, except the off-chip loop filter, are

probed on chip.

### 3.4.2 Modeling the Ring Oscillator to Fit Measurement Data

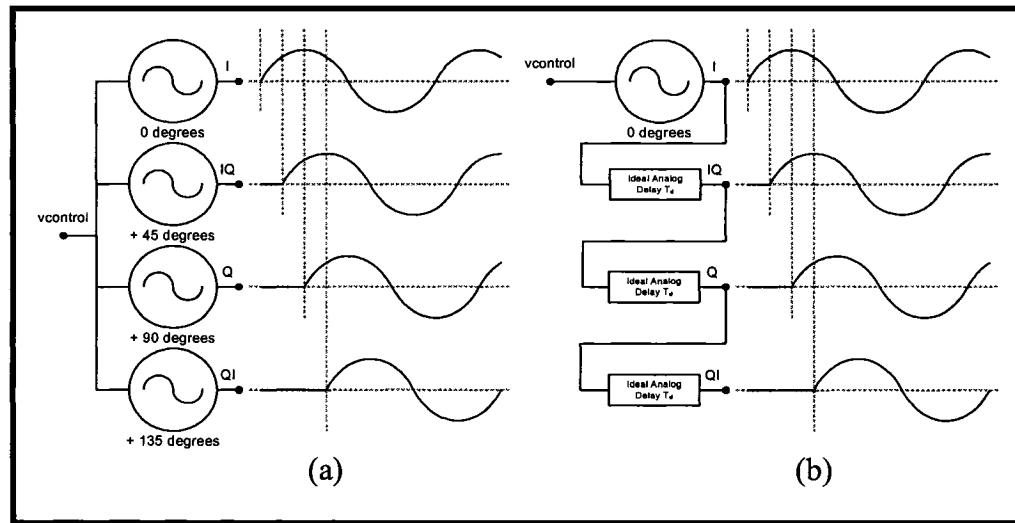
Verilog-A programming is not necessarily a prerequisite in order to build a simple model for circuit blocks ([39], [40] & [41]). Fortunately, with the presence of the Model-Builder CAD tool, it is possible to combine schematic and Verilog-A blocks to design straightforward behavioral models of complex circuit blocks. This type of synthesis is very useful and practical for post-fabrication verifications. In the case of the work in this thesis, the VCO's measurements and extracted performances were inconsistent. With the collected measurement results, a VCO model can be synthesized to operate at the measured frequency, i.e. about 3GHz. The required specifications are reported in the table below:

**Table 5-3** Compilation of some specifications of the measured VCO.

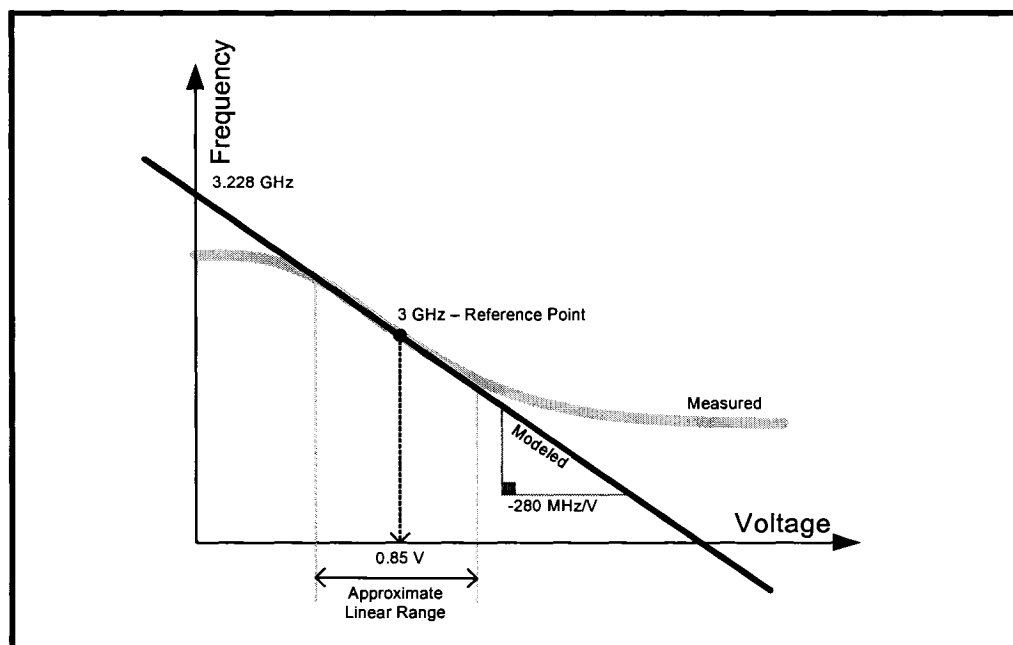
Characteristics	Values
Output DC voltage	1.2V
VCO gain of linear region	-280MHz/V
Output amplitude	300mV
Reference points	3.00GHz @ 0.85 V

There are several approaches to design the VCO. Taking advantage of the VCO block in ModelBuilder, we can construct a 4-phase half quadrature VCO. In order to generate the 4 half-quadrature phases, one can use 4 VCOs Verilog-A blocks, and then program a phase delay in each VCO, as illustrated in Figure 5-17. However, these VCO models don't come with phase delay parameters; therefore the phase difference must be set by using the ModelBuilder's analog delay block. The delay block must be given a starting frequency, otherwise the respective phases will not be half-quadrature, or 45 degrees apart. Furthermore, in order to complete the behavioral model, the output signal must be differential with a specified common-mode voltage. This can be set by utilizing the voltage-controlled voltage source (VCVS). The parameters of the VCO and the delay blocks were set in reference to the measurements frequency transfer function, the approximated gain of the lin-

ear range, and the reference points that are shown in Figure 5-18.



**Figure 5-17** The two different approaches in modeling a VCO with 4 half-quadrature phases: (a) 4 VCOs with a different phase delay programmed into each block, (b) one VCO with 3 analog delay blocks.



**Figure 5-18** VCO's frequency transfer plot, for both the actual and modelled devices.

Although this model is a first order approximation, the ABM VCO behaves similarly within the given linear range. One of the more complex limitations is the modeling of the phase noise. For this higher level simulation, we will neglect and ignore the phase noise

generated by the free running VCO, since the jitter generated in closed-loop is greater than the jitter generated by the free running VCO.

### 3.4.3 Mixed-Mode Simulated Jitter Generation

The simulated voltage at the loop filter shows that the CDR does tend towards a steady state, hence converging towards a locking state. The jitter generation can be estimated by producing an eye diagram in MATLAB with the help of a script described in the Appendix (please refer to the section entitled “Abbreviations” on page 109).

One of the major concerns in the design process of CDR systems is the lengthy simulation time, especially if one wants to observe the steady state result. Depending on the initial conditions, some simulations must go on for several milliseconds in order to truly assess the jitter generation of the entire CDR system. In this case, a several millisecond simulation is impractical, since it would require in the order of several hundred days on a decently fast computer. As shown in Figure 5-19, the system seems to reach some sort of steady state starting from  $12\mu\text{s}$ . Note that, for this specific example, the data input stops feeding in at around  $17\mu\text{s}$ , which explains the very quiet activity at that point forward. This also shows that the system can definitely handle trailing ones and zeros. Another simulation of the loop filter is shown in Figure 5-20 running at a 1.5-Gbps data rate. There is a clearly similar pattern of escalating up/down signals before the signal becomes somewhat stable. This consistency might suggest that this specific behavior comes from the design and the operation of the PFD.

Assuming that the CDR reaches a steady state and locked condition, fast moving jitter can be estimated from the thickness of the line, as indicated by Equation 5-2 below, where  $\Delta V_{fast}$  is the thickness of the loop filter line in volts,  $K_{VCO}$  is the gain of the VCO, and  $f_0$  is the frequency of oscillation of the recovered clock. However, in the presence of a slow-moving jitter, the actual peak-to-peak jitter can only be calculated by determining the  $t_{update}$ , the maximum update time during which the loop provides corrective information. Equation 5-3 calculates the drift or the jitter caused by a slow moving loop filter voltage  $\Delta V_{slow}$  when  $T_{bit}$ , the bit period, is much less than  $t_{update}$ . On the other hand,  $\Delta T_{fast}$  is the fast moving jitter and  $t_{update}$  is approximately equal to  $T_{bit}$ . Figure 5-21 graphically indi-

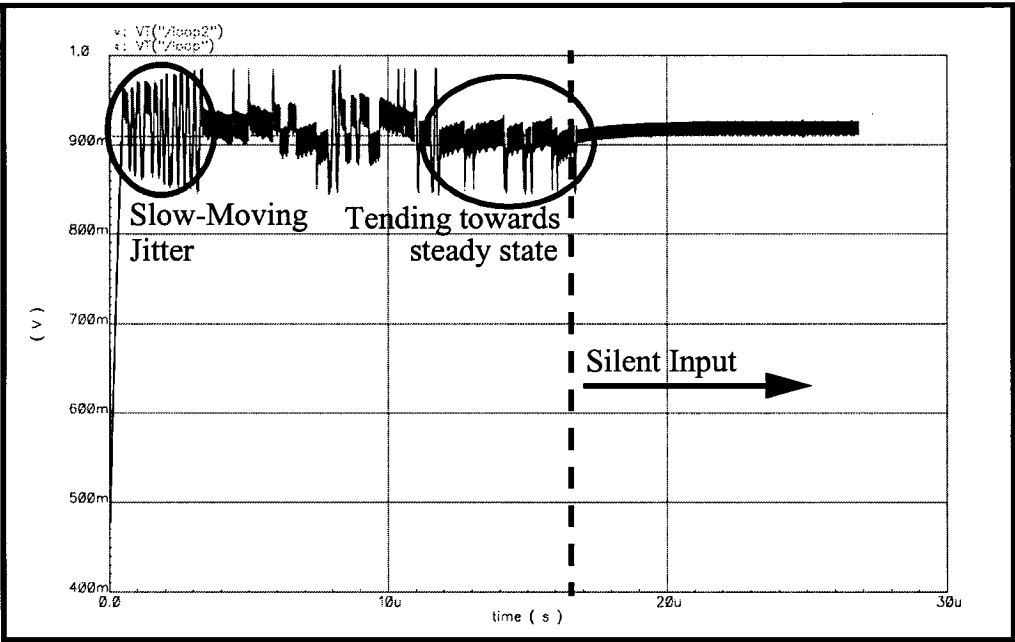


Figure 5-19 Simulated voltage of the loop filter with a capacitor  $C_P$  of 1nF and  $R_P$  of 900kOhms. This CDR was set to a 6-Gbps rate.

