SINGLE CHIP HIGH-SPEED SERIAL LINK COMMUNICATIONS FOR MULTI-CHANNEL AND MULTI-STANDARD APPLICATIONS

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Ming-ta Hsieh

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Dr. Gerald E. Sobelman, Advisor

November 2008

Acknowledgements

I would first like to thank my wife, Chia-hui Hung, for her love, understanding, and support over years of study. My deepest thanks are to Dr. Gerald E. Sobelman, my advisor, for his professionalism and patience. I am motivated by his strong work ethic, his creativity and his encouragement. Finally, many thanks are to my many supportive colleagues and managers at Toshiba America Electronic Components (TAEC), Inc. The last thanks are to my great neighbor and former colleague, Ms. Judith Teumer, for her great assistance on editing this dissertation.

Dedication

This study is dedicated to my late parents and my beloved family.

ABSTRACT

This thesis presents new design concepts for high-speed serial link systems, including those for standards such as Serial AT Attachment (SATA), Serial Attached SCSI (SAS), Peripheral Component Interconnect Express (PCIe) and Fibre Channel (FC). The research work also provides for co-existing operation of multi-standards in a single chip for multi-channel, serial link communications systems. A high-speed serial link starts with low-frequency parallel data sampled by a synthesized clock and serialized into a stream of data. The serialized data is then pre-emphasized and transmitted to the receiver through a bandwidth-limited channel. The receiver equalizes the received signal in order to compensate the high frequency spectrum loss before extracting the clock and data in the Clock and Data Recovery (CDR) loop. Next, the extracted data is sampled by the recovered clock and deserialized back into a parallel set of signals. The complete system for serializing / deserializing data transmission from transmitter (Tx) to receiver (Rx) is known as a SerDes. This research starts with a system-level performance assessment, using a top-down behavioral- to transistor-level design approach and a bottom-up transistor- to behavioral-level verification procedure. Clock synthesizer design is studied and a new approach for a spread spectrum clock generator is proposed. Then, the design of a system with multi-level, bi-directional signaling is investigated. Finally, the study of high-speed CDRs for a variety of applications is presented. The design consists of 4 SerDes subsystems with three clock synthesizers, using two ring VCO based PLLs and one LC VCO based PLL. This design demonstrates the simultaneous operation of a multi-rate, multi-standard, multi-channel serial communication system in a single chip.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

1 INTRODUCTION

Serial link systems have gradually dominated over parallel link systems in modern high-speed data link communications. The use of differential signal serial communications prolongs the length of the data transmission channels, which parallel communications can not match due to the signal degradation effects caused by, for example, crosstalk among parallel link wires. In addition, the maximum tolerable skew among the parallel link wires limits their maximum allowable data transmission speed. This chapter provides an introduction to serial link systems and also gives an outline of the thesis, which is devoted to the development of single chip, high-speed, serial link communications techniques for multi-channel and multi-standard applications.

1.1 INTRODUCTION OF SERIAL LINK SYSTEMS

High-speed serial data link communications are widely used in many applications and they continue to replace more traditional parallel data link systems such as the Integrated Drive Electronics (IDE) hard disk connectors [1], AT Attachment with Packet Interface (ATA/ATAPI) [2],\ and bi-directional parallel communications (IEEE-1284) [3]. The basic difference between a parallel and a serial communication channel is the number of designated physical wires used as transmission channels for transferring data between two devices. Parallel communications implies multiple physical wires, in addition to a ground connection. Serial communication often refers to only two designated wires used for a differential signal transmission without an additional ground connection between

two devices. Another difference between parallel and serial links is that parallel link systems usually run in a synchronous mode, while serial links often run in an asynchronous mode such that data transmission between two devices is allowed to have a frequency offset and so that there is no need to use the same reference clock source.

Before the development of high-speed serial link communication technologies, the preference of parallel links over serial links was driven by the misconception of their mathematical speed. The speed of a parallel data link system is equivalent to the bit rate of each individual wire multiplied by the number of parallel transmission wires. In practical designs, the skew among the parallel wires reduces the speed of data transmission to the slowest of all of the parallel transmission wires. Second, crosstalk creates interference between the parallel signal wires and places an upper limit on the length of a parallel data transmission, which is usually shorter than the length of a serial link system. However, parallel data links are easier to implement in hardware; designing a parallel link system is relatively simple, requiring only latches to transfer data from parallel wires onto the data bus. On the other hand, the data for most serial link communications must first be converted from parallel form into serial format before they can be transmitted, and then converted back into parallel form after they have been received and before they can be directly connected to a data bus.

The decreasing cost of integrated circuits, combined with greater consumer demand for higher speed and longer distance, has led to the replacement of parallel links by serial ones; for example, IEEE 1284 bi-directional parallel communications vs. Universal Serial Bus (USB) [4], and ATA vs. Serial AT Attachment (SATA) [5]. There are three main types of serial communication channels existing in contemporary high-speed serial links:

chip-to-chip connections between different Application-Specific Integrated Circuits (ASICs) or standard components on the same board; board-to-board through a backplane for system connectivity inside the rack; and chassis-to-chassis through a cable to connect or expand equipment.

A high-speed serial link starts with low-frequency parallel data sampled by a synthesized clock and serialized into a stream of data. The serialized data is then pre-emphasized and transmitted to the receiver through a bandwidth-limited channel. The receiver needs to equalize the received signal in order to compensate the high frequency spectrum loss before extracting the clock and data using a clock and data recovery (CDR) loop. Next, the extracted data is sampled by the recovered clock and deserialized back into a set of parallel signals. The design for serializing / deserializing data between transmitter and receiver is also known as a SerDes [6].

1.2 GOALS AND OBJECTIVES

From the viewpoint of cost effectiveness, a single chip design having a performance satisfying multi-channel and multi-standard requirements is the best choice to maximize the use of design. However, its major drawback is the increase in its design complexity. The goals and objectives of a multi-channel, multi-standard serial link are:

(a) A system-on-chip solution.

(b) Supporting a wide range of data rates.

(c) Capable of running multiple data rates concurrently among multiple channels.

(d) Reconfigurable numbers of data transmission channels and clock generators / synthesizers.

(e) Accommodating all specification requirements for the multiple standards in each of

the multiple channels.

Figure 1.1 shows the block diagram of the proposed single-chip, multi-channel serial link design for multi-standard applications. The design includes four clock generators / synthesizers and 4 pairs of transmitters (Tx) and receivers (Rx) in each chip. Each of 4 pairs of Tx-Rx is able to access each of 4 clock generators / synthesizers, which enables multi-rate operation to co-exist in a multi-channel SerDes. Two clock generators / synthesizers on each side can be driven by the same or separate reference clock inputs. The intended applications are for multi-generation of SATA, SAS, PCI Express and Fiber Channel applications.

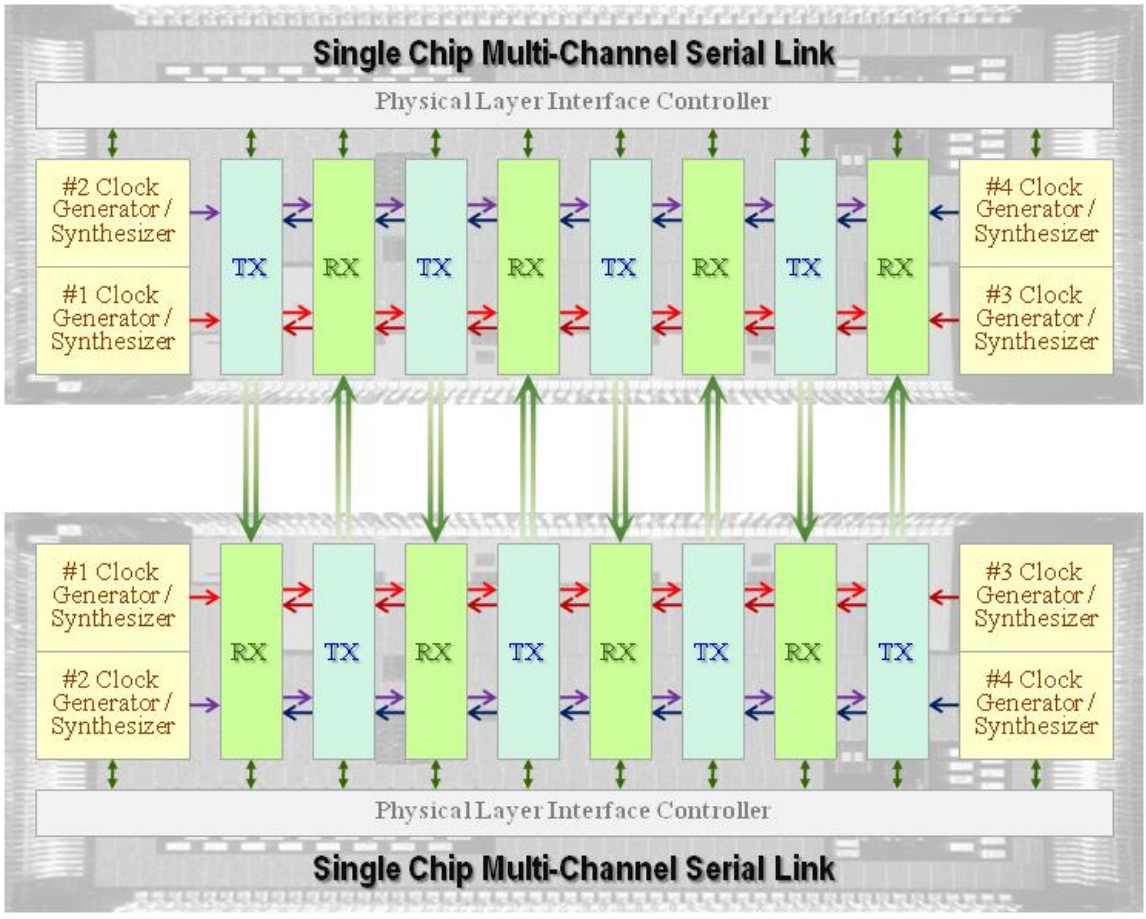


Figure 1.1 *Block diagram of the single chip multi-channel serial link design for multi-standard applications.*

In the following chapters, new results are given on serial link system modeling and performance prediction, clock generation / synthesis, data transmission with adaptive pre-emphasis control, and CDR architectures. Design strategies for single-chip, high-speed serial links for multi-channel and multi-standard applications may be developed based on the system and circuit concepts developed in this dissertation.

Chapter 2 discusses serial link system modeling and verification of high-speed wired links with Verilog-AMS. Non-ideal input clock and data generation and modeling examples are discussed. System and circuit designs with virtual built-in self-test (VBIST) using Verilog-AMS is proposed to speed up the design and verification processes.

Chapter 3 extends the serial link modeling of Chapter 2 to develop accurate prediction of jitter tolerance in high-speed serial links. The primary focus is on modeling clock and data recovery jitter tolerance in the receiver.

Chapter 4 investigates clock multiplication / synthesis with a comparison between Ring and LC VCO PLL designs. VCO topologies and tradeoffs are studied. Furthermore, the power dissipation, design layout area and performance of these designs are also investigated. The study provides an analytical framework for determining the best choice for a high-speed clock synthesizer design based on the constraints of the target application.

Chapter 5 proposes a low jitter, high EMI reduction, programmable spread spectrum clock (SSC) generator for multi-channel SerDes applications. The SSC clock generator utilizes a phase interpolator between the VCO output and the PLL feedback divider input to control the magnitude and frequency of clock spreading.

Chapter 6 proposes a simultaneous bi-directional PAM-4 link with built-in self-test (BIST) to adaptively adjust the level of pre-emphasis in the transmitter.

Chapter 7 studies the architecture for multi-gigabit wire-linked clock and data recovery. This chapter presents an overview and comparative study of the most commonly used CDR architectures. This analysis includes the circuit structures, design challenges, major performance limitations, and primary applications. Finally, the tradeoffs among the various CDR architectures are summarized.

Chapter 8 proposes a clock and data recovery (CDR) with adaptive loop gain for spread spectrum SerDes applications such as the Serial AT Attachment (SATA) [5]. The CDR design can be implemented in a digital CMOS process that reduces the design difficulty and cost.

Chapter 9 provides a summary of the work presented and scope for future work.

REFERENCES

[1] D. A. Sanders, et al., "Redundant Arrays of IDE Drives," *IEEE Transactions on Nuclear Science*, Vol.49, No.4, pp.1834-1840, August 2003.

[2] H.-K. Kim, et al., "A Single-Chip Storage LSI for Home Networks," *IEEE Communication Magazine*, Vol.43, Issue.5, pp.141-148, May 2005

[3] IEEE-SA Standards Board, "IEEE Standard for Data Delivery and Logical Channels for IEEE 1284 Interfaces," *The Institute of Electrical and Electronics Engineers, Inc.*, June 2000.

[4] A. Depari, et al., "USB Sensor Network for Industrial Applications," *IEEE Transaction on Instrumentation and Measurement,* Vol.57, no.7, pp.1344-1349,

July 2008.

[5] Serial ATA Workgroup, “Serial ATA,” Revision 2.6, May 2007.

[6] M. Sugawara et al., “1.5 Gbps, 5150 ppm SPREAD Spectrum SerDes PHY with a 0.3mW, 1.5 Gbps Level Detector for Serial ATA,” *Dig. Sym. VLSI Circuits*, pp. 60-63, Jun. 2002.

CHAPTER 2

2 MODELING AND VERIFICATION OF HIGH-SPEED WIRED LINKS WITH VERILOG-AMS

Behavioral modeling with virtual built-in self-test verification of high-speed wired link designs is described in this chapter. Our procedure is based on principles of top-down mixed-signal design combined with a behavioral description language and mixed-mode simulations. The use of Verilog-AMS is applied not only to circuit modeling but also for representing noise on the input signal. This approach provides system-level jitter tolerance estimation, circuit critical path search and overall design verification. Coding examples and simulation results are included.

2.1 INTRODUCTION

The idea of top-down mixed-signal circuit design using a behavioral description for system-level modeling has been a topic of research interest [1-3]. Typically, system architecture designers derive the required system or circuit specification for certain targeted applications. On the other hand, circuit designers would search for a type of circuit that is the best fit for the required specifications with minimal design cost and time and superior performance. Before going ahead with the detailed transistor-level design, behavioral models are usually used to verify the design at the system level in order to determine which type of circuit has the best performance. During the design process, transistor-level circuits may be mixed with behavioral models for performance evaluation. After completing each transistor-level circuit design, the corresponding

behavioral circuit for each module is tweaked to match to its transistor-level circuit performance. By the end of the design process, a more thorough system-level interconnection check and performance verification are executed. This final system–level verification is done based on either mixed-mode or fully behavioral simulations. The mixed-mode simulation includes both transistor and behavioral modules, which have their critical path delays based on transistor-level extraction. Of course, an accurate behavioral model is needed in order to provide useful information for the total system-level verification.

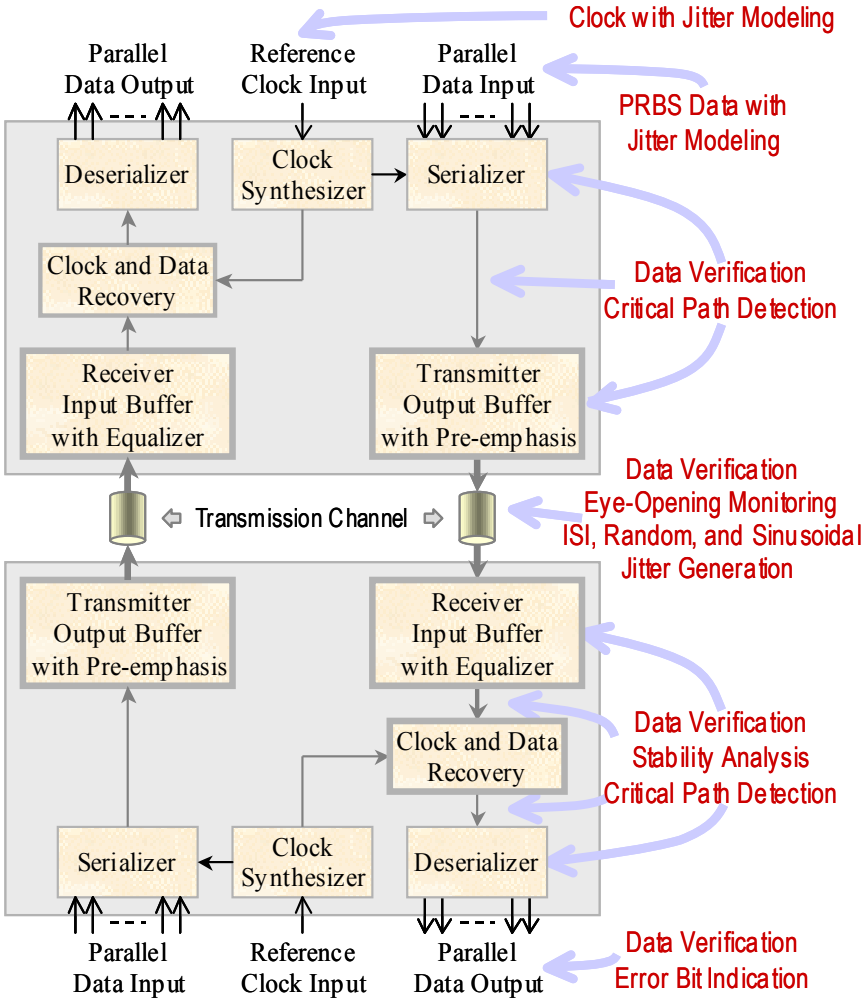


Figure 2.1 Application of behavioral modeling to high-speed link design.

A complete transistor-level system performance evaluation is not practical due to the enormous simulation time that would be required [4]. Using behavioral modeling lets designers easily explore different system-level architectures at the early design stages and thereby rapidly perform system-level verification. Thus, including behavioral models in the overall design process reduces time to market and helps ensure first time correct silicon. Figure 2.1 shows an example of the use of behavioral circuit models applied to high-speed wired link designs. The behavioral models may also be used to generate inputs such as clocks or PRBS data associated with jitter, to search for internal critical paths, to verify correctness of data transmission with virtual build-in-self-test. The ultimate goal is not only to use the behavioral description to model circuits for system-level verification, but also to identify the weak points in a system for subsequent analysis.

2.2 INPUT SIGNAL MODELING

In the real world, inputs are never ideal and are always associated with some kind of noise. In the time domain, the noise on clocks or input data is known as jitter. This section demonstrates the use of behavioral modeling to generate these non-ideal signals. The clock synthesizer module, as shown in Figure 2.1, often requires a reference clock input.

2.2.1 CLOCK WITH JITTER MODELING

This clock input normally contains a certain amount of random jitter (RJ). [5] This RJ information is needed when performing system-level verification or clock-synthesizer design in a high-speed wired link. For the other individual module designs, such as the serializer and clock data recovery (CDR) of Figure 2.1, the major concern is the impact

of jitter from the output clock of the synthesizer. Reference [6] provides a detailed analysis of this impact in high-speed serial links. An example of a clock with RJ modeling is shown in Figure 2.2. RJ typically has a Gaussian distribution. The magnitude of RJ is a function of the desired bit error rate (BER) for the system. [7] The expected jitter magnitude and time interval for the RJ distribution are taken as inputs to generate the RJ in the example below.

```
.........
@ ( initial_step ) begin
  next = $abstime; // next time
  vout = 0; // output logic
  randseed = 1; // random number seed
end
$bound_step(tp); // bounded step
tj = $rdist_normal(randseed, 0, tj_sd); // Gaussian Jitter
@ ( timer( next ) ) begin
  vout = !vout;
  tj = tj * ( ( abs(tj) >= tj_min ) && ( abs(tj) <= tj_max ) )
     + tj_min * ( abs(tj) < tj_min )
     + tj_max * ( abs(tj) > tj_max ); // limit jitter in allowed range
  tx = td + 10 * tp + tj; // clock edge time
  next = next + tp; // next time
end
V(clkip) <+ transition(vh* vout+vl*!vout,tx+tp*0.0,tr,tf ); // clkip
V(clkin) <+ transition(vh*!vout+vl* vout,tx+tp*0.0,tr,tf ); // clkin
V(clkqp) <+ transition(vh* vout+vl*!vout,tx+tp*0.5,tr,tf ); // clkqp
V(clkqn) <+ transition(vh*!vout+vl* vout,tx+tp*0.5,tr,tf ); // clkqn
.........
```

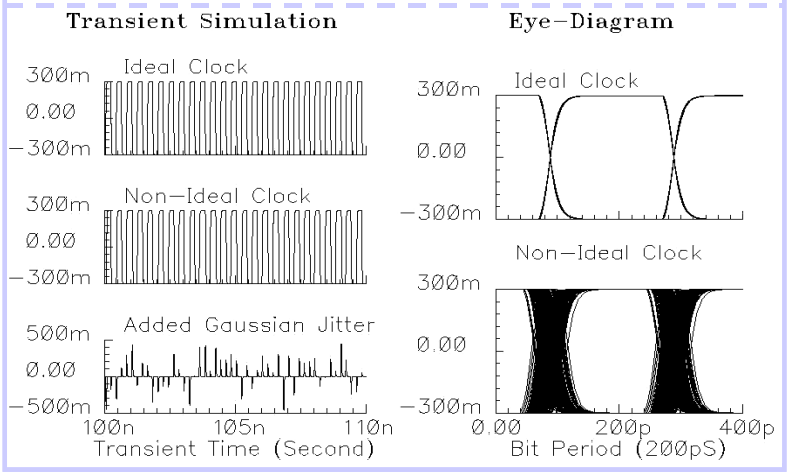


Figure 2.2 Example of modeling a clock with random jitter.

2.2.2 DATA WITH JITTER MODELING

```
.........
@ ( initial_step ) begin
  seed = prbsseed % pow(2 , prbs_bit); // trim seed within prbs length
  for (i = prbs_bit ; i >= 1 ; i = i - 1) begin
      bits[i] = (seed >= pow(2 , i - 1) ); // initialize prbs
      seed = seed - bits[i] * pow(2 , i - i);
  end
  bits[0] = bits[prbs_bit] ^ bits[prbs_fbb]; // bits xor for bit0
  next = $abstime; // next time
  vout = bits[prbs_bit]; // output data logic
  randseed = 1; // random number seed
end
$bound_step(tp);
rj = $rdist_normal(randseed, 0, rj_sd); // Gaussian rj
dj = $rdist_uniform(randseed, 0, dj_max) - dj_max/2; // Uniform dj
@ ( timer( next ) ) begin
  vout = bits[prbs_bit - 1]; // output data logic
  for (i = prbs_bit ; i >= 1 ; i = i - 1) begin
      bits[i] = bits[i - 1]; // prbs shift one bit
  end
  bits[0] = bits[prbs_bit] ^ bits[prbs_fbb]; // bit0 xor for bit0
  rj = rj * ( ( abs(rj) >= rj_min ) && ( abs(rj) <= rj_max ) )
     + rj_min * ( abs(rj) < rj_min )
     + rj_max * ( abs(rj) > rj_max ); // limit jitter in allowed range
  tx = td + 10 * tp + rj + dj; // data edge time
  next = next + tp; // next time
end
V(prbsp)  <+ transition(vh* vout+vl*!vout,tx    ,tr,tf ); // prbsp
V(prbsn)  <+ transition(vh*!vout+vl* vout,tx    ,tr,tf ); // prbsn
V(prbsdp) <+ transition(vh* vout+vl*!vout,tx+tdd,tr,tf ); // prbsdp
V(prbsdn) <+ transition(vh*!vout+vl* vout,tx+tdd,tr,tf ); // prbsdn
.........
```

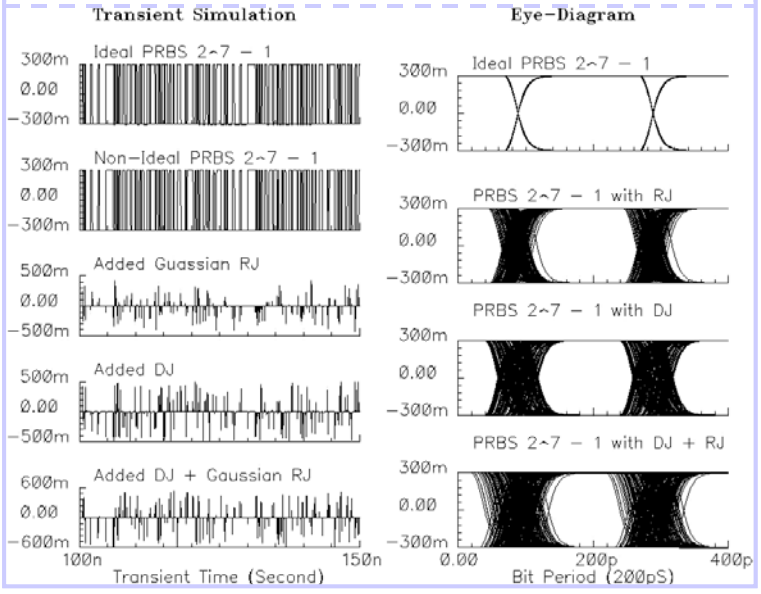


Figure 2.3 Example of modeling data having ISI and random jitter.

A bandwidth-limited transmission channel causes inter-symbol interference (ISI) on data, while an imperfect clock injects RJ into the data stream. Other jitters associated with the data are cross-talk (XT) and duty-cycle distortion (DCD). XT can be treated as a superposition of noise and signal. DCD is a skewing of the pulse width. Both ISI and DCD are part of deterministic jitter (DJ). A discussion of jitter associated with data can be found in Reference [8] and behavioral modeling of jitter is discussed in Reference [9]. An example of data having RJ and DJ is shown in Figure 2.3, which presents pseudorandom bit sequence (PRBS) data with a jitter generator. For the deserializer and the transmit driver designs, RJ is typically the only type included. For the receiver input and CDR design, however, both RJ and DJ should be considered.

2.3 SYSTEM AND CIRCUIT MODELING

System and circuit behavioral modeling allow designers to explore different system architectures or circuit topologies and make a quick feasibility analysis in the early stages of design [10]. Moreover, behavioral modeling is often used to prove the correctness of the fundamental theory and to estimate the overall performance of the system before starting the transistor-level design [3]. The goal is to provide an efficient system-level or hierarchical mixed-mode design verification platform.

2.3.1 SYSTEM-LEVEL MODELING

Behavioral modeling at the system-level often starts from a functional description in order to provide an intuitive sense for the signal flow from input to output as well as the overall system functionality. For example, a high-speed serial link starts with low-frequency parallel data sampled by a synthesized clock and serialized into a stream of

data. The serialized data is then pre-emphasized and transmitted to the receiver through a bandwidth-limited channel. The receiver needs to equalize the received signal in order to compensate the high frequency spectrum loss before extracting the clock and data from the CDR loop. Then the extracted data is sampled by the recovered clock and deserialized back into a set of parallel signals. Once a preliminary behavioral model has demonstrated the expected functional operation, then non-ideal components for jitter described in Section 2.2 are included to estimate jitter tolerance.

2.3.2 CIRCUIT LEVEL MODELING

Individual circuit modules below the system-level should also be modeled using behavioral descriptions. Such a circuit model could be used as an input signal provider or an output signal checker when running a transistor-level performance simulation for a specific circuit block. On the other hand, the circuit model may also be used for system-level mixed-mode or purely behavioral design verification.

2.4 DESIGN VERIFICATION

The design verification process may be divided into module and system levels of performance evaluation. It can be classified as being either full behavioral, mixed-mode or a complete transistor-level verification. Based on principles of top-down mixed-signal design, the design description proceeds from top to bottom (i.e., system-level to circuit level), and from purely behavioral to a purely physical level. On the other hand, the verification process may run from bottom to top and from physical to behavioral descriptions. Such a performance evaluation process is called Bottom-Up Verification [3].

2.4.1 TEST BENCH

We refer to a test bench based on behavioral modeling as virtual built-in self-test (VBIST). The primary focus of VBIST is to verify the designed physical circuit not only under different process, voltage and temperature (PVT) conditions, but also under extreme values of the input signals. For example, the receiver input stage, which includes an input buffer, DC-restore and equalizer could have different output responses based on an AC- or DC-coupled connection, the input common-mode voltage, the degree of duty-cycle distortion and the magnitude of DJ and RJ. If the input buffer gain and the equalizer frequency compensation are also variable, there will be on the order of 10 variations to consider during the analysis. The goal of VBIST is to create a test vector generator based on behavioral models that can accommodate all of these types of variations.

2.4.2 CRITICAL PATH DETECTOR

The primary purpose of critical path detection is to identify the components that are most likely to degrade the overall performance in a designed circuit or system. For example, the phase detector (PD) in the CDR block samples the input signal at the receiver input stage and often demultiplexes the sampled stream into two (or more) parallel output signals [11]. One of the output signals may have a higher probability of bit errors than the other signal(s). A critical path detector must be able to identify the weaker path and the location of the failing component. An example of such a simulation is shown in Figure 2.4. Of course, the same concepts can also be applied to serializer and deserializer design verification.

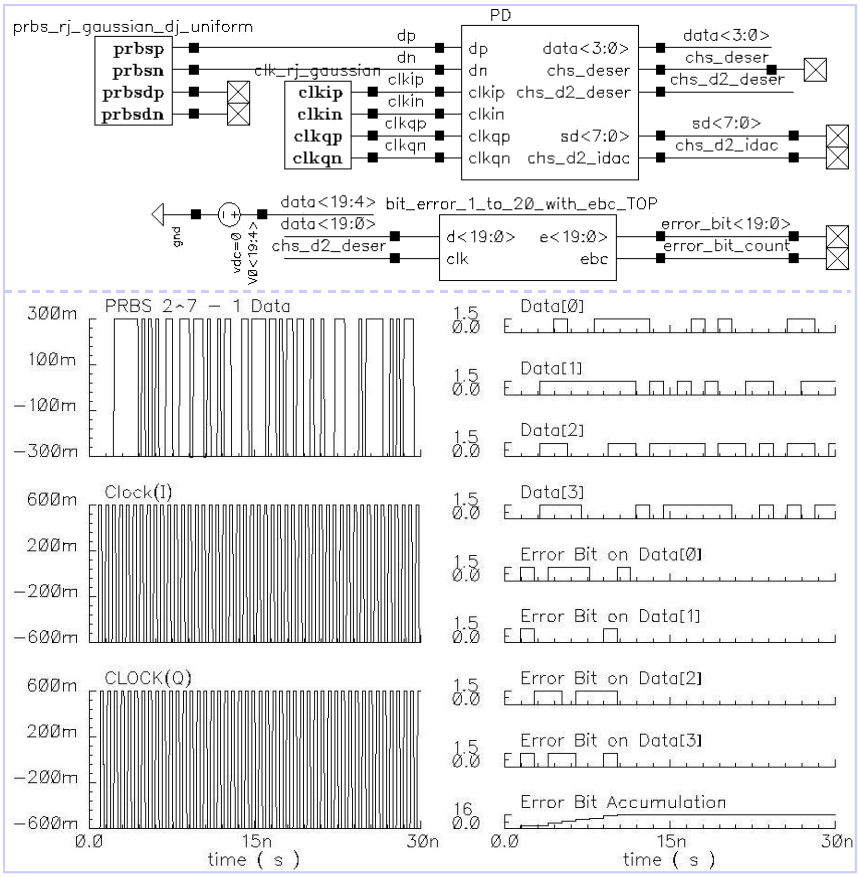


Figure 2.4 Phase detector modeling and simulation.

2.4.3 SYSTEM LEVEL VERIFICATION

Overall system-level performance is evaluated through a top-level verification. The first step of system-level verification is to check the interconnection of components and the overall functionality. This step is often done at the transistor-level in order to make sure that the whole system is working together and the physical layout has been done correctly. However, a detailed system-level performance evaluation including jitter generation, jitter tolerance and bit error rate (BER) analysis is conducted using mixed-mode or purely behavioral simulations [4]. Figure 2.5 shows an example of the bit error checker used in system-level verification.

```
.........
@ ( cross( V(clk) - vt_clk , 1 ) ) begin
  ct = $abstime; // get current time
  dx[ 0] = ( V(d00) > vt_d ); // transfer input data into dx in logic level
  .........
  dx[19] = ( V(d19) > vt_d ); // from bit 0 to bit 19
  err = 0; // set sum of error bits to be zero for 1st set comparison
  for ( i = 0 ; i < 30 - input_bits ; i = i + 1 ) begin
     p[i] = p[i + input_bits];
     d[i] = d[i + input_bits];
     e[i] = ( d[i] != p[i] );
     err = err + e[i];
  end
  for ( i = 30 - input_bits ; i < 30 ; i = i + 1 ) begin
     p[i] = ( p[i - prbs_bit] + p[i - prbs_fbb] ) % 2;
     d[i] = dx[i - 30 + input_bits];
     e[i] = ( d[i] != p[i] );
     err = err + e[i];
  end
end
if (err > 0) begin
    errx = 0; // set sum of error bits to be zero for 2nd set comparison
    for ( i = 0 ; i < prbs_bit ; i = i + 1 ) begin
       px[i] = d[i];
       ex[i] = ( d[i] != px[i] );
       errx = errx + ex[i];
    end
    for ( i = prbs_bit ; i < 30 ; i = i + 1 ) begin
       px[i] = ( px[i - prbs_bit] + px[i - prbs_fbb] ) % 2;
       ex[i] = ( d[i] != px[i] );
       errx = errx + ex[i];
    end
    if (errx == 0) begin
       for ( i = 0 ; i < 30 ; i = i + 1 ) begin
          p[i] = px[i];
          e[i] = ex[i];
       end
    end
    err = 0; // set sum of error bits to be zero for initial set comparison
    if ( ( ct - st ) >= ts_ebc ) begin
      for ( i = 0 ; i < input_bits ; i = i + 1 ) begin
         eb_cnt = eb_cnt + e[ 30 - input_bits +  i ]; // counting error bits
      end
    end
end
V(e00) <+ transition(voh*e[30-input_bits+ 0]
           +vol*!e[30-input_bits+ 0],td_e,tr_e,tr_e);
.........
V(e19) <+ transition(voh*e[30-input_bits+19]
           +vol*!e[30-input_bits+19],td_e,tr_e,tr_e);
V(ebc) <+ transition(eb_cnt, td_e, tr_e, tr_e);
.........
```

Figure 2.5 Verification model of the bit error checker.

2.5 CONCLUSIONS

In this chapter, we have demonstrated the use of behavioral modeling with Verilog-AMS for the design and verification of high-speed wired links. In the past, behavioral modeling was primarily used by system architects in order to estimate overall system performance. This chapter extends the use of behavioral models to handle non-ideal input signal generation and associated design verification issues in order to account for these important real-world effects. The ultimate goal of such behavioral modeling is to ensure better first-pass functioning silicon. Furthermore, behavioral models may be more easily reused in different design projects since their construction is not closely dependent on the details of a given process technology.

REFERENCES

[1] R. Sommer et. al., "From System Specification to Layout: Seamless Top-Down Design Methods for Analog and Mixed-Signal Applications," *IEEE Proceedings 2002 Design, Automation and Test in Europe Conference and Exhibition,* pp.884-891, March 2002

[2] P. Muller, A. Tajalli, M. Atarodi, and Y. Leblebici, "Top-down design of a low-power multi-channel 2.5-Gbit/s/channel gated oscillator clock-recovery circuit," *IEEE Proceedings 2005 Design, Automation and Test in Europe Conference and Exhibition,* Vol.1., pp.258-263, May 2005

[3] K. Kundert, "Principles of Top-Down Mixed Signal Design," The Designer's Guide Community http://www.designers-guide.org, 2003

[4] M. Wang, H. Maramis, D. Telian, and K. Chung, " New Techniques for Designing and Analyzing Multi-GigaHertz Serial Links," *DesignCon* 2005.

[5] C. Skach, "Signals predict Serdes jitter behavior," *EE Times* Asia, http:// www.eetasia.com, May 16, 2005.

[6] P.K. Hanumolu, B. Casper, R. Mooney, G. Wei, and U. Moon, "Analysis of PLL clock jitter in high-speed serial links," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, Vol.50, No.11, November 2003, pp. 879-886.

[7] M.A. Kossel and M.L. Schmatz, "Jitter Measurements of High-Speed Serial Links," IEEE Design and Test of Computers, Volume 21, Issue 6, pp. 536–543, No-vember 2004

[8] Lattice Semiconductor Corporation "Technical Note TN1084: SERDES Jitter," *Lattice Semiconductor Corpora*tion, March 2, 2005.

[9] M. Hsieh and G. Sobelman, "Accurate Prediction of Jitter Tolerance in High-Speed Serial Links," *International SoC Design Conference*, 2005.

[10] B. Antao, F. El-Turky, and R. Leonowich, "Mixed-mode simulation of phase-locked loops," *IEEE 1993 Custom Integrated Circuits Conference*, 9-12 May 1993, pp. 8.4.1 - 8.4.4

[11] A. Rezayee and K. Martin, "A 9-16Gb/s Clock and Data Recovery Circuit with Three state Phase Detector and Dual-Path Loop Architecture," *Proceedings of the 2003 European Solid-State Circuits Conference*, Estoril, Portugal, pp. 683-686.

CHAPTER 3

3 ACCURATE PREDICTION OF JITTER TOLERANCE IN HIGH-SPEED SERIAL LINKS

This chapter presents a novel mixed-signal verification methodology for jitter tolerance in high-speed serial-link receiver designs. The predictive approach includes identifying the jitter sources at the receiver input, generating the artificial jitter to add on input data and clock, and investigating the jitter sensitivities inside the receiver with bit error indicator. Both an artificial jitter generator and a bit error indicator are created using Verilog-AMS behavioral circuits. The simulation results show a good correlation with actual measurement data.

3.1 INTRODUCTION

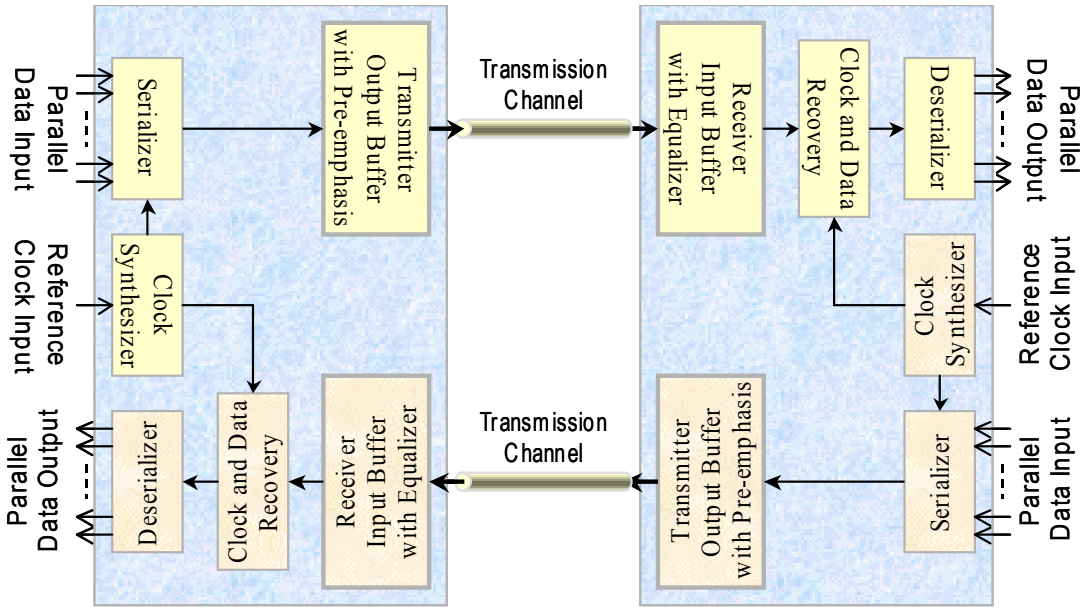


Figure 3.1 The generic block diagram of high-speed serial link communications.

There are three main types of communication channels exiting in contemporary high-speed serial links: chip-to-chip connections between different ASICs or standard components on the same board; board-to-board through a backplane for system connectivity inside the rack; and chassis-to-chassis through a cable to connect or expand equipment [1]. All three of these situations may be described by the generic block diagram of a high-speed serial communications link shown as in Figure 3.1.

Jitter is a measure of the time deviation that a waveform exhibits at significant instants compared to an ideal reference [2-4]. The unwanted jitter could arise from any component within the data or clock path. For example, the clock synthesizer rejects low frequency jitter from its reference clock but suffers from the high frequency random jitter (RJ) caused by thermal noise and up-conversion flicker noise in the voltage control oscillator (VCO). Once the data reach the transmitter output and travel across the transmission channel, it is further injected with deterministic jitter (DJ). The DJ is mainly caused by inter-symbol interference (ISI) and cross-talk. ISI arises from the low pass characteristic of the driver and the transmission channel, while cross-talk is caused by the signal coupling from adjacent channels due to the imperfect channel isolation. Once data reaches the receiver, any input offset creates a duty cycle distortion (DCD). The clock and data recovery circuit in the receiver extracts the clock from the input data stream and samples the input data. This process must remove most of DJ from the input data before sending the sampled data to the deserializer.

3.2 JITTER TOLERANCE VERIFICATION

The basic idea of the proposed predictive jitter tolerance simulation methodology is to inject various types of jitter into the received input signals and increase the magnitude of

this jitter until bit errors are observed at the receiver output. This approach is similar to the actual production jitter tolerance measurements that are taken using a bit error rate tester (BERT) scan. [5-6] This technique measures the BER of the transmitted data and fits the resulting BER curve to a mathematical jitter model to obtain the required jitter properties. The result of BERT scan is a graph of BER vs. sampling time through out one unit interval (UI) of data bit period as shown in Figure 3.2.

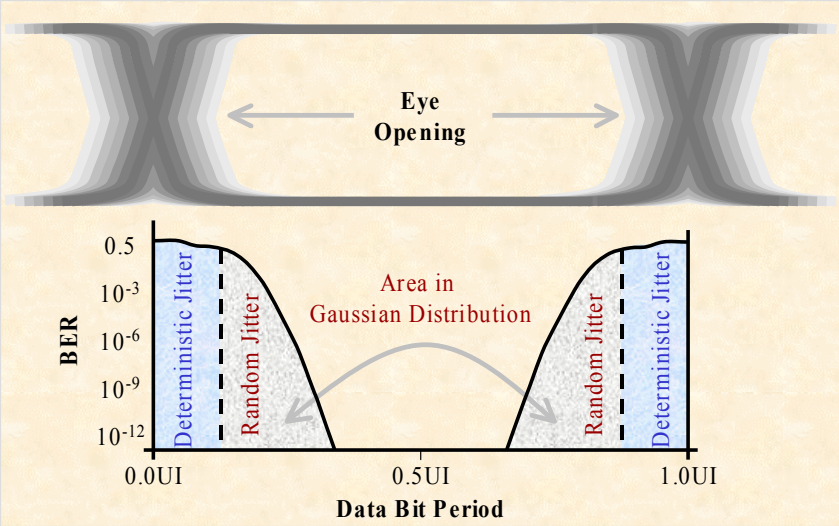


Figure 3.2 Data eye-diagram and BERT scan plot.

For simplicity and because of high isolation between channels in our design, we have considered no cross-talk in our analysis. Since cross-talk is a superposition of the receiver input signal and an attenuated adjacent channel signal, we can always add it back into our analysis as a further refinement. This would be based on the signal slew at the analyzed receiver input, the isolation between channels and the signal amplitude of the adjacent channel which creates the cross-talk [7]. Therefore, the generated receiver input data jitter consists of ISI and RJ and the reference clock only includes the RJ for our proposed jitter tolerance verification. The total jitter in time domain associated with data and clock

is given as:

$$TJ = DJ + RJ_{PP} = DJ + (\kappa_\sigma \times RJ_{RMS}) \tag{3.1}$$

Table 3.1 Multiplication factors for the translation from RJ_{RMS} to RJ_{PP}

BER	κ_σ	BER	κ_σ
10^{-4}	7.44	10^{-11}	13.41
10^{-5}	8.53	10^{-12}	14.07
10^{-6}	9.51	10^{-13}	14.70
10^{-7}	10.40	10^{-14}	15.30
10^{-8}	11.22	10^{-15}	15.88
10^{-9}	12.00	10^{-16}	16.44
10^{-10}	12.72	10^{-17}	16.93

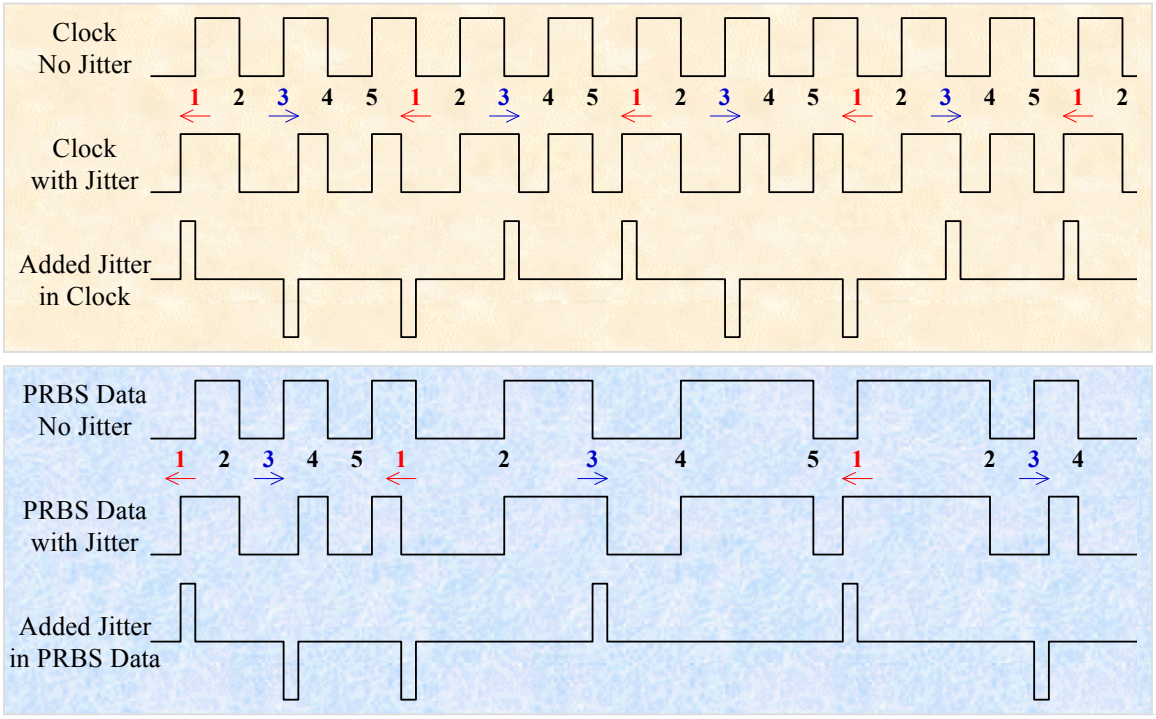


Figure 3.3 Illustration of clock and PRBS data with and without jitter used in the receiver simulation.

where RJ_{PP} and RJ_{RMS} are the peak-to-peak and root-mean-square (RMS) RJ and k_σ is the factor associated with different BER boundaries. The multiplication factors for the translation from RJ_{RMS} to RJ_{PP} are shown in Table 3.1 [5]. We have used a BER = 10^{-12} as meeting the specification. In many standards for data rate > 1 Gb/s, the BER is specified to be 10^{-12} or lower. RJ_{RMS} is a known value prior to the jitter tolerance verification for the designed receiver and its value is determined by evaluating the clock synthesizer performance.

```
……
……
……
@ ( timer( next_t ) ) begin
  vold = bits[prbs_bit];
  vout = bits[prbs_bit - 1];
  // shift prbs
  for (i = prbs_bit ; i >= 1 ; i = i - 1) begin
      bits[i] = bits[i - 1];
  end
  // modulo 2
  bits[0] = ( bits[prbs_bit] + bits[prbs_fbb] ) % 2;
  // set jitter
  td0 = td + tp + tj / 2.0 * ((count == edge_djp) - (count == edge_djn))
                           * ((ct - st ) >= dj_start);
  // edge counter
  count = (vout == vold) * count
          + (vout != vold)
          * ((count + 1) * (count < edges_dj) + (count >= edges_dj));
  next_t = next_t + tp;
  ct = next_t;
end
// output
V(prbsp)  <+ transition( vh *  vout + vl * !vout , td0 , tr , tf );
V(prbsn)  <+ transition( vh * !vout + vl *  vout , td0 , tr , tf );
……
……
……
```

Figure 3.4 Verilog-AMS code for PRBS with RJ.

Both receiver input data jitter and reference clock jitter for our jitter tolerance verification are generated from a behavioral model using Cadence Verilog-AMS. The reference clock only has RJ while the input data includes both DJ and RJ. An example of the RJ associated with the reference clock and the input data is shown in Figure 3.3. In our simulations, the added random jitter always takes its maximum absolute magnitude in order to create a worst-case situation. However, the pattern is user-defined depending on the specification requirements. The behavioral model that is used to generate DJ for the receiver input is based on the low-pass characteristic of FR-4 transmission lines. This behavioral model combines the ideal data together with the specified amount of RJ and then reshapes the signal based on the desired magnitude of ISI. The example of Verilog-AMS code for PRBS generator with RJ is shown in Figure 3.4.

