UNIVERSITY OF CINCINNATI

Date: March 28 2007

I, Xiaoshan Pan Gilmore,

hereby submit this work as part of the requirements for the degree
of:

Master of Science

in:

Computer Engineering

It is entitled:

Performance Analysis for Clock and Data Recovery

Circuits Under Process Variation

This work and its defense approved by:

Chair: Wen-Ben Jone

Ranga R. Vemuri

Hal Carter

Performance Analysis for Clock and Data Recovery Circuits under Process Variation

A thesis submitted to the
Division of Graduate Studies and Research
of the University of Cincinnati
in partial fulfillment of the
requirements of the degree of

MASTER OF SCIENCE

in the Department of
Electrical and Computer Engineering
of the College of Engineering

Mar. 28, 2007

by

Xiaoshan Pan Gilmore

Bachelor of Science in Electrical Engineering

Sun Yat-sen University, Guangzhou, China

June 2002

Thesis Advisor and Committee Chair: Dr. Wen-Ben Jone

Abstract

Clock and data recovery circuits play a very important role in modern data communication systems. It has very wide application in many areas, such as optical communications and interconnection between chips [1]. Today in IC industry, the shrinkage of feature size increasingly enlarges the uncertainty of circuit performance caused by process variation. As the data transmission speed dramatically increases, this uncertainty will heavily affect the clock and data recovery circuit performance and reliability in communication systems. Thus, research on performance variation of a clock and data recovery circuit caused by process variation is meaningful. The conclusion will have significant influence on chip testing.

In this research, a clock and data recovery circuit is laid out by TSMC 180nm technology. The performance variation caused by process variation is investigated by HSPICE simulation, and compared with the theoretical analysis results derived through the mathematical model of the clock and data recovery circuit. The results demonstrate that our theoretical model matches well with the real simulations. Both theoretical and simulation results also indicate that process variations in the low pass filter have significant impact on performance parameters such as damping ratio, natural frequency, and lock time of the clock and data recovery circuit.

Acknowledgement

Firstly and most importantly, I am very grateful to my advisor, Dr. Wen-Ben Jone, for his valuable guidance and encouragement. Without him, I could not have worked on this research smoothly. For this research, his time and efforts deserve the most credit.

Secondly, I would love to thank Dr. Carter and Dr. Vemuri for their time, instruction and guidance in my research.

Thirdly, I really appreciate all the love and support from my husband, Andrew Gilmore, from my grandparents, my parents and sister – Ms. Ruizhen He, Mr. Ye Chen, Ms. Xiuqun Huang, Mr. Zhuoyu Pan, Ms. Liming Pan, and from my parents-in-law, Mr. and Mrs. Gilmore.

Last, I really appreciate all the help and support from my good friends who are willing to give me a hand to help me proceed in my research and life – Abelardo, Weihuang, Tao and Rui.

Chapter 1	5
Chapter 2	9
2.1 Clock and Data Recovery Circuit Introduction.....	10
2.2 Design of Our Clock and Data Recovery Circuit.....	18
2.2.1 XN Programmer Design	19
2.2.1.1 Value of N	19
2.2.1.2 $X2$ Programmer Implementation	20
2.2.2 Phase Detector	21
2.2.2.1 XOR Gate Phase Detector Operation	21
2.2.2.2 XOR Gate Phase Detector Gain	27
2.2.3 Voltage Controlled Oscillator	28
2.2.3.1 VCO Stage Number	28
2.2.3.2 VCO Gain	32
2.2.4 Low Pass Filter	34
Chapter 3	45
3.1 Sources of Variation and Assumptions	47
3.2 Mathematical Models for Damping Ratio, Natural Frequency, and Lock Time	
.....	49
3.3 Process Variation Analysis for Damping Ratio, Natural Frequency, and Lock	
Time.....	50
Chapter 4	71
4.1 RC Value Assignment in HSPICE Simulations.....	71
4.2 Simulation Results and Analysis.....	72

4.2.1 Damping Ratio ζ	72
4.2.1.1 Calculation of Damping Ratio ζ	72
4.2.1.2 Derivation of Normalized Histogram of ζ	75
4.2.1.3. Summary.....	77
4.2.2 Natural Frequency W_n	79
4.2.2.1 Calculation of Natural Frequency W_n by Simulation	79
4.2.2.2 Normalized Histogram of W_n	80
4.2.2.3 Conclusion Based on Simulation Results.....	81
4.2.3 Lock Time T_{Lock}	83
4.2.3.1 Calculation of Lock Time T_{Lock} Based on Simulation.....	83
4.2.3.2 Normalized Histogram of T_{Lock}	84
4.2.3.3 Conclusions.....	85
Chapter 5	89

Chapter 1

Introduction

Over the past years, the rapid development of high speed data communication reveals the importance of clock and data recovery circuits. With wide applications in a lot of areas, clock and data recovery circuits are gaining more and more popularity among researchers [1-8] [11-30]. As hinted by its name, the circuit can extract a clock signal and use this clock signal to recover the data. Specifically, it relies on a phased locked loop (PLL) to generate a clock signal that matches the frequency of the data input, and the clock signal is used to recover the data. In data communication, this is called self synchronous scheme, as opposed to source synchronous scheme and system synchronous scheme [8].

For high speed data transmission, self synchronous scheme is superior to the other two schemes. In system synchronous scheme, a common clock signal is used between the transmitter and the receiver. To ensure synchronous operations within the system, clock signal delay management, which is impossible when data speed is very high, is necessary. In source synchronous scheme, the source sends out clock signal with the data. This scheme eliminates the clock signal delay management, and hence significantly simplifies

the timing parameters. However, the received data must be processed to synchronize with the global clock. Therefore, source synchronous scheme significantly increases the clock domains and hence introduces timing constraint and analysis complications.

As a counterpart, the scheme that clock and data recovery circuit implements, the self synchronous scheme, neither requires clock signal delay management nor results in different clock domains. Because of this, self synchronous scheme is the major trend in implementing high speed data communication systems, and clock and data recovery circuit is the fundamental component in such systems. Today, it is gaining more and more attention. Also, as the key component in transceivers, any variation of the clock and data recovery circuit performance directly imposes constraints on the upper bound of the data frequency that the system can handle. Therefore, the circuit performance management is very important for clock and data recovery circuit.

However, the clock and data recovery circuit performance management is becoming a challenge. As technology advances and transistor size gets smaller and smaller, uncertainty of circuit parameters increases. According to [9-10], for 50nm technology, effective channel length can vary up to 10%; gate oxide thickness can vary up to 5% and channel doping concentration can range up to 5%. All these fluctuations, coupled with process variations (uncertainties resulting from manufacturing process) introduce unexpected circuit performance variation in a large range. On the other hand, the upper bound of the data frequency that the system can recover is strictly constrained by these uncertainties and all these effects will become more and more prominent as the feature

size decreases. Among the existing research results for clock and data recovery circuits, to the best of our knowledge, there is no paper that discusses the circuit performance variation caused by process variation, which is a very meaningful topic today.

In this research, the clock and data recovery circuit performance variation caused by process variation is investigated. Specifically, a clock and data recovery circuit is laid out in MAGIC and simulated by HSPICE using TSMC 180nm technology. The most important circuit parameters, such as natural frequency, under process variation are analyzed and simulated. The simulation results on circuit performance variation are analyzed and compared with the predicted variation results derived from the mathematical model of a clock and data recovery circuit. We have found that, under the influence of process variation, the damping ratio, natural frequency and lock time, vary in the same trend as predicted by the mathematical model. The mean of the performance variation can be predicted by the mathematical model. However, the variation in performance by Hspice simulation is larger than the prediction from the mathematical model. Especially, for the cases of damping ratio and lock time, the variation in performance simulation is 5-6 times larger than the theoretical prediction. For the case of natural frequency, the variation in performance simulation is 40% larger than the theoretical prediction.

Further, with our assumption of the process variation in the low pass filter parameters (R and C values), based on the mathematical model, the variation of damping ratio, natural frequency and lock time caused by the variation of the R value is exactly the

same as the variation caused by the variation of the C value. The simulation results show that this prediction is valid. Finally, both theoretical analysis and Hspice simulation demonstrate that process variations in R and C values indeed result in great performance fluctuation for clock and data recovery circuits. This concludes that testing and reliability analysis for clock and data recovery circuits under process variation must be researched more, and this research is emergent as technology keeps advancing to smaller feature size.

This thesis is organized as follows:

Chapter 2 provides a brief introduction on clock and data recovery circuits and the detailed design of our home-made clock and data recovery circuit used for simulation.

Chapter 3 establishes a mathematical model for process variation, and derives the circuit performance variation (such as natural frequency, damping ratio, and lock time) based on the mathematical model of the clock and data recovery circuit.

Chapter 4 discusses the simulation results on the circuit performance variation when process variation is taken into consideration. The results are also compared with the prediction derived through mathematical model, and they are found to basically match each other.

Chapter 5 concludes the work of this research and discusses the future work of this research.

Chapter 2

Clock and Data Recovery Circuit

Introduction and Design

Clock and data recovery circuit performance management is increasingly important due to the explosive development of high speed communication systems. During the past years, extensive research has been done to investigate the clock and data recovery circuit [1-8] [11-30]. Specifically, the existing results mainly fall into three categories: clock and data recovery circuit design and implementation [1-8][11-13], clock and data recovery circuit mathematical model investigation [14-17], and testing schemes for clock and data recovery circuits [18-21]. To the best of our knowledge, there is no existing results that explore the effect of process variation on clock and data recovery circuit performance, though this topic is becoming more and more important and challenging due to the shrinkage of feature size. With this in mind, in this research, we focus on investigating the variation of circuit performance when process variation is taken into consideration. In this chapter, we will provide an introduction on clock and data recovery circuits in Section 2.1. In section 2.2, the detailed design of our clock and data recovery circuit is

described and the most fundamental circuit parameters are introduced and explained with design equations.

2.1 Clock and Data Recovery Circuit Introduction

As the key component in a high speed communication system, clock and data recovery circuit is a very important integrated circuit nowadays. As shown in Fig. 2.1, a clock and data recovery circuit mainly consists of six building blocks: phase detector, low pass filter, voltage-controlled oscillator (VCO), X/N programmer (X means multiplication) and data decision circuit. As a matter of fact, the circuit is a negative feedback system which generates a clock signal whose frequency is the same as the data input frequency with phase difference (shift) equal π . In other words, the transition edge of the clock signal occurs in the middle of the data eye. Hence with the clock signal generated, the data input can be easily recovered. Specifically, the phase detector receives data input and X/N programmer output signal as inputs and generates a sequence of pulses whose widths are proportional to the phase differences between these two inputs. The low pass filter integrates the output of the phase detector, and outputs the average value of the pulse as the control voltage of the VCO. For the VCO, the output signal frequency is decided by the input control voltage. The X/N programmer output signal, whose frequency is $\frac{1}{N}$ of the VCO output clock signal, is used to compare with the data input on phase.

Overall speaking, the working principle of the loop is that the phase detector generates a signal representing the phase difference between the data input and a signal whose frequency is $\frac{1}{N}$ of the VCO output signal. The low pass filter integrates the phase detector output before the later is applied to control the VCO output signal frequency. The XN programmer output signal, being $\frac{1}{N}$ of the VCO output signal on frequency, is fed into the phase detector as a feedback. With the data input and the feedback, the phase detector output further adjusts the VCO control voltage and hence the VCO output signal frequency to create a phase difference equal to π between the data input and the VCO output signal. Hence, when the system is locked, the phase difference between the data input and the VCO output signal is π . In other words, the transition edge of the clock signal occurs in the middle of the data eye. Thus, the data can be recovered nicely.

Fig. 2.1. Block diagram of a clock and data recovery circuit.

Among all different components in the clock and data recovery circuit, phase detector design gains the most attention from researchers [22-29] due to its direct and

significant influence on the circuit performance. Due to limited space, only a brief introduction on the phase detector design will be given in this section. Fortunately, a very good survey on the existing phase detector design can be found in [1]. The existing phase detector designs can be classified as the full rate phase detector and the half rate phase detector. With the full rate phase detector, the clock and data recovery circuit samples the data at full rate, and the VCO generates a clock signal whose frequency is the same as the data input frequency. As the counterpart, the half rate phase detector enables the clock and data recovery circuit operates at half rate, i.e., the data is sampled at the rising edge and the falling edge of its clock signal. The VCO output signal frequency is half of the data input frequency. For very high speed data communication, the application of half rate phase detector makes the design and implementation of VCO simpler.

