IOWA STATE UNIVERSITY
Digital Repository

Graduate Theses and Dissertations

Iowa State University Capstones, Theses and Dissertations

2017

Accurate Jitter Decomposition in High-Speed Links

Yan Duan
Iowa State University

Follow this and additional works at: https://lib.dr.iastate.edu/etd

 Part of the Electrical and Electronics Commons

Recommended Citation
Duan, Yan, "Accurate Jitter Decomposition in High-Speed Links" (2017). *Graduate Theses and Dissertations*. 16276.
https://lib.dr.iastate.edu/etd/16276

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Graduate Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.

www.manaraa.com

Accurate jitter decomposition in high-speed links

by

Yan Duan

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Electrical Engineering

Program of Study Committee:
Degang Chen, Major Professor
Randall L Geiger
Chris Chong-Nuen Chu
Yong Guan
Meng Lu

Iowa State University

Ames, Iowa

2017

Copyright © Yan Duan, 2017. All rights reserved.

www.manaraa.com

DEDICATION

To my family

www.manaraa.com

TABLE OF CONTENTS

www.manaraa.com

www.manaraa.com

ACKNOWLEDGMENTS

This dissertation cannot be completed without support of many people. First of all, I would like to express my deepest gratitude to my advisor Dr. Degang Chen for his valuable guidance, constructive suggestions, and patient encouragement at Iowa State University. Dr. Chen greatly helps me in developing both academic and personal skills.

I would also like to thank my committee members, Dr. Randall L Geiger, Dr. Chris Chong-Nuen Chu, Dr. Yong Guan and Dr. Meng Lu for their valuable e orts and suggestions on my dissertation. Thanks Dr. Jihong Ren, Dr. Wenyi Jin, Dr. Ruth Wei Li in Altera, Dr. Yuming Cao, Mr. Liang Gu in Huawei, Dr. Mike Peng Li in Intel for their technical assistance to my research. Thanks Dr. Yunfeng Li and Ms. Debbie Younkin for reviewing my dissertation and improving my English in the past a few years. I would also like to thank my classmate Dr. Yifei Li for helping me in various aspects during my time at Iowa State University. I would like to thank Rui Zhou and Tao Chen who loved me.

My special thanks go to my beloved family members. My parents, Yubing Zhou, my sisters, and my lovely nephews and niece gave me a lot of support in my study and my live. I do not think I can finish this without their endless encouragement and support.

www.manaraa.com

ABSTRACT

In a high-speed digital communication system, jitter performance plays a crucial role in Bit-Error Rate (BER). It is important to accurately derive each type of jitter as well as total jitter (TJ) and to identify the root causes of jitter by jitter decomposition. In this work, we propose new jitter decomposition techniques in high-speed links testing. The background of jitter decomposition is described in chapter 1.

In chapter 2, duty cycle distortion jitter amplification is introduced. As channel loss results in both ISI and jitter amplification, DCD amplification is a big concern in high-speed links. The derivation of a formula of DCD amplification for data channels is included and the calculation result matches the time-domain simulation in the system.

Chapter 3 provides an accurate jitter decomposition algorithm using Least Squares (LS) which simultaneously separates ISI, RJ, and PJ. A new time domain ISI model is proposed, which is faster and more accurate than the conventional ISI model. This algorithm obtains the estimated individual jitter component value with fine accuracy by using less samples of total jitter data compared with conventional methods. The simulation and measurement show the accuracy and efficiency of this algorithm with less data samples.

In chapter 4, a low-cost comparator-based jitter decomposition algorithm is proposed. Instead of using TIE jitter sequence to decompose, it uses a low cost and simple com parator network to identify the deviation of current sampling positions from the ideal sampling positions to represent the TIE. It simultaneously separates ISI, DCD, and PJ and can achieve similar accuracy compared to the instrument test. Both the simulation and measurement show the decomposition algorithm with great accuracy and efficiency.

www.manaraa.com

In chapter 5, a low cost and simple dithering method to improve the test of linearity of analog-to-digital converter (ADC) is proposed. This method exhibits an improve ment and enhancement for the ultrafast segmented model identification of linearity error (uSMILE) algorithm which reduces 99% of the test time compared to the conventional method. In this study, we proposed three types of distribution dithering methods adding to the ramp input signal to reduce the estimation error when uSMILE was applied in low resolution ADCs. The fix pattern distribution was proved as the most efficient and cost-effective method by comparing with the Gaussian, uniform, and fix-pattern distributions. Both the simulation results and hardware measurement indicate that the estimation error can be significantly reduced in 12-bit SAR ADC with effective dithering.

www.manaraa.com

CHAPTER 1. INTRODUCTION

The data processing capabilities of integrated circuits increased greatly due to scaling of integrated circuit technology in the past decade. However, digital data communication has become one of the dominant reasons which cause bottlenecks in these systems. To overcome these bottlenecks, more sophisticated high-speed link circuits have replaced those simple input/output (I/O) drivers integrated in these chips. The aggressive scaling of I/O bandwidth demands double every two to three years on average [1] as shown in Figure 1.1. However, the I/O performance is limited by electrical channel bandwidth limits.

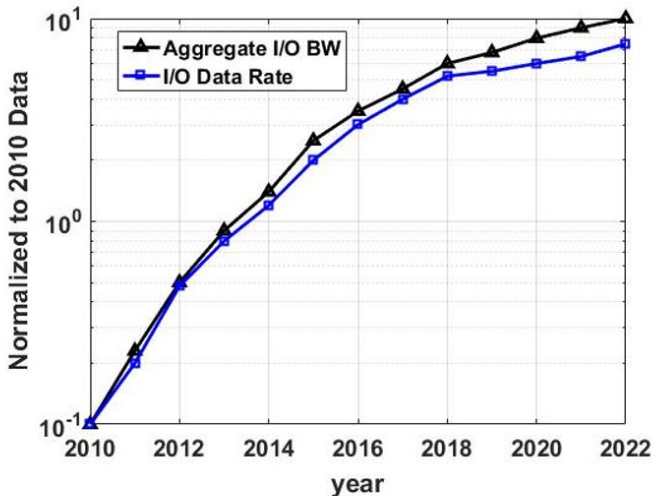

Figure 1.1 I/O data rate and bandwidth requirement

Meanwhile, as the data rate increases, the inter symbol interference (ISI) becomes

www.manaraa.com

severe due to the bandwidth limitation. Jitter and noise, generated inevitably in the transmitter, channel and receiver, impact the performance of the system.

1.1 Motivation

A typical high-speed link composes of the three main blocks, transmitter, channel and receiver [2-3] as shown in Figure 1.2.

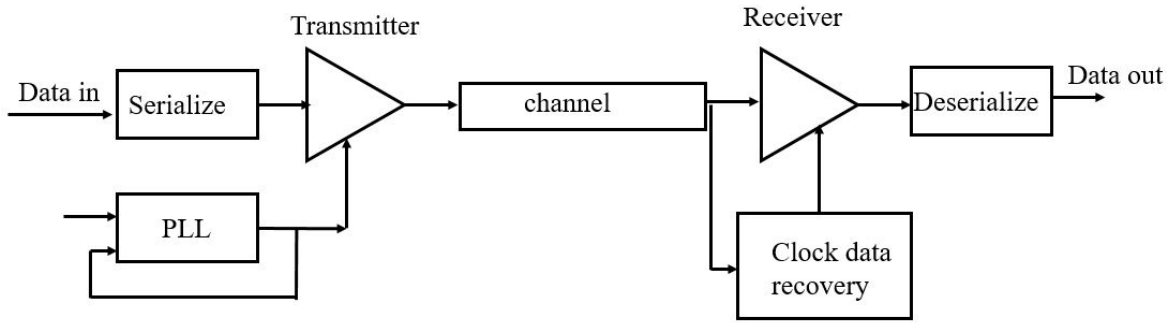

Figure 1.2 I/O High-speed link block diagram

The transmitter (TX) buffer is triggered by phase-locked loop (PLL). A high-frequency transmit clock is generated by PLL. The channel is the whole path from the output of the transmitter to the input and then to the receiver. The channel includes any printed circuit board (PCB) trace and coaxial cables that are used to connect the packages together. Signal degradation is caused by the channel due to a low pass characteristic. It limits data throughput by introducing noise and signal distortions, which both lead to ISI jitter. The transmitted data voltage is sampled by the receiver (RX). A clock and data recovery (CDR) circuit on the receiver usually incorporates a PLL. Some additional circuits are needed to synchronize the receiver with the incoming data stream.

As a synchronization circuit, the RX-PLL has to cope with input jitter, and it pro-

www.manaraa.com

vides certain robustness against timing variations. Timing variations will not be tracked correctly by the PLL if jitter exceeds critical amplitude. And hence, it leads to erroneous signal recovery of the received data, which the whole process will affect the bit-error-rate (BER). Thus, it is important to look carefully into the jitter and noise sources in order to determine the maximum data rate of the link with the guaranteed BER target in high-speed links. Jitter decomposition analysis involves the use of histogram or statistical analysis, frequency analysis, time domain methods. In this dissertation, several jitter decomposition analysis methods are developed and analyzed. They are also applied to practical simulations and measurement cases. This chapter will present some background information about jitter in high-speed links before starting jitter analysis methods.

1.2 Jitter Basics

1.2.1 Jitter Definition

Jitter is defined as the variation of ideal edges and actual edges in time domain both for clock and data. Figure 1.3 shows the definition of jitter. TIE jitter is the actual deviation from the ideal clock period over all clock periods

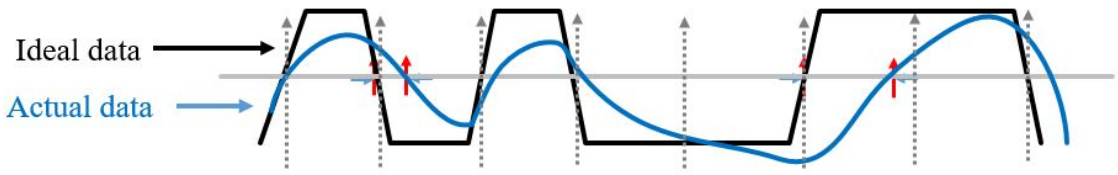

Figure 1.3 I/O High-speed link block diagram

www.manaraa.com

1.2.2 Jitter in High-Speed Link

Since timing uncertainty is the major reason for erroneous data recovery, a robust receiver architecture is one of the most challenging design criteria. Figure 1.4 shows various jitter sources contributing to the overall jitter in a high-speed link.

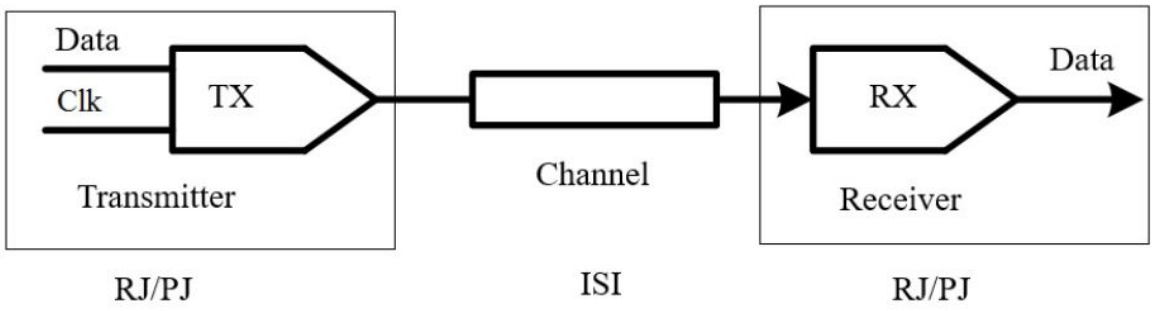

Figure 1.4 Different jitter within a typical serial link

As shown in Figure 1.4, there are two important sources of noise and jitter on the TX: the TX clock jitter and the TX power supply noise [2]. Any phase noise on the transmit clock will translate into timing jitter and the TX power supply noise is induced by the asynchronous current switching of all the I/Os, which may couple into the transmitted signal [2]. It can be lowered by careful layout and decoupling capacitors. Meanwhile, the RX also causes RX sampling jitter and RX power supply noise, which is similar to the TX end. However, the received signal is much more sensitive to the noise and jitter since the detected signal has a smaller voltage amplitude. The channel is the final element of a communication link that generates noise and jitter as illustrated in Figure 1.4. The channels limited bandwidth produces ISI, which varies the transition edge slope, inducing data-dependent jitter (DDJ)[2-3].

The eye diagram is a common way to highlight the problem of signal recovery and presence of jitter as depicted in Figure 1.5. Researchers use eye diagrams to quantify

www.manaraa.com

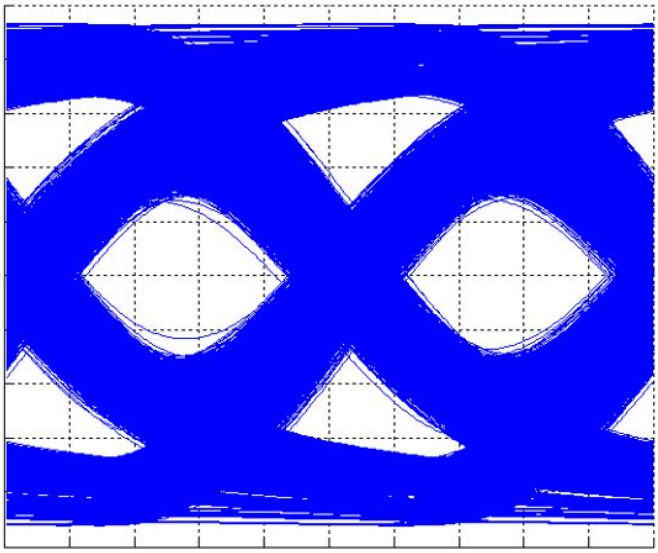

Figure 1.5 The effect of non idealities on an eye diagram

noise margin degradation. An ideal eye diagram shows a perfect rectangle. However, jitter and voltage noise cause the transition slope to be slower, thereby closing the eye opening. Jitter especially degrades system performance by causing a large horizontal eye closure voltage noise causes a large vertical eye closure.

An observed total jitter (TJ) distribution includes different components [2-3] shown in the scheme in Figure 1.6. It can basically be decomposed into a bounded deterministic jitter (DJ) and an unbounded random jitter (RJ). RJ is usually considered as Gaussian, unbounded probability behavior. It is observed at both distribution tails, extending them toward infinity. DJ can be of arbitrary shape and is expressed by various subcomponents in order to distinguish various root causes. DJ is further divided into periodic (PJ), bounded uncorrelated (BUJ), and data-dependent jitter (DDJ). DDJ is related to the transmitted data pattern. DDJ consists of duty-cycle distortion (DCD) and ISI. DCD is caused by a difference in the pulse width between logical high and low levels which is caused by voltage offsets or different rise and fall times at signal transitions. BUJ is always considered bounded because of the limited coupling strength.

www.manaraa.com

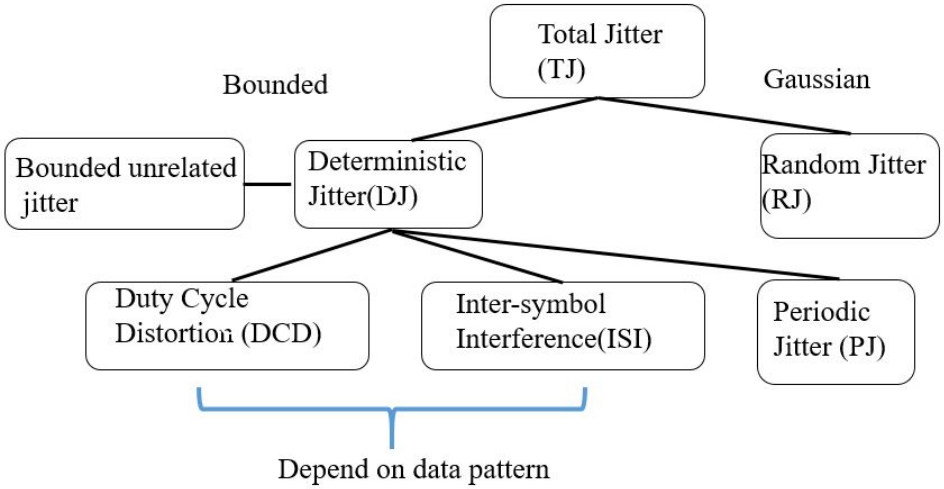

Figure 1.6 Jitter components classification

1.2.3 Jitter Decomposition Methods

With the increasing data rates in the high-speed communication system, there is a significant challenge in balancing test cost and quality to test high-speed interfaces. Currently available jitter measurement techniques require expensive and high precision measurement instruments, including the use of high-speed sampling scopes, time interval analyzers (TIAs) and bit error rate testers (BERTs) [2-3].

Different off-chip decomposition approaches have been developed in the past. Generally there are three popular category approaches to decompose jitter: 1) the ones based on histogram or statistical methods; 2) the ones with frequency-domain based analysis; and 3) the ones with time-domain based analysis relying on jitter measurements carried out in real-time. In the following sections these three analysis domains will be explained in more detail in order to give a comprehensive overview to the state-of-the-art in the topic.

www.manaraa.com

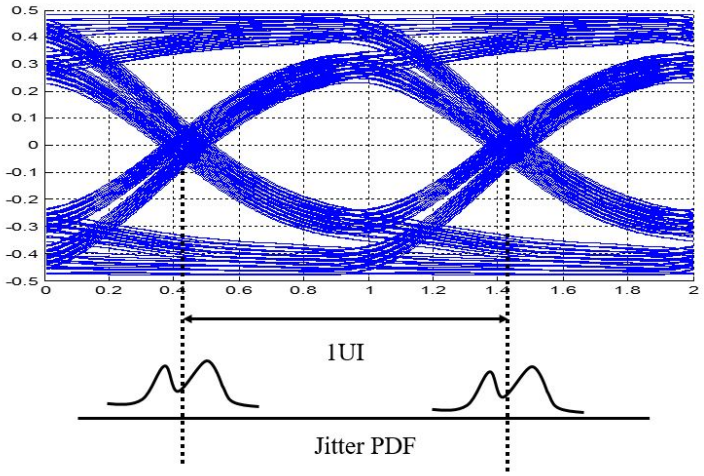

Figure 1.7 Histogram based method model

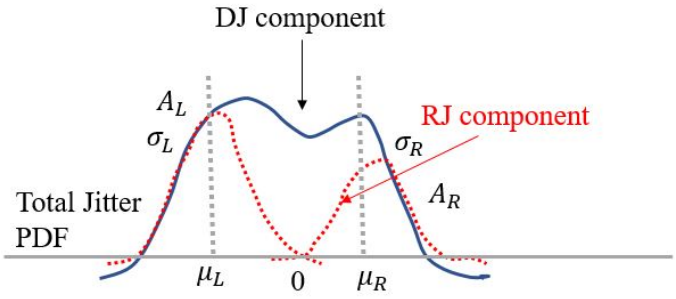

Figure 1.8 RJ and DJ components of a jitter PDF.

1.2.3.1 Histogram Based Analysis

Probability distributions of collected jitter values are used to estimate jitter influence in histogram or statistical domain-based methods. A jitter distribution in an eye diagram is obtained from the horizontal cross section at a desired signal level in Figure 1.7.

The collected distribution corresponds to probability density function (PDF) of jitter samples. Common model assumptions are the popular Gaussian tail model [2] which has the following features: 1) Jitter is a stationary random process. 2) The measured TJ distribution can be separated into two components, RJ and DJ. 3) RJ is observed at

www.manaraa.com

the outer tails of a TJ distribution, and follows an unbounded Gaussian by its mean , standard deviation and amplitude A. 4) DJ follows a finite, bounded distribution which is shown in Figure 1.8. Three model parameters μ, σ and A for both tails should be identified by analysis methods in order to correctly extrapolate a measured distribution. For this purpose, some researchers also refer these methods as tail fitting algorithms or jitter decomposition methods. Various methods were developed to decompose the RJ and DJ components with tail fitting algorithms [4-8]. However, there are two obvious drawbacks in tail fitting approaches. First, it needs to identify the tail part of the distribution before starting the optimization. The identification algorithm with conservative parameters works suboptimal for many of the distribution shapes. Secondly, the tail fitting algorithm is very complex and requires many samples.

1.2.3.2 Time-Domain Based Analysis

The time-domain based jitter analysis [8-10] rely on jitter measurements carried out in real-time. This is only possible for dedicated real-time measurement systems, such as high speed sampling scopes or TIAs. Some methods have been proposed in the past. Dou and Abraham [9-10] introduced correlation analysis. Unfortunately, this method still lacks a relation between extracted DJ subcomponents and the TJ. It is unable to apply to arbitrary jitter distributions and requires a large amount of test samples which is not a practical simulation application.

1.2.3.3 Frequency-Domain Based Analysis

Some research proposed the jitter decomposition based on the Fourier transform (FT) [11-13]. In frequency-domain analysis, the power spectral density (PSD) is used to represent the jitter spectrum by applying averaging techniques. Peaks in the spectrum can be interpreted as PJ or DDJ and the average noise floor denotes the power of RJ. In [9-10] four spectral regions of the jitter transfer function are defined to allow for

www.manaraa.com

BER analysis. The approach is restricted to Gaussian RJ combined with PJ. Although jitter measurements using external instruments can be performed in the lab for characterization, instruments are unable to lend themselves to fast parallel testing of devices with a large number of high-speed interfaces due to their hardware complexity, cost and scalability limitations.

1.3 Contributions of the Dissertation

In this dissertation, we will propose new jitter decomposition techniques that address the mentioned challenges above in high-speed links testing. The dissertation is organized with four major chapters, which each chapter presents and solves some technical issues in the area of jitter decomposition.

In chapter 2, duty cycle distortion (DCD) jitter amplification will be introduced. As channel loss results in both ISI and jitter amplification, DCD amplification is a big concern in high-speed links. An overview of the statistical jitter modeling and jitter amplification of clock channel will be briefly discussed. The derivation of a formula of DCD amplification for data channels will be included. The calculation result matches the time-domain simulation in the system.

Chapter 3 will provide an accurate jitter decomposition algorithm using Least Squares (LS) which simultaneously separates ISI, RJ, and PJ (called TIE-based method in the dissertation). Jitter basic information with PJ, DCD and RJ will be introduced. A new time-domain ISI model will be used in the algorithm and it is faster and more accurate than the conventional ISI model. This algorithm will obtain estimated individual jitter component values with fine accuracy by using less samples of total jitter data compared to conventional methods.

In chapter 4, a low-cost comparator based jitter decomposition algorithm will be presented. Instead of using TIE jitter sequence to decompose, it will use a low cost and

www.manaraa.com

simple comparator network to identify the deviation of current sampling positions from the ideal sampling positions to represent the TIE (called comparator-based method in this thesis). It can simultaneously separate ISI, DCD, and PJ and this algorithm achieves similar accuracy compared to the instrument test.

In chapter 5, a low cost and simple dithering method to improve the test of linearity of analog-to-digital converter (ADC) will be discussed. This method is an improvement and enhancement for the ultrafast segmented model identification of linearity error (uSMILE) algorithm, which reduces 99% of the test time compared to the conventional method. Since uSMILE produces large estimation errors in low resolution ADCs (10-12 bits) when the input is a ramp signal, in which the quantization noise of ADC becomes a dominant part in the total noise. We will compare with three types of distribution dithering methods to add the ramp input signal to reduce the estimation errors when uSMILE is applied in low resolution ADCs.