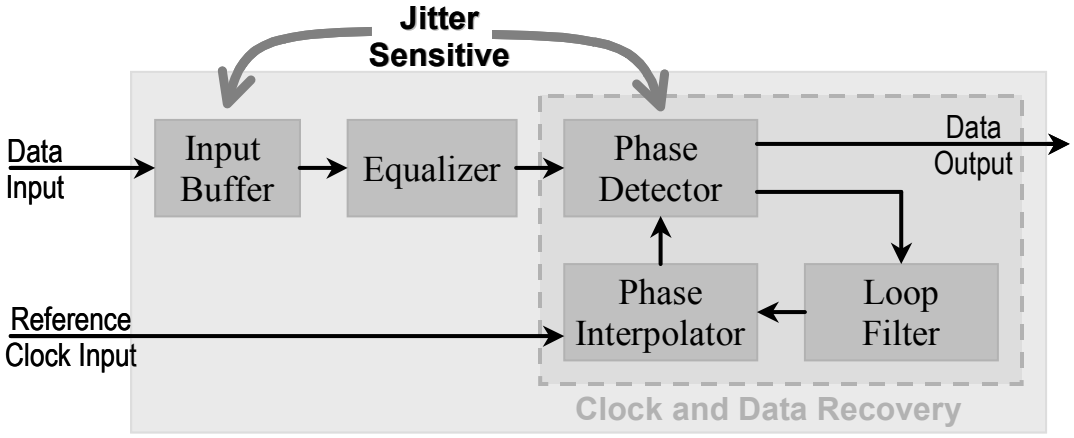


Figure 3.5 Circuit diagram of a receiver (no deserializer) based on phase interpolator type clock and data recovery.

In order to verify if the receiver output data is correct, an additional behavioral model is used. It is created to check for bit errors across the receiver from input to output. This behavioral model helps us to identify the jitter-sensitive components as well as the

locations of the bit errors. The ultimate goal is to pinpoint those components in the system which have particularly low jitter tolerance.

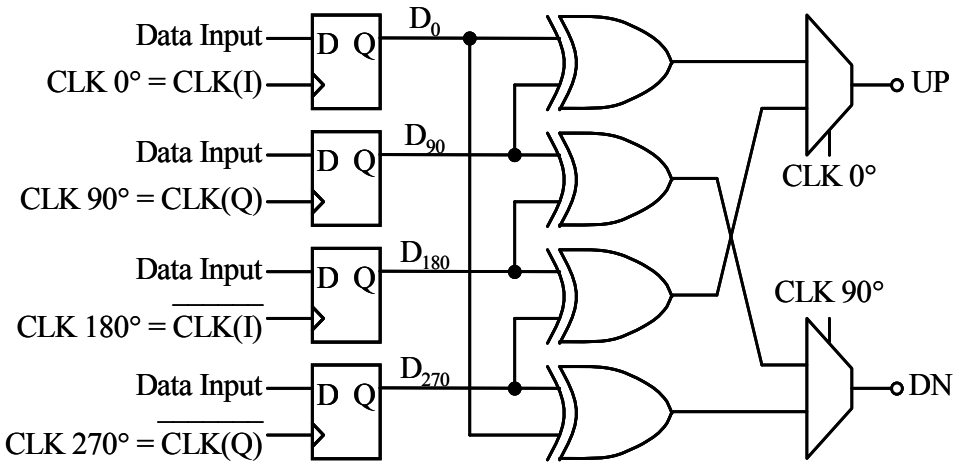


Figure 3.6 Half-rate Alexander type phase detector.

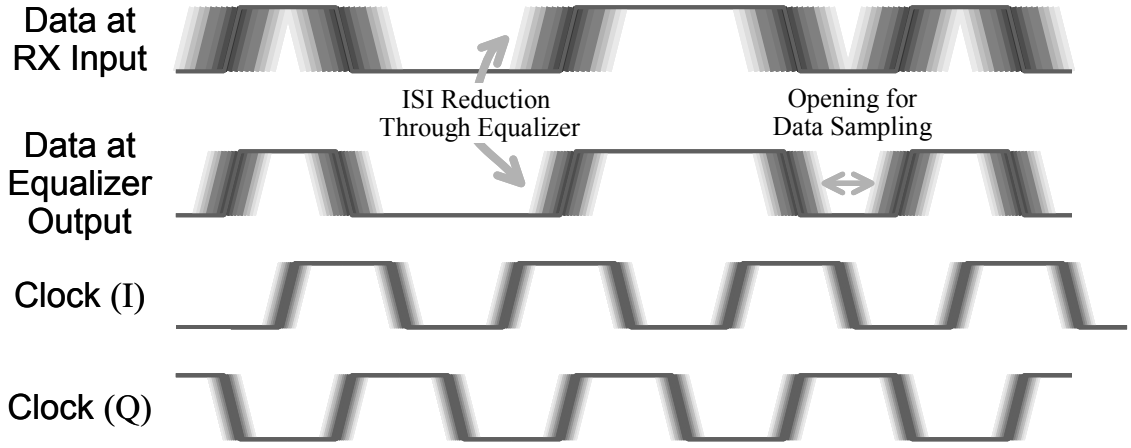


Figure 3.7 Receiver data and clock with jitter.

The designed receiver used for jitter tolerance verification is shown in Figure 3.5. It includes an input buffer, a source degenerated type of equalizer, a phase interpolator type of clock and data recovery, a half-rate Alexander type of phase detector and a 10-bit output deserializer. The receiver input buffer is an active stage to isolate the signals

inside and outside the chip.

The first step in the verification process is to identify the sensitive components. Input offset occurring at the input buffer stage in Figure 3.5 will create a DCD which will cause performance degradation in the following stages. Figure 3.6 shows the example of a half-rate Alexander type phase detector [8]. The four D-type flip-flops are the decision devices. The receiver BER performance is dominated by the decision devices in the phase detector when the eye-opening at the input of the decision device is small. Figure 3.7 gives the illustration of receiver data and clock with jitter. This indicates the important role of the decision device in the phase detector in terms of the BER performance. On the other hand, the equalizer provides ISI reduction due to its high-pass characteristic which serves to compensate the low-pass response of the transmitter and the channel.

3.3 SIMULATION AND MEASUREMENT

The designed receiver for this jitter tolerance verification is based on a 0.13 μm CMOS process and a PBGA package. The jitter tolerance simulation has been done using Cadence SPECTRE. In order to indicate the pass or fail of jitter tolerance for the designed receivers, an additional behavioral circuit model is created in Cadence Verilog-AMS to identify the bit errors and their occurred locations. The behavioral model for the bit error indicator must know the PRBS pattern at the receiver data input in advance in order to verify the correction of receiver data output. With an 8-bit/10-bit dc-balanced encoder used in the designed circuit, the 2^7-1 bits of the PRBS pattern is applied to the receiver input data for this jitter tolerance verification. For fast simulation in the initial verification stage, the static and non-high-speed path components with no jitter contribution are substituted with behavioral circuits in order to quickly identify the weak

jitter tolerance locations. However, the final verification is done using a capacitance-extracted simulation.

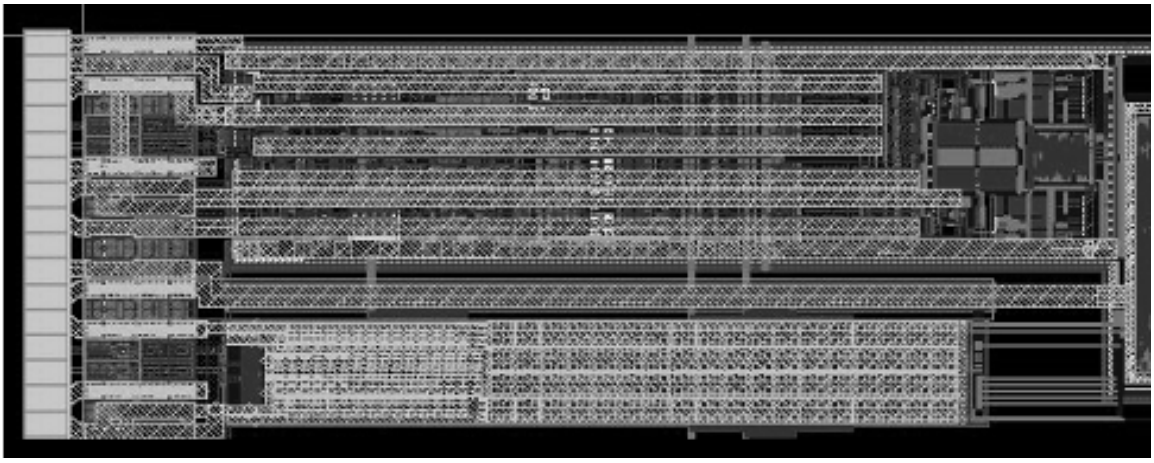


Figure 3.8 Layout of high-speed serial link receiver design

Figure 3.8 shows the layout of high-speed serial link receiver with a phase interpolator based CDR design. Figure 3.9 shows the signal eye diagrams inside the designed receiver and bit error indicator at the output of the receiver under jitter tolerance simulation with the equalizer either "OFF" and "ON" in order to illustrate its effect on performance.

The measured RJ_{RMS} from clock synthesizer is about 3.5 ps at 1.6 GHz. For BER=10^{-12} at a data rate of 3.2 Gbps, RJ_{PP} is approximately 50 ps = 0.16 UI. Therefore, the injected clock jitter is set at 0.16 UI. The injected data jitter is set at 0.2 UI RJ plus ISI starting from 0.4 UI. 0.2 UI RJ of data jitter is based on 0.16 UI RJ from clock synthesizer and 0.04 UI RJ jitter generation from transmitter serializer path. The total simulated input data bits stream length is 5,000 UI after the CDR loop in the steady state. The simulated equalizer output eye diagrams indicate an ISI reduction through the equalization process, which improves the deterministic jitter tolerance. On the other hand, the recovered clock

eye diagrams indicate the jitter generation inside the phase interpolation due to the phase shifting step resolution in the phase interpolator.

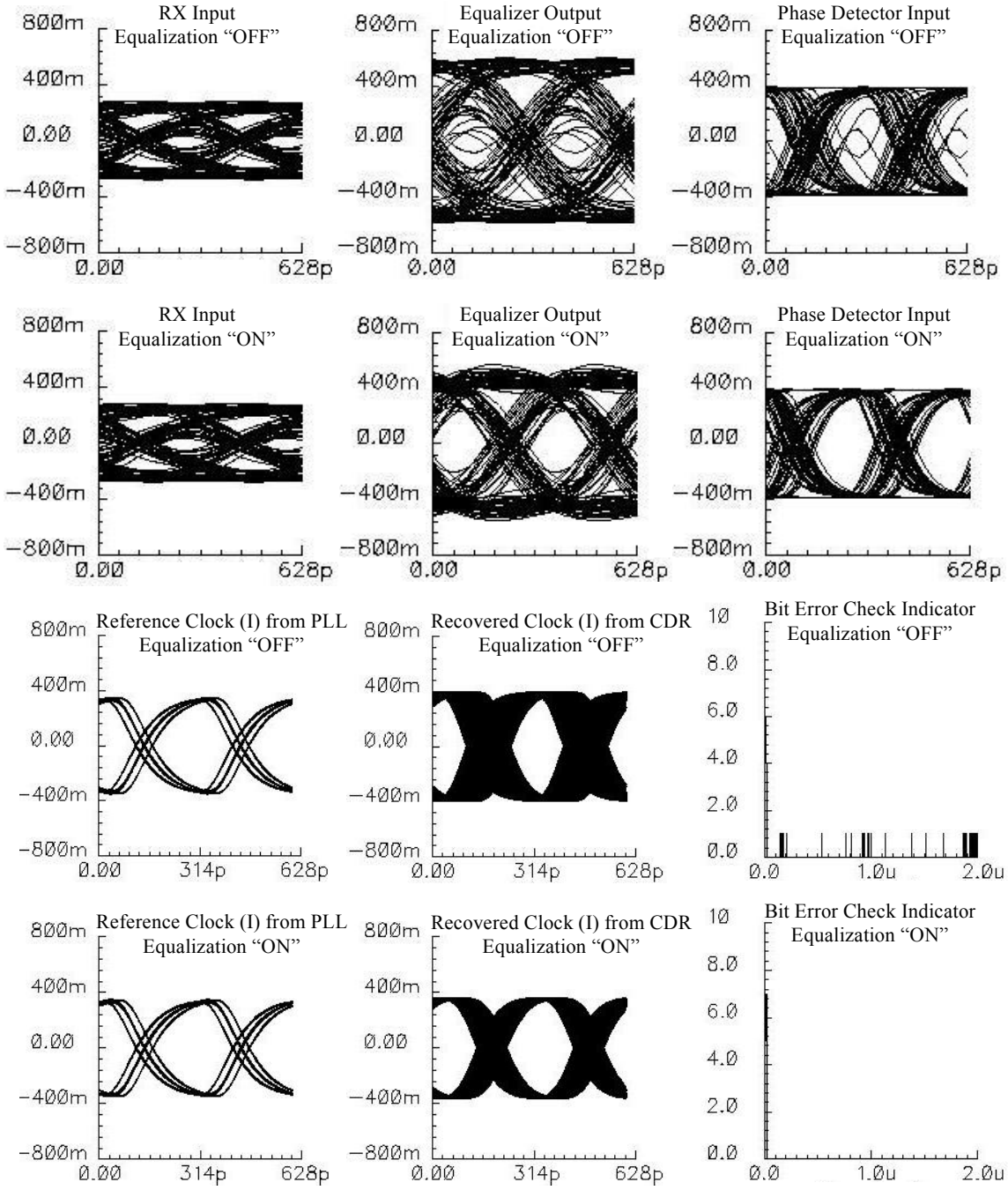


Figure 3.9 Data and clock eye diagram for receiver jitter tolerance simulation with equalizer "OFF" and "ON."

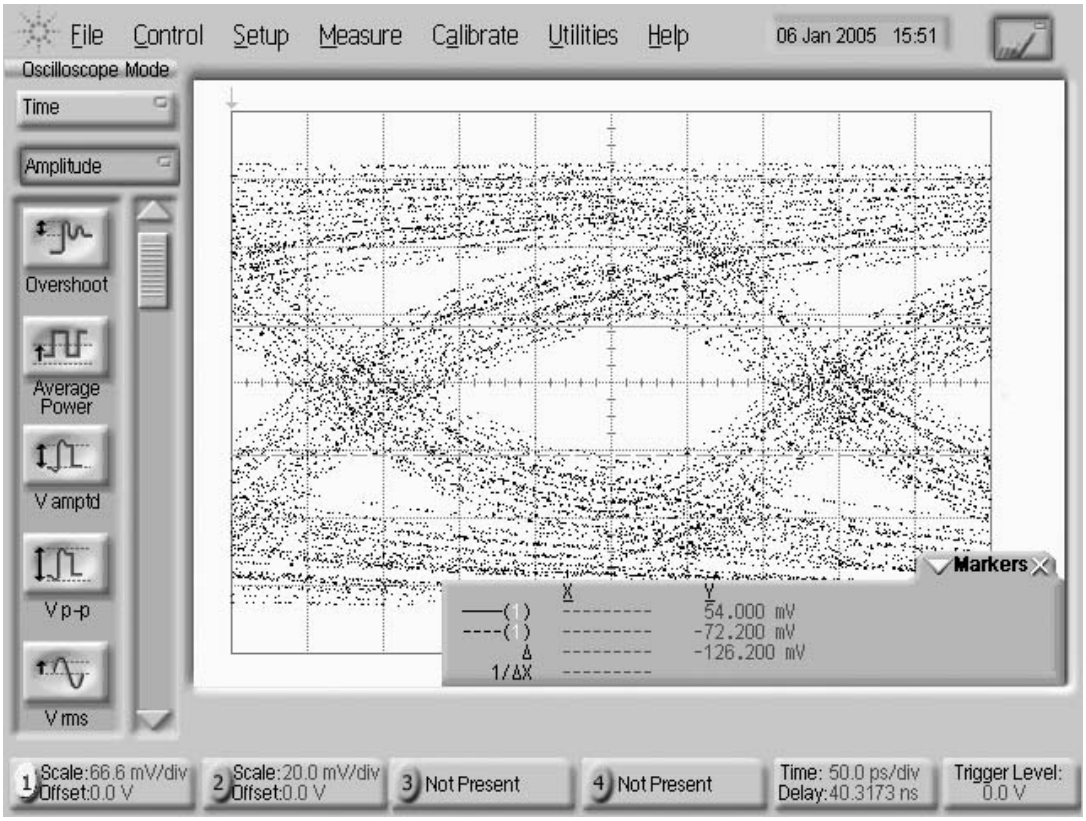


Figure 3.10 RX input eye diagram at 3.2 Gbps with 0.4 UI ISI for jitter tolerance measurement.

Table 3.2 Summary of the simulated and measured receiver jitter tolerance.

Data Rate : 3.2Gbps 1 UI : 625pS Equalization "ON"	**Injected Jitter Types**	**Process, Voltage and Temperature (PVT) Corner**		
		Slow	**Typical**	**Fast**
Simulation	Data Jitter	0.59 UI	0.62 UI	0.67 UI
	Clock Jitter	0.16 UI	0.16 UI	0.16 UI
Measurement	Data Jitter		0.69 UI	
	Clock Jitter		0.16 UI	

Figure 3.10 shows the measured results for jitter tolerance in the case of 0.4 UI ISI generation through a long cable without RJ. A PRBS pattern generator, a bit error rate tester and a jitter generator are used to make the measurement. The expected amount of ISI is controlled by selecting the data rate, the type of material used and the length of the

line. Table 3.2 gives a summary of the simulated and measured jitter tolerance results. As shown, there is quite a good correlation between the simulation and the measurement. The simulated data jitter tolerance value is slightly less than the actual measured value while the clock jitter tolerance is the same for both the simulation and the measurement.

3.4 CONCLUSIONS

We have described an accurate predictive methodology for jitter tolerance verification in high-speed serial links. We combine the actual transistor-level circuits of the receiver with behavioral models for the jitter sources and the checker to obtain good simulation accuracy. Cadence Verilog-AMS is used to create behavioral circuit models to generate the required jitter sources. The jitter magnitude is increased until bit errors occur at the receiver output. A checker behavioral model is used to create a bit error indicator across the receiver from input to output to identify the components having the weakest jitter tolerance. Our methodology for jitter tolerance shows good correlation between simulation and measurement.

REFERENCES

[1] J. Caroselli, "High-Speed Transceivers Require Systems Modeling," *EE Times* http://www.eetimes.com, November 7, 2003.

[2] M. Li, "Statistical and System Transfer Function based Method for Jitter and Noise in Communication Design and Test," *DesignCon 2005*.

[3] Lattice Semiconductor Corporation, Technical Note TN1084: "SERDES Jitter," Lattice Semiconductor Corporation, March 2, 2005.

[4] N. Ou, et al., “Jitter Models for the Design and Test of Gbps-Speed Serial Interconnects,” *IEEE Design and Test of Computers*, vol.21, no.4, pp. 302-313, July/August 2004.

[5] M.A. Kossel and M.L. Schmatz, “Jitter Measurements of High-Speed Serial Links,” *IEEE Design and Test of Computers*, Volume 21, Issue 6, pp. 536 – 543, November 2004.

[6] M. Mueller et al., “Total Jitter Measurement at Low Probability Levels, Using Optimized BERT Scan Method,” *DesignCon 2005.*

[7] V. Stojanovic and M. Horowitz, “Modeling and Analysis of High Speed Links,” *IEEE 2003 Custom Integrated Circuits Conference*, September 2003.

[8] A. Rezayee, and K. Martin, "A 9-16Gb/s Clock and Data Recovery Circuit with Three state Phase Detector and Dual-Path Loop Architecture," *Proceedings of the 2003 European Solid-State Circuits Conference*, Estoril, Portugal, pp. 683-686.

CHAPTER 4

4 LC AND RING VCO PLL DESIGN COMPARISON

This chapter presents performance, power and area comparisons of LC vs. Ring VCO-based PLL designs in multi-GHz applications using a standard 90 nm digital CMOS process. The primary goal of this chapter is to develop an analytical framework for determining the best option for a high-speed clock synthesizer design based on the constraints of the target application. A type-II 3rd order PLL is used in this study because of its capability for allowing independent adjustments to the damping factor, loop-bandwidth and loop gain. Cadence Spectre and SpectreRF are used to verify the transient response and phase noise analysis. The measured results show good correlation with the simulated data.

4.1 INTRODUCTION

Phase locked loops (PLLs) are commonly used in high-speed digital systems to perform a variety of clock processing tasks such as the clock recovery, skew cancellation, clock generation, spread spectrum clocking, clock distribution, jitter / noise reduction and frequency synthesis [1-4]. Figure 4.1 shows a typical circuit diagram of a type-II 3rd order PLL design used for a clock frequency multiplication application [2,4]. It takes a low frequency reference clock input, R_{CLK}, and multiplies it by the divider ratio, M, to generate a high frequency PLL output clock, PLL_{CLK}. The PLL in Figure 4.1 consists of a phase-frequency detector (PFD), charge-pump (CP), loop filter (LF), voltage controlled oscillator (VCO) and frequency divider. The PFD compares the frequency and phase

alignment between the input reference clock, R_{CLK}, and the frequency divider output clock, F_{CLK}, whose frequency is divided down by the divider ratio, M, from VCO output clock, PLL_{CLK}. CP injects current into or removes current from the LF based on the UP and DN signals from the PFD output. The filtered VCO control voltage, V_{CNT}, from LF with a low-pass frequency response drives the VCO to a target oscillation frequency according to the multiplication of the divider ratio, M, and the input reference clock frequency.

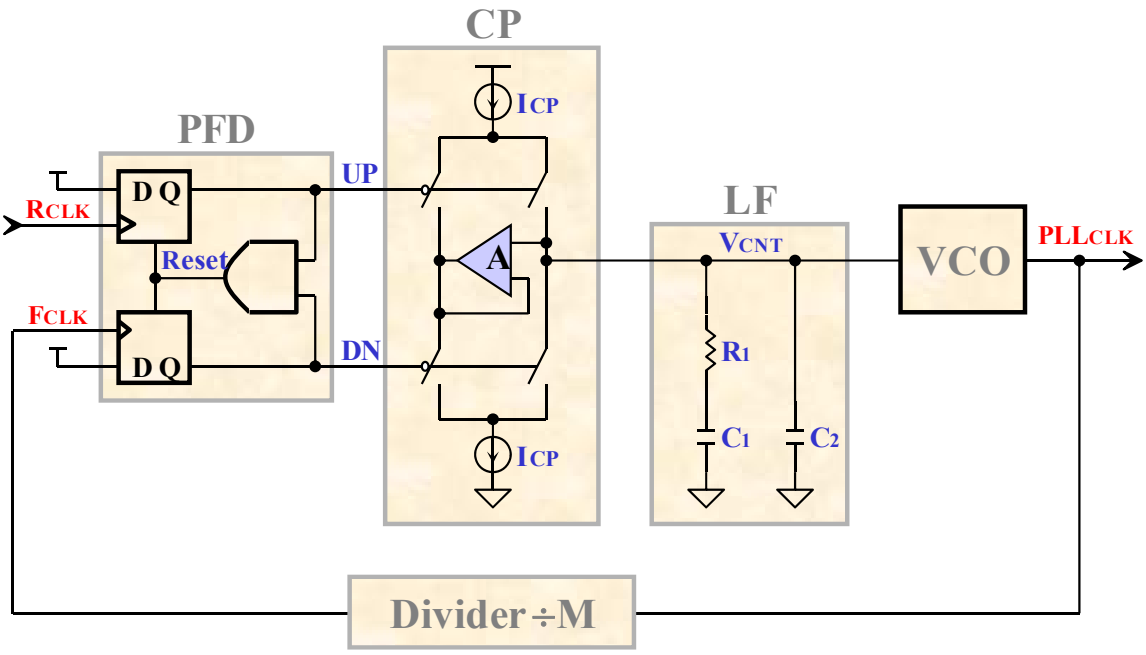


Figure 4.1 A typical circuit diagram of a type-II 3rd order PLL.

The VCO is one of the key components in a PLL, and its jitter response and tuning range have a large impact on the PLL's overall performance [3]. For CMOS VCO designs, LC and ring oscillator (henceforth called Ring) based VCOs are the two main choices used in monolithic PLL designs [4]. LC VCOs have a superior phase noise / jitter performance because of their high quality factor (High-Q) frequency response compared with Ring VCOs. However, LC VCOs normally have a smaller tuning range,

require larger layout area, have higher design complexity and possibly dissipate higher power [5]. For reasons of design simplicity and cost effectiveness, a Ring VCO based PLL is often considered first to determine if it can meet the performance requirements.

While previous studies have been done on the optimization of LC VCOs [6-8], Ring VCOs [9-11] and PLLs [12-14], they do not provide detailed performance, power or area comparisons between LC and Ring VCOs based PLL designs. Reference [5] discusses these issues but it only presents very limited information, as its primary focus was to illustrate the differences between LC and Ring VCOs based PLLs across different process technology nodes at the maximum operating frequency.

Our previous work [15] has also discussed the power, area, and performance comparison. The main difference from reference [5] is that our study concentrates on the same process technology node and investigates the best match between LC and Ring VCOs for PLL designs based on the constraints of the application.

This chapter is a significantly expanded version of our previous work [15] and its primary focus is to develop an analytical framework for determining the best choice between an LC and a Ring VCO at a specific process technology node based on the constraints of the application. Section II discusses the PLL architecture and presents the comparative analysis for the LC and Ring VCO based PLLs. Simulated performance results based on Cadence Spectre and SpectreRF along with measured data on fabricated chips are given in Section 4.3. The trade-off among the performance, power and layout area is also discussed in Section 4.3.

4.2 PLL ARCHITECTURES AND ANALYSIS

A type-II 3rd order PLL architecture is used in this chapter as shown in Figure 4.1,

which is also called a charge pump PLL [16-19]. The order of a PLL is defined by the number of poles in the loop transfer function, and the type of a PLL indicates the number of perfect integrators with poles at s = 0 in the loop response [20]. The transfer function of the 2nd order LF and 1st order VCO in Figure 4.1 can be expressed as Equation 4.1 and 4.2, respectively.

$$H_{LF}(s) = \frac{1 + sR_1C_1}{s^2R_1C_1C_2 + s(C_1 + C_2)} \tag{4.1}$$

$$H_{VCO}(s) = \frac{K_{VCO}}{s} \tag{4.2}$$

where K_{VCO} is the VCO gain in Hz/V. Assuming the VCO gain, K_{VCO}, and the PFD-CP gain, K_{CP}, are relatively constant with frequency, the 2nd-order transfer function from LF and the 1st-order transfer function from VCO result in a 3rd-order PLL loop response. The use of a 2nd-order LF in Figure 4.1 with a zero from R_1 is for stability purposes. (There would be two poles at zero frequency in the PLL loop response if R_1 and C_2 were not present.) The 2nd pole in LF, contributed by C_2, is used for smoothing the ripple on the VCO control voltage, V_{CNT}, after R_1 is included [4]. The added C_2 imposes a stability concern. C_2 is usually chosen to be less than C_1 / 10 in order to meet the stability requirement, but a value greater than C_1 / 50 may be employed to maintain a low periodic jitter response at the PLL output [2].

Type-I PLL is not chosen due to the difficulty in adjusting its loop bandwidth, damping factor and loop gain independently [20]. In addition, it has limited acquisition range and is difficult to implement in high-performance digital integrated circuit designs [4]. Furthermore, a higher-order PLL is infrequently considered because of the difficulty to stabilize such a system, especially when the process, supply, and temperature variations

are taken into account [20].

Type-II PLL has advantages of a large locking range, a high-speed capturing process and an infinite DC-gain. The latter is provided by the pole at zero frequency through the combination of PFD and CP with the loop filter, which results in zero phase error under an ideal locked state [20-21]. However, a negative aspect is that the sampling operation from the PFD-CP introduces spurious tones at the VCO output [20,22], and the loop bandwidth is limited by stability considerations. Another drawback of type-II PLL is the inherent, undesirable peaking behavior of jitter transference from the input reference clock to the PLL output in the loop gain response [13,17,19-20]. This peaking can be especially challenging in a serial cascaded system such as Synchronous Optical Network (SONET) with a requirement of less than 0.1dB of peaking for jitter transference [4,23]. A detailed system analysis of a charge pump PLL can be found in [16-19].

4.2.1 VCO TOPOLOGIES AND TRADE-OFF

Comparative studies of CMOS LC VCOs for multi-GHz applications have been reported in [24-30]. Figure 4.2(a) [24,26-27] and (b) [6-8,19,24,26,28] are the two commonly used LC VCO topologies. LC VCOs with a PMOS-only cross-coupled diff-pair topology as shown in Figure 4.2(c) [7,29] is used less often because of its lower maximum unity current-gain cut-off frequency, f_T, even though PMOS devices may have an order of magnitude lower *1/f* flicker noise [30] and contribute less drain current thermal noise for the same transconductance, g_m, than NMOS devices [7]. The tail current can be either sourcing from the supply, sinking to ground or completely removed. Figure 4.2(a) [24,26-27], (d) [7,15,20,25] and (e) [26-27] show examples of three commonly used LC VCO topologies with a complementary cross-coupled diff-pair. The

examples with the opposite tail current topologies shown in Figure 4.2(b) and (c) have been reported in [25,30].

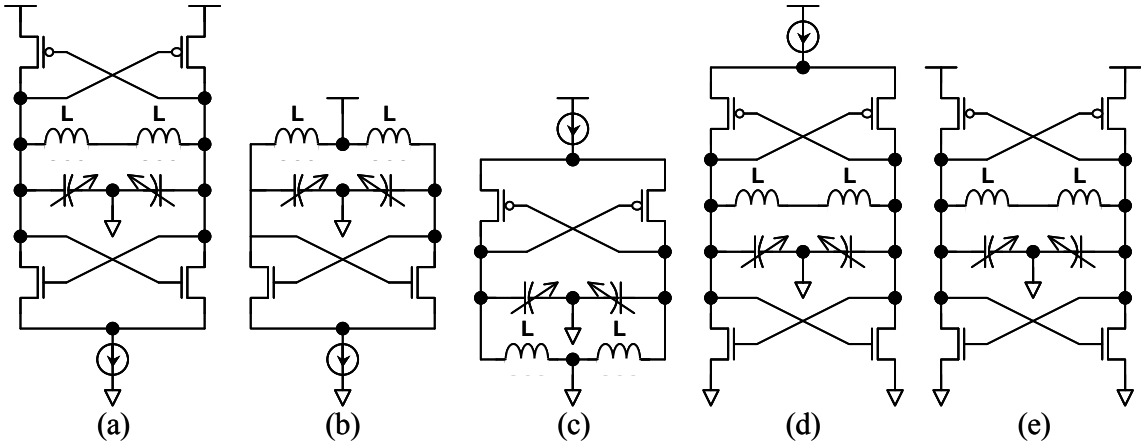


Figure 4.2 LC VCO (a) Complementary with current source (b) NMOS with current sink (c) PMOS with current source (d) Complementary with current sink (e) Complementary without current source or sink.

The VCO tail current to the LC resonator limits the voltage swing across the resonator, which also adds to the overall VCO phase noise [25]. However, the advantage of the tail current is to give a constant bias current to the differential pair, which makes it more immune to supply voltage variation [25]. Furthermore, the tail current provides better management of power dissipation [25]. Generally, tail current sourcing from the supply is preferred because it provides a certain degree of isolation between the supply and the LC resonator, which reduces the supply noise injected into the LC resonator [24]. Also, its smaller 1/f flicker noise leads to a reduction in the overall VCO phase noise [29]. Generally, the impact of phase noise from tail current can be minimized by adding a large capacitor at the tail current common node [29-30].

For a given tail current, the complementary VCO provides a higher transconductance, which results in faster switching of the cross-coupled differential-pair than in NMOS- or

PMOS-only VCOs [24]. Furthermore, the complementary topology has better rise / fall time symmetry, which tends to reduce up conversion of *1/f* flicker noise and results in better phase noise performance [26-27]. One of the major disadvantages of the complementary structure is that the maximum signal swing is limited to the supply voltage. Furthermore, the complementary structure often suffers from larger parasitic capacitance, which leads to a lower maximum oscillation frequency [25]. On the other hand, NMOS- or PMOS-only structures can provide signal swings greater than the supply voltage at the expense of increased tail current and reduced power efficiency. Moreover, large signal swings above the supply voltage may also result in reduced device reliability [24].

A comparative study of CMOS Ring VCOs has been reported in [31-36]. Six different differential delay cells as shown in Figure 4.3(b)~(g) were studied, where V_I, V_O and V_C are the delay cell's input, output and bias voltage control for frequency, respectively. V_B is the bias current control for providing constant tail current in the differential delay stage. Figure 4.3(a) presents a typical Ring VCO circuit diagram with 4-stage differential delay cells [11,32,34].

The source coupled delay cell [11,33-34], as shown in Figure 4.3(b) is the simplest delay cell design with the poorest phase noise performance because the PMOS load can be either in the triode or saturation region [33]. The clamped-load delay cell [32,35], as shown in Figure 4.3(c), has excellent noise and supply noise rejection characteristics because the cross-coupled diode-loads clamp the output swing making the delay cell insensitive to common-mode noise [32]. The symmetric load delay cell, as shown in Figure 4.3(d) [10-11,32-33] also has very good supply noise rejection characteristics and

has been used extensively in PLL and clock generator designs [33].

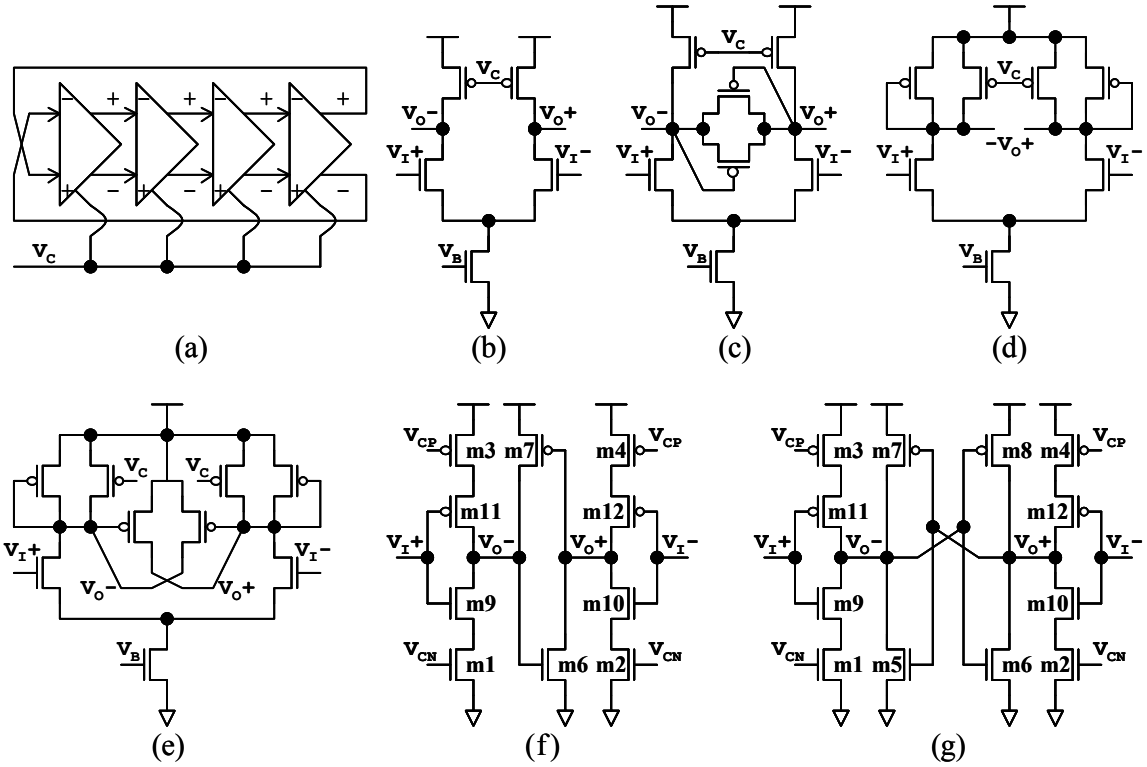


Figure 4.3 Ring VCO (a) 4-Stage Ring VCO (b) source coupled delay cell (c) source coupled with clamped load (d) source coupled with symmetric load delay cell (e) source coupled with symmetric and cross-coupled active load delay cell (f) dual inverter delay cell (g) dual inverter with balanced cross-coupled delay cell.

The cross-coupled load delay cell, as shown in Figure 4.3(e) offers the lowest phase noise in the $1/f^3$ region compared with Figure 4.3(b)~(d) because of a more symmetric signal than in the other three [32]. The dual inverter delay cell [31,34] and dual inverter with balanced cross-couple delay cell [31,36], as shown in Figure 4.3(f) and (g), respectively, offer the best possible phase noise performance for a given power consumption in the upper portion of the VCO's frequency tuning range because of the larger signal swing in the delay cell. The m1~m4 devices in Figure 4.3(f) and (g) for current-starved control are less resistive in the upper VCO frequency tuning range with

increased power drawn into m9~m12, which also further reduces the phase noise [34]. The delay cell in Figure 4.3(g) has better noise rejection than the one in Figure 4.3(f) because of its symmetric load [31]. However, the delay cell in Figure 4.3(g) suffers from larger parasitic capacitance and exhibits a lower maximum oscillation frequency.

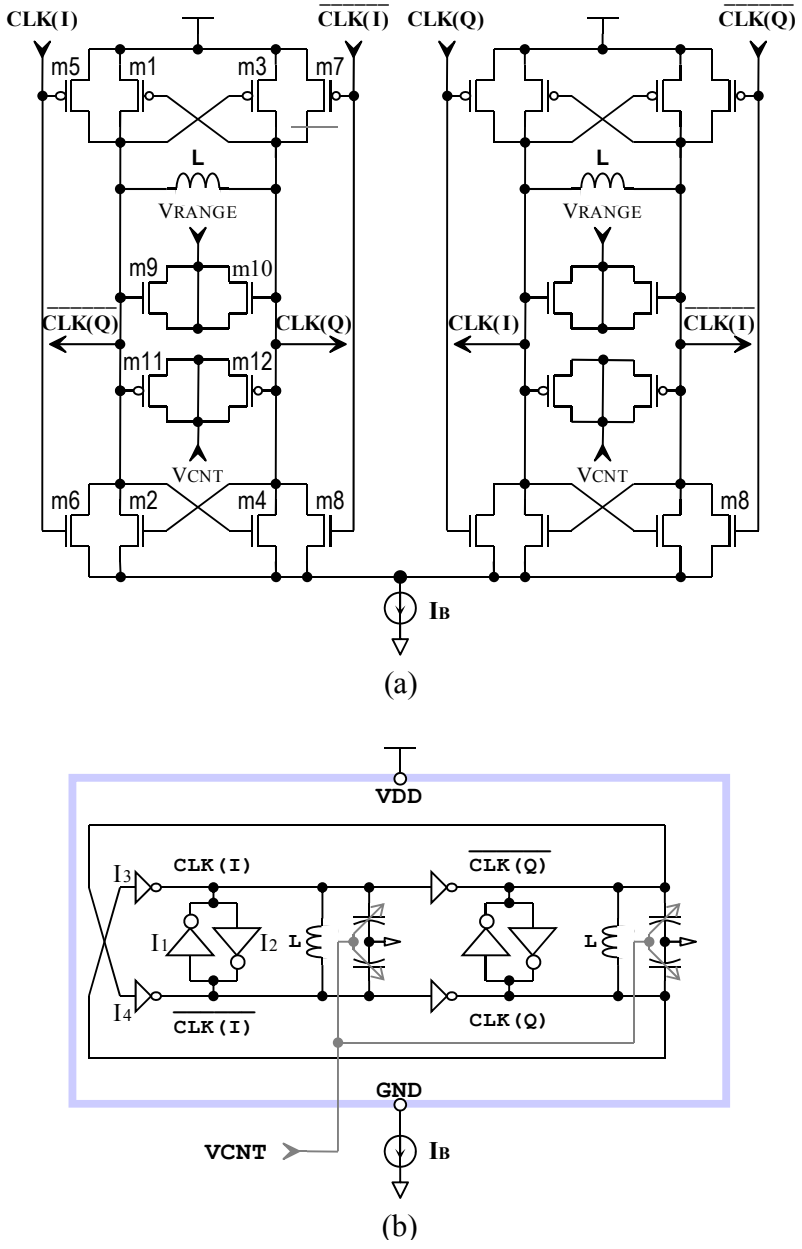


Figure 4.4 Circuit schematic of the designed LC VCO with quadrature outputs (a) transistor-level (b) logic-level.

The LC and Ring VCO based PLLs in this study use a common topology consisting of PFD, CP, LF and divider, as shown in Figure 4.1. However, the CP current gain and LF frequency responses are individually adjustable in order to optimize the PLL performance as needed. The designs of the LC and Ring VCOs used in this study are shown in Figure 4.4 and Figure 4.5, respectively. Both VCOs support differential in-phase, CLK(I), and quadrature-phase, CLK(Q), clock outputs [2,4].

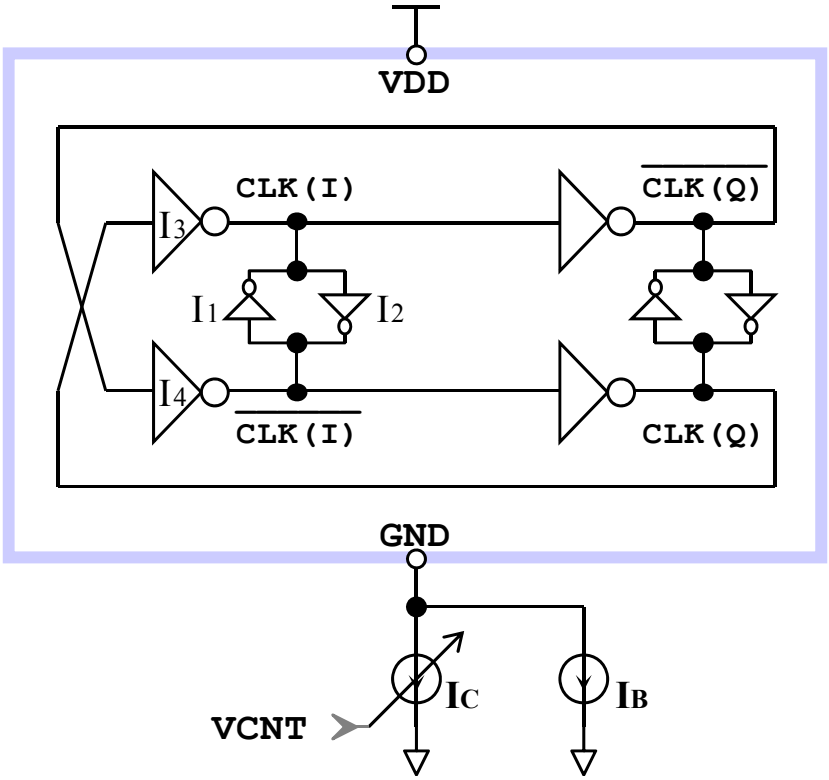


Figure 4.5 Circuit schematic of the designed Ring VCO with quadrature outputs.

The LC VCO in Figure 4.2(a) has the best phase noise performance and the least tail current requirement [24], but only supports clock outputs with 0 and 180 degree phases. The designed LC VCO in this study is shown in Figure 4.4, which is derived from Figure 4.2(a) to support quadrature phase clock outputs. It is also a modification of [37] by

adding the tail current sink to ground in order to manage the power dissipation over process, supply and temperature variations. The ratio of sizes between cross coupling transistors m1~m4 and driving transistors m5~m8 in Figure 4.4(a) is set to 3 [37], where m1~m4 and m5~m8 in Figure 4.4(a) are equivalent to I_1~I_2 and I_3~I_4 in Figure 4.4(b), respectively. The small sizes of m5~m8 means the oscillation frequency primarily depends on the LC resonator rather than the loop delay. However, it maintains the minimum required gain for accurate 90 degree phase-shift and frequency synchronization between the two LC resonators.

The frequency control for the designed LC VCO is through MOSFET based varactors having low threshold voltages for better frequency tuning. Depletion mode MOSFET and PN junction diode based varactors are often unavailable in a low cost standard digital CMOS process. In simple CP and LF designs, differential frequency tuning of an LC VCO in [37] is not used. Binary-weighted NMOS-based varactors, m9~m10, are used for frequency range control, and each NMOS-based varactor is either driven to VDD or GND depending on the selected VCO frequency range. Single-ended PMOS-based varactors, m11~m12, are used for frequency tuning that is directly driven by the VCO control voltage, V_{CNT}, from the LF. The use of combined PMOS- and NMOS-based varactors provides less capacitance variation at different voltage potentials in the LC resonator.

The Ring VCO used in this study is shown in Figure 4.5, which consists of two stages of delay cells [31,36] having similar structure, like in Figure 4.3(a). The delay cell is derived from Figure 4.3(g) without m1~m4, which offers the best possible noise performance [31]. Figure 4.5 is a modified version of [36] without tail current sourcing

from the supply to simplify the oscillation frequency control. Furthermore, the tail currents sinking to ground in both delay cells are combined, which averages out the dynamic tail current and results in less overall VCO noise contributed by tail current.

The two stages of Ring VCO design in Figure 4.5 have the same topology as the LC VCO design in Figure 4.4 but without the inductors and varactors. However, the ratio of sizes between the phase inverting latch of I_1~I_2 and the driving delay stages from I_3~I_4 is set to 1/3. The small size of I_1~I_2 only needs to meet the minimum required gain for phase inversion. This also minimizes the parasitic capacitance in order to maximize the oscillation frequency. The frequency control for the Ring VCO is through the voltage control current source, I_C, which is usually an NMOS transistor driven by the VCO control voltage, V_{CNT}. The constant bias current, I_B, sinked to ground provides a minimum initial oscillation frequency for the Ring VCO.

4.2.2 FREQUENCY, POWER, AND AREA ANALYSIS

The oscillation frequency of an LC VCO is given by Equation 4.3 [19].

$$\omega_{LC}^2 = \frac{1}{LC}\frac{1+\frac{1}{Q_C^2}}{1+\frac{1}{Q_L^2}} \tag{4.3}$$

where Q_C and Q_L are the quality factors for L and C, respectively. Q_C is typically about 20 or higher [38]. If Q_L is greater than 7, the error in the oscillation frequency calculation by eliminating Q_C and Q_L will be less than 1%. Then, Equation 4.3 can be reduced to Equations 4.4~4.5 [4,6-7, 37].

$$\omega_{LC}^2 \approx \frac{1}{LC} \tag{4.4}$$

$$f_{LC} \approx \frac{1}{2\pi\sqrt{LC}} = \frac{1}{2\pi\sqrt{L(C_{VAR} + C_{MOS} + C_{PAR})}} \tag{4.5}$$

Here C is the sum of the varactor capacitance, C_{VAR}, MOSFET capacitance, C_{MOS}, and all other parasitic capacitance, C_{PAR}, from routing traces and the spiral inductor traces. In order to ensure the start-up of oscillation across the entire tuning range, the minimum total transconductance, G_m, of the LC VCO cross-coupled diff-pair of Figure 4.4 needs to satisfy Equation 4.6 [25].

$$G_M = gm_{m1} + gm_{m2} = \sqrt{2I_{LC}\mu_{eff}C_{OX}\frac{W_{eff}}{L_{eff}}} \geq \frac{2}{Q_{LC}^2 R_S} \approx \frac{2}{Q_L^2 R_S} = \frac{R_S}{2\pi^2 f_{LC}^2 L^2} = \frac{2}{R_P} \tag{4.6}$$

$$W_{eff} = W_N + W_P \tag{4.7}$$

$$\mu_{eff} = \frac{\mu_N W_N + \mu_P W_P}{W_N + W_P} \tag{4.8}$$

$$\frac{1}{Q_{LC}} = \frac{1}{Q_L} + \frac{1}{Q_C} \approx \frac{1}{Q_L} \Rightarrow Q_{LC} \approx Q_L \tag{4.9}$$

where gm_{m1} and gm_{m2} are the transconductances for devices m1 and m2, as shown in Figure 4.4. R_S and R_P are the equivalent series and parallel resistances for the LC resonator in the VCO. $I_{LC} = I_B/2$ is the LC resonator tail current for each VCO in Figure 4.4. C_{OX} is the total gate capacitance. W_{eff}, L_{eff}, and μ_{eff} are the effective device width, length and mobility of the NMOS and PMOS cross-coupled diff-pair. W_N and W_P are the NMOS and PMOS total channel width, respectively. μ_N *and* μ_P are the NMOS and PMOS mobility, respectively. As noted above, in practical designs, Q_C is typically much larger than Q_L [38], which leads $Q_{LC} \approx Q_L$ as shown in Equation 4.9 [24]. The LC VCO oscillation amplitude is a function of the LC resonator equivalent parallel resistance, R_P, and the tail current, I_{LC}, as given in Equation 4.10 [4,25,28]:

$$V_{LC} = I_{LC} R_P \tag{4.10}$$

$$R_P \approx \frac{\omega_{LC}^2 L^2}{R_S} = Q_L^2 R_S \tag{4.11}$$

$$I_{LC} = \frac{V_{LC}}{R_P} \approx \frac{V_{LC} R_S}{\omega_{LC}^2 L^2} = \frac{V_{LC}}{Q_L^2 R_S} \propto \frac{1}{\omega_{LC}^2 L^2} \propto \frac{1}{Q_L^2} \tag{4.12}$$

The equivalent LC resonator parallel resistance, R_P, can be approximately calculated by Equation 4.11 [4,25]. For a constant oscillation amplitude, the current dissipation is inversely proportional to the squares of the oscillation frequency, inductance and quality factor as shown in Equation 4.12.

Based on Equations 4.5~4.6 and 4.10, with a targeted oscillation frequency, desired amplitude and allowable tail current dissipation, one can estimate the minimum required device sizes for the LC VCO design in Figure 4.4, which leads to the layout area required for the cross-coupled diff-pair and tail current mirror. The size of the inductor is estimated through the required inductance and maximum R_S (minimum R_P or minimum quality factor, Q) to meet the desired oscillation frequency and amplitude. Cadence's Virtuoso Passive Component Design (VPCD) tool is used in this study for optimizing the inductor design based on the required inductance and minimum quality factor. Once I_{TAIL} is defined, the LC VCO power dissipation can be calculated by Equation 4.13:

$$P_{LC} = I_B V_{DD} = 2 I_{LC} V_{DD} \tag{4.13}$$

where V_{DD} is the supply voltage for the LC VCO, and where I_B is the total tail current for the two LC resonators in Figure 4.4.