For full rate phase detector, Hogge phase detector and Alexander phase detector are the two very popular designs. Most of the phase detector designs, including full rate and half rate, are modified versions of the Hogge [1] [22-24] or Alexander phase detector [1]. Hence, here we will just expand discussion on these two phase detectors and their half rate versions. Fig. 2.2 shows the architecture of a Hogge phase detector, and Fig. 2.3 shows an example of the signal change in the Hogge phase detector. In the Hogge phase detector, as shown in Fig. 2.2, signal B is a delayed version of the data input and the delay is equal to the phase difference between the data input and the rising edge of the clock signal. Hence, signal Y, the XOR output of signal B and the data input, will output a pulse whose width equals the phase difference between the data input and the rising edge of the clock signal whenever there is a data transition. However, if we just use the Y

signal, the average value cannot uniquely represent the phase difference since the average value also depends on the transition density. So we need a reference signal on every data transition.

Based on this architecture, we can see that signal A is a copy of the signal B with a delay equal half of a clock cycle. As the XOR output of A and B, signal X will output a pulse whose width equals to half of the clock cycle whenever there is a data transition. With this reference signal, when the system is locked, the phase difference between the data input and the clock signal will be equal to half of a clock period. In other words, the rising edge of the clock signal will occur in the middle of the data eye. The shortcoming of this architecture is that since the X signal and the Y signal are not valid simultaneously, there will be jitters in the system. Also, because the Clock-Q delay of the D flip-flop, the Y signal will have a pulse whose width is equal to the phase difference between the data input and the clock signal plus the Clock-Q delay of the D flip-flop. Eventually, this will result in a half of the clock period minus Clock-Q delay phase difference between the data input and the clock signal when the system is locked. This might make the system fail to recover data when the data input frequency is very high.

Fig. 2.2. Hogge phase detector architecture [1].

Fig. 2.3. Signals change in Hogge phase detector [1].

For the Alexander phase detector, the working principle can be explained with Fig. 2.4 [1]. As shown in the figure, the data is sampled by three consecutive clock transition edges. If $S1 \oplus S2$ is logic '1' and $S2 \oplus S3$ is logic '0', the clock lags the data. If $S1 \oplus S2$ is logic '0' and $S2 \oplus S3$ is logic '1', the clock is early. If $S1 \oplus S2$ is equal to $S2 \oplus S3$, no data transition is present. Based on this observation, the Alexander phase detector architecture is designed as shown in Fig. 2.5. For Alexander phase detector, the X signal and the Y signal are valid simultaneously. Hence, it does not cause jitter like Hogge phase detector does.

Fig. 2.4. Working principle of Alexander phase detector [1].

Fig. 2.5. Architecture of Alexander phase detector [1].

On the half rate phase detector, [1] gives a half rate phase detector design which is based on the Hogge phase detector. The architecture is shown in Fig. 2.6. Fig. 2.7 shows an example of the signal change for this phase detector. Specifically, L1, L2, L3, L4 are all latches. L1 is transparent when the clock signal is logic '1' and L2 is transparent when the clock signal is logic '0'. From Fig. 2.7, it is obvious that every data transition occurring when the clock signal is high will appear in the output of L1. Similarly, every data transition occurring when the clock signal is low will appear in the output of L2. Accordingly, every data transition occurring when the clock signal is high will not be bypassed to L2, but the signal will show up on L2 output when the clock signal drops to 0. Similarly, every data transition occurring when the clock signal is low will not be bypassed to L1, but the signal will show up on L1 output when the clock signal becomes 1. Hence, whenever there is a data transition, the XOR output of L1 output and L2 output, which is signal V_{out1} , will have a pulse whose width equals to the phase difference between the data and the nearest clock signal transition edge. Similar to the Hogge phase

detector, a reference signal is needed for this half rate phase detector. For this purpose, two more latches, L3 and L4 are added as shown in Fig. 2.6. As discussed above, all the data transitions that occur when the clock signal is high will appear in the output of latch L1, and hence the corresponding new value the data assumes after transition will appear in the output of L3 for the next clock cycle. Similarly, all the data transitions that occur when the clock signal is low will appear in the output of L2, and the new value the data assumes after this transition will appear in the output of L4 for the next clock cycle. Hence the XOR output of the output of L3 and the output of L4, which is signal V_{out2} , will have a pulse whose width equals to half of the clock cycle whenever there is a data transition. So, V_{out2} works as a reference signal for this half rate phase detector. The work in [1] also provides a half rate phase detector design which is based on the Alexander phase detector. The design and the signal relationship are shown in Fig. 2.8. As shown in Fig. 2.8, with the two clock signals whose phase difference equals to $\frac{\pi}{2}$, we can easily see that A3, A2, and A1 correspond to signal S3, S2 and S1 in the Alexander phase detector respectively.

Fig.2.6. Architecture of half rate phase detector modified from Hogge phase detector [1].

Fig.2.7. Signal changes for the half rate phase detector modified from Hogge phase detector [1].

Fig. 2.8. Half rate phase detector design based on Alexander phase detector [1].

In summary, clock and data recovery circuit is a very important integrated circuit nowadays due to the explosive development of high speed data communication technologies and dramatic increase of the data transmission speed. The circuit employs a phase locked loop to recover a clock signal that is used to recover the data. With this circuit, no clock signal needs to be sent by the transmitter, and hence significantly

reduces the complexity of the communication system design. With the development of high speed digital communication, this self synchronous communication scheme becomes the trend, and hence clock and data recovery circuit becomes one of the most prominent circuits today.

2.2 Design of Our Clock and Data Recovery Circuit

As discussed in Section 2.1, the clock and data recovery circuit is a key component in serial link data communication systems. As the rapid development of serial link data communication and dramatic increase of data transmission speed, the clock and data recovery circuit is gaining more and more popularity [1-8] [11-30]. However, to the best of our knowledge, there is no research result on performance variation of a clock and data recovery circuit caused by process variation, i.e., the effect of fabrication process uncertainty which includes inter-wafer and intra-wafer transistor and thermal variations. Obviously, process variation will increase as the technology advances and feature size decreases. Eventually, these uncertainties will have significant effects on circuit performance. So, the focus of this research is set to investigate the clock and data recovery circuit performance variation caused by process variation. For this purpose, a clock and data recovery circuit, as shown in Fig. 2.9, is designed and laid out by MAGIC and simulated in HSPICE using TSMC 180nm technology. To explain how the circuit works, the design of each component is investigated in the following sections.

Fig. 2.9. A clock and data recovery circuit [30].

2.2.1 XN Programmer Design

2.2.1.1 Value of N

In the system feedback path, an XN (X means multiplication) programmer is applied to generate signals whose period is N times the VCO output signal period. By using this device, if the input to the system is a reference clock, we can generate clock signals whose frequency is N times the reference clock frequency at the VCO output. The reason for this is that when the system locks, the two inputs to the phase detector have the same period. In other words, the reference clock period is the same as the XN programmer output signal period, which is N times the VCO output signal period. Hence, the VCO output signal frequency is N times the reference clock frequency when the system stabilizes. Similarly, when the input to the system is data, the time it takes to output two bit data is N times the VCO output signal period. Since data period refers to the time it

takes to output one bit of data, when the system locks, the VCO output signal frequency is $\frac{N}{2}$ times the data frequency. For the purpose of data recovery, the clock output, i.e., the VCO output, of the system should have rising edge in the middle of the data eye, as shown in Fig. 2.10. As a result, the VCO output frequency should be the same as the data frequency. Based on the analysis above, the value of N must be 2 such that the data period and the $X N$ programmer output period are the same. Hence, an $X 2$ programmer is applied in the feedback path.

Fig. 2.10. Relationship between clock and data in a stable system.

2.2.1.2 $X 2$ Programmer Implementation

To achieve an $X 2$ programmer, a flip flop and an inverter are connected as shown in Fig. 2.11 – the output of the flip-flop goes through an inverter and gets back to the input the flip flop. With this connection, the $X 2$ programmer output changes its logic level every rising edge of the programmer input, as shown in Fig. 2.12. Hence, the $X 2$ programmer output frequency is 50% of the programmer input frequency. In other words, the output signal has a period twice the input signal period. If the VCO output clock signal is used as the programmer input (the clock), we can get a signal whose period is twice the VCO output signal period.

Fig. 2.11. X 2 programmer implementation.

Fig. 2.12. X 2 programmer input, output and internal signal changes.

2.2.2 Phase Detector

2.2.2.1 XOR Gate Phase Detector Operation

For our clock and data recovery circuit, an *XOR* gate is applied as the phase detector. The reason is that *XOR* gate phase detector is the most fundamental phase detector design and it is very commonly used in data communication systems [30]. The way it works can be demonstrated by Fig. 2.13 – the width of the *XOR* gate output pulse is proportional to the phase difference of its two inputs, *in1* and *in2*. As a result, the output of the *XOR* gate represents the phase difference of the two inputs and hence can work as a phase detector.

XOR gate phase detector has several characteristics. To ensure the system is working right, we should understand the following conditions and requirements [30].

Fig. 2.13. Input and output of the *XOR* gate phase detector.

- Before the system locks, data input must change logic level for a minimum number of times within a certain time interval.

Firstly, let us analyze what the phase detector output is like when the system is locked. As discussed in Section 2.2.1, when the system is in lock, the two inputs of the *XOR* gate phase detector should be of the same frequency. Also, for the purpose of data recovery, when the system is locked, the clock output, i.e., the VCO output, should have rising edge in the middle of the data eye, as shown in Fig. 2.10. With the $X/2$ programmer in the feedback path, we can predict that both inputs of the *XOR* gate phase detector has a phase difference of $\frac{\pi}{2}$, as shown in Fig. 2.14, when the system is locked. Hence, the phase detector output has 50% duty cycle when the system is locked. If the phase detector output is connected with a low pass filter, the filtered output will be $\frac{VDD}{2}$. So, the VCO

should be designed so that it will output the desired clock frequency when the control voltage is $\frac{VDD}{2}$.

Fig. 2.14. Phase relationship among different signals in a locked system.

Now, As shown in Fig. 2.15, if the incoming data does not have any transitions, say, always staying low, the output of the phase detector is just a replica of the $\times 2$ programmer output, the other input of the phase detector. Since the programmer output is a series of 1 0 1 0 with period two times the VCO output signal period, the phase detector output will have 50% duty cycle. So the system “appears” to be in the “locked” status, while there is no way to judge if the output of the $\times 2$ programmer really has $\frac{\pi}{2}$ phase difference with the data input. When the data input changes its logic level, the system might not work if the $\times 2$ programmer output rising edge is not in the middle of the data eye consequently. Consequently, before the system locks, the data input must change its logic level for a minimum number of times over a certain time interval.

Fig. 2.15. *XOR* gate phase detector output with a quiet data [30].

- If the phase detector is connected with a low pass filter, without any data transition present, the low pass filter output is $\frac{VDD}{2}$.

As mentioned above, when there is no data transition present, the output of the phase detector is a series of 1 0 1 0 with 50% duty cycle. So if the output of the phase detector is connected with the low pass filter, the filtered output is 50% of the maximum possible output - VDD . Hence the filtered output will be $\frac{VDD}{2}$ when the system is locked. As we know, the filtered output of the phase detector is connected with the VCO. So again, it is important to set the desired control voltage (the voltage that VCO will output clock signal with the desired frequency) as $\frac{VDD}{2}$, so that after the system is locked, it can continue to lock when there is no data transition present.

- The *XOR* clock and data recovery circuit has good noise rejection.

The reason for this is that the phase detector output is integrated by the low pass filter. Noise in the data, whose mean is 0, can be rejected due to the integration effect. This is one of the very important reasons for the common use of *XOR* gate phase detector in clock and data recovery circuits for data communication systems. As a counterpart, phase frequency detector does not have good noise rejection, and hence it is not commonly used in data communication systems [30].

- The VCO should operate much slower than $2 f_{clock}$ and much faster than $0.5 f_{clock}$,

where f_{clock} is VCO output signal frequency when the control voltage is $\frac{VDD}{2}$.

The reason for this is that the clock and data recovery circuit with *XOR* gate phase detector can lock at harmonics of input data. For example, as shown in Fig. 2.16(a), if the data input is *Data-1*, the VCO output is *Clock*, the output of the *XOR* gate has 50% duty cycle. Hence, the filtered output of the *XOR* phase detector will stabilize around $\frac{VDD}{2}$ and the system locks. As shown in Fig. 2.16(b), if we replace *Data-1* with *Data-2*, whose frequency is half of *Data-1*, the output of the *XOR* gate still has 50% duty cycle and the filtered phase detector output also stabilizes around $\frac{VDD}{2}$. So the system locks. Finally, as shown in Fig. 2.16(c), if *Data-1* is replaced with *Data-3*, whose frequency is twice *Data-1*, the output of the *XOR* gate still has 50% duty cycle and hence the filtered phase detector output will also stabilize around $\frac{VDD}{2}$. As a result, the system will appear to be

in the “locked” status again. Now, it is clear that though the system appears to be locked when the data input is *Data-1*, *Data-2*, or *Data-3*, it is falsely locked when the data input is *Data-2* or *Data-3*. So, we should limit the data input frequency to be much less than $2 f_{clock}$ and much greater than $0.5 f_{clock}$

Fig. 2.16 (a). Status of different signals when system locks at *Data-1*.