In chapter 6 concludes the whole dissertation.

www.manaraa.com

REFERENCE

[1] International Technology Roadmap for Semiconductors, 2011 edition, [Online]. Available: http://public.itrs.net

[2] M. P. Li, "Jitter, noise and signal integrity at high-speed," Prentice Hall Pearson Education, 2007

[3] D.Oh and X.Yuan, "High speed signaling: jitter modeling analysis, and budgeting," Prentice Hall Pearson Education, 2011, Chapter 3, pp.44-64

[4] J.-L. Huang, "A Random Jitter Extraction Technique in the Presence of Sinusoidal Jitter," IEEE Asian Test Symp. (ATS06), pp. 318326, Nov. 2006.

[6] R. Stephens, "Separation of Random and Deterministic Components of Jitter," U.S. Patent 7149638, Dec. 12, 2006.

[6] R. Stephens, "Separation of a Random Component of Jitter and a Deterministic Component of Jitter," U.S. Patent 7 191080, Mar. 13, 2007.

[7] S. Wisetphanichkij and K. Dejhan, "Jitter Decomposition by Derivatived Gaussian Wavelet Transform," IEEE Int. Symp. Communication and Information Technology (ISCIT04),vol. 2, pp. 11601165, Oct. 2004.

[8] F. Nan, Y. Wang, F. Li, W. Yang, and X. Ma, "A Better Method than Tail-fitting Algorithm for Jitter Separation Based on Gaussian Mixture Model," J. of Electronic Testing: Theory and Applications, vol. 25, no. 6, pp. 337342, Dec. 2009.

www.manaraa.com

[9] Q. Dou and J. Abraham, "Jitter Decomposition by Time Lag Correlation," IEEE Int. Symp. Quality Electronic Design (ISQED06), Mar. 2006

[10] "Physical Layer Performance: Testing the Bit Error Ratio (BER)," Technical Article, Maxim Inc., Sep. 2004

[11] H. Pang, J. Zhu, and W. Huang, "Jitter decomposition by fast Fourier transform and time lag correlation," in IEEE Int. Conf. Communications, Circuits and Systems ICCCAS, Jul. 2009, pp. 365368.

[12] T. Yamaguchi, H. Hou, K. Takayama, D. Armstrong, M. Ishida et al., "An FFT-based jitter separation method for high-frequency jitter testing with a 10x reduction in test time," IEEE Int. Test Conf. (ITC07), pp. 18, Oct. 2007.

www.manaraa.com

CHAPTER 2. DUTY CYCLE DISTORTION AMPLIFICATION IN HIGH-SPEED DATA CHANNELS

Clock channel jitter amplification scales exponentially with channel loss and is the worst for duty cycle distortion (DCD) due to its high frequency nature. As the channel loss results in both inter-symbol interference (ISI) and jitter amplification, the DCD amplification in data channel is not as well understood and clearly quantified as clock channel DCD amplification. This chapter presents a general formula to calculate the data channel DCD amplification based on peak distortion analysis and statistical jitter modeling methodology. The presented methodology is validated by time-domain simulation on different lossy channels.

2.1 Introduction

As data rates continue to climb at an ever-increasing rate, jitter and noise in high-speed links have become a performance bottleneck in addition to signal integrity issues such as inter-symbol interference (ISI) and crosstalk. ISI has three main causes: 1) bandwidth limitation of transmission medium; 2) nonlinear phase response of the transmission medium; 3) reflection. Random jitter (RJ) is commonly modeled by the Gaussian distribution function. Common sources of RJ include shot noise and thermal noise. Duty-cycle distortion (DCD) is caused different rise and fall times at signal transitions and device mismatch in signal path. Electromagnetic interference from other devices or the system can also induce current on signal wires and affect the signal voltage biasing and reference

www.manaraa.com

voltage.

Jitter exists in both receivers and transmitters in high-speed links. Compared to receiver jitter, transmitter jitter is more detrimental as transmitter jitter modulates transmitted pulse width directly. The amount of jitter is modulated by channel dispersion as signals propagate in the system. It has been shown that clock channel jitter amplification scales exponentially with channel loss even when the channel is linear, passive, and noiseless [1-4]. The mechanism of jitter amplification is discussed in terms of channel impulse/step response in [2-4]. In particular, DCD and RJ amplifications in clock signals are shown to scale uniquely with channel loss [2], indicating that loss is responsible for the effect. It is the worst amplification for duty-cycle distortion (DCD) as DCD has the highest jitter frequency contents [1-4]. DCD amplification presents a great limitation on clock forwarding architecture at high data rates. In modern high-speed serial links shown in Figure 2.1, the clock is typically embedded in high-speed data stream, and a clock and data recovery (CDR) circuit are used to recover both the data and clock. As channel loss results in both ISI and jitter amplification, DCD amplification in data channels is not as well understood and quantified.

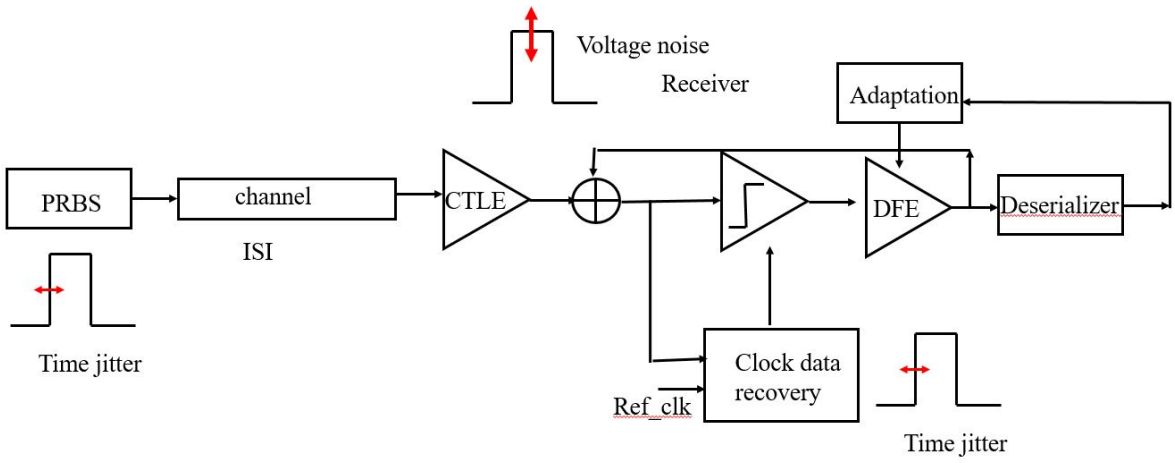

Figure 2.1 Typical high-speed link

www.manaraa.com

Among the typical mixed-signal equalization techniques such as transmitter finite impulse response (FIR) equalizer, analog continuous time linear equalizer (CTLE), and decision feedback equalizer (DFE), only CTLE can reduce the jitter amplification effect due to its continuous time nature. Due to the limitation of CTLE gain-bandwidth product, DFE is typically heavily-relied upon for high-loss channels. Therefore, it is important that the impact of transmitter jitter, especially transmitter DCD, is accurately estimated and accounted for timing budget for high-speed serial links.

DCD amplification in clock channels was well explained in frequency domain in [4]. A clock signal with DCD consists of two major frequency components, a DC component and the clock tone itself. The lossy passive channel attenuates the high frequency clock tone much more than the DC component and thus it results in DCD amplification shown in Figure 2.2.

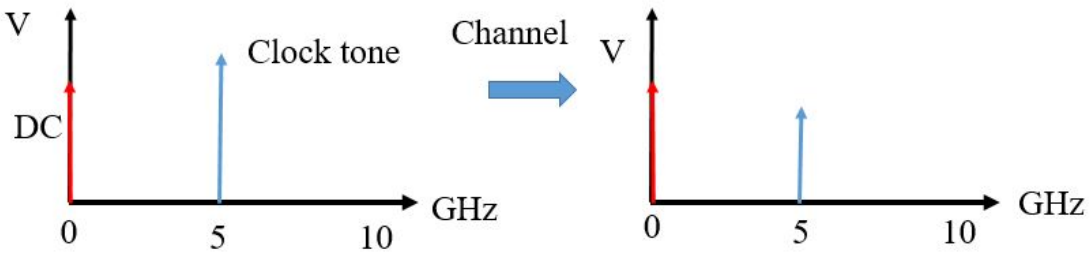

Figure 2.2 Frequency analysis for jitter amplification in clock pattern

In [2-3], a general statistical formulation was developed to model transmitter jitter amplification in clock channels based on channel step responses. Clock jitter was first converted into voltage noise and then converted back to timing jitter using the slope of the output clock signal. All the analysis and data in [1-4] show that clock channel DCD amplification scales exponentially with the channel loss.

The analysis in [1-4] was limited to DCD amplification in clock channels. The pe-

www.manaraa.com

riodicity of the 1010 clock pattern eliminates ISI jitter therefore jitter at the channel output is entirely induced by input jitter. For clock channels, jitter amplification factor is defined as the ratio of output jitter over input jitter. Since the channel loss results in both ISI jitter and jitter amplification, this definition of jitter amplification is no longer applicable to data channels. In contrast, what really matters in data channels is the additional margin loss due to transmitter jitter. DCD amplification in data channel is defined as the ratio of additional margin loss and input DCD. For example, if 3%UI DCD results in 6%UI additional margin loss (J0 J1) at the channel output compared with zero DCD case, the DCD amplification factor is 2, shown in Figure 2.3. It is important in balancing timing budget in high-speed serial links to insight how data channel DCD amplification scales with channel loss, especially for high data rate applications on high-loss channels.

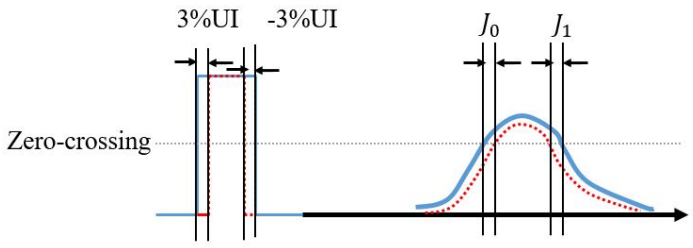

Figure 2.3 DCD amplification definition in data channel

This chapter extends the DCD amplification analysis for clock channels in [3] to data channels and presents a general formula to calculate the factor of data channel DCD amplification based on peak distortion analysis and statistical jitter modeling methodology. The rest of chapter is organized as follows. Section II describes DCD jitter amplification analysis based on a statistical modeling methodology for data channels and clock channels. Section III validates the methodology with time-domain simulation results. Section V summarizes this chapter.

www.manaraa.com

2.1.1 Jitter Amplification Background

This section first reviews channel single bit response, the statistical transmitter jitter modeling methodology presented in [3, 5] and peak distortion analysis [6-7] as they are the basis of this work. Then a general formula for data channel DCD amplification is derived.

2.1.1.1 Channel Single Bit Response

In a high-speed link, the channel has low-pass filter characteristic due to the skin-effect and dielectric loss. This means an ideal narrow pulse at the input of the channel will be significantly attenuated and much wider at the output of the channel. It occupies the pre-cursor $(h_{-1}, h_{-2}, \cdots)$ and post cursor samples $(h1, h2,, \cdots)$. Other than that, the first pre-cursor (h_{-1}) and post-cursor (h_1) samples are very large due to the pulse dispersion from low-pass filtering, h_0 is the main-cursor. Both effects would make it very difficult to correctly detect the bits that are transmitted in a sequence.

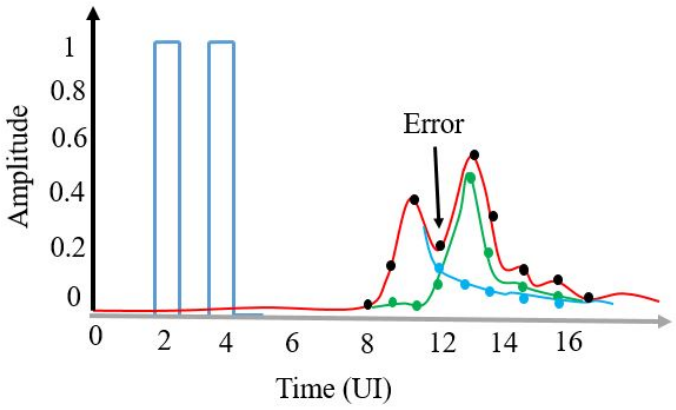

Figure 2.4 ISI cause bit error

For example, a received sample as shown in Figure 2.4 that corresponds to bit zero drops to only 0.4 due to interference from the previous bit by 0.2 and the next bit by 0.1 in a one zero-one pattern sent from the transmitter. As a result, this bit is received

www.manaraa.com

as an error.

This inter-symbol interference (ISI) effect is deterministic since it is repeated by transmitting the same data pattern. It is obvious that this effect becomes severe as the width of the transmitted bit decreases. As such, ISI is clearly one of the most significant effects that limit the achievable data rates in high-speed backplane links.

2.1.2 General Statistical System Jitter Modeling

Assuming linear time invariance (LTI) throughout the rest of this chapter, the channel output signal $y(t)$ without transmitter and receiver jitter can be expressed in terms of the superposition of progressively delayed channel step responses

$$y[n] = \sum_k (d_k - d_{k-1}) s(t - kT) \tag{2.1}$$

where $s(t)$ is the step response of the channel, k is the input symbol index, T is the symbol time, and d_k is the transmit symbol at time k. The step responses can be derived from the S-parameter of the passive channel.

With the transmitter jitter, ε_k^{TX}, the output of the channel becomes

$$y[n] = \sum_k (d_k - d_{k-1}) s(t + \varepsilon_k^{TX} - kT) \tag{2.2}$$

After sampling at $t = mT$, the output y_m can be expressed in terms of the superposition of progressively delayed channel step responses as follows [3, 5]:

$$y_m = \sum_k (d_k - d_{k-1}) s(\varepsilon_k^{TX} + (m-k)T) \tag{2.3}$$

For small transmitter jitter ε_k^{TX}, following the method in [9], the channel output signal can be decomposed into the ideal signal and the effective voltage noise n_m^{TX} induced by ε_k^{TX} using Taylor series expansion as follows:

$$y_m(t) = \sum_k (d_k - d_{k-1}) s((m-k)T) + \sum_k ((d_k - d_{k-1}) \varepsilon_k^{TX} h_{m-k}) \tag{2.4}$$

www.manaraa.com

where h_m is the data rate sampled impulse response of the channel. The effective voltage noise n_m^{TX} due to transmitter jitter at the input of the sampler is:

$$n_m^{TX} = \sum_k ((d_k - d_{k-1})\varepsilon_k^{TX} h_{m-k}) \tag{2.5}$$

Based on equation 2.4, data with small DCD ε_k^{TX} can be explained as the superposition of a noiseless data stream d_k in Figure 2.5 and two noise pulses ε_k^{TX} and ε_{k+1}^{TX} [8]. Since the DCD is much narrower than the impulse response of the channel filter and the reference symbol pulse, we can approximate them with delta functions as described in [8] when the noise data passes through the channel filter.

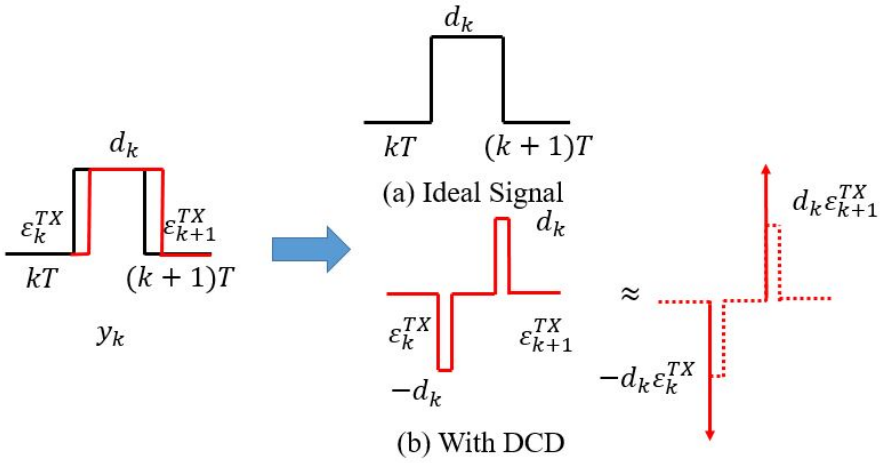

Figure 2.5 DCD converts to noise

A symbol transmitted with DCD is converted into a symbol with no jitter Figure 2.5(a) and a noise term where the width values of the noise symbols Figure 2.5(b) are equal to ε_k^{TX} and ε_{k+1}^{TX}.Therefore,the transmitter jitter can be mapped into effective noise n_m^{TX} shown Figure 2.6.

www.manaraa.com

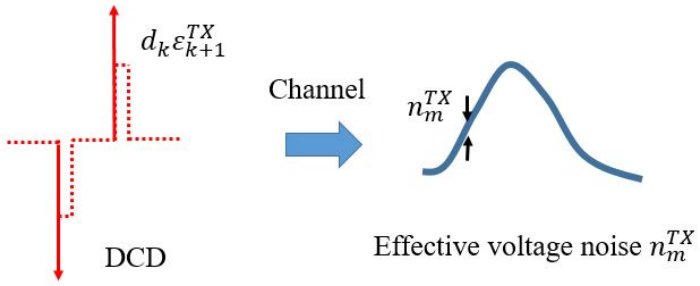

Figure 2.6 Data with DCD decomposition

2.2 DCD Jitter Amplification Analysis

This section described a general formula for data channel DCD amplification. The first step was to convert the receiver effective noise into zero-crossing jitter. The second step was to obtain the DCD amplification by using the worst-case pattern based on peak-distortion analysis. Meanwhile, we obtained the DCD amplification for clock pattern.

2.2.1 Converting Effective Noise to Jitter

Equation 2.5 shows that transmitter jitter on any one edge affects the channel output waveform in the vicinity of neighboring edges in a way determined by the channel impulse response and data pattern. The higher the channel loss is, the longer the channel impulse response and hence the larger the jitter accumulation will be. The effective voltage noise n_m^{TX} at the zero crossings can be translated back to timing jitter using the slope of the channel output waveform [3] S_m shown in Figure 2.7.

In Figure 2.7, jitter at zero-crossing which is converted by the effective noise n_m^{TX}, Jm is the timing jitter at mth zero-crossing. The jitter at the zero crossing is given by

$$J_M = n_m^{TX}/S_M \tag{2.6}$$

For any data pattern, the slope of the channel output at the zero crossing is given by

$$S_M = \sum_k ((d_k - d_{k-1})h_{m-k}) \tag{2.7}$$

www.manaraa.com

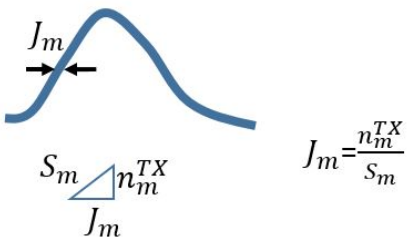

Figure 2.7 Converting the effective noise to jitter at zero-crossing

Mapping the effective voltage noise back to timing jitter,we obtained

$$J_M = \sum_k ((d_k - d_{k-1})\varepsilon_k^{TX} h_{m-k}) / \sum_k ((d_k - d_{k-1})h_{m-k}) \tag{2.8}$$

2.2.2 DCD Amplification of Data Channel

Note that both the slope and the effective voltage noise are various for different data patterns. The definition of jitter amplification as the ratio of output jitter over input jitter is no longer valid. In contrast, as mentioned earlier, what matters is the additional margin loss due to DCD for data channel. For a random bit pattern, peak distortion analysis is widely used to estimate worst-case eye opening [6-7]. The difference between the worst-case eye opening with and without transmitter DCD is a good estimation of the additional margin loss due to transmitter DCD.

To determine the worst-case voltage or timing margin, the worst-case received eye shape is extracted along with the peak sampling boundary. Since sources such as ISI have truncated distributions, the associated worst-case magnitudes can be direct from the single bit response of the system [9]. Based on peak-distortion analysis, the worst-case pattern that results in the worst-case eye opening can be easily extracted for a given channel single bit response (SBR). Given a SBR, the largest undershoot can be estimated by choosing the signs of the other data bits that result in negative (positive) ISIs for the case where current data bit is 1 (-1) [6-7]. Given the symbol-spaced single bit response,

www.manaraa.com

pulse cursor coefficient is given by

$$h = [\cdots, h_{-2}, h_{-1}, h_0, h_1, h_2, \cdots] \tag{2.9}$$

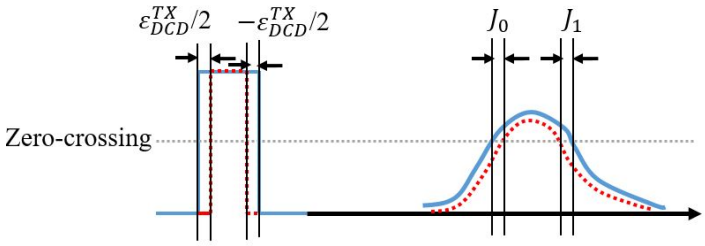

Figure 2.8 Single bit and worst case pattern

Flip pulse matrix about cursor h_0 and the bits are inverted sign of the pulse ISI, the worst-case pattern can be obtained by

$$\overrightarrow{d}^{T}_{worst} = [\cdots, -sign(h_2), -sign(h_1), 1, -sign(h_{-1}), -sign(h_{-2}), -sign(h_{-2}), \cdots] \tag{2.10}$$

For a smooth lossy channel where all ISIs are positive, its worst-case pattern is simply a single bit shown in Figure 2.8. For this worst-case data pattern, if the transmitter jitter at rising edge is (ε_0^{TX}) and at falling edge is ε_1^{TX} , the calculated jitter at the rising(J_0) and falling edges (J_1) of the single bit according to equation 2.8 after channel is:

$$J_0 = \frac{h_0\varepsilon_0^{TX} - h_{-1}\varepsilon_1^{TX}}{h_0 - h_{-1}} \tag{2.11}$$

$$J_1 = \frac{-h_0\varepsilon_1^{TX} + h_1\varepsilon_0^{TX}}{h_0 - h_1} \tag{2.12}$$

The additional margin loss due to jitter is the sum of J_0 and J_1 which is given by

$$\text{margin loss} = \frac{h_0^2 - h_1h_{-1}}{(h_0 - h_1)(h - h_{-1})}(\varepsilon_0^{TX} - \varepsilon_1^{TX}) \tag{2.13}$$

www.manaraa.com

For sinusoidal jitter, margin loss is the biggest when $\varepsilon_0^{TX} - \varepsilon_1^{TX}$ is at the maximum value. The higher the sinusoidal jitter frequency is, the bigger $\varepsilon_0^{TX} - \varepsilon_1^{TX}$ is for the same sinusoidal jitter magnitude, therefore the higher the margin loss.

DCD is the highest frequency sinusoidal jitter. For DCD, $\varepsilon_0^{TX} = -\varepsilon_1^{TX} = \varepsilon_{DCD}^{TX}/2$, where ε_{DCD}^{TX} is peak-to-peak transmitter DCD. From (13), the DCD amplification factor defined as additional margin loss over transmitter DCD is given by

$$A_{DCD} = (\frac{h_0 + h_{-1}}{h_0 - h_{-1}} + \frac{h_0 + h_1}{h_0 - h_1})/2 \tag{2.14}$$

Equation 2.14 shows that the DCD impact on margin loss is amplified when we have positive h_{-1} and h_1, which is the case for lossy channels. The higher the loss is, the bigger h_1 and h_{-1} are and the bigger the amplification factor is. Therefore, data channel DCD amplification increases with channel loss.