For a Ring VCO with N stages of identical delay cells, the oscillation frequency vs. number of delay stages is illustrated in Figure 4.6 [31]. The oscillation frequency of the Ring VCO is inversely proportional to the number of delay stages. This is why the Ring

VCO design in Figure 4.5 uses the minimum required number (i.e., two) of stages. However, using a small number of stages in the Ring VCO also results in higher phase noise [9]. The oscillation frequency for a Ring oscillator is given in Equation 4.14 [39-40].

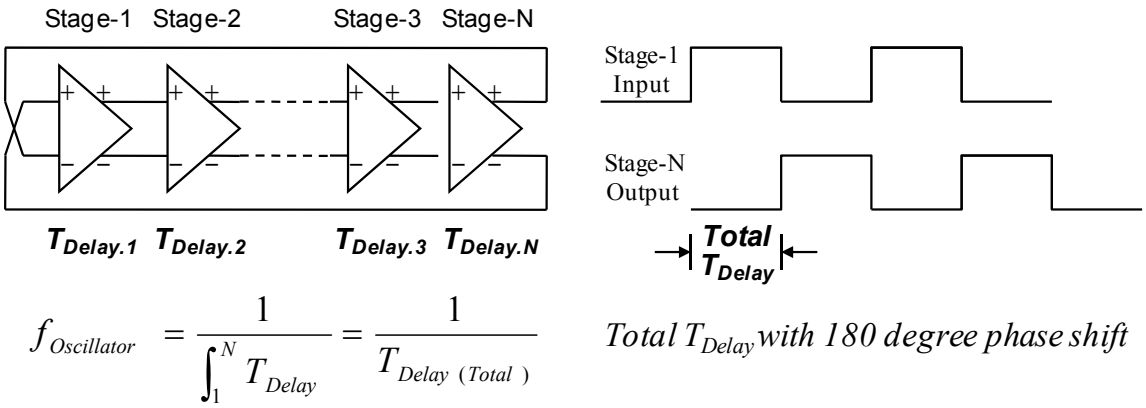


Figure 4.6 Illustration of Ring VCO oscillation and stage delay.

$$f_{RING} = \frac{1}{2Nt_{Delay}} \approx \frac{1}{\eta N(t_R + t_F)} \approx \frac{1}{2\eta N t_R} \approx \frac{I_{RING}}{2\eta N C_L V_{RING}} \approx \frac{\mu_{eff} W_{eff} C_{OX} \Delta V_{RING}^2}{8\eta N L_{eff} C_L V_{RING}} \tag{4.14}$$

$$\Delta V_{RING} = \frac{V_{RING}}{2} - V_{TH} \tag{4.15}$$

$$I_{RING} \approx 2\eta N C_L V_{RING} f_{RING} \propto N C_L f_{RING} \tag{4.16}$$

where N and t_{Delay} are the number of stages and delay time for each stage, respectively. t_R and t_F are the signal rise time and fall time, respectively, associated with maximum slew rate. For a Ring VCO design, one can assume $t_R \approx t_F$. η is a proportionality constant less than 1 [40]. $I_{RING} = (I_B+I_C)/2$ is the current dissipated in each delay stage in Figure 4.5. ΔV_{RING}, C_L and V_{RING} are the gate overdrive voltage, equivalent capacitive load in each delay stage and the voltage amplitude, respectively. V_{TH} is the MOSFET threshold voltage. For a constant oscillation amplitude, the Ring VCO current dissipation is

proportional to the oscillation frequency and the capacitive load as shown in Equation 4.16.

Based on the required oscillation frequency, allowable tail current and preferred oscillation amplitude, one can estimate the required device sizes for the delay stage from Equation 4.14. The power dissipation can be calculated with Equation 4.17:

$$P_{RING} = (I_B + I_C)V_{DD} = 2I_{RING}V_{DD} \tag{4.17}$$

The PFD, CP and LF normally dissipate negligible power compared to the power used by the VCO. However, the LF layout area may take up a large percentage of the overall PLL area. This, in turn, depends on the required capacitor size used in the LF in order to meet the PLL bandwidth requirement. The power dissipation of the Clock Buffer and Divider shown in Figure 4.1 can be estimated using the first order calculation of Equation 4.18 [40].

$$P = \frac{1}{2}(\sum_i C_i f_i)V_{DD}^2 \tag{4.18}$$

Here, f_i and C_i are the individual clock frequency and parasitic capacitance at the nodes of interest. The power dissipation is proportional to the capacitive load, operating frequency and the square of supply voltage.

4.2.3 PERFORMANCE AND CIRCUIT ANALYSIS

PLL design starts from a target output clock frequency with a required jitter performance as well as the available reference clock frequency and the power / area constraints discussed above. Depending on the application, other requirements may also have to be met such as loop stability constraints, settling-time, hold / pull-in / lock-in range, and so on. In this work, the jitter performance, power and area are used to

determine the trade-off between LC and Ring VCO based PLLs. From stability considerations, the designed PLL loop bandwidth should be less than 1/10 of the input reference clock frequency [2] and the preferred closed loop phase margin is greater than 50 degrees.

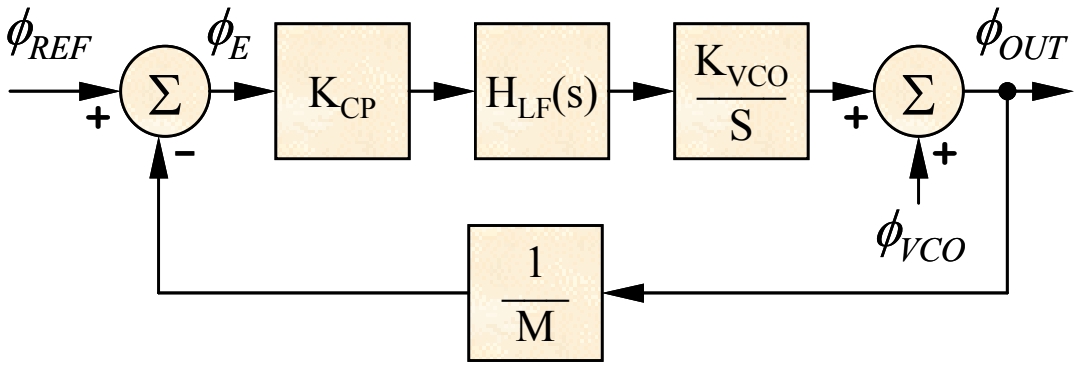


Figure 4.7 Linear model of a PLL.

Figure 4.7 illustrates the linear phase transfer function model of a PLL design based on Figure 4.1, which provides an intuitive way to derive the phase noise transfer function from the VCO input and reference clock input to the PLL output as shown in Equations 4.19~4.20 [3-4, 20,42].

$$\frac{\phi_{OUT}(s)}{\phi_{VCO}(s)} = \frac{sM}{sM + K_{CP}H_{LF}(s)K_{VCO}} \tag{4.19}$$

$$\frac{\phi_{OUT}(s)}{\phi_{REF}(s)} = \frac{K_{CP}H_{LF}(s)K_{VCO}M}{sM + K_{CP}H_{LF}(s)K_{VCO}} \tag{4.20}$$

where $\phi_{OUT}(s)$, $\phi_{VCO}(s)$ and $\phi_{REF}(s)$ are phase errors at the PLL clock output, VCO input, and reference clock input, respectively. K_{CP}, $H_{LF}(s)$, and K_{VCO} are the gains for PFD-CP, LF, and VCO, respectively. M is the PLL feedback divider ratio. Equations 4.19~4.20 indicate that the PLL is a high-pass filter to the VCO input noise but a low pass filter to reference clock input noise. Equation 4.20 also indicates that the phase noise from the

input reference clock linearly increases with divider ratio, M, when s→0.

An example of the power spectral density of phase noise and timing jitter transfer function plots of the VCO and PLL are shown in Figure 4.8 [42], where $S\phi_{OUT}(\omega)$ is the phase noise power spectrum and σ_τ is the RMS jitter in τ seconds The typical VCO phase noise and timing jitter of Figure 4.8(a)~(b) illustrate that the majority of VCO noise close to the oscillation frequency is from the up-conversion of the *1/f* device flicker noise.

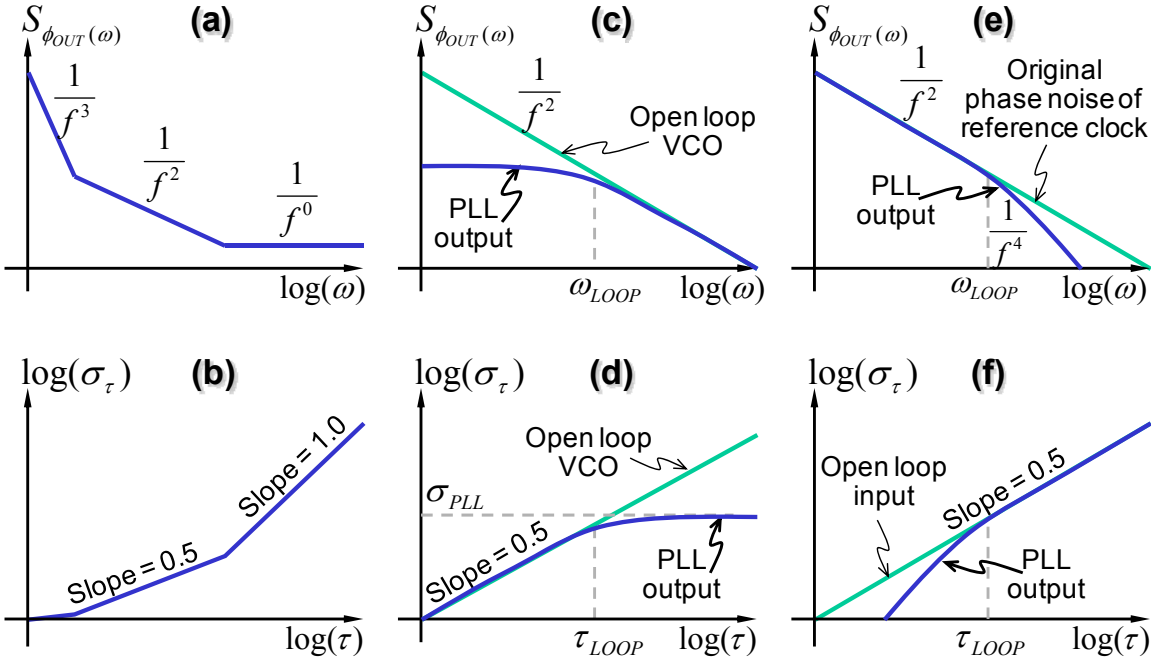


Figure 4.8 (a)~(b) phase noise and timing jitter of a free running VCO, (c)~(d) phase noise and timing jitter of a PLL with ideal reference clock and noisy VCO, (e)~(f) phase noise and timing jitter of a PLL with noisy reference clock and ideal VCO.

The phase noise and timing jitter for Equation 4.19 with a high-pass noise frequency response from the VCO input to the PLL output is illustrated in Figure 4.8(c)~(d). Here, ω_{LOOP} and τ_{LOOP} are the PLL loop bandwidth and loop time constant, respectively. A higher PLL loop bandwidth would result in less noise transferred from VCO input to PLL

output.

The phase noise and timing jitter of Equation 4.20 with a low-pass noise frequency response from the reference clock input to the PLL output is illustrated in Figure 4.8(e)~(f). A lower PLL loop bandwidth would result in less noise transferred from the reference clock input to the PLL output. The phase noise to timing jitter conversion is calculated through Equation 4.21 [43]:

$$Jitter_{[\sec]} = \frac{\sqrt{\int_{-\infty}^{+\infty} 10^{\frac{PhaseNoise_{[dBc/Hz]}}{10}}}}{2 \cdot \pi \cdot f_{PLL}} = \frac{\sqrt{2\int_{0}^{\infty} 10^{\frac{PhaseNoise_{[dBc/Hz]}}{10}}}}{2 \cdot \pi \cdot f_{PLL}} \tag{4.21}$$

where f_{PLL} is the PLL output frequency. In practical simulations, one only needs to integrate the phase noise power within 10X of the PLL loop bandwidth since the phase noise outside of this range is negligible.

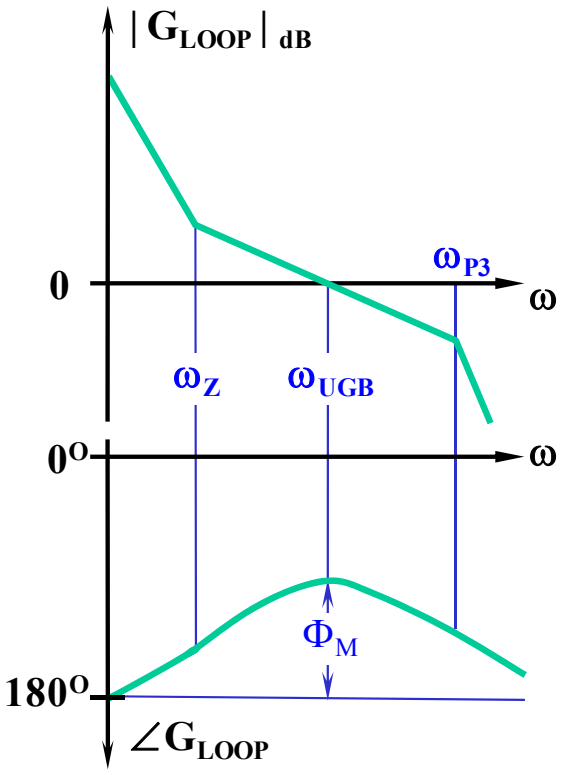
Based on Equations 4.19~4.20 and Figure 4.8(c)~(f), an increase in loop bandwidth will suppress the effects of the internal VCO jitter and the VCO control voltage noise. On the other hand, a decrease in loop bandwidth will reduce the external input and the phase detection noise. Therefore, PLL bandwidth must be optimized in order to minimize the phase noise at the PLL output. The optimized PLL loop bandwidth for minimum phase noise for a given PLL architecture is expressed in 4.22 [12].

$$\omega_{LOOP(opt)} = f_{REF} \cdot \left(\frac{\delta\tau_{VCO} + \frac{K_{VCO}}{2\pi \cdot f_{REF}^{\,2}} \cdot \frac{\delta V_{LF}}{M}}{\delta\tau_{REF} + \frac{1}{2\pi \cdot f_{REF}} \cdot \frac{\delta I_{CP}}{K_{CP}}}\right) \tag{4.22}$$

where f_{REF}, $\delta\tau_{VCO}$, $\delta\tau_{REF}$, δV_{LF} and δI_{CP} are the input reference clock frequency, VCO RMS timing jitter, reference clock RMS timing jitter, LF RMS voltage noise and CP RMS current noise, respectively.

The next step is to derive the required device sizes for the LF based on the given

optimized PLL loop bandwidth and the desired PLL loop phase margin. An example of a type-II 3rd order conceptual PLL loop gain and phase Bode plot is shown in Figure 4.9[16]. It shows that the 3rd pole degrades the PLL closed-loop phase margin.



Phase Margin:

$$\Phi_M = \tan^{-1}\left(\frac{\omega_{UGB}}{\omega_Z}\right) - \tan^{-1}\left(\frac{\omega_{UGB}}{\omega_{P3}}\right)$$

Zero Frequency:

$$\omega_Z = \frac{1}{R \cdot C_1}$$

3rd Pole Frequency:

$$\omega_{P3} = \frac{1}{R \cdot \left(\frac{C_1 \cdot C_2}{C_1 + C_2}\right)}$$

Unity Gain Bandwidth Frequency:

$$\omega_{UGB} = \omega_Z \sqrt{\frac{C_1}{C_2} + 1}$$

Figure 4.9 A type-II 3rd order conceptual PLL loop gain Bode plot.

However, optimized LF capacitor and resistor size may lead to optimized phase margin, which results in a PLL loop that is relatively immune to process, supply and temperature variations [16]. It is important to optimize the phase margin, Φ_M by carefully choosing the LF capacitor ratio with Equation 4.23 [16]. Then, the optimized loop bandwidth in Equation 4.22 and capacitance ratio from Equation 4.23 can be used to derive the optimal zero frequency, ω_Z by Equation 4.24. The results from Equations 4.23 and 4.24 also lead to the optimal 3rd pole location by Equation 4.25. The final

capacitance and resistance values are calculated through Equations 4.26~4.28, where the 3rd pole C_2 is derived from the PFD-CP gain, K_{CP}, and VCO gain, K_{VCO}, in Equation 4.26 [19,44].

$$\frac{C_1}{C_2} = 2 \cdot (\tan^2 \Phi_M + \tan \Phi_M \sqrt{\tan^2 \Phi_M + 1}) = \alpha \tag{4.23}$$

$$\omega_{LOOP(opt)} \approx \omega_{UGB} = \omega_Z \sqrt{\frac{C_1}{C_2} + 1} \Rightarrow \omega_Z = \frac{\omega_{UGB}}{\sqrt{\frac{C_1}{C_2} + 1}} \tag{4.24}$$

$$\Phi_M = \tan^{-1}(\frac{\omega_{UGB}}{\omega_Z}) - \tan^{-1}(\frac{\omega_{UGB}}{\omega_{P3}}) \Rightarrow \omega_{P3} = \frac{\omega_{UGB}}{\tan(\tan^{-1}(\frac{\omega_{UGB}}{\omega_Z}) - \Phi_M)} \tag{4.25}$$

$$\omega_{P3} = \frac{C_1 + C}{RC_1C_2} \Rightarrow C_2 = \frac{(C1 + C2)\omega_Z}{\omega_{P3}} = \frac{K_{CP}K_{VCO}}{M\omega_{UGB}\omega_{P3}} \tag{4.26}$$

$$C_1 = \alpha \cdot C_2 \tag{4.27}$$

$$R = \frac{1}{\omega_Z \cdot C_1} \tag{4.28}$$

The phase noise performance of differential LC and Ring VCOs are expressed in Equations 4.29~4.30 [5].

$$L_{LC}\{\Delta\omega_{LC}\} = \frac{FR_ST\kappa}{V_{LC}^2 Q_{LC}} \frac{\omega_{LC}^2}{\Delta\omega_{LC}^2} \tag{4.29}$$

$$L_{RING}\{\Delta\omega_{RING}\} = \frac{8NT\kappa}{3\eta P_{RING}} (\frac{V_{DD}}{\Delta V} + \frac{V_{DD}}{V_{RING}}) \frac{\omega_{RING}^2}{\Delta\omega_{RING}^2} \tag{4.30}$$

where F, T, *and* κ are Leeson's noise factor, temperature and Boltzmann's constant, respecitvely. A simulation-based method to estimate the effective phase noise of a PLL and its individual blocks can be found in [45].

4.3 SIMULATION COMPARISON AND MEASURED RESULTS

We use Cadence SpectreRF to perform the PLL phase noise simulation. Reference

[46] has suggested a good correlation between measured and Cadence SpectreRF simulated phase noise / jitter results. According to Equation 4.12, LC VCO based PLLs will have the same power dissipation from low to high frequency. However, from Equation 4.16, Ring VCO based PLLs will dissipate less power at low frequency.

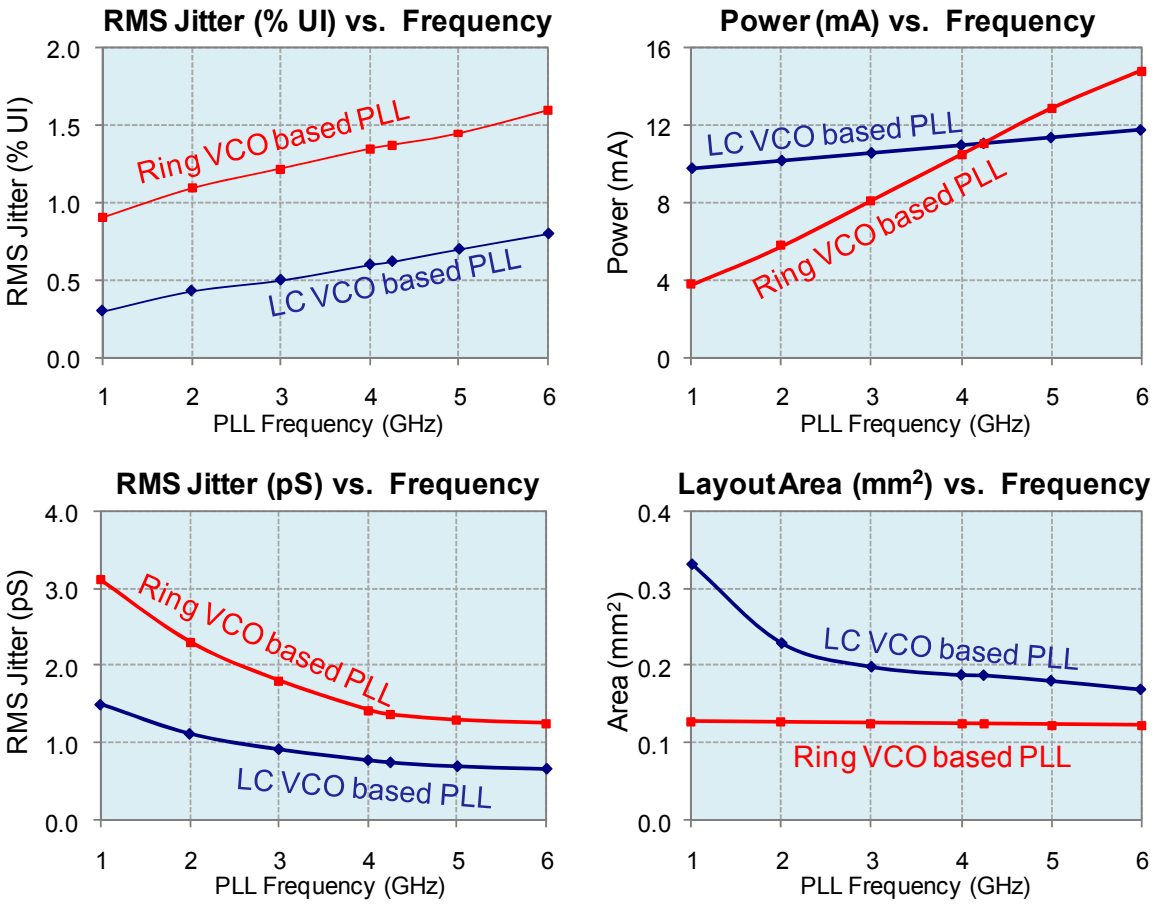


Figure 4.10 PLL Jitter, Power, and Area Comparison.

To simplify our experiments, we set the LC and Ring PLLs to dissipate the same power at 4.25 GHz, and kept the same jitter performance ratio between LC and Ring PLLs for all frequencies, as shown in Figure 4.10. The PLL feedback divider ratio is set to be 40 for all frequencies from 1 ~ 6 GHz while the reference frequency varies from 25 ~ 150 MHz. The layout area for the Ring PLL has almost no dependence on the

operating frequency. However, the LC PLL layout area varies from 3X to 1.5X of the Ring PLL layout area for frequencies from 1 GHz to 6 GHz, which results from inductor size reduction when moving from low to high frequency operation. The RMS random jitter stays relatively fixed at a 2X ratio between the Ring and LC PLLs in this study. According to Figure 4.10, the LC PLL has less area disadvantage at higher operating frequencies, but no advantage in either power dissipation or layout area for low frequency applications.

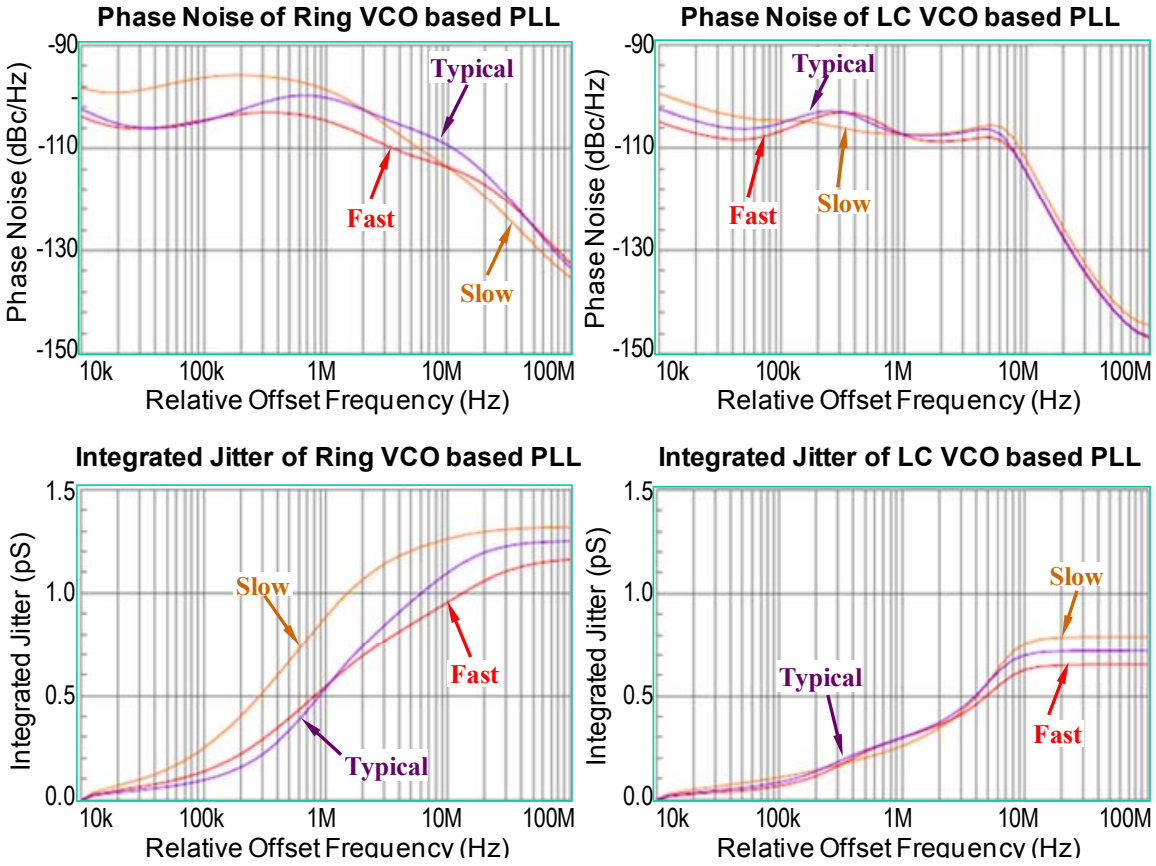


Figure 4.11 Simulated Phase noise and integrated RMS jitter for both Ring and LC VCO based PLLs at 106.25 MHz input reference and 4.25 GHz output clock.

The targeted gain peaking for the jitter transfer function is less than 3dB, and the preferred PLL lock time is less than 2 μS in order to meet specification requirements for

applications such as the PCI Express [47].

Two of each LC and Ring VCO based PLLs were designed and fabricated in a standard 90 nm digital CMOS process. The designed frequency operating ranges for LC VCO based PLLs are 2 ~ 3 GHz for one and 3 ~ 4.5 GHz for the other one, which represents ± 20% of frequency tuning from their center frequency. The designed frequency operating ranges for Ring VCO based PLLs are 1.6 ~ 4 GHz for one and 2 ~ 5 GHz for the other, which represents 2.5X tuning range from the minimum to maximum frequency. The common frequency operating range between LC and Ring PLLs is 2 ~ 4.5 GHz. More than 2X tuning range from the minimum to maximum frequency allows PLLs to generate any frequency equal to or below 4.5 GHz by using a divider at the output of PLL. For example, a 1.3 GHz clock will be running from PLL at 2.6 GHz with a ÷2 divider at PLL output.

Figure 4.11 shows an example of the simulated phase noise spectrum and the integrated RMS random jitter versus the relative offset frequency for the designed Ring and LC PLLs with the input reference clock R_{CLK} = 106.25 MHz, and the PLL output clock, PLL_{CLK} = 4.25 GHz (i.e., divider ratio M=40). The results of Figure 4.11 imply that a Ring PLL with an RMS jitter less than 1.5 pS can meet the specification requirement of 8.5 Gbps Fiber Channel designs [48].

The designed and fabricated VCO and PLL layouts for LC and Ring PLLs are shown in Figure 4.12 and Figure 4.13. Only Ring VCO and PLL layouts with a 2 ~ 5 GHz tuning range are shown in Figure 4.12 and Figure 4.13, respectively, because of no major layout differences between two designed Ring PLLs with 1.6 ~ 4 GHz and 2 ~ 5 GHz tuning ranges, respectively. From the VCO and PLL layout plots in Figure 4.12 and Figure

4.13, the LC PLL designs do have layout differences between 2 ~ 3 GHz and 3 ~ 4.5 GHz tuning ranges.

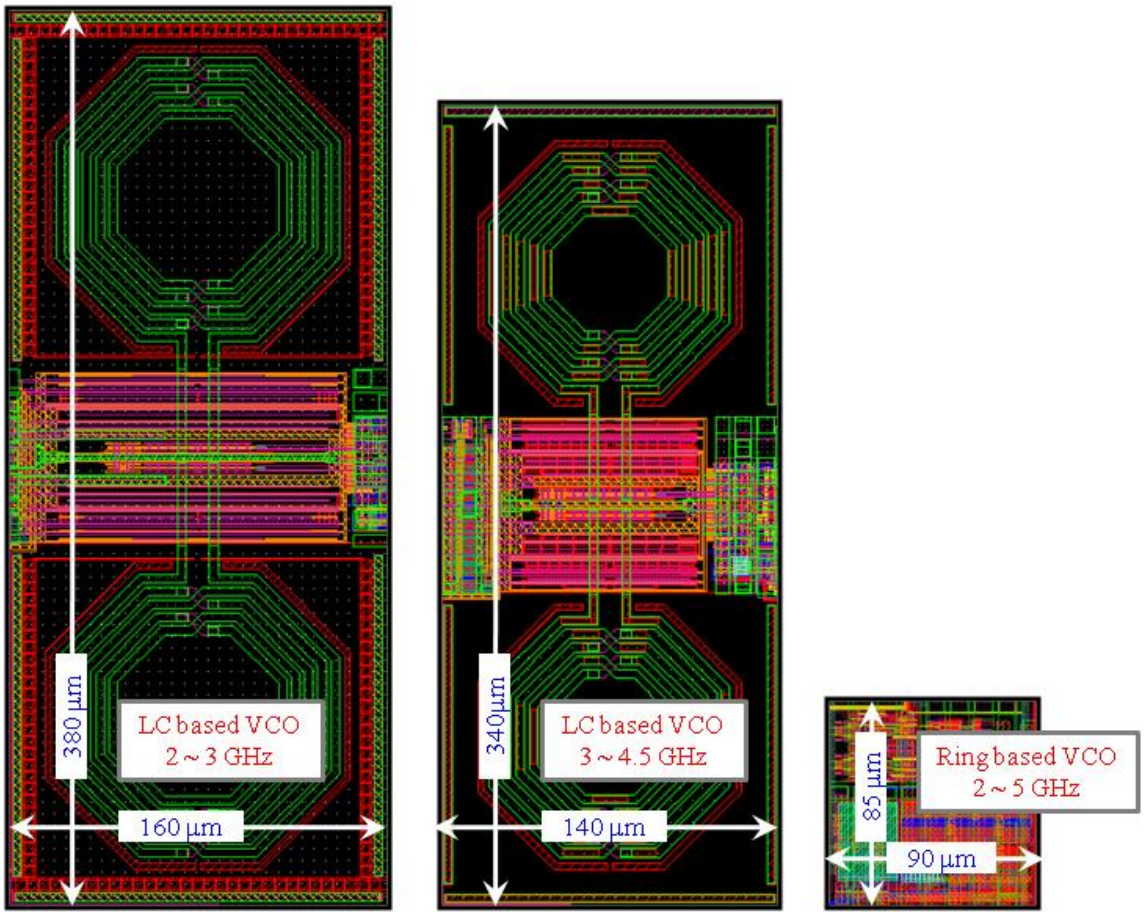


Figure 4.12 LC VCO and Ring VCO Layout

The layout ratio of two LC VCOs in Figure 4.12 is about 1.3. However, the layout ratio of the two LC PLLs is only 1.07. In another comparison, the layout ratio of LC and Ring VCOs is about 6 ~ 8 in Figure 4.12. However, the layout ratio of LC and Ring PLLs drops to about 1.5 ~ 1.6 in Figure 4.13. For all PLLs in this comparison, the same PFD, PD and divider with a similar size of LF are used.

The jitter analysis measurement for the PLL designs uses a Tektronix TDS6124C digital storage oscilloscope, which supports up to 12 GHz / 40 Gbps bandwidth and has

enough resolution to characterize the PLL designs with a tuning range from 2 to 5 GHz. Figure 4.14 shows the measured LC and Ring PLL frequency spectra at 4.25 GHz, respectively. The plot for the LC PLL has a lower noise frequency spectrum relative to oscillation frequency compared with the plot for the Ring PLL, which clearly gives a better jitter performance.

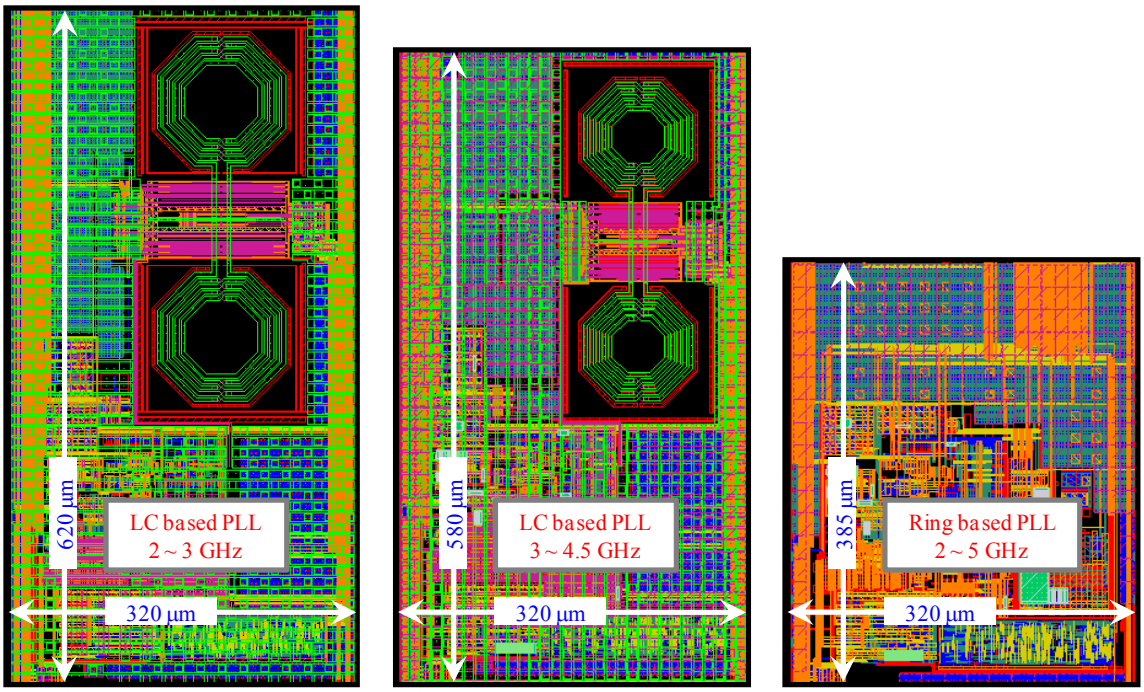


Figure 4.13 LC and Ring based PLL Layout

Simulated and measured RMS random jitter for both the LC and Ring PLL are shown in Figure 4.15. The measured LC PLL RMS random jitter result shows lower values than the simulated ones at higher frequency. From the initial investigation, the lower measured LC PLL RMS random jitter is likely caused by the overestimated interconnect resistance between inductor and VCO or inductor series resistance. On the other hand, the Ring PLL measured RMS jitter shows higher values than the simulated RMS jitter at

lower frequency. From transient simulation, the PLL shows worse jitter performance at lower frequency if PMOS and NMOS devices are skewed in opposite directions. The higher jitter is caused unbalanced rising / falling edges when lower VCO amplitude and oppositely skewed PMOS and NMOS are both present at lower operating frequency. Generally, the measured jitter analysis shows a good correlation with simulated data even though there are some discrepancies.

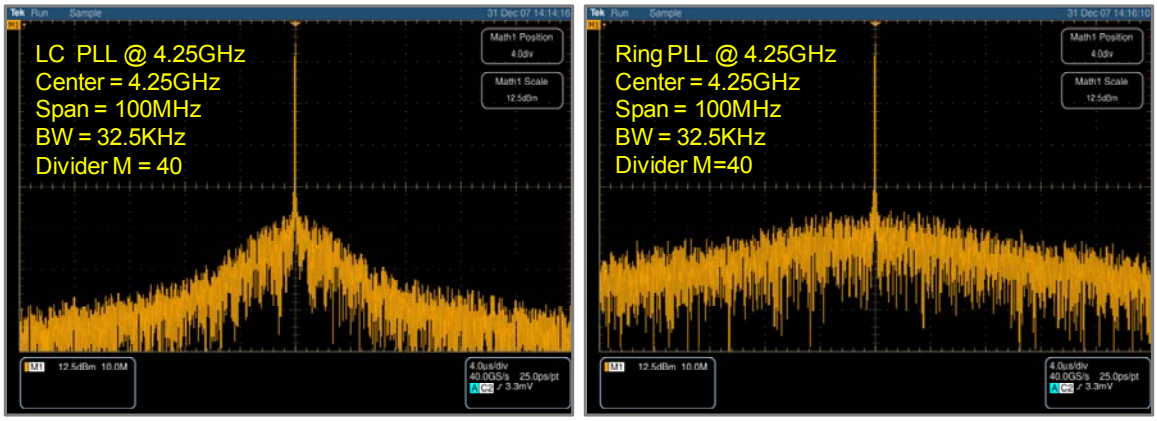


Figure 4.14 Measured LC and Ring VCO based PLL frequency spectrum at 4.25 GHz, respectively.

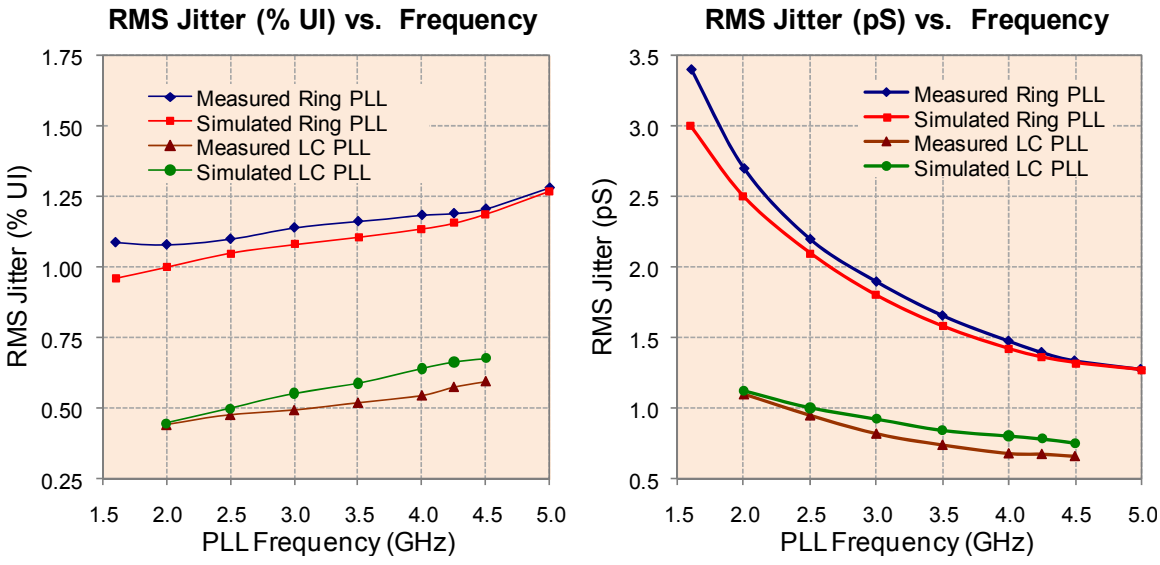


Figure 4.15 Measured RMS Jitter for Ring and LC VCO based PLL Designs

Ring PLLs typically have more than a 2X frequency tuning range, while LC PLLs normally have a tuning range of approximately ± 20% around the center frequency. For a PLL design requiring a wide frequency range, an LC VCO based PLL would need multiple VCOs to switch between frequency ranges, which would make the LC VCO more undesirable in terms of the layout area.

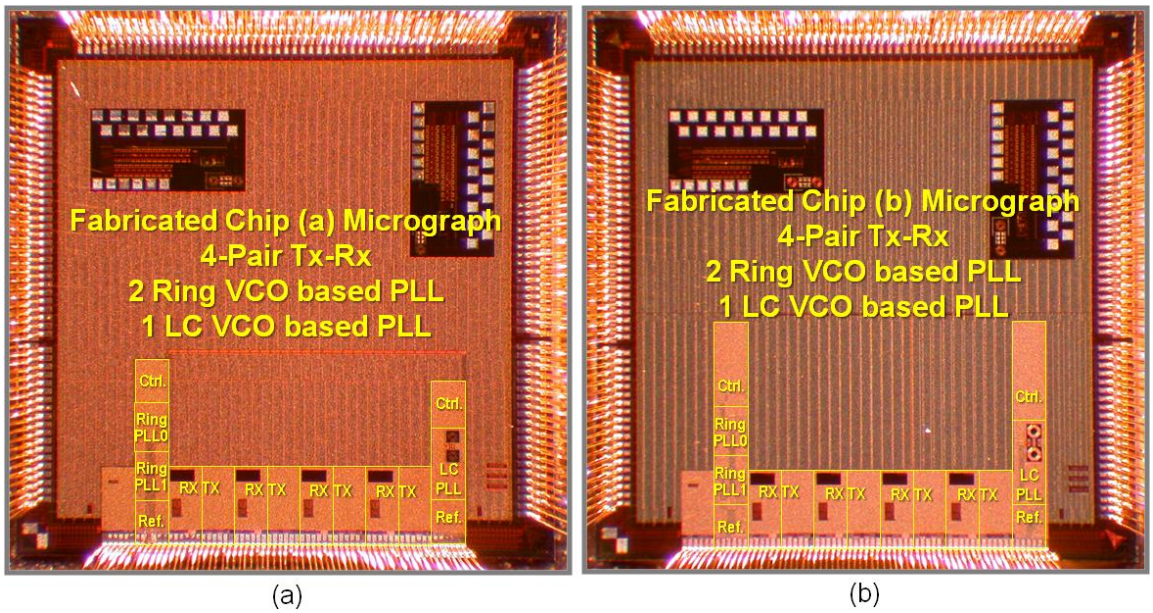


Figure 4.16 Fabricated chips with different frequency tuning ranges. (a) LC PLL 3 ~ 4.5GHz and Ring PLL 2 ~ 5 GHz (b) LC PLL 2 ~ 3 GHz and Ring PLL 1.5 ~ 4GHz.

On the other hand, an LC PLL generates a lower RMS jitter than a Ring PLL for a given power and frequency because of its high-Q frequency response in the VCO. It is possible to have Ring PLLs with a lower jitter value close to that of the LC PLL jitter performance of Figure 4.10 if the power dissipation or the VCO amplitude is increased. However, the improvement in Ring PLL jitter performance is only by an approximately square-root relationship. This implies about a 4X increase in power dissipation is needed for Ring PLLs to achieve a similar jitter performance as for LC PLLs, which is very undesirable in low power applications.

Figure 4.16 shows the chips which were fabricated in a standard 90 nm digital CMOS process, where "Ref." is a bandgap reference bias generator and "Ctrl." is the built-in controller and self-tester. Each chip consists of 4 pairs of transmitters (Tx) and receivers (Rx) for multi-channel applications.

Table 4.1 LC and Ring VCO based PLL trade-off.

	ADVANTAGES	**DISADVANTAGES**
Ring VCO based PLL	Common approach for digital chips Many ways to control frequency Multi-phase clock generation Wide frequency tuning range	High phase noise → widen loop BW to reduce High VCO gain → sensitive to disturbance Not suitable for SONET transmit clocks Poor stability at high frequency
LC VCO based PLL	Common approach for RF design Good stability Long-term and period jitter filtering Low long-term and period jitter Low phase noise	Large layout area → large area for inductor Narrow tuning frequency range Require a lot of characterization Poor integration and more complicated design

Table 4.1 presents a summary of the trade-off between Ring and LC VCO based PLL designs. Each of them has its particular advantages and disadvantages. Generally, a Ring VCO based PLL has wider loop bandwidth with higher phase noise response and is most suitable for clock and data recovery circuits. The power dissipation proportional to the operation frequency also leads a Ring VCO based PLL to be more attractive for a low power and low frequency application. On the other hand, an LC VCO based PLL has the power dissipation relatively constant to the operation frequency and narrower loop bandwidth with superior phase noise response, which is well matched for high frequency RF applications.

4.4 CONCLUSION

We have presented a comparison of LC and Ring VCO based PLL designs in a standard 90 nm digital CMOS process technology. The power, layout area, operating frequency and loop bandwidth optimization have all been considered in order to obtain a fair comparison. The results show that a Ring VCO based PLL design can meet the performance requirements of a certain high-speed clock synthesizer application if the PLL loop bandwidth and power dissipation are properly optimized. Furthermore, a Ring VCO based PLL has the advantages of larger tuning range and smaller layout area compared with an LC VCO based PLL. However, an LC VCO based PLL design exhibits a superior phase noise performance. It also has a larger power advantage at higher frequency. Furthermore, the impact on layout area from the inductors is less of a concern for higher frequency operation due to the smaller inductors which are needed.

REFERENCES

[1] J. G. Maneatis, "Selecting PLLs for ASIC Applications Requires Tradeoffs," *2004 FSA Semiconductor IP Workshop*, Oct. 7, 2004

[2] D. Fischette, "Practical Phase-Locked Loop Design," *IEEE Journal of Solid-State Circuits Tutorial,* http://www.delroy.com/pll, 2004

[3] J. Savoj, and B. Razavi, "High-Speed CMOS Circuits for Optical receivers," *Kluwer Academic Publisher,* Boston, 2001

[4] B. Razavi, "Design of Integrated Circuits for Optical Communications," *McGraw-Hill*, New York, 2003

[5] T. Miyazaki, M. Hashimoto, and H. Onodera, "A Performance Comparison of PLLs

for Clock Generation Using Ring Oscillator VCO and LC Oscillator in a Digital CMOS Process," *IEEE 2004 Asia and South Pacific Design Automation Conference,* pp.545-546, Jan. 2004

[6] D. Ham and A. Harjimiri, "Concepts and Method in Optimization of Integrated LC VCOs," *IEEE Journal of Solid-State Circuits*, vol. 36, no.6, pp.896-909, Jul. 2001

[7] A. Jerng and C.G. Sodini, "The Impact of Device Type and Sizing on Phase Noise Mechanisms," *IEEE Journal of Solid-State Circuits*, vol. 40, no.2, pp.360-369, Jul. 2001

[8] X. He, W. Kong, T. Firestone, R. Newcomb, and M. Peckerar, "Phase Noise Optimization of A Symmetric CMOS LC VCO," *IEEE 2006 International Symposium on Industrial Electronics,* vol.4, pp.2820-2823, Jul. 2006

[9] R. Navid, T. H. Lee, and R.W. Dutton, "Minimum Achievable Phase Noise of RC Oscillators," *IEEE Journal of Solid-State Circuits*, vol.40, no.3, pp.630-637, Mar. 2005

[10] L. Dai, and R. Harjani, "Design of Low-Phase-Noise CMOS Ring Oscillators," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol.49, no.5, pp.328-338, May 2002

[11] A.A. Abidi, "Phase Noise and Jitter in CMOS Ring Oscillators," *IEEE Journal of Solid-State Circuits*, vol.40, no.3, pp.1803-1816, Aug. 2006

[12] K. Lim, C.-H. Park, A.-S. Kim and B. Kim, "A Low-Noise Phase-Locked Loop Design by Bandwidth Optimization," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 6, pp.807-815, Jun. 2000

[13] M. Mansuri and C.-K. Ken, "Jitter Optimization based on Phase-Locked Loop

Design Parameters," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 11, pp.1375-1382, Nov. 2002

[14] S. Dosho, N. Yanagisawa, and A. Matsuzawa, "A Background Optimization Method for PLL by Measuring Phase Jitter Performance," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 4, pp.941-950, Apr. 2005

[15] M.-t. Hsieh and G. E. Sobelman, "Comparison of LC and Ring VCOs for PLLs in a 90 nm Digital CMOS Process," *Proceedings, International SoC Design Conference,* pp.19-22, 2006

[16] Z. Wang, "An Analysis of Charge-Pump Phase-Locked Loops," *IEEE Transaction on Circuits and System – 1: Regular Paper,* vol.52, no.10, pp.2128-2138, Oct. 2005

[17] P.K. Hanumolu, M. Brownlee, K. Mayaram, and U.-K. Moon, "Analysis of Charge-Pump Phase Locked Loops," *IEEE Transaction on Circuits and System – 1: Regular Paper,* vol.51, no.9, pp.1665-1674, Sept. 2004

[18] F. You and S. He, " Analysis of Third-Order Charge Pump PLL," *IEEE International Conference on Communications, Circuits and Systems*, vol.2, pp.1372-1376, 2004

[19] A.L.S. Loke, et al, "A Versatile 90-nm CMOS Charge-Pump PLL for SerDes Transmitter Clocking," *IEEE Journal of Solid-State Circuits*, vol.41, no.8, pp.1894-1907, Aug. 2006

[20] A.Y.V. Lopez, "Design of Frequency Synthesizer for Short Range Wireless Transceivers," *Ph.D. Thesis, Texas A&M University,* May 2004

[21] B. Razavi, "Design of Monolithic Phase-Locked Loops and Clock Recovery Circuits: A Tutorial," *In Monolithic Phase-Locked Loops and Clock Recovery*

Circuits. Theory and Design (B. Razavi, ed.), *IEEE Press*, New York, pp.1-39, 1996

[22] M.H. Perrott, "Design and Simulation of Fractional-N Frequency Synthesizers: A Tutorial," *In Analog Circuit Design Fractional-N Synthesizers, Design for Robustness, Line and Bus Drivers (A.H. M. van Roermund, M. Steyaert, J.H. Huijsing, ed.)*, *Kluwer Academic Publisher*, Norwell, pp.27-49, 2003

[23] A. Schultz, "Selecting Passive Components for PLLs," *Electronics Design, Strategy, News*, http://www.edn.com, Mar. 20, 2003.