Fig. 2.16 (b). Status of different signals when system locks at *Data-2* falsely.

Fig. 2.16 (c). Status of different signals when system locks at *Data-3* falsely.

- After the system is locked, the VCO output signal frequency will still be oscillating within a small range due to the VCO control voltage ripple.

When the system is locked, the phase detector output is still oscillating. The oscillation will cause ripples on the low pass filter output, which is also the VCO control voltage. With an oscillating control voltage, the VCO output signal frequency oscillates. This is an unwanted characteristic of a clock and data recovery circuit. This effect can be reduced by adjusting the R , C values of the low pass filter. For example, when the product of RC is increased, the ripple will be smaller since the integration effect will be more significant. By doing this, what we sacrifice is the lock time, which is the time it takes for the system to lock at a certain data input. So, a tradeoff needs to be obtained. Usually, the higher frequency the data input has, the higher limiting upper bound on the ripple the system can handle, i.e., the system can still recover the data nicely. Hence, a larger RC value is needed and longer lock time is expected.

2.2.2.2 XOR Gate Phase Detector Gain

The gain of the *XOR* gate phase detector is one of the most important components in the clock and data recovery circuit mathematical model. To derive the *XOR* phase detector gain, we use backward derivation. Firstly, for convenience, we denote both inputs of the *XOR* gate phase detector as *data* and *dclock* as [30]. As discussed in Section

2.2.1, when the system locks, the phase difference between *data* and *dclock* is $\frac{\pi}{2}$, and the

filtered phase detector output is $\frac{VDD}{2}$. If we define the gain of the phase detector as K_{PD} , we can get:

$$K_{PD} \frac{\pi}{2} = \frac{VDD}{2} \quad (2.1)$$

$$\therefore K_{PD} = \frac{VDD}{\pi} \quad (2.2)$$

2.2.3 Voltage Controlled Oscillator

2.2.3.1 VCO Stage Number

For our clock and data recovery circuit, the current-starved VCO design is adopted. Its schematic is shown in Fig. 2.17 [30]. The operation of this VCO is explained in the following discussions. – First, MOSFETs $M2$ and $M3$ work as an inverter, while MOSFETs $M1$ and $M4$ work as current control for $M2$ and $M3$. The name of this design, current starved VCO, comes from the fact that the inverter is starved for current. Based on the circuit schematic, we can easily see that the drain currents of MOSFETs $M5$ and $M6$ are the same and are mirrored in every following stage in the VCO. Note that the number of inverter stages must be odd, such that oscillation can occur. Since the drain current of $M6$ is set by VCO control voltage, the VCO control voltage decides the drain current of $M5$, $M6$ and hence the current in every inverter stage. Due to the fact that the

current in every inverter stage affects the VCO oscillation frequency, the VCO control voltage controls the VCO oscillation frequency. To understand the VCO mathematical model, consider the schematic of two stages of the VCO, as shown in Fig. 2.18. The total drain capacitance of $M2$ and $M3$ is given by Equation (2.3) [30].

$$\begin{aligned}
 C_{tot} &= C_{out} + C_{in} = C'_{ox}(W_P L_P + W_n L_n) + \frac{3}{2} C'_{ox}(W_P L_P + W_n L_n) \\
 &= \frac{5}{2} C'_{ox}(W_P L_P + W_n L_n)
 \end{aligned} \tag{2.3}$$

If the drain current of $M3$ is denoted by I_{D_2} and the time it takes to charge C_{tot} from 0 to V is denoted by t_1 , we can get:

$$t_1 = C_{tot} \frac{V}{I_{D_2}} \tag{2.4}$$

Fig. 2.17. Current starved voltage controlled oscillator [30].

Fig. 2.18. Two stages of the voltage controlled oscillator designed [30].

Similarly, if the drain current of $M2$ is denoted by I_{D_1} and the time it takes to discharge C_{tot} from V to 0 is denoted by t_2 , we can get:

$$t_2 = C_{tot} \frac{VDD - V}{I_{D_1}} \quad (2.5)$$

Since $I_{D_1} = I_{D_2}$, if we denote the value as I_D , we can get

$$t_1 + t_2 = \frac{C_{tot} VDD}{I_D} \quad (2.6)$$

Hence, the oscillation frequency of an N stage current-starved VCO is:

$$f_{osc} = \frac{1}{N(t_1 + t_2)} = \frac{I_D}{NC_{tot} VDD} \quad (2.7)$$

For our research, we want to design a clock and data recovery circuit which can recover 600 MHz data. In other words, the VCO working frequency (i.e., the VCO output

signal frequency when the input control voltage equals to $\frac{VDD}{2}$) should be designed to be 600 MHz. With the assumption that the mobility of electrons is twice the mobility of holes, in our layout, the width of the PMOS is set to be 20λ and the width of the NMOS is set to be 10λ . The length of the PMOS and NMOS are both set to be 4λ (with TSMC 180nm technology, $\lambda = 0.1\mu m$). Plugging in the values into Equation (2.3), we can get:

$$\begin{aligned}
 C_{tot} &= \frac{5}{2} C'_{ox} (W_P L_P + W_n L_n) \\
 &= \frac{5}{2} \frac{\xi_{ox}}{t_{ox}} (W_P L_P + W_n L_n) \\
 &= \frac{5}{2} \frac{3.5 \times 10^{-11}}{4.1 \times 10^{-9}} (20 \times 4 + 10 \times 4) (0.1 \mu m)^2 \\
 &= 25.6 \text{ fF}
 \end{aligned}$$

An one-stage VCO is laid out and the control voltage is set to $\frac{VDD}{2}$, the drain current is measured, and the value is 530uA as shown in Fig. 2.19. Replacing the symbol I_D in Equation (2.7) with this value, we can get the number of stages N as

$$N = \frac{I_D}{f_{osc} C_{tot} VDD} = \frac{530 \mu A}{600 \text{ MHz} \cdot 25.6 \text{ fF} \cdot 5V} = 6.$$

Since N has to be an odd number, we design a five stage current starved VCO. The five stages VCO is laid out in magic and simulated using HSPICE. As shown in Fig. 2.20, the VCO working frequency, i.e., the output frequency when the control voltage is $\frac{VDD}{2}$, is 600 MHz.

Fig. 2.19. Drain current of a single stage when the control voltage is 2.5v.

Fig. 2.20. VCO designed output frequency when the control voltage is equal to $\frac{VDD}{2}$.

2.2.3.2 VCO Gain

Based on [30], the VCO gain in time domain and in complex domain, $K_{VCO}(t)$ and $K_{VCO}(s)$, is given by:

$$K_{VCO}(t) = 2\pi \frac{f_{\max} - f_{\min}}{V_{\max} - V_{\min}} \quad (2.8)$$

$$K_{VCO}(s) = 2\pi \frac{f_{\max} - f_{\min}}{(V_{\max} - V_{\min})s} \quad (2.9)$$

So to calculate the gain of our VCO, we need to investigate the relationship between the input voltage and output frequency. For this purpose, we vary the input voltage and measure the output frequency and the relationship is as shown in Fig. 2.21. Now with Equation (2.8), we can get the value of the VCO gain in time domain as:

$$\begin{aligned} K_{VCO} &= 2\pi \frac{f_{\max} - f_{\min}}{V_{\max} - V_{\min}} \\ &= 2\pi \frac{6.7E+8 - 0}{3.2 - 0.4} \\ &= 1.5E+9 \end{aligned}$$

Fig. 2.21. Relationship between VCO control voltage and VCO output frequency.

2.2.4 Low Pass Filter

For our clock and data recovery circuit, a RC low pass filter is designed to integrate the phase detector output. The schematic is shown in Fig. 2.22. As the brain of the whole system, if the low pass filter parameter values are not selected correctly, the lock time might be too long, or the system is not stable, i.e., even though the system is locked, a small input data variation may cause the system to unlock. To get the best values for R and C , analysis of the whole system in the complex domain is needed.

Fig. 2.22. RC low pass filter schematic [30].

Fig.2.23. Clock and data recovery circuit in complex system.

In the complex domain, we can represent the system as shown in Fig. 2.23. In this figure, the symbolic representations are given as follows:

$\theta_i(s)$: Phase Input,

$\theta e(s)$: Phase Error,

$\theta o(s)$: Output Phase,

$G(s)$: Product of the individual feed forward transfer functions,

$H(s)$: Product of the individual feedback transfer function.

From the figure, we can get the following equations:

$$\theta o(s) = G(s) \theta e(s), \quad (2.10)$$

$$\theta e(s) = \theta i(s) - \theta o(s) H(s). \quad (2.11)$$

From Equations (2.10) and (2.11), we can get

$$\theta o(s) = \frac{G(s)}{1 + G(s)H(s)} \theta i(s). \quad (2.12)$$

If the gain of the phase detector is denoted as K_{PD} , the gain of the low pass filter as K_F , we can get

$$G(s) = K_{PD}(s)K_F(s)K_{VCO}(s) \quad (2.13)$$

$$H(s) = \frac{1}{2} \quad (2.14)$$

As discussed in Section 2.2.2, 2.2.3, we have

$$K_{PD}(s) = \frac{VDD}{\pi} \quad (2.2)$$

$$K_{vco}(t) = 2\pi \frac{f_{\max} - f_{\min}}{(f_{\max} - f_{\min})s} \quad (2.8)$$

$$K_{vco}(s) = 2\pi \frac{f_{\max} - f_{\min}}{(f_{\max} - f_{\min})s} \quad (2.9)$$

Based on the architecture of the low pass filter, the gain of the low pass filter is given by:

$$K_F(s) = \frac{\frac{1}{j\omega C}}{\frac{1}{j\omega C} + R} = \frac{\frac{1}{sC}}{\frac{1}{sC} + R} = \frac{1}{1 + sRC} \quad (2.15)$$

Based on Equations (2.12 – 2.15), the transport function of the system is given by:

$$\begin{aligned} H(s) &= \frac{\phi_{clock}}{\phi_{data}} = \frac{K_{PD} K_F K_{vco}}{1 + \frac{1}{2} K_{PD} K_F K_{vco}} \\ &= \frac{K_{PD} K_{vco}(t)}{S^2 RC + S + \frac{1}{2} K_{PD} K_{vco}(t)} \end{aligned} \quad (2.16)$$

Obviously, the characteristic equation of Equation (2.16) is:

$$S^2 RC + S + \frac{1}{2} K_{PD} \frac{K_{vco}(s)}{s} = 0.$$

When we plug in the standard second order feedback system characteristic equation [31-32]:

$$S^2 + 2\zeta W_n S + W_n^2 = 0 \quad (2.17)$$

we can get:

$$W_n = \sqrt{\frac{\frac{1}{2} K_{PD} \frac{K_{vco}(s)}{s}}{RC}} = \sqrt{\frac{K_{PD} K_{vco}}{2RC}} \quad (2.18)$$

$$\zeta = \frac{1}{2RCW_n} = \frac{1}{2RC} \sqrt{\frac{2RC}{K_{PD} K_{vco}}} = \frac{1}{2} \sqrt{\frac{2}{RCK_{PD} K_{vco}}} \quad (2.19)$$

Specifically, W_n is the natural frequency of the system, while ζ is the damping ratio of the system. In the mathematical model of the data recovery circuit, these two values

are the building blocks used to build the mathematical model for other criteria. To clarify the physical meaning of these two terms, the change of the VCO control voltage is shown in Fig. 2.24. From this figure, we can see that the control voltage of the VCO starts to oscillate when the data input is available. As time goes by, the control voltage stabilizes at the desired voltage. In the mathematical model of the clock and data recovery circuit, the oscillation portion of the control voltage is represented as

$$V_{VCO-Control} = Ae^{-at} \cos(\omega t + \theta) + V_{Stable} \quad (2.20)$$

where $a = \zeta W_n$, $\omega = W_n \sqrt{1 - \zeta^2}$, and V_{Stable} is the desired voltage [31-32]. For our design, it is $\frac{VDD}{2}$ (2.5 V). Note that V_{Stable} is added by us due to the observation that the VCO control voltage has a bias which equals to V_{Stable} .

Fig. 2.24. Change of VCO control voltage in clock and data recovery circuit.

In Equation (2.20), variable a represents how fast the VCO control voltage decay with time. It is proportional to the product of the damping ratio and the natural frequency. Variable ω is the VCO control voltage oscillation frequency and it is the product of the

natural frequency and the damping factor $\sqrt{1-\zeta^2}$. Hence, w is also called “damped frequency”. The larger the value of ζ is, the more rapid decay the oscillation will be, but it takes longer time for the step response of the circuit to rise from 10% to 90% of the steady state value [31-32].