2.2.3 DCD Jitter Amplification of Clock channel

Clock channels can be considered as a data channel with a clock pattern: $\overrightarrow{d}^T = [\cdots, -1, +1, -1, \cdots]$, taking $\overrightarrow{d}^T$ into equation 2.8 we have [3]

$$J_M = \frac{\sum_k h_{m-k}\varepsilon_{DCD}^{TX}/2}{\sum_k (-1)^k h_{m-k}} \tag{2.15}$$

$$A_{DCD}^{CLK} = \frac{\sum_k h_{m-k}}{\sum_k (-1)^k h_{m-k}} \tag{2.16}$$

In contrast to the fact that equation 2.14 which only depends on h_1 and h_{-1},equation 2.16 shows that clock channel DCD amplification depends on all terms in the channel impulse response therefore it could be much more severe than data channel DCD amplification for high loss channels.

Alternatively, the difference between clock channel DCD amplification and data channel DCD amplification can be explained from jitter frequency point of view. As shown in

www.manaraa.com

[1-4], for clock channel, DCD is the highest frequency jitter which results in the biggest jitter amplification while random jitter with wide spectrum shows much less jitter amplification. The DCD amplification of clock pattern in frequency-domain is

$$F_{DCD} = (\frac{H(2w_0)}{H(w)} + \frac{H(0)}{H(-w_0)})/2 \tag{2.17}$$

where $H(\omega)$ is the transfer function of channel. When the impedance mismatch in the channel is negligible, $H(\omega)$ is the channel forward S-parameter. ω_0 is the fundamental frequency of clock signal. In a lossy channel, $H(\omega)$ decays exponentially with frequency. $H(\omega - \omega_0)$ is attenuated less than $H(\omega_0)$, producing a gain in the output that leads to jitter amplification. From equation 2.17, it shows that DCD amplification is caused by the attenuation difference between the DC component introduced by DCD and the fundamental in lossy channels. Different from clock channel DCD, data channel DCD is being modulated by a random data pattern. Let ε be a random sequence of 0s and 1s, where 0 means no edge transition and 1 means edge transition. The data channel DCD sequence can be modeled as the point-wise multiplication of the DCD sequence with the random sequence ε. This results in convolution in frequency domain. The random sequence ε has wide frequency content. Therefore, intuitive data channel DCD should result in less jitter amplification than clock channel DCD.

2.3 Simulation Results

To validate the theory, three channels with different amount of loss were used in time-domain simulation. They have 5dB, 9dB and 11dB insertion losses at 5Gzh, respectively as shown in Figure 2.9 The time-domain test bench was constructed in the Simulink environment. The simulation time step was swept to make sure the time step is fine enough to simulate small jitters. All simulations are done at 10Gbps. These channels have eye openings at 10Gbps without equalization.

www.manaraa.com

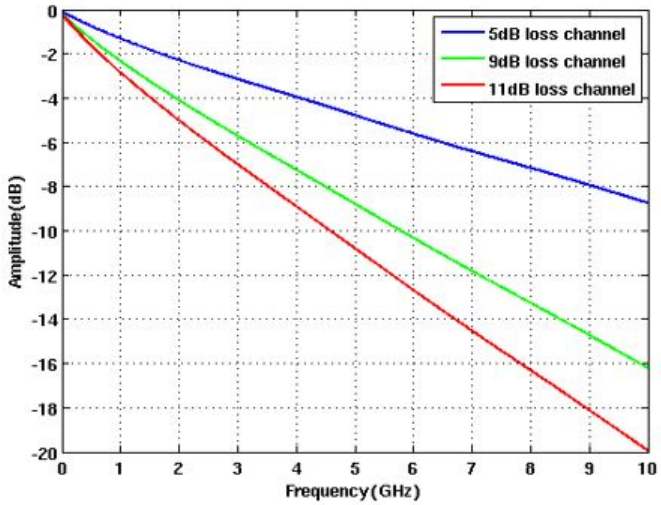

Figure 2.9 Channels under consideration

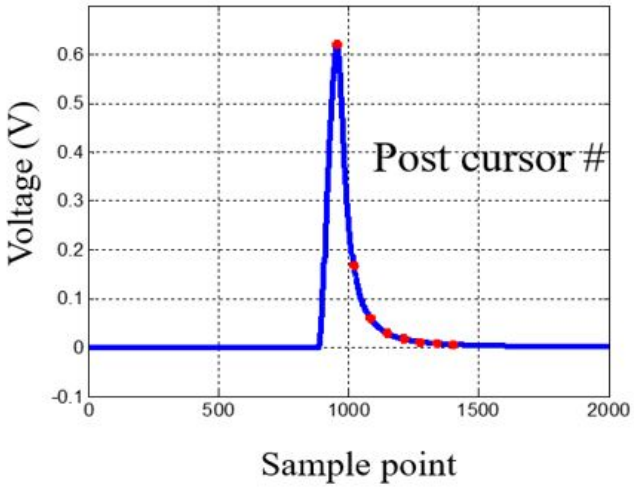

Figure 2.10 Single bit response to extract the worst-case patternn

The single bit responses were first simulated for the worst-case pattern extraction, shown in Figure 2.10. Different data patterns, including PRBS7, PRBS15, worst-case pattern and PRBS15 combined with worst-case pattern were used to simulate DCD amplification. For PRBS7 and PRBS15 patterns, at least two full cycles of PRBS pattern length were simulated. The simulation results are shown in Figure 2.11. These simula-

www.manaraa.com

tions were done with 3%UI DCD injection.

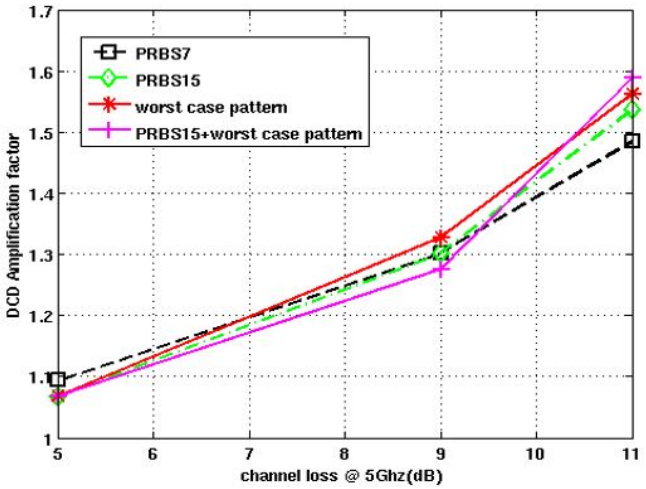

Figure 2.11 DCD amplification for different data patterns

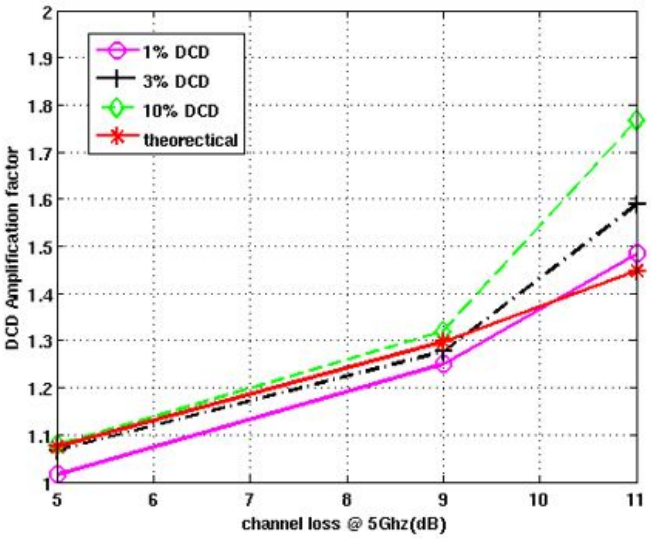

Figure 2.12 Comparison of DCD amplification for different amount of DCD

All data patterns have similar DCD amplification factors. This confirms our intuition that worst-case data pattern can be used to estimate DCD amplification for random data

www.manaraa.com

patterns. Note that the worst-case pattern is much shorter to simulate. As expected, the amplification factor increases as channel loss increases, close to 1.1 for the 5dB low loss channel and up to 1.6 for the 11dB high loss channel. For the 11dB high loss channel, 3% transmitter DCD resulted in 4.8%UI margin loss.

Different amount of DCD signals are injected to show the impact of DCD magnitude on DCD amplification. Figure 2.12 shows DCD amplification factors for different channel losses with 1%, 3%, and 10%UI DCD injection. For the 11dB loss channel, 10%UI DCD results in more than 17%UI margin loss and the largest amplification factor. This again shows the importance of tightening the transmitter DCD budget. The theoretical calculation based on equation 2.8 is also shown in Figure 2.11. It matches well with the time-domain simulation for small DCD as equation 2.8 is based on Taylor series expansion and only holds for small transmitter jitter.

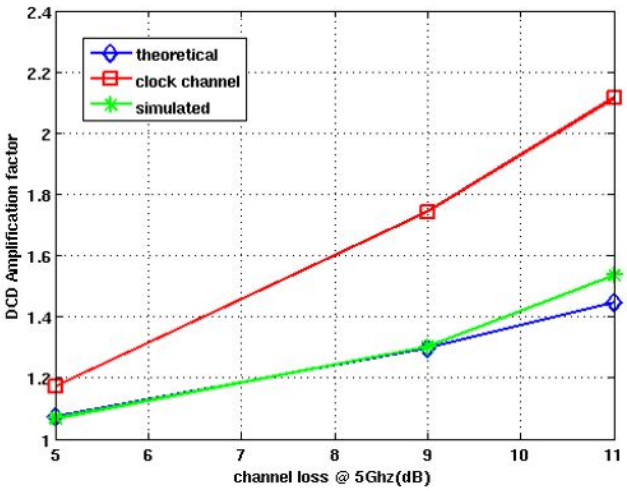

Figure 2.13 Comparison of DCD amplification for different cases

Figure 2.13 overlays the DCD amplification curves for the clock pattern, the worst-case pattern and the theoretical calculation based on equation 2.8. As expected, data channel DCD amplification is smaller than clock channel DCD amplification. Clock channel DCD amplification scales faster with channel loss. For the 5dB low loss channel,

www.manaraa.com

DCD amplification factors for both data and clock channels are close to 1. For the 11dB loss channel, DCD amplification factors for clock and data channels are 2.2 and 1.6, respectively.

2.4 Conclusion

This chapter investigates transmitter DCD amplification in data channels and its impact on link margin. Data channel DCD amplification is defined as the ratio of additional margin loss over DCD. Based on peak distortion analysis and statistical jitter modeling methodology, a general formula is derived to calculate data channel DCD amplification. Simulation results confirm the theory and show that data channel DCD amplification also scales with channel loss but at a lower rate than the clock channel DCD.

www.manaraa.com

REFERENCE

[1] S. Chaudhuri, W. Anderson, J. Bryan, J. McCall, and S. Dabrai,"Jitter amplification characterization of passive clock channels at 6.4 and 9.6 Gb/s," Proc. IEEE 15th Topical Meeting on Electrical Performance of Electronic Packaging, Scottsdale, AZ, Oct. 2006, pp. 21-24.

[2] C. Madden, S. Chang, D. Oh and C. Yuan,"Jitter amplification considerations for PCB clock channel design, " Proc. IEEE 16th Topical Meeting on Electrical. Performance Electronic Packaging, Atlanta, GA, Oct. 2007, pp. 135-138.

[3] S .Chang, D.Oh,and C .Madden,"Jitter modeling in statistical link simulation," IEEE Internatioal Symposium on Electromagnetic Compatibility, Detroit, MI, Aug. 2008, pp. 1-4.

[4] F. Rao and S. Hindi,"Frequency domain analysis of jitter amplification in clock channel," Proc. IEEE 21th Topical Meeting on Electrical Performance of Electronic Packaging, Tempe, AZ, Oct. 2012, pp. 51-54.

[5] D. Oh, F. Lambrecht, S. Chang, Q. Lin, J. Ren, C. Yuan, J. Zerbe, and V. Stojanovic,"Accurate system voltage and timing margin simulation in high-speed I/O system design " IEEE Transaction on Advanced Packaging, vol. 31,no. 4, pp. 722-730, Nov. 2008.

[6] B. K. Casper, M. Haycok, and R. Mooney,"An accurate and efficientanalysis method

www.manaraa.com

for multi-Gb/s chip-to-chip signaling schemes," IEEE Symposium on VLSI Circuits, June 2002, pp. 54-57.

[7] J.Ren, D.Oh,"Multiple edge responses for fast and accurate syestem simulations, " IEEE transactions on Advanced Packageing, vol.31,no.4,pp.741-748. Nov.2008

[8] V.Stojanovic, M.Horowitz, "Modeling and Analysis of High-Speed Links, IEEE Custom Integrated Circuits Conference, September 2003.

[9] J. Caroselli and C. Liu,"An analytic system model for high speed interconnects and its application to the specification of signaling and equalization architectures for 10Gbps backplane communication," DesignCon, February 2006.

www.manaraa.com

CHAPTER 3. ACCURATE JITTER DECOMPOSITION IN HIGH-SPEED LINKS

In a high-speed digital communication system, jitter performance plays a crucial role in Bit-Error Rate (BER). It is important to accurately derive each type of jitter as well as total jitter (TJ) and to identify the root causes of jitter by jitter decomposition. In this chatper, a jitter decomposition algorithm using least squares (LS) is proposed, which simultaneously separates inter-symbol interference (ISI), random jitter (RJ) and periodic jitter (PJ). This algorithm includes a new time domain ISI model, which is more effective than the conventional cursor convolution techniques. The new proposed algorithm only requires the time invariant condition, which is true for almost all systems, while the traditional ISI model is valid only with the linear time invariant assumption. Compared to conventional methods, the proposed jitter decomposition method is able to obtain the estimated individual jitter component values with fine accuracy by using fewer samples of total jitter data. The efficiency and accuracy of the proposed method are demonstrated by simulation and hardware experiments.

3.1 Introduction

With the ever-increasing demand of high speed data rate in serial communication systems, jitter becomes a dominant factor affecting system performance and the bit-error-rate (BER). As it also limits timing margin for the system today, accurate jitter analysis is crucial for next-generation I/O design in order to obtain an acceptable BER.

www.manaraa.com

Jitter is defined as the deviation of transition edges from their ideal location in time and contains multiple components each with different characteristics. The total jitter (TJ) in data signal consists of deterministic jitter (DJ) and random Jitter (RJ). RJ follows unbounded Gaussian distribution due to noise sources (such as thermal noise, flick noise, shot noise etc). DJ obeys bounded distribution and can be decomposed into periodic jitter (PJ) and data dependent jitter (DDJ), bounded uncorrelated Jitter (BUJ). PJ comes from external deterministic noise sources coupling into a system, such as switching power supply noises. DDJ is further divided into duty cycle distortion (DCD) and inter-symbol Interference (ISI). Non-idealities, such as asymmetric rising and falling edges of the clock path generate DCD which the duration of logical 1 is different from the duration of a logical 0. ISI is caused by the bandwidth limitation, loss and reflection of the channel. DDJ is related to the bit sequence.

Understanding the amount of jitter introduced by each jitter source is imperative for predicting overall system performance [1]. Jitter decomposition is a key tool used in such scenarios to identify the root causes of jitter. Jitter can be measured using different methods either using external instrument (oscilloscope, spectrum analyzer and time interval analyzer (TIA)) or on-chip circuit jitter measurement. However, these on-chip jitter measurement circuits [2-6] require a large amount of die area if the jitter histograms have to be collected in real-time. Different off-chip decomposition approaches have been developed in the past. Generally, there are three popular category approaches to decompose jitter: 1) the ones based on histogram or statistical methods; 2) the ones with frequency-domain based analysis; and 3) the ones with time-domain based analysis relying on jitter measurements carried out in real-time.

The histogram methods are based on the popular Gaussian tail model [1] by using the probability distribution of collected jitter values. For example, RJ could be observed at the outer tails of a TJ distribution. Thus, this type of method is also referred to as a tail fitting algorithm. Researchers reported various methods to separate RJ and DJ

www.manaraa.com

components with different tail fitting algorithms [7-9]. However, a large amount of jitter samples is required to correctly identify the fit tail part of distribution. Deconvolution methods [10] rely on the idea that in histogram based analysis, a TJ probability density function (PDF) is given as convolution result of the RJ and DJ components. If one of these two components is approximately estimated, one can use a deconvolution algorithm to determine other components, and thus to retrieve the Gaussian model parameters. However, a major drawback of these methods is that they suffer from lack of accuracy because either the DJ or RJ component has to be estimated prior to the deconvolution.

For the frequency-based method, the time-domain series of jitter can also be represented and analyzed in the frequency domain using the Fourier transform (FT) [11-14]. Then, researchers can use the power spectral density (PSD) to represent the jitter spectrum by applying averaging techniques. Correspondingly, peaks in the spectrum can be interpreted as PJ or DDJ, while the average noise floor denotes the power of RJ. However, they used a clock pattern to estimate the RJ and PJ in the system and the ISI cannot be derived from long run-length patterns. Jitter analysis techniques based on time-domain [15-17] rely on jitter measurements carried out in a real-time mode. This is only possible to those dedicated real-time measurement systems, such as high-speed sampling scopes or TIA. Those instruments also include the histogram or spectral test methods. Unfortunately, histogram, spectral and time-domain methods need sufficient memory depth to acquire enough data so that the accuracy can be assured through these digital signal processing (DSP) techniques [18]. In [19], a low-cost jitter separation method based on ADC testing was developed. However, this method is limited to separate jitter in only clock channel with low speeds.

One important jitter component needed to be addressed is ISI jitter. The ISI jitter plays an important part in the TJ. The ISI modelling methods have been introduced in the past. In [12, 20], they modelled the channel as a first-order and a second-order low-pass filter. However, in presence of discontinuities, such a model is too simple to

www.manaraa.com

represent the real channel and becomes invalid. In [21], the ISI modelling is commonly based on the convolution technique. However, it is very time consuming since the cursor is usually 100-bit long.

In this chapter, we present a jitter decomposition algorithm based on LS which has advantages of 1) accurate estimation for PJ, ISI and RJ; 2) fewer data samples than the instrument test which saves the memory requirement; and 3) the new ISI modelling is accurate and efficient to ISI jitter estimation. In this decomposition algorithm, the ISI jitter model in time domain is simpler than the conventional ISI cursors convolution technique. Another advantage of the proposed ISI modelling is its being more accurate and realistic than the low pass filter model. The PJ and RJ were also modelled by a traditional method [1]. The LS was used to obtain parameter values in the above-mentioned jitter model [22-23]. Compared with the simulation and conventional instrument test methods, the proposed method shows great accuracy in the jitter decomposition.

The rest of the chapter is organized as follows: In Section II, the conventional PJ, RJ models are reviewed and the new ISI jitter model is presented. In Section III, the TJ model is introduced. A LS method is applied to the TJ model and the estimation of jitter component is derived. In Section IV, the validation for the ISI model and jitter decomposition simulation results are presented. In Section V, the hardware experiment is presented. Section VI concludes the chapter.

3.2 Jitter Modeling

This section presents PJ, RJ and ISI jitter models considered in the algorithm. PJ and RJ models are widely used in popular jitter analysis. We developed a new time domain ISI jitter model for a lossy channel which is efficient at ISI estimation.

www.manaraa.com

3.2.1 Periodic Jitter(PJ)

PJ is a repeating jitter signal at certain frequencies. It is typically derived from the noise in a switching power supply or caused by PLL reference clock feedthrough. The mathematical model of PJ [1] is

$$\triangle t_{PJ}[n] = A\sin(2\pi f_0(t-nT)+\emptyset) = a\sin\left(\frac{2\pi f_0 n}{f_s}\right) + b\cos\left(\frac{2\pi f_0 n}{f_s}\right) \tag{3.1}$$

where $\triangle t_{PJ}[n]$ is a jitter amount at sampling time nT; A is the amplitude of PJ; f_0 is the frequency of PJ. In a real system, it can be input reference clock of PLL or power supply noise and extracted from the data through spectral analysis. In this chapter, we consider the PJ from reference clock as an example; f_s is frequency of data stream, and ϕ is the phase.

In equation 3.1, a and b are the estimation parameters for PJ in the algorithm.

3.2.2 Random Jitter (RJ)

RJ is caused by unbounded jitter sources, such as thermal noise, flick noise, and shot noise which can be modeled as Gaussian white noise. Gaussian jitter PDF is defined as

$$f_{RJ}(\Delta t) = \frac{1}{\sqrt{2\pi\sigma}}\exp -\frac{(\Delta t-\mu)^2}{2\sigma^2} \tag{3.2}$$

In equation 3.2, μ is the mean and σ is the standard deviation of Gaussian distribution. In this model, the μ is zero and σ is the estimation parameter for RJ in the algorithm.

3.2.3 Inter-Symbol Interference Jitter (ISIJ)

ISI is caused by reflections and loss in a channel. Figure 3.1 shows a single-bit response after a lossy channel. The pulse becomes widened and attenuated, and it occupies the pre-cursor and post cursor samples. Traditionally, ISI cursors convolution technique is used in ISI modeling. However, this process is time-consuming when the number of

www.manaraa.com

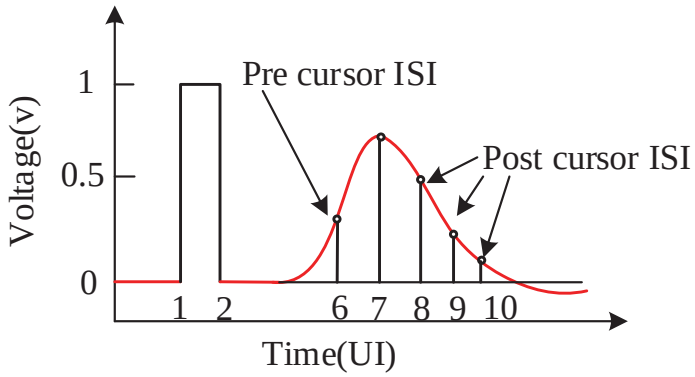

Figure 3.1 The pulse response of the channel

cursor is large. For instance, typical single-bit responses are often more than 100bit long [21]. This convolution approach fails when the non-linearity of the system is severe, or when the data pattern is non-white.

Since sophisticated methods, like transmitter finite impulse response (TX-FIR) equalizer, are needed to properly equalize to the precursor, we consider the k-bit post cursor has a dominant effect on ISI. We use an example to illustrate the time domain ISI model. Considering the data sequence b1-b6 as shown in Figure 3.2, the black curve is the ideal data sequence and blue curve is the actual data sequence due to ISI. b6 is the current bit, b1-b5 are preceding five bits (5 bits post cursor). b1-b5 has 32 binary combinations. If b1-b5 is 01011 as shown in Figure 3.2(a), the time deviation of actual b6 edge and the ideal b6 edge is defined as ISI induced jitter J_{11}. The index 11 is the decimal representation of 01011. If b1-b5 is 01101 shown in Figure 3.2(b), the time deviation of actual b6 edge and the ideal b6 edge is thus noted as jitter J_{13}. The index 13 is the decimal representation of 01101. Different b1-b5 binary combinations generate different ISI jitter amount to current bit b6. Table 3.1 describes the ISI model parameters which include binary combinations, a corresponding jitter J and a sign C. In this example, the

www.manaraa.com

5-bit post cursor has 32 binary combinations and each binary combination generates a corresponding jitter value J. The corresponding sign C is used to represent which kind of binary combination of 5-bit post cursor is selected. In Figure 3.2(a), the 5-bit binary combination is 01011 and the corresponding jitter value is J_{11}, the corresponding sign C_{11} is equal to 1 while other corresponding signs are all zeros.