[24] M. Rachedine, et al, "Performance Review of Integrated CMOS VCO Circuits for Wireless Communications," *IEEE Radio Frequency Integrated Circuits (RFIC) Symposium,* pp.77-80, Jun. 2003

[25] A. Fard, T. Johnson, M. Linder, and D. Aberg, "A Comparative Study of CMOS LC VCO Topologies for Wide-Band Multi-Standard Transceivers," *IEEE Midwest Symposium on Circuits and Systems,* vol.3, pp.17-20, Jul. 2004

[26] Y. Park, S. Chakraborty, C.-H. Lee, S. Nuttinck, and J. Laskar, "Wide-Band CMOS VCO and Frequency Divider Design for Quadrature Signal Generation," *IEEE MTT-S International Microwave Symposium Digest*, vol.3, pp.1493-1496, Jun. 2004

[27] Z. Wang, H.S. Savci, and N.S. Dogan, "1-V Ultra-Low-Power CMOS LC VCO for UHF Quadrature Signal Generation," *IEEE International Symposium on Circuits and Systems*, pp.4022-4025, May 2006

[28] A. Hajimiri and T.H. Lee, "Design Issues in CMOS Differential LC Oscillators," *IEEE Journal of Solid State Circuits*, vol.32, no.5, pp.717-724, May 1999

[29] G. De Astis, D. Cordeau, J.-M. Paillot, and L. Dascalescu, "A 5-GHz Fully Integrated Full PMOS Low-Phase-Noise LC VCO," *IEEE Journal of Solid State*

Circuits, vol.40, no.10, pp.2087-2091, Oct. 2005

[30] K.K. O, N. Park, and D-J. Yang, "*1/f* Noise of NMOS and PMOS Transistors and their Implications to Design of Voltage Controlled Oscillators," *IEEE Radio Frequency Integrated Circuits (RFIC) Symposium*, pp.59-62, Jun. 2002

[31] M. Thamsirianunt and T.A. Kwasniewski, "CMOS VCO for PLL Frequency Synthesis in GHz Digital Mobile Radio Communications," *IEEE Journal of Solid State Circuits*, vol.32, no.10, pp.1511-1524, Oct. 1997

[32] R.J. Betancourt-Zamora and T.H. Lee, "CMOS VCOs for Frequency Synthesis in Wireless Biotelemetry," *IEEE 1998 International Symposium on Low Power Electronics and Design,* Aug. 1998

[33] L. Dai and R. Harjani, "Comparison and Analysis of Phase Noise in Ring Oscillators," *IEEE International Symposium on Circuits and Systems,* 2000

[34] D.A. Badillo and S. Kiaei, "Comparison of Contemporary CMOS Ring Oscillators," *IEEE Radio Frequency Integrated Circuits (RFIC) Symposium*, pp.281-284, Jun. 2004

[35] M. Horowitz, et al., "PLL design for a 500 MB/s interface, *IEEE International Solid-State Circuits Conference*, pp.160-161 and 282, Feb. 1993

[36] L. Dai and R. Harjani, "A Low-Phase-Noise CMOS Ring Oscillator with Differential Control and Quadrature Outputs," *IEEE International ASIC/SOC Conference*, pp.134-138, Sept. 2001

[37] M. Tiebout, "Low-Power Low-Phase-Noise Differentially Tuned Quadrature VCO Design in Standard CMOS," *IEEE Journal of Solid-State Circuits*, vol.36, no.7, pp.1018-1024, Jul. 2001

[38] J. Maget, M. Tiebout, and R. Kraus, "Influence of Novel MOS Varactors on the Performance of a Fully Integrated UMTS VCO in Standard 0.25-μm CMOS Technology," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 7, pp.953–958, July 2002

[39] A. Harjimiri, S. Limotyrakis, and T.H. Lee, "Jitter and Phase Noise in Ring Oscillator," *IEEE Journal of Solid-State Circuits*, vol. 34, no. 6, pp.790-804, Jun. 1999

[40] M.J. Deen, M.H. Kazemeini, and S. Naseh, "Performance Characteristics of an Ultra-Low Power VCO," *IEEE International Symposium on Circuits and Systems,* pp.697-700, May 2003

[41] H.-C. Tang and M.-C. Shiau, "Power Dissipation Models and Performance Improvement Techniques for CMOS Inverters with RC Line and Tree Interconnections," *IEE Proceedings G Circuits, Devices & Systems*, vol. 140, issue 6, pp.437-443, Dec. 1993

[42] A. Hajimiri, "Noise in Phase-Locked Loops," *Proc. of IEEE Southwest Symposium on Mixed-Signal Circuits*, pp.1-6, Feb. 2001.

[43] Maxim Integrated Products Inc., "Clock (CLK) Jitter and Phase Noise Conversion – Application Note 3359," *Maxim Integrated Products Inc.*, http://ww.maxim-ic.com, Step. 2004

[44] Microchip Technology Inc., "Basic PLL Filters for the rfPICTM / rfHCS – Application Note 846," *Microchip Technology Inc.*, http://www.mictochip.com, 2002

[45] K. Kundert, "Predicting the Phase Noise and Jitter of PLL-Based Frequency Synthesizers," *The Designer's Guide Community*, http://www.designers-guide.com, Ver.4g,

Aug. 2006

[46] M. Takahashi, K. Ogawa, and K.S. Kundert, "VCO Jitter Simulation and its Comparison with Measurement," *IEEE Asia and South Pacific Design Automation Conference,* vol.1, pp.85-88, Jan. 1999

[47] PCI SIG, "PCI Express Jitter White Paper - PCI Express Jitter Modeling, Revision 1.0RD," *PCI SIG,* http://www.pcisig.com , Jul. 14, 2004

[48] American National Standards Institute, "Fibre Channel Physical Interface-4 (FC-PI-4) Rev. 6.10," *American National Standards Institute*, Feb. 27, 2007

CHAPTER 5

5 PROGRAMMABLE, LOW JITTER SPREAD SPECTRUM CLOCK GENERATOR WITH HIGH EMI REDUCTION

A PLL based SSC clock generator in 90 nm CMOS having low jitter and high electromagnetic interference reduction is described in this chapter. The SSC clock generator utilizes a phase interpolator between the VCO output and the PLL feedback divider input to control the magnitude and frequency of clock spreading. The use of a phase interpolator with a phase shift up/down controller enables an up/center/down frequency spreading SSC clock. The supported operating frequency range is 2 - 4.25 GHz with rms RJ jitter < 1.3 pS and < 1.5 pS for non-SSC and SSC modes, respectively. The measured EMI reduction is 18 dB and 21 dB for 2300 ppm and 4600 ppm SSC, respectively, when operating at 3 GHz.

5.1 INTRODUCTION

High-speed serial link systems often require electromagnetic interference (EMI) reduction at the chip level in order to reduce the effort and cost to meet EMI specifications at the system level. For example, both Serial AT Attachment (SATA) [1] and Serial Attached SCSI (SAS) [2] for storage applications have suggested using Spread Spectrum Clocking (SSC) for EMI reduction. Most publications [3-7] have a fixed down-spread 0 ~ -5000 ppm SSC capability, which only meets the SATA specification. However, other types of frequency spreading SSC are also needed, such as the SAS specification [2] which has suggested using center spread +/- 2300 ppm SSC and half

down spread 0 ~ -2300 ppm SSC for EMI reduction.

Using a phase interpolator (PI) to directly modulate the VCO clock into an SSC clock in a serializer-deserializer (SerDes) transmitter-receiver (TX-RX) [3-4] introduces excessive cycle-to-cycle jitter caused by minimum phase step resolution, differential non-linearity (DNL) and integral non-linearity (INL). Another approach is to use a counter type of PLL feedback frequency divider [5-6], which often leads to large phase step shift at the divider output and causes large jitter performance degradation at the PLL output. Excessive multi-phase clock selection from the VCO with a sigma-delta modulator controller reduces the phase step shift at the divider output [7]. However, the smaller phase spacing produced by the VCO leads to a lower VCO frequency, higher power dissipation and a larger area to accommodate the increased number of clock phases. A design using a sigma-delta modulator and a fractional dual-modulus prescaler is described in [8]. A programmable SSC clock generator operating at lower clock frequencies and using a sigma-delta modulator has been described in [9]. Another programmable design uses a digital modulation wave generator outside of the loop [10].

5.2 PLL BASED SSC CLOCK GENERATOR ARCHITECTURE

The block diagram of the SSC clock generator, which is based on a 3rd order PLL architecture, is shown in Figure 5.1 It consists of a phase-frequency detector (PFD), charge pump (CP), loop filter (LF), Ring based VCO, PI and frequency divider with the PI placed between the VCO output and the divider input. The PI design uses the topology of [11] but with 96 phase steps per clock period. The number of phase steps per up or down shift can be set to -2, -1, 0, 1 or 2. This up/down phase shift capability enables the SSC clock generator to provide up, center and down frequency spreading

SSC clocks. Also, the PI controller runs at one-fourth of the VCO output clock rate. The divider is held constant during the PLL operation.

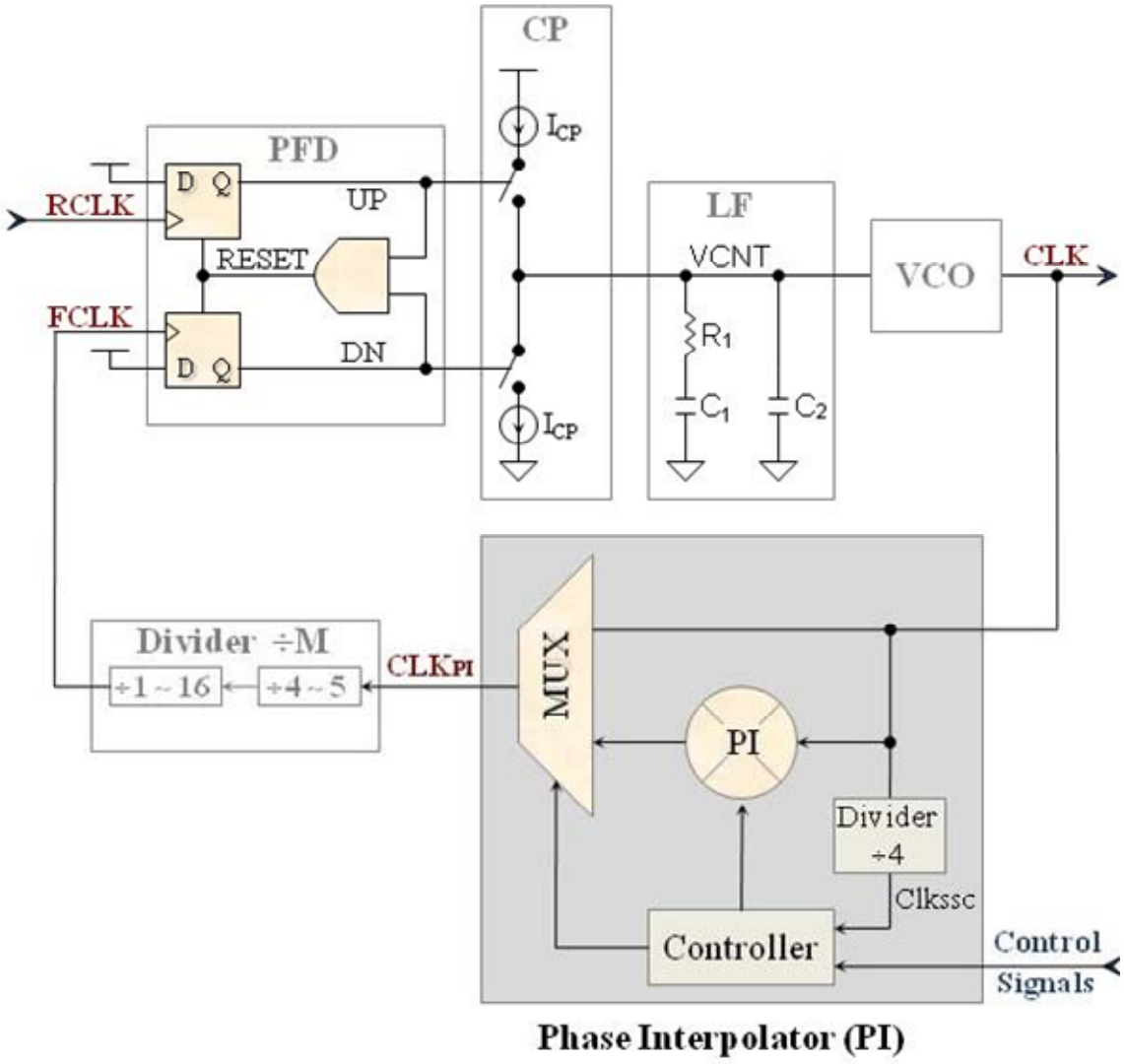


Figure 5.1 Block diagram of the PLL based SSC clock generator..

The use of the PI provides a smaller phase step shifting at the divider output compared to the designs in [4-5]. The small phase steps reduce the low frequency noise at the divider output, which leads to lower jitter. The high frequency noise from the PI into the divider can be filtered by the LF. The major concern is noise directly coupling into the VCO, which can be minimized by layout isolation between the PI and the VCO.

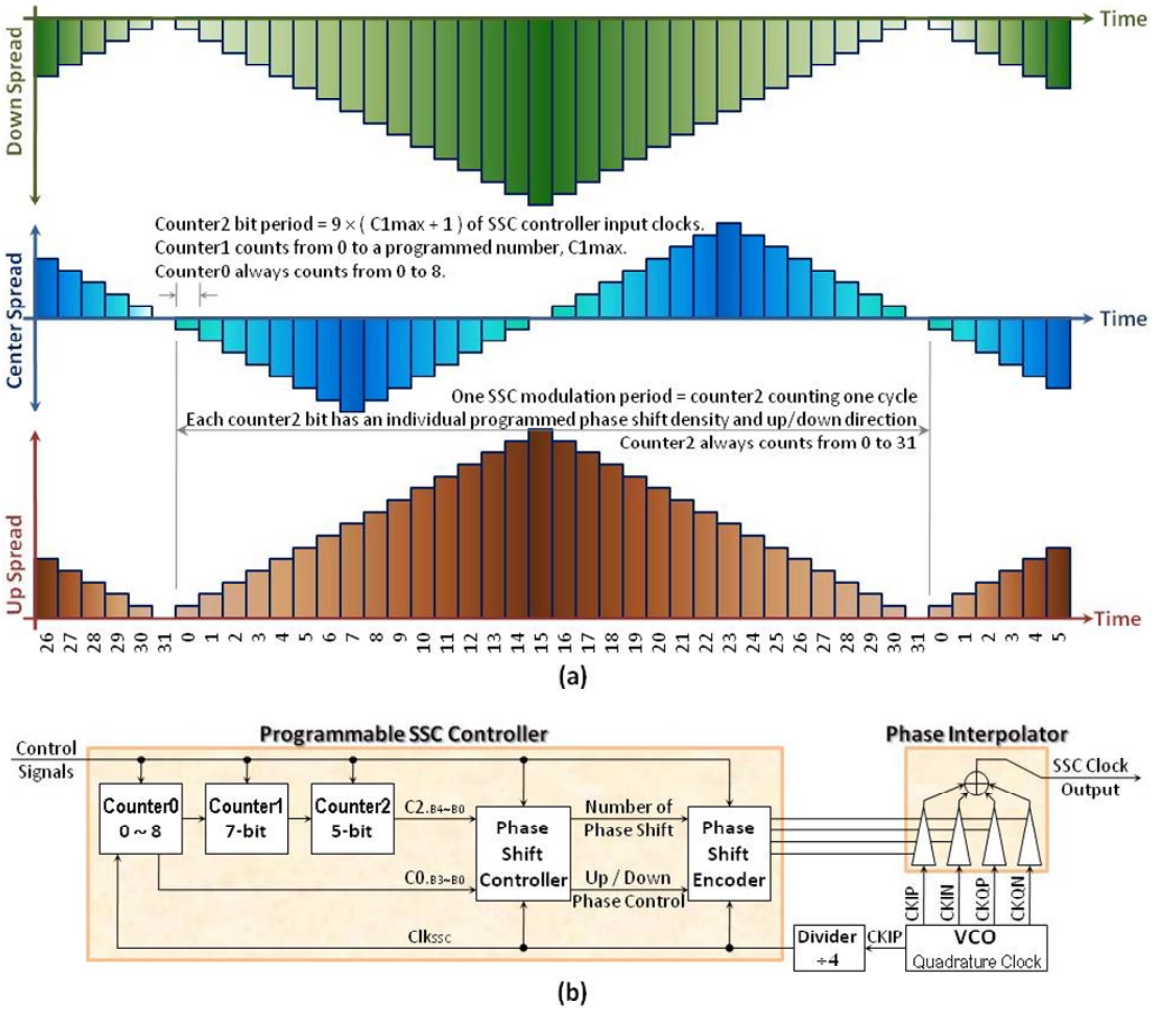


Figure 5.2 (a) Illustration of up, center, down frequency spreading SSC (b) block diagram of the programmable SSC clock generator controller.

Figure 5.2 shows the block diagram of the programmable SSC clock generator controller with an illustration of the up, center and down frequency spreading SSC. The SSC modulation frequency $f_M = f_{PLL}/[4\times9\times32\times(C1max+1)]$ is programmable by setting C1max, which is the maximum value of counter1. The factors of 4, 9 and 32 are due to the clock divider ÷4, 9 steps of counter0 and the 32 steps of counter2, respectively. f_{PLL} is the PLL output frequency. The maximum reachable frequency spreading $\Delta f_{SSC} = 2/(4\times96)\times10^6$ ppm = 5208 ppm, where the terms 2, 4 and 96 are the maximum of 2 phase shift steps, the clock divider ÷4 and with 96 phase steps in one clock period in the PI,

respectively. For this design, Δf_{SSC} = 2300 ppm and 4600 ppm with f_M = 30~33 kHz at f_{PLL} = 3GHz are used to meet both the SATA and SAS specifications. With 8 or 16 phase steps in 9 periods of Clk_{SSC} in counter0, the effective frequency spreading Δf_{SSC} = $(16/9)/(4\times96)\times10^6$ ppm = 4630 ppm ≈ 4600 ppm, or $(8/9)/(4\times96)\times10^6$ ppm = 2315 ppm ≈ 2300 ppm. For C1max = 82, the SSC modulation frequency f_M = $3\times10^9/[(4\times9\times32\times(82+1)]$ = 31.38 kHz.

5.3 DESIGN WITH SIMULATED AND MEASURED RESULTS

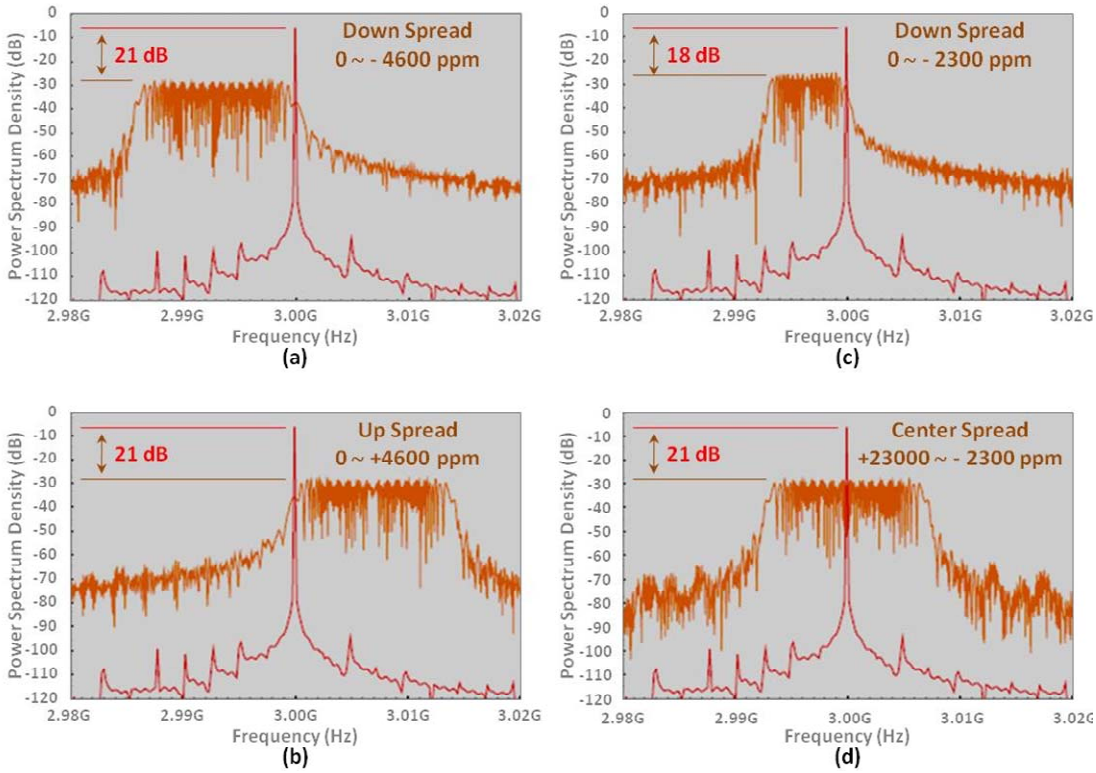
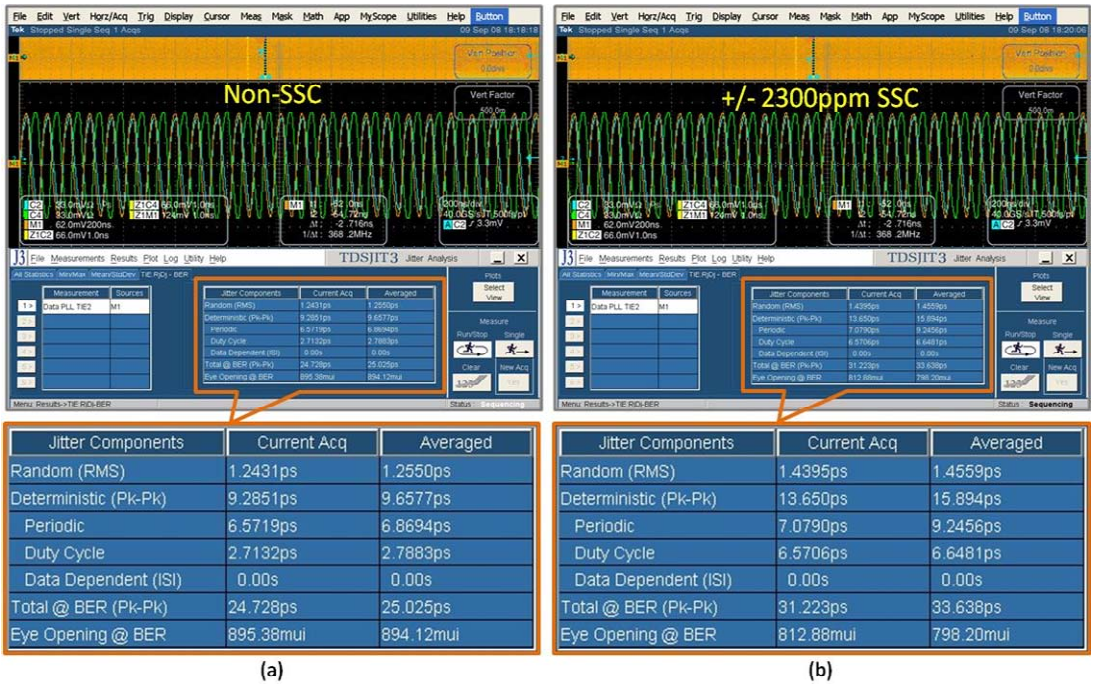


Figure 5.3 Simulated 3 GHz power spectrum density plots for (a) down spread 0 ~ -4600ppm SSC (b) up spread 0 ~ +4600ppm SSC (c) down spread 0 ~ -2300ppm SSC (d) center spread +2300 ~ -2300ppm SSC.

The simulated power spectrum density plots for the 2300 ppm and 4600 ppm modes with up, center and down frequency spreading SSC clocks is shown in Figure 5.3. Each is compared with the non-SSC power spectrum density, which shows 21dB and 18 dB

EMI reductions for peak-to-peak frequency spreading at 4600 ppm and 2300 ppm, respectively.

The measurement process uses a Tektronix TDS6124C digital storage oscilloscope which supports up to a 12 GHz / 40 Gbps bandwidth. The built-in SATA and SAS compliance configurations in the scope are used in the jitter analysis setup. The SSC clock is measured at the far-end TX output driven by the SSC clock generator using the clock pattern output.



Jitter Components	Current Acq	Averaged
Random (RMS)	1.2431ps	1.2550ps
Deterministic (Pk-Pk)	9.2851ps	9.6577ps
Periodic	6.5719ps	6.8694ps
Duty Cycle	2.7132ps	2.7883ps
Data Dependent (ISI)	0.00s	0.00s
Total @ BER (Pk-Pk)	24.728ps	25.025ps
Eye Opening @ BER	895.38mui	894.12mui

(a)

Jitter Components	Current Acq	Averaged
Random (RMS)	1.4395ps	1.4559ps
Deterministic (Pk-Pk)	13.650ps	15.894ps
Periodic	7.0790ps	9.2456ps
Duty Cycle	6.5706ps	6.6481ps
Data Dependent (ISI)	0.00s	0.00s
Total @ BER (Pk-Pk)	31.223ps	33.638ps
Eye Opening @ BER	812.88mui	798.20mui

(b)

Figure 5.4 Measured 3 GHz (a) clock period vs. time of center spread +/-2300ppm SSC (b) power spectrum density plot of non-SSC (c) power spectrum density plot of center spread +/-2300ppm SSC.

Figure 5.4 shows the detailed jitter measurement results at 3 GHz with both non-SSC and center spread +/-2300 ppm SSC. The measured results indicate that the degradation of jitter performance caused by SSC is small. It has the same amount of increased rms random jitter (RJ) going from the non-SSC to the SSC mode but at twice the clock frequency and with only half the total rms RJ compared to reference [6].

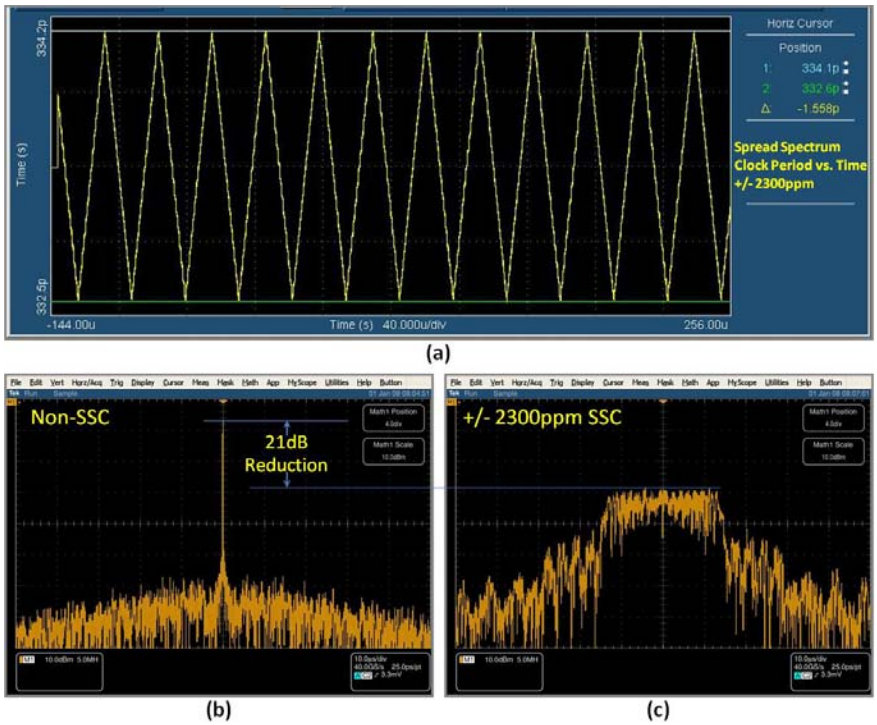


Figure 5.5 Measured jitter performance at 3 GHz (a) non-SSC (b) center spread +/- 2300ppm SSC.

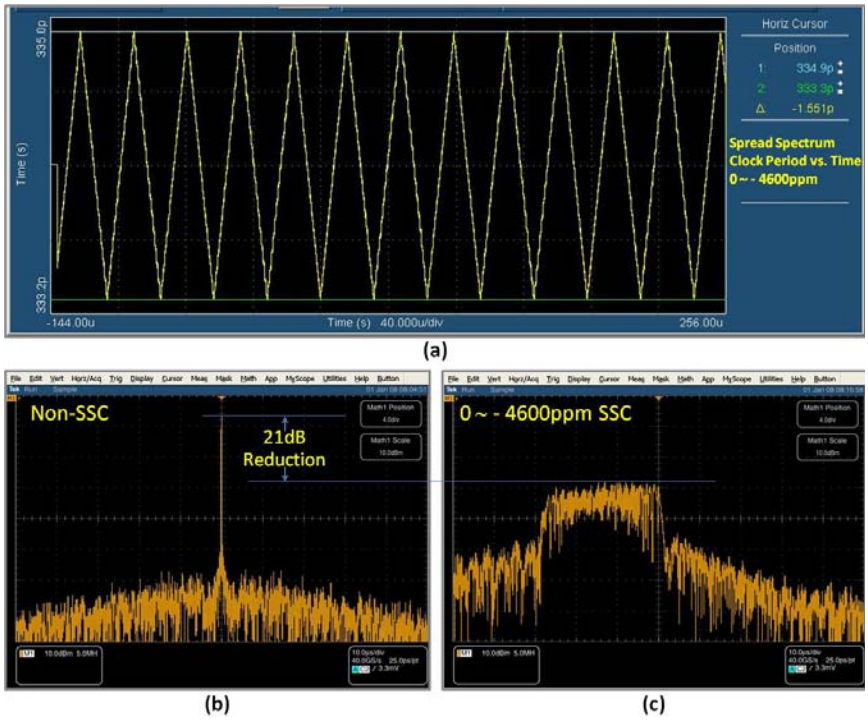


Figure 5.6 Measured 3 GHz (a) clock period vs. time of down spread 0 ~ -4600 ppm SSC (b) power spectrum density plot of non-SSC (c) power spectrum density plot of down spread 0 ~ -4600 ppm SSC.

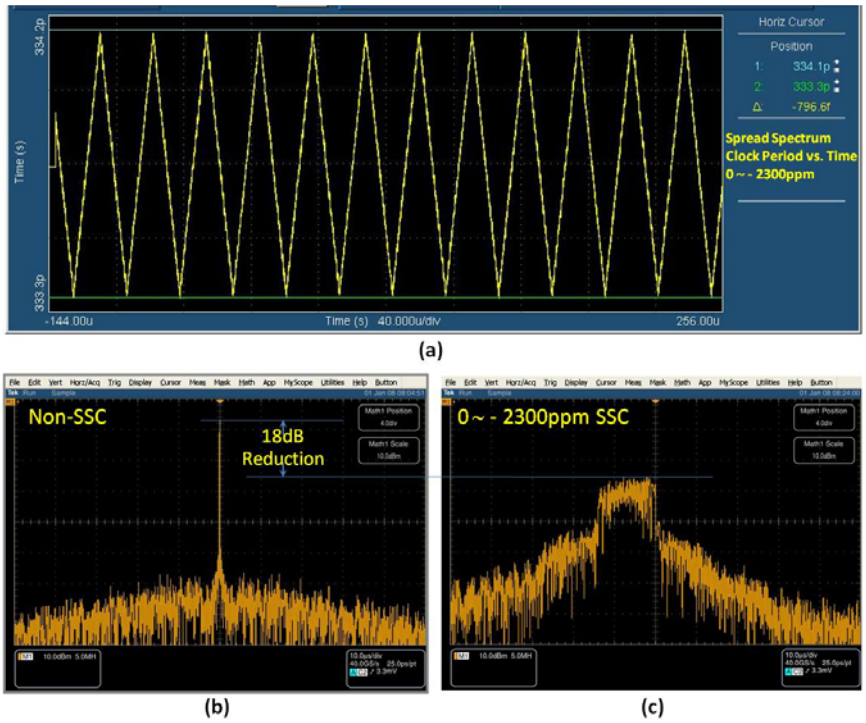


Figure 5.7 Measured 3GHz (a) clock period vs. time of down spread 0 ~ -2300 ppm SSC (b) power spectrum density plot of non-SSC (c) power spectrum density plot of down spread 0 ~ -2300 ppm SSC.

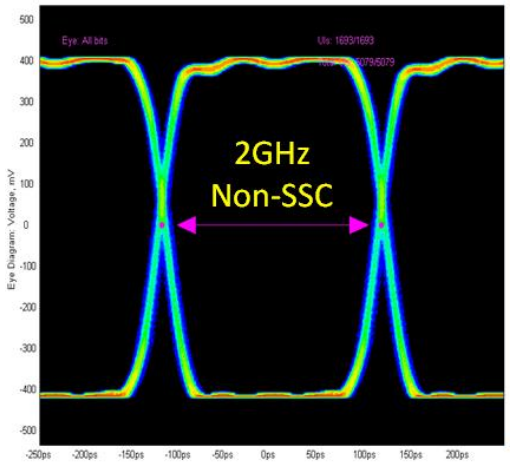
Figure 5.5 shows the center spread +/-2300 ppm SSC clock period vs. time and the power spectrum density plots for both non-SSC and SSC modes. The clock period vs. time shows a smooth triangle frequency center spreading to +/-2300 ppm without any cycle-to-cycle jitter caused by the PI. Both DNL and INL effects from the PI are minimized or filtered out by the PLL loop filter. The 0 ~ -4600 ppm and 0 ~ -2300 ppm down spread SSC clock period vs. time and the power spectrum density plots are shown in Figure 5.6 and Figure 5.7. The measured power spectrum density plots are well matched with the simulated data of Figure 5.3, which indicates a 21dB EMI reduction between non-SSC and center spread +/-2300 ppm SSC. The measured EMI reductions for down spread 0 ~ -2300 ppm and 0 ~ -4600 ppm are 18 dB and 21 dB, respectively. These are also matched with the simulated results shown in Figure 5.3. The clock eye

diagrams with non-SSC and jitter performance with non-SSC and center spread +/-2300 ppm SSC at 2, 3 and 4.25 GHz are shown in Figure 5.8, which indicates a jitter increase going from 2 to 4.25 GHz but with no significant difference between the non-SSC and SSC modes across the wide frequency range. The random jitter at 3 GHz for SATA and SAS applications is < 1.3 pS and < 1.5 pS for non-SSC and SSC modes, respectively.

Figure 5.9 shows the designed 375×320μm^2 PLL based SSC clock generator layout with the performance summary. The design also includes a lock detector, DCC, and loss of reference clock detector to support a valid output clock indication, output clock duty-cycle / quadrature phase correction, and presence of reference clock indication, respectively. The designed SSC clock generator supports the frequency range 2 ~ 4.25 GHz. The output clock divider with available frequency dividing ratios of 1, 2, 4, and 8 extends the available output frequency range to 0.25~4.25 GHz. The loop filter capacitor is completely shielded with a ground layer metal to avoid any possible noise caused by adjacent signals on the top.

Figure 5.10 shows the micrograph and layout plot of the test chip fabricated in a 90 nm CMOS process. There are a total 4 identical PLLs, with 2 PLLs each on the left and right sides. Each of the 4 pairs of TX-RX is able to access each clock of 4 PLLs, which enables multi-rate, non-SSC and SSC operation co-existing in a multi-channel SerDes. The two PLLs on each side can be driven by either the same or separate reference clock inputs. If both PLLs have the same reference clock with one of the PLLs having the SSC at a fixed phase shift rate by stalling at specific counter2 bit in Figure 5.2, it will also provide an on-chip frequency offset test for the TX-RX. Table 5.1 gives a summary of the jitter performance and power dissipation for the SSC clock generator with non-SSC

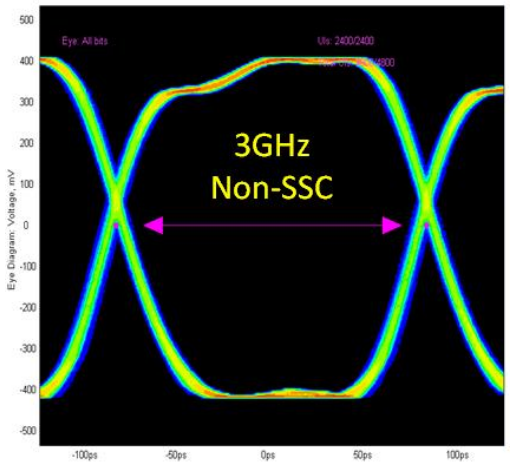
and SSC operation at 2, 3 and 4 GHz.



2GHz, Non-SSC	Current Acq	Averaged
Random (RMS)	1.9729ps	1.9172ps
Deterministic (Pk-Pk)	12.327ps	11.622ps
Periodic	9.1260ps	8.4652ps
Duty Cycle	3.2009ps	3.1572ps
Data Dependent (ISI)	0.00s	0.00s
Total @ BER (Pk-Pk)	35.589ps	34.326ps
Eye Opening @ BER	825.78mui	831.97mui

2GHz, SSC+/-2300ppm	Current Acq	Averaged
Random (RMS)	2.1522ps	2.1896ps
Deterministic (Pk-Pk)	11.736ps	17.304ps
Periodic	8.0768ps	13.695ps
Duty Cycle	3.6594ps	3.6081ps
Data Dependent (ISI)	0.00s	0.00s
Total @ BER (Pk-Pk)	37.657ps	43.493ps
Eye Opening @ BER	815.68mui	786.78mui

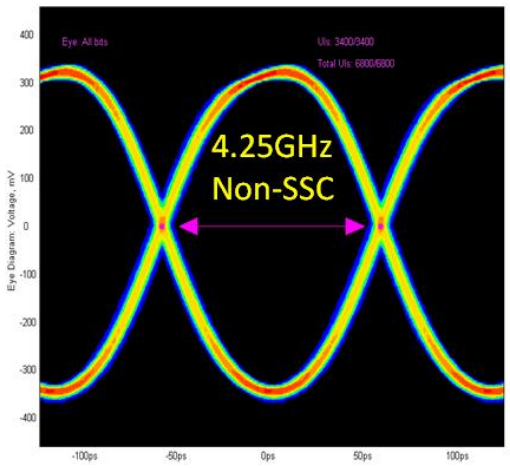
(a)



3GHz, Non-SSC	Current Acq	Averaged
Random (RMS)	1.2431ps	1.2550ps
Deterministic (Pk-Pk)	9.2851ps	9.6577ps
Periodic	6.5719ps	6.8694ps
Duty Cycle	2.7132ps	2.7883ps
Data Dependent (ISI)	0.00s	0.00s
Total @ BER (Pk-Pk)	24.728ps	25.025ps
Eye Opening @ BER	895.38mui	894.12mui

3GHz, SSC+/-2300ppm	Current Acq	Averaged
Random (RMS)	1.4395ps	1.4559ps
Deterministic (Pk-Pk)	13.650ps	15.894ps
Periodic	7.0790ps	9.2456ps
Duty Cycle	6.5706ps	6.6481ps
Data Dependent (ISI)	0.00s	0.00s
Total @ BER (Pk-Pk)	31.223ps	33.638ps
Eye Opening @ BER	812.88mui	798.20mui

(b)



4.25GHz, Non-SSC	Current Acq	Averaged
Random (RMS)	976.70fs	1.0112ps
Deterministic (Pk-Pk)	10.330ps	10.812ps
Periodic	5.9373ps	6.5289ps
Duty Cycle	4.3928ps	4.2836ps
Data Dependent (ISI)	0.00s	0.00s
Total @ BER (Pk-Pk)	22.196ps	23.052ps
Eye Opening @ BER	811.33mui	804.06mui

4.25GHz, SSC+/-2300ppm	Current Acq	Averaged
Random (RMS)	1.1093ps	1.1128ps
Deterministic (Pk-Pk)	12.986ps	12.515ps
Periodic	9.0369ps	8.4097ps
Duty Cycle	3.9493ps	4.1052ps
Data Dependent (ISI)	0.00s	0.00s
Total @ BER (Pk-Pk)	26.214ps	25.840ps
Eye Opening @ BER	777.60mui	780.47mui

(c)

Figure 5.8 Measured clock eye diagram and non-SSC / SSC jitter performance at (a) 2 GHz (b) 3 GHz, and (c) 4.25 GHz.

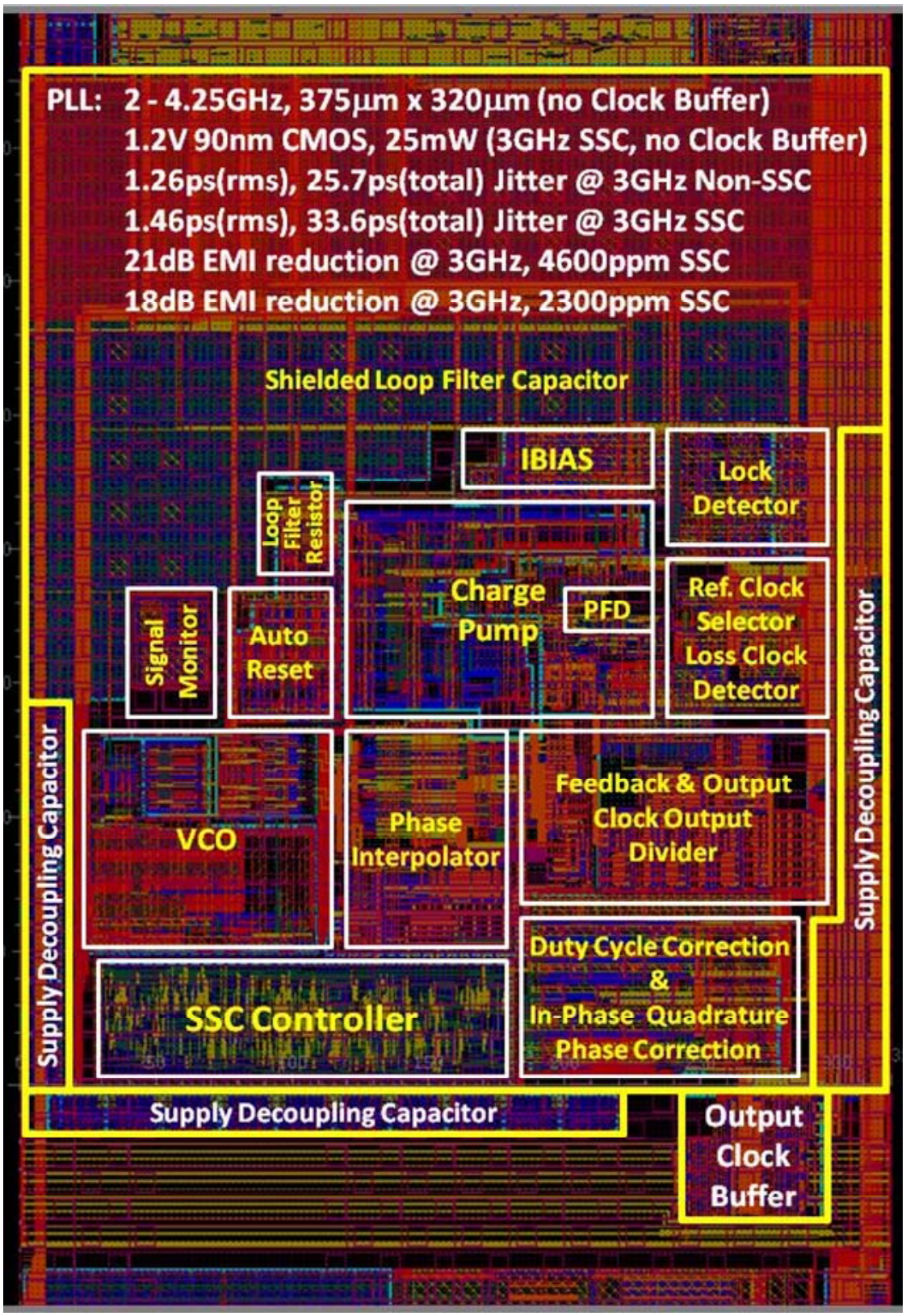


Figure 5.9 PLL based SSC clock generator layout plot with performance summary.

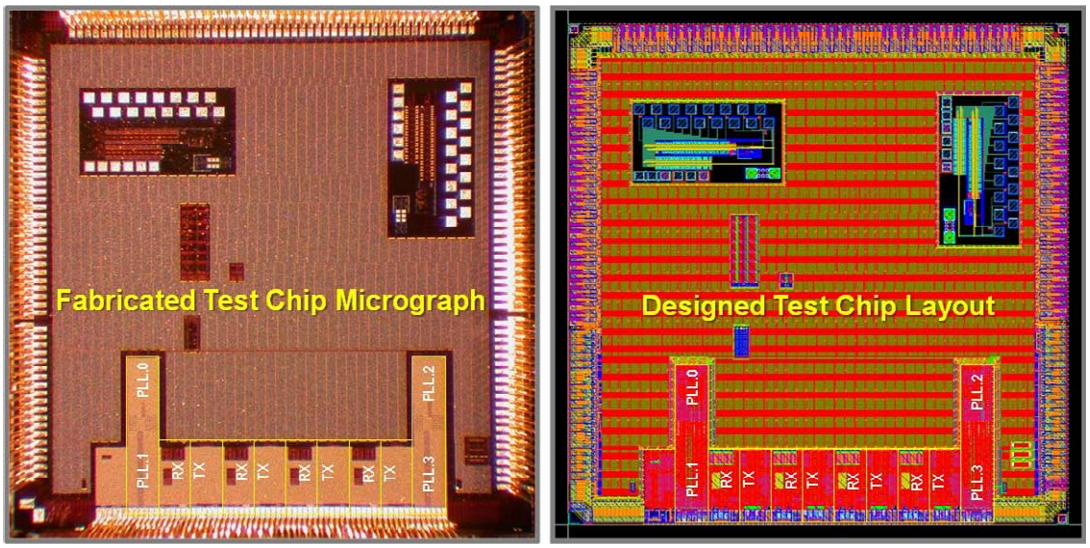


Figure 5.10 Fabricated test chip micrograph and layout plot in 90nm CMOS process.

Table 5.1 Summary of jitter performance and power dissipation.

Frequency (GHz)	Non-SSC			SSC		
	Random Jitter (pS.rms)	Total Jitter (pS)	Total Power (mW)	Random Jitter (pS.rms)	Total Jitter (pS)	Total Power (mW)
2	1.92	34.3	14	2.19	43.5	16
3	1.26	25.7	19	1.46	33.6	23
4.25	1.01	23.1	25	1.11	25.8	29

5.4 CONCLUSIONS

The main advantages of this design compared to references [3-6] are its easy programmability, high EMI reduction and multiple up/center/down frequency spreading SSC clocks. On the other hand, it does not offer both non-SSC and SSC clocks at the same time from the same PLL.

REFERENCES

[1] Serial ATA Workgroup, "Serial ATA," Revision 2.6, May 2007

[2] INCITS, "Serial Attached SCSI – 2 (SAS-2)," Revision 14a, May 2008

[3] M. Aoyama et al., "3Gbps, 5000ppm Spread Spectrum SerDes PHY with frequency tracking Phase Interpolator for Serial ATA," *Dig. Sym. VLSI Circuits*, pp.107-110, Jun. 2003

[4] M. Sugawara et al., "1.5 Gbps, 5150 ppm SPREAD Spectrum SerDes PHY with a 0.3mW, 1.5 Gbps Level Detector for Serial ATA," *Dig. Sym. VLSI Circuits*, pp. 60-63, Jun. 2002

[5] T. Yoshikawa et al., "An Over-1-Gb/s Transceiver Core for Integration Into Large System-on-Chips for Consumer Electronics," *IEEE Transactions on VLSI*, vol.16, no.9, pp.1187-1198, Sept. 2008.

[6] M. Kokubo et al., "Spread-Spectrum Clock Generator for Serial ATA using Fractional PLL Controlled by ΔΣ Modulators with Level Shift," *ISSCC Dig. Tech. Papers*, pp.160-161, Feb. 2005.

[7] H.-R. Lee et al., "A Low Jitter 5000ppm Spread Spectrum Clock Generator for Multi-channel SATA Transceiver in 0.18μm CMOS," *ISSCC Dig. Tech. Papers*, pp.162-163, Feb. 2005

[8] D.-S. Shen and S.-I. Liu, "A Low-Jitter Spread Spectrum Clock Generator Using FDMP," *IEEE Trans. on Circuits and Systems II: Express Briefs*, Vol. 54, pp. 979 – 983, Nov., 2007.

[9] J. Zheng et al, "A Programmable Spread Spectrum Clock Generation," *Proceedings, 7th International Conf. on ASIC*, pp. 337 – 339, 2007.

[10] R. Saraswat et al, "Programmable Spread Spectrum Clock Generation Based on Successive Phase Selection Technique," *Proceedings, IEEE International Symposium on Circuits and Systems*, pp. 2845 – 2848, 2008.