Based on industry experiences [31-32], the most satisfactory circuit performance is found when the damping ratio is about 0.7. For convenience, the numerical value $\frac{\sqrt{2}}{2}$ (0.707), when a and W are equal, is commonly used. Specifically, with the value of damping ratio around 0.7, the clock and data recovery circuit responds most rapidly, and overshoot which is the percentage difference between the maximum and final steady state values is only about 5% [31-32]. For this theory, we tested it using a clock and data recovery circuit layout with three different ζ values 0.4, 0.7 and 1, and the results are compared.

With data represented by the yellow lines and recovered data represented by the red lines, from Fig. 2.25, Fig. 2.26 and Fig. 2.27, we can see that the system locks fastest when ζ equals 0.7. Specifically, when ζ equals 1, it takes the system 14ns to lock. When ζ equals 0.7, it takes the system only 11ns to lock. When ζ equals 0.4, it takes the system 16ns to lock. Specifically, when the VCO control voltage stabilizes within $\pm 3\%$ of $\frac{VDD}{2}$, the system is regarded as locked. Also, note that the value of ζ is decided by the product of the R value and the C value. Hence, with a certain value of ζ , based on

Equation (2.19), we can get the product of the R value and the C value. Keeping this product unchanged, the R value and the C value are randomly selected.

Fig. 2.25. The system locks at 14ns ($\zeta = 1$, $R = 15K$, $C = 15fF$).

Fig. 2.26. The System Locks at 11ns ($\zeta = 0.7$ $R=23K$, $C = 19300pF$).

Fig. 2.27. The system locks at 16ns ($\zeta = 0.4$ $R = 46K$, $C = 30fF$).

In data communication, synchronization data whose pattern is 1 0 1 0 is sent first to lock the system. After the system is stabilized, data with real information, whose pattern is relatively random, is sent. However, with $\zeta = 0.7$ or 1 or 0.4, when the data with real information is present to the system, the clock output does not have rising edge in the middle of the data eye any more, as shown in figures 2.28, 2.29 and 2.30 (the data is represented by yellow line and the clock signal is represented by the red line).

Fig. 2.28. Clock generated when data with real information is present ($\zeta = 1$).

Fig. 2.29. Clock generated when data with real information is present ($\zeta = 0.7$).

Fig. 2.30. Clock generated when data with real information is present ($\zeta = 0.4$).

Only when the R and C values are increased so that the damping ratio is decreased to be smaller or equal to 0.066, the rising edge of the clock generated maintains to be in the middle of the data eye. The reason for this is that the low pass filter output will be stable only when $RC \gg T_{change}$, where T_{change} refers to the maximum period of the change at the low pass filter input. When the system is locked, the rising edge of the clock signal is in the middle of the data eye, and the X 2 programmer output has a rising edge / falling edge

in the middle of the data eye, as shown in Fig. 2.14. Hence, for each data, no matter whether it is logic “1” or logic “0”, the output of the phase detector (the input of the low pass filter) is at logic “1” for half of the data period and at logic “0” at the other half of the data period. So, we can conclude, the maximum T_{change} equals to twice the data period. So, the product of RC has to be much larger than twice of the data period.

To make our case as an example, twice the data input period equals to $3.3e^{-9}$ seconds. Based on our simulation results, to ensure the system remaining locked when the data input is present, the product of the R value and the C value needs to be at least $5e^{-8}$. So, for our clock and data recovery system to function properly, the product of the R value and the C value needs to be at least 10 times larger than the maximum T_{change} of the low pass filter input, which is twice the data period. Consequently, based on the data input frequency, the product of RC can be estimated. For our system, we have R value set to be 50K, and C value set to be 1000fF, yielding a product equals to $5e^{-8}$. As shown in Fig. 2.31, the clock rising edges are still in the middle of data eye, when the data pattern becomes relatively random. And, the data is nicely recovered as shown in Fig. 2.32. Also, different data input pattern with the same frequency are tested and when the system is locked, as shown in Fig. 2.33 and 2.26, the rising edge of the clock signal generated occurs in the middle of the data eye. Specifically, the clock signal is shown by the yellow lines and the data input is shown by the red lines.

Fig. 2.31. Clock generated when data is present with pattern 1($\zeta = 0.066$).

Fig. 2.32. Data recovered when $\zeta = 0.066$.

Fig. 2.33. Clock generated when data is present with pattern 2($\zeta = 0.066$).

Fig. 2.34. Clock generated when data is present with pattern 3($\zeta=0.066$).

Chapter 3

Theoretical Analysis of Circuit Damping Ratio, Natural Frequency and Lock Time Variation

As discussed in Chapter 2, the designed clock and data recovery circuit, as a second-order feedback system, is characterized by damping ratio ζ and natural frequency W_n . In the mathematical model of a clock and data recovery circuit, these two terms are derived through the characteristics equation of the system transport function. As all other performance-related criteria are derived based on ζ and W_n , the performance of a clock and data recovery circuit can be predicted if ζ and W_n are known.

Lock time, denoted as T_{Lock} , being one of the most important criteria of the circuit performance, refers to the time it takes the circuit to align the VCO output phase with the data input phase. In other words, it refers to the time it takes the VCO control voltage to stabilize within a certain range around the desired voltage – for our simulations, a $\pm 3\%$ range is set as the “lock range”. Therefore, when the VCO control voltage stabilizes

within $\pm 3\%$ of the desired voltage, we say the clock and data recovery circuit is locked.

As shown in Fig. 3.1, the phase of the VCO output does align with the phase of the data input when the VCO control voltage stabilizes within $\pm 3\%$ of the desired voltage

$$\left(\frac{VDD}{2}\right).$$

Fig. 3.1. Alignment between data input (data) and VCO output (clk).

With process variation taken into consideration, to investigate the variation of the clock and data recovery circuit, the variations of ζ and W_n are our first research focus. The reason is that the performance variation of a clock and data recovery circuit can be predicted, when the variation of these two terms is available. The variation of T_{Lock} is also investigated in this research due to its importance in determining the circuit performance. The methods used can be expanded to investigate other circuit performance criteria.

3.1 Sources of Variation and Assumptions

As discussed in Chapter 2, low pass filter is the brain of a clock and data recovery circuit. The R and C values directly decide whether or not the system works and how well it works. Also, in the mathematical model of a clock and data recovery circuit, the values of ζ , W_n and T_{Lock} are intimately related to the R and C values of the low pass filter. As a result, our research focus is on the variation of ζ , W_n and T_{Lock} introduced by the variation of the R and C values in the low pass filter. By doing this, we are investigating the variation of ζ , W_n and T_{Lock} caused by the variations of R and C that have the most significant impact on the whole circuit.

The major reason that causes the variation of R and C values is the process variation – Process variations are introduced by the fabrication process uncertainty, including intra-wafer, inter-wafer transistor and thermal variations. Obviously, the existence of process variation will become more and more prominent as feature size decreases. Eventually, these uncertainties will have significant effect on circuit performance.

Before simulating R and C values variation, a mathematical model representing each process variation is needed. However, the 0.18 μ m process variation data are unavailable to us. Hence, general assumptions are made. For this research, the R and C values are assumed to follow Gaussian (normal) distributions. The probability density function curve of a normal distribution is shown in Figure 3.2. According to probability theory, random data that follow normal distribution assume values within the interval between

$\mu - \sigma$ and $\mu + \sigma$ with 68.26% of probability, within the interval between $\mu - 2\sigma$ and $\mu + 2\sigma$ with 95.44% of probability, and within the interval between $\mu - 3\sigma$ and $\mu + 3\sigma$ with 99.74% of probability. We assume that each of R and C has $\pm 10\%$ variation at 3σ , i.e., the variation of R or C values will not exceed 10% with 99.74% confidence. When we conducted simulations on circuit performance variation caused by C , the value of C is assigned by the HSPICE command “.param C = gauss(1000fF, 0.1, 3)”. Similarly, when we perform simulations on variation caused by R , the value of R is assigned by the HSPICE command “.param R = gauss(50K, 0.1, 3)”. So, before the start of each simulation, random numbers which follow normal distribution with mean equal 1000fF/ 50K are generated and assigned to C / R , respectively. Further, each of R and C values has $\pm 10\%$ variation at 3σ .

Fig. 3.2. Probability density curve of a normal distribution.

3.2 Mathematical Models for Damping Ratio, Natural Frequency, and Lock Time

Based on the mathematical model of a clock and data recovery circuit and the assumption that process variation caused by R and C values follows Gaussian (normal) distribution, the theoretical characteristics of damping ratio, natural frequency and lock time are explored in this section. Specifically, the explanation in definition and the deduction in design equation for lock time are given. Note that formal definition and equation derivation of natural frequency and damping ratio have been discussed in Section 2.4. Also, theoretical probability density functions of these three parameters are derived and shown in several figures.

As discussed in Section 2.4, natural frequency W_n and damping ratio ζ can uniquely characterize a clock and data recovery circuit. The system behavior is decided by these two parameters. As discussed in Section 2.4, natural frequency is the oscillation frequency when damping is not present. Damping ratio indicates the damping effect on the oscillation frequency. The design equations for the damping ratio and natural frequency of our system are given by:

$$W_n = \sqrt{\frac{K_{PD}K_{VCO}}{2RC}}, \text{ and} \quad (2.18)$$

$$\zeta = \frac{1}{2} \sqrt{\frac{2}{RC K_{PD} K_{VCO}}}. \quad (2.19)$$

Lock time, as manifested by its name, refers to the time it takes for the system to lock at a certain data input. When the system is locked, the phase difference between the input data and the VCO output signal is $\frac{\pi}{2}$ and the phase detector filtered output is $\frac{VDD}{2}$. In the mathematical model of a clock and data recovery circuit, the lock time is decided by the natural frequency and its value is given by:

$$T_{Lock} = \frac{2\pi}{W_n}, \quad (3.1)$$

where T_{Lock} denotes the lock time. Plug in equation (2.18) into (3.1), we can further reduce equation (3.1) as:

$$\begin{aligned} T_{Lock} &= \frac{2\pi}{\sqrt{\frac{K_{PD}K_{VCO}}{2RC}}} \\ &= 2\pi \sqrt{\frac{2RC}{K_{PD}K_{VCO}}} \end{aligned} \quad (3.2)$$

3.3 Process Variation Analysis for Damping Ratio, Natural Frequency, and Lock Time

To explore the effect caused by process variation, the circuit behavior change caused by R value variation and C value variation are investigated. Specifically, when effects caused by R value variation are investigated, the R value is assumed to vary following a normal distribution and the C value is fixed. On the contrary, when effects caused by C value variation are investigated, the C value is assumed to vary following another normal

distribution and the R value is fixed. Under this assumption, the natural frequency, damping ratio and lock time also vary by following certain probability distribution functions, and the probability density function curve for each parameter is analyzed and shown by figures. Finally, as demonstrated by the Equations (2.18), (2.19) and (3.2), the symmetric relationship between R and C values in each equation is obvious. So we can only focus on the case where the C value varies by following a normal distribution and R value is fixed. With slight modification, the analysis results can be automatically applied to the case where the R value varies following another normal distribution and the C value is fixed.

(a) *Damping Ratio ζ*

- The R value is fixed, and the C value varies by following a normal distribution

Based on the Equation $\zeta = \frac{1}{2} \sqrt{\frac{2}{K_{PD}K_{VCO}RC}}$, if we have $K_1 = \frac{1}{2} \sqrt{\frac{2}{K_{PD}K_{VCO}R}}$,

then we can simplify the equation to be

$$\zeta = K_1 \sqrt{\frac{1}{C}} = K_1 \frac{1}{\sqrt{C}}. \quad (3.3)$$

From Equation (3.3), we can see that if C shows probability behavior due to process variation, so does ζ . Here we start from the assumption that the variation of C value follows a normal distribution. For convenience, the cumulative density function (CDF) of

ζ value is denoted as $F_{\zeta}(x)$ and the probability density function (PDF) of ζ is denoted as $f_{\zeta}(x)$. Based on the definition of CDF, we can get:

$$F_{\zeta}(x) = P(\zeta < x) \quad (3.4)$$

By plugging in Equation (3.3), Equation (3.4) becomes

$$F_{\zeta}(x) = P\left(K_1 \frac{1}{\sqrt{C}} < x\right) \quad (3.5)$$

Rearranging Equation (3.5), we can get:

$$\begin{aligned} F_{\zeta}(x) &= P\left(C > \left(\frac{K_1}{x}\right)^2\right) \\ &= 1 - P\left(C < \left(\frac{K_1}{x}\right)^2\right) \end{aligned} \quad (3.6)$$

Based on the definition of CDF, we can represent Equation (3.6) as:

$$F_{\zeta}(x) = 1 - F_C\left(\left(\frac{K_1}{x}\right)^2\right) \quad (3.7)$$

As we know, for a function $F(x)$, its PDF can be obtained by deriving its CDF. Based on this theorem, we can get:

$$f_{\zeta}(x) = F'_{\zeta}(x) \quad (3.8)$$

Plugging in Equation (3.7) to Equation (3.8), we can get:

$$f_{\zeta}(x) = \left[1 - F_C\left(\left(\frac{K_1}{x}\right)^2\right)\right]' \quad (3.9)$$

With the PDF of the C value, which is assumed to be a normal distribution, denoted as $f_C(x)$, we can further reduce the Equation (3.9):

$$\begin{aligned}
f_{\zeta}(x) &= - \left[F_C \left(\left(\frac{K_1}{x} \right)^2 \right) \right]' \\
&= - f_c \left(\left(\frac{K_1}{x} \right)^2 \right) \left(\left(\frac{K_1}{x} \right)^2 \right)' \\
&= f_C \left(\left(\frac{K_1}{x} \right)^2 \right) \frac{2K_1^2}{x^3}
\end{aligned} \tag{3.10}$$

To visualize this analytical result, a MATLAB program was developed. With the help of a built-in function “normpdf” in MATLAB, the theoretical PDF of ζ is shown in Fig. 3.3. We can see that, under the assumption that process variation in the C value is limited within 10% of the mean with 99.74% of confidence, the PDF curve of ζ looks like a symmetric bell and with high probability that data assumes values around the mean. Also, since the value of C will be larger than 90% (i.e., $u - 3\sigma$) of the mean and smaller than 110% (i.e., $u + 3\sigma$) of the mean with 99.74% of confidence, based on the Equation (3.3), we can get the conclusion that the value of damping ratio will vary within $\frac{1}{\sqrt{110\%}}$ (95%) and $\frac{1}{\sqrt{90\%}}$ (105%) of the mean with 99.74% of confidence. In other words, the damping ration will vary within 5% of the mean theoretically with 99.74% of confidence.