Table 3.1 The parameters of ISI model

b1-b5 binary combinations	Corresponding Jitter	Corresponding sign
00000	J_0	C_0
...	...	...
01011	J_{11}	C_{11}
...	...	...
11111	J_31	C_{31}

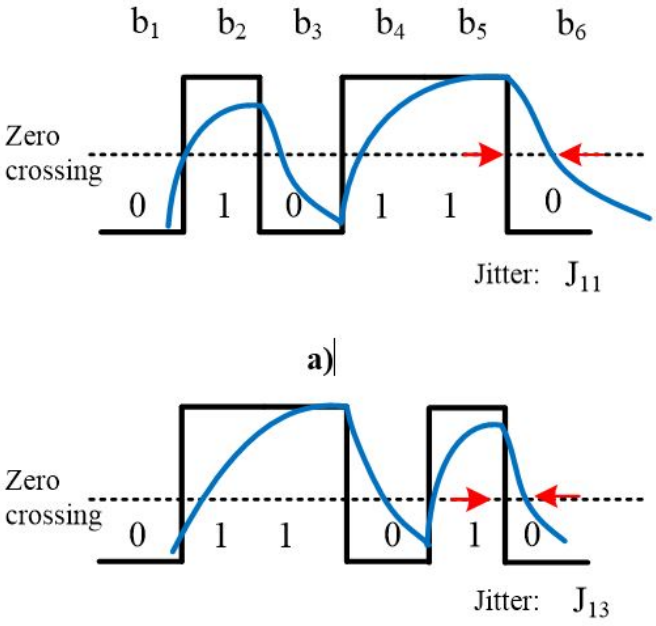

Figure 3.2 An example of ISI modeling

Based on the observation that different post cursor binary combinations generate different ISI jitter amounts on the current bit, we can use a formula to describe the ISI

www.manaraa.com

jitter model. The equation is given by

$$\triangle t_{ISI}[n] = \sum_{0}^{2^K-1} J_l \times C_l[n] \tag{3.3}$$

$$C_l = \begin{cases} 1, \text{if binary to decimal } b_{(n-k)}b_{(n-k+1)}\cdots b_{(n-1)} == l \\ 0, \text{otherwise} \end{cases} \tag{3.4}$$

where $\triangle t_{ISI}[n]$ is ISI jitter of data bit n at sample time nT. The jitter of current bit b_n is determined by binary combinations of previous k bits from b_{n-k} to b_{n-1}. l is the decimal number of binary combination $b_{(n-k)}b_{(n-k+1)}\cdots b_{(n-1)}$. J_l is the jitter value of the lth binary combination to the current bit b_n. C_l is a corresponding sign which represents the binary combination of the previous $k-$bit. The equation describes that ISI jitter of the current b_n is the jitter amount of the previous k-bit. This model does not assume any linearity or superposition requirements on the ISI jitter as a function of the previous k-bit, nor does it rely on linearity in the conversion from data waveform voltage errors to timing errors near zero crossing, thus making the ISI model more robust to channel non-idealities. J_l is the estimation parameter in the algorithm.

The post cursor number k can be obtained from the channel pulse response given the threshold voltage as shown in Figure 3.1

3.3 Jitter Decomposition Algorithm

In this section, we present the TJ modelling used in this study, and then explain the details of the decomposition algorithm.

3.3.1 Total Jitter

The TJ in time domain is considered as the linear sum of DJ components and the square root of the RJ components [21]. With the proposed ISI jitter modelling, the TJ

www.manaraa.com

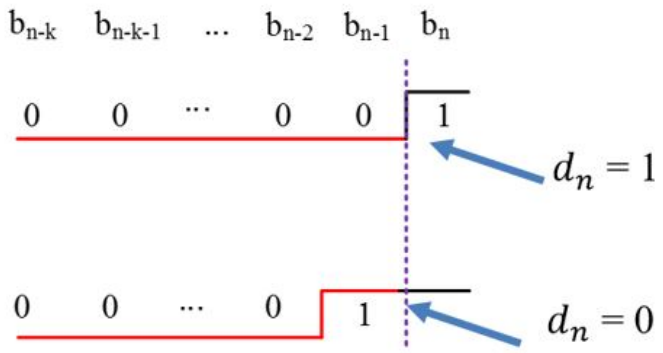

Figure 3.3 Data transition sign d_n

in bit n with PJ, RJ and ISI can be simulated as:

$$x[n] = d_n \times [\triangle t_{PJ}[n] + \triangle t_{ISI}[n] + \triangle t_{RJ}[n]] \tag{3.5}$$

where $x[n]$ is the TIE amount at sampling time nT. d_n is data transition sign used to indicate the existence of a 0 to 1 or 1 to 0 transition from bit $n-1$ to bit n. d_n is 1 only when there is a falling or rising edge from bit $n-1$ to bit n as shown in data (1) of Figure 3.3 d_n is 0 when there is no data transition from bit $n-1$ to bit n as shown in data (2) of Figure 3.3 When d_n is 0, no jitter exists for the current bit n.

3.3.2 Jitter decompostion by Least Squares (LS)

In order to estimate the parameters in the proposed model, a LS based decomposition method is proposed. Equation (4) shows that TJ is a linear equation. For a linear time invariant system, LS estimation overcomes the convergence problem [22-23] and does not require any special distribution properties for the input. Based on this, we applied the LS to estimate the PJ, RJ, and ISI parameters $[a, b, J_0, J_1, \cdots, J_{(2^k-1)}]$.

Define that M bits absolute TIE sequence is shown in Figure 3.4 The absolute TIE in each bit is $x[1], x[2], \cdots, x[M]$ taken at sampling time $1T, 2T, \cdots, MT$, respectively.

www.manaraa.com

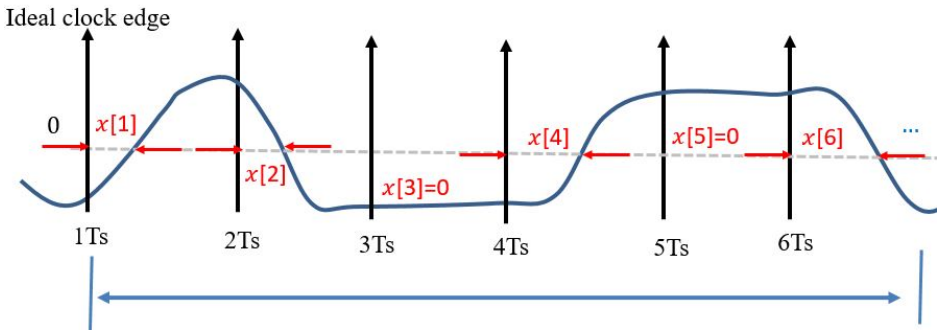

Figure 3.4 M bits total jitter sequence

Then the absolute TIE sequence Z_M can be expressed by the following equation [21]

$$Z_M = \begin{bmatrix} x(1) \\ x(2) \\ \vdots \\ x(M) \end{bmatrix} = H_M\theta + V_M \tag{3.6}$$

where V_M is RJ vector (denoted as $[[1],[2],cdots,[M]]^T$) for the TIE sequence, representing $\left[a, b, J_0, J_1, \cdots, J_{(2^k-1)}\right]^T$ is the estimation parameters for PJ, DCD and ISI. H_M is the coefficient matrix for the whole jitter sequence. H_M is the coefficient matrix for the whole jitter sequence. The submatrix A in H_M is PJ coefficients matrix and submatrix B is ISI coefficients matrix.

$$H_M = \begin{bmatrix} A & B \end{bmatrix} A = \begin{bmatrix} d_1\sin\left(\frac{2\pi f_0}{f_s}\right) & d_1\cos\left(\frac{2\pi f_0}{f_s}\right) \\ \vdots\vdots & \ddots\vdots \\ d_M\sin\left(\frac{2\pi f_0 M}{f_s}\right) & d_M\cos\left(\frac{2\pi f_0 M}{f_s}\right) \end{bmatrix} B = \begin{bmatrix} d_1C_{01} & \cdots & d_1C_{2^k-11} \\ d_2C_{02} & \cdots & d_2C_{2^k-12} \\ \vdots\vdots & \ddots\vdots & \\ d_MC_{0M} & \cdots & d_MC_{2^k-1M} \end{bmatrix} \tag{3.7}$$

In equation 3.7, C_{li} can be extracted from the data stream and store in look up table. For instance, if binary combination of post-cursor of the ith bit data stream is 01101, then C_{12i} is 1 and other C_{xi} is 0. Since the PJ frequency f_0 can be obtained from spectral analysis, we assume the f_0 is a known parameter in this chapter. f_s is the data rate.

www.manaraa.com

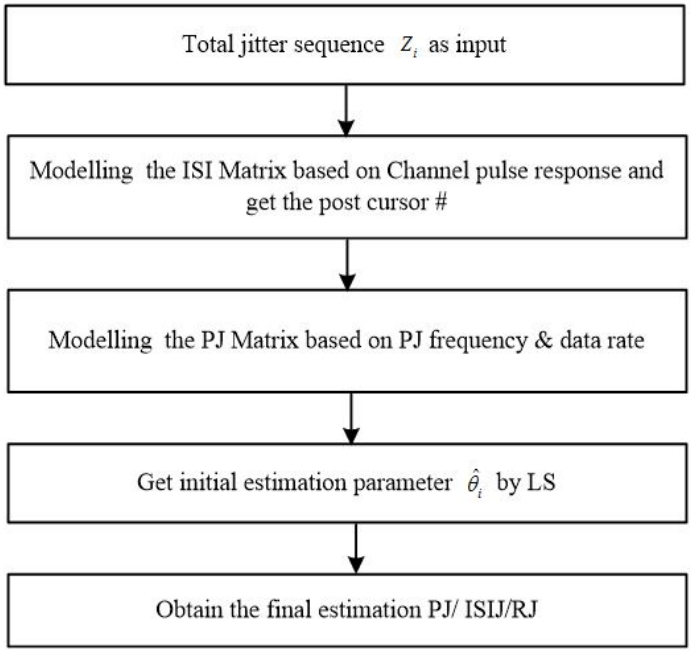

Figure 3.5 The flow chart of proposed algorithm

The solution $\widehat{\theta}_M$ for length of absolute TIE M bits is

$$\widehat{\theta} = \left[H_M^T H_M\right]^{-1} H_M^T Z_M \tag{3.8}$$

The estimation parameter $\widehat{\theta}_M$ is

$$\widehat{\theta} = \left[\widehat{a}, \widehat{b}, \widehat{J}_0, \widehat{J}_1, \cdots, \widehat{J}_{2^K-1}\right] \tag{3.9}$$

The amplitude of estimation PJ is

$$\widehat{A} = \sqrt{\widehat{a}^2 + \widehat{b}^2} \tag{3.10}$$

The ISI is calculated by

$$\widehat{ISI}_{pk-pk} = max(\widehat{J}_0, \widehat{J}_1, \cdots, \widehat{J}_{2^K-1}) - min(\widehat{J}_0, \widehat{J}_1, \cdots, \widehat{J}_{2^K-1}) \tag{3.11}$$

The variance of RJ is

$$\sigma^2 = var(Z_M - H_M\widehat{\theta}) \tag{3.12}$$

www.manaraa.com

The flow chart of the proposed algorithm is given in Figure 3.5. The jitter sequence with PJ, ISI, and RJ is the input of the algorithm. The ISI matrix is formed based on data stream information and the PJ matrix is formed based on the data rate and PJ frequency (from the spectral analysis). According to equation 3.8, the initial estimation jitter component value $\widehat{\theta}$ could be obtained by LS. The final estimation of PJ, ISI, and RJ values is based on equation 3.10 3.11 and 3.12 respectively.

3.4 Simulation Results

In this section, the proposed decomposition methods are validated by Matlab simulation. PRBS-7 data length is 1.27k bits and the data rate are 10Gb/s and 25Gb/s in the simulation respectively. The PJ was a sine wave with 100MHz frequency at 10Gb/s and 250MHz frequency at 25 Gb/s.

3.4.1 Validation of the ISI Jitter Estimation

In order to verify that the ISI jitter estimation is previous k-bit dependent, we used ISI TIE jitter sequence as reference criteria. The extraction of S-parameter of a PCB transmission line (channel A) with insertion loss 3.5dB at 5GHz and 7.5dB at 12.5GHz was used to generate the ISI TIE sequence. We classified the ISI jitter sequence to 2^k binary combinations. The post cursor number k of the transmission line is 5, which was obtained from the channel pulse response. These 5 bits post cursors have 32 binary combinations from 00000 to 11111.The corresponding jitter amount are from J_0 to J_{31} . The red dots in Figure 3.6 and Figure 3.7 represents the TIE ISI jitter in each ISI binary combination which shows that different binary combinations correspond to different ISI jitter at 10Gb/s and 25Gb/s respectively.The eye diagram of PRBS 7 only with ISI in Figure 3.8 shows ISI jitter (pk-pk) in this transmission line is about 7.2 ps at 25Gb/s.

www.manaraa.com

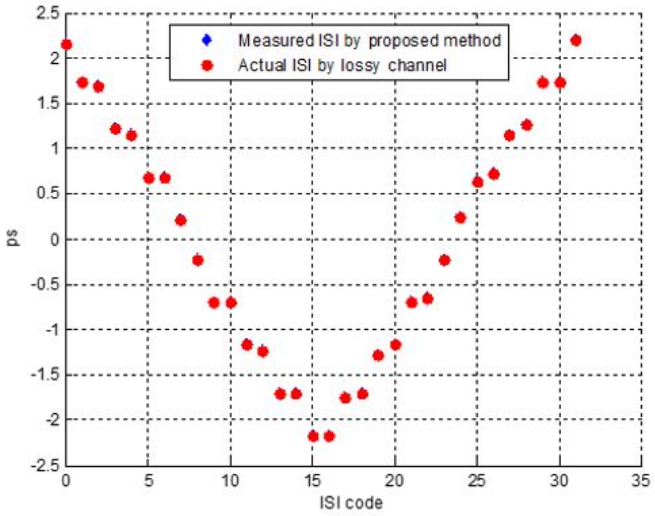

Figure 3.6 Comparison of ISI TIE binary combination and estimation result for 10Gb/s

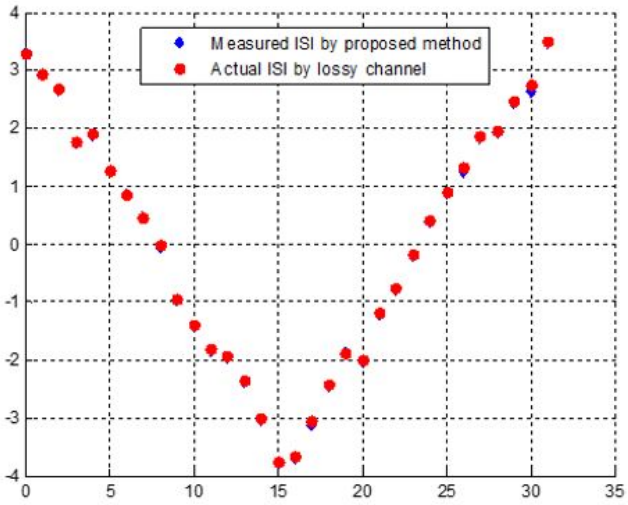

Figure 3.7 Comparison of true and estimation ISI for 25Gb/s

In order to verify the decomposition algorithm, the whole PRBS7 data stream with only ISI jitter was sent to the proposed method.

www.manaraa.com

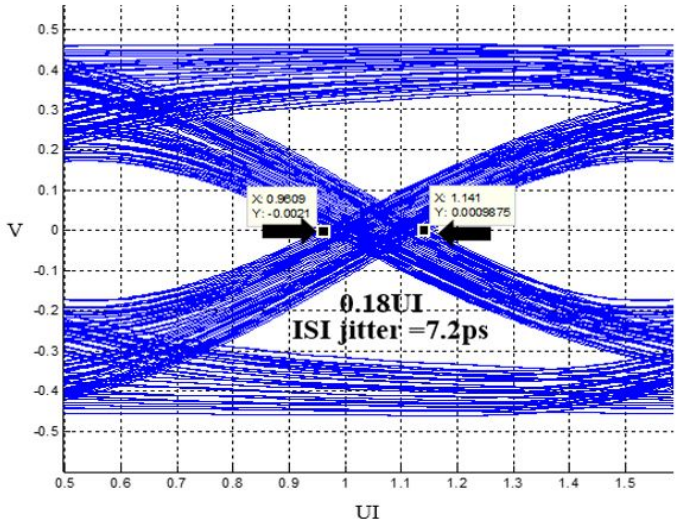

Figure 3.8 Eye diagram of PRBS-7 only with ISI at 25Gb/s

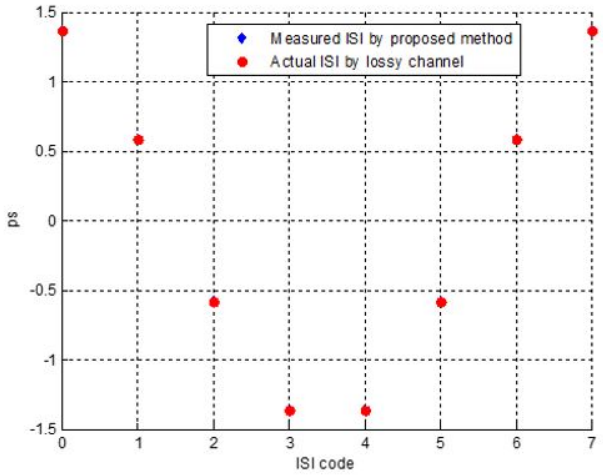

Figure 3.9 Comparison of true and estimation ISI for Channel B at 10Gb/s

The estimated ISI jitter for each binary combination is represented by blue dots in Figure 3.6 for 10Gb/s and in Figure 3.7(a) for 25Gb/s. They show that the estimated ISI and actual ISI are very close both at 10Gb/s and 25Gb/s. The estimated ISI (pk-pk) value is about 4.6 ps at 10Gb/s and 7.2 ps at 25Gb/s which is very close to the pk-pk

www.manaraa.com

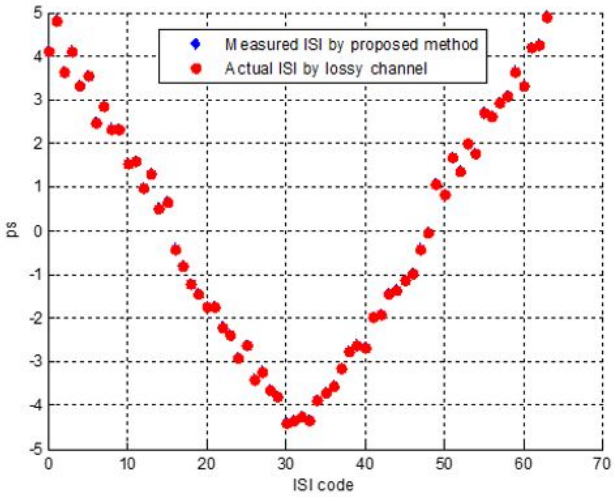

Figure 3.10 Comparison of true and estimation ISI for Channel C at 10Gb/s

jitter obtained from the eye diagram in Figure 3.8.

We also modeled channel B with 3dB loss and channel C with 5dB loss at 5GHz to verify the proposed ISI model. The simulation results are shown in Figure 3.9 and 3.10, and mathes the true ISI well.

3.4.2 Validation of PJ, DCD and ISI Jitter

In the 10Gb/s simulation, PRBS-7 data stream were generated by Matlab Simulink toolbox with different PJ (0ps, 20ps pk-pk), RJ (0ps, 2.13ps, rms value) and ISI jitter caused by channel A (4.6 ps, pk-pk) was sent to the algorithm. The simulation results are summarized in the third column group of Table 3.2. It shows the estimated jitter is very close to the added jitter.

In the 25Gb/s simulation, PRBS-7 data stream with different jitter component were generated. The data stream with different PJ (0,1.5ps, 4ps, 8ps peak-peak value), RJ (0,2.13 ps) and ISI jitter caused by channel A (7.2ps, pk-pk value) was sent to the algorithm. The simulation results are summarized in Table 3.3. The estimation error of ISI is less than 0.5 ps. The estimation error of PJ is close to 0 ps. DCD estimation were

www.manaraa.com

larger than the added caused by the jitter amplification due to channel loss.

3.5 Measurement Results

To verify the ISI (pk-pk) estimation and the accuracy and efficiency of the algorithm, a hardware test bench was used (shown in Figure 3.11) to measure the jitter components for 10Gb/s. A Tektronix BSA286C BERTscope was used to generate data stream with PJ. An Agilent Infiniium Wide-Bandwidth Oscilloscope was used to measure the jitter with internal software. The same PCB transmission line in simulation part A was used to generate the ISI jitter. All experiments were done at a data rate of 10Gb/s.

We also used the result of JNEye with conventional decomposition algorithms as reference to compare the proposed method. JNEye is Intels state-of-the art jitter and noise link analysis tool for evaluate high-speed serial link performance. The jitter decomposition of this platform is based on hybrid algorithms with statistical, frequency-domain, and time-domain analysis with dedicated jitter components modeling and extraction. The accuracy of JNEye has been validated with both simulation and measurement correlations [22]. All tests for comparison were done at 10Gb/s and 25Gb/s.

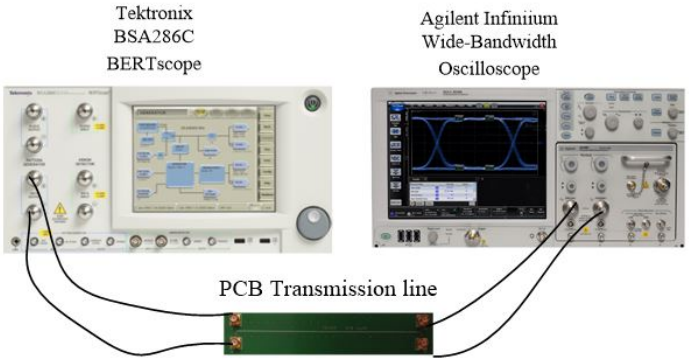

Figure 3.11 The 10Gb/s experimental test bench

www.manaraa.com

3.5.1 Validation of the ISI Jitter pk-pk Estimation

Since the ISI jitter (pk-pk) generated by the PCB transmission line (channel A) is unknown, the ISI measurement result of the oscilloscope is as a reference. In the experiment, jitter free PRBS7 data generated by BERTscope was sent to the PCB transmission line and the output of transmission line was connected to oscilloscope to obtain the ISI jitter value.

In the proposed method, the PRBS-7 data stream with ISI jitter was sent to the algorithm and the estimation ISI value was obtained. The channel A was used in JNEye to generate PRBS-7 with ISI and then decompose the ISI jitter. The comparison is listed in the Table 3.2. In the added jitter term, the unknown for ISI means that the ISI jitter amount is unknown when the transmission line is added. The ISI value in the measurement is 5.1 ps and the estimation in JNEye is 4.4ps. The one in the proposed method is 4.6 ps. All values are very close. Therefore, the ISI modeling is validated. PJ in the proposed method is 0ps (pk-pk) while the oscilloscope result is about 1.4ps. RJ (rms value) in the simulation is 0ps while the one in the measurement is 1.4ps due to the instrument noise.