[11] Ming-ta Hsieh and Gerald E. Sobelman, "Clock and Data Recovery with Adaptive Loop Gain for Spread Spectrum SerDes Applications," *Proceedings, IEEE International Symposium on Circuits and Systems,* pp. 4883-4886, 2005

CHAPTER 6

6 SIMULTANEOUS BI-DIRECTIONAL PAM-4 LINK WITH BUILT-IN SELF-TEST

This chapter presents a new design of a simultaneous bidirectional PAM-4 wired transmission system that uses built-in self-test (BIST) to adjust the level of pre-emphahsis that is applied. The BIST circuitry consists of a pattern generator and detector, a signal comparator and a high-pass filter. It outputs an error indicator that is used as a control signal in the adaptive pre-emphasis block. The feedback loop inherent in a simultaneous bidirectional link provides a natural opportunity to carry information about the channel characteristics without the need for an extra dedicated wire. The design has been verified using the Cadence SpectreRF and Verilog-A simulators and the channel loss characteristics are based on an FR-4 material model extracted from the Cadence Transmission Line Model Generator.

6.1 INTRODUCTION

The limited I/O on a chip and the loss characteristics of transmission channels have proven to be major challenges to overall system performance [1]. For low cost serial data communication application such as Serial ATA [2], the need to obtain higher total I/O throughput and channel bandwidth without increasing the system clock rate have led several researchers to study simultaneous bidirectional signaling [3-5] and multilevel signaling [6-8], respectively. For a given total I/O throughput and transmission channel bandwidth, both the simultaneous bidirectional transmission and multilevel signaling

provide a higher timing margin but a lower voltage margin [9-10]. Therefore, these systems are most suitable for low loss or short link communications applications. There has also been recent industry interest in the possible combination of multilevel signaling and simultaneous bidirectional transmission for extending the bandwidth of today's high-speed serial backplanes [11]. The effective throughput of simultaneous bidirectional transmission with N-level signaling is $2 \cdot \log_2(N)$ times that of a traditional unidirectional binary transmission system for a given pin-count and data symbol rate. For simplicity and robustness, in this investigation we have chosen to work with 4-level pulse amplitude modulated (PAM-4) bidirectional signaling.

Pre-emphasis is a signal booster for energy in the high frequency portion of the spectrum and is often used to compensate the channel loss in the gigahertz range where the attenuation increases rapidly with the increased data rate [12]. The advantages of using transmitter pre-emphasis compared to receiver equalization are lower power consumption, superior performance, and interoperability [13]. However, pre-emphasis is not always the preferred option. For example, receiver equalization may give superior performance if crosstalk is one of the major factors causing the data degradation [12]. One disadvantage normally associated with adaptive pre-emphasis is the need for an extra dedicated wire to carry the control signal from the receiver back to the transmitter. However, the feedback loop inherent in a bidirectional link provides a natural opportunity for incorporating adaptive pre-emphasis into the design without the need for any additional line.

6.2 ARCHITECTURE

The basic circuit diagram and illustration of the input/output signals of simultaneous

bidirectional PAM-4 transmission are shown in Figure 6.1 and Figure 6.2, respectively. Replica-Diver.L and Replica-Driver.R have the same characteristics as Driver.L and Driver.R, respectively. Therefore, echo cancellation can completely remove its own side driver outgoing signal and extract the received signal, V_{EL} or V_{ER}. T_O represents the delay time for the wave traveling through the transmission channel. A 7-level signal is observed on the channel even though it is only a 4-level signal transmission system, which is due to the superposition of signals originating at each end of the line.

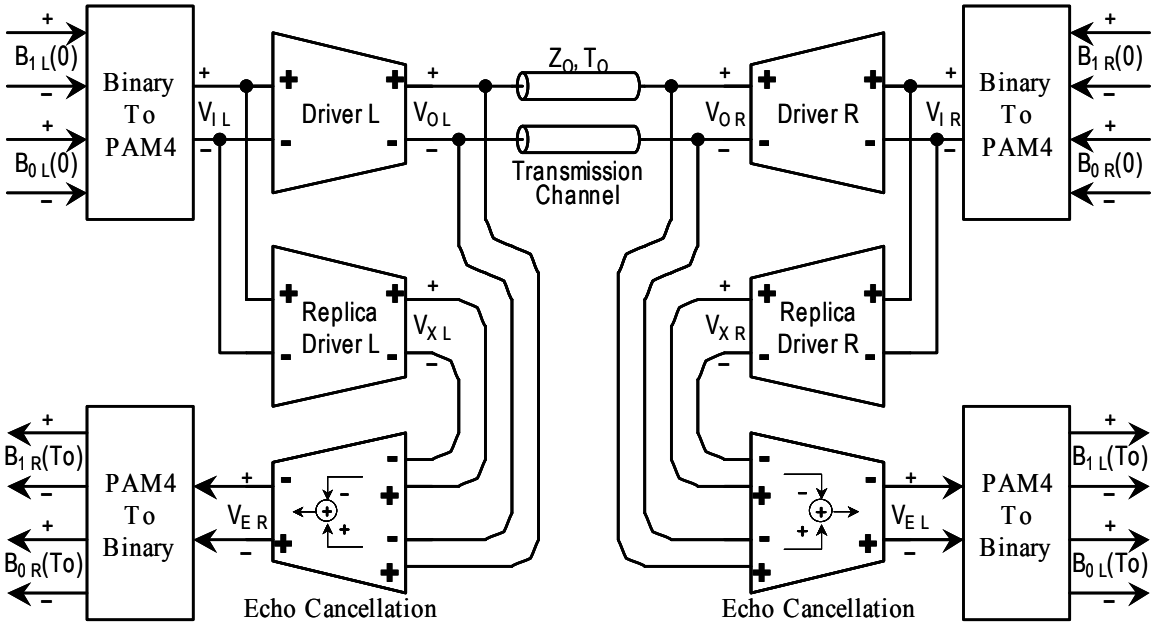


Figure 6.1 The block diagram of a simultaneous bidirectional PAM-4 wired link.

Figure 6.3 shows the basic circuit diagram for our proposed simultaneous bidirectional multilevel signaling system with adaptive channel pre-emphasis. For simplicity and readability, the figure has been drawn in a single-ended mode. Our design approach is to first determine the degree of degradation on the received signal through a Built-In Self-Test (BIST) circuit.

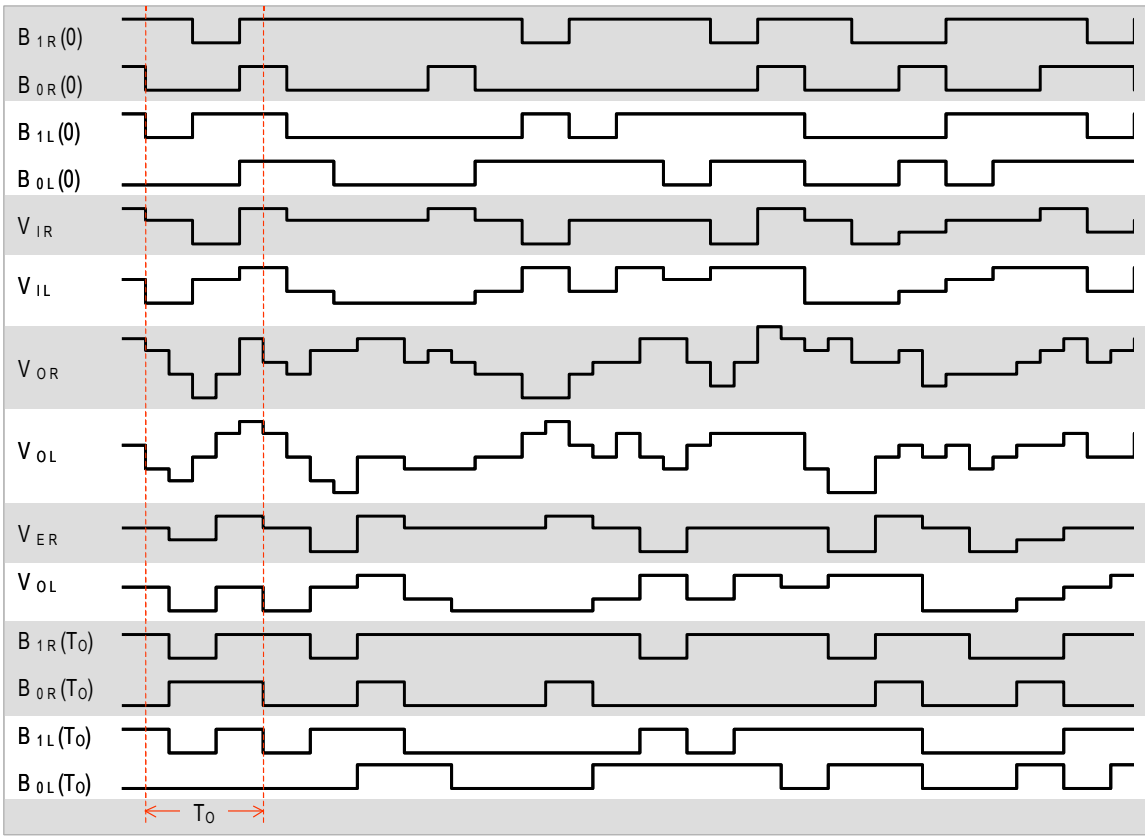


Figure 6.2 Input/output signals illustrating the operation of a PAM-4 simultaneous bidirectional signaling system.

Figure 6.4 shows the conceptual circuit diagram for implementing the proposed adaptive channel pre-emphasis. At the transmitter side, the BIST generates a Pseudo Random Binary Sequence (PRBS). A known pattern or comma character is used for pattern sequence detection. On the receiver side, the BIST unit consists of a pattern detector, a PRBS generator, a signal comparator, and an adaptive pre-emphasis control signal generator. The pattern detector first detects the pattern sequence and then aligns its own PRBS signal with the received signal. The signal comparator is an exclusive OR function, which compares the aligned received and PRBS signals bit by bit and outputs a bit error signal if they are not the same. Then, a high-pass filter controller averages the

error bit indicator signals and generates a moderately low frequency (e.g. 1/10 the normal data rate) pulse width modulated (PWM) signal, V_{PWL} and V_{PWR} as shown in Figure 6.3, which gets fed back to the transmitting side through the existing simultaneous bidirectional link. The moderately low frequency provides minimum signal loss on the transmission channel. The received PWM signal is averaged through a low pass filter and is used to tune the level of the pre-emphasis, producing the signals V_{PEL} and V_{PER} of Figure 6.3. The transmitter pre-emphasis design is based on a current-mode logic (CML) driver with an adjustable 2-tap pre-emphasis. For our design, the transmitter driver pre-emphasis has been implemented with a constant current difference between the driver and pre-emphasis paths so that the amplitude of the signal eye opening is kept constant. The novelty of our approach is that we make use of the inherent bidirectional nature of the link to carry the control information without any additional wire, as well as the fact that the pre-emphasis is applied to signals propagating in both directions.

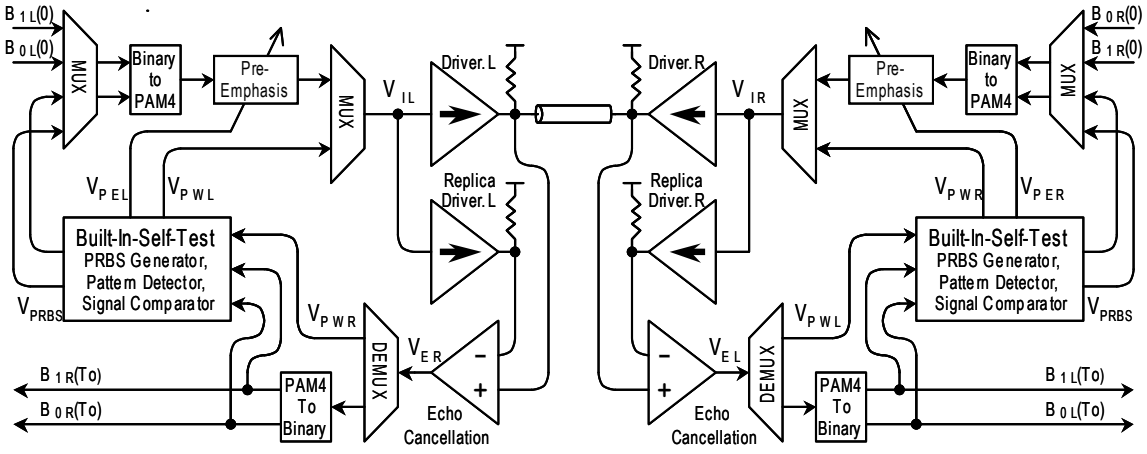


Figure 6.3 The simultaneous bidirectional signaling system with a BIST-based adaptive channel pre-emphasis feature.

The design of the echo cancellation circuit is shown in Figure 6.5, which is based on the adaptive equalizer of Reference [14]. However, a high-pass filter is used instead of a

low-pass filter before the variable gain amplifier. The justification for this is that Driver.L and Driver.R always drive I/O and the off-chip transport channel and thus they are more bandwidth limited than Replica-Driver.L and Replica-Driver.R of Figure 6.1. The area within the circled dashed line represents the adjustable high-pass filter section.

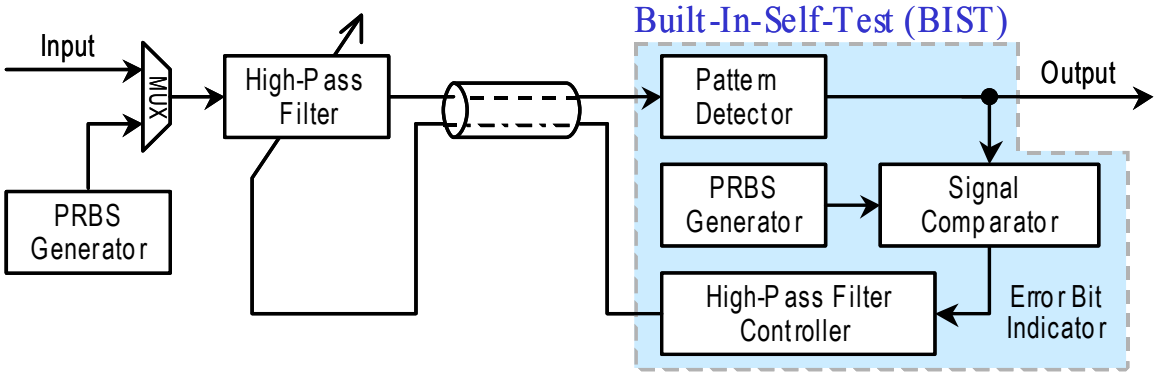


Figure 6.4 A conceptual circuit diagram for the implementation of the BIST-based adaptive channel pre-emphasis.

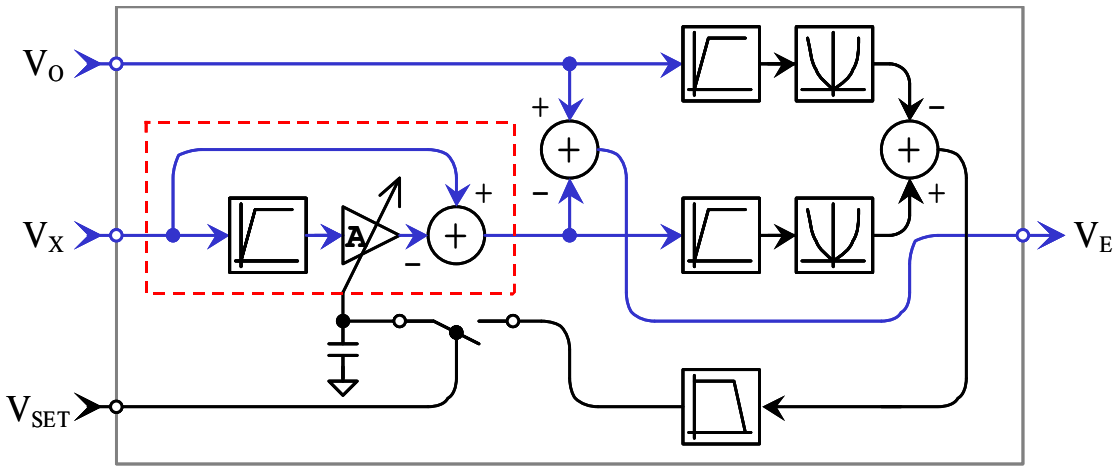


Figure 6.5 The basic circuit diagram for echo cancellation.

6.3 SIMULATION RESULTS

The proposed design has been verified with Cadence SpectreRF and Verilog-A simulators. The transmission channel loss characteristic is based on an FR-4 material,

which is the standard glass epoxy substrate. The transmission channel trace model is extracted from the Cadence Transmission Line Model Generator and is based on Reference [15] with approximately 1 dB of loss per inch at 10 GHz.

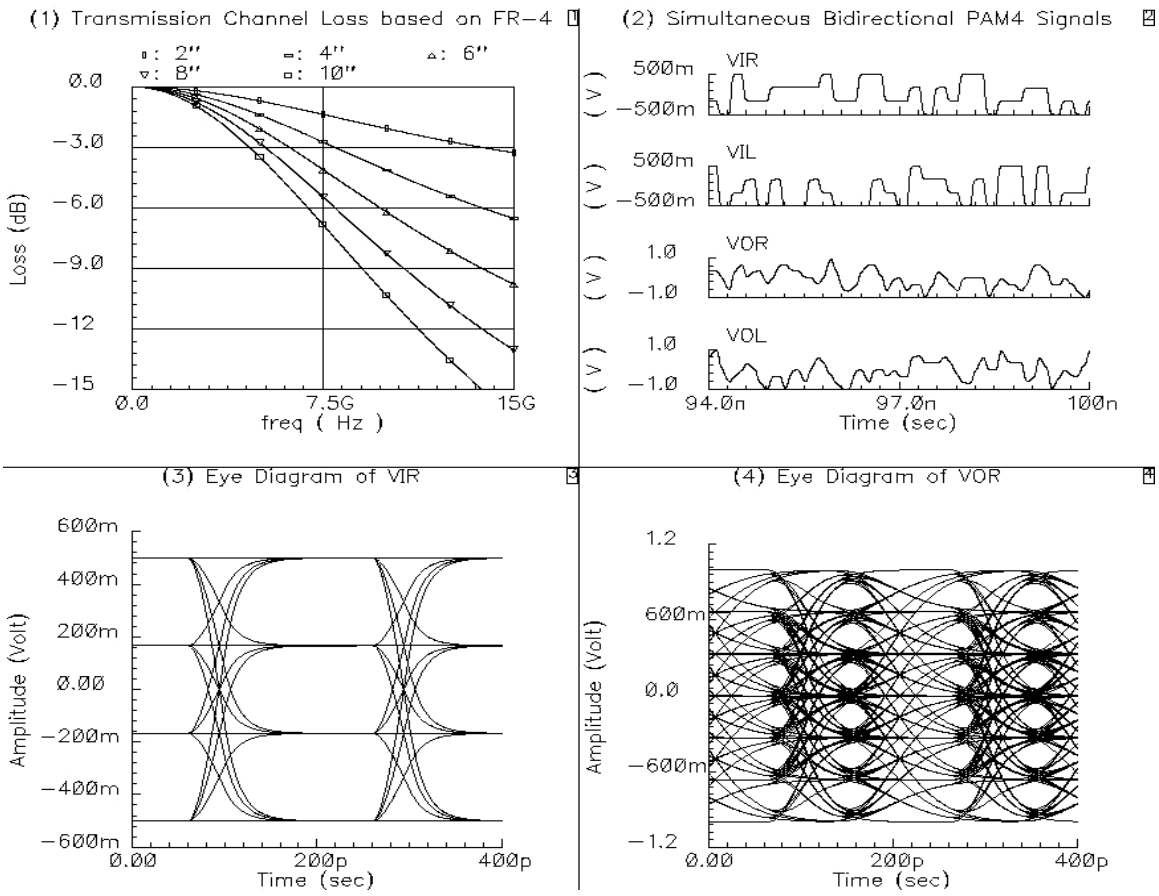


Figure 6.6 The simultaneous bidirectional PAM-4 transmission simulation results for the circuit of Figure 6.1.

Figure 6.6, part (1) shows simulation results for the signal loss in the channel versus the signal frequency at 2", 4", 6", 8" and 10" lengths. Figure 6.6, parts (2)~(4) show the simultaneous bidirectional PAM-4 signaling transient simulation results at 5 Gbps for the circuit of Figure 6.1. Input and output signal eye-diagrams of the driver show how the 7-level characteristic on the transport channel arises due to the superposition from each end. The transmission channel length used for the transient simulation in Figure 6.6 is 10

inches and the driver input signals, VIR and VIL, are individual PAM-4 signals. Each of the PAM-4 signals is produced by a different pseudo-random-binary-sequence (PRBS) generator of length $2^7 - 1$ bits.

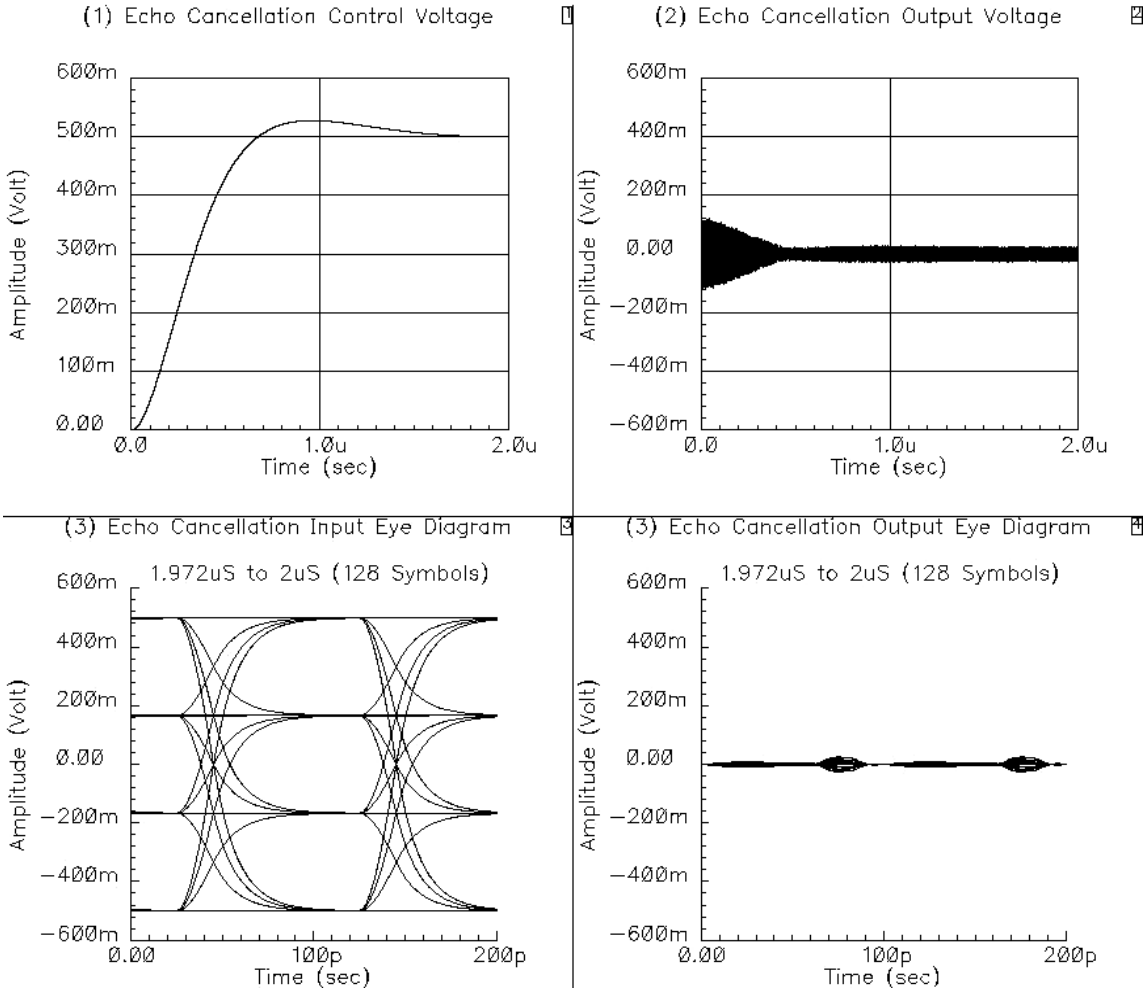


Figure 6.7 The echo cancellation initialization setup simulation results for the circuits of Figure 6.1 and Figure 6.5

Figure 6.7 shows the echo cancellation initialization setup simulation results for the circuits of Figure 6.1 and Figure 6.5. The damping factor is set to be close to 0.707 for the tunable low-pass filter in echo cancellation so that the initialization time is optimized without stability concerns. The required initialization time is less than 2 uS after reaching a steady-state mode, as shown in Figure 6.7, part (1). The output of echo

cancellation, as shown in Figure 6.7, parts (2) and (4), should be as small as possible in order to minimize the received signal degradation.

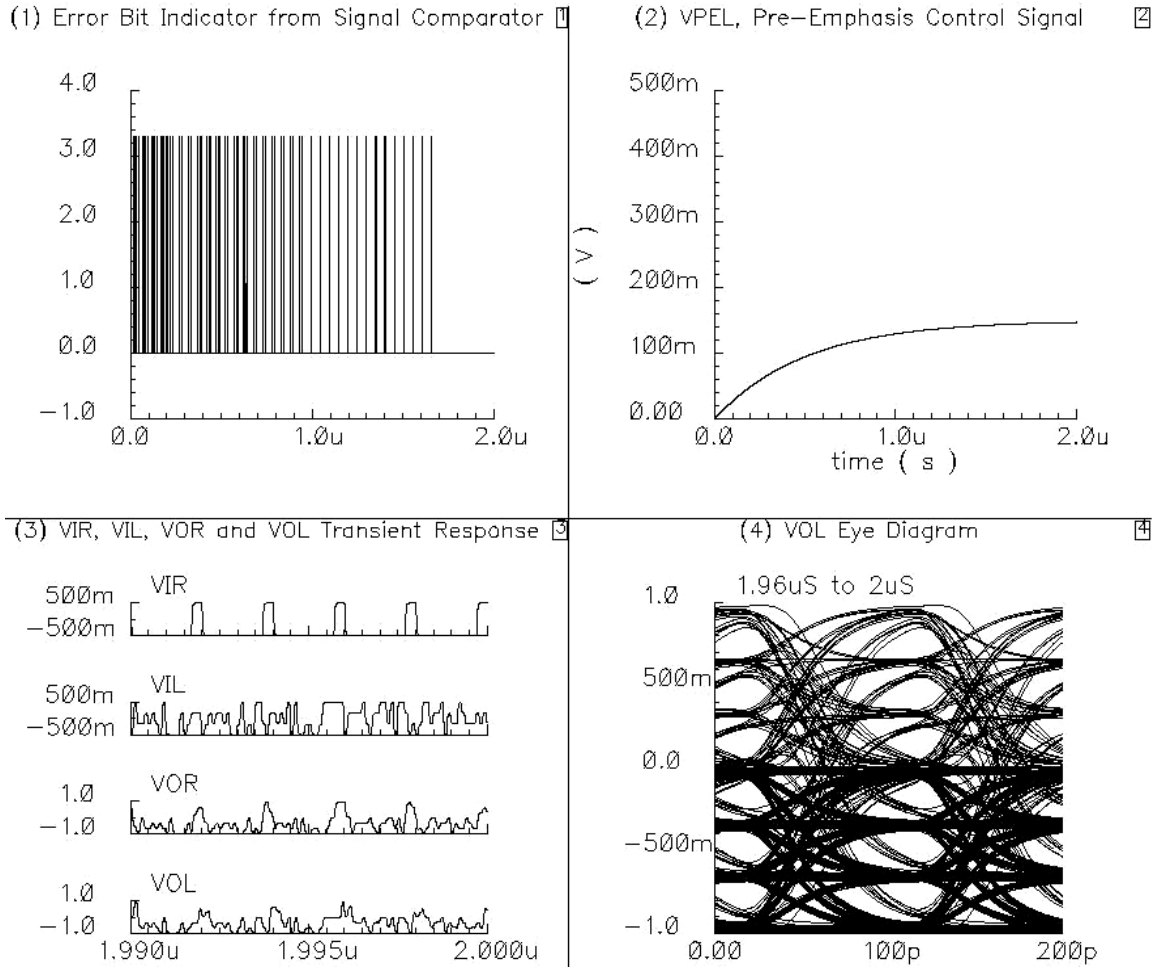


Figure 6.8 The adaptive channel pre-emphasis initialization setup simulation results for the circuit of Figure 6.3.

Figure 6.8 shows the adaptive channel pre-emphasis initialization setup simulation results for the circuit of Figure 6.3. The required initialization time is also less than 2 uS in order to reach a steady-state mode, as shown in Figure 6.8, parts (1) and (2). The transient signal amplitude at the data driver output increases as the pre-emphasis voltage increases, which maintains the same difference between driver current and pre-emphasis current so that the amplitude of the eye opening is kept constant, as shown in the other

parts of the figure.

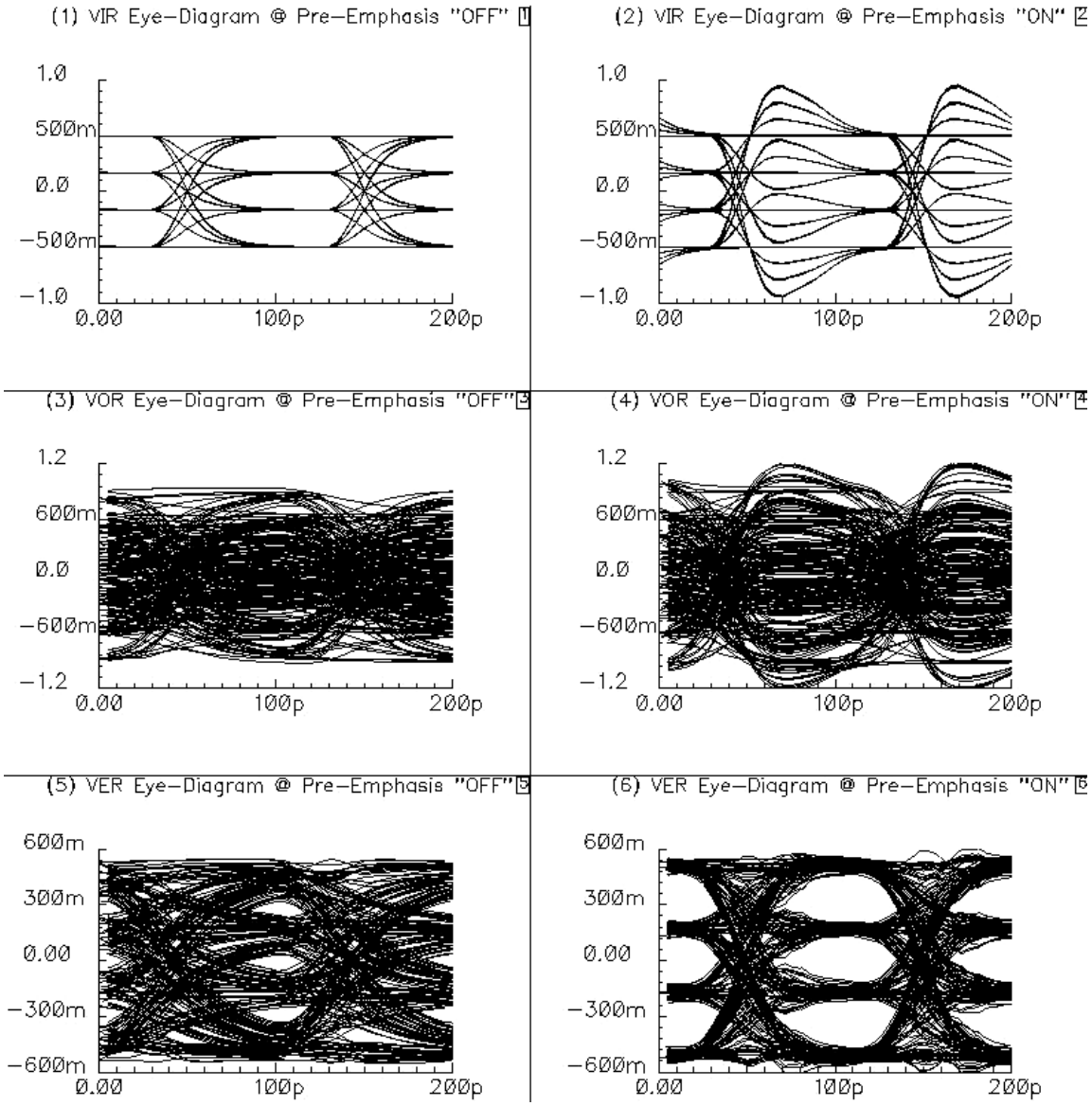


Figure 6.9 Simulation results for simultaneous bidirectional PAM-4 transmission with and without adaptive channel pre-emphasis after echo cancellation and pre-emphasis have been initialized.

Figure 6.9 shows the simulation results for simultaneous bidirectional PAM-4 signaling with and without an adaptive channel pre-emphasis after echo cancellation and pre-emphasis have been initialized. It shows that the adaptive channel pre-emphasis helps to open the eye in the VER eye diagram, which improves the rise and fall times of the

received signal.

6.4 CONCLUSIONS

This chapter presents a new design for a differential current-mode simultaneous bidirectional PAM-4 transmission system that uses BIST-based adaptive channel pre-emphasis. The BIST module compares the received signal against the known pattern sequence. The result of the comparison is an error bit indicator signal, which is fed back to the transmitter by utilizing the inherent feedback loop of the simultaneous bidirectional channel. In this way, it automatically adjusts the pre-emphasis to compensate for the actual channel loss characteristics. The proposed design has been verified using Cadence SpetreRF and Verilog-A simulators. The simulations show that the applied adaptive pre-emphasis greatly improves the quality of the link and therefore reduces the efforts needed to reliably recover the data at each receiver side.

REFERENCES

[1] Genesys Logic America, Inc., "Multi-Gigabit SerDes: The Cornerstone of High Speed Serial Interconnects," *Multi-Gigabit SerDes White Paper,* Gensys Logic America, Inc., Sept. 15, 2003.

[2] The Serial ATA Working Group, "SATA 1.0a Specification," The Serial ATA Working Group, http://www.serialata.org, Jan. 7, 2003.

[3] R. Drost, "Architecture and Design of a Simultaneously Bidirectional Single-ended High Speed Chip-to-Chip Interface," *Ph.D. Thesis, Stanford University, Palo Alto, CA,* Nov. 2001.

[4] D.N. de Araujo, M. Cases, and N. Pham, "Design Optimization Methodology for Simultaneous Bidirectional Interface," *IEEE Electrical Performance of Electronic Packaging*, pp. 295-298, Oct. 2001.

[5] K.S. Canagasaby, S. Rajagopalan, and S. Dabral, "Interconnect Design Challenges in Source Synchronous Simultaneous Bidirectional Links," *IEEE Elect. Perf. of Electronic Packaging*, pp. 11-14, Oct. 2002.

[6] R. Farjad-Rad et al, "A 0.3-μm CMOS 8-Gb/s 4-PAM Serial Link Transceiver" IEEE Jour. Solid-State Circuits, Vol.35, No.5, pp.757-764, May 2000.

[7] J. Sonntag et al. "An adaptive PAM-4 5 Gb/s backplane transceiver in 0.25μm CMOS," *IEEE Custom Integrated Circuits Conference*, pp.363-366, 2002.

[8] D. J.. Foley and M. P. Flynn "A low-power 8-PAM serial transceiver in 0.5- μm digital CMOS," *IEEE Journal of Solid-State Circuits*, Vol.37, No.3, pp.310-316, March 2002.

[9] D.N. de Araujo, M. Cases, N. Pham, and D. Dreps, "Unidirectional vs. Simultaneous Bidirectional Source Synchronous Signaling," *IEEE Electrical Perf. of Electronic Packaging*, pp. 7-10, Oct. 2002.

[10] M. Fukaishi, "GHz serial link transceiver using multiple-valued data representation," *IEEE 11th International Workshop on Post Binary ULSI*, May 2002

[11] L. Harrison and H. Takatori, "Extending the Life of Today's Backplanes through Simultaneous Bidirectional Transmission," *DesignCon* 2004

[12] J. Zhang and Z. Wong, "White Paper on Transmit Pre-emphasis and Receive Equalization," Mindspeed Technologies – A Conexant Bus., Oct. 2002.

[13] K. Dalmia, “High Performance Backplane Design Using the Marvell AlaskaTM X Quad 3.125 Gb/s SERDES,” Marvell white paper, Nov. 2001.

[14] J.N. Babanezhad, “A 3.3 V analog Adaptive Line-Equalizer for Fast Ethernet Data Communication,” *IEEE Custom Integrated Circuits Conference*, pp. 343-346, May 1998.

[15] R. Kollipara, G Yeh, B. Chia, and M. Agarwal, “Design, Modeling and Characterization of High Speed Backplane Interconnect,” *DesignCon* 2003

CHAPTER 7

7 ARCHITECTURES FOR MULTI-GIGABIT WIRE-LINKED CLOCK AND DATA RECOVERY

Clock and data recovery (CDR) architectures used in high-speed wire-linked communication receivers are often shown as PLL or DLL based topologies. However, there are many other types of CDR architectures such as phase-interpolator, oversampling and injection locked based topologies. The best choice for the CDR topology will depend on the application and the specification requirements. This chapter presents an overview and comparative study of the most commonly used CDR architectures. This analysis includes the circuit structures, design challenges, major performance limitations, and primary applications. Finally, the tradeoffs among the various CDR architectures are summarized.

7.1 INTRODUCTION

Data bandwidth for state of the art wire-linked communication systems is growing at an extremely fast rate. In 2007, the International Technology Roadmap for Semiconductors (ITRS) predicted that the non-return to zero (NRZ) data rate for high-performance differential pair point-to-point nets on a package would reach 100 gigabits per second (Gbps) by the year 2019, as shown in Figure 7.1 [1]. The data in such high-speed wire-linked communication systems often become severely distorted by both external and internal noise during transmission, which leads to jitter and skew in the received data.

A clock and data recovery (CDR) circuit is an essential block in many high-speed wire-

linked data transmission applications such as optical communications systems, backplane data-link routing and chip-to-chip interconnection. The important role of a CDR is to extract the transmitted data sequence from the distorted received signal and to recover the associated clock timing information. Figure 7.2 illustrates a simplified functional diagram of clock recovery and data retiming using a CDR circuit. The clock recovery circuit detects the transitions in the received data and generates a periodic clock. The decision circuit often uses D-type Flip-Flops (DFFs) driven by the recovered clock to retime the received data, which samples noisy data and then regenerates it with less jitter and skew [2].

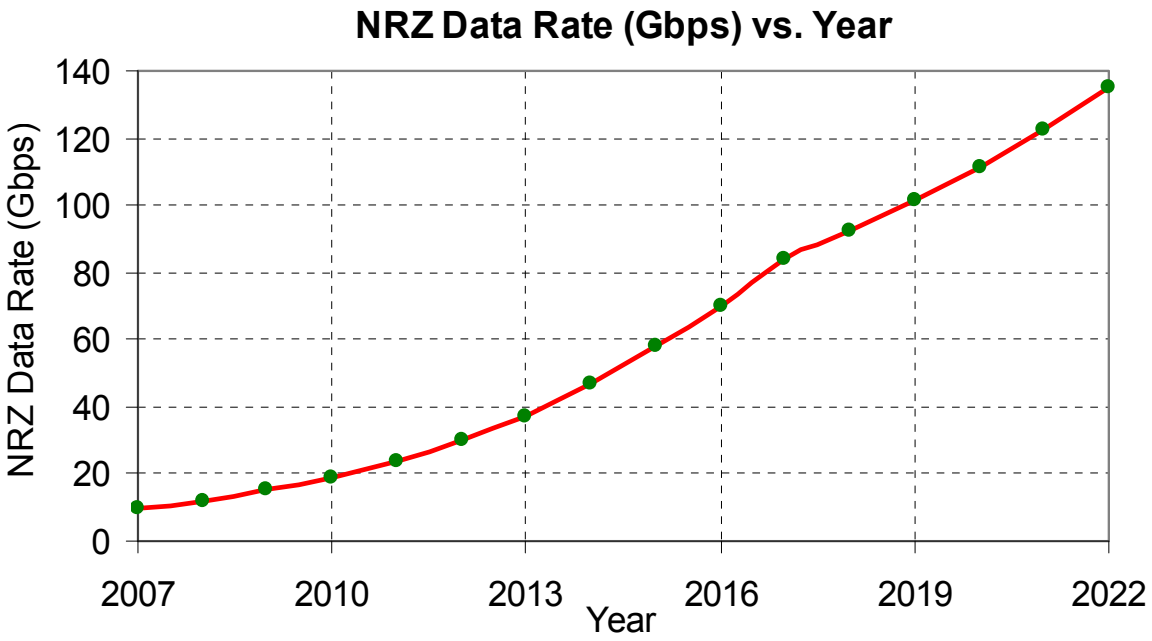


Figure 7.1 NRZ data rate for high performance differential pair point-to-point nets on a package, based on the ITRS 2007 roadmap predication.

A generic block diagram of a high-speed wire-linked data transmission system is shown in Figure 7.3, where the received data is equalized in the receiver input buffer and retimed in the CDR module before proceeding into the deserializer module. A source-

asynchronous system is shown, in which the transmitting and receiving sides use different clock sources. This results in a possible frequency offset between the transmitted data and the local clock on the receiver side due to natural device mismatches, creating additional challenges for the CDR circuit. Most wire-linked communication systems fall into this category. In contrast to this, data transmission systems such as chip-to-chip interconnect in which both the transmitter and receiver use the same clock source are known as source-synchronous systems. A CDR for this type of system only needs to provide a finite phase capturing range.

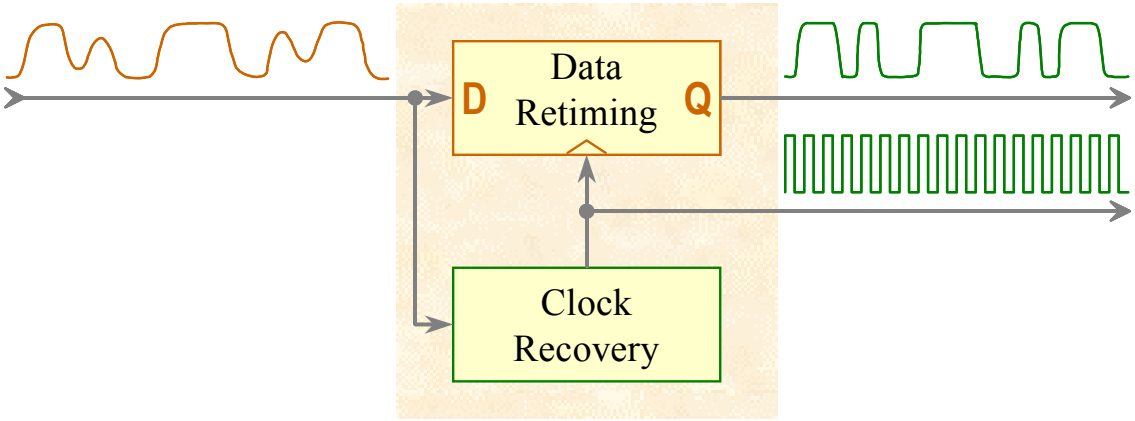


Figure 7.2 Clock recovery and data retiming for a CDR circuit.

The clock synthesizer in Figure 7.3 may also drive multiple transmitters and receivers (TXs / RXs), which is known as a multi-channel configuration. Having multiple TXs / RXs use the same clock synthesizer reduces the area and power overhead.

Many researchers have proposed a wide variety of CDR designs for high-speed wire-linked data transmission applications, such as those based on an analog phase locked loop (APLL) [3-6], a digital phase locked loop (DPLL) [7-8], a delay locked loop (DLL) [9-10], a phase interpolator [11-13], injection locking [14-15], oversampling [16-19], a

gated oscillator [20-22], and a high-Q bandpass filter [23-25]. The goal of this chapter is to provide a comprehensive overview and comparative performance analysis for all of these types of multi-gigabit rate CDRs.

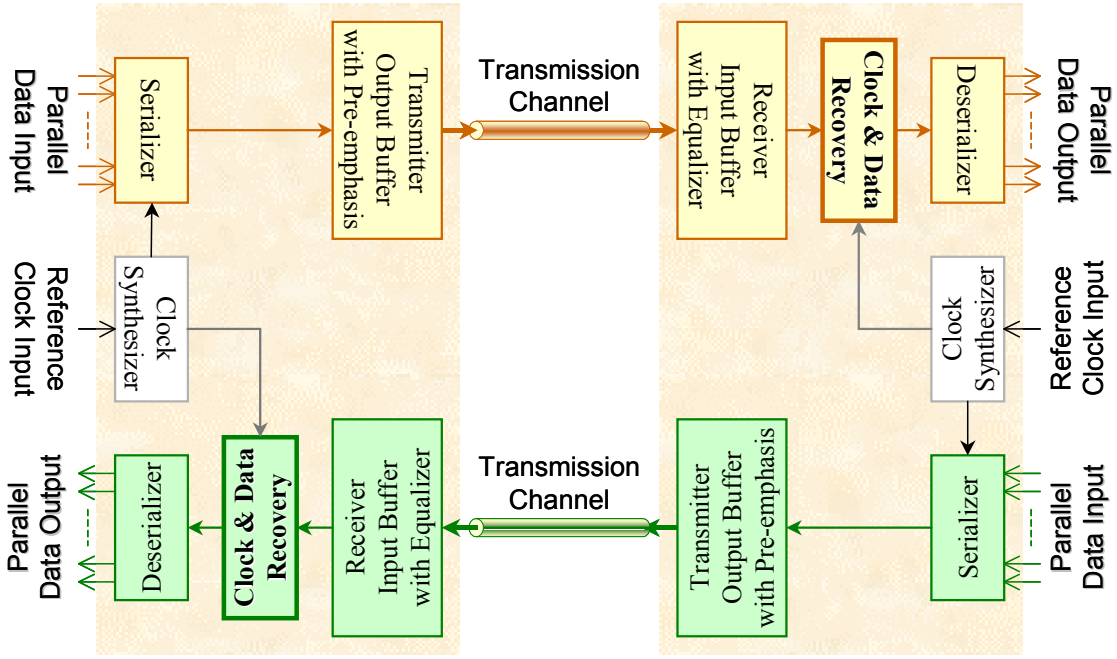


Figure 7.3 Block diagram of a generic high-speed wire-linked transmission system.

The remainder of this chapter is organized as follows: Section 7.2 provides an overview of CDR architectures that are commonly used in modern high-speed wire-linked data transmission and discusses the design challenges and other considerations for each type of CDR. Section 7.3 discusses the performance tradeoffs among these architectures. The chapter concludes with a summary of appropriate CDR architectures for a wide range of applications having various performance requirements.

7.2 CDR ARCHITECTURES

CDR architectures can be classified according to the phase relationship between the received input data and the local clock at the receiver. Commonly used CDR topologies

may be divided into three major categories:

(1) Topologies using feedback phase tracking, including, phase locked loop (PLL), delay locked loop (DLL), phase interpolator (PI) and injection locked (IL) structures.

(2) An oversampling-based topology without feedback phase tracking.

(3) Topologies using phase alignment but without feedback phase tracking, including gated oscillator and high-Q bandpass filter architectures.

In the following subsections, we will present the structure, operation, advantages and design challenges for each of these types of CDR architectures.

7.2.1 PLL-BASED CDR

CDR designs based on a PLL topology can be categorized according to whether or not they utilize a reference clock. They can be further categorized as analog or digital PLL-based CDR designs. PLL-based CDR designs inherently provide a tunable bit rate and are easily integrated in a monolithic design. However, a frequency acquisition aid is typically required in order to prevent false locking.

7.2.1.1 PLL BASED CDR DESIGNS WITHOUT REFERENCE CLOCK

Figure 7.4(a) shows an architecture without a reference clock [5], where a frequency tracking loop provides a frequency comparison through the frequency detector (FD) and a phase tracking loop leads to phase locking through the phase detector (PD). The FD module provides a frequency comparison between the input data, D(in), and the voltage-controlled oscillator (VCO) output clock, which eliminates the need for an external reference frequency. During either CDR startup or loss of phase lock, the FD is activated to produce a control voltage through the charge pump (CP) and the loop filter (LF), which moves the VCO oscillation frequency toward the input data rate. Once the

frequency difference falls within the phase tracking loop's capture range, the PD takes over and allows the VCO output clock phase to lock onto the input data phase.

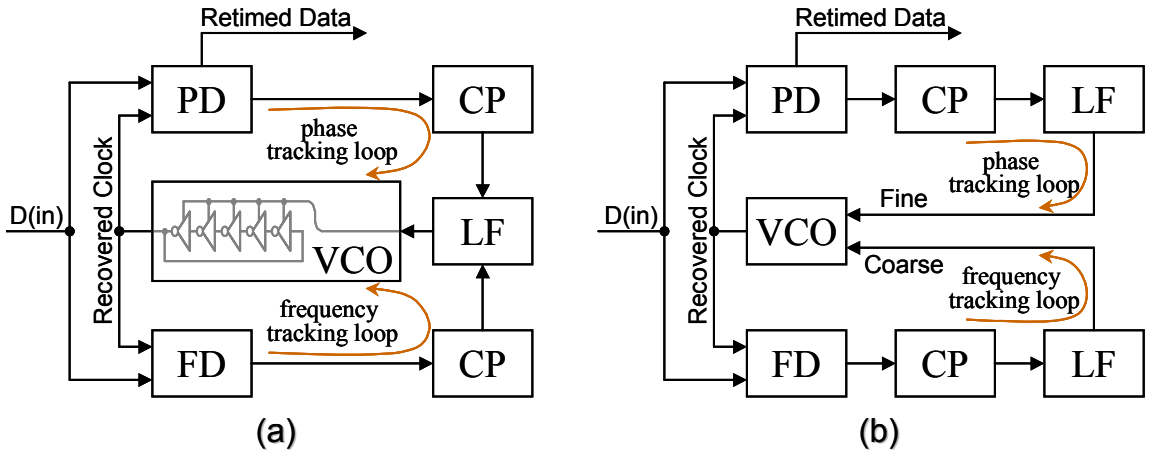


Figure 7.4 CDR without a reference clock. (a) Single control of VCO frequency tuning. (b) Coarse and fine control of VCO frequency tuning.