Fig. 3.3. PDF of damping ratio (Theory) with R value fixed.

- The C value is fixed, and the R value varies by following a normal distribution.

As discussed before, in the mathematical model of damping ratio, R and C values are symmetric. Therefore, by interchanging the R and C symbols, the theoretical analysis conducted for process variation in C can be used as the theoretical analysis for R . Specifically, with CDF of R value denoted as $F_R(x)$ and PDF of R value denoted as $f_R(x)$, the CDF of ζ , $F_\zeta(x)$ is now given by:

$$F_\zeta(x) = 1 - F_R\left(\left(\frac{K_2}{x}\right)^2\right) \quad (3.11)$$

Thus the PDF of ζ , $f_\zeta(x)$ becomes:

$$f_{\zeta}(x) = f_R\left(\left(\frac{K_2}{x}\right)^2\right) \frac{2K_2^2}{x^3} \quad (3.12)$$

where $K_2 = \frac{1}{2} \sqrt{\frac{2}{K_{PD} K_{VCO} C}}$.

Similarly, a MATLAB program was developed to visualize this analytical result – the PDF curve is shown in Fig. 3.4. From Fig. 3.3 and Fig. 3.4, we can see the PDF curves match exactly with each other. So, the variation of damping ratio will also be limited within 5% of the mean with 99.74% of confidence. Conceptually, this demonstrates the symmetric relationship between the R and C values in the mathematical model of the damping ratio. Theoretically, this can be proved based on the PDF of each of the R distribution and the C distribution.

Fig. 3.4. PDF of damping ratio (Theory) with C value fixed.

Here, for convenience, m_C is used to represent the constant value 1000fF and m_R is used to represent the constant value 50K Ω . With our assumption, when the R value is fixed and equals to m_R , the C value varies following a normal distribution whose mean is m_C and the standard deviation σ_C , is given by:

$$\sigma_C = 10\% \ m_C / 3 \quad (3.13)$$

Similarly, when the C value is fixed and equals to m_C , the R value varies following a normal distribution whose mean is m_R and the standard deviation σ_R , is given by:

$$\sigma_R = 10\% \ m_R / 3 \quad (3.14)$$

By assigning $K = \frac{1}{2} \sqrt{\frac{2}{K_{PD}K_{VCO}}}$, we can represent K_1 in Equation (3.10) and K_2

In Equation (3.12) as:

$$K_1 = \frac{K}{\sqrt{m_R}} \text{ and } K_2 = \frac{K}{\sqrt{m_C}} \quad (3.15)$$

Thus, when the C value varies following a normal distribution and the R value is fixed at m_R , the PDF of damping ratio, given by Equation (3.10), can be represented in another form:

$$f_{\zeta}(x) = f_C \left(\left(\frac{K}{\sqrt{m_R} \bullet x} \right)^2 \right) \frac{2K^2}{m_R \bullet x^3} \quad (3.16)$$

Since the C value is assumed to vary following a normal distribution with mean equal m_C and standard deviation equal σ_C , the PDF of the C value can be written as:

$$f_C(x) = \frac{e^{-(x-m_C)^2 / 2\sigma_C^2}}{\sqrt{2\pi} \bullet \sigma_C} \quad (3.17)$$

Expand Equation (3.16) based on Equation (3.17), we can get:

$$f_{\zeta}(x) = \frac{e^{-\left(\left(\frac{K}{\sqrt{m_R \bullet x}}\right)^2 - m_C\right)^2 / 2\sigma_C^2}}{\sqrt{2\pi} \bullet \sigma_C} \frac{2K^2}{m_R \bullet x^3} \quad (3.18)$$

Plugging in Equation (3.13) to Equation (3.18), we can get:

$$f_{\zeta}(x) = \frac{e^{-\left(\left(\frac{K}{\sqrt{m_R \bullet x}}\right)^2 - m_C\right)^2 / 2(10\% \cdot m_C / 3)^2}}{\sqrt{2\pi} 10\% \cdot m_C / 3} \frac{2K^2}{m_R \bullet x^3} \quad (3.19)$$

Further reducing Equation (3.19), we can get:

$$f_{\zeta}(x) = \frac{e^{-\left(\left(\frac{K}{\sqrt{m_R \bullet m_C x}}\right)^2 - 1\right)^2 / 2(10\% / 3)^2}}{\sqrt{2\pi} 10\% / 3} \frac{2K^2}{m_C \bullet m_R x^3} \quad (3.20)$$

Similarly, when the R value varies following a normal distribution and the C value is fixed at m_C , the PDF of damping ratio, given by Equation (3.12), can be represented as:

$$f_{\zeta}(x) = f_R\left(\left(\frac{K}{\sqrt{m_C \bullet x}}\right)^2\right) \frac{2K^2}{m_C \bullet x^3} \quad (3.21)$$

Since the R value is assumed to vary following a normal distribution with mean equal m_R and standard derivation equal σ_R , we can get its PDF equation:

$$f_R(x) = \frac{e^{-(x-m_R)^2 / 2\sigma_R^2}}{\sqrt{2\pi} \bullet \sigma_R} \quad (3.22)$$

Expand Equation (3.21) based on Equation (3.22), we can get:

$$f_{\zeta}(x) = \frac{e^{-\left(\left(\frac{K}{\sqrt{m_C \bullet x}}\right)^2 - m_R\right)^2 / 2\sigma_R^2}}{\sqrt{2\pi} \sigma_R} \frac{2K^2}{m_C \bullet x^3} \quad (3.23)$$

Plugging in Equation (3.14) to Equation (3.23), we can get:

$$f_{\zeta}(x) = \frac{e^{-\left(\left(\frac{K}{\sqrt{m_C \bullet x}}\right)^2 - m_R\right)^2 / 2(10\% \cdot m_R / 3)^2}}{\sqrt{2\pi} 10\% \cdot m_R / 3} \frac{2K^2}{m_C \bullet x^3} \quad (3.24)$$

Further reducing Equation (3.24), we can get:

$$f_{\zeta}(x) = \frac{e^{-\left(\left(\frac{K}{\sqrt{m_R \bullet m_C x}}\right)^2 - 1\right)^2 / 2(10\% / 3)^2}}{\sqrt{2\pi} 10\% / 3} \frac{2K^2}{m_C \bullet m_R x^3} \quad (3.25)$$

By comparing Equation (3.20) with Equation (3.25), it is obvious that they are the same. Hence, we can conclude that the process variation in the R value and in the C value have the same effect on the damping ratio variation.

(b) *Natural Frequency* W_n :

- The R value is fixed, and the C value varies by following a normal distribution

Based on the Equation $W_n = \sqrt{\frac{K_{PD} K_{VCO}}{2RC}}$, if we have $K_3 = \sqrt{\frac{K_{PD} K_{VCO}}{2R}}$, we can

simplify the equation to be:

$$W_n = K_3 \frac{1}{\sqrt{C}} \quad (3.26)$$

From Equation (3.26), we can see that if C shows probability behavior due to process variation, so does W_n . Here we start from the assumption that the variation of C value follows a normal distribution. For convenience, the CDF of W_n value is denoted as

$F_{W_n}(x)$ and the PDF of W_n is denoted as $f_{W_n}(x)$. Based on the definition of CDF, we can get:

$$F_{W_n}(x) = P(W_n < x) \quad (3.27)$$

By plugging in Equation (3.26), Equation (3.27) becomes

$$F_{W_n}(x) = P\left(K_3 \frac{1}{\sqrt{C}} < x\right) \quad (3.28)$$

Rearranging equation (3.28), we can get:

$$\begin{aligned} F_{W_n}(x) &= P\left(C > \left(\frac{K_3}{x}\right)^2\right) \\ &= 1 - P\left(C < \left(\frac{K_3}{x}\right)^2\right) \end{aligned} \quad (3.29)$$

Based on the definition of CDF, we can represent Equation (3.29) as:

$$F_{W_n}(x) = 1 - F_C\left(\left(\frac{K_3}{x}\right)^2\right) \quad (3.30)$$

As discussed in the case of damping ratio, the PDF of natural frequency, $f_{W_n}(x)$, can be obtained by deriving its CDF, $F_{W_n}(x)$. Hence, we can get:

$$f_{W_n}(x) = F_{W_n}'(x) \quad (3.31)$$

Plugging in Equation (3.28) to Equation (3.29), we can get:

$$f_{W_n}(x) = \left[1 - F_C\left(\left(\frac{K_3}{x}\right)^2\right)\right]' \quad (3.32)$$

With the PDF of the C value, which is assumed to be a normal distribution, denoted as $f_C(x)$, we can further reduce the Equation (3.30):

$$\begin{aligned}
f_{W_n}(x) &= - \left(F_C \left(\left(\frac{K_3}{x} \right)^2 \right) \right)' \\
&= - f_C \left(\left(\frac{K_3}{x} \right)^2 \right) \left(\left(\frac{K_3}{x} \right)^2 \right)' \\
&= f_C \left(\left(\frac{K_3}{x} \right)^2 \right) \frac{2K_3^2}{x^3}
\end{aligned} \tag{3.33}$$

To visualize this analytical result, a MATLAB program was developed. With the help of a built-in function “normpdf” in MATLAB, the theoretical PDF of W_n is shown in Fig. 3.5. We can see that, under the assumption that process variation in the C value is limited within 10% of the mean with 99.74% of confidence, the PDF curve of W_n looks like a symmetric bell and with high probability that data assumes values around the mean. Also, since the value of C will be larger than 90% (i.e., $u - 3\sigma$) of the mean and smaller than 110% (i.e., $u + 3\sigma$) of the mean with 99.74% of confidence, based on the Equation (3.26), we can get the conclusion that the value of natural frequency will vary within $\frac{1}{\sqrt{110\%}}$ (95%) and $\frac{1}{\sqrt{90\%}}$ (105%) of the mean with 99.74% of confidence. In other words, the natural frequency will vary within 5% of the mean theoretically with 99.74% of confidence.

Fig. 3.5. PDF of natural frequency (Theory) with R value fixed.

- The C value is fixed, and the R value varies by following a normal distribution.

As discussed before, in the mathematical model of natural frequency, R and C values are symmetric. Therefore, by interchanging the R and C symbols, the theoretical analysis conducted for process variation in C can be used as the theoretical analysis for R . Specifically, with CDF of R value denoted as $F_R(x)$ and PDF of R value denoted as $f_R(x)$, the CDF of W_n , $F_{W_n}(x)$ is now given by:

$$F_{W_n}(x) = 1 - F_R\left(\left(\frac{K_4}{x}\right)^2\right) \quad (3.34)$$

Thus the PDF of W_n , $f_{W_n}(x)$ becomes:

$$f_{W_n}(x) = f_R \left(\left(\frac{K_4}{x} \right)^2 \right) \frac{2K_4^2}{x^3} \quad (3.35)$$

where $K_4 = \sqrt{\frac{K_{PD}K_{VCO}}{2C}}$.

Similarly, a MATLAB program was developed to visualize this analytical result – the PDF curve is shown in Fig. 3.6. From Fig. 3.5 and Fig. 3.6, we can see the PDF curves match each other. So, the variation of natural frequency is also limited within 5% of the mean with 99.74% of confidence. Conceptually, this demonstrates the symmetric relationship between the R and C values in the mathematical model of the natural frequency. Theoretically, this can be proved based on the PDF of the R value and C value.