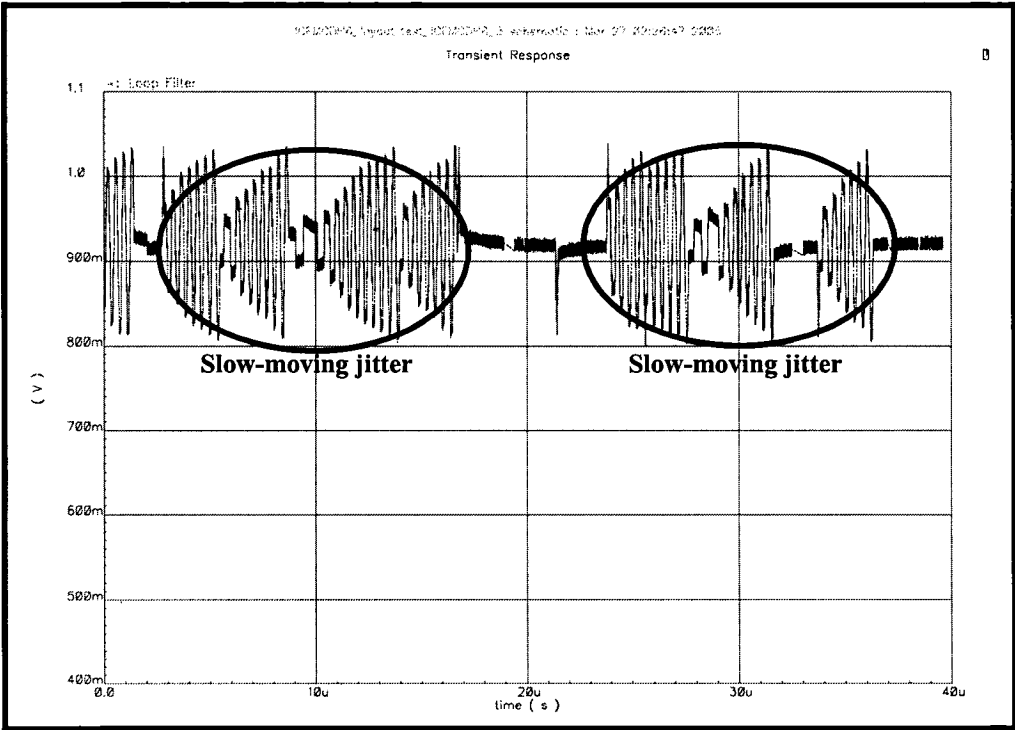


Figure 5-20 Simulated voltage of the loop filter with a capacitor  $C_P$  of 1nF and  $R_P$  of 900kOhms. This CDR was set to a 1.5-Gbps rate, with a longer transient time.

cates these variables for fast and slow jitter.

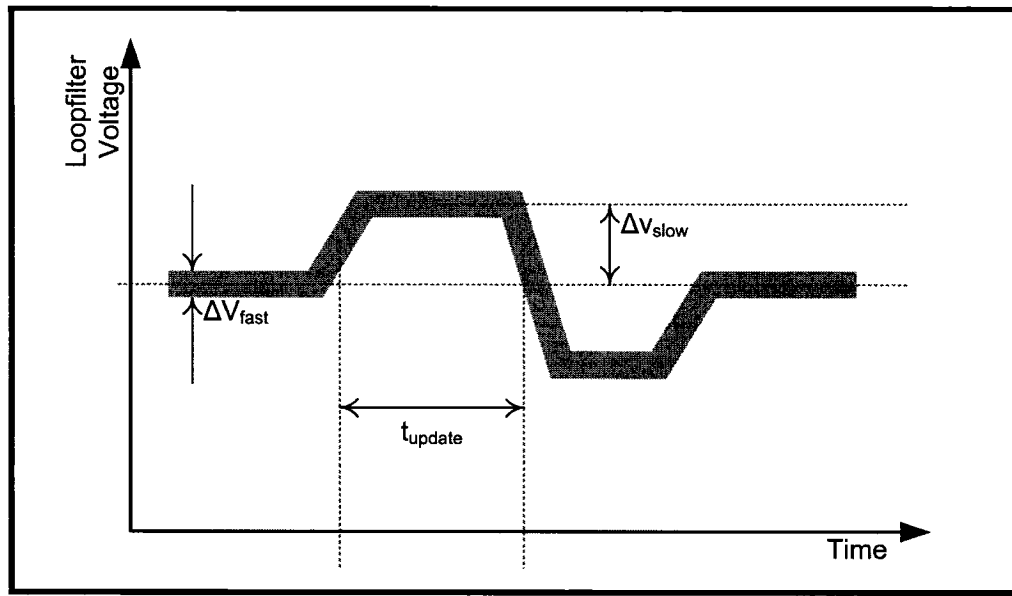


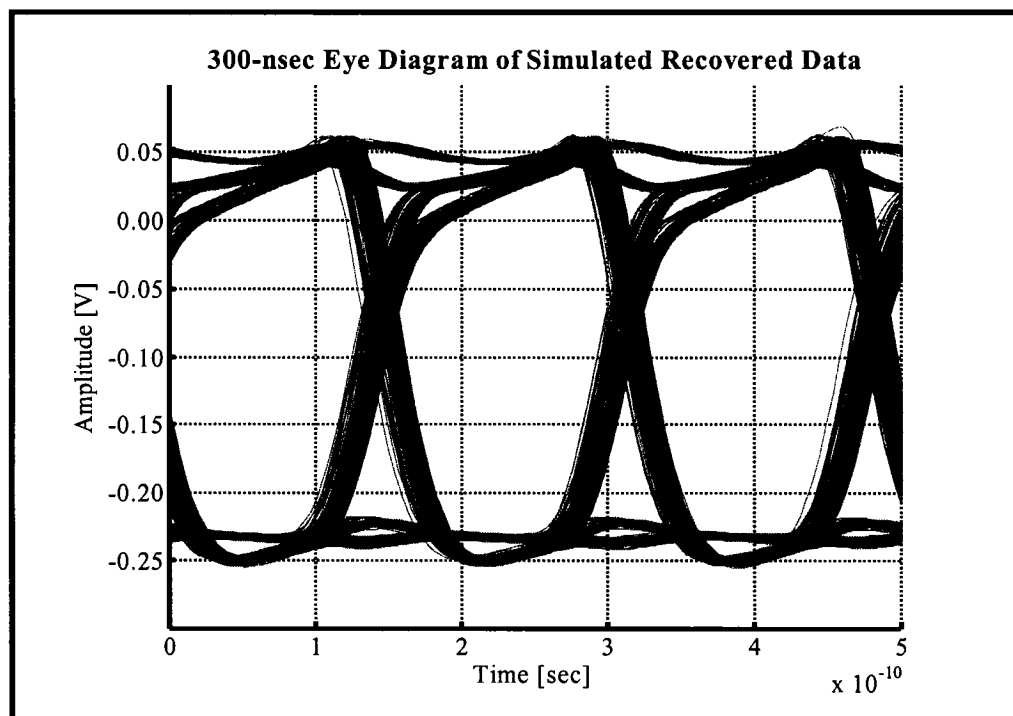
Figure 5-21 Transient of the loop filter along its corresponding variables for calculating fast and slow jitter.

$$\Delta T_{fast} = \left| \frac{\Delta f}{f_o^2} \right| = \left| \frac{\Delta V_{loopfilter} \cdot K_{VCO}}{f_o^2} \right|, \quad (5-2)$$

$$\Delta T_{slow} = \frac{t_{update}}{T_{bit}} \cdot \left| \frac{\Delta V_{slow} \cdot K_{VCO}}{f_o^2} \right| = \left| \frac{2 \cdot t_{update} \cdot \Delta V_{slow} \cdot K_{VCO}}{f_o^2} \right|. \quad (5-3)$$

In order to demonstrate the fast moving jitter phenomenon, the simulated recovered data is shown in Figure 5-22. The eye diagram was generated using MATLAB's eye diagram script described in Appendix (please refer to the section entitled "MATLAB Eye Diagram" on page 111) and was taken from a 300-nsec time frame of a fast-moving jittery loop filter signal, e.g. at about the 12- $\mu$ s time point of the signal shown in Figure 5-19. This eye diagram indicates a peak-to-peak jitter of approximately 20psec for this time period only. Had a larger time span been taken, the MATLAB generated eye diagram would look all garbled, due to the slow-moving jitter which is in the order of several unit intervals (UI) or bit period. This is problematic since a CDR is supposed to reject jitter.

The root cause of this behavior has been determined by running several verification simulations. The system was subjected to several simulation conditions in order to narrow



**Figure 5-22** Simulated 300-nsec eye-diagram window of the recovered data running at a 5.96-Gbps data rate. The 300-nsec section was taken from the most quiet loop filter activity.

down the source of this slow moving jitter. First, the CDR system was simulated with a periodic input instead of the usual PRBS. The results look very much like the plot in Figure 5-20. Although it may seem at first that the system reaches steady state after several microseconds, the loop filter voltage is not quite locked when simulated for longer times. The PD output indicates that the phases are constantly drifting, hence not locked to the input signal. This suggests that this PFD has some kind of a dead zone: at frequency differences below a certain critical threshold, the PFD no longer toggles. This is a fundamental flaw of this circuit and can certainly be the reason for the unusual activity with a PRBS input, observed in Figure 5-19 and Figure 5-20.

Another method to confirm this hypothesis is to use the test setup shown in Figure 5-24: the system loop is broken at the node where the loop filter meets the ring oscillator. A deterministic ring oscillator control voltage signal was used in order to observe the proper response at the charge pump. The results show that the response is correct, but is subjected to the density of the edges in the PRBS signal. One of the concerns is the time it is

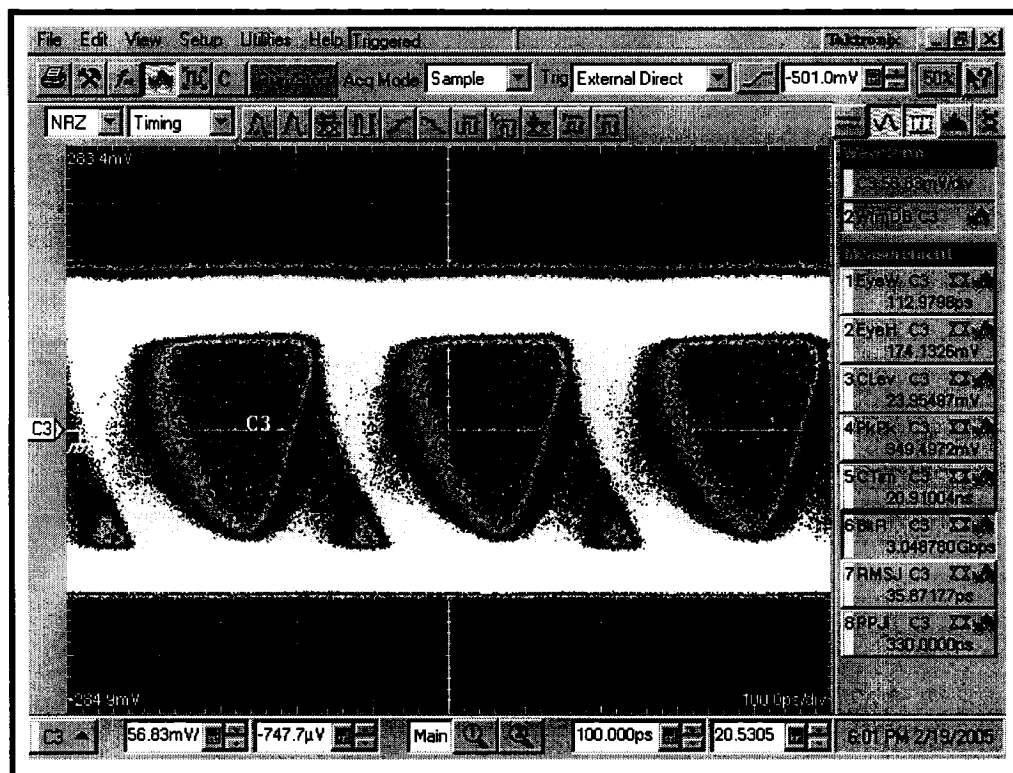


Figure 5-23 Eye diagram of the recovered data of the ICFMGDH6 (Die no. 1) at approximately a 3-Gbps data rate with a DC-balanced PRBS type  $2^{23}-1$

required for the PFD to detect the phase/frequency error,  $t_{update}$ . A simulation was run to show the so called “dead zone” of the PFD. Near its locking voltage, the PFD either remains idle or behaves somewhat randomly. Figure 5-25 clearly shows the possible origin of the anomalous jittery loop filter problem: the output of the charge pump becomes unpredictable as it approaches near the locking voltage. There is consistently an anomalous behavior at every crossing point of the locking voltage.