There are two possible issues associated with the CDR architecture of Figure 7.4(a). First, the frequency tracking loop and the phase tracking loop may potentially interfere with each other during the interval when the FD transfers control to the PD, resulting in a failure to lock onto the phase [3] and/or ripple generation on the VCO control line. Second, the FD could become momentarily "confused" about the actual input data rate if the received input data consists of random consecutive identical digits (CIDs) [3] or if the received rising and falling edges are corrupted by external or internal noise during the transmission. Because of these issues, the loop bandwidth of the frequency tracking loop is typically chosen to be much smaller than that of the phase tracking loop [3].

In order to independently select the bandwidths of the frequency locking loop and the phase locking loop, one can modify the system such that each loop not only has its own charge-pump (CP) but also its own loop filter (LF). This is illustrated in Figure 7.4(b)

[26], in which the frequency loop and phase loop drive the coarse control and fine control, respectively. However, this design has the disadvantage of requiring a larger total layout area due to the presence of two LFs. Reference [8] has suggested using a hybrid analog/digital loop filter in order to reduce this area overhead.

7.2.1.2 PLL-BASED CDR WITH AN EXTERNAL REFERENCE CLOCK

An example of a PLL-based CDR design with an external reference clock input is shown in Figure 7.5(a) [6], which uses a similar scheme of coarse and fine tracking loops. The frequency tracking loop with the phase-frequency detector (PFD) locks the output clock phase of VCO_2 to that of the input reference clock, F(ref). This is a completely stand-alone clock multiplication with VCO_2 acting as a replica circuit of VCO_1. The presence of the divided by M block in the frequency tracking loop allows the input reference clock to run at a low frequency. Since VCO_1 and VCO_2 are identical, the control voltage to VCO_2 can be used as a coarse control input to VCO_1 in such a way as to move the oscillation frequency of VCO_1 very close to or equal to the input data rate. Therefore, the frequency tracking loop provides a coarse control signal to VCO_1. The phase tracking loop with the PD locks the VCO_1 output clock phase to the input data to create a fine control signal for VCO_1. The gain of the phase tracking loop must be relatively low compared with that of the frequency tracking loop in order to maintain the fine control of VCO_1.

Figure 7.5(a). First, any mismatch between VCO_1 and VCO_2 could lead to a difference in oscillation frequencies even though the two VCOs share the same coarse input [3]. Second, the data rate of a high-speed serial link in an asynchronous mode of operation will often allow a certain frequency offset between the transmitted data rate and the

receiver's local clock frequency, which leads to a frequency offset between VCO_1 and VCO_2. A possible frequency pulling phenomena could move VCO_1 away from the received data rate and towards M × F(ref). This could be especially problematic when a spread spectrum clocking (SSC) scheme is required, such as in Serial AT Attachment (SATA) applications [13]. Another general concern regarding the CDR architecture shown in Figure 7.5(a) is the excessive layout area needed for the two VCO designs, especially in the case of using an LC VCO based PLL for clock generation [1-2]. However, the impact of the extra area for VCO_2 is less of a concern if the design is targeted for multi-channel applications since the frequency tracking loop is shared by multiple phase tracking loops.

On the other hand, having an independent clock multiplication from VCO_2 in Figure 7.5(a) makes it easier to satisfy the loop stability and bandwidth requirements. Furthermore, the availability of the coarse control signal from VCO_2 provides a large improvement in the acquisition time of the phase tracking loop.

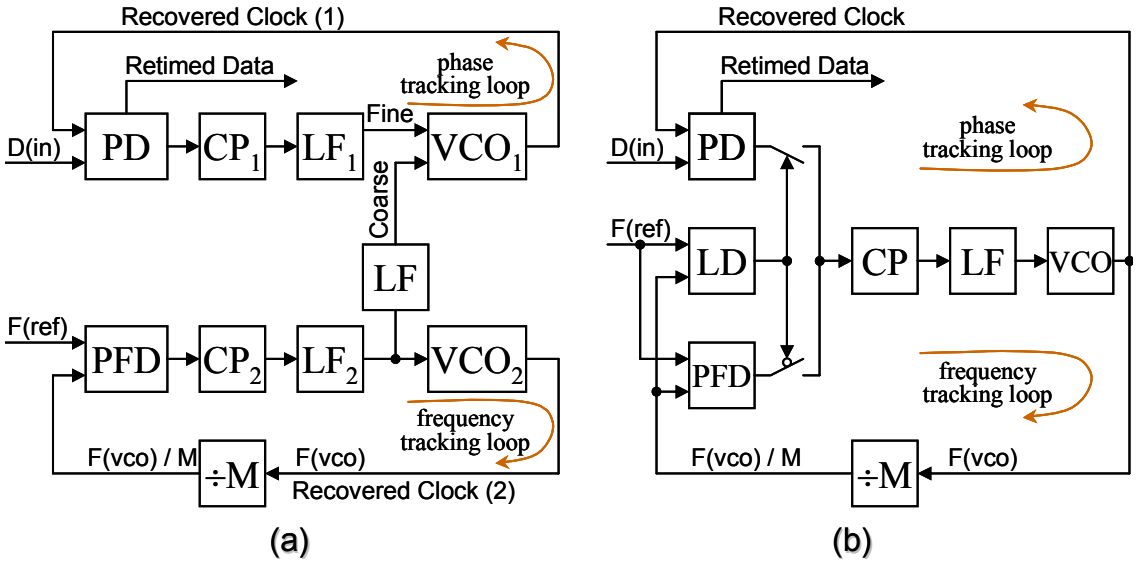


Figure 7.5 CDR with a reference. (a) Dual VCO locking. (b) Sequential locking.

One way to reduce the CDR design size in Figure 7.5(a) is through the sequential locking scheme of Figure 7.5(b). This uses a lock detector (LD) to sequentially enable the frequency loop and the phase loop, which eliminates the need for dual CPs, LFs and VCOs [27-28]. During CDR startup, the LD first activates the frequency loop, and moves the VCO oscillation frequency towards M × F(ref). Once the LD detects that the frequency of F(vco) ÷ M is equal to F(ref), it disables the frequency tracking loop and enables the phase tracking loop. If loss of phase locking occurs as a result of unexpected noise, the LD re-activates the frequency tracking loop and the phase tracking loop sequentially [2-3]. One potential issue with Figure 7.5(b) is that the transition from the frequency tracking loop to the phase tracking loop may disturb the VCO control signal and cause a VCO frequency shift when the FD transfers control to the PD. This, in turn, may result in a failure to lock phase due to a metastable transition and/or a large ripple on the VCO control signal, similar to the situation described earlier for the PLL-based CDR design without a reference clock [3].

7.2.1.3 DIGITAL PLL (DPLL) BASED CDR DESIGNS

CDR architectures in which the CP and LF are replaced by digital logic can minimize the required layout area and simplify the closed-loop stability analysis by minimizing the process, voltage, and temperature (PVT) variations of the LF. Figure 7.6(a) shows an example of a partially DPLL-based CDR design [29], which implements the CP and LF in the form of a digital LF (DLF) but which includes digital-to-analog converters (DACs) in both the frequency and phase tracking loops. This CDR architecture is similar to the one shown on Figure 7.4(a). Another significant advantage of using the DLF is that it allows the CP and LF functions to be easily programmable.

There are two important issues associated with this CDR design. First, the potentially long loop latency from the DLF and the DAC may degrade the phase and frequency tracking capability, especially in an SSC operational mode, which has reduced CDR jitter tolerance [13]. Second, the finite resolution of the DAC causes VCO frequency wandering between adjacent frequency steps, which increases jitter generation. The two DACs in Figure 7.6(a) can be eliminated by designing a VCO having digital switches to fine-tune the VCO frequency [8]. However, the issues of long loop latency and finite resolution still remain.

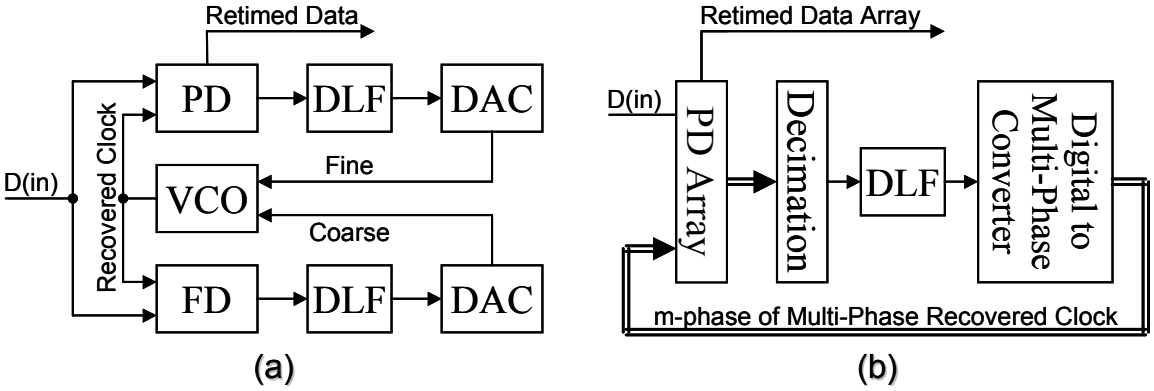


Figure 7.6 DPLL CDR architecture with (a) DLF and DAC. (b) DLF and DMPC.

The DPLL architecture shown in Figure 7.6(b) utilizes a digital-to-multi-phase converter (DMPC) [7], generating *m* clock phases which are fed back to the PD array. The PD array consists of multiple bang-bang phase detectors which use the multi-phase clocks from the DMPC to sample multiple data bits. The data sampling process produces multiple early or late indication signals of phase error for data transitions and neutrals for non-transitions. The decimation block reduces the multiple early/late/neutral signals to an effective early, late, or neutral signal at a lower rate. The CDR design of Figure 7.6(b) faces the same issues as the one in Figure 7.6(a) with a potential long loop latency from

the decimation in the DLF, and finite resolution from the DMPC. The DPLL-based CDR architecture of Figure 7.6(b) is generally used in low to moderate rate applications, but provides a programmable, all-digital design that is easily transferable between different process technologies.

7.2.2 DLL-BASED CDR

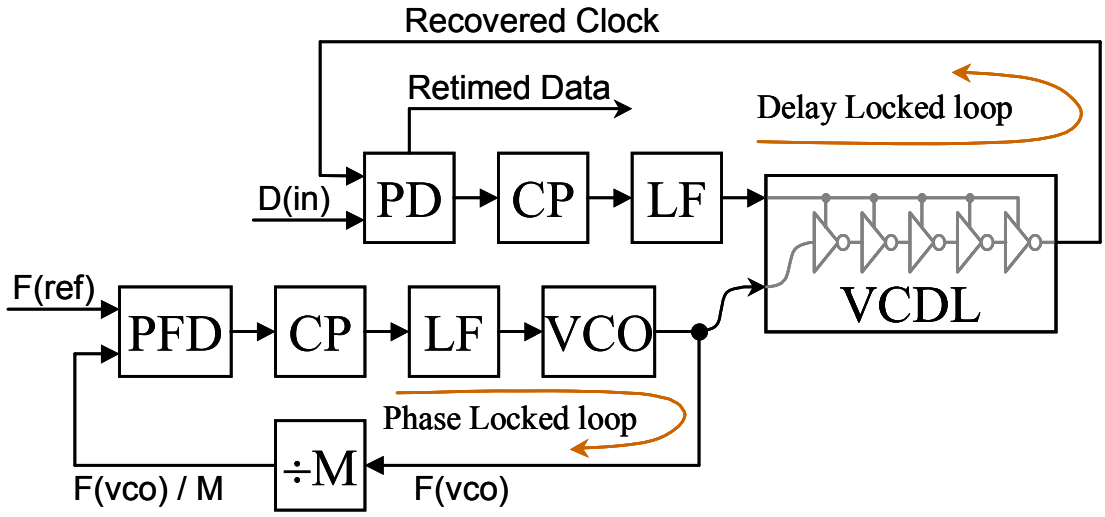


Figure 7.7 DLL-based CDR architecture.

The DLL-based CDR architecture shown in Figure 7.7 [9] often shares a common PLL-based reference clock generator among multiple channels. This structure avoids the drawbacks of multi-VCO coupling/pulling, high power dissipation and large area. The DLL-based CDR design shown in Figure 7.7 is similar to the PLL-based CDR design of Figure 7.5(a). Here, however, the frequency tracking loop provides a reference clock rather than a control voltage signal. Also, the phase tracking loop uses a voltage-controlled delay line (VCDL) for phase synchronization instead of a VCO. The reference clock for the VCDL, F(vco), must oscillate at the input data rate and is typically generated from a shared PLL-based clock multiplication which provides a low-pass

filtering of the input reference clock, F(ref), in order to reduce jitter transferred from F(ref) [2-3].

The primary benefit of using a DLL-based CDR is that it does not have the jitter accumulation issue [30] of a PLL-based CDR design [2]. Also, a DLL-based CDR provides a more stable system [30]. The VCDL control voltage directly alters the clock phase, whereas the VCO control voltage indirectly alters the clock phase through the integral of the dynamically changing clock frequency. Therefore, the VCDL does not introduce a pole in the loop transfer function. Furthermore, a DLL-based CDR design provides faster lock speed because there is no need for clock synthesis [9].

The primary drawback of the DLL-based CDR topology shown in Figure 7.7 is its limited phase capturing range, so that it is unable to handle any frequency offset between the transmitter and receiver. Therefore, the DLL-based CDR architecture shown in Figure 7.7 is most suitable for source-synchronous applications such as chip-to-chip interconnections [10].

7.2.3 COMBINATION OF PLL/DLL BASED CDR

The dual loop CDR topology shown in Figure 7.5 can have good input jitter rejection as a result of a narrow loop bandwidth in the phase tracking loop while also having a short acquisition time due to its frequency tracking loop. However, a PLL-based CDR topology with a second- or higher-order closed-loop frequency response often needs a closed-loop zero to stabilize the loop, which causes the PLL to exhibit jitter peaking in its input-to-output transfer function. This jitter peaking behavior is very undesirable, especially in an application such as SONET (Synchronous Optical Network), which cascades several CDRs as a string of repeaters, leading to the accumulation of jitter.

Reducing the PLL loop bandwidth can also minimize jitter peaking but with an increase in the acquisition time.

One way to eliminate jitter peaking and allow the PLL to maintain a small loop bandwidth without compromising acquisition speed is to combine the DLL-based and PLL-based CDR architectures, as shown in Figure 7.8(a) [30]. Here, a requirement is that the PLL should not provide a closed-loop zero, which is accomplished by modifying the loop filter so that it only uses a capacitor [30].

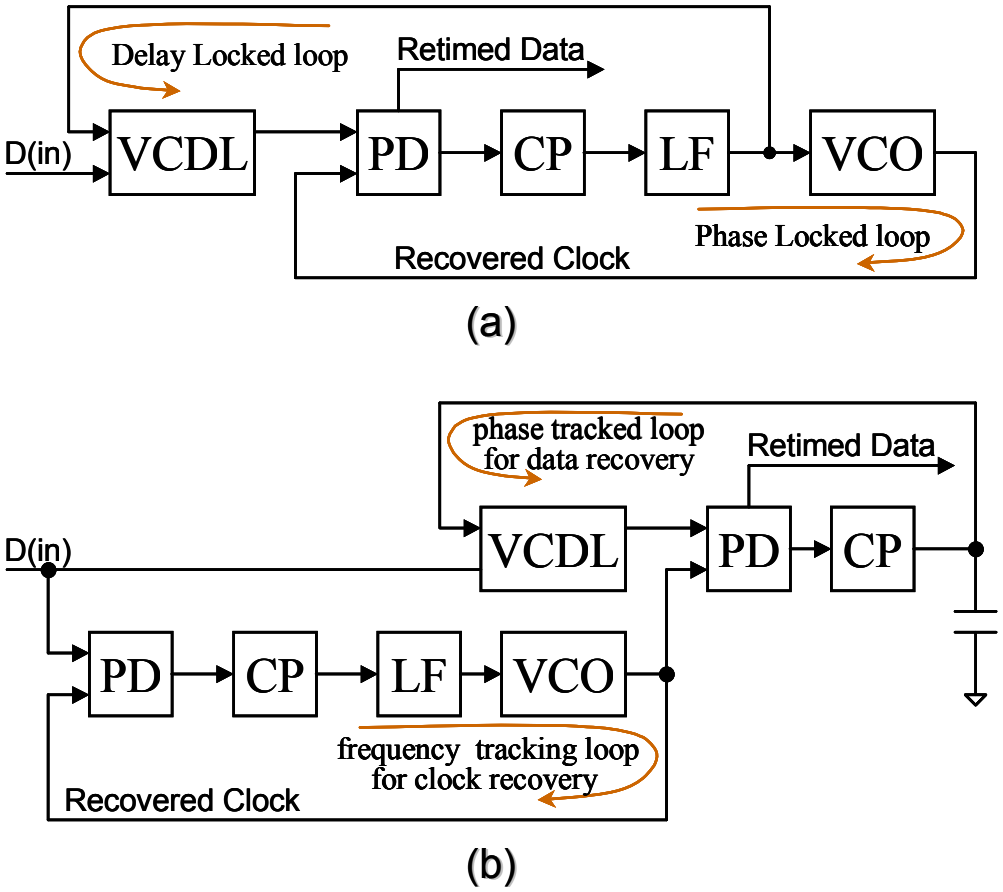


Figure 7.8 Combination of PLL-based and DLL-based CDR architectures. (a) A shared tracking loop. (b) Independent tracking loops.

The primary concern with the CDR topology of Figure 7.8(a) is that the loop can become unstable if the VCDL is driven to the edge of its delay range. This is due to the

fact that both the DLL and the PLL share the same control voltage. The stabilizing zero for the PLL provided by the DLL of Figure 7.8(a) is no longer present once the DLL is driven to its delay range limit and acts as an open loop response. One way to eliminate this potential problem is to constrain the VCO tuning range to be a subset of the VCDL tuning range. Furthermore, both the VCO and the VCDL must be driven in the same phase direction. Figure 7.8(b) [31] shows an alternative design having all of the benefits from the design of Figure 7.8(a) together with independent tracking loops. Here, the DLL loop dynamics do not affect the PLL performance, at a cost of requiring dual CPs and LFs.

7.2.4 PHASE INTERPOLATOR (PI) BASED CDR

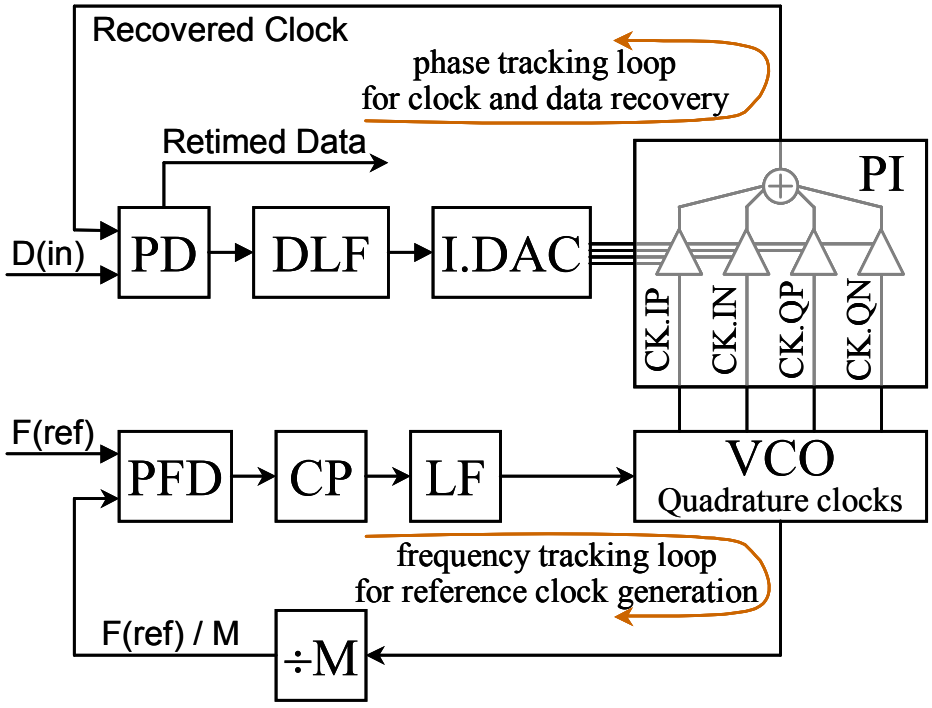


Figure 7.9 Phase Interpolator based CDR architecture.

The topology and operating mechanism of the Phase Interpolator (PI) based CDR

architecture shown in Figure 7.9 [11-13] are similar to those of the DLL-based CDR design of Figure 7.7. In this structure, however, the CP and LF are replaced by a digital LF (DLF) and a current digital-to-analog converter (I.DAC), and the VCDL is replaced by a PI. The recovered clock phase from the PI is driven directly by the I.DAC using a function proportional to the control voltage. Both DLL-based and PI-based CDR topologies offer the benefits of increased system stability, faster acquisition and a lack of jitter peaking compared with a PLL-based CDR. However, jitter peaking in PI-based CDR designs is absent only if the loop latency is not significantly larger than the PI phase update period. The reason for this is that the gradient of fast changing jitter has already reversed its direction by the time the phase shift control signal reaches the PI [10].

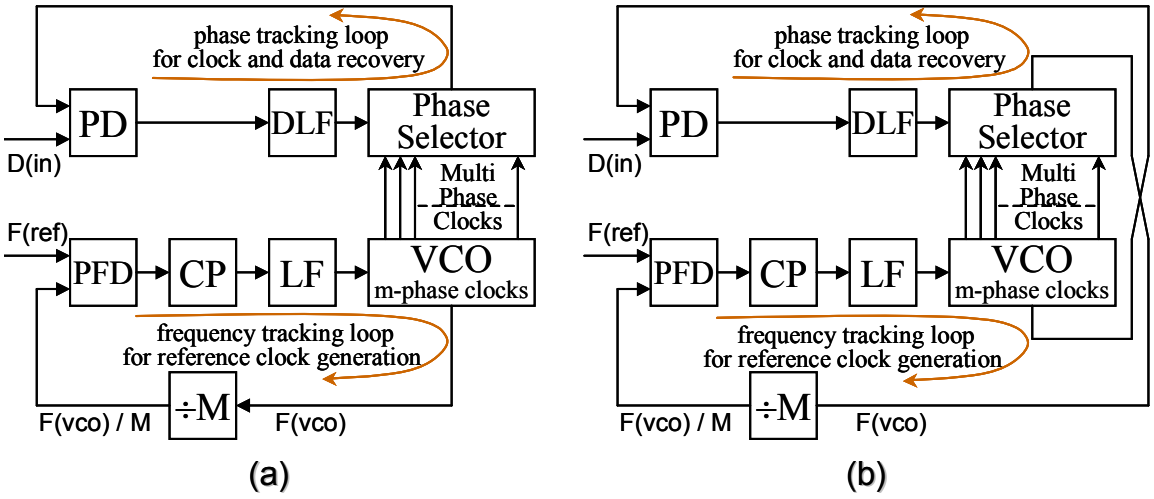


Figure 7.10 Two variants of the Phase Interpolator based CDR architecture with clock phase in the phase tracking loop. (a) Discrete shift. (b) Continuous shift.

The primary difference between PI-based and DLL-based CDR designs is that the PI-based CDR can operate over a wide range of data rates with a certain allowable frequency offset between transmitter and receiver in a source-asynchronous scenario. The design considerations for phase interpolator (PI) based CDRs are the I.DAC

resolution, PI phase shift linearity and the loop latency, all of which have a direct impact on CDR jitter performance. Furthermore, having the reference clock at the speed of the received data may pose a challenge when delivering quadrature clocks across a chip in multi-gigabit, multi-channel applications.

Two variants of the phase interpolator based CDR architectures are shown in Figure 7.10 [32]. These structures replace the I.DAC and PI of Figure 7.9 with a phase selector, which potentially can lead to a smaller design having fewer analog components. Figure 7.10(a) provides a discrete clock phase shift in the phase tracking loop. The main advantage is the use of independent phase/frequency tracking loops, which simplifies the loop bandwidth and stability requirements. Another advantage is the complete use of digital components in the phase tracking loop, which leads to less impact from process, supply voltage, and temperature variations. The primary issue with the design of Figure 7.10(a) is that the discrete clock phase shift step leads to larger cycle-to-cycle jitter. However, the smaller phase spacing produced by the VCO leads to lower VCO frequency, higher power dissipation, and a larger area to accommodate the increased number of clock phases. One way to smooth out the discrete phase shift step in Figure 7.10(a) is to swap the phase select and VCO output connections, as shown in Figure 7.10(b). The discrete change in phase selection from the DLF in the phase tracking loop is smoothed out by the LF and CP in the frequency tracking loop, which provides smooth frequency and phase drifting in the phase tracking loop. The major advantage of Figure 7.10(b) is that the loop bandwidths can be selected separately, however, it will not be able to support a multi-channel application with a single frequency tracking loop for reference clock generation.

7.2.5 INJECTION LOCKED BASED CDR

The Injection Locked (IL) based CDR architecture shown in Figure 7.11 [14] is also a variant of the phase interpolator based CDR topology of Figure 7.9 and shares the same advantages of being a more stable system, having a faster acquisition time and an absence of jitter peaking, as compared with PLL-based CDRs. The phase selector, slave oscillator and injection driver in Figure 7.11 perform the operations of the I.DAC and PI of Figure 7.9. Here, the slave oscillator is locked by the frequency and phase injection from the injection driver. However, the slave oscillator acts like a low-pass filter and smoothes out duty-cycle distortion from the phase selector. This means that the recovered clock exhibits a much smoother phase shift compared to the phase interpolator based CDR design of Figure 7.9. Under the proper injection locked condition, the two clocks from the phase selector must be 180 degrees out of phase in order to maintain balanced injection into the differential slave oscillator. Furthermore, the adjustable current gain in the slave oscillator must be reduced during the activation of the injection driver such that equal clock phase separation in the slave oscillator is maintained [14].

The injection locked based CDR design can exhibit a better duty-cycle balanced recovered clock and improved phase tracking jitter generation compared to the traditional phase interpolator based CDR design. (Phase tracking jitter is long-term jitter with respect to the ideal recovered clock.) However, these improvements trade off against the slave oscillator's lock range. Furthermore, a careful design and layout of the injection driver and the slave oscillator are needed in order to prevent any unwanted injection from such sources as the supply, substrate or any adjacent toggling signals.

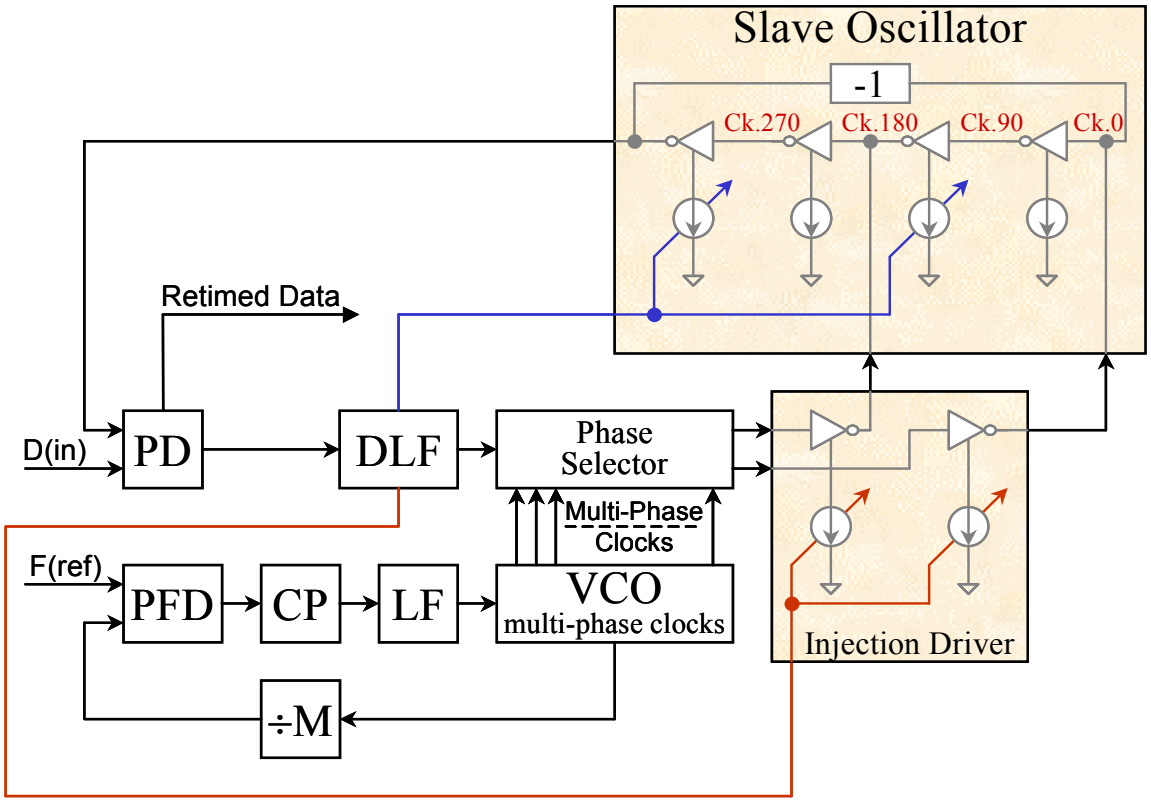


Figure 7.11 Injection Locked based CDR.

7.2.6 OVERSAMPLING BASED CDR

A CDR design based on the oversampling architecture, as shown in Figure 7.12 [16-17], provides data recovery without any time delay. Unlike the phase tracking based CDR design with its continuous adjustment of the recovered clock phase to track the received data phase, the oversampling based CDR circuit samples each received data bit at multiple points.

A minimum of 3 samples per received data bit are required for properly recovering the received data, as shown in Figure 7.12(b). The data recovery block in Figure 7.12(a) consists of a data register, bit boundary detector and data selector. The data register is a first-in first-out (FIFO) buffer, which temporarily stores the sampled data from the multi-

phase sampler while the data selector determines which ones will be retained. The bit boundary detector defines the data bit edge samples, which allows the data selector to determine the proper data sample to retain as the recovered data.

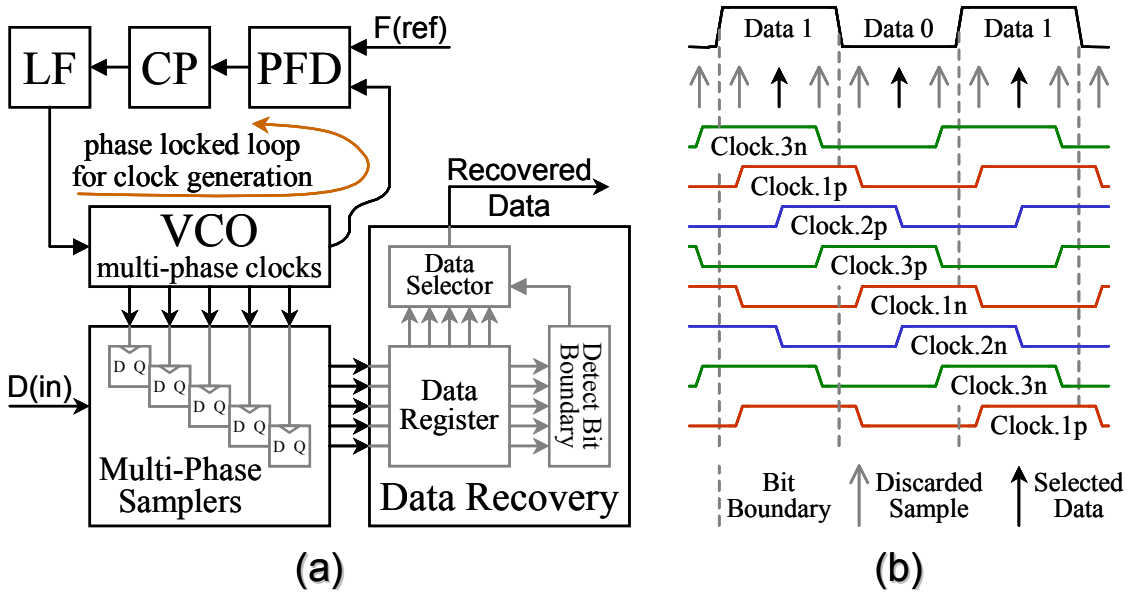


Figure 7.12 Oversampling based CDR design. (a) Circuit block diagram. (b) Example of received data and sampling clocks for 3X oversampling.

The primary advantages of the oversampling CDR design are its fast acquisition time and inherent stability. Furthermore, the feed-forward operation mechanism provides a very high data bandwidth. The oversampling CDR design technique is applicable in both burst-mode and continuous-mode data transmission because of the absence of feedback phase tracking and jitter transfer accumulation.

The drawbacks of an oversampling-based CDR are the need for high frequency data transitions to achieve high-frequency jitter rejection and the requirement of a large FIFO for sampled data storage, especially in a high-speed source-asynchronous system having a frequency offset between transmitter and receiver.

7.2.7 GATED OSCILLATOR BASED CDR

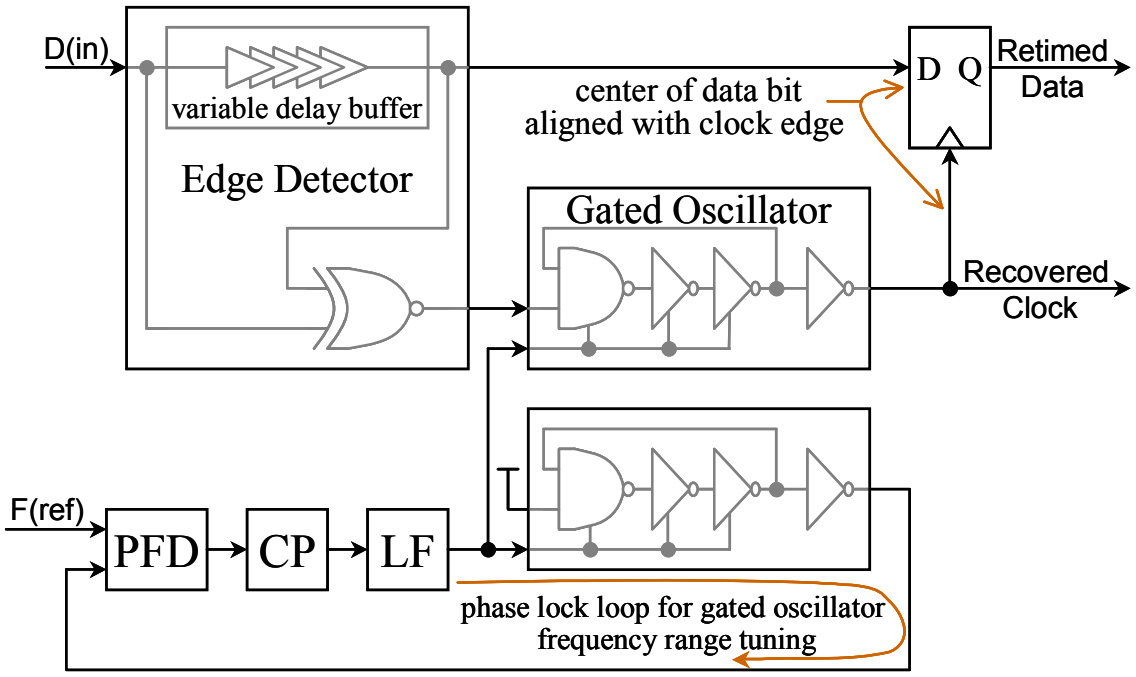


Figure 7.13 Gated oscillator based CDR.

Some applications such as passive optical networks (PONs) and optical packet routing systems impose no restrictions on the amount of jitter transfer but require a burst-mode operation to extract a synchronous clock and recover the received data immediately for each asynchronous packet [20,33]. The gated oscillator architecture shown in Figure 7.13 [20-22] is commonly used for such applications. The synchronous clock is derived from the gated oscillator which is triggered from the pulse generated in the edge detector and which follows the data transient edges. The frequency tuning for a gated oscillator is controlled through a replica gated oscillator from a PLL with its gated input tied to logic high. The variable delay buffer in the edge detector provides a data phase shift for the edge detector to determine the data transition edge. It also allows the received data to be phase aligned with the recovered clock.

In addition to its fast synchronous clock recovery and data acquisition, the gated oscillator based CDR design is also a simpler and smaller design having lower power for multi-channel operation compared to oversampling based CDR designs.

Its major drawback is that it has no jitter rejection due to its broadband open loop design without loop bandwidth filtering. Furthermore, the phase alignment between the received data and the recovered clock is sensitive to process, temperature, data rate, and supply voltage variations [4]. Finally, a gated oscillator based CDR design is more difficult to transfer from one process technology to another.

7.2.8 HIGH-Q FILTER BASED CDR

A simple open-loop based CDR design is shown in Figure 7.14 [4,23,25,34], which uses a high quality factor (high-Q) bandpass filter to replace the gated oscillator and PLL of Figure 7.13. The topology shown in Figure 7.14 is a technique that has been traditionally used in non-monolithic CDR designs [4]. The combination of the variable delay buffer and XOR logic gate in the edge detector operate as a pulse generator based on the received data transitions. The high-Q bandpass filter extracts the transition frequency, thereby recovering the clock at the received data rate.

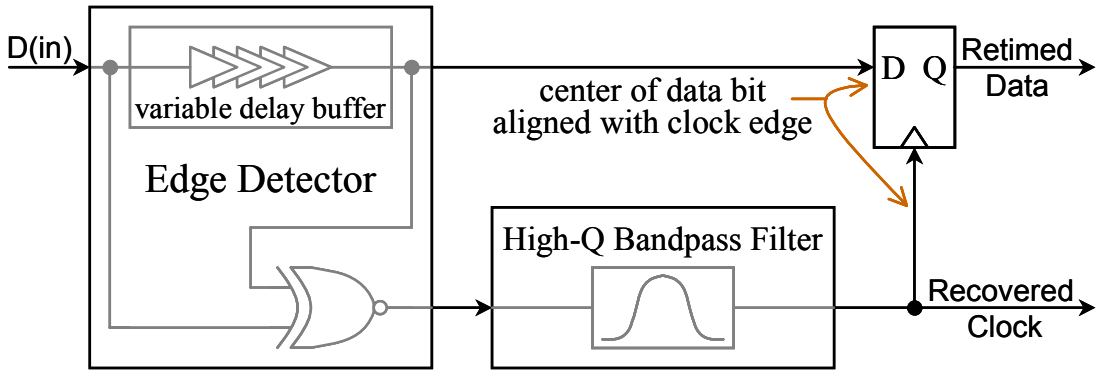


Figure 7.14 High-Q band-pass filter based CDR.

In addition to being a simple design with a low cost of development, the high-Q bandpass filter has the same advantages of fast synchronous clock recovery and data acquisition as the gated oscillator based CDR. The filter designs are often based on an LC tank, surface acoustic wave (SAW) filter, dielectric resonator or PLL [4]. However, a PLL based filter [31] will not be able to perform an instantaneous clock extraction due to its long feedback phase tracking settling time. Both of the CDR implementations in [25][34] use an off-chip SAW filter. The major limitation of this architecture is the difficulty of implementing a high-Q bandpass filter in a monolithic design. Also, there is no input jitter rejection and the clock-data phase alignment is sensitive to process, temperature, data rate and supply voltage variations as was the case for the gated oscillator based CDR designs.

7.3 PERFORMANCE COMPARISON AND TRADEOFFS

CDR applications can be categorized as being either burst-mode or continuous-mode. A burst-mode system is often used in a point-to-multipoint application, where different senders transmit bursts of packet data with a silence time slot between bursts [36]. The data transmission link is re-activated whenever a packet of data is requested to be transmitted and remains inactive at other times in order to leave the data transmission link available for other users. Burst-mode data transmission often requires very fast acquisition time in order to meet the low network latency requirement, which is usually within a few bytes of a preamble period [16]. Examples of burst-mode applications are the Fiber-To-The-Home (FTTH) Network, Asynchronous Transfer Mode (ATM) Network, Ethernet Passive Optical Network (EPON), Gigabit Passive Optical Network (GPON) and Local Area Networks (LANs). The commonly used CDR architectures for

burst-mode receivers are topologies without feedback phase tracking such as the gated oscillator and oversampling techniques in order to meet the low network latency requirement [22].

A continuous-mode system is often used in point-to-point applications, in which a steady and uninterrupted stream of bits is transmitted [36]. A fast acquisition time is often not required in such systems. However, some applications such as SONET have a stringent jitter transfer specification in order to avoid jitter accumulation from repeaters, which requires the CDR to have very low or no jitter generation [33]. Furthermore, SONET applications must also tolerate a long sequence of consecutive identical digits (CIDs) [35], which leads to fewer transitions in the transmitted data pattern and provides less frequency content for retrieving the clock. In addition to SONET, other examples of continuous-mode applications are Fiber Channel and Gigabit Ethernet. The commonly used CDR architectures for continuous-mode receivers are PLL, DLL and combined PLL/DLL based topologies. Recently, the phase interpolator, injection locked and oversampling techniques have also been used in continuous-mode CDR designs.

PLL-based CDR designs have very good input jitter rejection but suffer from jitter peaking and stability concerns. On the other hand, a DLL-based CDR topology has no jitter peaking or stability concerns. Furthermore, it is well suited to multi-channel applications due to the lack of crosstalk injection or frequency pulling among VCOs. However, it is generally restricted to source-synchronous systems due to its limited phase capturing range. The combined PLL/DLL-based CDR architecture has the benefits of both of the PLL and DLL. However, its design complexity is larger because of the need to analyze the behaviors of two loops.

A phase interpolator based CDR design does not have jitter peaking or stability concerns and it has an unlimited phase capturing range but it suffers from quantization errors. The injection locked CDR design provides duty cycle correction but forces a tradeoff between its tracking jitter performance and the slave oscillator lock range.

Table 7.1 CDR Architecture Comparison.

Architecture based on	Advantages	Disadvantages	Applications (Suggested / Reported)
PLL	Input Jitter Rejection Input Frequency Tracking Good for SSC Frequency Tracking	Jitter Peaking Large Loop Filter Area (Analog) Multi-Channel Crosstalk / Pulling Long Acquisition Time	Continuous-Mode Source-Asynchronous/Synchronous SONET / SDH / Gigabit Ethernet High Speed Serial Link
DLL	Stable / First Order System No Jitter Peaking Multi-Channel Share Input Clocks	Source-Synchronous Only Large Loop Filter Area (Analog) Limited Phase Capturing Range	Continuous-Mode Source- Synchronous High Speed Serial Link Chip-to-Chip Interconnection
PLL / DLL	No Jitter Peaking Small Loop BW / Fast Acquisition Track Frequency / Good for SSC	Multi-Channel Crosstalk / Pulling Require Analysis for Two Loops	Continuous-Mode Source-Asynchronous/Synchronous SONET / Ethernet /Fibre Channel Multi-Gbps Link / Optical Receiver
Phase Interpolator	Multi-Channel Share Input Clocks	Quantization Phase Error Multi-Phase Clock Routing Cycle-to-Cycle Jitter	Continuous-Mode Source-Asynchronous/Synchronous SONET / SDH / OIF CEI Serial AT Attachment (SATA)
Injection Locked	Good Jitter Tolerance Duty-Cycle Correction Multi-Channel Share Input Clocks	Quantization Phase Error Multi-Phase Clock Routing Cycle-to-Cycle-Jitter Large Oscillator Range, Large Jitter	Continuous-Mode Source-Asynchronous/Synchronous Cross-Point Switch / SONET Multi-Gbps Serial Link I/Os
Oversampling	No Feedback Phase Tracking Fast Acquisition Easy Transferred in Technologies No Stability Concern	Digital Circuit Complexity Large FIFO Size Possible Long Data Latency Require Multi-Phase Clocks	Burst / Continuous-Mode Source-Asynchronous/Synchronous Serial AT Attachment (SATA) Fiber Channel / Backplane / PON
Gated VCO	No Feedback Phase Tracking Fast Acquisition Small-Area	Data / Clock Phase Aligning No Input Jitter Rejection Possible Multi-Channel Crosstalk / Pulling	Burst-Mode Source-Asynchronous/Synchronous Fiber Channel / PON Short-Haul Data Transmission
High-Q Bandpass Filter	No Feedback Phase Tracking Fast Acquisition Low Power Fast Time / Low Cost Development	Data / Clock Phase Aligning Difficult to Design in Monolithic IC No Input Jitter Rejection	Burst-Mode Source-Asynchronous/Synchronous SONET / SDH Fiber-to-the-Desk (FTTD) / LAN

The oversampling, gated oscillator and high-Q bandpass filter based CDR designs all provide a rapid data recovery capability. The oversampling-based CDR topology offers a complete digital design solution, which is easily transferable between different process technologies. However, it has long data latency and it requires a large FIFO. The gated

oscillator and high-Q bandpass filter based CDR designs provide rapid clock and data recovery but have no input jitter rejection and no intrinsically aligned clock-data phase for optimum data sampling points. The high-Q bandpass based CDR has the lowest design cycle time but it is difficult to integrate into a monolithic design.

A listing of the advantages and disadvantages, including suggested or reported applications for each type of CDR architecture, is given in Table 7.1. This table provides a comparison and tradeoff summary amongst all of these CDR topologies so that appropriate candidate architectures for a given application of interest can be determined.

7.4 CONCLUSIONS

An overview of commonly used CDR architectures has been presented which discusses the applications and design challenges along with their advantages and limitations. PLL, DLL, Phase Interpolator and Injection Locked based CDR designs are suitable for continuous-mode communication. On the other hand, gated oscillator and high-Q bandpass filter based CDR designs are more applicable in burst mode systems. The oversampling based architecture is capable of handling both burst- and continuous-mode data. The DLL-based CDR is not applicable in a source-asynchronous system due to its limited phase capturing range. The strengths and weaknesses for each type of CDR design have been discussed in detail and a summary of the tradeoffs and applications for each type has been provided.