3.5.2 Comparison of the Accuracy and the Sample Data

In the hardware experiment, PRBS-7 data stream at 10Gb/s with different PJ (0ps and 20ps pk-pk) generated by BERTscope was sent to channel A and then sent to oscilloscope. PRBS7 data stream with different amounts of PJ and ISI jitter were sent to the proposed method and JNEye.

Table 3.2 shows the comparison results among oscilloscope, JNEye and the proposed method. When the measurement result is stable, the experiment should run several minutes according to test experience. When the added ISI is 0, the oscilloscope result of ISI jitter is about 3.2 ps due to the cable while in the proposed algorithm the estimation is about 0 ps. The DCD measurement result is 0.8ps while in the proposed method is 0.

www.manaraa.com

The data sample in each simulation is 8k, but the oscilloscope requires at least 200k data in the experiment. Therefore, the proposed algorithm has comparable accuracy using fewer data samples than instrument and have the same accuracy compare to JNEye.

Table 3.2 Comparasion the proposed method, instrument and JNeye at 10Gb/s unit (ps)

Added jitter			Instrument			JNeye			The propose methed		
RJ	PJ	ISI	RJ	PJ	ISI	RJ	PJ	ISI	RJ	PJ	ISI
0	0	unknown	1.5	1.4	5.1	0	0	4.4	0	0.07	4.6
0	20	0	1.4	20.5	3.2	0	20.1	4.4	0	20.2	4.7
2.13	20	unknown	2.49	20.3	6.5	2.2	20	4.4	2.14	20.1	4.6

Table 3.3 Comparasion the proposed method and JNeye at 25Gb/s unit (ps)

Added jitter			JNeye					The propose methed		
RJ	PJ	ISI	sample data	RJ	PJ	ISI	sample data	RJ	PJ	ISI
0	0	unknown	8k	0	0	4.4	1.27k	0	0.07	4.6
0	20	unknown	8k	0	20.1	4.4	1.27k	0	20.2	4.7
2.13	20	unknown	8k	2.1	20	4.4	1.27k	2.12	20.1	5.2
2.13	20	0	8k	2.2	19.8	0	1.27k	2.1	20.4	0

It shows that both of two methods have the same estimation accuracy with the same sample data at 10Gb/s. For the 25Gb/s comparison, PRBS7 data stream with different PJ (1.5ps,4ps,8ps pk-pk), RJ (0ps,2.13ps) and ISI jitter generated from channel A were sent to JNEye and the proposed method.

Table 3.3 shows the estimation results in the proposed method and the JNEye estimation result. Both of two methods have the same estimation accuracy with the same sample data. However, the proposed method has two obvious advantages. First, the proposed method was used less sample data than instrument test or commercial simulation platform. Second, the proposed method can provide ISI analysis in detail for each ISI binary combination.

www.manaraa.com

3.6 Conclusion

An efficient and accurate algorithm that simultaneously extracts the periodic jitter, random jitter and ISI jitter is presented. This method is based on time-domain ISI modelling which is simpler than the conventional cursor convolution technique. It utilizes fewer sample data while maintaining great estimation accuracy in both clock pattern and data pattern. The comparison between simulation results and the hardware test ones demonstrates the accuracy of the proposed jitter measurement method. However, there are still limitations in the proposed method. It is only applied in low lossy channels, which means the eye diagram should be open before the receiver of system and the data logic value can be correctly determined. The algorithm needs the TJ sequences as an input which requires extra instruments such as TIA to store the jitter sequence.

3.7 Acknowledge

The authors would like to thank Huawei Technologies Co. Ltd for providing the PCB board and instruments for hardware validation and Intel for providing JNeye validation in this chapter.

www.manaraa.com

Reference

[1] M. P. Li, "Jitter, noise and signal integrity at high-speed," Prentice Hall Pearson Education, 2007, Chapter 6, pp.163-185.

[2] K.-H. Cheng, J.-C. Liu, C.-Y. Chang, S.-Y. Jiang, and K.-W. Hong, "Built-in jitter measurement circuit with calibration techniques for a 3-GHz Clock Generator,"IEEE Trans. VLSI Systems, vol. PP, no. 99, pp. 111, Jun. 2010.

[3] J.-C. Hsu and C. Su, "BIST for measuring clock jitter of charge-pump phase-locked loops," IEEE Trans. Instrumentation and Measurement, vol. 57, no. 2, pp. 276285, Feb. 2008.

[4] K. Ichiyama, M. Ishida, T. J. Yamaguchi, and M. Soma, "Novel CMOS circuits to measure data-dependent jitter, random jitter, and sinusoidal jitter in real time," IEEE Trans. Microw. Theory Tech., vol. 56, no. 5, pp. 12781285, May 2008.

[5] S.-Y. Jiang, K.-H. Cheng, and P.-Y. Jian, "A 2.5-GHz built-in jitter measurement system in a serial-link transceiver," IEEE Trans. VLSI Systems, vol. 17, no. 12, pp. 1698 1708, Dec. 2009.

[6] K. Nose, M. Kajita, and M. Mizuno, "A 1-ps resolution jitter-measurement macro using interpolated jitter oversampling," IEEE J. Solid-State Circuits, vol. 41, no. 12, pp. 29112920, Dec. 2006.

[7] M. Li, J. Wilstrup, R. Jessen, and D. Petrich, "A new method for jitter decompo-

www.manaraa.com

sition through its distribution tail fitting," in Proc. International Test Conference. IEEE, Sept. 1999, pp. 788794.

[8] G. Hnsel, K. Stieglbauer, G. Schulze, and J. Moreira, "Implementation of an economic jitter compliance test for a multi-gigabit device on ATE," IEEE Int. Test Conf. (ITC04), pp. 13031312, Oct. 2004.

[9] D. Hong and K.-T. Cheng, "An accurate jitter estimation technique for efficient high speed I/O Testing," IEEE Asian Test Symp. (ATS07), pp. 224229, Oct. 2007.

[10] R. Stephens, "Separation of random and deterministic components of jitter," U.S. Patent 7 149 638, Dec. 12, 2006.

[11] C.-K. Ong, D. Hong, K.-T. Cheng, and L.-C. Wang, "Jitter spectral extraction for multi-gigahertz signal," in Proc. ASP-DAC, Asia and South Pacific. IEEE, Jan. 2004, pp. 298 303.

[12] V.K Sharma,J.N.Tripath, R.Nagpal, S.Deb and R.Malik,"A Comparative analyis of jitter estimation techniques," in proc.IEEE International Conference on Electronics,communication and computational engineering(ICECCE), 2014

[13] H. Pang, J. Zhu, and W. Huang, "Jitter decomposition by fast Fourier transform and time lag correlation," in IEEE Int. Conf. Communications, Circuits and Systems ICCCAS, Jul. 2009, pp. 365368.

[14] T. Yamaguchi, H. Hou, K. Takayama, D. Armstrong, M. Ishida et al., "An FFT-based jitter separation method for high-frequency jitter testing with a 10x reduction in test time," IEEE Int. Test Conf. (ITC07), pp. 18, Oct. 2007.

[15] Q. Dou and J. Abraham, "Jitter decomposition by time lag correlation,aration method for high-frequency jitter testing with a 10x reduction in test time," IEEE Int. Symp. Quality Electronic Design (ISQED06), Mar. 2006.

www.manaraa.com

[16] J. Zhu and W. Huang, "Jitter analysis and decomposition based on EMD/HHT in high-speed serial communications," in Proc. of IEEE Int. Conf. Testing and Diagnosis (ICTD09), Apr. 2009, pp. 14.

[17] K. Bidaj, J.Baptiste and J.Deroo, "Time-domain PLL modeling and RJ/DJ jitter decomposition," in Proc. of IEEE, 22-25 May 2016

[18] Agilent Technologies Application Note, "Analyzing jitter using Agilent EZJIT plus software," Literature Number AN-5989-3776EN, 2005

[19] L.Xu, Y. Duan, D. Chen, "A low cost jitter separation and characterization method," in proc. IEEE International Symposium on VLSI Test Symposium (VTS), May,2015

[20] J. Buckwalter, B. Analui and A. Hajimiri, "Predicting data dependent jitter," IEEE Trans. Circuits and Sys. II, vol. 51, pp. 453-457 Sep. 2004.

[21] D.Oh and X.Yuan ,"High speed signaling: jitter modeling analysis, and budgeting," Prentice Hall Pearson Education, 2011, Chapter 3, pp.44-64

[22] Gustafsson F. "Adaptive filtering and change detection," West Sussex: Wiley; 2000.

[23] Liu H, He Z. "A sliding-exponential window RLS adaptive algorithm," properties and applications. Signal Process 1995;45:35768.

www.manaraa.com

CHAPTER 4. A LOW-COST COMPARATOR-BASED METHOD FOR ACCURATE DECOMPOSITION OF DETERMINISTIC JITTER IN HIGH-SPEED LINKS

Jitter decomposition is a key tool to identify root causes of jitters in a high-speed digital communication system. It is such a huge challenge in balancing the test cost and precision for conventional decomposition methods implemented in instruments where the time interval error (TIE) data is necessary. In this chapter, we propose a deterministic jitter decomposition method using Boolean output from a network of simple low-cost comparators to identify the deviation of current sampling position from the ideal sampling position instead of TIE data. The new method simultaneously separates inter-symbol interference (ISI), periodic jitter (PJ) and duty cycle distortion (DCD). Simulation and measurement results demonstrate that the proposed method can estimate the ISI, PJ and DCD with sufficient accuracy using significantly fewer data samples than the state-of-the-art instrument test, and thus reduce test cost greatly. Furthermore, the comparators have extremely relaxed design requirements, offering potential for possible on-chip implementation for built-in self-test (BIST) or background test.

4.1 Introduction

To satisfy the aggressive demand for higher data rate of communication system, the input/output (IO) speed double every two to three years on average [1]. Using the Optical Internetworking Forum (OIF) Common Electrical I/O (CEI) implementation

www.manaraa.com

agreement as an example, its speed has increased from 6.5G (Gen1), to 11.3G (Gen2), to 28G (Gen3) .As the data rate increases, the unit interval (UI) shrinks. The UI reduction implies that the total timing budget for the I/O link decreases. Meanwhile, as the data rate increases, the inter symbol interference (ISI) becomes severe due to the bandwidth limitation. Jitter and noise, generated inevitably in the transmitter, channel and receiver, impact the performance of the system. It is important to understand the amount of jitter introduced by each jitter source to predict the overall system performance [2]. Jitter decomposition is a key tool to identify the root causes of jitter at the chip design, simulation, and characterization stages. However, the test of high-speed interfaces has posed significant challenges in terms of test cost and quality. Currently available jitter measurement techniques require expensive measurement instruments but they do not guarantee sufficient test quality.

Jitter is defined as the variation of transition edges from their ideal locations in time [2]. It becomes a dominant factor affecting the bit error rate (BER) with increasing data rate in high-speed serial communication systems. Total jitter (TJ) in a data signal often consists of deterministic jitter (DJ) and random jitter (RJ) [2]. RJ follows unbounded Gaussian distribution due to thermal noise and shot noise, etc. DJ obeys bounded distributions and can be decomposed into periodic jitter (PJ) and data dependent jitter (DDJ), bounded uncorrelated jitter (BUJ). PJ is caused by power supply switching frequency or phase locked loop (PLL) clock feed through. BUJ is caused by the channel crosstalk. DDJ is further divided into ISI and duty cycle distortion (DCD). ISI is caused by the lossy characteristics of the channel. Non-idealities including asymmetric rising and falling edges of the clock path generate DCD.

Many researchers have proposed various algorithms to decompose jitter components. These algorithms fall into three main categories. The first one is frequency-domain based analysis. The time domain series of jitter can be analyzed in frequency domain through the Fourier transform [3-4]. The power spectral density (PSD) represent the

www.manaraa.com

jitter spectrum by applying averaging techniques. However, the clock pattern is used to estimate the RJ and PJ in the system and the ISI cannot be derived from clock pattern.

The second one is based on histogram or statistical method using probability distributions of collected jitter values. A TJ distribution can be decomposed into two Gaussian tails and is also referred to as tail fitting algorithms. Various methods were developed to separate the RJ and DJ components with tail fitting algorithms [5-8]. Some other jitter decomposition methods are based on deconvolution [9-10] of the wavelet transforms [11] and Gaussian mixture models [12]. Deconvolution methods rely on the jitter distribution rule that a total jitter probability distribution function (PDF) is given as the convolution result of the DJ components and RJ (having a Gaussian distribution) in histogram based analysis. However, a large amount of jitter samples is required for fitting algorithms.

The third one is time-domain based analysis [13-14] relying on jitter measurements carried out in a real-time mode. This is only feasible for these dedicated real-time measurement systems, such as high-speed sampling scopes or time interval analyzers (TIAs).

In industry, dedicated instruments are widely used to measure and decompose jitter. The jitter analysis algorithms in these instruments are usually implemented using the histogram method or spectral test. All these algorithms require large samples of TIE jitter data. TIE jitter is the actual deviation from the ideal clock period over all clock periods. TIE data must be measured by an instrument with: 1) sufficient bandwidth (three times the data rate is usually adequate) to represent the signal; 2) sufficient memory depth to acquire enough data so that the digital signal processing (DSP) techniques are accurate [15]; 3) low noise. These requirements can be satisfied with high precision circuits such as extremely fast ADC and ideal PLL. Unfortunately, the manufacturing cost for such instrument is very high and the instrument design also remains a big challenge when the data rate is extremely high. For example, testing the 25Gb/s high speed I/O requires a sampling rate of more than 50Gb/s by the instrument.

www.manaraa.com

In this chapter, instead of using TIE data, we present a jitter decomposition algorithm through a series of digitized 0s and 1s from comparator network which is directly related to TIE (called comparator network based method). The comparator network based method offers several advantages. First, no TIE data is needed, meaning that no extremely high precision circuit is necessary which greatly reduces the complexity of circuit design and test cost. Also, this method provides accurate estimation for PJ, ISI and DCD. In addition, it requires much fewer data samples than instrument testing. Moreover, the ISI modeling proposed by our previous research [16] is accurate and more efficient for jitter estimation than conventional ISI convolution technique [17] because the traditional ISI modeling is time-consuming. In addition, it is more realistic than the first order or second order low pass filter model [18-19].

In our proposed algorithm, a comparator network was used to sample the edge of integer periods of pseudo-random binary sequence (PRBS) data and obtain the 0s and 1s. Given an initial sampling position which is different from the slicer of receiver, the outputs (0s and 1s) of comparators were sent to the decomposition algorithm based on Least squares (LS) [20-21]. Then, it obtained the new comparator network sampling position to acquire a new set of output 0s and 1s. This iteration process would continue till zero crossing points were found, where the difference in the number of 0s and 1s is the smallest. Meanwhile, the new sampling position includes PJ, ISI and DCD information. This proposed algorithm shows great accuracy for jitter decomposition and requires much fewer data samples compared to the conventional instrument test method.

The rest of the chapter is organized as follows: In section II, the comparator based jitter decomposition method is described in detail. In section III, the simulation results are presented. In section IV, the measurements are provided. Section V concludes the chapter.

www.manaraa.com

4.2 Comparator Based Decompostion Method

In this section, we proposed the comparator based method to decompose the PJ, DCD and ISI in detail. First, individual jitter component model for PJ, DCD and ISI are introduced. Second, the process of using comparator output to replace the TIE is developed. Third, using comparator output to estimate the jitter component by Least squares (LS) in one iteration is described in detail. Fourth, the block iteration is applied to data groups to obtain the final jitter component estimation. Fifth, some parameters in this method and comparator design considerations are addressed.

4.2.1 Deterministic Jitter Modeling

PJ and DCD are modeled by the traditional method [2, 17]. ISI jitter is modeled in time domain. PJ is a repeating jitter whose frequency typically known. It is caused by noise in a switching power supply or PLL reference clock feed through. PJ is modeled mathematically as a sum of sinusoids (here a single sinusoid is shown as example):

$$\triangle t_{PJ}[n] = A\sin(2\pi f_0(t - nT) + \emptyset) = a\sin\left(\frac{2\pi f_0 n}{f_s}\right) + b\cos\left(\frac{2\pi f_0 n}{f_s}\right) \tag{4.1}$$

where $\triangle t_{PJ}[n]$ represents a PJ amount at sampling time nT; f_0 is the fundamental frequency of PJ; A is the amplitude of PJ; In a real system, it can be power supply noise or input reference clock of PLL which can be extracted from the data through spectral analysis. In this chapter, the PJ from reference clock is considered as an example in simulation; f_s is the frequency of data stream, and is the phase of PJ. In equation 4.1, a and b are the estimation parameters in this algorithm.

DCD creates the widest set of frequency components in the clock pattern and can be viewed as a series of adjacent positive and negative impulses at the input. The frequency is half of the data rate, which can be modeled as [17]

$$\triangle t_{DCD}[n] = J_{DCD} \times \cos(n\pi) = [-J_{DCD}, J_{DCD}, -J_{DCD}, J_{DCD}, \cdots] \tag{4.2}$$

www.manaraa.com

where $\triangle t_{DCD}[n]$ is a DCD amount at sampling time nT; J_{DCD} is the DCD amplitude which is to be estimated in the algorithm.

ISI is caused by reflections and loss in a channel. The pulse of a single bit becomes widened and attenuated after a lossy channel, and it occupies the pre-cursor and post cursor samples. The ISI model used here follows that used in [16] and is more accurate and efficient than conventional modeling based on convolution and first or second order low pass filtering.

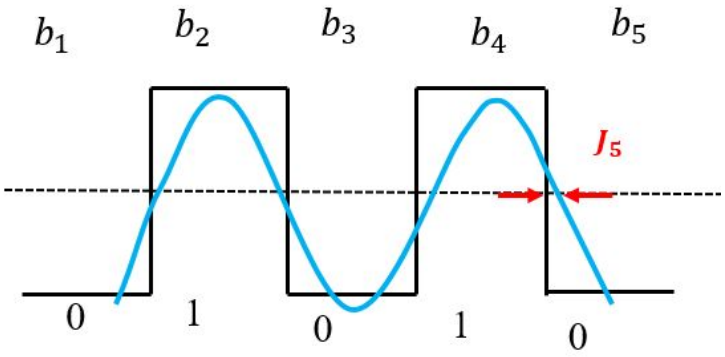

Figure 4.1 An Example of ISI modeling

In a high speed link, since some kind of equalizer is typically used to properly equalize to the precursor, the post cursors from the previous k bits is considered in modeling the ISI effect. The selection of the number of post cursor k depends on the amount of loss of the channel and the data rate, and can be determined during channel characterization. The previous k bits has 2^k binary combinations. In the works case, each binary combination generates a different ISI jitter amount on the current bit (main cursor). For example, as shown in Figure 4.1 the blue curve is the actual data sequence due to ISI and the black curve is the ideal data sequence for data sequence $b1-b5$. b5 is the current bit, $b1-b4$ are preceding four bits. $b1-b4$ has 16 binary combinations. If $b1-b4$ is 0101 as shown in Figure 4.1, the time deviation of actual b5 edge and the ideal b5 edge is defined as ISI induced jitter J_5. The index 5 is the decimal representation of 0101. ISI

www.manaraa.com

jitter model is described in the following equation

$$\triangle t_{ISI}[n] = \sum_{0}^{2^k-1} J_l \times C_l[n] \tag{4.3}$$

$$C_l = \begin{cases} 1, \text{if binary to decimal } b_{(n-k)}b_{(n-k+1)}\cdots b_{(n-1)} == l \\ 0, \text{otherwise} \end{cases} \tag{4.4}$$

where $\triangle t_{ISI}[n]$ is ISI jitter of data bit n at sample time nT. The jitter of current bit b_n is determined by binary combinations of previous k bits from b_{n-k} to b_{n-1}. l is the decimal number of binary combination $b_{(n-k)}b_{(n-k+1)}\cdots b_{(n-1)}$. J_l is the jitter value of the lth binary combination to the current bit b_n. C_l is a corresponding sign which represents the binary combination of the previous $k-$bit. The equation describes that ISI jitter of the current b_n is the jitter amount of the previous k-bit. This model does not assume any linearity or superposition requirements on the ISI jitter as a function of the previous k-bit, nor does it rely on linearity in the conversion from data waveform voltage errors to timing errors near zero crossing, thus making the ISI model more robust to channel non-idealities. J_l is the estimation parameter in the algorithm.

All DJ components, as well as RJ, contribute to total jitter [2], which results in timing interval errors. The deterministic part of TIE at data bit n with PJ, DCD and ISI can be modeled as:

$$x[n] = d_n \times [\triangle t_{PJ}[n] + \triangle t_{DCD}[n] + \triangle t_{ISI}[n]] \tag{4.5}$$

where $x[n]$ is the TIE amount at sampling time nT. d_n is data transition sign used to indicate the existence of a 0 to 1 or 1 to 0 transition from bit $n-1$ to bit n. d_n is 1 only when there is a falling or rising edge from bit $n-1$ to bit n as shown in data (1) of Figure 4.2 d_n is 0 when there is no data transition from bit $n-1$ to bit n as shown in data (2) of Figure 4.2. When d_n is 0, no jitter exists for the current bit n.

www.manaraa.com

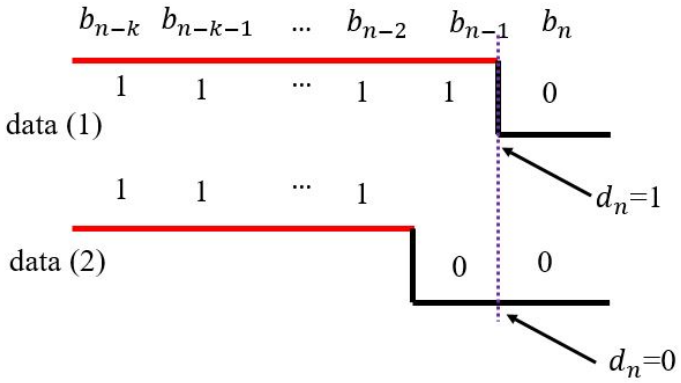

Figure 4.2 The data transition sign d_n

4.2.2 Replacing TIE Sequece using Comparator Network

Existing methods for jitter decomposition take TIE data as input. TIE is defined as the timing difference between the zero-crossing time of the actual data and that of the ideal data, as indicated by $x[n]$ for bit b_n in Figure 4.3. In this sense, $x[n]$ is called the absolute TIE since it is relative to the ideal zero-crossing. However, obtaining TIE is a very challenging task, as explained in the introduction. Therefore we would like to replace TIE by something that is much easier to obtain.

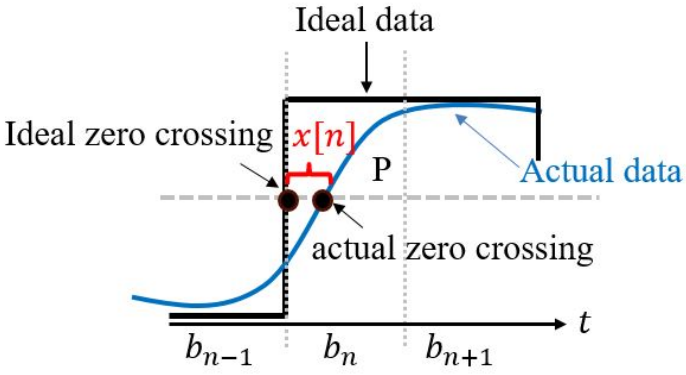

Figure 4.3 The definition of absolute TIE

For simplicity, let us assume there is no overshooting, undershooting, ringing and

www.manaraa.com

other signal integrity problems in the data edges, i.e., the data edges are monotonic both at rising and falling edge, as shown in Figure 4.3.