Finally, jitter tolerance is an important result that cannot be simulated: with the current models, it is very difficult to setup such a verification method. For the prototype under test, there is little hope to obtain good jitter tolerance due to the abnormal behavior of the PFD. It is expected that the jitter tolerance would be poor and constant through fast and slow input jitters. If we were to measure jitter tolerance, such tests can only be done with



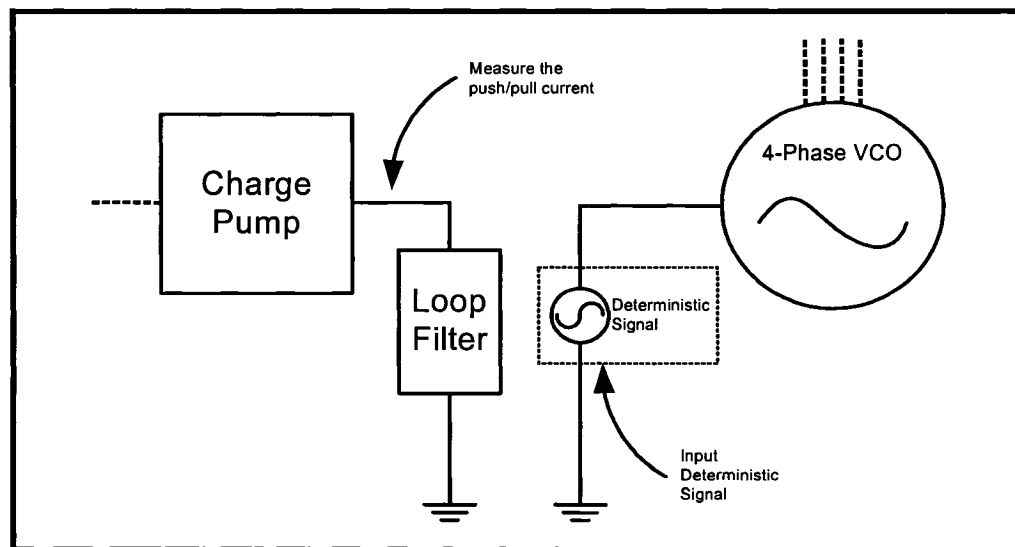


Figure 5-24 System setup for open-loop simulation: Input a deterministic signal at the control voltage and measure the push/pull current response.

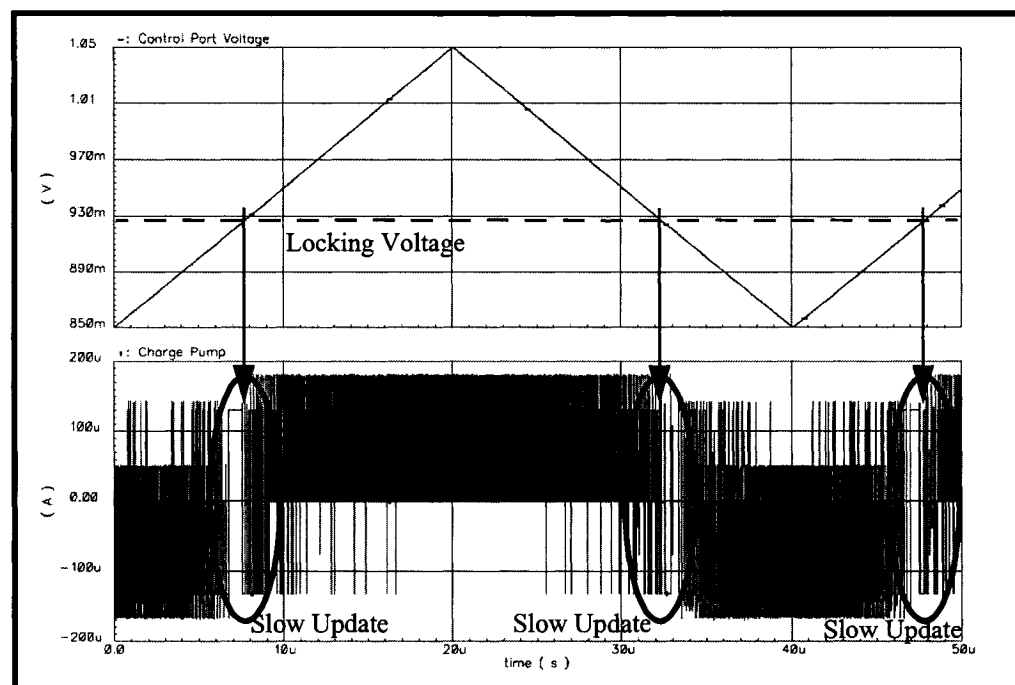


Figure 5-25 Simulation of the open-loop system as described in Figure 5-24. The transition near the locking voltages should be digital or binary, but instead it is somewhat random and unpredictable.

the proper instruments, such as Agilent's N4900 serial BERT series testing instrument.

### 3.4.4 Measurement of the Recovered Data

Several measurements of the recovered data were recorded using the Tektronix CDS

8000 oscilloscope. Figure 5-23, Figure 5-26, and Figure 5-27 are the measured eye diagrams of the recovered data for the data rates of 6Gbps, 3Gbps, and 1.5Gbps, respectively. According to the verification results of the closed-loop operation, the eye diagram plots should look garbled and hence unreadable. However, the high-speed digital oscilloscope has a practical feature of database waveform that allows to weigh the occurrence of each overlapped plot. This histogram feature allows to determine a more clear eye diagram pattern, as shown in the figures.

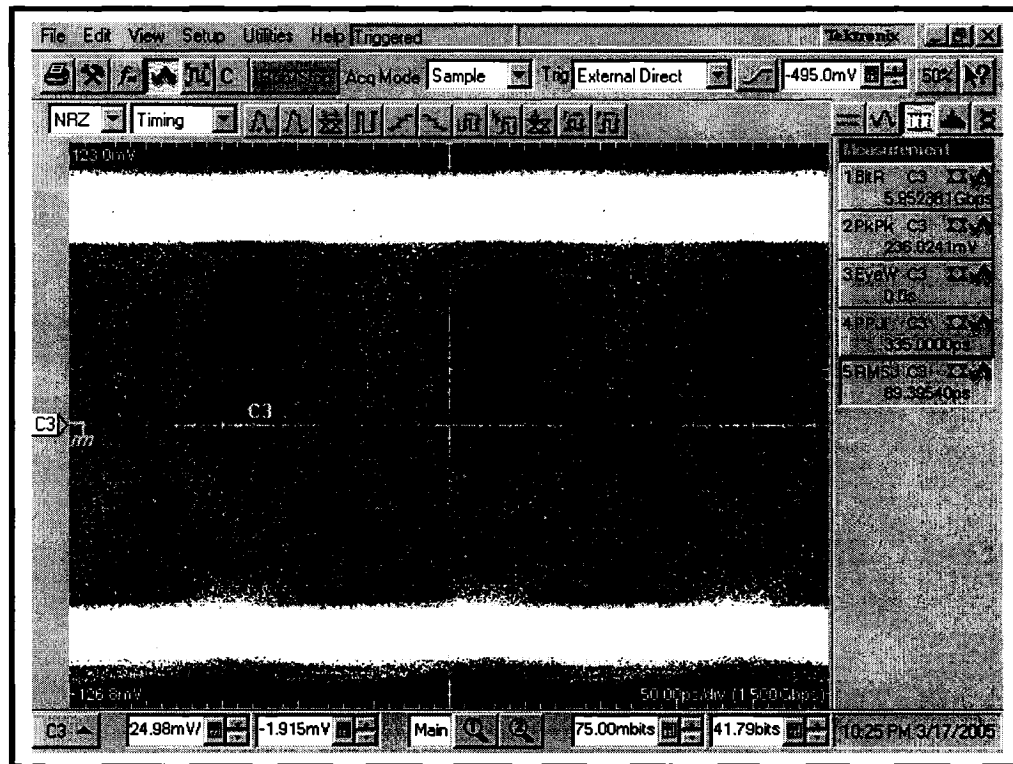


Figure 5-26 Eye diagram of the recovered data of the ICFMGDH6 (Die no. 2) at approximately 6-Gbps data rate with a DC-balanced PRBS type  $2^{23}-1$

The measured results show that there is definitely a large amount of jitter, quite consistent with the results obtained from the mixed-mode simulation. However, the high-speed digital oscilloscope can detect a dominant pattern of an eye-diagram. Despite the pulsating jitter generation, the CDR does tend to track the bit rate. This result therefore confirms that the PFD is not working properly, not as predicted during the design stage in Chapter 4.

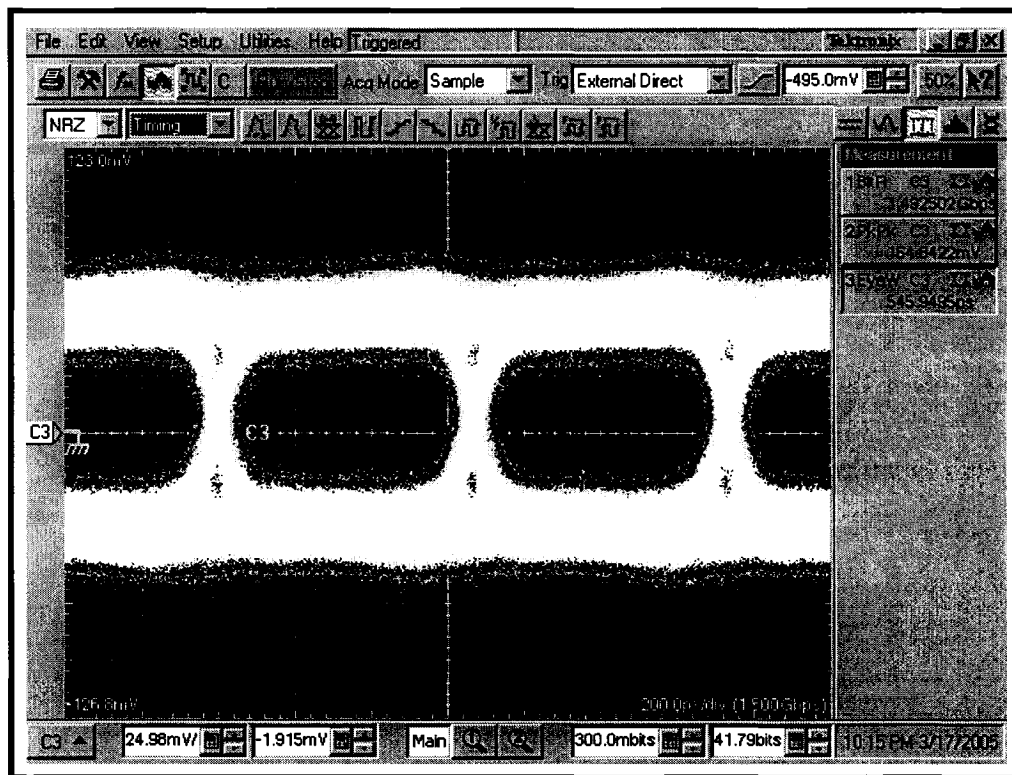


Figure 5-27 Eye diagram of the recovered data of the ICFMGDH6 (Die no. 2) at approximately 1.5-Gbps data rate with a DC-balanced PRBS type  $2^{23}-1$

### 3.5 Analysis of Results

Many measurements were taken in order to assess the performance of the system. Figure 5-28 shows that the jitter is definitely bounded and that the system does track the input PRBS data. Once again, these measurement results are consistent with the system level verification: the loop filter behavior, shown in both Figure 5-19 and Figure 5-20, corroborates with the fact that the jitter is bounded, as shown in Figure 5-28. Furthermore, it was previously calculated that the estimated slow jitter can be as large as in the order of 1 unit interval (UI).