Fig. 3.6. PDF of Natural Frequency (Theory) with C value fixed.

With m_C , m_R , σ_C , σ_R representing the values discussed above, if we reassign K as

$$K = \sqrt{\frac{K_{PD}K_{VCO}}{2}}, \text{ we can represent } K_3 \text{ in Equation (3.33) and } K_4 \text{ in Equation (3.35)}$$

as:

$$K_3 = \frac{K}{\sqrt{m_R}} \text{ and } K_4 = \frac{K}{\sqrt{m_C}} \quad (3.36)$$

From Equation (3.15) and Equation (3.36), we can see that K_1 and K_3 , K_2 and K_4 have the same format, respectively. Thus, following the procedure used to prove the symmetric relationship between the R value and the C value in the mathematical model of damping ratio, we can prove the R value and the C value have the same effect on natural frequency variation too. Specifically, in both cases, where the R value varies following a normal

distribution with the C value fixed and the C value varies following a normal distribution with the R value fixed, the PDF of natural frequency can be represented by the same equation:

$$f_{W_n}(x) = \frac{e^{-\left(\left(\frac{K}{\sqrt{m_R \bullet m_C x}}\right)^2 - 1\right) / 2(10\% / 3)^2}}{\sqrt{2\pi} 10\% / 3} \frac{2K^2}{m_C \bullet m_R x^3} \quad (3.37)$$

(c) On Lock Time T_{Lock} :

- The R value is fixed, and the C value varies by following a normal distribution

Based on the Equation $T_{Lock} = 2\pi \sqrt{\frac{NRC}{K_{PD}K_{VCO}}}$, if we have $K_5 = 2\pi \sqrt{\frac{NR}{K_{PD}K_{VCO}}}$, then

we can simplify the equation to be

$$T_{Lock} = K_5 \sqrt{C} . \quad (3.38)$$

From Equation (3.38), we can see that if C shows probability behavior due to process variation, so does T_{Lock} . Here we start from the assumption that the variation of C value follows a normal distribution. For convenience, the CDF of T_{Lock} value is denoted as $F_{T_{Lock}}(x)$ and the PDF of T_{Lock} is denoted as $f_{T_{Lock}}(x)$. Based on the definition of CDF, we can get:

$$F_{T_{Lock}}(x) = P(T_{Lock} < x) \quad (3.39)$$

By plugging in Equation (3.38), equation (3.39) becomes

$$F_{T_{Lock}}(x) = P(K_5 \sqrt{C} < x) \quad (3.40)$$

Rearranging Equation (3.40), we can get:

$$\begin{aligned}
 F_{T_{Lock}}(x) &= P(K_5 \sqrt{C} < x) \\
 &= P\left(C < \left(\frac{x}{K_5}\right)^2\right)
 \end{aligned} \tag{3.41}$$

Based on the definition of CDF, we can represent Equation (3.41) as:

$$F_{T_{Lock}}(x) = F_C\left(\left(\frac{x}{K_5}\right)^2\right) \tag{3.42}$$

As discussed in the case of damping ratio, the PDF of lock time, $f_{T_{Lock}}(x)$, can be obtained by deriving its CDF, $F_{T_{Lock}}(x)$. Hence, we can get:

$$f_{T_{Lock}}(x) = \left(F_{T_{Lock}}(x)\right)' \tag{3.43}$$

Plugging in Equation (3.42) to Equation (3.43), we can get:

$$f_{T_{Lock}}(x) = \left[F_C\left(\left(\frac{x}{K_5}\right)^2\right)\right]' \tag{3.44}$$

With the PDF of the C value, which is assumed to be a normal distribution, denoted as $f_C(x)$, we can further reduce the Equation (3.44):

$$\begin{aligned}
 f_{T_{Lock}}(x) &= f_C\left(\left(\frac{x}{K_5}\right)^2\right)\left(\left(\frac{x}{K_5}\right)^2\right)' \\
 &= f_C\left(\left(\frac{x}{K_5}\right)^2\right)\frac{2x}{K_5^2}
 \end{aligned} \tag{3.45}$$

To visualize this analytical result, a MATLAB program was developed. With the help of a built-in function “normpdf” in MATLAB, the theoretical PDF curve of T_{Lock} , $f_{T_{Lock}}(x)$, is shown in Fig. 3.7. We can see that, under the assumption that process variation in the C value is limited within 10% of the mean with 99.74% of confidence, the PDF curve of T_{Lock} looks like a symmetric bell and with high probability that data assumes values around the mean. Also, since the value of C will be larger than 90% (i.e., $u - 3\sigma$) of the mean and smaller than 110% (i.e., $u + 3\sigma$) of the mean with 99.74% of confidence, based on the Equation (3.38), we can get the conclusion that the value of lock time will vary within $\sqrt{90\%}$ (95%) and $\sqrt{110\%}$ (105%) of the mean with 99.74% of confidence. In other words, the lock time will vary within 5% of the mean theoretically with 99.74% of confidence.

Fig. 3.7. PDF of lock time (Theory) with R value fixed.

- The C value is fixed, and the R value varies by following a normal distribution.

As discussed before, in the mathematical model of lock time, R and C values are symmetric. Therefore, by interchanging the R and C symbols, the theoretical analysis conducted for process variation in C can be used as the theoretical analysis for R . Specifically, with CDF of R value denoted as $F_R(x)$ and PDF of R value denoted as $f_R(x)$, the CDF of T_{Lock} , $F_{T_{Lock}}(x)$ is now given by:

$$F_{T_{Lock}}(x) = f_R\left(\left(\frac{x}{K_6}\right)^2\right) \quad (3.46)$$

Thus, the PDF of T_{Lock} , $f_{T_{Lock}}(x)$ becomes:

$$f_{T_{Lock}}(x) = f_R\left(\left(\frac{x}{K_6}\right)^2\right) \frac{2x}{K_6^2}, \quad (3.47)$$

where $K_6 = 2\pi\sqrt{\frac{NC}{K_{PD}K_{VCO}}}$.

Similarly, a MATLAB program was developed to visualize this analytical result – the PDF curve is shown in Fig. 3.8. From Fig. 3.7 and Fig. 3.8, we can see the PDF curves match each other. So, the variation of lock time is also limited within 5% of the mean with 99.74% of confidence. Conceptually, this demonstrates the symmetric relationship between the R and C values in the mathematical model of the lock time. Theoretically, this can be proved based on the PDF of the R value and C value.

Fig. 3.8. PDF of Lock Time (Theory) with C value fixed.

With m_C , m_R , σ_C , σ_R representing the values discussed above, if we reassign K as

$$K = 2\pi \sqrt{\frac{N}{K_{PD}K_{VCO}}}, \text{ we can represent } K_5 \text{ in Equation (3.45) and } K_6 \text{ in Equation (3.47)}$$

as:

$$K_5 = K\sqrt{m_R} \text{ and } K_6 = K\sqrt{m_C} \quad (3.48)$$

Thus, when the C value varies following a normal distribution and the R value is fixed at m_R , the PDF of lock time, given by Equation (3.45), can be represented in another form:

$$f_{T_{Lock}}(x) = f_C\left(\left(\frac{x}{K\sqrt{m_R}}\right)^2\right) \frac{2x}{\left(K\sqrt{m_R}\right)^2} \quad (3.49)$$

Since the C value is assumed to vary following a normal distribution with mean equal m_C and standard deviation equal σ_C , the PDF of the C value can be written as Equation (3.17). For convenience, Equation (3.17) is listed below for reference:

$$f_C(x) = \frac{e^{-(x-m_C)^2 / 2\sigma_C^2}}{\sqrt{2\pi} \cdot \sigma_C} \quad (3.17)$$

Expand Equation (3.49) based on Equation (3.17), we can get:

$$f_{T_{Lock}}(x) = \frac{e^{-\left(\left(\frac{x}{K\sqrt{m_R}}\right)^2 - m_C\right)^2 / 2\sigma_C^2}}{\sqrt{2\pi}\sigma_C} \frac{2x}{\left(K\sqrt{m_R}\right)^2} \quad (3.50)$$

Plugging in Equation (3.13) to Equation (3.50), we can get:

$$f_{T_{Lock}}(x) = \frac{e^{-\left(\left(\frac{x}{K\sqrt{m_R}}\right)^2 - m_C\right)^2 / 2(10\% \cdot m_C / 3)^2}}{\sqrt{2\pi} 10\% \cdot m_C / 3} \frac{2x}{\left(K\sqrt{m_R}\right)^2} \quad (3.51)$$

Further reducing Equation (3.51), we can get:

$$f_{T_{Lock}}(x) = \frac{e^{-\left(\left(\frac{x}{K\sqrt{m_C \cdot m_R}}\right)^2 - 1\right)^2 / 2(10\% / 3)^2}}{\sqrt{2\pi} 10\% \cdot m_C \cdot m_R / 3} \frac{2x}{K^2} \quad (3.52)$$

Similarly, when the R value varies following a normal distribution and the C value is fixed at m_C , the PDF of lock time, given by Equation (3.47), can be represented as:

$$f_{T_{Lock}}(x) = f_R\left(\left(\frac{x}{K\sqrt{m_C}}\right)^2\right) \frac{2x}{\left(K\sqrt{m_C}\right)^2} \quad (3.53)$$

Since the R value is assumed to vary following a normal distribution with mean equal m_R and standard derivation equal σ_R , its PDF equation can be represented by Equation (3.22). For convenience, Equation (3.22) is listed below for reference.

$$f_R(x) = \frac{e^{-(x-m_R)^2 / 2\sigma_R^2}}{\sqrt{2\pi} \bullet \sigma_R} \quad (3.22)$$

Expand Equation (3.53) based on Equation (3.22), we can get:

$$f_{T_{Lock}}(x) = \frac{e^{-\left(\left(\frac{x}{K\sqrt{m_C}}\right)^2 - m_R\right)^2 / 2\sigma_R^2}}{\sqrt{2\pi} \bullet \sigma_R} \frac{2x}{\left(K\sqrt{m_C}\right)^2} \quad (3.54)$$

Plugging in Equation (3.14) to Equation (3.54), we can obtain:

$$f_{T_{Lock}}(x) = \frac{e^{-\left(\left(\frac{x}{K\sqrt{m_C}}\right)^2 - m_R\right)^2 / 2(10\% m_R / 3)^2}}{\sqrt{2\pi} 10\% m_R / 3} \frac{2x}{\left(K\sqrt{m_C}\right)^2} \quad (3.55)$$

Further reducing Equation (3.55), we can have:

$$f_{T_{Lock}}(x) = \frac{e^{-\left(\left(\frac{x}{K\sqrt{m_C} \bullet m_R}\right)^2 - 1\right)^2 / 2(10\% / 3)^2}}{\sqrt{2\pi} 10\% \bullet m_C \bullet m_R / 3} \frac{2x}{K^2} \quad (3.56)$$

By Comparing Equation (3.52) and Equation (3.56), it is obvious that they are exactly the same. Hence, we can conclude that process variation in the R value and in the C value have the same effect on the lock time variation.

Chapter 4

HSPICE Simulations of Damping Ratio, Natural Frequency and Lock Time Variation

In this chapter, design of experimental simulations will be described in detail. Also, to analyze the simulation results, C++ programs are developed and the algorithm for each program is provided and explained. Finally, based on the simulation results, normalized histograms of the parameters are generated and shown in several figures. Analysis and conclusions based on simulation results are given.

4.1 R C Value Assignment in HSPICE Simulations

As discussed in Chapter 2, the focus of this research is to investigate the variation of ζ , W_n and T_{Lock} caused by variation of R and C . The values of R and C are assumed to vary according to normal distributions and the variation is $\pm 10\%$ at 3σ , i.e., with 99.74% of confidence that the R or C variation will not exceed 10%. Specifically, when

the performance variation introduced by the variation of R is investigated, the value of C is fixed and the R value is assigned by the command “.param R = gauss(50K, 0.1, 3)”. So, before every simulation, random numbers are generated based on the Gaussian distribution whose mean equals to $50K\Omega$ and assigned to R . Similarly, when the performance variation introduced by the variation of C is investigated, the value of R is fixed and the C value is assigned by the command “.param C = gauss(1000fF, 0.1, 3)”. In other words, before every simulation, random numbers are generated based on the Gaussian distribution whose mean equals to $1000fF$ and assigned to C . For each purpose, one thousand times HSPICE monte carlo simulations are conducted.

4.2 Simulation Results and Analysis

4.2.1 Damping Ratio ζ

4.2.1.1 Calculation of Damping Ratio ζ

At each simulation, the value of VCO control voltage is recorded as V_{CTRL} . As shown in Fig. 4.1, the VCO control voltage starts to oscillate when the data input is present and the control voltage eventually stabilizes at the desired voltage. To calculate ζ , we need to study two peaks in the oscillation of the VCO control voltage after it starts to oscillate around the desired voltage. In the CAD tool developed, the first two peaks

occur after 200ns are investigated. By doing this, we can make sure that the peaks studied occur after the VCO control voltage starts to oscillate around the desired voltage.