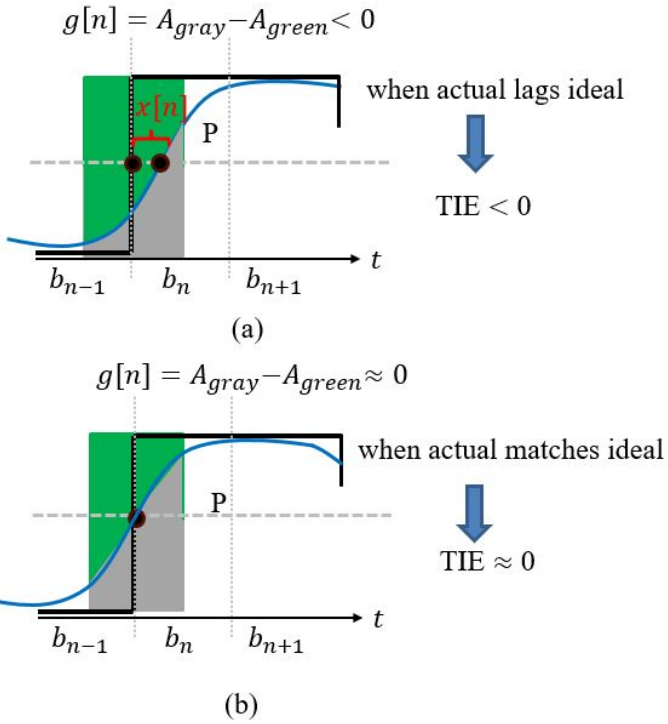

Figure 4.4 Converting finding zero crossing point to area difference function (a)

This is not a serious limitation since the proposed method focus near zero crossing and monotonic near zero crossing is usually satisfied. Let us place a rectangular box centered at the zero crossing of the ideal edge. The actual data waveform divides the box into two parts, the gray and the green. It is fairly simple to show that when $x[n]$ is small, the area difference between the gray and green areas is proportional to $x[n]$. Let us define the area difference between the gray and green area as function $g(n)$. As seen in Figure 4.4(a), when $x[n] < 0$, the actual data curve makes zero-crossings behind the ideal data, and the area difference function $g(n)$ is negative. When the actual and ideal zero-crossings match, i.e., $x[n]0$, the gray and green areas are approximately the same and $g[n] \approx 0$, as seen in Figure 4.4(b). When the actual data leads the ideal data, i.e.,

www.manaraa.com

$x[n] > 0$, the gray area will be more than the green area and we have $g[n] > 0$, as seen in Figure 4.5(a). Therefore, $g[n]$ is locally proportional to $x[n]$ and could be used as a candidate for replacing the TIE $x[n]$.

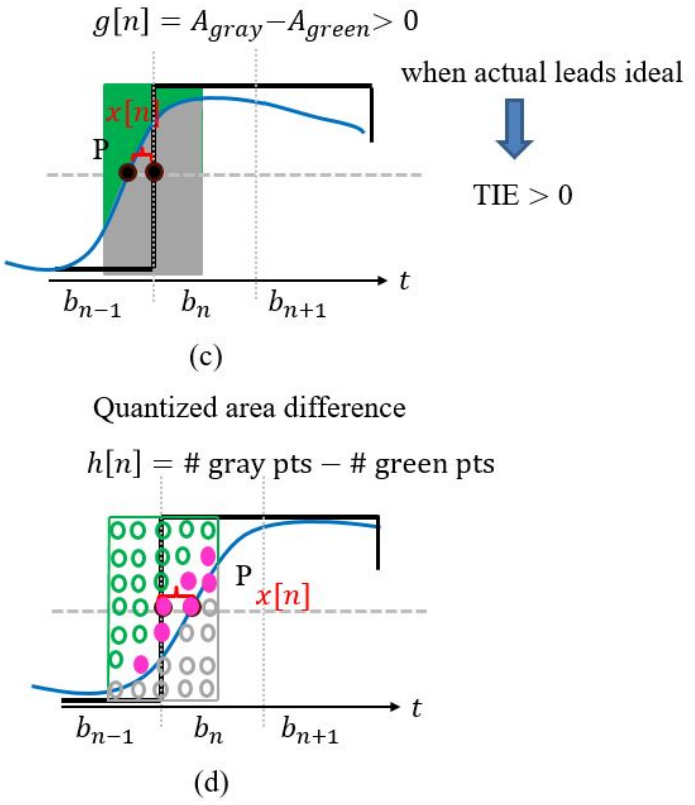

Figure 4.5 Converting finding zero crossing point to area difference function (b)

The area difference function $g(n)$, however, is an analog quantity and is not directly available. To solve this problem, we can use a quantized representation to approximate the area function, by placing a set of grid points in the box. The gray area will be approximated by the number of gray dots and the green area is approximated by the number of green dots. With this, the quantized area difference function $h(n)$ is given by

$$h[n] = (\sharp\, green\ dots - \sharp\, gray\ dots)_n = quantized(A_{gray} - A_{green})_n \qquad (4.6)$$

This is illustrated in Figure 4.5.

www.manaraa.com

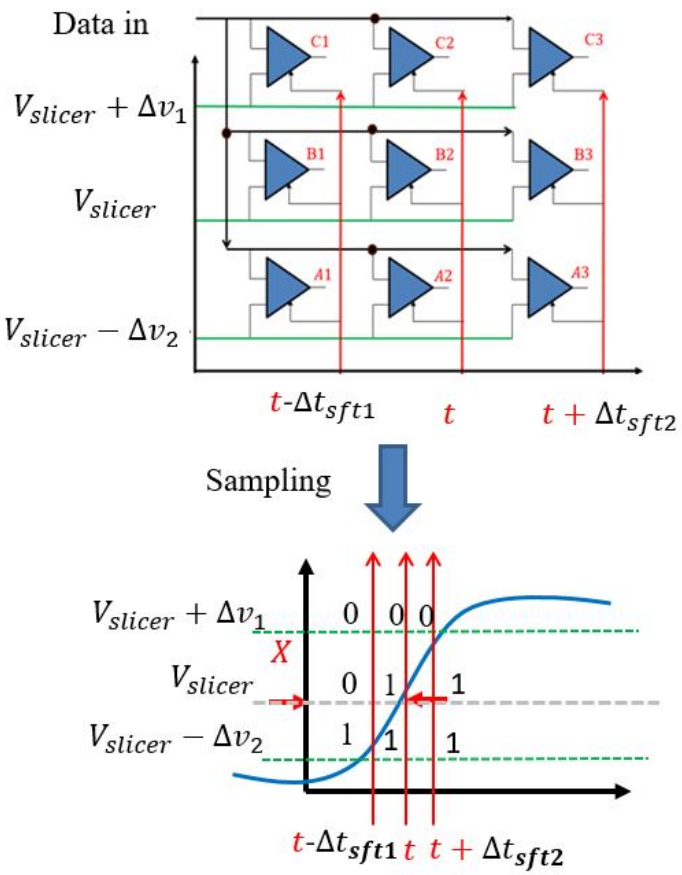

Figure 4.6 An example of comparator network

A simple way to count dots is to place a comparator at each grid point, with both varying threshold voltages and triggering times, as shown in Figure 4.6(a) ($3 * 3$ $A1, \cdots, C3$ as an example). The number of comparators with output equal to 1 represents the quantized gray area, and the number of comparators with output equal to 0 represents the green area. In the voltage domain, comparators in each row have the same nominal threshold voltage, and between adjacent rows the threshold voltages differ by $\triangle v$. In the time domain, comparators in each column are triggered together, and comparators in adjacent columns have trigger times differ by a small time-delay $\triangle t$. In the Figure 4.6(a), there are two voltage intervals ($\triangle v1$ and $\triangle v2$) and two time-delays $\triangle t_{sft1}$ and $\triangle t_{sft2}$). The $\triangle v$ and $\triangle t_{sft}$ do not have to have the same values. The voltage intervals

www.manaraa.com

can be different for different rows, and within one row each comparator can have its own threshold variations. Similarly, the time-delays do not need accurate controls and within each column each comparator can have its own aperture uncertainties. The reason for this relaxed requirement is that in the presence of variation, we can still use the difference in the number of output 1 and the number of output 0 as a quantized representation of the area difference. More details are discussed in design consideration section. For simplicity, we assume the voltage intervals $\triangle v$ between each row are the same and so are the time delay $\triangle t_s ft$ between each column. The 9 comparators in Figure 4.6(b) produced 9 Boolean outputs. The number of 0 represents the green area and the number of 1 represents the gray area. The difference of 0s and 1s represents the area difference function h(n) shown in Figure 4.5(d) and is given by equation

$$h[n] = (\sharp\, 1 - \sharp\, 0)_n \; = quantized(A_{gray} - A_{green})_n \tag{4.7}$$

Now that we have a quantized area function that is easily obtained by a comparator network, we would like to use it to replace TIE in a jitter decomposition algorithm. Before we can do that, we need to figure out the approximate proportionality coefficient between $h(n)$ and TIE, that is, we need an approximate value of α such that $\alpha \triangle h[n] \approx x[n]$.

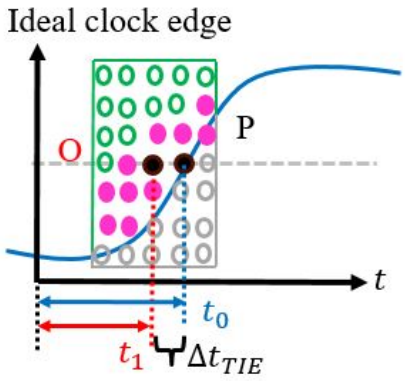

Figure 4.7 The relative TIE definition

To approximate α, we move the center of the grid point box by a time interval $\triangle t_{TIE}$

www.manaraa.com

to O from P as shown in Figure 4.7. Some of the comparators output will change from 1 to 0. This will cause a change in the value of the quantized area function $\triangle h(n)$. To a first order approximation, we will have:

$$\triangle t_{TIE} \approx \alpha \triangle h[n] = \alpha(\triangle\sharp\, 0 - \triangle\sharp\, 1)_n \tag{4.8}$$

This process can be done either in simulation with a good model of the channel, or in characterization if the channel and comparator network hardware is available. In this chapter, we measure the channel characteristics and use simulation to obtain since the comparator network are not in hardware. α can also be roughly obtained from the average slope of the actual data edge waveform. The relative TIE $\triangle t_{TIE}$ represents the deviation of a given reference point and the zero crossing point. For example, every edge in a data stream shown in Figure 4.8 was sampled by the comparator network (one green arrow refer to a comparator network) bit by bit and the position of comparator network is the reference point of the relative TIE. The gray dashed arrows represent the ideal clock edges used in definition of absolute TIE. Therefore, the relative TIE sequence $(\triangle t_{TIE}[1], \triangle t_{TIE}[2], \cdots, \triangle t_{TIE}[M])$ of M bit data stream shown in Figure 4.8 can be expressed by a series difference of 1 and 0 in each sampling time 1T, 2T,$\cdots$,MT respectively which is given in the following

$$\triangle TIE[M] = \begin{bmatrix} \alpha_1(\triangle\sharp\, 0 - \triangle\sharp\, 1)[1] \\ \alpha_2(\triangle\sharp\, 0 - \triangle\sharp\, 1)[2] \\ \vdots \\ \alpha_M(\triangle\sharp\, 0 - \triangle\sharp\, 1)[M] \end{bmatrix} \approx \begin{bmatrix} \alpha(\triangle\sharp\, 0 - \triangle\sharp\, 1)[1] \\ \alpha(\triangle\sharp\, 0 - \triangle\sharp\, 1)[2] \\ \vdots \\ \alpha(\triangle\sharp\, 0 - \triangle\sharp\, 1)[M] \end{bmatrix} \tag{4.9}$$

Although different data edges have different slopes, we can use a roughly estimated value for all the rising edges and $-\alpha$ for all the falling edges.

www.manaraa.com

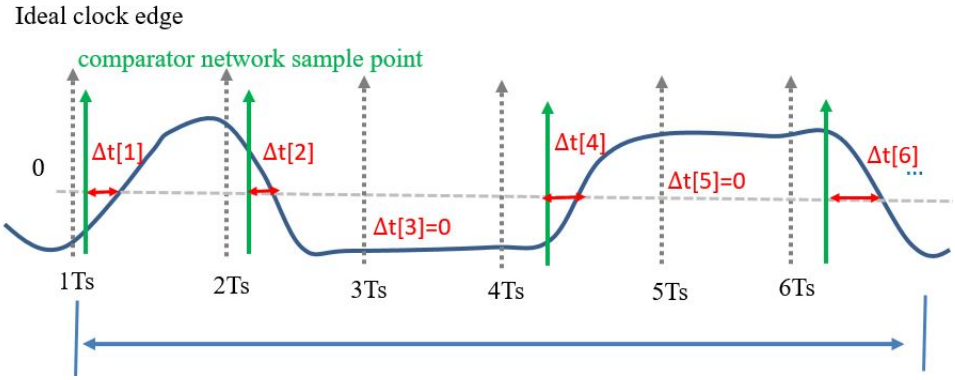

Figure 4.8 M-bit relative TIE

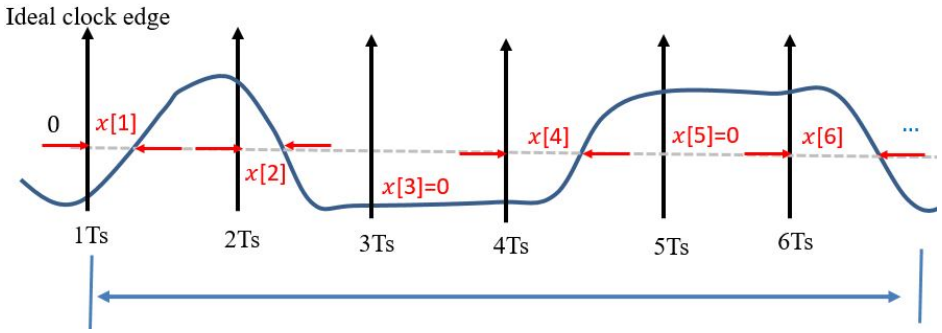

Figure 4.9 M-bit absolute TIE

4.2.3 Jitter Decompostion by Least Squares (LS)

Using LS can estimate the PJ and ISI in the proposed model if absolute TIE is known which was proved by our previous work [16]. Equation 4.5 shows that TIE is a linear equation. For a linear time invariant system, LS estimation overcomes the convergence problem [20-21] and does not require any special distribution properties for the input. Based on this, we applied the LS to estimate the PJ, DCD and ISI parameters $[a, b, J_{DCD}, J_0, J_1, \cdots, J_{(2^k-1)}]$.

Define that M bits absolute TIE sequence is shown in Figure 4.9 The absolute TIE in each bit is $x[1], x[2], \cdots, x[M]$ taken at sampling time $1T, 2T, \cdots, MT$, respectively.

www.manaraa.com

Then the absolute TIE sequence Z_M can be expressed by the following equation

$$Z_M = \begin{bmatrix} x(1) \\ x(2) \\ \vdots \\ x(M) \end{bmatrix} = H_M\theta + V_M \tag{4.10}$$

where V_M is RJ vector (denoted as $[[1],[2],\cdots,[M]]^T$) for the TIE sequence, θ representing $\left[a, b, J_{DCD}, J_0, J_1, \cdots, J_{(2^k-1)}\right]^T$ is the estimation parameters for PJ, DCD and ISI. H_M is the coefficient matrix for the whole jitter sequence. The submatrix A in H_M is PJ coefficients matrix. The submatrix B is DCD coefficients and submatrix C is ISI coefficients matrix.

$$H_M = \begin{bmatrix} A\ B\ C \end{bmatrix} A = \begin{bmatrix} d_1\sin\left(\frac{2\pi f_0}{f_s}\right) & d_1\cos\left(\frac{2\pi f_0}{f_s}\right) \\ \vdots\vdots & \ddots\vdots \\ d_M\sin\left(\frac{2\pi f_0 M}{f_s}\right) & d_M\cos\left(\frac{2\pi f_0 M}{f_s}\right) \end{bmatrix} B = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} C = \begin{bmatrix} d_1C_{01} & \cdots & d_1C_{2^k-11} \\ d_2C_{02} & \cdots & d_1C_{2^k-12} \\ \vdots\vdots & \ddots & \vdots \\ d_MC_{0M} & \cdots & d_1C_{2^k-1M} \end{bmatrix} \tag{4.11}$$

In equation 4.11, C_{li} can be extracted from the data stream and store in look up table. For instance, if binary combination of post-cursor of the ith bit data stream is 01001, then C_{9i} is 1 and other C_{xi} is 0. Since the PJ frequency f_0 can be obtained from spectral analysis, we assume the f_0 is a known parameter in this chapter. f_s is the data rate. The solution $\widehat{\theta}$ for length of absolute TIE M bits is

$$\widehat{\theta} = \left[H_M^T H_M\right]^{-1} H_M^T Z_M \tag{4.12}$$

The estimation parameter $\widehat{\theta}$ is

$$\widehat{\theta} = \left[\widehat{a}, \widehat{b}, \widehat{J}_{DCD}, \widehat{J}_0, \widehat{J}_1, \cdots, \widehat{J}_{2^K-1}\right] \tag{4.13}$$

www.manaraa.com

The amplitude of estimation PJ is

$$\widehat{A} = \sqrt{\widehat{a}^2 + \widehat{b}^2} \tag{4.14}$$

The ISI is calculated by

$$\widehat{ISI}_{pk-pk} = max(\widehat{J}_0, \widehat{J}_1, \cdots, \widehat{J}_{2^K-1}) - min(\widehat{J}_0, \widehat{J}_1, \cdots, \widehat{J}_{2^K-1}) \tag{4.15}$$

Since we are unable to know the ideal clock edge as a reference point and absolute TIE data is difficult to obtain, we set an initial guess reference point S_{old} (the middle column of the comparator network was post in S_{old}) and the initial guess of the jitter component in the point S_{old} is θ_{old} which is $\left[a_{old}, b_{old}, J_{DCDold}, J_{0old}, J_{1old}, \cdots, J_{(2^k-1)old}\right]$. As mentioned in part B of this section, the difference of 0 and 1 of comparator output is approximately proportional to relative TIE $\triangle t_{TIE}$ of position O and zero crossing point P shown in equation 4.9. Therefore, the M bits relative TIE sequence can be expressed the difference of 1 and 0 by the following equation

$$\triangle \widehat{\theta} = \left[H_M^T H_M\right]^{-1} \alpha[\sharp 0 - \sharp 1]_M \tag{4.16}$$

$\triangle\theta$ represents estimate parameters $[\triangle a, \triangle b, \triangle J_{DCD}, \triangle J_0, \triangle J_1, \cdots, \triangle J_{2^k-1}]$ and we combine the equation 4.12 and 4.13, the estimated $\triangle\theta$ is obtained by

$$\widehat{\theta}_{new} = \theta_{old} + \triangle\widehat{\theta} \tag{4.17}$$

4.2.4 Block Iteration

In reality, performed estimation one time is not suffice due to the nonlinear edges and inaccurate value. Therefore, recursive iteration is needed to approach the expected solution. We herein use an example to explain the whole process to obtain the zero-crossing point P and PJ, DCD and ISI estimation (as shown in Figure 4.10). The incoming PRBS data stream with PJ, DCD ISI jitter was divided into several blocks. Each block had the same integer periods of PRBS data (M bits) and should cover at

www.manaraa.com

least 1 period PJ. First, in block 1, given an initial PJ, DCD ISI jitter θ_0 and calculated the initial sample position S_0 based on equation 4.13 4.14(shown in Figure 4.10), the comparator network (the red arrow shows) samples the edge of each data at S_0 position, the outputs (0 and 1) of comparator network were counted. According to LS calculation and equation 4.16 4.17, the estimation distance of the initial sample position S_0 to zero-crossing point $\triangle\theta_0$ can be obtained the following equation. Then, the new jitter estimated $\widehat{\theta_1}$ can be obtained based on the previous block θ_0 and $\triangle\widehat{\theta_0}$ in the following equation

$$\widehat{\theta_1} = \theta_0 + \beta\triangle\widehat{\theta_0} \tag{4.18}$$

In equation 4.18, β is gain factor and $\beta < 1$ which guarantees iteration converge to zero crossing point.

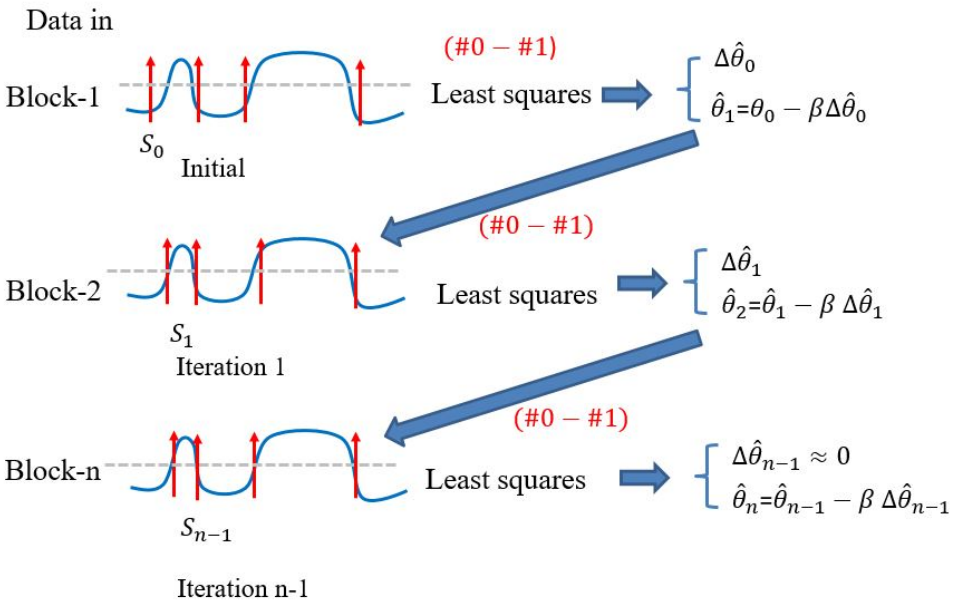

Figure 4.10 Whole iteration process

Second, in block 2, the new sampling positon S_1 was determined according to θ_1 .The comparator network samples the edge of each data at S_1 position, the outputs (0 and 1) of comparator network were sent to LS estimat θ_1 and $\triangle\theta_1$. This iteration continues

www.manaraa.com

to different blocks until $\sharp 1 - \sharp 0$ approaches zero where is the zero-crossing point. The final estimation θ_n contains the PJ, DCD and ISI information. The PJ is obtained by equation 4.14 and ISI peak to peak (pk-pk) is calculated by equation 4.15.

The flow chart of the proposed method is given in Figure 4.11. First, given initial jitter θ_{old} for the data block 1 and calculated the comparator network sampling instance based on equation 4.6, every data block length is M bits.