Another eye diagram measurement shows the re-timer circuit with the ring oscillator shut off. Figure 5-29 shows by far the cleanest eye-diagram measurement. This result further eliminates the possibility that the jitter may be originated from the re-timer circuit. Figure 5-30 shows the screen capture of the clock signal on the high-speed oscilloscope. due to the large slow-moving jitter signal noted during the verification process, the clock does not have a clear pattern and looks garbled as predicted. Although the clock signal

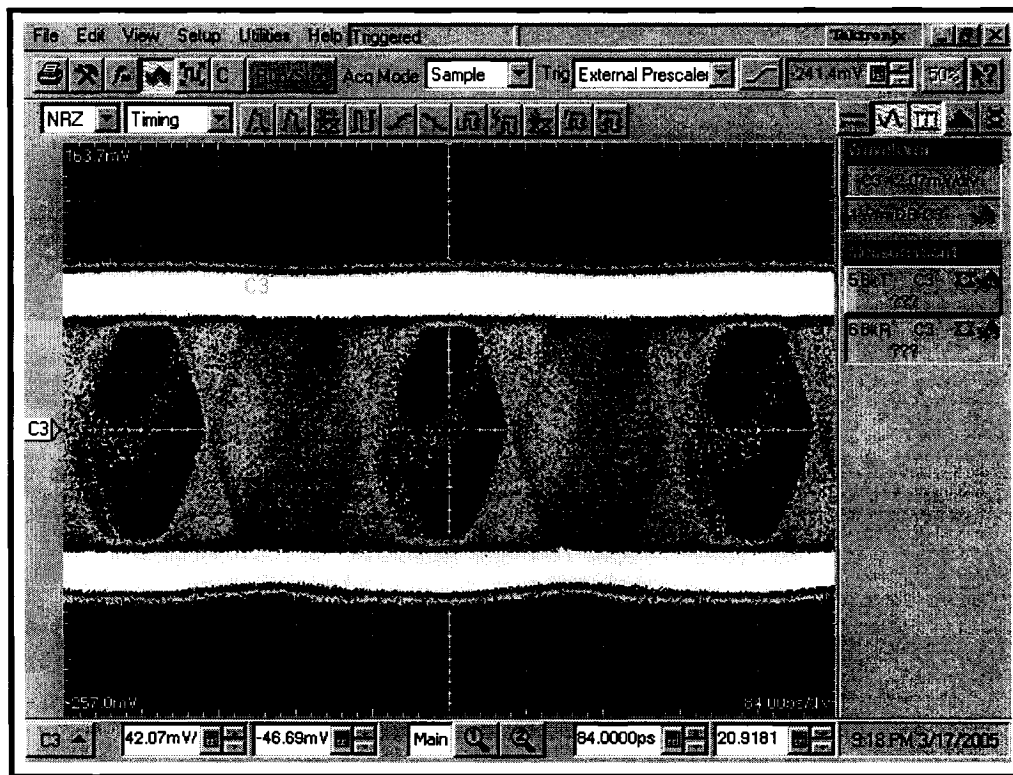


Figure 5-28 Another eye diagram measurement of the ICFMGDH6 (Die no. 2) showing the bounded or clamped jitter.

cannot be measured, the recovered data can be extracted by reducing the bit rate by an integer multiple, i.e. using a 0.75-Gbps rate instead of a 1.5-Gbps rate. As the input bit period increases, the bounded slow moving jitter takes a smaller fraction of a UI. Figure 5-31 shows a measurement taken with a lower bit rate than the acquisition rate in order to open the eye.

### 3.5.1 BERT & Eye-Diagram

First of all, in order to set the proper input data bit rate, the locking range of the system must be determined. This can be defined by measuring the ring oscillator's frequency transfer characteristic curves. For better BER results, the bit rate can be slower by a multiple integer of the actual operating rate. Since the implemented system is half-rate, the corresponding bit rate is twice the frequency of the ring oscillator, e.g.  $x$ -GHz oscillation corresponds to a locking bit rate of  $2x$ -Gbps data rate.

The BER of as low as  $10^{-12}$  were recorded with the Anritsu Error Detector at rates of

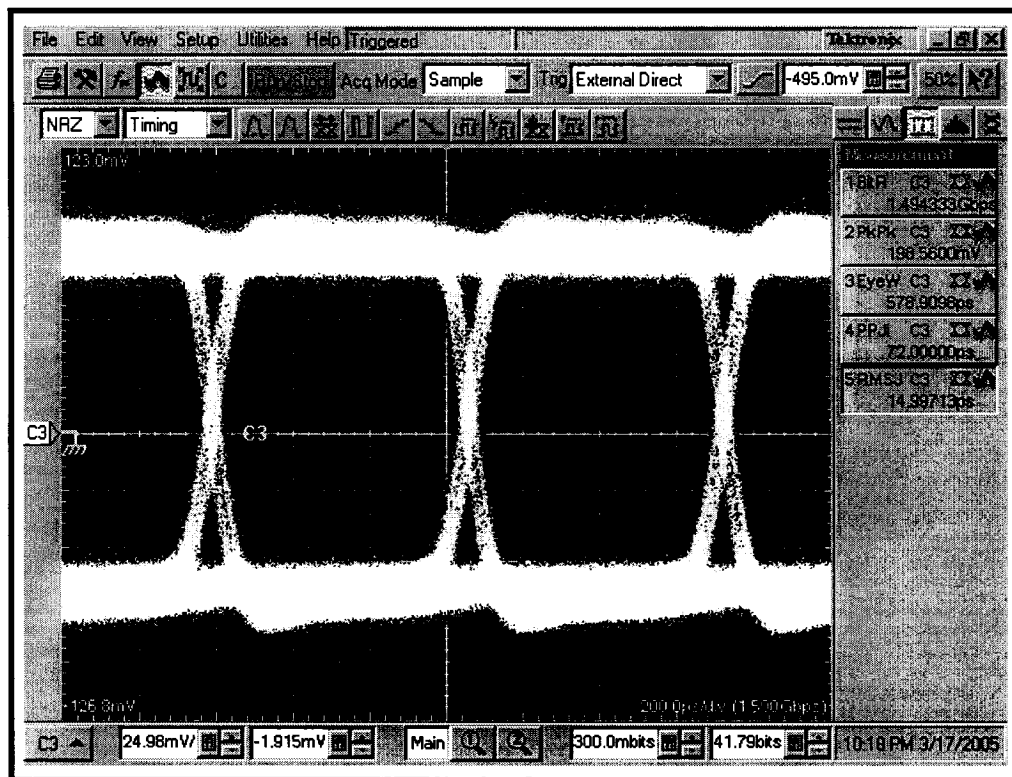


Figure 5-29 Eye diagram measurement of the retimer circuit with the ring-oscillator turned off.

approximately 1.5 Gbps. The BER of higher bit rates could not be determined due to the small eye-opening.

#### 4 SUMMARY

This chapter described the layout and the implementation of the testing board used to characterize the system built here. The testing procedures, verifications, as well as results were reported in this chapter, in order to demonstrate the validity of the modeling through measurements. Furthermore, this chapter addressed some challenges in setting up a mixed-mode simulation with Verilog-A, in order to reduce the simulation time with a reasonable compromise in the accuracy of the results. This section completes the bottom-up implementation with a top-down verification design flow.

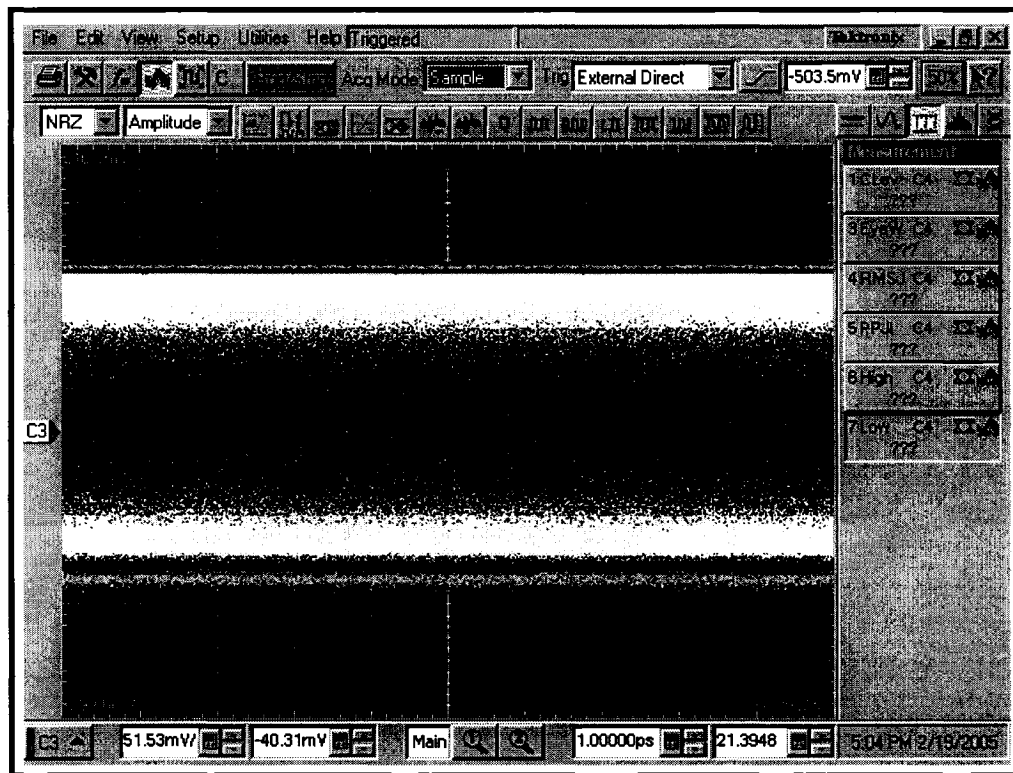


Figure 5-30 Measurement of the recovered clock. There is no recognizable sinusoidal or clock pattern due to the slow jitter caused by the PFD.