REFERENCES

[1] ITRS, "International Technology Roadmap for Semiconductors 2007 Edition: Assembly and Packaging," *International Technology Roadmap for Semiconductors (ITRS)*, http://www.itrs.net, 2007

[2] B. Razavi, "Design of Integrated Circuits for Optical Communications," *McGraw-Hill*, New York, 2003

[3] B. Razavi, "Challenges in the Design of High-Speed Clock Data Recovery Circuit," *IEEE Communication Magazine*, pp.94-101, Aug. 2002

[4] R. C. Walker, "Designing Bang-Bang PLLs for Clock and Data Recovery in Serial Data Transmission Systems (Book style with paper title and editor)," *Phase-Locking in High-Performance Systems from Devices to Architectures*, B. Razavi, Ed. New Jersey: John Wiley & Sons, Inc., 2003, pp.34–45, 2003

[5] A. Pottbäcker, U. Langmann, and H.-U. Schreiber, "A Si Bipolar Phase and Frequency Detector IC for Clock Extraction Up to 8 Gb/s," *IEEE Journal of Solid-State Circuits*, vol.27, no.12, pp.1747–1751, Dec. 1992

[6] J.C. Scheytt, G. Hanke, and U. Langmann, "A 0.155-, 0.622, and 2.488-Gb/s Automatic Bit-Rate Selecting Clock and Data Recovery IC for Bit-Rate Transparent SDH Systems," *IEEE Journal of Solid-State Circuits*, vol. 34, no.12, pp.1935–1943, Dec. 2003

[7] J. L. Sonntag and J. Stonick, "A Digital Clock and Data Recovery Architecture for Multi-Gigabit/s Binary Links," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 8, pp.1867–1875, Aug. 2006

[8] M.H. Perrott, et al., "A 2.5-Gb/s Multi-Rate 0.25-μm CMOS Clock and Data Recovery Circuit Utilizing a Hybrid Analog/Digital Loop Filter and All-Digital Referenceless Frequency Acquisition," *IEEE Journal of Solid-State Circuits*, vol. 41, no.12, pp.2930–2944, Dec. 2006

[9] X. Maillard, F. Devisch, and M. Kuijk, "A 900-Mb/s CMOS Data Recovery DLL Using Half-Frequency Clock," *IEEE Journal of Solid-State Circuits*, vol.37, no.6, pp.711–715, Dec. 2002

[10] H.-H. Chang, R.-J. Yang, and S.-I. Liu, "Low Jitter and Multirate Clock and Data Recovery Circuit Using a MSADLL for Chip-to-Chip Interconnection," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol.51, no.12, pp.2356–2364, Dec. 2004

[11] R. Kreienkamp, U. Langmann, C. Zimmermann, T. Aoyama, and H. Siedhoff, "A 10-Gb/s CMOS Clock and Data Recovery Circuit with an Analog Phase Interpolator," *IEEE Journal of Solid-State Circuits*, vol. 40, no.3, pp.736–743, Mar. 2005

[12] M. Y. He and J. Poulton, "A CMOS Mixed-Signal Clock and Data Recovery Circuit for OIF CEI-6G+ Backplane Transceiver," *IEEE Journal of Solid-State Circuits*, vol. 41, no.3, pp.597–606, Mar. 2006

[13] M. Hsieh, and G.E. Sobelman, "Clock and Data Recovery with Adaptive Loop Gain for Spread Spectrum SerDes Applications," *IEEE International Symposium on Circuits and Systems*, pp.4883-4886, May 2005

[14] H.-T. Ng, et.al , "A Second-Order Semidigital Clock Recovery Circuit Based on Injection Locking," *IEEE Journal of Solid-State Circuits*, vol. 38, no.12, pp.2101–2110, Dec. 2003

[15] R. Farjad-Rad, et al., "A 33-mW 8-Gb/s CMOS Clock Multiplier and CDR for Highly Integrated I/Os," *IEEE Journal of Solid-State Circuits*, vol.39, no.9, pp.1553–1561, Sept. 2004

[16] J. Kim and D.-K. Jeong, "Multi-Gigabit-Rate Clock and Data Recovery Based on Blind Oversampling," *IEEE Communication Magazine*, pp.68-74, Dec. 2003

[17] S. I. Ahmed and T. A. Kwasniewski, "Overview Of Oversampling Clock and Data Recovery Circuits" *Canadian Conference on Electrical and Computer Engineering*, pp.1876-1881, May 1-4, 2005

[18] M. Ierssel, A. Sheikholeslami, H. Tamura, and W.W. Walker, "A 3.2 Gb/s CDR Using Semi-Blind Oversampling to Achieve High Jitter Tolerance," *IEEE Journal of Solid-State Circuits*, vol.42, no.10, pp.2224–2234, Oct. 2007

[19] S.-H. Lee, et al., "A 5-Gb/s 0.25-μm CMOS Jitter-Tolerant Variable-Interval Oversampling Clock/Data Recovery Circuit," *IEEE Journal of Solid-State Circuits*, vol.37, no.12, pp.1822–1830, Dec. 2002

[20] M. Nogawa, et al., "A 10 Gb/s burst-mode CDR IC in 0.13μm CMOS," *IEEE International Solid-State Circuits Conference*, Feb. 2005

[21] A. Tajalli, P. Muller, M. Atarodi, and Y Leblebici, "A Multichannel 3.5mW/Gbps/Channel Gated Oscillator Based CDR in 0.18μm Digital CMOS Technology," *Proceedings of the 31st European Solid-State Circuits Conference,* pp.193-196, Sept. 2005

[22] A. Tajalli, P. Muller, and Y. Leblebici, "A Power-Efficient Clock and Data Recovery Circuit in 0.18um CMOS Technology for Multi-Channel Short-Haul Optical Data Communication," *IEEE Journal of Solid-State Circuits*, vol.42, no.10, pp.2235–2244, Oct. 2007

[23] R. Walker, "Clock and Data Recovery for Serial Digital Communications," *IEEE International Solid-State Circuits Conference - Short Course*, http://www.omnisterra.com/walker/pdfs.talks/ISSCC2002.pdf, Feb. 2002

[24] K. Ishii, K. Kishine, and H. Ichino, "A Jitter Suppression Technique for a 2.48832-Gb/s Clock and Data Recovery Circuit," *IEEE Transactions on Circuits and Systems II: Analog And Digital Signal Processing,* vol.49, no.4, pp.266–273, Apr. 2002

[25] K. Yamashita, M. Nakata, N. Kamogawa, O. Yumoto, and H. Kodera, "Compact-Same-Size 52- and 156 Mb/s SDH Optical Transceiver Modules," IEEE *Journal of Lightwave Technology,* vol.12, no.9, pp.1607–1615, Sept. 1994

[26] H. Wang and R. Nottenburg, "A 1 Gb/s CMOS Clock and Data Recovery Circuit," *IEEE International Solid-State Circuits Conference*, Feb. 1999

[27] L. Henrickson, et al., "Low Power Fully Integrated 10-Gb/s SONET/SDH Transceiver in 0.13-μm CMOS," *IEEE Journal of Solid-State Circuits*, vol.38, no.10, pp.1595–1601, Oct. 2003

[28] H. S. Muthali, T. P. Thomas, and I. A. Young, "A CMOS 10-Gb/s SONET Transceiver, " *IEEE Journal of Solid-State Circuits*, vol. 39, no.7, pp. 1026–1033, Jul. 2004

[29] P. K. Hanumolu, M.G. Kim, G.-Y. Wei, and U.-K. Moon, “A 1.6Gbps Digital Clock and Data Recovery Circuit,” *IEEE Custom Integrated Circuits Conference*, pp.603–606, Sept. 2006

[30] D. Dalton, et al., “12.5-Mb/s to 2.7-Gb/s Continuous-Rate CDR with Automatic Frequency Acquisition and Data-Rate Readback,” *IEEE Journal of Solid-State Circuits*, vol.40, no.12, pp.2713–2725, Dec. 2005

[31] W. Rhee, et al., “A 10-Gb/s CMOS Clock and Data Recovery Circuit Using a Secondary Delay-Locked Loop,” *Proceedings of the IEEE 2003 Custom Integrated Circuits Conference*, pp.81–84, Sept. 2003

[32] P. Larsson, “A 2–1600-MHz CMOS Clock Recovery PLL with Low-Vdd Capability,” *IEEE Journal of Solid-State Circuits*, vol.34, no.12, pp. 1951–1999, Dec. 1999

[33] J. Lee, and M. Liu, “A 20Gb/s Burst-Mode CDR Circuit Using Injection-Locking Technique,” *IEEE International Solid-State Circuits Conference*, Feb. 2007

[34] I. Radovanovic´, W. Etten, and H. Freriks, “Ethernet-Based Passive Optical Local-Area Networks for Fiber to the Desk Application,” *IEEE Journal of Lightwave Technology,* vol.21, no.11, pp.2534-2545, Nov. 2003

[35] B. Analui and A. Hajimiri, “Instantaneous Clockless Data Recovery and Demultiplexing,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol.52, no.8, pp.437-441, Aug. 2005

[36] K. Schneider and H. Zimmermann, “Highly Sensitive Optical Receivers,” *Springer Berlin Heidelberg*, New York, 2006

[37] C.-H. Liang, S.-C. Hwu, and S.-I. Liu, "A 2.5Gbps Burst-Mode Clock and Data Recovery Circuit," *IEEE Asian Solid-State Circuits Conference*, pp.457–460, Nov. 2005

[38] P. Wallace, R. Bayruns, J. Smith, T. Laverick, and R. Shuster , "A GaAs l.SGb/s Clock Recovery and Data Retiming Circuit," *IEEE International Solid-State Circuits Conference*, Feb. 1990

[39] J. Kenney, et al., "A 9.95 to 11.1Gb/s XFP Transceiver in 0.13μm CMOS," *IEEE International Solid-State Circuits Conference*, Feb. 2006

CHAPTER 8

8 CLOCK AND DATA RECOVERY WITH ADAPTIVE LOOP GAIN FOR SPREAD SPECTRUM SERDES APPLICATIONS

A novel clock and data recovery architecture with adaptive loop gain is proposed for spread spectrum SerDes applications such as the Serial AT Attachment. The proposed design consists of a half-rate Alexander phase detector, a phase-shifting phase interpolator with a frequency differentiator and an adaptive loop gain filter. The frequency differentiator determines the clock rate difference between the referenced clock and the recovered clock. This value is then used to adjust the gain of the adaptive loop filter for better acquisition of lock with minimized jitter. The proposed design can be implemented in a digital CMOS process that reduces the design difficulty and cost. The system operation has been verified using the Cadence SpectreRF and Verilog-A simulators. The results show that the system is capable of recovering a ±5000 ppm spread spectrum data with up to a maximum of 0.5 UI of deterministic jitter.

8.1 INTRODUCTION

Clock and data recovery (CDR) has played an important role in modern high-speed serial data transmission applications such as optical communications, backplane routing and chip-to-chip interconnection. The transmitted data in a communications system are often corrupted by both external and internal noise, which leads to jitter and skew in the received data. Furthermore, high-speed serial data transmission systems are often operated in an asynchronous manner between the transmitter and receiver. Therefore, the

receiving end must have a CDR circuit to extract the clock signal from the received data. In addition, the received data must also be retimed and latched using the recovered clock. Reference [1] has discussed several of the challenges in the design of high-speed CDR circuits and architectures, including jitter, skew and acquisition of lock.

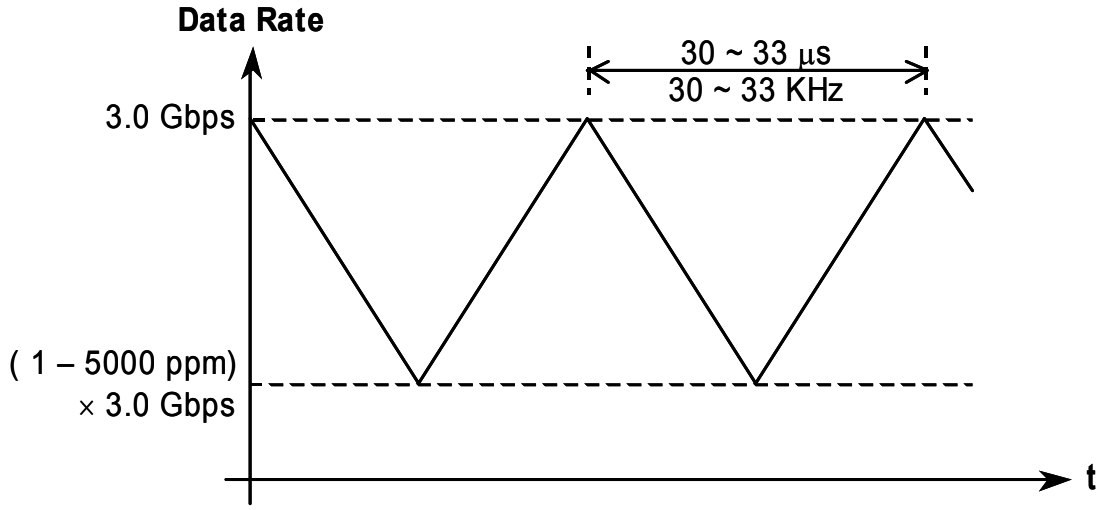


Figure 8.1 Time domain spread-spectrum clocking operation.

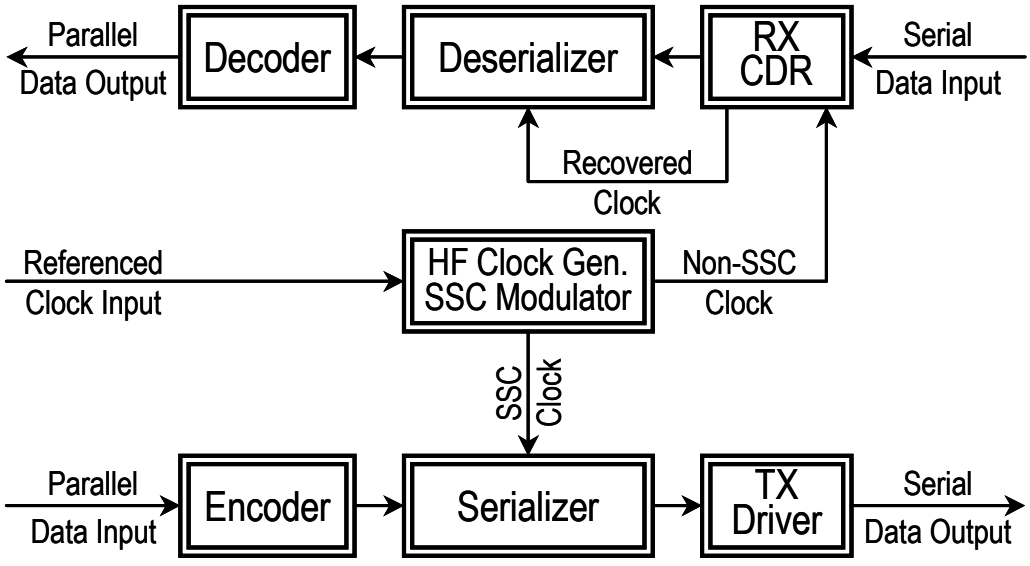


Figure 8.2 Block diagram of a spread-spectrum SerDes.

Lock acquisition for a CDR design used in spread spectrum SerDes applications such as the Serial AT Attachment (SATA) [2] is particularly challenging because the data rate varies periodically. Moreover, jitter and skew are generated during the data transmission. An example of time domain spread spectrum clocking (SSC) based on the serial-ATA Gen. II specification is shown in Figure 8.1 [2]. The purpose of this type of clocking is to spread the spectral energy and thereby reduce the effect of electromagnetic interference in the electronic system. The frequency moves between 0 ppm and –5000 ppm, which is known as a down-spread spectrum [2]. Since the period of the spread spectrum modulation is between 30 and 33 kHz, the time for acquisition of lock must be much less than 30-33 uS. This is required in order to minimize the jitter or phase noise due to the periodic variation of the data rate. A possible transceiver block diagram with spread spectrum clocking is shown in Figure 8.2. The SSC clock is used in the transmitting path and extracted in the receiving path.

8.2 CDR ARCHITECTURE

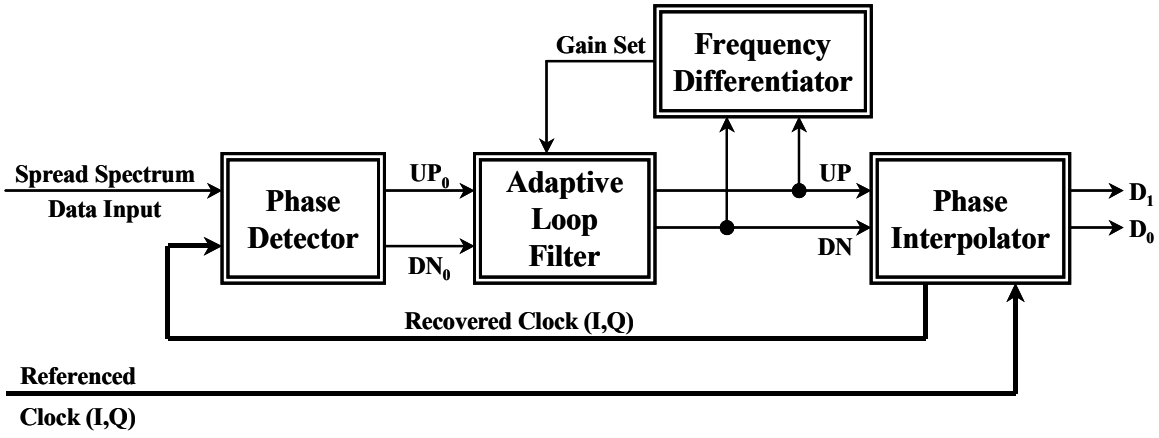


Figure 8.3 A block diagram of the proposed CDR architecture.

The proposed architecture uses a phase interpolator based CDR design including a half-rate Alexander type phase detector [3] and a phase-shifting type phase interpolator [4] with a frequency differentiator and an adaptive loop gain filter. Figure 8.3 shows the basic block diagram of the proposed CDR architecture. First, the frequency differentiator periodically determines the clock rate difference between the referenced clock and the recovered clock. This difference in clock rates is then used to periodically adjust the gain of an adaptive loop filter for better acquisition of lock. In the case of a large difference in the two clock rates, a higher loop gain is needed for quick acquisition of lock. However, for a smaller difference of clock rates, a lower loop gain is needed for minimized jitter and phase noise.

8.2.1 PHASE DETECTOR

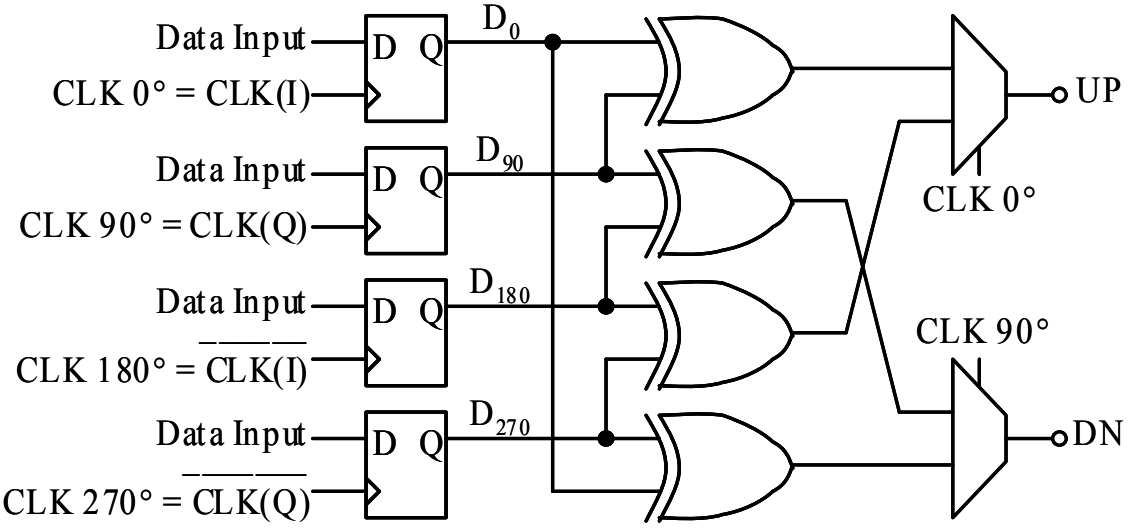


Figure 8.4 Half-rate Alexander type phase detector.

The half-rate Alexander type phase detector shown in Figure 8.4 [3] samples the data at 0°, 90°, 180° and 270° of the clock signal and produces the corresponding D_0, D_{90}, D_{180}

and D_{270} signals. The outputs UP and DN are then generated from the XOR gates and 2-to-1 multiplexers. Once the CDR is in the locked state, the quadrature clock edges are aligned with the received data transitions. Therefore, D_0 and D_{180} are the recovered, demultiplexed data streams.

8.2.2 PHASE INTERPOLATOR

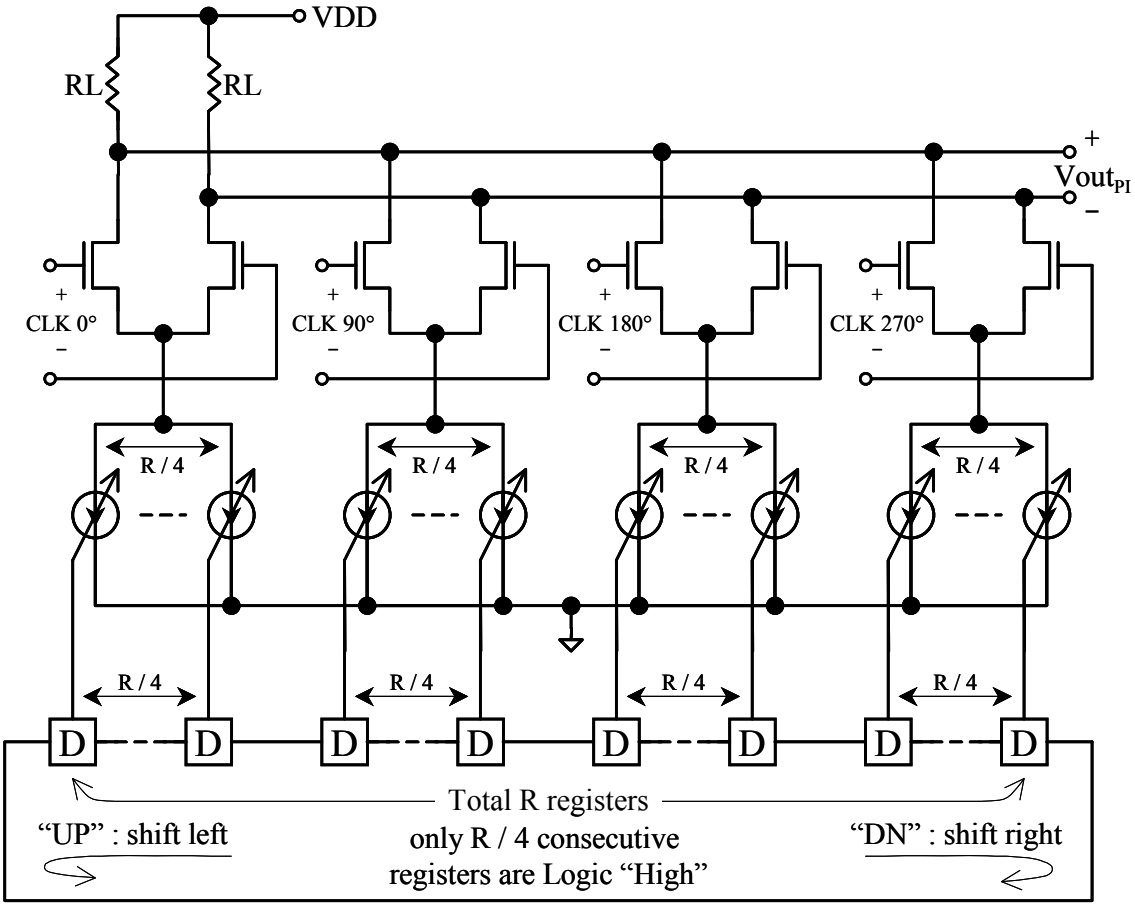


Figure 8.5 Implementation example of phase interpolator.

The phase-shifting type phase interpolator shown in Figure 8.5 [4] is used, where I_1, I_2, I_3 and I_4 are controlled by a shift register. At any given time, only one-fourth of the consecutive shift register stages have logic "high", which turns on the tail current of the

phase interpolator. For a data rate variation greater than 0 ppm, the shift register stages with logic "high" will shift to the left to ramp up the recovered clock frequency and vice-versa. The recovered clock voltage signal from the phase interpolator output is shown in Equation (8.1)

$$Vout_{PI} = RL \cdot (I_1 + I_2 + I_3 + I_4) \tag{8.1}$$

where RL is the load resistance. For every received UP pulse, phase interpolator shifts the data to the left and increases the average recovered clock frequency. For every received DN pulse, it shifts the data to the right and decreases the average recovered clock frequency.

8.2.3 FREQUENCY DIFFERENTIATOR

The frequency differentiator design is based on a digital integrator, which adds one for every UP pulse and subtracts one for every DN pulse. Assume that there are total of *R* registers in the phase interpolator. Shifting registers with logic "high" to the left *R* times in every 1000 reference clocks causes the recovered clock phase to shift down by one referenced clock period. In other words, shifting the recovered clock phase down for one referenced clock period every 1000 referenced clocks moves the average recovered clock frequency up by 1000 ppm and vice-versa. Hence, the clock rate difference between the referenced clock and the recovered clock for a given time period can be calculated as follows:

$$\Delta f(CLK_{CDR}) = \frac{f(CLK_{CDR}) - f(CLK_{REF})}{f(CLK_{REF})} = \frac{P/R}{C} \tag{8.2}$$

In this equation, *C* is the number of referenced clock periods for a given time, *P* is the total number of UP pulses minus the total number of DN pulses, *R* is the number of shift

registers stages used in the phase interpolator, $f(CLK_{CDR})$ is the frequency of the recovered clock, $f(CLK_{REF})$ is the frequency of the referenced clock and $\Delta f(CLK_{CDR})$ is the frequency variation of the recovered clock. In order to have the output of the frequency differentiator track the spread spectrum frequency variation, the system must reset the frequency differentiator periodically in a time much less than spread spectrum modulation period.

8.2.4 ADAPTIVE LOOP GAIN FILTER

The gain of the adaptive loop filter is set based on the output of the frequency differentiator. The loop filter removes some of the UP and DN pulses depending on the value of the gain setting. If the recovered clock frequency is moving up away from the reference clock frequency, the adaptive loop filter will let more UP pulses into the phase interpolator. On the other hand, if the recovered clock frequency is moving down away from referenced clock frequency, it will allow more DN pulses into the phase interpolator.

This proposed CDR architecture is designed to handle ±5000 ppm of spread spectrum data variation with a minimum of 0.5 UI deterministic jitter tolerance and a maximum of 0.1 UI internally generated jitter. In order to meet these requirements, the circuit has 80 shift register stages in the phase interpolator and seven levels of gain in the adaptive loop filter. In the half-rate Alexander type phase detector, one bit of the data period is equal to one-half of the recovered clock period. In other words, the phase step is 2 UI ÷ 80 = 0.025 UI, which can handle a maximum 4-step phase shifting due to internal jitter in the locked state and still meet the maximum 0.1 UI jitter generation requirement.

Table 8.1 Gain Setting of the Fixed Loop Filter

Fixed Loop Filter Gain	Δf(Data) Range ppm	"UP"			"DN"		
		Filter Pass/Block Ratio	Maximum Df(Clock) ppm	Maximum Phase Error ppm	Filter Pass / Block Ratio	Maximum Df(Clock) ppm	Maximum Phase Error ppm
0	+ 5000 ~ - 5000	1 / 1	+ 6250	11250	1 / 1	− 6250	11250

Table 8.2 Gain Setting of the Adaptive Loop Filter

Adaptive Loop Filter Gain	Δf(Data) Range ppm	"UP"			"DN"		
		Filter Pass/Block Ratio	Maximum Df(Clock) ppm	Maximum Phase Error ppm	Filter Pass / Block Ratio	Maximum Df(Clock) ppm	Maximum Phase Error ppm
+ 3	+ 5000 ~ + 4000	1 / 1	+ 6250	1250	1 / 14	- 833	5833
+ 2	+ 4000 ~ + 2400	2 / 3	+ 5000	1000	1 / 14	- 833	4833
+ 1	+ 2400 ~ + 800	1 / 3	+ 4167	1767	1 / 14	- 833	3233
0	+ 800 ~ - 800	1 / 4	+ 2500	1700	1 / 4	- 2500	1700
- 1	- 800 ~ - 2400	1 / 14	+ 833	3233	1 / 3	- 4167	1767
- 2	- 2400 ~ - 4000	1 / 14	+ 833	4833	2 / 3	- 5000	1000
- 3	- 4000 ~ - 5000	1 / 14	+ 833	5833	1 / 1	- 6250	1250

Table 8.1 and Table 8.2 show the input data rate variation range, the maximum recovered clock rate tracking and the phase error between data and clock for fixed and adaptive loop filters, respectively. The gain setting in the adaptive loop filter is based on the minimum required gain for the recovered clock frequency to track the spread spectrum data rate and still meet the maximum allowed internal jitter generation and minimum deterministic jitter tolerance. The maximum Δf(clock) and phase error for a given gain setting are calculated in the following equations:

$$\Delta f\,(Clock)[ppm] = \frac{S_P}{S_P + S_B} \cdot \frac{10^6}{R} \tag{8.3}$$

$$PE[ppm] = \left|\Delta f\,(Clock) - \Delta f\,(Data)\right| \tag{8.4}$$

$$PE[UI] = PE[ppm] \cdot Delay[UI] \tag{8.5}$$

where S_P and S_B are the pass and block values from Table 8.1 and Table 8.2, R is the number of shift register stages in the phase interpolator and PE[ppm] and PE[UI] are the phase error in ppm and Unit Intervals (UI), respectively. One UI is equal to one bit of the data period and Delay[UI] is the CDR loop delay in UI. For this design, the loop delay is 8 UI.

8.3 CALCULATION AND SIMULATION RESULTS

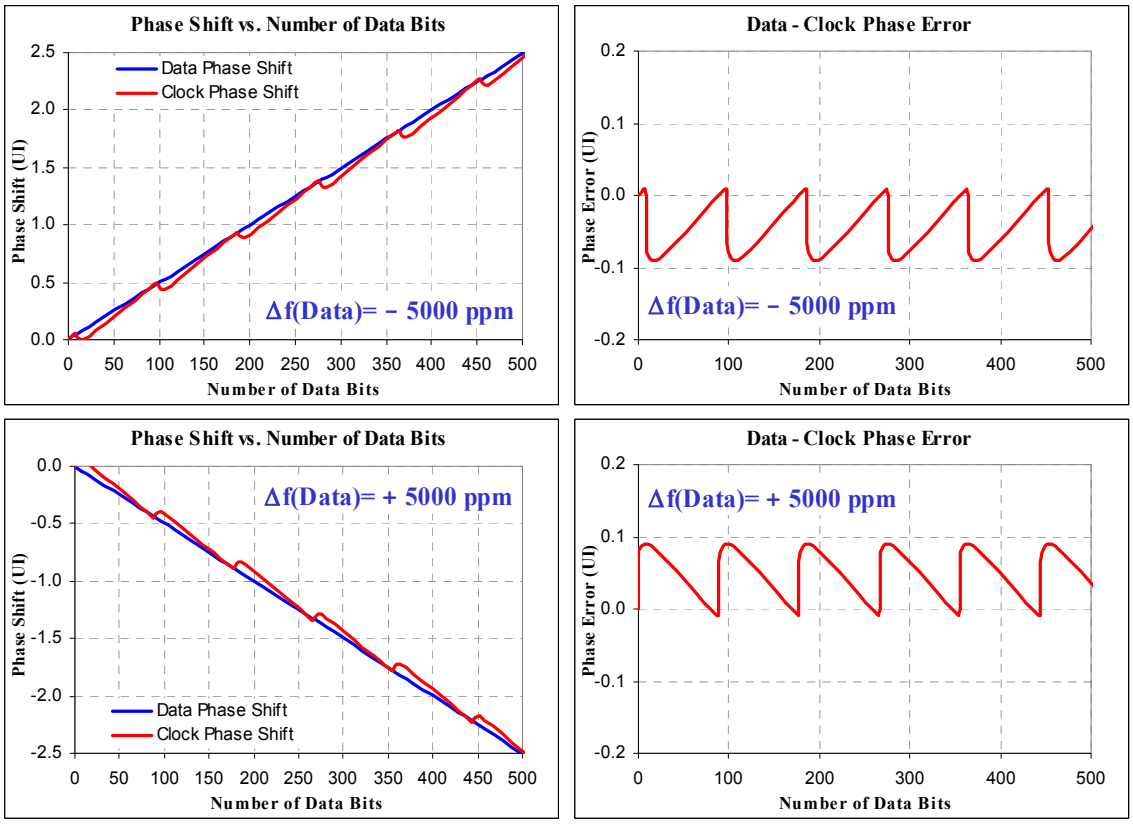


Figure 8.6 Calculated phase shift and phase error for a fixed loop gain CDR design.

Figure 8.6 and Figure 8.7 show the calculated phase shift and phase error for fixed loop gain and adaptive loop gain CDR designs, respectively. The calculation is based on the maximum data rate variation and the available maximum recovered clock frequency

tracking values of Table 8.1. Comparing the calculated phase errors between the fixed gain and adaptive gain, the fixed gain setup has twice as much phase error as the adaptive gain case. This is due to the extra UP and DOWN gain under positive and negative input data rate variations, respectively.

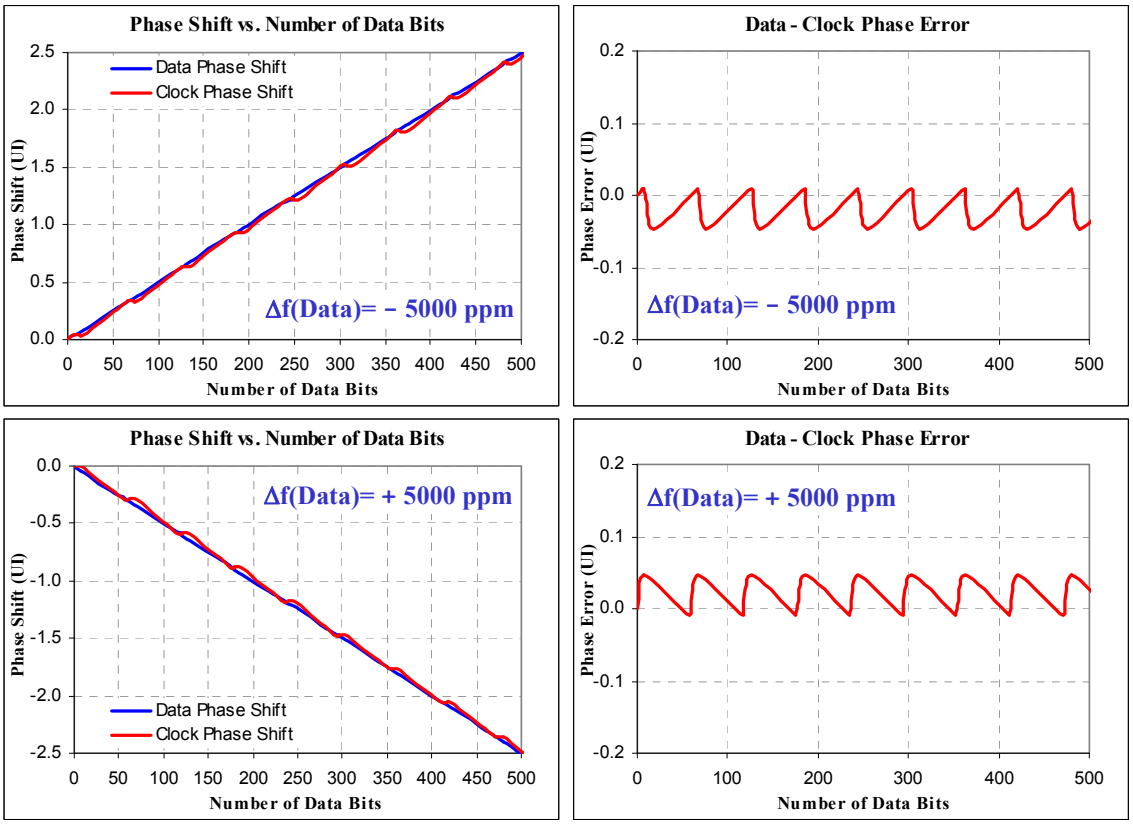


Figure 8.7 Calculated phase shift and phase error for an adaptive loop gain CDR

Figure 8.8 and Figure 8.9 show the simulation results for the phase shift and phase error in the fixed loop gain and adaptive loop gain CDR designs, respectively. The fixed gain scheme has twice as much phase error as the adaptive gain case, which was also predicted in the calculated results. The input data frequency versus time for phase shift and phase error simulations are also shown in these figures. Figure 8.9 for the adaptive loop gain CDR design shows how the internal gain tracks the input data rate variation.

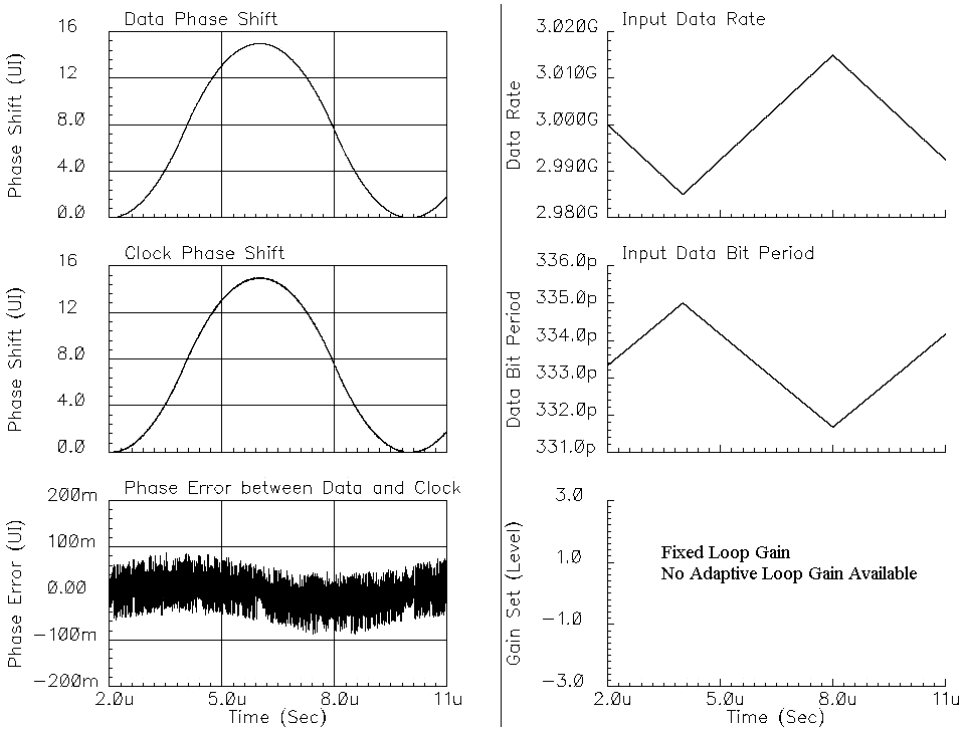


Figure 8.8 Simulated phase shift and phase error for fixed loop gain CDR

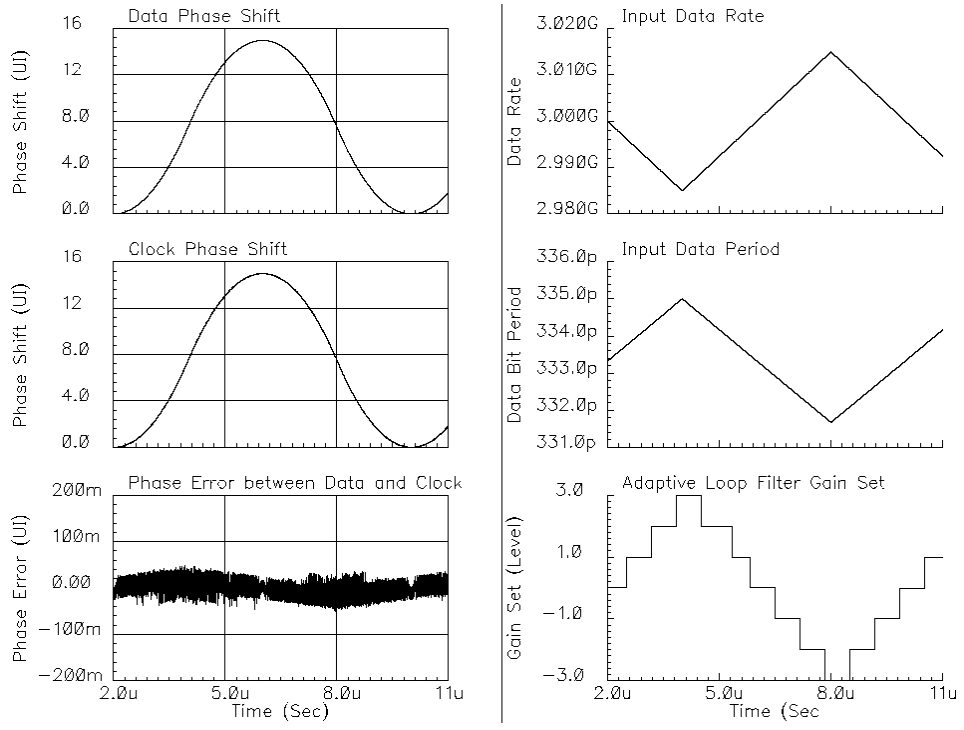


Figure 8.9 Simulated phase shift and phase error for adaptive loop gain CDR

Figure 8.10 shows eye diagrams of the input data, the retimed data, the referenced clock and the recovered clock at –3000 ppm of the normal data rate with 0.5 UI jitter associated with the data for the adaptive loop gain CDR design.

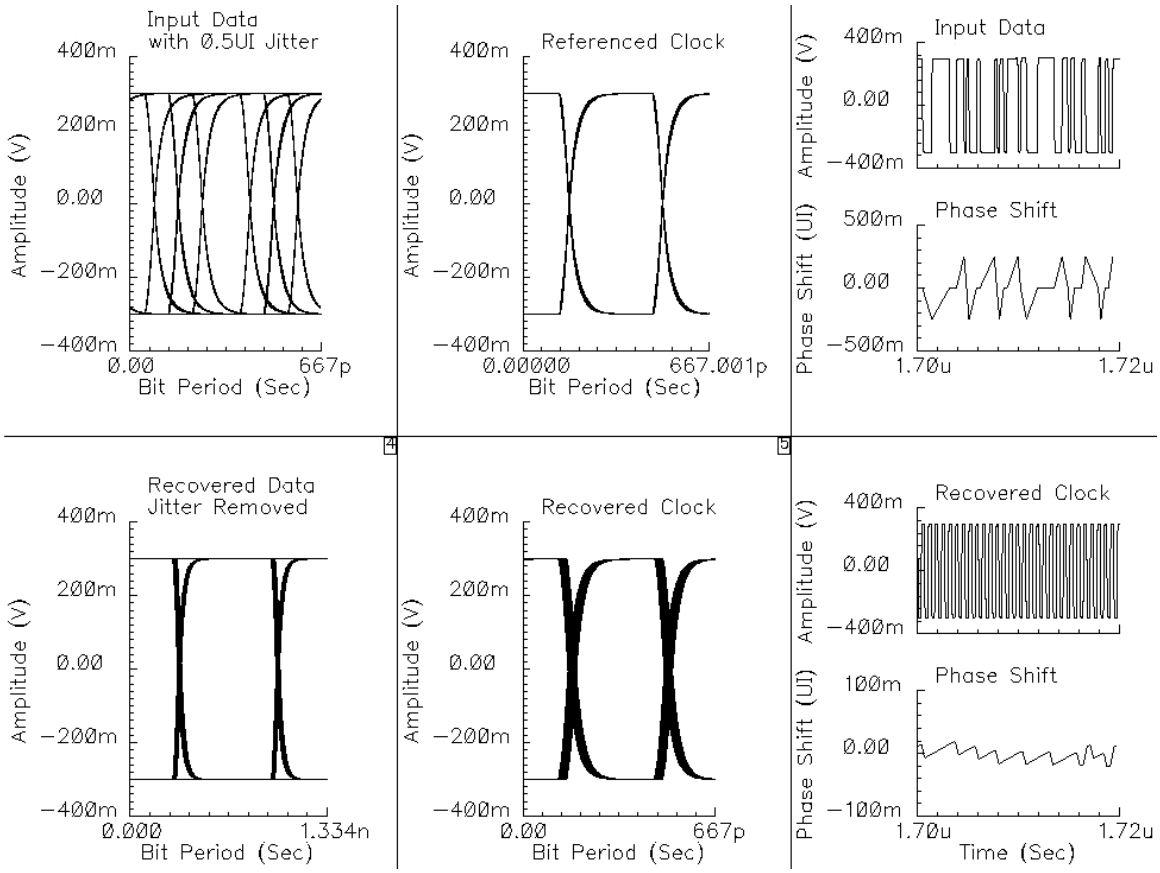


Figure 8.10 Eye diagrams of the input data, retimed data, referenced clock and recovered clock at –3000 ppm of the normal data rate with 0.5 UI jitter associated with the data for the adaptive loop gain CDR

The input data shown Figure 8.10 has deterministic jitter jumping around at the ±0.25 UI and 0 UI points for the 3.0 Gbps data rate. The retimed data is one of two demultiplexed data streams from the phase detector and has a data rate 1.5 Gbps. The simulation results show the removal of deterministic jitter associated with the input data. The only residue jitter in the retimed data is the CDR internally generated jitter from the

tracking phase error between the input data and the recovered clock.

8.4 CONCLUSION

A requirement for spread-spectrum clocking exists in some SerDes applications such as the Serial AT Attachment. This reduces the effect of electromagnetic interference in the electronic system but presents challenges in the CDR design. The proposed phase interpolator based CDR with adaptive loop gain fulfills the need for better acquisition of lock with only a small amount of internally generated jitter. Both calculated and simulated results shows that the use of adaptive gain in the CDR design eliminates half of the internally generated jitter from the tracking phase error between the input data and the recovered clock and still meets the requirement for high deterministic jitter tolerance.

REFERENCES

[1] B. Razavi, “Challenges in the Design of High-Speed Clock Data Recovery Circuit,” *IEEE Communication Magazine*, August 2002, pp 94-101.

[2] Serial ATA Working Group, “Serial ATA II: Electrical Specification, Revision 1.0”, *Serial ATA Working Group* http://www.sata-io.org, May 2004.

[3] A. Rezayee, and K. Martin, "A 9-16Gb/s Clock and Data Recovery Circuit with Threestate Phase Detector and Dual-Path Loop Architecture," *Proceedings of the 2003 European Solid-State Circuits Conference*, Estoril, Portugal, pp. 683-686.

[4] R. Kreienkamp, U. Langmann, C. Zimmermann, and T. Aoyama, “A 10-Gb/s CMOS clock and data recovery circuit with an analog phase interpolator,” *Proceedings of the IEEE 2003 Custom Integrated Circuits Conference*, pp. 73-76.

CHAPTER 9

9 CONCLUSION AND FUTURE WORK

This dissertation demonstrates design procedures of a serial link from the top system level, using modeling and performance estimation, to the bottom circuit level with detailed functional design. A serial link design is divided into three major subsystems which consist of the phase locked loop (PLL) based clock generator / synthesizer with optional spread spectrum clocking (SSC), the transmitter with pre-emphasis and the receiver with equalizer and clock data recovery (CDR). The ultimate goal of this serial link study was to design a single-chip, multi-channel, serial link system with multiple clock generators / synthesizers for concurrent, multi-rate, multi-standard, multi-channel applications.

9.1 SINGLE CHIP MULTI-CHANNEL SERIAL LINK

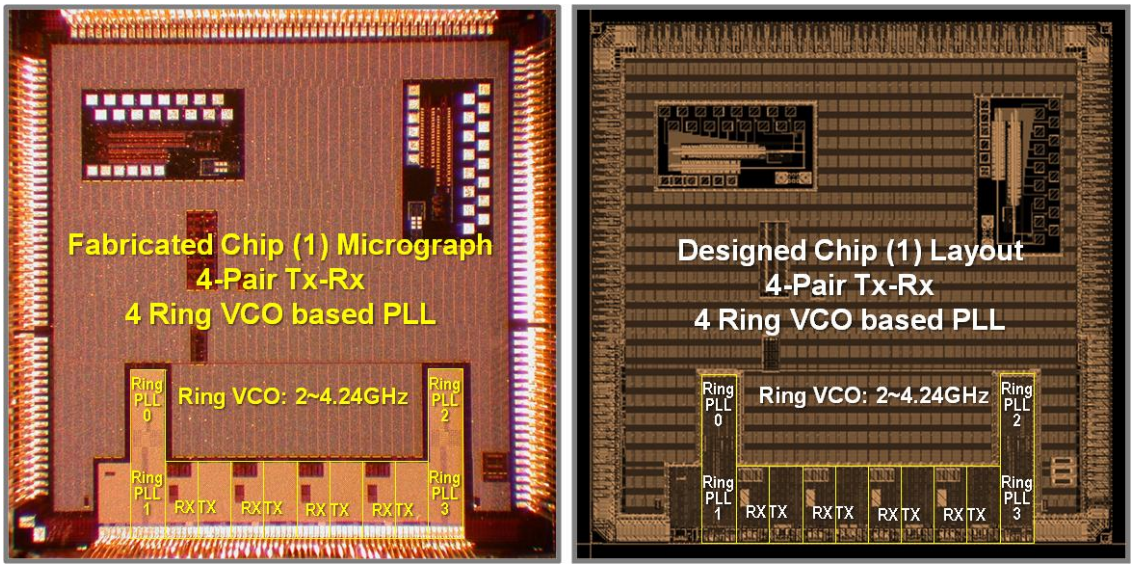


Figure 9.1 The 1st designed single chip serial link system.

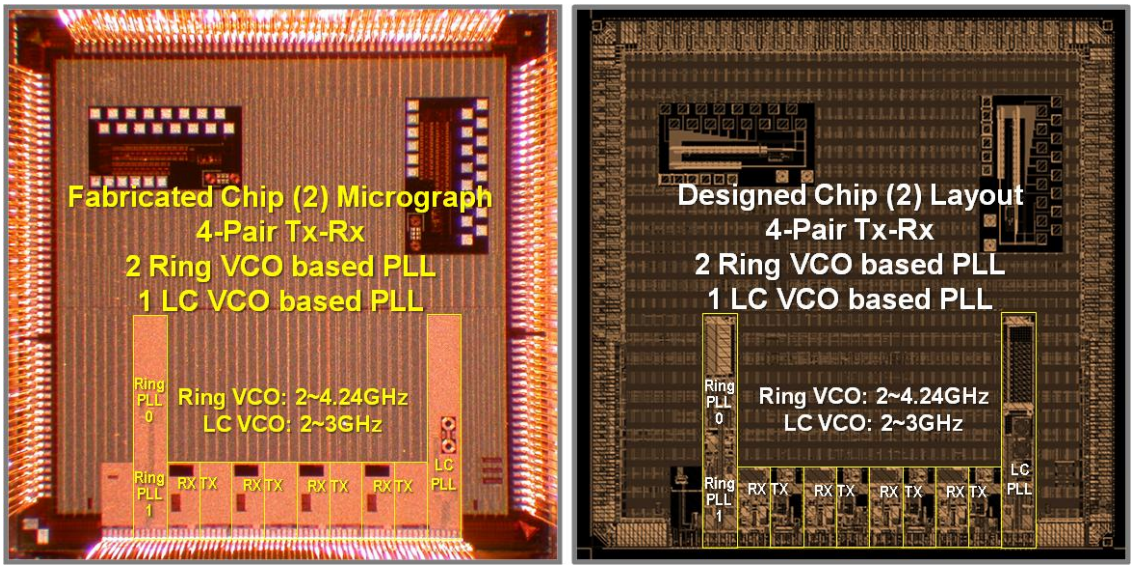


Figure 9.2 The 2nd designed single chip serial link system.

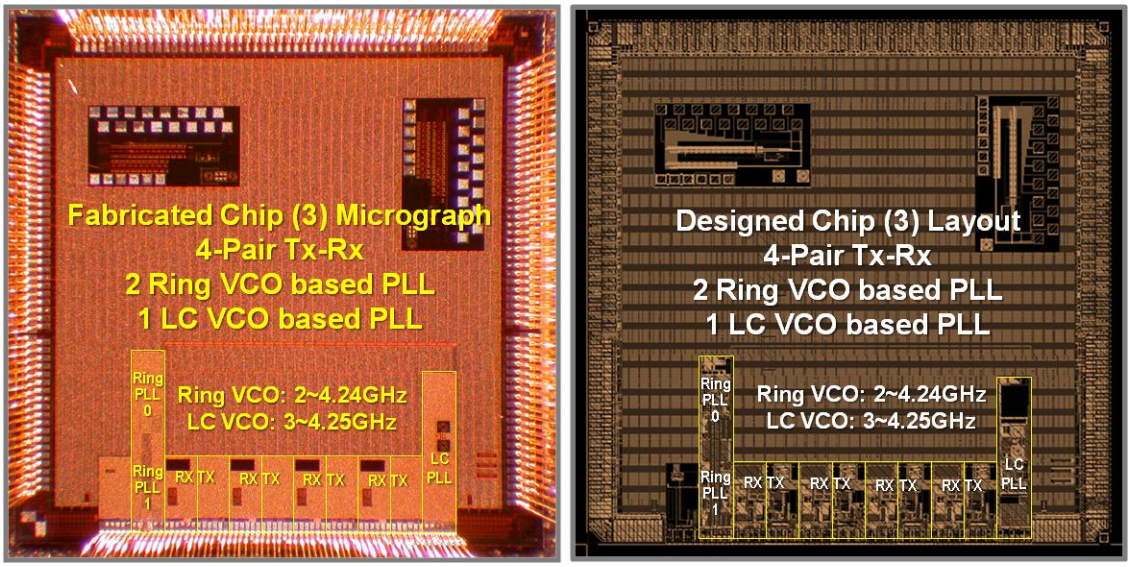


Figure 9.3 The 3rd designed single chip serial link system.