Fig. 4.1. Change of VCO control voltage in clock and data recovery circuit.

Fig. 4.2 shows part of the change in VCO control voltage. From Fig. 4.2, it is easy to find out the time t_1 and t_2 when V_{CTRL} reaches its first peak and second peak V_{CTRL-1} and V_{CTRL-2} after 200ns. For example, in the simulation shown in Fig. 4.2, we have $t_1 = 215\text{ns}$, $t_2 = 272\text{ns}$, $V_{CTRL-1} = 2.9\text{V}$, and $V_{CTRL-2} = 2.75\text{V}$.

Fig. 4.2. Part of the change in VCO control voltage.

In the clock and data recovery circuit mathematical model, the oscillation of the VCO control voltage is represented as [31-32]:

$$V_{VCO-Control} = Ae^{-at} \cos(wt + \theta) + V_{Stable}, \quad (4.1)$$

where $a = \zeta W_n$, $w = W_n \sqrt{1 - \zeta^2}$. Note that the variable V_{Stable} is added here by us due to the observation that the oscillation of the VCO control voltage has a V_{Stable} bias. As discussed in Chapter 2, w refers to the true oscillation frequency of the VCO control voltage and its value is given by

$$w = 2\pi f_{V_{VCO-Control}} = 2\pi \frac{1}{T_{V_{VCO-Control}}} = 2\pi \frac{1}{t_2 - t_1} \quad (4.2)$$

where $f_{V_{VCO-Control}}$ is the oscillation frequency of the VCO control voltage, while $T_{V_{VCO-Control}}$ is the VCO control voltage oscillation period.

Similarly, a , as the decay speed of the VCO control voltage, can be derived through Equation (4.1) by

$$\begin{aligned} \frac{Ae^{-at_1}}{Ae^{-at_2}} &= \frac{e^{-at_1}}{e^{-at_2}} \\ &= e^{a(t_2 - t_1)} \\ &= \frac{V_{CTRL-1} - V_{Stable}}{V_{CTRL-2} - V_{Stable}} \end{aligned} \quad (4.3)$$

Based on equations (4.3), we can get:

$$a = \frac{\log_e \left(\frac{V_{CTRL-1} - V_{Stable}}{V_{CTRL-2} - V_{Stable}} \right)}{t_2 - t_1} \quad (4.4)$$

Due to the fact of $a = \zeta W_n$, we have

$$\zeta W_n = \frac{\log_e \left(\frac{V_{CTRL-1} - V_{Stable}}{V_{CTRL-2} - V_{Stable}} \right)}{t_2 - t_1} \quad (4.5)$$

Similarly, since we have $w = W_n \sqrt{1 - \zeta^2}$ and Equation (4.2), we can get:

$$W_n \sqrt{1 - \zeta^2} = 2\pi \frac{1}{t_2 - t_1} \quad (4.6)$$

If we further simplify Equations (4.5) and (4.6), ζ can be represented using t_1 , t_2 ,

V_{CTRL-1} and V_{CTRL-2} as follows:

$$\zeta = \frac{\sqrt{\left(\frac{\log_e \left(\frac{V_{CTRL-1} - V_{Stable}}{V_{CTRL-2} - V_{Stable}} \right)}{t_2 - t_1} \right)^2}}{\sqrt{\left(\frac{\log_e \left(\frac{V_{CTRL-1} - V_{Stable}}{V_{CTRL-2} - V_{Stable}} \right)}{t_2 - t_1} \right)^2 + \left(\frac{2\pi}{t_2 - t_1} \right)^2}} \quad (4.7)$$

With the record of V_{CTRL} , we can find out the values of t_1 , t_2 , V_{CTRL-1} and V_{CTRL-2} for each simulation. Based on Equation (4.7), we can get the value of ζ from the data generated in HSPICE. Using these two steps, a CAD tool is developed in C++ to calculate the value of ζ for each simulation.

4.2.1.2 Derivation of Normalized Histogram of ζ

According to [33], a normalized histogram is usually used to estimate its corresponding probability density function. Hence, based on the simulation results, normalized histograms can be generated. Here, the ζ value range is divided into equal fields, and the number of ζ values located in each field is normalized. The number of ζ in each field divided (normalized) by the product of the number of simulations and the width in each field. The normalized histogram derived from the results of simulations with the R value fixed is shown in Fig. 4.3, while the normalized histogram derived from the results of simulations with the C value fixed is shown in Fig. 4.4.

Fig. 4.3. Normalized histogram of damping ratio (simulation results) with R value fixed.

Fig. 4.4. Normalized histogram of damping ratio (simulation results) with C value fixed.

4.2.1.3. Summary

From Fig. 4.3 and Fig. 4.4, we can get the following summaries:

- The shape of the PDF curve obtained from the simulation results matches the shape of the theoretical PDF curve well. In other words, the variation of damping ratio caused by R or C variation can be predicted by theoretical analysis.
- For simulations with R value fixed and C value fixed respectively, the PDF curves

derived from the simulations are very similar in shape with basically the same mean and the same value range. As proved in Chapter 2, in the mathematical model of damping ratio, the variations in the C value and in the R value have the same effect on the variation of damping ratio. We can see that as predicted by the model, the variations in the C value and in the R value do cause the same effect on the variation of damping ratio. Also, the theoretical value of the mean (0.0645) in damping ratio is of the same order as the value obtained through simulation results (0.055). Hence, the mathematical model can be used to predict the order of the ζ value and its distribution.

- According to our assumption, when the effect caused by the variation of the value of C (R) is simulated, the value of C (R) varies following a normal distribution whose standard deviation is equal to 10% of the mean value. In other words, with 99.74% of probability, the variation of the C (R) value is within 10% of the mean. As discussed in Chapter 3, based on the mathematical model of damping ratio, with 99.74% of probability, the value of the damping ratio will vary within 5% of the mean. A C++ program was developed to analyze the simulation results, and the analysis results show that for simulations in which the value of C varies following a normal distribution and the value of R is fixed, the variation of the damping ratio is within 32.48% of the mean with 99.74% of probability. For simulations in which the value of R varies following a normal distribution and the value of C is fixed, the variation of the damping ratio is within 33.82% of the mean with 99.74% of probability. These two numbers are very close to each

other, which once again prove the effects caused by the variation of the R value and the C value are the same. Also, we can see that the variation of the damping ratio in simulations is much larger than the variation predicted by the theory. Specifically, it is five times larger. As a matter of fact, the simulation results are much closer to the reality than the theoretical prediction, since many more parameters are taken into consideration in Hspice simulation than in mathematical model. Hence, we can predict that the variation in reality is around five times larger than the variation predicted by the model.

4.2.2 Natural Frequency W_n

4.2.2.1 Calculation of Natural Frequency W_n by Simulation

As shown in Equation (4.8), damping ratio ζ can be represented by t_1 , t_2 , V_{CTRL-1} and V_{CTRL-2} . Similarly, through Equations (4.6) and (4.7), we can get a representation of W_n in terms of t_1 , t_2 , V_{CTRL-1} and V_{CTRL-2} as follows:

$$W_n = \sqrt{\left(\frac{\log_e \left(\frac{V_{CTRL-1} - V_{Stable}}{V_{CTRL-2} - V_{Stable}} \right)}{t_2 - t_1} \right)^2 + \left(\frac{2\pi}{t_2 - t_1} \right)^2} \quad (4.8)$$

From the V_{CTRL} curve for each simulation, we can find out the t_1 , t_2 , V_{CTRL-1} , V_{CTRL-2} . Now, by Equation (4.8), we can calculate the value of W_n . For this purpose, a CAD tool is developed in C++ in order to calculate the value of W_n for each simulation.

4.2.2.2 Normalized Histogram of W_n

Based on the simulation results, normalized histograms can be generated. The W_n value range is divided into equal fields, and the number of W_n values located in each field is normalized. The number is then divided by the product of the number of simulations and the width of each field. The normalized histogram derived from the results of simulations with R (C) value fixed is shown in Fig. 4.5, 4.6.

Fig. 4.5. Normalized histogram of natural frequency (simulation results) with R value fixed.

Fig. 4.6. Normalized histogram of natural frequency (simulation results) with C value fixed.

4.2.2.3 Conclusion Based on Simulation Results

From Fig. 4.4 and Fig. 4.5, we can get the following conclusions:

- Based on the observation that the PDF curve obtained from the simulation and the theoretical PDF curve derived from the mathematical model of natural frequency have the same shape, we can conclude that the variation of natural frequency caused by R or C values variation match the theoretical prediction. Also, the PDF curve derived from the two sets of simulations share very similar shape with

basically the same mean and the same value range. This also match our conclusion in Chapter 2 that the variation of the value of R and the value of C cause same effect on the variation of natural frequency.

- Comparing the order of the mean of natural frequency calculated through simulation results ($1.215\text{E}+8$) with the order of the mean of natural frequency ($1.55\text{E}+8$) derived through mathematical model, we can conclude that the mathematical model of natural frequency can be used to predict the order of the W_n value.
- In Chapter 3, we already reached a conclusion that according to the mathematical model of natural frequency, with our assumption of process variations in the value of R and the value of C , the value of natural frequency varies within 5% of the mean with 99.74% of probability. Based on the simulation results, a program developed to analyze the data shows that when the natural frequency variation caused by the variation of C value is simulated, the variation of the natural frequency varies within 7.21% of the mean with 99.74% of probability. When the natural frequency variation caused by the variation of R value is simulated, the variation of the natural frequency varies within 7.08% of the mean with 99.74% of probability. From this results, it is obvious that the effects caused by the variation of the R value and the C value are basically the same, and the variation of the natural frequency in simulations can be predicted by the theory.

4.2.3 Lock Time T_{Lock}

4.2.3.1 Calculation of Lock Time T_{Lock} Based on Simulation

To calculate T_{Lock} , the definition of “lock time” and specification used need to be explained. Based on the concept, the clock and data recovery circuit is “locked”, when and only when the VCO output phase align with the data input phase. In reality, the VCO output phase will oscillate around the data input phase within a range, and the range will get smaller and smaller as time goes by, and the depth of “lock” increases. For this thesis, we specify that if the VCO control voltage oscillates around the desired voltage within $\pm 3\%$ of range, we say the system is in lock.

With this specification, an algorithm is designed to find out the lock time, when V_{CTRL} stabilizes within $\pm 3\%$ of V_{Stable} for 100ns. We say the circuit is locked and the first time point in this 100ns is recorded down as T_{Lock} . The reason for this specification is that, if V_{CTRL} can be within $\pm 3\%$ of V_{Stable} for 100ns, it is oscillating around the V_{Stable} in a very small range and this range is getting smaller and smaller (equal to or smaller than $\pm 3\%$ of V_{Stable}). So, it is obvious that the system is in locked. To save the computation time, we can stop here and can record the first time point of this 100ns as the lock time. A CAD tool is developed in C++ to implement this algorithm.

4.2.3.2 Normalized Histogram of T_{Lock}

Based on the simulation results, normalized histograms are generated. The T_{Lock} value range is divided into equal fields, and the number of T_{Lock} values located in each field is normalized. The number is divided by the product of the number of simulations and the width of each field. The normalized histogram derived from the results of simulations with R (C) value fixed is shown in Fig. 4.7, 4.8.

Fig. 4.7. Normalized histogram of lock time (simulation results) with R value fixed.

Fig. 4.8. Normalized histogram of lock time (simulation results) with C value fixed.

4.2.3.3 Conclusions

From Fig. 4.7 and Fig. 4.8, we can get the following conclusions:

- From Fig. 4.7 and 4.8, we can see that the mathematical model of lock time provides correct prediction concerning the variation of lock time caused by the variation of the R value and the C value. The shape of the PDF curves obtained from the simulation results match the shape of the theoretical PDF curves. Also, as predicted by the model, the variations in the R value and in the C value cause

the same effect in the lock time value. That is, for both sets of simulations, the PDF curves derived for lock time have basically the same mean and the same value range.

- No matter whether it is for simulations with R value fixed or C value fixed, the theoretical mean of T_{Lock} is 40ns, while the mean of T_{Lock} based on simulation results is 450ns. The mean calculated from simulation results is roughly 10 times larger than the theoretical mean. We believe there are at least two reasons for this:
 - The mathematical model used in HSPICE takes a lot more into consideration than the simplified model proposed in [30]. So the lock time calculated by HSPICE is much larger than the one calculated by the mathematical model proposed in [30]. We believe the HSPICE simulation results are much more accurate than the one predicted by the simplified mathematical model proposed in [30].
 - Another reason that causes this discrepancy comes from the definition of “lock time”. For our research, we set the lock time as the time from which the VCO control voltage oscillates within $\pm 3\%$ around V_{Stable} . From Fig. 3.1, it is obvious that from about 100 ns, the VCO control voltage starts to oscillate about V_{Stable} . We can roughly regard 100ns as the lock time since from this time point on, the phase of the VCO output is oscillating around the phase of the data input, and the oscillation range is getting smaller and smaller.