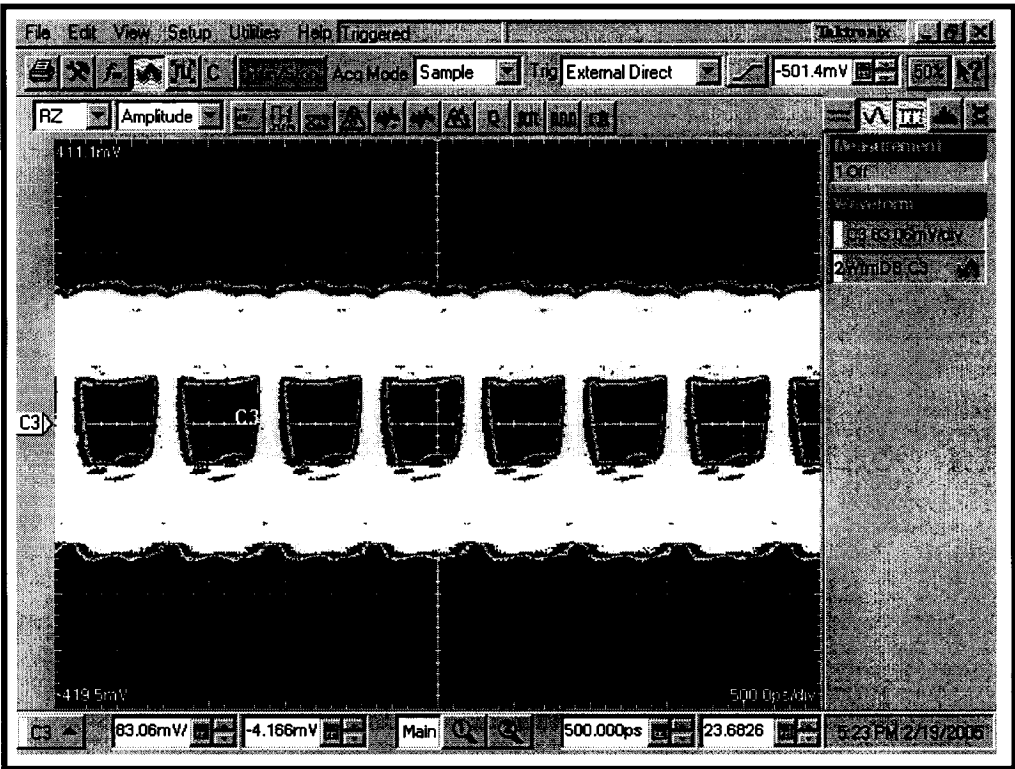


Figure 5-31 Eye-diagram measurement of the ICFMGDH6 (Die no. 1) for a rate of 0.75-Gbps rate running at an acquisition rate of 1.5Gbps.

---

# Chapter 6

Summary & Conclusion

---



## 1 MODELING & SIMULATION

### 1.1 Bottom-Up Methodology of Design

The modeling methodology used here is adequate for determining the system's performance. However, a higher-level system model should be designed in order to speed up the simulation, as well as to detect the weaknesses and strengths of the system before getting into the implementation phase.

This research followed a bottom-up approach of design. Such large systems are better designed top-down. This methodology can be much faster. Figure 6-1 shows a typical top-down design flow of a system.

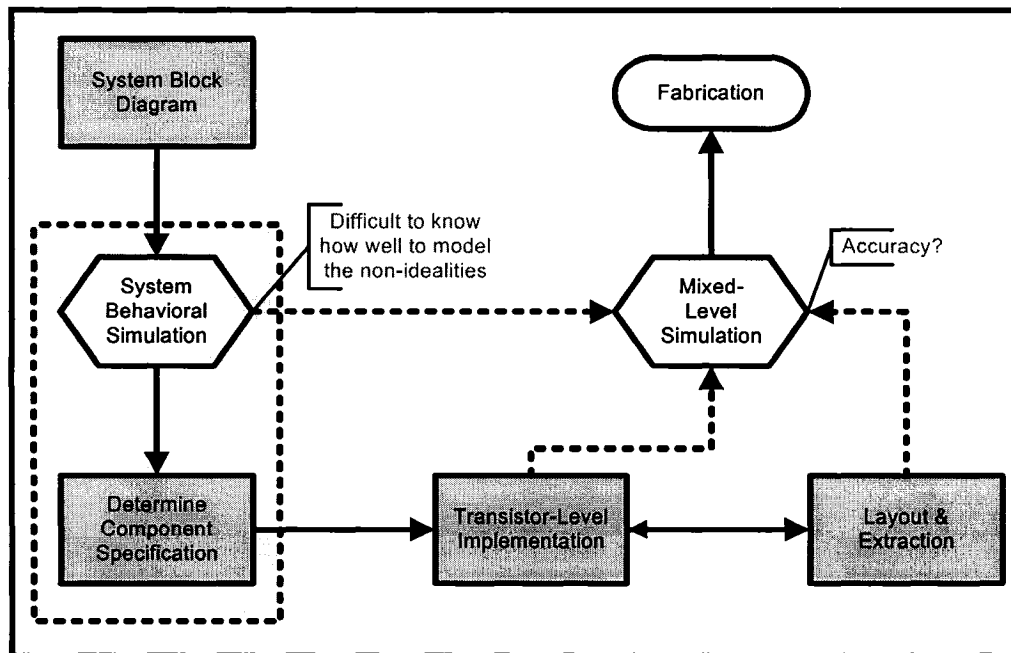


Figure 6-1 Flow chart of a typical top-down methodology of design. Most large scale systems are designed using this approach.

A top-down methodology implies the development and design of high-level models, making complete abstraction of the hardware level blocks. Such models may be oversimplified, but may provide considerable speed over accuracy. This approach can definitely provide useful verification strategies: for instance, jitter generation and tolerance as well as BER are difficult to obtain from time-domain simulations. With the ABM models, it

may be possible to verify such results within reasonable computational times.

## 1.2 Summary of Results and Issues

A CDR circuit is a large system that integrates many components such as the PFD, charge pump, VCO, divider network, retimer circuit, loop filter, etc. Therefore, the testing and verification results must be analyzed carefully in order to draw proper conclusions. In short, this work showed a working PFD, differential charge pump, and a half-quadrature ring oscillator working all at 6 Gbps, supported by eye diagrams and verification results.

This half rate binary PFD architecture shows convergence with some minor issues. Both open-loop and closed loop verifications were performed in order to support the measured results. Proper eye diagrams were shown and analyzed at all rates: 6 Gbps, 3 Gbps, and 1.5 Gbps. Better performances were shown generally at lower bit rates. Consequently, these results contribute to the fact that the charge pump, the divider network, the VCO, and the retimer circuit were working properly during the tests. In addition, bit error rates of as low as  $10^{-12}$  were measured at lower bit rates.

At the heart of the system, the quadrature ring oscillator showed very good phase noise performance considering that it does not use high-quality LC tanks. The phase noise and jitter were consistent with simulation and measurements. The VCO's measured tuning range showed accurate consistency to verification simulations. We discovered that due to specific modeling techniques of the extractor, the oscillating frequency did not coincide.

In order to further analyze the results and the performance of the prototype CDR, useful verification methods were developed. These verification methods allowed a more thorough comparison and analysis of results. Based on these observations, future improvements and suggestions were reported in this work.

## 2 TESTING

---

### 2.1 On-Chip Probing Versus Off-Chip Ports

Efficient and simple testing methods are always sought in order to minimize overhead in the final stages of the design flow. In this case, all high-speed signals were probed deli-

cately on-chip; although they have their advantages, this approach was a time-consuming task and requires a lot of expensive infrastructure in place (probes, probe arms, microscope, etc.). Typically, a full test setup would take up to 1 hour and another hour for dismantling it. Given the resources, it is worthy to explore and study the available high-speed packages that can allow the input and output signals to go through the board and SMA or BNC cables rather than on-chip probing. The quality of the extracted signal highly depends on the PCB and package quality.

## **2.2 Demultiplexing and Multiplexing Input/Output**

One popular approach is to design a demultiplexer network, e.g. 16:1 demux network, that time interleaves 16 parallel channels into a NRZ serial channel. This approach is called time division multiplexing (TDM). This implies that the data rate going off the chip runs only at 1/16th of the CDR's system bit rate. This multiplexing network can be implemented in order to output the serial channel to 16 parallel channels, running at 1/16th of the system's bit rate. For example, a 10-Gbps CDR system can be implemented with 16 input and output channels of 625-Mbps bit rate each. Although this block takes up more silicon space, it can alleviate some of the design criteria of the input and output stages, as well as the PCB design. Although this system would have 16 input and output channels, one channel suffices to be connected in order to really test the system's performance and functionality.

## **3 SUGGESTED MODIFICATIONS FOR FUTURE DESIGN**

---

### **3.1 Low-Risk Design Approach**

One of the low-risk approaches adopted in designing this CDR system is the implementation of the switchable bit rate, originally designed for 10, 5, and 2.5-Gbps rates. Also, in order to minimize the risk of degradation of high-speed signals, the input and output ports were designed for on-chip probing. Furthermore, this design choice does not require the complication of adding parallel input and output channels.

The interaction between circuit blocks can arise from a multitude of unexpected

behaviors. The low-risk approach would be to make standalone duplicates of each important circuit blocks for individual testing. This approach, of course, will demand more silicon area, but definitely is an advantage, especially when debugging or testing the system. In this research, the DTTL PFD has been verified in simulation, but it could not be verified for a complete cycle of a PRBS. A stand alone DTTL component such as the DTTL PD and PFD circuit blocks could have been added, in order to observe the behavior over the entire cycle of the PRBS.

### 3.2 Layout Techniques

One of the problems encountered in the CDR system is the substrate and supply feedthrough to the charge pump. Any presence of noise is directly manifested at the output of the charge pump. Furthermore, the entire system is mostly digital, except for the charge pump and the VCO, which are analog circuit blocks. Therefore, any noise or coupling related to these blocks can affect the operation of the system. The VCO can be considered as an amplitude modulation to phase modulation converter, i.e. the VCO generates phase noise proportional to the amount of noise at its control port. There are few techniques that can be applied in order to minimize such effects and obtain optimal results for the quiescent jitter generation.

First of all, it is possible to implement a voltage regulator on-chip at the expense of some silicon area. Motorola has presented a design that can provide a stable voltage on chip for circuits that require large switching of currents [37]. This approach can further minimize the supply and ground bouncing, hence reducing noise coupling through the supplies and grounds of the system.

Substrate noise simulation is not readily available in most circuit simulators. It is therefore very important to layout the system such that sensitive circuit blocks do not couple noise through the substrate. Triple well technology is currently readily available for most CMOS processes. Proper isolation techniques, as described in [38], can provide isolations as low as -120dB.

### 3.3 Alternate VCO Design

As explained in Chapter 5 Section 3.3, the measurement results fall short from simulation in terms of center frequency. The hypothesis is that this discrepancy may come from the common centroid capacitor array at the center of the VCO. This configuration of capacitors may have forced the Diva extractor program beyond its modeling limitations. As a solution, it would be interesting to verify this design with other extractor programs such as Mentor Graphics Calibre XRC upon availability of these tools.

Another approach to circumvent this problem in the future is to design LC-delay oscillators. These oscillators depend strongly on the modeling of inductors: the tools for modeling inductors are quite mature and accurate. Although the phase noise of the designed oscillator is quite impressive for a current-starved ring oscillator, better phase noise and jitter performance can be reached with LC delays or tanks.