Three chips, as shown in Figure 9.3, were designed and fabricated in a standard 1.2V digital 90 nm CMOS process during the course of this research. All designed serial link systems consist of 4 pairs of transmitters (Tx) and receivers (Rx) to support multiple transmission channels in a single chip. The 1st designed chip, as shown in Figure 9.1,

includes 4 PLL based clock generators / synthesizers. The PLL design is based on a ring type of voltage controlled oscillator (Ring VCO), which has been extensively discussed in Chapter 4. The supported frequency range of the Ring VCO based PLL is 2~4.25 GHz. Both the 2nd and 3rd designed chips, as shown in Figure 9.2 and Figure 9.3, consist of 2 Ring VCO based PLLs and 1 LC VCO based PLL as the clock generators / synthesizers. The supported frequency ranges of the Ring VCO based PLL in the 2nd and 3rd designed chips are same as the ones in the 1st chip. However, the supported frequency ranges of LC VCO based PLLs are 2~3GHz and 3~4.25 GHz for the 2nd and 3rd designed chips, respectively. The primary goal of the 2nd and 3rd designed chips was to support a continuous 2~4.25 GHz frequency range for future single-chip serial link system designs having ultra low clock jitter. A second purpose was to use them as vehicles for performing the power, area and performance study between Ring and LC VCO based PLL discussed in Chapter 4. The main applications and general features of this designed serial link system are:

(a). Serial data rate from 1 Gbps to 8.5 Gbps.

(b). Supporting concurrent multi-rate, multi-standard, multi-channel operation.

(c). Reconfigurable total number of Tx-Rx transmission channels and clock generator.

(d). Compliance with Serial AT Attachment (SATA) [1], Serial Attached SCSI (SAS) [2], Peripheral Component Interconnect Express (PCIe) [3] and Fibre Channel (FC) [4] standards.

(e). Parallel data bus widths of 8, 10, 16 and 20 bits with supported parallel bus clock rates up to 425 MHz.

(f). Supporting spread spectrum clocking (SSC) with -2300ppm~+2300ppm (center

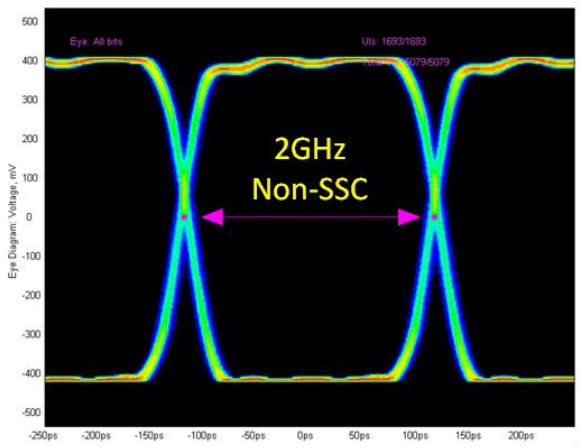
spread), 0~±4600ppm (up or down spread) or 0~±2300ppm (up or down spread) for EMI reduction.

(g). Hot-pluggable Tx-Rx design.

(h). Adjustable Tx driver output amplitude for a variety of compliance requirements.

(i). Tunable Tx pre-emphasis and Rx equalization for compensating different transmission channel losses.

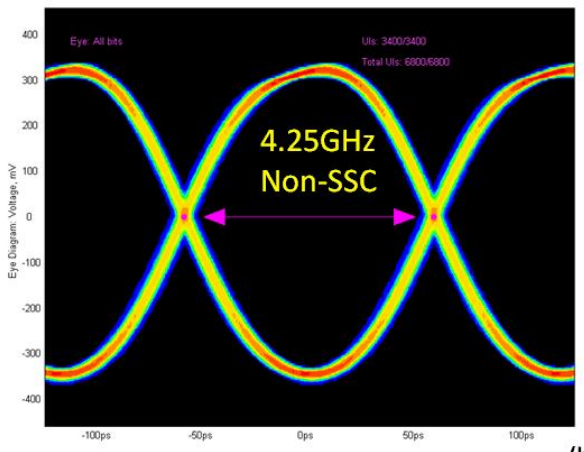
(j). Built-in pseudo random binary sequence (PRBS) pattern generator and pattern checker for Tx-Rx internal and external loopback built-in-self-test (BIST).



2GHz, Non-SSC	Current Acq	Averaged
Random (RMS)	1.9729ps	1.9172ps
Deterministic (Pk-Pk)	12.327ps	11.622ps
Periodic	9.1260ps	8.4652ps
Duty Cycle	3.2009ps	3.1572ps
Data Dependent (ISI)	0.00s	0.00s
Total @ BER (Pk-Pk)	35.589ps	34.326ps
Eye Opening @ BER	825.78mui	831.97mui

2GHz, SSC+/-2300ppm	Current Acq	Averaged
Random (RMS)	2.1522ps	2.1896ps
Deterministic (Pk-Pk)	11.736ps	17.304ps
Periodic	8.0768ps	13.695ps
Duty Cycle	3.6594ps	3.6081ps
Data Dependent (ISI)	0.00s	0.00s
Total @ BER (Pk-Pk)	37.657ps	43.493ps
Eye Opening @ BER	815.68mui	786.78mui

(a)



4.25GHz, Non-SSC	Current Acq	Averaged
Random (RMS)	976.70fs	1.0112ps
Deterministic (Pk-Pk)	10.330ps	10.812ps
Periodic	5.9373ps	6.5289ps
Duty Cycle	4.3928ps	4.2836ps
Data Dependent (ISI)	0.00s	0.00s
Total @ BER (Pk-Pk)	22.196ps	23.052ps
Eye Opening @ BER	811.33mui	804.06mui

4.25GHz, SSC+/-2300ppm	Current Acq	Averaged
Random (RMS)	1.1093ps	1.1128ps
Deterministic (Pk-Pk)	12.986ps	12.515ps
Periodic	9.0369ps	8.4097ps
Duty Cycle	3.9493ps	4.1052ps
Data Dependent (ISI)	0.00s	0.00s
Total @ BER (Pk-Pk)	26.214ps	25.840ps
Eye Opening @ BER	777.60mui	780.47mui

(b)

Figure 9.4 Measured clock eye-diagram and jitter performance of the designed Ring VCO based PLL clock generator / synthesize output at 2 GHz and 4.25 GHz.

Figure 9.4 shows examples of measured clock eye-diagram and jitter performance of the designed Ring VCO based PLL clock generator / synthesizer output at 2 GHz and 4.25 GHz. The measured jitter performance results of both non-SSC and SSC clock generation indicate that the degradation of jitter performance caused by SSC is small.

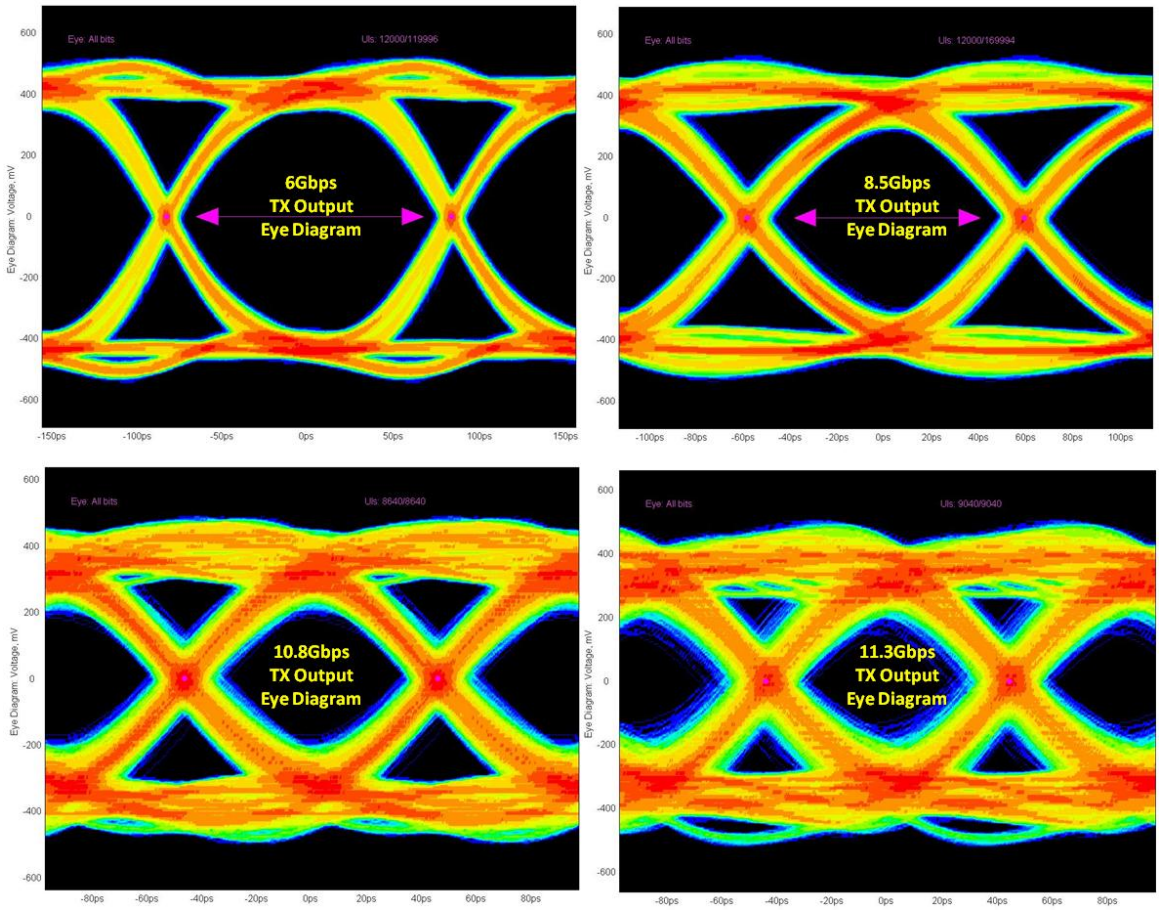


Figure 9.5 Measured Tx driver output data signal eye-diagram at 6 Gbps, 8.5 Gbps, 10.8 Gbps and 11.3Gbps.

Figure 9.5 shows the data signal eye-diagrams from the transmitter driver output of the designed single chip serial link system at 6 Gbps, 8 Gbps, 10.8 Gbps and 11.3 Gbps. The diagrams at 10.8 Gbps and 11.3 Gbps indicate a possible bandwidth limitation at the transmitter driver output. Table 9.1 shows the designed serial link transmitter jitter

generation measurement results in unit intervals (UI) at 8.5Gbps, where 1 UI=117.65 pS is one bit period at 8.5 Gbps, and where DJ is deterministic jitter, TJ is the total jitter and RJ is random jitter. RJ is mainly from the clock generator / synthesizer.

Table 9.1 Designed serial link transmitter jitter generation measurement results.

Process / Temperature / Supply / Measurement	Slow				Typical				Fast			
	125°C		0°C		125°C		0°C		125°C		0°C	
	1.14V	1.26V	1.14V	1.26V	1.14V	1.26V	1.14V	1.26V	1.14V	1.26V	1.14V	1.26V
Eye H: All Bits (mV)	759	810	760	849	807	887	851	910	870	927	898	1004
Eye H: Non-Tr Bits (mV)	885	932	820	884	959	1018	929	1011	1094	1172	1063	1145
Eye H: Trans Bits (mV)	759	810	760	849	807	887	851	910	870	927	898	1004
Eye Width (UI)	0.77	0.80	0.70	0.75	0.81	0.84	0.81	0.83	0.80	0.80	0.71	0.73
Jitter: Eye Opening (UI)	0.69	0.70	0.61	0.66	0.75	0.76	0.74	0.75	0.71	0.72	0.61	0.64
Jitter: Determ (DJ) (UI)	0.22	0.18	0.28	0.22	0.17	0.15	0.18	0.15	0.19	0.17	0.26	0.22
Jitter: Total (TJ) (UI)	0.31	0.30	0.39	0.34	0.25	0.24	0.26	0.25	0.29	0.28	0.39	0.36

Table 9.2 Designed serial link receiver jitter tolerance measurement results.

Wafer Corner	VDD 1.2 (V)	Jitter Tolerance (UI)	
		0(°C)	125(°C)
Typical	1.14	0.43	0.45
	1.26	0.43	0.45
Slow	1.14	0.45	0.35
	1.26	0.50	0.40
Fast	1.14	0.40	0.35
	1.26	0.50	0.45

The receiver performance evaluation of the designed serial link system is done through a jitter tolerance test, which has been extensively discussed in Chapter 3. The approach uses a bit error rate tester (BERT) scan method[5], which is a jitter characterization method based on scanning the bit error rate within the eye diagram. Table 9.1 shows the summary of the receiver jitter tolerance measurements at 8.5 Gbps.

All of the measurements are done at a bit error rate (BER) of 1E-12. The package design is based on a 2-layer plastic ball grid array (PBGA) package. The experimental

results shown in the above plots and tables demonstrate the success of the designed single chip serial link system for multi-rate, multi-standard, multi-channel **applications**.

9.2 RESEARCH SUMMARY AND FUTURE WORK

Three primary types of high-speed serial links are chip-to-chip (for short link distance and mostly synchronous communication), board-to-board (for moderate link distance and mostly asynchronous communication), and chassis-to-chassis (for long link distance and mostly asynchronous communication). This research falls into the last category. A single-chip, multi-channel design satisfying multi-standard requirements was implemented. Using one design for multiple standards eliminates the cost of multiple design development and multiple chip inventory. A System-on-Chip with multi-channel operation minimizes the system-level design size and cost.

The goals accomplished within this research on high-speed serial links are as follows: (i) creating a procedure for top-down design and bottom-up verification for high-speed serial link development; (ii) developing system-level performance prediction with virtual built-in-self-test (VBIST); (iii) categorizing applicable high-speed clock and data recovery architectures; (iv) designing a clock and data recovery circuit having a large frequency offset tolerance for spread spectrum clocking; (v) creating a new architecture for a spread spectrum clock generator; and (vi) designing a single-chip serial link for multi-channel and multi-standard (SATA, SAS, PCIe and FC) applications at rates up to 8.5 Gbps in a 90 nm digital CMOS technology.

The major factors limiting the performance of high-speed serial links are the timing uncertainty (jitter), on-chip clock frequency limitations, and most importantly the transmission channel bandwidth. Without modifying the transmission channel

characteristics, design optimization and tuning of pre-emphasis at the transmitter and equalization at the receiver will continue to be the best solution to increase the Rx CDR jitter tolerance performance, which, in turn, increases the maximum allowable data transmission rate and/or length of the serial link system.

A possible future research direction is to explore the use of adaptive equalization, such as combining traditional feed-forward equalization (FFE) with a decision feedback equalizer (DFE) to minimize DJ such as inter-symbol-interference (ISI) jitter. RJ is usually non-removable noise in a serial link system. Therefore, the goal of future work is to focus on removing or minimizing systematic patterned noise DJ in order to maximize serial link system performance.

REFERENCES

[1] Serial ATA Workgroup, "Serial ATA," Revision 2.6, May 2007.

[2] INCITS, "Serial Attached SCSI – 2 (SAS-2)," Revision 14a, May 2008.

[3] PCI SIG, "PCI Express Jitter White Paper - PCI Express Jitter Modeling, Revision 1.0RD," *PCI SIG,* http://www.pcisig.com , Jul. 14, 2004.

[4] American National Standards Institute, "Fibre Channel Physical Interface-4 (FC-PI-4) Rev. 6.10," *American National Standards Institute*, Feb. 27, 2007.

[5] M.A. Kossel and M.L. Schmatz, "Jitter Measurements of High-Speed Serial Links," *IEEE Design and Test of Computers*, Volume 21, Issue 6, pp. 536–543, November 2004.

BIBLIOGRAPHY

[Ch1.1] D. A. Sanders, et al., "Redundant Arrays of IDE Drives," *IEEE Transactions on Nuclear Science*, Vol.49, No.4, pp.1834-1840, August 2003.

[Ch1.2] H.-K. Kim, et al., "A Single-Chip Storage LSI for Home Networks," *IEEE Communication Magazine*, Vol.43, Issue.5, pp.141-148, May 2005.

[Ch1.3] IEEE-SA Standards Board, "IEEE Standard for Data Delivery and Logical Channels for IEEE 1284 Interfaces," *The Institute of Electrical and Electronics Engineers, Inc.*, June 2000.

[Ch1.4] A. Depari, et al., "USB Sensor Network for Industrial Applications," *IEEE Transaction on Instrumentation and Measurement,* Vol.57, no.7, pp.1344-1349, July 2008.

[Ch1.5] Serial ATA Workgroup, "Serial ATA," Revision 2.6, May 2007.

[Ch1.6] M. Sugawara et al., "1.5 Gbps, 5150 ppm SPREAD Spectrum SerDes PHY with a 0.3mW, 1.5 Gbps Level Detector for Serial ATA," *Dig. Sym. VLSI Circuits*, pp. 60-63, Jun. 2002.

[Ch2.1] R. Sommer et. al., "From System Specification to Layout: Seamless Top-Down Design Methods for Analog and Mixed-Signal Applications," *IEEE Proceedings 2002 Design, Automation and Test in Europe Conference and Exhibition,* pp.884-891, March 2002

[Ch2.2] P. Muller, A. Tajalli, M. Atarodi, and Y. Leblebici, "Top-down design of a low-power multi-channel 2.5-Gbit/s/channel gated oscillator clock-recovery circuit," *IEEE Proceedings 2005 Design, Automation and Test in Europe Conference and Exhibition,* Vol.1., pp.258-263, May 2005

[Ch2.3] K. Kundert, "Principles of Top-Down Mixed Signal Design," The Designer's Guide Community http://www.designers-guide.org, 2003

[Ch2.4] M. Wang, H. Maramis, D. Telian, and K. Chung, " New Techniques for Designing and Analyzing Multi-GigaHertz Serial Links," *DesignCon* 2005.

[Ch2.5] C. Skach, "Signals predict Serdes jitter behavior," *EE Times* Asia, http:// www.eetasia.com, May 16, 2005.

[Ch2.6] P.K. Hanumolu, B. Casper, R. Mooney, G. Wei, and U. Moon, "Analysis of PLL clock jitter in high-speed serial links," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, Vol.50, No.11, November 2003, pp. 879-886.

[Ch2.7] M.A. Kossel and M.L. Schmatz, "Jitter Measurements of High-Speed Serial Links," IEEE Design and Test of Computers, Volume 21, Issue 6, pp. 536–543, No-vember 2004

[Ch2.8] Lattice Semiconductor Corporation "Technical Note TN1084: SERDES Jitter," *Lattice Semiconductor Corpora*tion, March 2, 2005.

[Ch2.9] M. Hsieh and G. Sobelman, "Accurate Prediction of Jitter Tolerance in High-Speed Serial Links," *International SoC Design Conference*, 2005.

[Ch2.10] B. Antao, F. El-Turky, and R. Leonowich, “Mixed-mode simulation of phase-locked loops,” *IEEE 1993 Custom Integrated Circuits Conference*, 9-12 May 1993, pp. 8.4.1 - 8.4.4

[Ch2.11] A. Rezayee and K. Martin, "A 9-16Gb/s Clock and Data Recovery Circuit with Three state Phase Detector and Dual-Path Loop Architecture," *Proceedings of the 2003 European Solid-State Circuits Conference*, Estoril, Portugal, pp. 683-686.

[Ch3.1] J. Caroselli, “High-Speed Transceivers Require Systems Modeling,” *EE Times* http://www.eetimes.com, November 7, 2003.

[Ch3.2] M. Li, “Statistical and System Transfer Function based Method for Jitter and Noise in Communication Design and Test,” *DesignCon 2005.*

[Ch3.3] Lattice Semiconductor Corporation, Technical Note TN1084: “SERDES Jitter,” Lattice Semiconductor Corporation, March 2, 2005.

[Ch3.4] N. Ou, et al., “Jitter Models for the Design and Test of Gbps-Speed Serial Interconnects,” *IEEE Design and Test of Computers*, vol.21, no.4, pp. 302-313, July/August 2004.

[Ch3.5] M.A. Kossel and M.L. Schmatz, “Jitter Measurements of High-Speed Serial Links,” *IEEE Design and Test of Computers*, Volume 21, Issue 6, pp. 536 – 543, November 2004.

[Ch3.6] M. Mueller et al., “Total Jitter Measurement at Low Probability Levels, Using Optimized BERT Scan Method,” *DesignCon 2005.*

[Ch3.7] V. Stojanovic and M. Horowitz, “Modeling and Analysis of High Speed Links,” *IEEE 2003 Custom Integrated Circuits Conference*, September 2003.

[Ch3.8] A. Rezayee, and K. Martin, "A 9-16Gb/s Clock and Data Recovery Circuit with Three state Phase Detector and Dual-Path Loop Architecture," *Proceedings of the 2003 European Solid-State Circuits Conference*, Estoril, Portugal, pp. 683-686.

[Ch4.1] J. G. Maneatis, “Selecting PLLs for ASIC Applications Requires Tradeoffs,” *2004 FSA Semiconductor IP Workshop*, Oct. 7, 2004

[Ch4.2] D. Fischette, “Practical Phase-Locked Loop Design,” *IEEE Journal of Solid-State Circuits Tutorial,* http://www.delroy.com/pll, 2004

[Ch4.3] J. Savoj, and B. Razavi, “High-Speed CMOS Circuits for Optical receivers,” *Kluwer Academic Publisher,* Boston, 2001

[Ch4.4] B. Razavi, “Design of Integrated Circuits for Optical Communications,” *McGraw-Hill*, New York, 2003

[Ch4.5] T. Miyazaki, M. Hashimoto, and H. Onodera, “A Performance Comparison of PLLs for Clock Generation Using Ring Oscillator VCO and LC Oscillator in a Digital CMOS Process,” *IEEE 2004 Asia and South Pacific Design Automation Conference,* pp.545-546, Jan. 2004

[Ch4.6] D. Ham and A. Harjimiri, “Concepts and Method in Optimization of Integrated LC VCOs,” *IEEE Journal of Solid-State Circuits*, vol. 36, no.6, pp.896-909, Jul. 2001

[Ch4.7] A. Jerng and C.G. Sodini, “The Impact of Device Type and Sizing on Phase Noise Mechanisms,” *IEEE Journal of Solid-State Circuits*, vol. 40, no.2,

pp.360-369, Jul. 2001

[Ch4.8] X. He, W. Kong, T. Firestone, R. Newcomb, and M. Peckerar, "Phase Noise Optimization of A Symmetric CMOS LC VCO," *IEEE 2006 International Symposium on Industrial Electronics,* vol.4, pp.2820-2823, Jul. 2006

[Ch4.9] R. Navid, T. H. Lee, and R.W. Dutton, "Minimum Achievable Phase Noise of RC Oscillators," *IEEE Journal of Solid-State Circuits*, vol.40, no.3, pp.630-637, Mar. 2005

[Ch4.10] L. Dai, and R. Harjani, "Design of Low-Phase-Noise CMOS Ring Oscillators," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol.49, no.5, pp.328-338, May 2002

[Ch4.11] A.A. Abidi, "Phase Noise and Jitter in CMOS Ring Oscillators," *IEEE Journal of Solid-State Circuits*, vol.40, no.3, pp.1803-1816, Aug. 2006

[Ch4.12] K. Lim, C.-H. Park, A.-S. Kim and B. Kim, "A Low-Noise Phase-Locked Loop Design by Bandwidth Optimization," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 6, pp.807-815, Jun. 2000

[Ch4.13] M. Mansuri and C.-K. Ken, "Jitter Optimization based on Phase-Locked Loop Design Parameters," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 11, pp.1375-1382, Nov. 2002

[Ch4.14] S. Dosho, N. Yanagisawa, and A. Matsuzawa, "A Background Optimization Method for PLL by Measuring Phase Jitter Performance," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 4, pp.941-950, Apr. 2005

[Ch4.15] M.-t. Hsieh and G. E. Sobelman, "Comparison of LC and Ring VCOs for PLLs in a 90 nm Digital CMOS Process," *Proceedings, International SoC Design Conference,* pp.19-22, 2006

[Ch4.16] Z. Wang, "An Analysis of Charge-Pump Phase-Locked Loops," *IEEE Transaction on Circuits and System – 1: Regular Paper,* vol.52, no.10, pp.2128-2138, Oct. 2005

[Ch4.17] P.K. Hanumolu, M. Brownlee, K. Mayaram, and U.-K. Moon, "Analysis of Charge-Pump Phase Locked Loops," *IEEE Transaction on Circuits and System – 1: Regular Paper,* vol.51, no.9, pp.1665-1674, Sept. 2004

[Ch4.18] F. You and S. He, " Analysis of Third-Order Charge Pump PLL," *IEEE International Conference on Communications, Circuits and Systems*, vol.2, pp.1372-1376, 2004

[Ch4.19] A.L.S. Loke, et al, "A Versatile 90-nm CMOS Charge-Pump PLL for SerDes Transmitter Clocking," *IEEE Journal of Solid-State Circuits*, vol.41, no.8, pp.1894-1907, Aug. 2006

[Ch4.20] A.Y.V. Lopez, "Design of Frequency Synthesizer for Short Range Wireless Transceivers," *Ph.D. Thesis, Texas A&M University,* May 2004

[Ch4.21] B. Razavi, "Design of Monolithic Phase-Locked Loops and Clock Recovery Circuits: A Tutorial," *In Monolithic Phase-Locked Loops and Clock Recovery Circuits. Theory and Design (B. Razavi, ed.)*, *IEEE Press*, New York, pp.1-39, 1996

[Ch4.22] M.H. Perrott, "Design and Simulation of Fractional-N Frequency Synthesizers: A Tutorial," *In Analog Circuit Design Fractional-N Synthesizers, Design for*

Robustness, Line and Bus Drivers (A.H. M. van Roermund, M. Steyaert, J.H. Huijsing, ed.), *Kluwer Academic Publisher*, Norwell, pp.27-49, 2003

[Ch4.23] A. Schultz, "Selecting Passive Components for PLLs," *Electronics Design, Strategy, News*, http://www.edn.com, Mar. 20, 2003.

[Ch4.24] M. Rachedine, et al, "Performance Review of Integrated CMOS VCO Circuits for Wireless Communications," *IEEE Radio Frequency Integrated Circuits (RFIC) Symposium,* pp.77-80, Jun. 2003

[Ch4.25] A. Fard, T. Johnson, M. Linder, and D. Aberg, "A Comparative Study of CMOS LC VCO Topologies for Wide-Band Multi-Standard Transceivers," *IEEE Midwest Symposium on Circuits and Systems,* vol.3, pp.17-20, Jul. 2004

[Ch4.26] Y. Park, S. Chakraborty, C.-H. Lee, S. Nuttinck, and J. Laskar, "Wide-Band CMOS VCO and Frequency Divider Design for Quadrature Signal Generation," *IEEE MTT-S International Microwave Symposium Digest*, vol.3, pp.1493-1496, Jun. 2004

[Ch4.27] Z. Wang, H.S. Savci, and N.S. Dogan, "1-V Ultra-Low-Power CMOS LC VCO for UHF Quadrature Signal Generation," *IEEE International Symposium on Circuits and Systems*, pp.4022-4025, May 2006

[Ch4.28] A. Hajimiri and T.H. Lee, "Design Issues in CMOS Differential LC Oscillators," *IEEE Journal of Solid State Circuits*, vol.32, no.5, pp.717-724, May 1999

[Ch4.29] G. De Astis, D. Cordeau, J.-M. Paillot, and L. Dascalescu, "A 5-GHz Fully Integrated Full PMOS Low-Phase-Noise LC VCO," *IEEE Journal of Solid State Circuits*, vol.40, no.10, pp.2087-2091, Oct. 2005

[Ch4.30] K.K. O, N. Park, and D-J. Yang, "*1/f* Noise of NMOS and PMOS Transistors and their Implications to Design of Voltage Controlled Oscillators," *IEEE Radio Frequency Integrated Circuits (RFIC) Symposium*, pp.59-62, Jun. 2002

[Ch4.31] M. Thamsirianunt and T.A. Kwasniewski, "CMOS VCO for PLL Frequency Synthesis in GHz Digital Mobile Radio Communications," *IEEE Journal of Solid State Circuits*, vol.32, no.10, pp.1511-1524, Oct. 1997

[Ch4.32] R.J. Betancourt-Zamora and T.H. Lee, "CMOS VCOs for Frequency Synthesis in Wireless Biotelemetry," *IEEE 1998 International Symposium on Low Power Electronics and Design,* Aug. 1998

[Ch4.33] L. Dai and R. Harjani, "Comparison and Analysis of Phase Noise in Ring Oscillators," *IEEE International Symposium on Circuits and Systems,* 2000

[Ch4.34] D.A. Badillo and S. Kiaei, "Comparison of Contemporary CMOS Ring Oscillators," *IEEE Radio Frequency Integrated Circuits (RFIC) Symposium*, pp.281-284, Jun. 2004

[Ch4.35] M. Horowitz, et al., "PLL design for a 500 MB/s interface, *IEEE International Solid-State Circuits Conference*, pp.160-161 and 282, Feb. 1993

[Ch4.36] L. Dai and R. Harjani, "A Low-Phase-Noise CMOS Ring Oscillator with Differential Control and Quadrature Outputs," *IEEE International ASIC/SOC Conference*, pp.134-138, Sept. 2001

[Ch4.37] M. Tiebout, "Low-Power Low-Phase-Noise Differentially Tuned Quadrature VCO Design in Standard CMOS," *IEEE Journal of Solid-State Circuits*,

vol.36, no.7, pp.1018-1024, Jul. 2001

[Ch4.38] J. Maget, M. Tiebout, and R. Kraus, "Influence of Novel MOS Varactors on the Performance of a Fully Integrated UMTS VCO in Standard 0.25-μm CMOS Technology," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 7, pp.953–958, July 2002

[Ch4.39] A. Harjimiri, S. Limotyrakis, and T.H. Lee, "Jitter and Phase Noise in Ring Oscillator," *IEEE Journal of Solid-State Circuits*, vol. 34, no. 6, pp.790-804, Jun. 1999

[Ch4.40] M.J. Deen, M.H. Kazemeini, and S. Naseh, "Performance Characteristics of an Ultra-Low Power VCO," *IEEE International Symposium on Circuits and Systems,* pp.697-700, May 2003

[Ch4.41] H.-C. Tang and M.-C. Shiau, "Power Dissipation Models and Performance Improvement Techniques for CMOS Inverters with RC Line and Tree Interconnections," *IEE Proceedings G Circuits, Devices & Systems*, vol. 140, issue 6, pp.437-443, Dec. 1993

[Ch4.42] A. Hajimiri, "Noise in Phase-Locked Loops," *Proc. of IEEE Southwest Symposium on Mixed-Signal Circuits*, pp.1-6, Feb. 2001.

[Ch4.43] Maxim Integrated Products Inc., "Clock (CLK) Jitter and Phase Noise Conversion – Application Note 3359," *Maxim Integrated Products Inc.*, http://ww.maxim-ic.com, Step. 2004.

[Ch4.44] Microchip Technology Inc., "Basic PLL Filters for the rfPICTM / rfHCS – Application Note 846," *Microchip Technology Inc.*, http://www.mictochip.com, 2002.

[Ch4.45] K. Kundert, "Predicting the Phase Noise and Jitter of PLL-Based Frequency Synthesizers," *The Designer's Guide Community*, http://www.designers-guide.com, Ver.4g, Aug. 2006.

[Ch4.46] M. Takahashi, K. Ogawa, and K.S. Kundert, "VCO Jitter Simulation and its Comparison with Measurement," *IEEE Asia and South Pacific Design Automation Conference,* vol.1, pp.85-88, Jan. 1999.

[Ch4.47] PCI SIG, "PCI Express Jitter White Paper - PCI Express Jitter Modeling, Revision 1.0RD," *PCI SIG,* http://www.pcisig.com , Jul. 14, 2004.

[Ch4.48] American National Standards Institute, "Fibre Channel Physical Interface-4 (FC-PI-4) Rev. 6.10," *American National Standards Institute*, Feb. 27, 2007.

[Ch5.1] Serial ATA Workgroup, "Serial ATA," Revision 2.6, May 2007.

[Ch5.2] INCITS, "Serial Attached SCSI – 2 (SAS-2)," Revision 14a, May 2008.

[Ch5.3] M. Aoyama et al., "3Gbps, 5000ppm Spread Spectrum SerDes PHY with frequency tracking Phase Interpolator for Serial ATA," *Dig. Sym. VLSI Circuits*, pp.107-110, Jun. 2003.

[Ch5.4] M. Sugawara et al., "1.5 Gbps, 5150 ppm SPREAD Spectrum SerDes PHY with a 0.3mW, 1.5 Gbps Level Detector for Serial ATA," *Dig. Sym. VLSI Circuits*, pp. 60-63, Jun. 2002.

[Ch5.5] T. Yoshikawa et al., "An Over-1-Gb/s Transceiver Core for Integration Into Large System-on-Chips for Consumer Electronics," *IEEE Transactions on VLSI*, vol.16, no.9, pp.1187-1198, Sept. 2008.

[Ch5.6] M. Kokubo et al., "Spread-Spectrum Clock Generator for Serial ATA using Fractional PLL Controlled by ΔΣ Modulators with Level Shift," *ISSCC Dig. Tech. Papers*, pp.160-161, Feb. 2005.

[Ch5.7] H.-R. Lee et al., "A Low Jitter 5000ppm Spread Spectrum Clock Generator for Multi-channel SATA Transceiver in 0.18μm CMOS," *ISSCC Dig. Tech. Papers*, pp.162-163, Feb. 2005.

[Ch5.8] D.-S. Shen and S.-I. Liu, "A Low-Jitter Spread Spectrum Clock Generator Using FDMP," *IEEE Trans. on Circuits and Systems II: Express Briefs*, Vol. 54, pp. 979 – 983, Nov., 2007.

[Ch5.9] J. Zheng et al, "A Programmable Spread Spectrum Clock Generation," *Proceedings, 7th International Conf. on ASIC*, pp. 337 – 339, 2007.

[Ch5.10] R. Saraswat et al, "Programmable Spread Spectrum Clock Generation Based on Successive Phase Selection Technique," *Proceedings, IEEE International Symposium on Circuits and Systems*, pp. 2845 – 2848, 2008.

[Ch5.11] Ming-ta Hsieh and Gerald E. Sobelman, "Clock and Data Recovery with Adaptive Loop Gain for Spread Spectrum SerDes Applications," *Proceedings, IEEE International Symposium on Circuits and Systems,* pp. 4883-4886, 2005

[Ch6.1] Genesys Logic America, Inc., "Multi-Gigabit SerDes: The Cornerstone of High Speed Serial Interconnects," *Multi-Gigabit SerDes White Paper,* Gensys Logic America, Inc., Sept. 15, 2003.

[Ch6.2] The Serial ATA Working Group, "SATA 1.0a Specification," The Serial ATA Working Group, http://www.serialata.org, Jan. 7, 2003.

[Ch6.3] R. Drost, "Architecture and Design of a Simultaneously Bidirectional Single-ended High Speed Chip-to-Chip Interface," *Ph.D. Thesis, Stanford University, Palo Alto, CA,* Nov. 2001.

[Ch6.4] D.N. de Araujo, M. Cases, and N. Pham, "Design Optimization Methodology for Simultaneous Bidirectional Interface," *IEEE Electrical Performance of Electronic Packaging*, pp. 295-298, Oct. 2001.

[Ch6.5] K.S. Canagasaby, S. Rajagopalan, and S. Dabral, "Interconnect Design Challenges in Source Synchronous Simultaneous Bidirectional Links," *IEEE Elect. Perf. of Electronic Packaging*, pp. 11-14, Oct. 2002.

[Ch6.6] R. Farjad-Rad et al, "A 0.3-μm CMOS 8-Gb/s 4-PAM Serial Link Transceiver" IEEE Jour. Solid-State Circuits, Vol.35, No.5, pp.757-764, May 2000.

[Ch6.7] J. Sonntag et al. "An adaptive PAM-4 5 Gb/s backplane transceiver in 0.25μm CMOS," *IEEE Custom Integrated Circuits Conference*, pp.363-366, 2002.

[Ch6.8] D. J.. Foley and M. P. Flynn "A low-power 8-PAM serial transceiver in 0.5-μm digital CMOS," *IEEE Journal of Solid-State Circuits*, Vol.37, No.3, pp.310-316, March 2002.

[Ch6.9] D.N. de Araujo, M. Cases, N. Pham, and D. Dreps, "Unidirectional vs. Simultaneous Bidirectional Source Synchronous Signaling," *IEEE Electrical Perf. of Electronic Packaging*, pp. 7-10, Oct. 2002.

[Ch6.10] M. Fukaishi, "GHz serial link transceiver using multiple-valued data representation," *IEEE 11th International Workshop on Post Binary ULSI*, May 2002.

[Ch6.11] L. Harrison and H. Takatori, "Extending the Life of Today's Backplanes through Simultaneous Bidirectional Transmission," *DesignCon* 2004.

[Ch6.12] J. Zhang and Z. Wong, "White Paper on Transmit Pre-emphasis and Receive Equalization," Mindspeed Technologies – A Conexant Bus., Oct. 2002.

[Ch6.13] K. Dalmia, "High Performance Backplane Design Using the Marvell AlaskaTM X Quad 3.125 Gb/s SERDES," Marvell white paper, Nov. 2001.

[Ch6.14] J.N. Babanezhad, "A 3.3 V analog Adaptive Line-Equalizer for Fast Ethernet Data Communication," *IEEE Custom Integrated Circuits Conference*, pp. 343-346, May 1998.

[Ch6.15] R. Kollipara, G Yeh, B. Chia, and M. Agarwal, "Design, Modeling and Characterization of High Speed Backplane Interconnect," *DesignCon* 2003.

[Ch7.1] ITRS, "International Technology Roadmap for Semiconductors 2007 Edition: Assembly and Packaging," *International Technology Roadmap for Semiconductors (ITRS)*, http://www.itrs.net, 2007

[Ch7.2] B. Razavi, "Design of Integrated Circuits for Optical Communications," *McGraw-Hill*, New York, 2003

[Ch7.3] B. Razavi, "Challenges in the Design of High-Speed Clock Data Recovery Circuit*," IEEE Communication Magazine*, pp.94-101, Aug. 2002

[Ch7.4] R. C. Walker, "Designing Bang-Bang PLLs for Clock and Data Recovery in Serial Data Transmission Systems (Book style with paper title and editor)," *Phase-Locking in High-Performance Systems from Devices to Architectures*, B. Razavi, Ed. New Jersey: John Wiley & Sons, Inc., 2003, pp.34–45, 2003

[Ch7.5] A. Pottbäcker, U. Langmann, and H.-U. Schreiber, "A Si Bipolar Phase and Frequency Detector IC for Clock Extraction Up to 8 Gb/s," *IEEE Journal of Solid-State Circuits*, vol.27, no.12, pp.1747–1751, Dec. 1992

[Ch7.6] J.C. Scheytt, G. Hanke, and U. Langmann, "A 0.155-, 0.622, and 2.488-Gb/s Automatic Bit-Rate Selecting Clock and Data Recovery IC for Bit-Rate Transparent SDH Systems," *IEEE Journal of Solid-State Circuits*, vol. 34, no.12, pp.1935–1943, Dec. 2003

[Ch7.7] J. L. Sonntag and J. Stonick, "A Digital Clock and Data Recovery Architecture for Multi-Gigabit/s Binary Links," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 8, pp.1867–1875, Aug. 2006

[Ch7.8] M.H. Perrott, et al., "A 2.5-Gb/s Multi-Rate 0.25-µm CMOS Clock and Data Recovery Circuit Utilizing a Hybrid Analog/Digital Loop Filter and All-Digital Referenceless Frequency Acquisition," *IEEE Journal of Solid-State Circuits*, vol. 41, no.12, pp.2930–2944, Dec. 2006

[Ch7.9] X. Maillard, F. Devisch, and M. Kuijk, "A 900-Mb/s CMOS Data Recovery DLL Using Half-Frequency Clock," *IEEE Journal of Solid-State Circuits*, vol.37, no.6, pp.711–715, Dec. 2002

[Ch7.10] H.-H. Chang, R.-J. Yang, and S.-I. Liu, "Low Jitter and Multirate Clock and Data Recovery Circuit Using a MSADLL for Chip-to-Chip Interconnection,"

IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, vol.51, no.12, pp.2356–2364, Dec. 2004

[Ch7.11] R. Kreienkamp, U. Langmann, C. Zimmermann, T. Aoyama, and H. Siedhoff, "A 10-Gb/s CMOS Clock and Data Recovery Circuit with an Analog Phase Interpolator," *IEEE Journal of Solid-State Circuits*, vol. 40, no.3, pp.736–743, Mar. 2005

[Ch7.12] M. Y. He and J. Poulton, "A CMOS Mixed-Signal Clock and Data Recovery Circuit for OIF CEI-6G+ Backplane Transceiver," *IEEE Journal of Solid-State Circuits*, vol. 41, no.3, pp.597–606, Mar. 2006

[Ch7.13] M. Hsieh, and G.E. Sobelman, "Clock and Data Recovery with Adaptive Loop Gain for Spread Spectrum SerDes Applications," *IEEE International Symposium on Circuits and Systems*, pp.4883-4886, May 2005

[Ch7.13] H.-T. Ng, et.al , "A Second-Order Semidigital Clock Recovery Circuit Based on Injection Locking," *IEEE Journal of Solid-State Circuits*, vol. 38, no.12, pp.2101–2110, Dec. 2003

[Ch7.14] R. Farjad-Rad, et al., "A 33-mW 8-Gb/s CMOS Clock Multiplier and CDR for Highly Integrated I/Os," *IEEE Journal of Solid-State Circuits*, vol.39, no.9, pp.1553–1561, Sept. 2004

[Ch7.15] J. Kim and D.-K. Jeong, "Multi-Gigabit-Rate Clock and Data Recovery Based on Blind Oversampling," *IEEE Communication Magazine*, pp.68-74, Dec. 2003

[Ch7.16] S. I. Ahmed and T. A. Kwasniewski, "Overview Of Oversampling Clock and Data Recovery Circuits" *Canadian Conference on Electrical and Computer Engineering*, pp.1876-1881, May 1-4, 2005

[Ch7.17] M. Ierssel, A. Sheikholeslami, H. Tamura, and W.W. Walker, "A 3.2 Gb/s CDR Using Semi-Blind Oversampling to Achieve High Jitter Tolerance," *IEEE Journal of Solid-State Circuits*, vol.42, no.10, pp.2224–2234, Oct. 2007

[Ch7.18] S.-H. Lee, et al., "A 5-Gb/s 0.25-μm CMOS Jitter-Tolerant Variable-Interval Oversampling Clock/Data Recovery Circuit," *IEEE Journal of Solid-State Circuits*, vol.37, no.12, pp.1822–1830, Dec. 2002

[Ch7.19] M. Nogawa, et al., "A 10 Gb/s burst-mode CDR IC in 0.13μm CMOS," *IEEE International Solid-State Circuits Conference*, Feb. 2005

[Ch7.20] A. Tajalli, P. Muller, M. Atarodi, and Y Leblebici, "A Multichannel 3.5mW/Gbps/Channel Gated Oscillator Based CDR in 0.18μm Digital CMOS Technology," *Proceedings of the 31st European Solid-State Circuits Conference,* pp.193-196, Sept. 2005

[Ch7.21] A. Tajalli, P. Muller, and Y. Leblebici, "A Power-Efficient Clock and Data Recovery Circuit in 0.18um CMOS Technology for Multi-Channel Short-Haul Optical Data Communication," *IEEE Journal of Solid-State Circuits*, vol.42, no.10, pp.2235–2244, Oct. 2007

[Ch7.22] R. Walker, "Clock and Data Recovery for Serial Digital Communications," *IEEE International Solid-State Circuits Conference - Short Course*, http://www.omnisterra.com/walker/pdfs.talks/ISSCC2002.pdf, Feb. 2002

[Ch7.23] K. Ishii, K. Kishine, and H. Ichino, "A Jitter Suppression Technique for a 2.48832-Gb/s Clock and Data Recovery Circuit," *IEEE Transactions on Circuits and Systems II: Analog And Digital Signal Processing,* vol.49, no.4, pp.266–273, Apr. 2002

[Ch7.24] K. Yamashita, M. Nakata, N. Kamogawa, O. Yumoto, and H. Kodera, "Compact-Same-Size 52- and 156 Mb/s SDH Optical Transceiver Modules," IEEE *Journal of Lightwave Technology,* vol.12, no.9, pp.1607–1615, Sept. 1994

[Ch7.25] H. Wang and R. Nottenburg, "A 1 Gb/s CMOS Clock and Data Recovery Circuit," *IEEE International Solid-State Circuits Conference*, Feb. 1999

[Ch7.26] L. Henrickson, et al., "Low Power Fully Integrated 10-Gb/s SONET/SDH Transceiver in 0.13-μm CMOS," *IEEE Journal of Solid-State Circuits*, vol.38, no.10, pp.1595–1601, Oct. 2003

[Ch7.27] H. S. Muthali, T. P. Thomas, and I. A. Young, "A CMOS 10-Gb/s SONET Transceiver, " *IEEE Journal of Solid-State Circuits*, vol. 39, no.7, pp. 1026–1033, Jul. 2004

[Ch7.28] P. K. Hanumolu, M.G. Kim, G.-Y. Wei, and U.-K. Moon, "A 1.6Gbps Digital Clock and Data Recovery Circuit," *IEEE Custom Integrated Circuits Conference*, pp.603–606, Sept. 2006

[Ch7.29] D. Dalton, et al., "12.5-Mb/s to 2.7-Gb/s Continuous-Rate CDR with Automatic Frequency Acquisition and Data-Rate Readback," *IEEE Journal of Solid-State Circuits*, vol.40, no.12, pp.2713–2725, Dec. 2005

[Ch7.30] W. Rhee, et al., "A 10-Gb/s CMOS Clock and Data Recovery Circuit Using a Secondary Delay-Locked Loop," *Proceedings of the IEEE 2003 Custom Integrated Circuits Conference*, pp.81–84, Sept. 2003

[Ch7.31] P. Larsson, "A 2–1600-MHz CMOS Clock Recovery PLL with Low-Vdd Capability," *IEEE Journal of Solid-State Circuits*, vol.34, no.12, pp. 1951–1999, Dec. 1999

[Ch7.32] J. Lee, and M. Liu, "A 20Gb/s Burst-Mode CDR Circuit Using Injection-Locking Technique," *IEEE International Solid-State Circuits Conference*, Feb. 2007

[Ch7.33] I. Radovanovic´, W. Etten, and H. Freriks, "Ethernet-Based Passive Optical Local-Area Networks for Fiber to the Desk Application," *IEEE Journal of Lightwave Technology,* vol.21, no.11, pp.2534-2545, Nov. 2003

[Ch7.34] B. Analui and A. Hajimiri, "Instantaneous Clockless Data Recovery and Demultiplexing," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol.52, no.8, pp.437-441, Aug. 2005

[Ch7.35.] K. Schneider and H. Zimmermann, "Highly Sensitive Optical Receivers," *Springer Berlin Heidelberg*, New York, 2006

[Ch7.36] C.-H. Liang, S.-C. Hwu, and S.-I. Liu, "A 2.5Gbps Burst-Mode Clock and Data Recovery Circuit," *IEEE Asian Solid-State Circuits Conference*, pp.457–460, Nov. 2005

[Ch7.37] P. Wallace, R. Bayruns, J. Smith, T. Laverick, and R. Shuster , "A GaAs l.SGb/s Clock Recovery and Data Retiming Circuit," *IEEE International Solid-State Circuits Conference*, Feb. 1990

[Ch7.38] J. Kenney, et al., "A 9.95 to 11.1Gb/s XFP Transceiver in 0.13µm CMOS," *IEEE International Solid-State Circuits Conference*, Feb. 2006.

[Ch8.1] B. Razavi, "Challenges in the Design of High-Speed Clock Data Recovery Circuit," *IEEE Communication Magazine*, August 2002, pp 94-101.

[Ch8.2] Serial ATA Working Group, "Serial ATA II: Electrical Specification, Revision 1.0", *Serial ATA Working Group* http://www.sata-io.org, May 2004.

[Ch8.3] A. Rezayee, and K. Martin, "A 9-16Gb/s Clock and Data Recovery Circuit with Threestate Phase Detector and Dual-Path Loop Architecture," *Proceedings of the 2003 European Solid-State Circuits Conference*, Estoril, Portugal, pp. 683-686.

[Ch8.4] R. Kreienkamp, U. Langmann, C. Zimmermann, and T. Aoyama, "A 10-Gb/s CMOS clock and data recovery circuit with an analog phase interpolator," *Proceedings of the IEEE 2003 Custom Integrated Circuits Conference*, pp. 73-76.

[Ch9.1] Serial ATA Workgroup, "Serial ATA," Revision 2.6, May 2007.

[Ch9.2] INCITS, "Serial Attached SCSI – 2 (SAS-2)," Revision 14a, May 2008.

[Ch9.3] PCI SIG, "PCI Express Jitter White Paper - PCI Express Jitter Modeling, Revision 1.0RD," *PCI SIG,* http://www.pcisig.com , Jul. 14, 2004.

[Ch9.4] American National Standards Institute, "Fibre Channel Physical Interface-4 (FC-PI-4) Rev. 6.10," *American National Standards Institute*, Feb. 27, 2007.