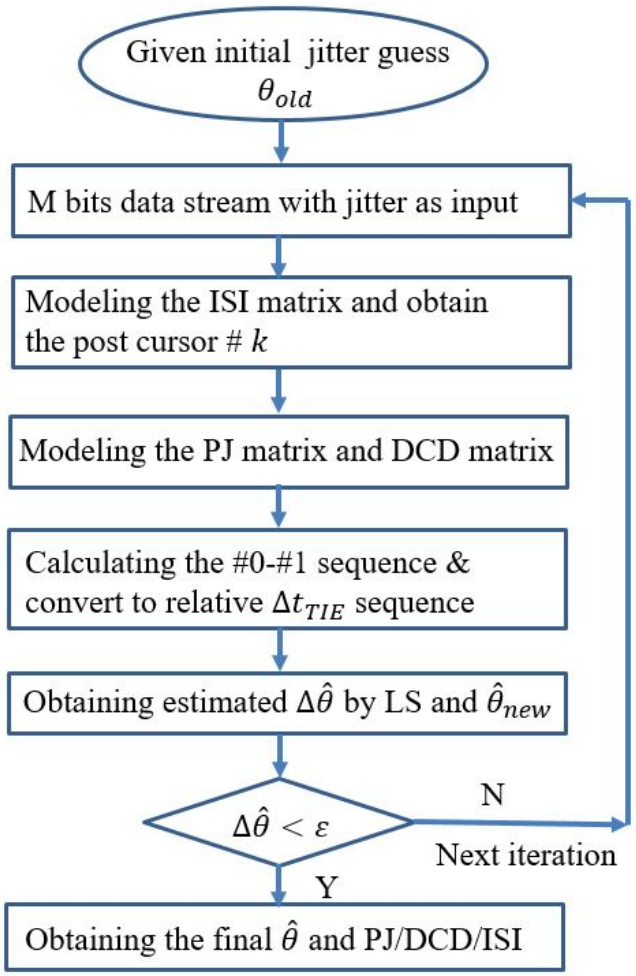

Figure 4.11 The flow chart of the proposed method

Second, modeled the ISI matrix based on data pattern and obtained the post cursor number k. Modeled the PJ matrix and DCD matrix based on data rate and PJ frequency. Third, calculated the difference sequence of 0 and 1 based on comparator network output

www.manaraa.com

for each edge and converted relative $\triangle t_{TIE}$ sequence. Fourth, used the LS to the relative $\triangle t_{TIE}$ sequence and obtained the estimated $\triangle\theta$ and $\triangle\theta_{new}$ based on equation 4.16 4.17.If $\triangle\theta$ less than a threshold value ξ, calculated the final PJ,DCD and ISI estimation based on equation 4.14 4.15. Otherwise, the θ_{new} is considered as the θ_{old} of the next block iteration until the $\triangle\widehat{\theta}$ less than a threshold value ξ.

4.2.5 Alogorithem Realization Analysis

The requirements of comparator network in this algorithm are not stringent. First, the speed is the data rate of I/O rather than the requirement that 3 times higher than data rate in a real-time testing instrument. Second, the sampling clocks $t - \triangle t_{sft1}$, t and $t + \triangle t_{sft1}$ shown in Figure 4.6 are no need to be ideal or jitter-free. The jitter existing in sampling clocks helps to generate the different time intervals thus no extra precision circuit are needed to generate the exact time interval. For example, the $\triangle t_{sft}$ can be 2.5ps in 10Gb/s and 1.2 ps in 25Gb/s. It is difficult to design precision circuit to generate such a small delay $\triangle t_{sft}$. The PVT and comparator mismatch can easily generate the different time intervals $\triangle t_{sft}$ and different voltage intervals $\triangle v$ followed by the Gaussian distribution. Third, the time interval $\triangle t_{sft1}$ and $\triangle t_{sft2}$ are no need to be equal. The voltage intervals $\triangle v_1$ and $\triangle v_2$ between rows are not required to be equal. All these relaxed requirements reduce the comparator design complexity greatly and benefit the algorithm.

4.3 Simulation Results

In this section, the proposed decomposition methods are validated by Matlab simulation. PRBS-7 data length is 8k bits and each block is 1.27k bits according the previous section analysis and the data rate are 10Gb/s and 25Gb/s in the simulation respectively. The PJ was a sine wave with 100MHz frequency at 10Gb/s and 250MHz frequency at

www.manaraa.com

25 Gb/s. The 5*5 comparator network were modeled by Matlab. Each horizontal comparator time interval is random with $\mu = 2.5$ps, $\sigma = 0.5$ps in 10Gb/s and $\mu = 1$ps, $\sigma = 0.5$ps in 25Gb/s, respectively. The vertical voltage interval μ is 10% supply voltage, $\sigma = 20mV$ for 10Gb/s and 25Gb/s if supply voltage is 1v , α was roughly guessed as 0.3 in 10Gb/s and 0.5 in 25Gb/s based on the observation of data edge waveform and $\beta = 1$.

4.3.1 Validation of the ISI Jitter Estimation

In order to verify that the ISI jitter estimation is previous k-bit dependent, we used ISI TIE jitter sequence as reference criteria. The extraction of S-parameter of a PCB transmission line (channel A) with insertion loss 3.5dB at 5GHz and 7.5dB at 12.5GHz was used to generate the ISI TIE sequence. We classified the ISI jitter sequence to 2^k binary combinations. The post cursor number k of the transmission line is 6, which was obtained from the channel pulse response. These 6 bits post cursors have 64 binary combinations from 000000 to 111111. The corresponding jitter amount are from J_0 to J_{63} . The red dots in Figure 4.12 and Figure 4.13(a) represents the TIE ISI jitter in each ISI binary combination which shows that different binary combinations correspond to different ISI jitter at 10Gb/s and 25Gb/s respectively. The eye diagram of PRBS 7 only with ISI in Figure 4.14 shows ISI jitter (pk-pk) in this transmission line is about 7.2 ps at 25Gb/s. In order to verify the decomposition algorithm, the whole PRBS7 data stream with only ISI jitter was sent to the proposed method. The estimated ISI jitter for each binary combination is represented by blue dots in Figure 4.12 for 10Gb/s and in Figure 4.13(a) for 25Gb/s. Figure 4.13(b) shows the error between the actual ISI and estimated ISI. They show that the estimated ISI and actual ISI are very close both at 10Gb/s and 25Gb/s. The estimated ISI (pk-pk) value is about 4.6 ps at 10Gb/s and 7.2 ps at 25Gb/s which is very close to the pk-pk jitter obtained from the eye diagram in Figure 4.14.

www.manaraa.com

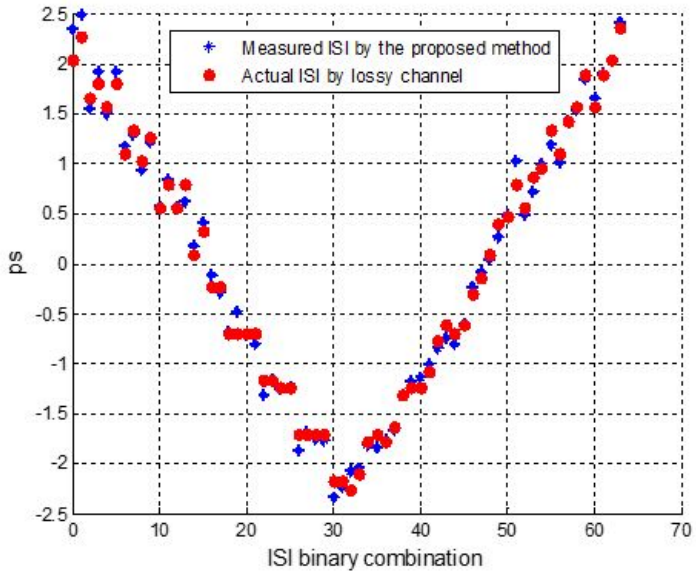

Figure 4.12 Comparison of ISI TIE binary combination and estimation result for 10Gb/s

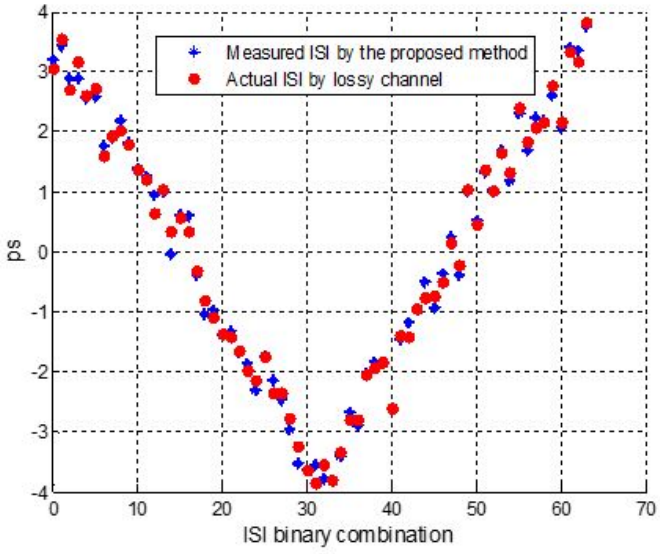

Figure 4.13 Comparison of ISI TIE binary combination and estimation result for 25Gb/s

www.manaraa.com

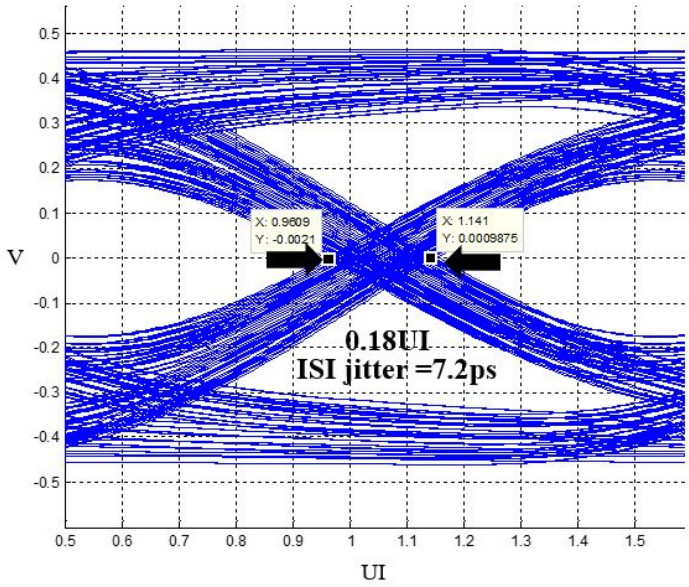

Figure 4.14 Eye diagram of PRBS-7 only with ISI at 25Gb/s

4.3.2 Validation of Convergence of the Algorithm

Given an initial PJ (10ps), ISI (0ps for each binary combination) and DCD (0ps) at 25Gb/s simulation, the comparator network shift to initial sample position. The outputs of comparator network were sent to decomposition algorithm for DJ parameter estimation. The estimation is done in block recursive least squares. In each recursive iteration, the least square fitting error can be computed. Figure 4.15 shows the sequence fitting errors after each iteration. It can be seen that the errors were reduced as more iterations were conducted. The fitting error sequence after iteration 5 is shown in Figure 4.16. It is clear that the errors have been reduced to within 1, which is the quantization error of the quantized area function, thus demonstrating convergence of the iteration process. The RMS values of fitting error sequences (excluding locations where no transitions occur) are: 14.11, 6.1, 1.83, 1, 1 and 1, respectively from the initial iteration to the 5th iteration.

www.manaraa.com

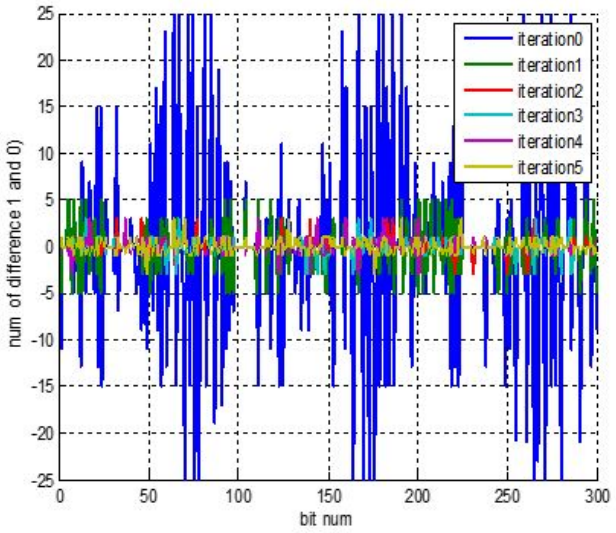

Figure 4.15 The iteration process at 25Gb/s

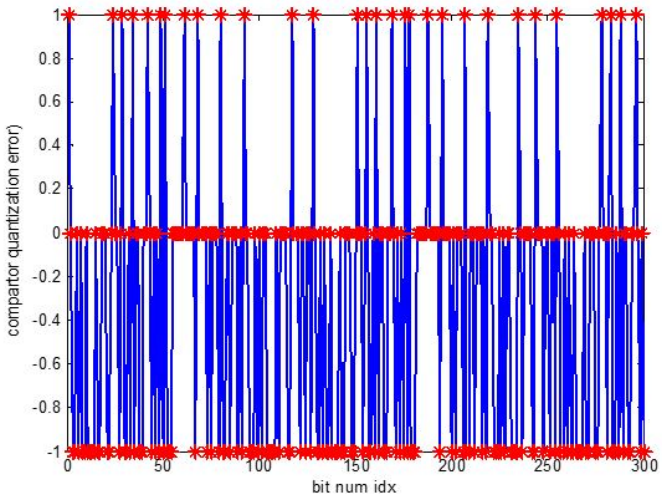

Figure 4.16 The last iteration result

4.3.3 Validation of PJ, DCD and ISI Jitter

In the 10Gb/s simulation, PRBS-7 data stream with different PJ (10ps, 15ps, 25ps pk-pk), DCD (4ps,8ps,10ps) and ISI jitter caused by channel A (4.6 ps, pk-pk) was sent

www.manaraa.com

to the algorithm. The simulation results are summarized in the third column group of Table 4.2. It shows the estimated jitter is very close to the added jitter.

In the 25Gb/s simulation, PRBS-7 data stream with different jitter component were generated. The data stream with different PJ (0,1.5ps, 4ps,8ps peak-peak value), DCD (0,2ps,4ps) and ISI jitter caused by channel A (7.2ps, pk-pk value) was sent to the algorithm. The simulation results are summarized in Table 4.3. The estimation error of ISI is less than 0.5 ps. The estimation error of PJ is close to 0 ps. DCD estimation were larger than the added caused by the jitter amplification due to channel loss.

4.4 Measurement Results

To verify the ISI (pk-pk) estimation and the accuracy and efficiency of the algorithm, a hardware test bench was used (shown in Figure 4.17) to measure the jitter components for 10Gb/s. A Tektronix BSA286C BERTscope was used to generate data stream with PJ. An Agilent Infiniium Wide-Bandwidth Oscilloscope was used to measure the jitter with internal software. The same PCB transmission line in simulation part A was used to generate the ISI jitter. All experiments were done at a data rate of 10Gb/s. We also used the result of JNEye with conventional decomposition algorithms as reference to compare the proposed method. JNEye is Intels state-of-the art jitter and noise link analysis tool for evaluate high-speed serial link performance. The jitter decomposition of this platform is based on hybrid algorithms with statistical, frequency-domain, and time-domain analysis with dedicated jitter components modeling and extraction. The accuracy of JNEye has been validated with both simulation and measurement correlations [22]. All tests for comparison were done at 10Gb/s and 25Gb/s.

www.manaraa.com

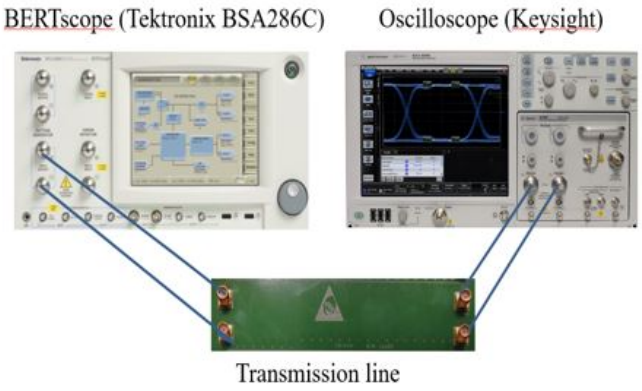

Figure 4.17 The 10Gb/s experimental test bench

4.4.1 Validation of the ISI Jitter pk-pk Estimation

Since the ISI jitter (pk-pk) generated by the PCB transmission line (channel A) is unknown, the ISI measurement result of the oscilloscope is as a reference. In the experiment, jitter free PRBS7 data generated by BERTscope was sent to the PCB transmission line and the output of transmission line was connected to oscilloscope to obtain the ISI jitter value. In the proposed method, the PRBS-7 data stream with ISI jitter was sent to the algorithm and the estimation ISI value was obtained. The channel A was used in JNEye to generate PRBS-7 with ISI and then decompose the ISI jitter. The comparison is listed in the third row of Table 4.1. In the added jitter term, the unknown for ISI means that the ISI jitter amount is unknown when the transmission line is added. The ISI value in the measurement is 5.1 ps and the estimation in JNEye is 4.4ps. The one in the proposed method is 4.6 ps. All values are very close. Therefore, the ISI modeling is validated. DCD in the simulation is 0ps while the one in the measurement is 0.5ps due to the instrument noise. PJ in the proposed method is 0ps (pk-pk) while the oscilloscope result is about 1.4ps.

www.manaraa.com

4.4.2 Comparison of the Accuracy and the Sample Data

In the hardware experiment, PRBS-7 data stream at 10Gb/s with different PJ (0ps and 20ps pk-pk) generated by BERTscope was sent to channel A and then sent to oscilloscope. PRBS7 data stream with different amounts of PJ and ISI jitter were sent to the proposed method and JNEye. Table 4.1 shows the comparison results among oscilloscope, JNEye and the proposed method. When the measurement result is stable, the experiment should run several minutes according to test experience. When the added ISI is 0, the oscilloscope result of ISI jitter is about 3.2 ps due to the cable while in the proposed algorithm the estimation is about 0 ps. The DCD measurement result is 0.8ps while in the proposed method is 0. The data sample in each simulation is 8k, but the oscilloscope requires at least 200k data in the experiment. Therefore, the proposed algorithm has comparable accuracy using fewer data samples than instrument and have the same accuracy compare to JNEye.

Table 4.1 Comparision (channel A) the proposed method/instrumen/ JNeye at 10Gb/s

Added jitter			Instrument			JNeye			The propose methed		
DCD	PJ	ISI	DCD	PJ	ISI	DCD	PJ	ISI	DCD	PJ	ISI
0	0	unknown	0.5	1.4	5.1	0	0	4.4	0	0.07	4.6
0	20	unknown	0.6	20.5	3.2	0	20.1	4.4	0	20.2	4.7
0	20	unknown	0.8	20.3	6.5	0	20	4.4	0	20.1	4.6

Table 4.2 Comparision the proposed method and JNeye at 10Gb/s

Channel	Added jitter			JNeye			The proposed method		
	DCD	PJ	ISI	DCD	PJ	ISI	DCD	PJ	ISI
Channel A	4	15	unknown	4.3	15.5	4.4	0	0.07	4.6
	10	10	unknown	10.6	10.5	4.4	10.7	10.6	4.7
	8	25	unknown	8.2	25.3	4.4	8.6	25.2	4.6
Channel B	0	0	unknown	4.5	4.48	3.5	0	0	3.6
Channel C	0	0	unknown	0	0	9.2	0	0	9.3

www.manaraa.com

Table 4.3 Comparision the proposed method and JNeye for channel A at 25Gb/s

Added jitter			JNeye			The proposed method		
DCD	PJ	ISI	DCD	PJ	ISI	DCD	PJ	ISI
0	0	unknown	0	0	7.58	0	0.07	7.28
0	8	unknown	0	8.36	7.58	0	7.99	7.65
2	1.5	unknown	2.2	1.56	7.58	2.34	1.63	7.61
4	4	unknown	4.5	4.48	7.58	2.64	3.75	7.24

We also modeled channel B with 3dB loss and channel C with 5dB loss at 5GHz to verify the proposed method. With different ISI, the comparison of JNEye and the proposed method is listed in the sixth and seven row of Table 4.2. It shows that both of two methods have the same estimation accuracy with the same sample data at 10Gb/s. For the 25Gb/s comparison, PRBS7 data stream with different PJ (1.5ps,4ps,8ps pk-pk), DCD (0ps,2ps, 4ps) and ISI jitter generated from channel A were sent to JNEye and the proposed method. Table 4.3 shows the estimation results in the proposed method and the JNEye estimation result. Both of two methods have the same estimation accuracy with the same sample data. However, the proposed method has two obvious advantages. First, the JNEye or other similar commercial simulation platform with different jitter decomposition methods is only applied in off-chip simulation, while the comparator based method can be applied on chip design with less complexity and low cost and have a great potential to reduce the whole test cost. Second, the proposed method can provide ISI analysis in detail for each ISI binary combination.

4.5 Conclusion

An efficient and accurate comparator based method is presented that simultaneously extracts periodic jitter, duty cycle distortion and ISI jitter. It uses Boolean output from a network of simple low-cost comparators for decomposing the jitter components instead of using the much more expensive TIE data. This method is based on time-domain ISI modeling which is simpler than the conventional cursor convolution technique

www.manaraa.com

while provides precise ISI analysis for each binary combination. It utilizes significantly fewer data samples than standard instrument test while maintaining sufficiently high estimation accuracy in both clock pattern and data pattern. Comparison of results among simulation, the hardware tests and Intels state of the art jitter decomposition simulation platform demonstrate the accuracy of the proposed jitter method. Beside the above advantages, one significant property of the comparator-based method is that it offers great potential for being adopted for on-chip test implementation, which could lead to significant benefits in test time and test cost reduction.

4.6 Acknowledgment

The authors would like to thank Intel for providing JNEye hardware validation in this chapter.

www.manaraa.com

REFERENCE

[1] International Technology Roadmap for Semiconductors, 2011 edition, [Online]. Available: http://public.itrs.net

[2] M. P. Li, "Jitter, noise and signal integrity at high-speed," Prentice Hall Pearson Education, 2007

[3] H. Pang, J. Zhu, and W. Huang, "Jitter decomposition by fast Fourier transform and time lag correlation," in IEEE Int. Conf. Communications, Circuits and Systems ICCCAS, Jul. 2009, pp. 365368.

[4] T. Yamaguchi, H. Hou, K. Takayama, D. Armstrong, M. Ishida et al., "An FFT-based jitter separation method for high-frequency jitter testing with a 10x reduction in test time," IEEE Int. Test Conf. (ITC07), pp. 18, Oct. 2007.

[5] M. Li, J. Wilstrup, R. Jessen, and D. Petrich, "A New Method for Jitter DecompositionThrough its Distribution Tail Fitting," IEEE Int. Test Conf. (ITC99), pp. 788794,Sep.

[6] G. Hnsel, K. Stieglbauer, G. Schulze, and J. Moreira, "Implementation of an Economic Jitter Compliance Test for a Multi-Gigabit Device on ATE," IEEE Int. Test Conf. (ITC04), pp. 13031312, Oct. 2004.

[7] D. Hong and K.-T. Cheng, "An Accurate Jitter Estimation Technique for Efficient High Speed I/O Testing," IEEE Asian Test Symp. (ATS07), pp. 224229, Oct. 2007.

www.manaraa.com

[8] J.-L. Huang, "A Random Jitter Extraction Technique in the Presence of Sinusoidal Jitter," IEEE Asian Test Symp. (ATS06), pp. 318326, Nov. 2006.