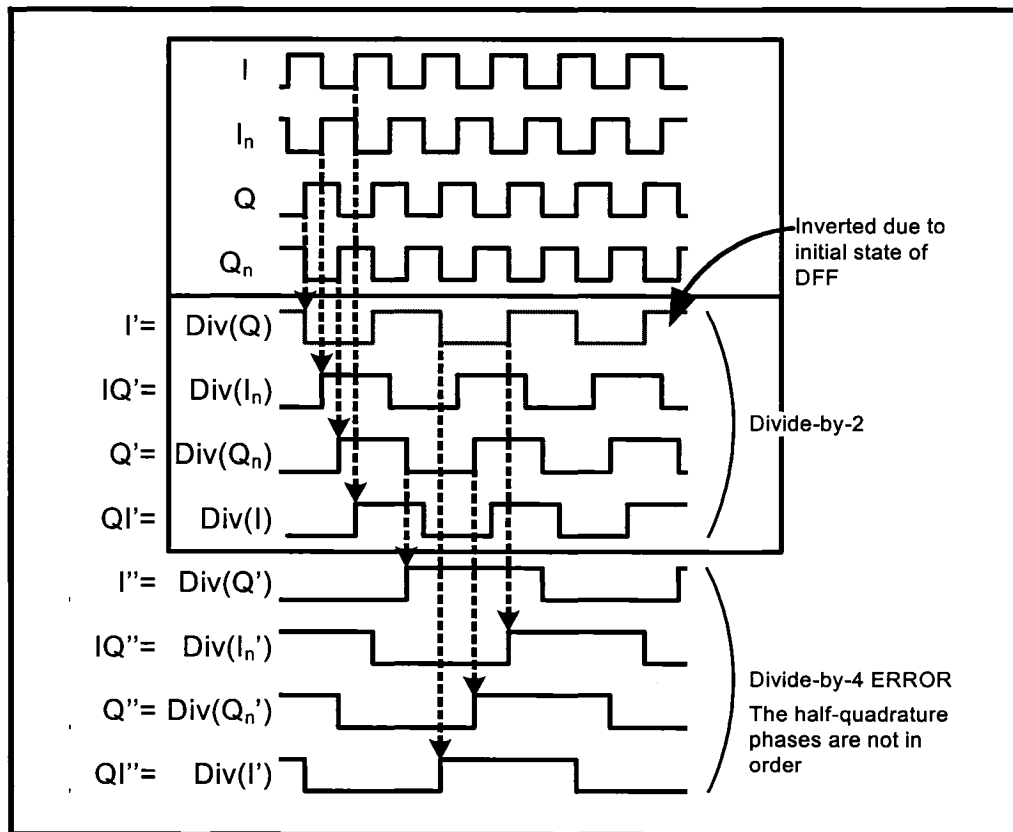
### 3.4 Modification to the 4-Phase Clock Divider

As mentioned earlier, the divider network used was designed in order to scale down the bit rate in the case the system did not perform as expected at high speeds. Figure 6-2 illustrates the problem of the divide-by-4 glitch. Although the divider network has shown to be working during tests and verification, the system may not behave as it was intended to for all possible cases. Fortunately, because the system is entirely double edge-triggered, the divide-by-2 portion of the network is not affected by the memory property of the DFF. On the other hand, the divide-by-4 can be affected by the same problem.

In order to further make this block robust, it suffices to insert set and reset switches for all divide-by-2 circuit blocks. This will guarantee that the initial conditions of the dividers will always be known, and therefore the output will be predictable for all cases.

### 3.5 Use of Foundry Technology

In this research, the CMOS 0.18-micron technology was used because of its availability and its maturity in terms of support and models. However, the speed of this technology is one of the limits for designing robust systems at 10-Gbps data rates.



**Figure 6-2** The flaw in the design of the half-quadrature clock divider network. The error only occurs for the divide-by-4 case, due the memory property of the DFF blocks.

For future related research, it is advisable to explore other technologies such as the CMOS 0.13-micron and the 90-nanometer technology. The tested CDR system was shown to work with a supply of as low as 1.5V. Given this fact, the current design can definitely be migrated to a faster technology without too much concern about voltage scaling.

Taking a different path, the SiGe technology is also of particular future research interest because it is a BiCMOS process. This technology combines the best of both worlds: bipolar transistors provide high  $g_m$  and  $f_T$  for high-speed gates, while CMOS transistors are used for large scale digital circuit integration.

A different angle for future research is to explore the mature 0.18-micron technology process with a 3.3-V supply instead of the nominal 1.8V. Having the extra voltage headroom may alleviate certain design constraints. With this extra degree of freedom, gates can be designed to run faster, and therefore offer a more robust CDR system running at the 10-

Gbps rate.

## 4 SUMMARY

---

In this research, a half-rate binary PFD was implemented using a bottom-up methodology. This study was very useful in terms of verification: the models are proven to be consistent with simulations and measurements, with the exception of the VCO (ring oscillator) for which the hypothetical cause of speed down is a modeling issue by the extractor program. Although the design of the PFD causes the entire system to function marginally, the measurements are consistent with the verified simulations.

The design flow started with the design of basic logic gates in CML. We addressed the pros and cons of adopting CML instead of the popular CMOS logic style. Once the basic logic blocks are constructed with CML style, basic building blocks such as the PD, FD, CP, multiplexors, etc. are implemented in order to construct a complete CDR system.

---

# Appendix

---



---

**1 ABBREVIATIONS**

---

ABM	Analog Behavioral Model
ABS	Absolute
BER	Bit Error Rate
BERT	Bit Error Rate Test
BiCMOS	Bipolar CMOS
BJT	Bipolar Junction Transistor
CAD	Computer Aided Design
CDR	Clock and Data Recovery
CMC	Canadian Microsystems Corporation / Canadian Microelectronics Corporation
CML	Current-Mode Logic
CMOS	Complementary Metal Oxide Semiconductor
CMRR	Common-Mode Rejection Ratio
CP	Charge Pump
CPL	Complimentary Pass Logic
CTC	Cycle-to-Cycle
dB	Decibels
DC	Direct Current
DCVSL	Differential Cascode Voltage Logic Swing
DETDFF	Double Edge-Triggered D-Flip-Flop
DETFE	Double Edge-Triggered D-Flip-Flop
DTMOS	Dynamic Threshold CMOS
DTTL	Data Tracking Transition Loop
ECL	Emitter-Coupled Logic
ETDFF	Edge-Triggered D-Flip-Flop
ETFE	Edge-Triggered D-Flip-Flop
FD	Frequency Detector
FSCL	Folded-Source Mode Logic
Gbps	Gigabits per second
GBWP	Gain-Bandwidth Product
GHz	Gigahertz
GSG	Ground-Signal-Ground
ISI	Inter-Symbol Interference
KHz	Kilohertz
LAN	Local Area Network
LFSR	Linear Feedback Shift Register
mA	milli-amperes
MATLAB	Matrix Laboratory (Software)
MCML	MOS Current Mode Logic
MHz	Megahertz
MOS	Metal Oxide Semiconductor
MOSFET	MOS Field Effect Transistors

MUX	Multiplexer
NMOS	N-Channel MOS
NRZ	Non-Return-to-Zero
PCB	Printed Circuit Board
PD	Phase Detector
PFD	Phase and Frequency Detector
PFD	Probability Density Function
PLL	Phase Locked Loop
PMOS	P-Channel MOS
PRBS	Pseudo-Random Bit Sequence
PSA	Power Spectrum Analyzer
PSK	Phase Shift Keying
RBS	Random Bit Sequence
RMS	Root Mean Squared
RSBMOS	Reduced Supply Bounce CMOS
RSS	Root of Squared Sum
RZ	Return-to-Zero
SATA	Serial Advance Technology Attachment
SGS	Signal-Ground-Signal
SiGe	Silicon Germanium
SNR	Signal-to-Noise Ratio
TSMC	Taiwan Semiconductor Microelectronics Corporation
TV	Television
UGBW	Unity-Gain Bandwidth
UI	Unit Interval
USB	Universal Serial Bus
VCO	Voltage Controlled Oscillator
VCVS	Voltage Controlled Voltage Source

---

## 2 JITTER & EYE DIAGRAM MEASUREMENTS

---

### 2.1 Microsoft Excel

First of all, the same method is used in order to produce a plot that measures the jitter or the quality of an eye diagram. For example, Figure 4-17 was generated with Microsoft Excel. This file has been plotted by outputting the transient time of a free running VCO exported from a Cadence Spectre simulation.

- Run the simulation in march mode in order to produce a space delimited file in column format, such as in Figure 7-1. The retimed data must be perfectly locked in

order for the eye diagram to make any sense. Refer to the cadence manual in order to produce a march file with a delay write feature.

```
nstrips 2
title "Transient Analysis `timeSweep`: time = (0 s -> 10 ns)"
xstart 0.000000e+00
xstop 1.000000e-08
logx 0
logy 0
ydelay 3
ybump 0.1
begin
0.000000000000e+00 1.198559662323e+00 1.237845270145e+00
1.875000000000e-12 1.198559662322e+00 1.237845270145e+00
5.625000000000e-12 1.198559457634e+00 1.237845066901e+00
1.031250000000e-11 1.198557675103e+00 1.237843304109e+00
1.500000000000e-11 1.198551601821e+00 1.237837314024e+00
2.437500000000e-11 1.198538154576e+00 1.237824332136e+00
2.570312500000e-11 1.198536279848e+00 1.237822528648e+00
2.835937500000e-11 1.19853382818e+00 1.237819794005e+00
3.167968750000e-11 1.198532119544e+00 1.237818741648e+00
3.500000000000e-11 1.198534217550e+00 1.237821028224e+00
3.666015625000e-11 1.198536101488e+00 1.237822996672e+00
3.998046875000e-11 1.198547508581e+00 1.237834476204e+00
4.662109375000e-11 1.198599091540e+00 1.237885249841e+00
5.574890749252e-11 1.198702611987e+00 1.237985708644e+00
5.806168061939e-11 1.198736755105e+00 1.238018498130e+00
6.153084030970e-11 1.198794288872e+00 1.238073095415e+00
6.500000000000e-11 1.198855018622e+00 1.238130136514e+00
. . .
end
```

**Figure 7-1** Example of a marchfile with several columns: first one for time and the other(s) for signal value (voltage or current) specified by the simulation setup in Cadence's Analog Environment.

- Open this marchfile with Microsoft Excel. Microsoft Excel will prompt a wizard that will rearrange the file into a spreadsheet table.
- Create an extra column that takes the modulus of the time. The syntax should be something like so  $\text{=mod}(\langle \text{time} \rangle + \langle \text{delay} \rangle, \langle \text{window} \rangle)$ , where  $\langle \text{time} \rangle$  is the cell referring to the transient time,  $\langle \text{delay} \rangle$  is a time value less than a bit period that shifts the eye diagram left and right, and  $\langle \text{window} \rangle$  is the exact display time length that is a multiple of the measured bit period.
- Play with the  $\langle \text{delay} \rangle$  value until the eye diagram is centered.

## 2.2 MATLAB Eye Diagram

Similarly, the eye diagram can be produced with a MATLAB script as shown in Figure 7-2. The data can be loaded using the load, e.g. `load('data.txt')`, command once the march file has been cleaned up, as explained above.

```

function eye_plot(data, window_size, offset)
% Syntax
% =====
%
%     eye_plot(data, window_size, offset)
%
% Description
% =====
%
% This function reads a 2-column data (time, signal) and plots
% an eye diagram given the following parameters.
%
%     windows_size = size of the eye-diagram windows or displayed
%                   time interval.
%     offset = the time offset value that shifts the eye
%              diagram sideways.
%
% =====
% Created by David S. Hong
% July 28, 2003
% =====

figure;
hold on;

% Divide the transient into eye-segments
seg = mod(data(:,1)-offset,window_size);

% Find the zero crossings of the eye diagram
index = find(floor(diff(seg)));

% plot the data
for k = 1:length(index)-1
    plot( seg(index(k)+1:index(k+1)), data(index(k)+1:index(k+1),2));
end;
hold off;

```

**Figure 7-2** MATLAB script for generating an eye diagram plot given the data matrix (time and signal columns), eye plot window size, and offset.

### 3 PRBS GENERATION IN MATLAB

Figure 7-3 and Figure 7-4 are the scripts in MATLAB along with their descriptions of the functions in order to generate a PRBS by imitating an LFSR: the first script (Figure 7-3) generates the binary sequence, and the second script (Figure 7-4) generates the PRBS transient file given the binary sequence. In order to produce a maximal length sequence, i.e. toggle through all possible combination of bit pattern, Table 7-1 describes the corresponding tap configuration given the LFSR's size.