The definition itself is in fact very strict. By this definition, the system is regarded to be in locked status only when the VCO control voltage is within $\pm 3\%$ of V_{Stable} , i.e., the phase of the VCO output is oscillating around the data input phase within a very small range. This definition makes sure that the VCO output phase strictly aligns with the data input. So, depending on the specification of “lock time”, we can get very different values for the lock time for each simulation. If we are using a very relaxed specification, 100ns can be regarded as lock time and it will be only two times of the theoretical value.

Finally, we conclude that the mathematical model proposed in [30] for lock time can be used to predict the variation of the lock time caused by variation of R or C . However, it cannot be used to predict the value of the lock time in reality, as demonstrated by the HSPICE results.

- As discussed in Chapter 3, theoretically, the value of lock time varies within 5% of the mean with 99.74% of probability. A C++ program was developed to analyze the results. The analysis results show that for simulations in which the value of C varies following a normal distribution and the value of R is fixed, the variation of the lock time varies within 23.47% of the mean with 99.74% of probability. For simulations in which the value of R varies following a normal distribution and the value of C is fixed, the variation of the lock time is within 19.66% of the mean with 99.74% of probability. Obviously, the effects caused by

the variation of the R value and the C value are basically the same. Also, the variation of lock time in simulations is three times larger than the theoretical prediction. As a matter of fact, the simulation results are much closer to the reality than the theoretical prediction, since many more parameters are taken into consideration in Hspice simulation than in mathematical model. Hence, we can predict that the variation in reality is around three times larger than the variation predicted by the model.

Chapter 5

Conclusion and Future Research

We have simulated the variation in the damping ratio, natural frequency and lock time of a clock and data recovery circuit caused by process variations in the brain of the circuit, the low pass filter. Specifically, the variation in the circuit damping ratio, natural frequency and lock time caused by the variation of the R value and the C value in the low pass filter are simulated. The mathematical models of these three parameters are analyzed, and the theoretical variations of these three parameters are derived based on the assumption of the R and C process variations. Some important conclusions are reached in this research:

- For damping ratio, natural frequency and lock time, the PDF curves derived through the simulation results match the PDF curves derived from their mathematical models in shape. This demonstrates that the trend of the variation of these three parameters matches the mathematical prediction. Specifically, the PDF function curves derived from the simulation results and from the mathematical models all look like symmetric bells. With high probability, each parameter of the clock and data recovery circuit assumes values around the mean.

- With the assumption that the value of R and the value of C both vary following normal distributions whose standard deviation equal to 10% of the corresponding mean value, we prove that the variations of damping ratio, natural frequency and lock time caused by the variations of the R value and the C value are the same. This is proved by theory. Also, as shown in figures in the thesis, the PDF function curves derived from simulations where the value of R varies following a normal distribution and the value of C is fixed, and the PDF function curves derived from the simulations where the value of C varies following a normal distribution and the value of R is fixed, match exactly with each other. These figures visualize the conclusion that with our assumption on the process variation in the value of R and the value of C , the variations in damping ratio, natural frequency and lock time caused by the variations of the R value and the C value are the same.
- According to the mathematical model, based on our assumption that with 99.74% of probability, the variation of the R value and the C value is limited within 10% of the mean, the variation of damping ratio, natural frequency and lock time is within 5% of the mean value with 99.74% of probability. Based on the simulation results, the variations of these three parameters caused by the variations of the R value and the C value are larger than the theoretical prediction, especially for the cases of damping ratio and lock time. Specifically, with 99.74% of probability, for damping ratio, the variation is within 32.48% of

the mean when the effect caused by the variation of C value is simulated, and 33.82% of the mean when the effect caused by the variation of R value is simulated. Hence, for damping ratio, the variation by simulation is six times larger than the mathematical prediction. For natural frequency, with 99.74% of probability, the variation is limited within 7.21% of the mean when the effect caused by the variation of C value is simulated, and 7.08% of the mean when the effect caused by the variation of R value is simulated. Hence, for natural frequency, the variation by simulation is basically the same as the mathematical prediction. For lock time, with 99.74% of probability, the variation is limited within 23.47% of the mean when the effect caused by the variation of C value is simulated, and 19.66% of the mean when the effect caused by the variation of R value is simulated. Hence, for lock time, the variation by simulation is about five times larger than the mathematical prediction.

- Based on simulation results, the mean value of damping ratio (natural frequency) is of the same order as the value predicted by the theory. So we can use the mathematical model to predict the mean of damping ratio (natural frequency). As for the lock time, it depends on the specification of “lock time”. If the specification is “when the clock signal starts to oscillate around the middle of the data eye”, the lock time in simulation is of the same order as the mathematical predication. According to our specification – the oscillation of the VCO control voltage is within $\pm 3\%$ of the desired voltage, the lock time by simulation is ten times larger than the predication. The reason is that our

specification is very strict. We want to make sure the clock signal is oscillating within a very small range around the middle of the data eye, before we regard the system as in “locked”.

The future research work includes: 1) investigating the variations of the damping ratio, natural frequency and lock time caused by the combination of the variations of the R value and the C value; 2) investigating the variation of other parameters of the clock and data recovery circuit with the method we developed in this thesis; 3) design a test strategy to filter out all chips whose performances vary out of certain boundaries based on the characteristics of the parameter variations.

Reference

1. B. Razavi, "Challenges in the design high-speed clock and data recovery circuits," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 94- 101, Aug. 2002.
2. K. Kishine, N. Ishihara, K. Takiguchi, and H. Ichino, "A 2.5-Gb/s clock and data recovery IC with tunable jitter characteristics for use in LANs and WANs," *IEEE Journal of Solid-State Circuits*, vol. 34, no. 6, pp. 805-812, June 1999.
3. 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.
4. 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.
5. Z. O. Gursoy, and Y. Leblebici, "Design and realization of a 2.4 Gbps – 3.2 Gbps clock and data recovery circuit using deep submicron digital CMOS technology," *Proceedings of IEEE International System on Chip Conference*, pp. 99-102, Sept. 2003.
6. Y. Liu, G. Ning, H. Yang, and H. Wang, "A novel clock and data recovery scheme based on sigma-delta quantization," *6th International Conference on ASIC*, vol. 1, pp. 436-440, Oct. 2005.
7. C. H. Jiun, T. Z. A. Zulkifli, Z. A. A. Aziz, and N. M. Noh, "A monolithic 622 Mb/S half rate clock and data recovery circuit utilizing a novel linear phase detector," *IEEE Region 10 Conference*, vol. 4, pp. 348-351, Nov. 2004.

8. A. Athavale, and C. Christensen, *High-speed serial I/O made simple*. [Online]
Available: <http://www.xilinx.com/publications/books/serialio/serialio-book.pdf>
9. K. A. Bowman, X. Tang, J. C. Eble, and J. D. Meindl, "Impact of extrinsic and intrinsic parameter variations on CMOS system on a chip performance," *Proc. 12th Annu. IEEE Int. ASIC/SOC Conference*, pp. 267-271, Sept. 1999.
10. G. Higashi, and T. Lill, *Identifying front-end challenges for 90nm design*.
[Online].Available:
http://www.eetasia.com/ART_8800316159_480200_a22c2a5f200309.HTM
11. R. Yang, K. Chao, and S. Liu, "A 200-Mbps/spl sim/2-Gbps continuous-rate clock-and-data-recovery circuit," *IEEE Transactions on Circuits and Systems*, vol. 53, no. 4, pp. 842 – 847, Apr. 2006.
12. C. Yan, C. Lee, and Y. Lee, "A 1.25Gb/s half-rate clock and data recovery circuit," *IEEE VLSI-TSA International Symposium on VLSI Design, Automation and Test*, pp. 116-119, Apr. 2005.
13. J. Weng, M. T. Tsai, J. M. Lin, C. Y. Yang, "A 1.8 Gb/s burst-mode clock and data recovery circuit with a 1/4 –rate clock technique," *IEEE International Symposium on Circuits and Systems*, pp. 4, May 2006.
14. K. Vichienchom, and W. Liu, "Analysis of phase noise due to bang-bang phase detector in PLL-based clock and data recovery circuits," *Proceedings of the International Symposium on Circuits and Systems*, vol. 1, pp. I-167 – I-620, May 2003.

15. M. Ramezani, C. Andre, and T. Salama, "Jitter analysis of a PLL-based CDR with a bang-bang phase detector," *45th Midwest Symposium on Circuits and Systems*, vol. 3, pp. III-393 – III 396, Aug. 2002.
16. J. Lee, K. S. Kundert, and B. Razavi, "Analysis and modeling of bang-bang clock and data recovery circuits," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 9, pp. 1571 – 1580, Sept. 2004.
17. A. Tajalli, P. Muller, M. Atarodi, and Y. Leblebici, "Analysis and modeling of jitter and frequency tolerance in gated oscillator based CDRs," *IEEE International Symposium on Circuits and Systems*, vol. 39, no. 9, pp. 1571 – 1580, Sept. 2004.
18. Y. Ohtomo, T. Kawamura, K. Nishimura, M. Nogawa, H. Koizumi, and M. Togashi, "A 12.5Gb/s CMOS BER test using a jitter-tolerant parallel CDR," *IEEE International Solid-State Conference*, vol. 1, pp. 174-520, Feb. 2004.
19. D. Hong, C.-K. Ong, and K.-T. Cheng, "Bit-error rate estimation for high-speed serial links," *IEEE Transactions on Circuits and Systems I*, vol. 53, no. 12, pp. 2616-2627, Dec. 2006.
20. D. S. Hong, and M. N. El-Gamal, "An efficient design & verification workflow for evaluating the proper operation of clock and data recovery systems," *48th Midwest Symposium on Circuits and Systems*, vol. 2, pp. 1827-1830, Aug. 2005.
21. D. Hong, and K. Cheng, "Bit error rate estimation for improving jitter testing of high-speed serial links," *IEEE International Test Conference*, pp. 1-10, Oct. 2006.
22. M. H. Perrott, Y. Huang, R. T. Baird, B. W. Garlepp, D. Pastorello, E. T. King, Q. Yu, B. D. Kasha, P. Steiner, L. Zhang, J. Hein, and B. Del Signore, "A 2.5-Gb/s

- multi-rate 0.25- μ m 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.
23. Y. M. Greshishchev, and P. Schvan, “SiGe clock and data recovery IC with linear-type PLL for 10-Gb/s SONET application,” *IEEE Journal of Solid-State Circuits*, vol. 35, no. 9, pp. 1353 – 1359.
 24. J. Savoj, and B. Razavi, “10-Gb/s CMOS clock and data recovery circuit with a half-rate linear phase detector,” *IEEE Journal of Solid-State*, vol. 36, no. 5, pp. 761-768, May 2001.
 25. S. Byun, J. C. Lee, J. H. Shim, K. Kim, and H.-K. Yu, “A 10-Gb/s CMOS CDR and DEMUX IC with a quarter-rate linear phase detector,” *IEEE Journal of Solid-State Circuits*, vol. 41, no. 11, pp. 2566-2576, Nov. 2006.
 26. X. Chen, and M. M. Green, “A CMOS 10 Gb/s clock and data recovery circuit with a novel adjustable K/sub pd/phase detector,” *Proceedings of the 2004 International Symposium on Circuits and Systems*, vol. 4, pp. IV-301-4, May 2004.
 27. Y. Tang, and R. L. Geiger, “Phase detector for PLL-based high-speed data recovery,” *Electronics Letters*, vol. 38, no. 23, pp. 1417-1419, Nov. 2002.
 28. A. Hati, M. Ghosh, and B. C. Sarkar, “Phase detector for data-clock recovery circuit,” *Electronics Letters*, vol. 38, no. 4, pp. 161 – 163, Feb. 2002.
 29. S. M. Alavi, and O. Shoaiei, “A 2.5-Gb/s clock and data recovery circuit with a 1/4 – rate linear phase detector,” *The 17th, International Conference on Microelectronics*, pp. 4, Dec. 2005.

30. R. Jacob Baker, *CMOS circuit design, layout and simulation*, Second Edition, Wiley-IEEE Press, Danvers, Massachusetts, 2005.
31. G. H. Hostetter, *Design of feedback control systems*, CBS College Publishing, New York, New York, 1982.
32. J. Vande Vegte, *Feedback control systems*, Prentice-Hall, New Jersey, New Jersey, 1986.
33. *Engineering statistics handbook*. [Online]. Available:
<http://www.itl.nist.gov/div898/handbook/eda/section3/histogra.htm>