**Table 7-1** Tabulation of the corresponding taps for maximal length sequences given the number of LFSR

LFSR Size (m)	Maximal Sequence	Taps
9	511	3, 8
10	1,023	2, 9
11	2,047	1, 10

**Table 7-1** Tabulation of the corresponding taps for maximal length sequences given the number of LFSR

LFSR Size (m)	Maximal Sequence	Taps
12	4,096	0, 3, 5, 11
13	8,191	0, 2, 3, 12
14	16,383	0, 2, 4, 13
15	32,767	0, 14
16	64,535	1, 2, 4, 15
17	131,071	2, 16
18	262,143	6, 17
19	524,287	0, 1, 4, 18
20	1,048,575	2, 19
21	2,097,151	1, 20
22	4,194,303	0, 21
23	8,388,607	4, 22

Figure 7-4 is the MATLAB script with descriptions of its function to generate a piecewise linear file (PWLF) for integrating this signal to a Cadence simulation. This function takes in the information of amplitude, rise and fall times, the bit rate, and the binary sequence to generate a PWLF file – a text file that contains a column for time and another for corresponding voltage.

## **4 VERILOG-A CODES FOR THE RING OSCILLATOR COMPONENTS**

### **4.1 Ideal Continuous-Time (Analog) Delay Block**

This code has been generated by a model writer – a synthesis tool that generates the corresponding Verilog-A code given the type of module and the proper parameters; in this case frequency gain, center of oscillation (0V), and output amplitude. The code is shown in Figure 7-5.

```

function seq = prbs2(srs,length_srs,taps)
% Generated pseudo-random sequence (version 2)
%
% seq = prbs2(srs,length,taps)
% seq = output file with PRBS bit sequence
% srs = number of shift registers, N
% length = number of bits in the sequence
% taps = vector indicating the taps
%
% This function generates a sequence of bit
% pattern that imitates the PRBS of a linear
% shift register with 'srs' latches and xnor
% with taps at the specified vector given by
% the variable 'tap'.
%
% The 'seq' variable is created with the PRBS
% pattern with 'length'-bit long.
%
% * Modified by David Hong, originally created by
% * Francis Beaudoin, 2003 - 2005.

% Size of shift register
% and set the sequence 0 0 0 ... 1
r = [zeros(1,srs-1) 1];

% Shifting process
for k = 1: length_srs
    for p = 1: length(taps)-1
        % Compute taps
        newbit = xor(r(taps(p)+1), r(taps(p+1)+1));
    end
    % Push and Shift
    r = [not(newbit) r];
    clear p;
end

% Take the bit sequence other than the first 'srs' bits
% and transpose the matrix
seq = r(srs+1:end)';

```

**Figure 7-3** MATLAB script for generating an LFSR sequence given the number of serial shift registers (srs), length of sequence (length\_srs), and tap configuration (taps).

```

function [] = seq2pwlf(seq, period, rise, A)
% seq2pwlf(sequence, period, rise, amplitude)
%
% This function generates a 'temp.txt' file with time
% and voltage column given the following variables.
% sequence = the bit sequence variable
% period = time span of 2 consecutive bits in seconds
% risefall_time = rise time and fall time in seconds
% amplitude = signal amplitude of the transient sequence in Volts
%
% * Modified by David S. Hong, Originally created by
% * Francis Beaudoin, 2002 - 2003

% Set the initial size of the 'out' variable
out = zeros(2*(length(seq)), 2);

for k = [1 : length(seq)-1]
    out( 2*k-1, : ) = [k*period/2-rise seq(k)];
    out( 2*k, : ) = [ k*period/2 seq(k+1)];
end;

out = out(1:end-2,:);
out(:,2) = A*(2*out(:,2)-1);

save temp.txt out -ASCII

```

**Figure 7-4** MATLAB function script “seq2pwlf.m.” This function takes a binary sequence and generates a piece-wise linear file (PWLF) in order to generate a PRBS in Cadence.

```
// FUNCTION: Delay Element
// VERSION: $Revision: 2.5 $
// AUTHORS: Modelwriter Standard Library
//          Cadence Design Systems, Inc.
//
// GENERATED BY: Cadence Modelwriter 2.24
//              ON: Mon Mar 21 17:21:58 EST 2005
//
// Description: Analog Delay Element
//
// Parameters:
//      td - delay time from input to output

`include "discipline.h"
`include "constants.h"

module delay_elem (vin, vout);
    input vin;
    output vout;
    electrical vin, vout;

    parameter real td = 41.666666666666p from (0:inf);

    analog begin
        V(vout) <+ absdelay(V(vin), td );
    end
endmodule
```

**Figure 7-5 Spectres' Verilog-A code provided by the ModelBuilder for the delay component.**

## 4.2 Ideal Single-Phase Voltage Controlled Oscillator (VCO)

This code has been generated by a model writer – a synthesis tool that generates the corresponding Verilog-A code, given the type of module and the proper parameters; in this case frequency gain, center of oscillation (0V), and output amplitude. The code is shown in Figure 7-6.

## 5 RING OSCILLATOR SUPPLEMENTARY RESULTS

The ring oscillator has been measured over different biasing points in order to be able to shift the tuning range, in case of any error. The ring oscillator's result were not expected to be that far off the simulated ones. However, the different transfer and gain curves have been measured for verification.

The frequency versus control voltage are measured in Figure 7-7. The corresponding gains of these curves are displayed in Figure 7-8. In all measurements, the voltage supply of the ring oscillator was set to approximately 1.5V.

```

// FUNCTION: VCO
// VERSION: $Revision: 2.6 $
// AUTHORS: Modelwriter Standard Library
//          Cadence Design Systems, Inc.
// GENERATED BY: Cadence Modelwriter 2.24
// ON: Mon Mar 21 15:15:30 EST 2005
// Description: Voltage Controlled Oscillator
// vin:frequency control voltage [V,A]
// vout: sine wave output [V,A]
// PARAMETERS:
// vco_amp = Sinewave output amplitude [V]
// vco_cf = Output Frequency for Zero Control Voltage [Hz]
// vco_gain = Frequency shift per volt of Control signal change
// [Hz/Volt]
// vco_ppc = Limit simulator timestep to calculate N points per
// cycle
//
`include "discipline.h"
`include "constants.h"

module vco(vin, vout);
input vin;
output vout;
electrical vin, vout;
parameter real vco_amp = 1.0 from (0:inf);
parameter real vco_cf = 3.238G from (0:inf);
parameter real vco_gain = -280M exclude 0.0;
parameter integer vco_ppc = 40 from [4:inf];

real wc; // center freq in rad/s
real phase_lin; // wc*time component of phase
real phase_nonlin; // the idt(k*f(t)) of phase
integer num_cycles; // number of cycles in linear phase component
real inst_freq; // instantaneous frequency

analog begin

  @ ( initial step ) begin
    wc = `M_TWO_PI * vco_cf;
  end

  //
  // linear portion is calculated so that it remains in the +/- 2`PI range
  // This is to ensure it's value doesn't get too large and cause rounding
  // problems for calculation of the phase.
  //
  phase_lin = wc * $abstime;
  num_cycles = phase_lin / `M_TWO_PI;
  phase_lin = phase_lin - num_cycles * `M_TWO_PI;

  phase_nonlin = `M_TWO_PI * vco_gain * idtmod ( V(vin), 0, 1000.0, 0.0);

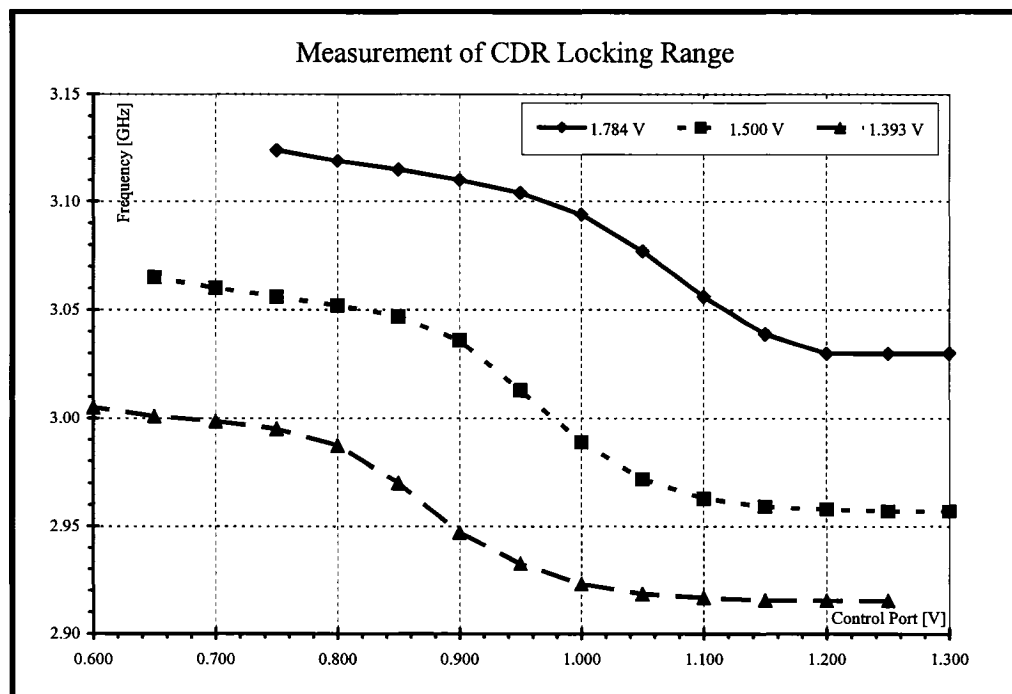
  V(vout) <+ vco_amp * sin (phase_lin + phase_nonlin);

  //
  // ensure that modulator output recalculated soon.
  //
  inst_freq = vco_cf + vco_gain * V(vin);
  $bound_step (1/(vco_ppc * inst_freq));
end
endmodule

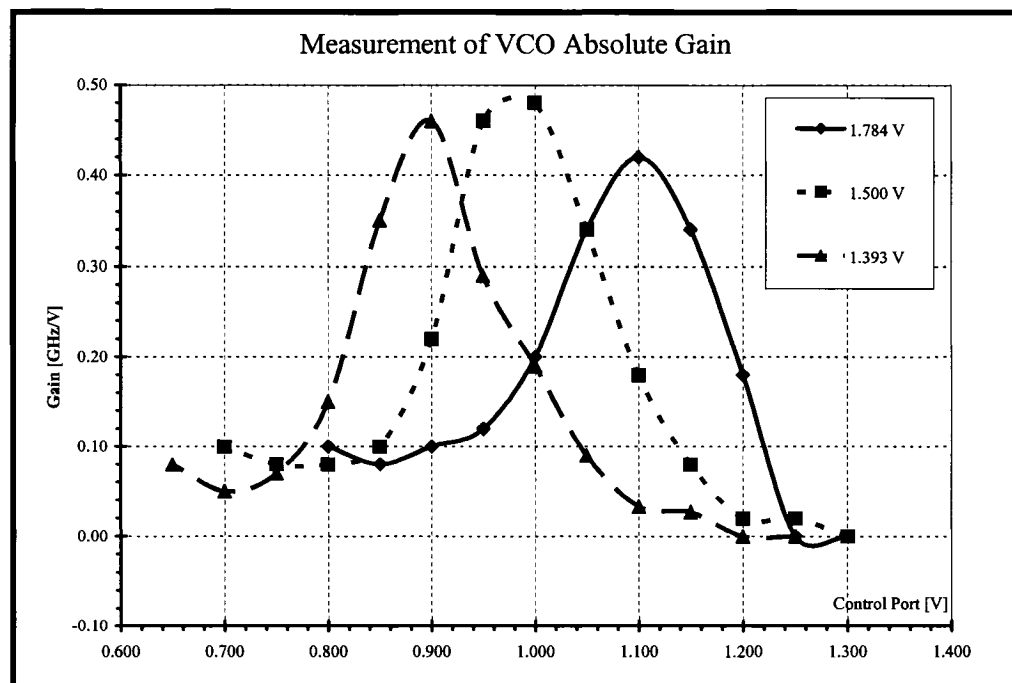
```

**Figure 7-6 Spectres' Verilog-A code provided by the ModelBuilder for the VCO component.**





**Figure 7-7** Measurement of the half-quadrature ring oscillator's frequency versus control voltage for various external coarse tuning voltages.



**Figure 7-8** Gain of the half-rate ring oscillator for various external coarse tuning voltages.

---

# References

---

- [1] K. Murata and T. Otsuji, "A Novel Clock Recovery Circuit for Fully Monolithic Integration," *IEEE T. on microwave theory and techniques*, vol. 47, no. 12, pp. 2528-2533, November 1999.
- [2] R. Walker, "Clock and Data Recovery for Serial Digital Communication: focusing on bang-bang loop CDR design methodology," *ISSCC Short Course*, February 2002.
- [3] K. Schrödinger, J. Stimma, and M. Mauthe, "A Fully Integrated CMOS Receiver Front-End for optic Gigabit Ethernet," *IEEE J. Solid-State Circuits*, vol. 37, pp. 874-880, July 2002.
- [4] B. Razavi, "Prospects of CMOS Technology for High-Speed Optical Communication Circuits," *IEEE J. Solid-State Circuits*, vol. 37, no. 9, pp. 1135-1145, March 2002.
- [5] S.-H. Lee, M.-S. Hwang, Y. Choi, S. Kim, Y. Moon, B.-J. Lee, D.-K. Jeong, W. Kim, Y.-J. Park, and G. Ahn, "A 5-Gb/s 0.25- $\mu$ m CMOS Jitter-Tolerant Variable-Interval Oversampling Clock/Data Recovery Circuit," *IEEE J. Solid-State Circuits*, vol. 37, no. 12, pp. 1822-1830, December 2002.
- [6] H. Djahanshahi, and A. T. Salama, "Differential CMOS Circuits for 622-MHz/933-MHz Clock and Data Recovery Applications," *IEEE J. Solid-State Circuits*, vol. 35, no. 6, pp. 847-855, June 2000.
- [7] J. Cao, M. Green, A. Momtaz, K. Vakilian, D. Chung, K.-C. Jen, M. Caresosa, X. Wang, W.-G. Tan, Y. Cai, I. Fujimori, and A. Hairapetian, "OC-192 Transmitter and Receiver in Standard 0.18- $\mu$ m CMOS," *IEEE J. Solid-State Circuits*, vol. 37, no. 12, pp. 1768-1780, December 2002.
- [8] J. E. Rogers and J. R. Long, "A 10-Gb/s CDR/DEMUX with LC Delay Line VCO in 0.18- $\mu$ m CMOS," *IEEE J. Solid-State Circuits*, vol. 37, no. 12, pp. 1781-1789, December 2002.
- [9] Internet Growth Statistics: Today's Road to e-Commerce and Global Trade," Internet World Stats: usage and population statistics. Accessed March 2005. <<http://www.internetworldstats.com/emarketing.htm>>.
- [10] "Nielsen's Law of Internet Bandwidth." Usit: Alert Box. Accessed March 2005. <<http://www.useit.com/alertbox/980405.html>>.

- [11] J. Savoy and B. Razavi, "A 10-Gb/s CMOS Clock and Data Recovery Circuit with a Half-Rate Binary Phase/Frequency Detector," *IEEE J. Solid-State Circuits*, vol. 38, pp. 13-21, January 2003.
- [12] A. X. Widmer and P. A. Franaszek, "A DC-Balanced, Partitioned-Block, 8B/10B Transmission Code," *IBM J. Res. and Develop.*, vol. 2, pp. 440-451, Sept. 1983.
- [13] R. C. Walker and R. Dugan, "Low Overhead Coding Proposal for 10Gb/s Serial Links," *IEEE 802.3 High-Speed Study Group*, Nov. 1999, [http://group-per.ieee.org/groups/802/3/10G\\_study/public/nov99/walker\\_1\\_1199.pdf](http://group-per.ieee.org/groups/802/3/10G_study/public/nov99/walker_1_1199.pdf).
- [14] B. Razavi, "Design of Integrated Circuits for Optical Communications", McGraw Hill, 2003.
- [15] F. Beaudoin, "Design and Implementation of a Gigabit-Rate Optical Receiver and a Digital Frequency-Locked Loop for Phase-Locked Loop Based Applications," Masters thesis, McGill University, 2003.
- [16] F. Herzel, B. Razavi, "A study of Oscillator Jitter Due to Supply and Substrate Noise," *IEEE Trans. Circuits and Systems II: Analog and Digital Signal Processing*, vol. 46, no. 1, pp. 56 - 62, Jan. 1999.
- [17] Alfredsson, J. & Oelmann, B., "Trading Speed and Power for Reduced Substrate Noise from Digital CMOS Circuits", *Proceedings of International Conference on Signals and Electronic Systems*, ICSES'04, Poznan, Poland, 13-15 September 2004.
- [18] J. Musicer and J. Rabaey, "MOS Current Mode Logic for Low Power, Low Noise CORDIC Computation in Mixed-Signal Environments", *Proc. ISPLPED*, pp. 102-107, July 2000.
- [19] M.W. Allam and M. I. Elmasry, "Dynamic Current Mode Logic (DyCML), A New Low-Power High Performance Logic Style," *IEEE Journal of Solid State Circuits*, Vol. 36. No.3, March 2001, pp.550-558.
- [20] J. Rabaey, "Digital Integrated Circuits: A Design Perspective", Prentice Hall, 1996
- [21] M. Anis, M. Allam and M.Elmasry, "Impact of Technology Scaling on CMOS Logic Styles," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, Vol. 49, No. 8, August 2002, pp. 577-588.

- [22] H. Hassan, M. Anis, and M. Elmasry, "Design and Optimization of MOS Current Mode Circuits for Parameter Variations", *Integration, the VLSI Journal*, Vol. 38, no. 3, January 2005, pp. 417-437.
- [23] C. R. Hogge, "A Self-Correcting Clock Recovery Circuit," *IEEE J. Lightwave Tech.* vol. 3, pp. 1312-1314, Dec. 1985.
- [24] J. D. H. Alexander, "Clock Recover Architecture and Monolithic Implementation," *Electronics Letters*, vol. 11, pp. 541-542, Oct. 1975.
- [25] A. Hajimiri, A. Limotyrakis, T. H. Lee, "Jitter and Phase Noise in Ring Oscillators." *IEEE J. Solid-State Circuits*, vol. 34, no. 6, pp. 790 - 804, June 1999.
- [26] J. Savoj and B. Razavi, "A 10-Gb/s CMOS clock and data recovery circuit with a half-rate linear phase detector," *IEEE J. Solid-State Circuits*, vol. 36, pp. 761-768, May 2001.
- [27] M. Rau, T. Oberst, R. Lares, A. Rothermel, R. Schweer, and N. Menoux, "Clock/Data recovery PLL using Half-Frequency Clock," *IEEE J. Solid-State Circuits*, vol. 32, no. 7, pp. 1156-1159, July 1997.
- [28] Ramezani, M.; Andre, C.; Salama, T., "Analysis of a half-rate bang-bang phase-locked-loop," *IEEE Transactions on [see also Circuits and Systems II: Express Briefs, IEEE Transactions on]*, vol. 49, no. 7, pp. 505- 509, July 2002.
- [29] Savoj, J., Razavi, B., "Design of half-rate clock and data recovery circuits for optical communication systems," *Design Automation Conference, 2001. Proceedings*, pp. 121 - 126, 2001.
- [30] Larsson, P., "An offset-cancelled CMOS clock-recovery/demux with a half-rate linear phase detector for 2.5 Gb/s optical communication," *Solid-State Circuits Conference, 2001. Digest of Technical Papers. ISSCC. 2001 IEEE International*, pp. 74 - 75, 434, 5 - 7 Feb. 2001.
- [31] H. Nosaka, K. Ishii, T. Enoki, and T. Shibata, "A 10-Gb/s Data-Pattern Independent Clock and Data Recovery Circuit With a Two-Mode Phase Comparator," *IEEE J. Solid-State Circuits*, vol. 38, no. 2, pp. 192-197, February 2003
- [32] D. Richman, "Color-Carrier Reference Phase Synchronization Accuracy in NTSC Color Television." *Proc. IRE*, vol. 42, pp. 106-133, Jan. 1954.
- [33] J. A. McNeil, "Jitter in Ring Oscillators," *IEEE J. Solid-State Circuits*, vol. 32, no. 6, pp. 870-879, June 1997.

- [34] E. A. M. Klumperink, S. L. J. Gierkink, A. P. van der Wel, B. Nauta, B. "Reducing MOSFET 1/f Noise and Power Consumption by Switched Biasing." *Solid-State Circuits, IEEE J.*, vol. 35, no. 7, pp. 994 - 1001, July 2000.
- [35] S. Mohan, M. Hershenson, S. Boyd, and T. Lee, "Bandwidth Extension in CMOS with Optimized On-Chip Inductors," *IEEE J. Solid-State Circuits*, vol. 35, pp. 346-355, Mar. 2000.
- [36] REG1117, REG1117A Burr-Brown from Texas Instruments. "800mA and 1 A Low Dropout Positive Regulator 1.8V, 2.5V, 2.85, 3.3V, 5V, and Adjustable." Texas: Texas Instruments Incorporated, October 1992, revised July 2004.
- [37] L. Connell, N. Hollenbeck, M. Bushman, D. McCarthy, S. Bergstedt, R. Cieslak, and J. Caldwell, "A CMOS Broadband Tuner IC", *IEEE International Solid-State Circuits Conference*, vol. 45, pp. 400-401, February 2002.
- [38] Tallis Blalack, Youri Leclercq, and Patrick Yue. "On-Chip RF-Isolation Techniques." TechOnLine. Cadence Design Systems and Atheros Communications. February 2005 <[http://www.techonline.com/community/ed\\_resource/feature\\_article/21590](http://www.techonline.com/community/ed_resource/feature_article/21590)>
- [39] Ayman Ibrahim Refaat Ahmed. "RF Frequency Synthesizers." Thesis of Ain Shams University, Faculty of Engineer: Electronics and Communications Engineering Department. 1994.
- [40] Ramin Shariat-Yazdi. "Mixed Signal Design Flow: A mixed signal PLL case study." Thesis of University of Waterloo, Electrical & Computer Engineering, Waterloo, Ontario, Canada, 2001.
- [41] Dan Fitzpatrick, Ira Miller. "Analog Behavioral Modeling with the Verilog-A Language." Kluwer Academic Publishers. Third printing, printed in United States of America 2003.
- [42] J. D. van der Tang, D. Kasperkovitz, and A. van Roermund, "A 9.8-11.5-GHz Quadrature Ring Oscillator for Optical Receivers," *IEEE J. Solid-State Circuits*, vol. 37, no. 3, pp. 438-442, March 2002.