[9] R. Stephens, "Separation of Random and Deterministic Components of Jitter," U.S. Patent 7149638, Dec. 12, 2006.

[10] R. Stephens, "Separation of a Random Component of Jitter and a Deterministic Component of Jitter," U.S. Patent 7 191080, Mar. 13, 2007.

[11] S. Wisetphanichkij and K. Dejhan, "Jitter Decomposition by Derivatived Gaussian Wavelet Transform," IEEE Int. Symp. Communication and Information Technology (ISCIT04),vol. 2, pp. 11601165, Oct. 2004.

[12] F. Nan, Y. Wang, F. Li, W. Yang, and X. Ma, "A Better Method than Tail-fitting Algorithm for Jitter Separation Based on Gaussian Mixture Model," J. of Electronic Testing: Theory and Applications, vol. 25, no. 6, pp. 337342, Dec. 2009.

[13] Q. Dou and J. Abraham, "Jitter Decomposition by Time Lag Correlation," IEEE Int. Symp. Quality Electronic Design (ISQED06), Mar. 2006

[14] "Physical Layer Performance: Testing the Bit Error Ratio (BER)," Technical Article, Maxim Inc., Sep. 2004.

[15] Agilent Technologies Application Note, "Analyzing Jitter Using Agilent EZJIT Plus Software," Literature Number AN-5989- 3776EN, 2005

[16] Y.Duan, D.J.Chen "Accurate Jitter Decomposition in High Speed Link," accepted by IEEE International Symposium on VLSI Test Symposium (VTS),2017.

[17] D.Oh and X.Yuan ,"High speed signaling: jitter modeling analysis, and budgeting," Prentice Hall Pearson Education, 2011, Chapter 3, pp.44-64

www.manaraa.com

[18] V.K Sharma,J.N.Tripath, R.Nagpal, S.Deb and R.Malik,"A Comparative analyis of jitter estimation techniques," in proc.IEEE International Conference on Electronics,communication and computational engineering(ICECCE), 2014

[19] J. Buckwalter, B. Analui and A. Hajimiri, "Predicting data dependent jitter," IEEE Trans. Circuits and Sys. II, vol. 51, pp. 453-457 Sep. 2004.

[20] Liu H, He Z. "A sliding-exponential window RLS adaptive algorithm," properties and applications. Signal Process 1995;45:35768.

[21] Gustafsson F. "Adaptive filtering and change detection," West Sussex: Wiley; 2000.

[22] M. Li, M. Shimanouchi, H. Wu, "Advancements in High-Speed Link Modeling and Simulation," An Invited Paper, Proc. of IEEE Custom Integrated Circuits Conference (CICC), 2013.

www.manaraa.com

CHAPTER 5. A LOW-COST DITHERING METHOD FOR IMPROVING ADC LINEARITY TEST APPLIED IN USMILE ALGORITHM

Analog-to-digital converter (ADC) is an important component in electronics design. One of the difficulties being faced is to be able to accurately and cost-effectively test the continually higher performance of ADCs under budget constraint. Test time for static linearity is a major portion of the total test cost. Our group proposed an ultrafast segmented model identification of linearity error (uSMILE) algorithm before estimating linearity, which reduces 99% of the test time compared to the conventional method. However, this algorithm produces large estimation errors in low resolution ADCs (10-12 bits) when the input is a ramp signal, at which the quantization noise of ADC becomes a dominant part in the total noise. In this study, we proposed three types of distribution dithering methods adding to the ramp input signal to reduce the estimation error when uSMILE was applied in low resolution ADCs. Fixed pattern was proved to be the most efficient and cost-effective method by comparing to the Gaussian, uniform, and fix-pattern distributions. The simulation results indicate that the estimation error can be significantly reduced in a 12-bit SAR ADC with effective dithering. Furthermore, a hardware evaluation board with commercial ADC products was used to validate the effectiveness of the fix-pattern dithering method, our measurement shows the INL estimation error can be reduced to less than 0.1 LSB.

www.manaraa.com

5.1 Introduction

The analog-to-digital converter (ADC) is one of the most common-used, mixed-signal products [1]. Testing high-performance ADC under cost/budget constraints remains a huge challenge for decades in the semiconductor industry. As manufacturing costs drop gradually, the ADC testing cost becomes a major portion of the overall cost. ADC linearity test including integral and differential nonlinearity (INL and DNL) test is cost-sensitive and time-consuming. The linearity test is conventionally conducted by a histogram method [2-4] using either a pure sine wave, or a very linear ramp or triangle wave as stimulus.

There are two key challenges in ADC linearity test: linear stimulus generation [5-7] and data acquisition. For the linear stimulus generation, the stimulus are required to offer 3-4 bits resolution higher than the ADC under the test. However, it is difficult to generate stimulus for the high solution ADC (e.g. 16-bit). Meanwhile, the histogram method requires much more samples than the number of transitions in the ADC for data acquisition. It uses several tens even hundreds of hits per code to accurately test the ADC nonlinearity, which results in a long data acquisition time. For high resolution ADCs higher than 16-bit, it is usually even not practical to fully test the ADC linearity due to the extremely long test time.

Researchers have proposed various methods to reduce the stringent requirements on the linearity of stimulus and data acquisition. A stimulus error identification and removal (SEIR) algorithm using nonlinear stimulus was introduced previously by our group [8-10]. It demonstrated that a 7-bit linear ramp signal can be used to test a high resolution ADC and achieved more than 16 bits accuracy. However, the SEIR method was based on the histogram method, which means its data acquisition time is still very long. Our group also attempted a system identification approach to identify the parameters in a pipeline ADC and then reconstruct the full code linearity information [13]. In

www.manaraa.com

addition, another test method was also proposed to estimate the ADC′s INL based on fast Fourier transform (FFT) [11-12]. Some researchers also combined Kalman filtering in the standard histogram method and presented new ADC linearity test algorithms that can reduce the data acquisition time by several times [14-15]. Goyal et al. [16] developed a selective code measurement method to reduce the test time of SAR ADCs. All the methods in the literature attempt to reduce linearity test time by sacrificing some aspects of test accuracy than histogram method.

Yu, et al. [17] in our group proposed an ultrafast segmented model identification of linearity error algorithm (uSMILE) for accurate linearity test in a high resolution ADC with dramatically reduced data acquisition. With the segmented non-parametric model, the algorithm can reduce the test data to 1% sample and achieve a test accuracy superior to the histogram method.

However, the uSMILE algorithm caused large estimation INL error when the input is ramp signals applied low-mid resolution ADC because the quantization noise is the dominant part of the ADC noise. In the ADC production test, large testing errors will be resulted in yield loss. A quarter LSB estimation error is difficult to achieve under limited samples. Since the application of low-mid resolution ADC is very common and its product volume is in the order of billion each year, it has a great potential to reduce the test cost by diminishing the estimation INL error of uSMILE for low-mid resolution ADC.

This study was conducted by different dithering methods to identify the most effective and easy-to-implement way to reduce the estimation error. The dithering includes the Gaussian, uniform, and fix-pattern distributions. The fixed-pattern dithering method was verified to be the most efficient due to its less complexity of implementation and effectiveness in averaging the quantization noise. The estimation error can be reduced to less than 0.5LSB.

The rest of this chapter will be organized as follows. In section II, a brief overview of

www.manaraa.com

uSMILE algorithm will be presented and quantization noised causing the INL estimation error will be also addressed. In section III, we will analyze different dithering techniques to average the quantization error and a low-cost dithering generating circuit will be introduced. In section IV, the simulation result for different dithering approaches are presented. In section V, measurement results will show the validation of our proposed methods and section VI will present the conclusions of this study.

Table 5.1 SNR comparison in selected industrial ADCs

ADC	SNR(Ideal)	SNR(Measured)
20bit (LTC-2378)	122	104
18bit (LTC-2379)	110	101.2
18bit (TI ADS8881)	110	100
16bit(TI ADS8353)	98	89
14bit(TI ADS7853)	86	82
12bit(TI ADS7253)	74	73.5

5.2 Problem Statement

In this section, the noise in ADC is first discussed. We investigated many commercial ADC with different resolutions. For some low-resolution ADCs, the noise is dominated by the quantization error. Then, the uSMILE algorithm is reviewed and the INL estimation error caused by quantization noise in uSMILE is addressed.

5.2.0.1 Noise in ADC

The noise in an ADC is composed by two components: true noise N_r and quantization noise N_q. The quantization noise (or quantization error) is due to the finite resolution of the ADC. The true noise is from external sources such as input signal noise and random clock jitter, and the ADC system noise such as aperture jitter, KT/C in sampling capacitors, comparator noise and residue amplifier. One specification of ADC performance is

www.manaraa.com

signal (S) to noise (N) ratio (SNR), which is caculated by equation 5.1

$$SNR = 20log(S/N) \tag{5.1}$$

For ADCs with different resolutions, the noise contributions is different. The study focuses on the Nyquist rate ADCs. The comparison of various industry ADC products from 10-bit to 20-bit is shown in Table 5.1. Only the quantization noise is considered in the noise term N of ideal SNR calculation. The actual measured SNR is obtained from the datasheet for comparison. For fair comparison, only low frequency measurement is used. The comparison shows that the noise in high resolution ADC (16-20 bits) is dominated by the true noise. However, the noise in low and middle resolution ADC (10-14 bits) is dominated by quantization noise. For example, the theoretical calculation of SNR in 20-bit ADC is about 122dB while the measurement result is about 104dB, where the SNR is mainly limited by the true noise. The calculated SNR in 12-bit ADC is about 74dB but some 12-bit ADCs can achieve SNR close to 74dB SNR.

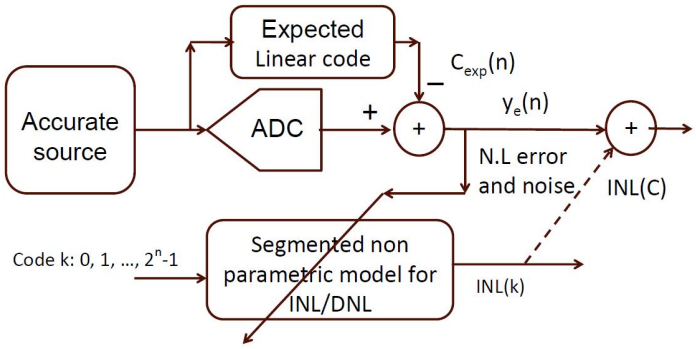

Figure 5.1 uSMILE algorithm implementation

5.2.1 uSMILE Algorithm

The uSMILE algorithm is developed to reduce the test time for high resolution ADCs. It takes a system identification approach with a segmented non-parametric INL model. It assumes that for an N-bit ADC, all the INL/DNL errors are highly correlated and

www.manaraa.com

are deterministic functions of a much smaller number of independent errors (component size error, parasitic, voltage coefficients, etc.), which is true for most ADC architectures except flash ADCs and Sigma-Delta ADCs.

In this model, the INL is first broken into multiple-most-significant-bit (MSB) segments. Each MSB segment has an error term $E_M(C_{MSB})$ corresponding to the MSB code C_{MSB}. Then, for each MSB segment, the small INL curve can be further divided into smaller segments for intermediate significant bits (ISB). Each ISB segment has an error term $E_I(C_{ISB})$ corresponding to the ISB code C_{ISB}. Similarly, each ISB segment can be further divided into smaller segments for all the less significant bits (LSB). For example, an INL curve of 18-bit ADC can be broken into 64 MSB segments, 64 ISB segments,64 LSB segments if 6-bit MSBs, 6-bit ISBs and 6-bit LSBs are used. There are 64 MSB error terms that denoted as $E_M(0)$, $E_M(1)$, $\cdots$, $E_M(63)$ and 64 ISB error terms $E_I(0)$, $E_I(1)$, $\cdots$, $E_I(63)$. Similarly, E_L has 64 E_L. The estimated INL for code C can be expressed by

$$INL_{est}(C) = E_M(C_{MSB}) + E_I(C_{ISB}) + E_L(C_{LSB}) \tag{5.2}$$

For instance, the code of ADC output is 1100110111100000111, then $C_{MSB} = 110011$, $C_{ISB} = 0111100$ and $C_{LSB} = 000111$. With the above model, we first identified all the independent error terms and then used the model to compute the full code of INL/DNL. The implementation of this method is shown in Figure 5.1. The converted output code from the ADC under test is compared with the expected linear code C_{exp}. The difference y_e reflects the total ADC error caused by the ADC nonlinearity and the noise. Therefore, the input and output relationships can be expressed as:

$$C_{exp} - C + N_{noise} = E_M(C_{MSB}) + E_I(C_{ISB}) + E_L(C_{LSB}) \tag{5.3}$$

If the average value of the N_{noise} term in each segment is zero, the estimated INL can be accurately estimated as the true INL. The input signal can be either a sine wave or a linear ramp signal. With the sine wave input, the expected code can be obtained

www.manaraa.com

using Fourier transform. The DC and the fundamental are the corresponding linear part. With ramp as input signal, least square can be used to extract the best fit line as the expected code.

5.2.2 Problem Statement

In the uSMILE algorithm, the estimated MSB segment error for code i (MSB code) is shown in equation 5.4. The exact derivation is in the Appendix A. In this equation, the estimation error is equals to the average value of $C_{exp} - C + N_{noise}$ of i-th MSB segment.

$$E_M(i) \approx \frac{1}{\#C_{MSB} == i} \sum_{C_{MSB}==i} (C_{exp} - C) + \frac{1}{\#C_{MSB} == i} \sum_{C_{MSB}==i} (N_{noise}) \qquad (5.4)$$

These similar equations can be derived for ISB and LSB error terms respectively. For equation 5.4, if the mean value of the noise in each segment is not zero, the estimation will not be accurate. Unlike the histogram ramp test, uSMILE algorithm can use a

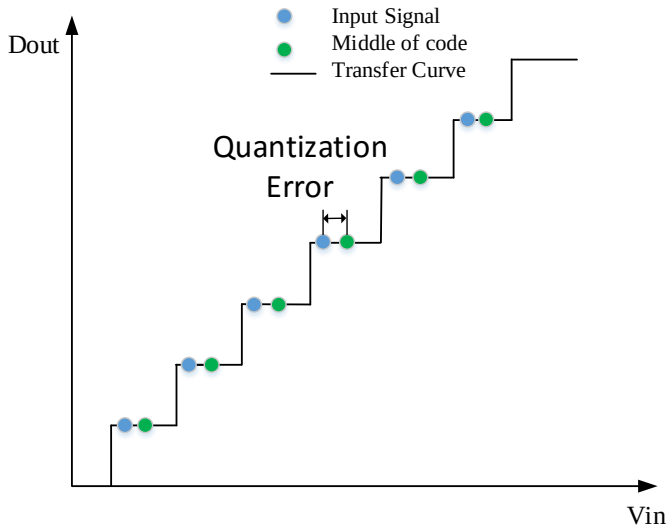

Figure 5.2 One MSB segment of ADC transfer curve and quantization definition

ramp signal close or less than 1 hit/code to effectively test the ADCs INL/DNL. If the ramp signal with 1 hit/code is used, each increment is 1 LSB in voltage. It is unknown the position where each input signal hits in the transfer curve of the ADC as shown in

www.manaraa.com

Figure 5.2. It is a zoomed-in part of the transfer curve of a MSB segment. The blue dots represent the actual input signals and the green dots are the middle points for each code bin. The difference between green and blue dots in each LSB is the quantization error, ranging from -0.5 LSB to 0.5 LSB. Each input signal may has the same quantization error in this segment if the LSB segments have good linearity and the increment of the voltage is exactly 1 LSB. As discussed previously in part A, the noise in the 10-12 bits ADC is dominated by the quantization and the true noise could be ignored. Therefore, the average quantization noise in each MSB segment is not zero and the estimated E_M is not accurate based on equation 5.4.

For a randomly generated 12-bit SAR ADC in Matlab, a linear ramp input signal with 1 hit/code without extra noise was used in the simulation. We used 4-bit MSB, 4-bit ISB and 4-bit LSB as the segmentation in the uSMILE algorithm. The estimated INL and true INL are plotted in top plot of Figure 5.3. The INL estimation error is as large as 0.5 LSB, shown in the bottom plot of Figure 5.3 bottom.

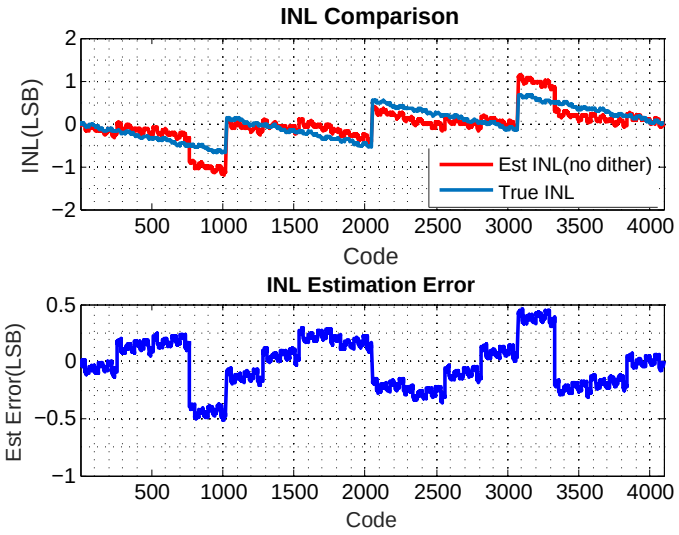

Figure 5.3 Comparison of true INL and estimation INL for uSMILE without input noises

It shows the quantization noise of each sample and there is a clear pattern shown

www.manaraa.com

in the quantization noise distribution. Multiple red lines plotted in Figure 5.4 show the averaged quantization noise of each MSB segment. The averaged quantization noise range is very wide and the distribution matches the shape of the estimation error plotted in Figure 5.3 bottom. As many ADCs with similar resolution have a very good SNR. The application of uSMILE in these ADCs will be an issue with the dominated quantization error. To reduce the estimation error, a dither signal has to be added to the input signal to average the quantization noise.

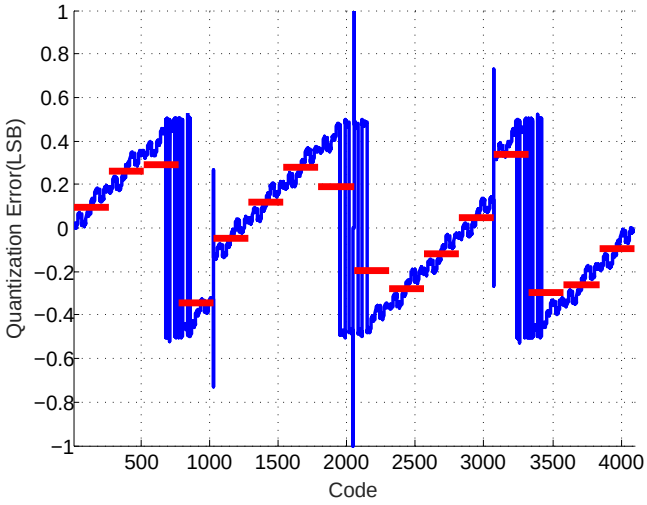

Figure 5.4 Quantization Error for each sample

5.3 Proposed Dithering Method

In this section, three commonly-used dithering forms were investigated, including uniform noise, fixed dithering pattern and Gaussian noise to reduce the INL estimation errors. The implementation is to add extra dither to the ramp input to test the ADC, shown in Figure 5.5.

www.manaraa.com

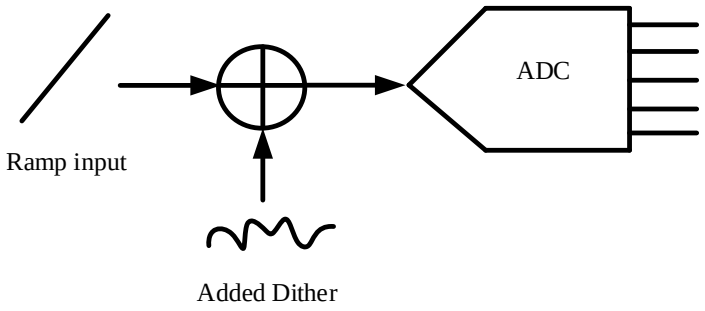

Figure 5.5 Dithering Implementation

5.3.1 Dithering Amplitude

From equation 5.4, the estimation INL error is approximately equal to the averaged quantization noise in each MSB segment. The aim of adding the dither is to make the averaged quantization noise value in every segment to be zero.

Assume that every code bin width in each MSB segment is exactly 1 LSB for the simplicity in the rest of the chapter. Define V_{in} as the input of an n-bit ADC, V_q as the quantization noise for each sample which the pdf is shown in Figure 5.6. For ADC output code C, the relation among V_{in} and quantization noise for k-th sample can be expressed as:

$$\begin{aligned} V_{in}(k) &= C \cdot LSB + V_q(k) + 0.5 \cdot LSB \\ V_q(k) &\in [-0.5\ 0.5]\ LSB \end{aligned} \tag{5.5}$$

where LSB is ideal LSB voltage.

Define that $V_d(k)$ is the dithering signal added to k-th sample, $V_{ind}(k)$ is the actual input of ADC after adding the dither. The $V_{ind}(k)$ is then expressed as:

$$V_{ind}(k) = C \cdot LSB + V_q(k) + 0.5 \cdot LSB + V_d(k) \tag{5.6}$$

With added dither, the ADC's output code may not be the same. If the dither is too large, the output code will increase. Then, the quantization error will change according

www.manaraa.com

to the actual output code. No matter how the code changes, the quantization error is alway within +/- 0.5 LSB. The new quantization noise $V_{dq}(k)$ can be expressed as:

$$\begin{aligned} V_{dq}(k) &= V_q(k) + V_d(k) - n \cdot LSB \\ V_{dq}(k) &\in [-0.5\ 0.5]LSB \end{aligned} \tag{5.7}$$

where n is an integer to make V_{dq} within +/- 0.5 LSB.

For many samples in each segment, if the dithering signal $V_d(k)$ is too small (much smaller than 1 LSB), the voltage $V_{ind}(k)$ after adding the dither is still close to the original signal $V_{in}(k)$ and the quantization noise $V_{dq}(k)$ hardly changes (at this case $n = 0$). Thus, the ADC code will remain as the same code k even after adding dither signal under this circumstance.

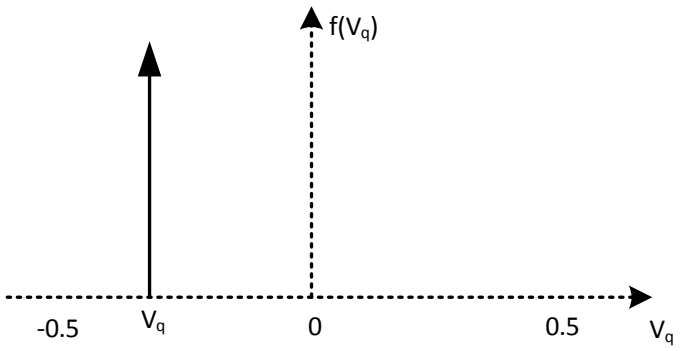

Figure 5.6 Quantization Noise Function

If the term of $V_q(C_{MSB}) + V_d(k)$ is beyond +/- 0.5 LSB (larger than 0.5 LSB or smaller than -0.5 LSB), the sampled voltage $V_{ind}(k)$ will result in a different code $(C-n)$ compared to V_{in}. Therefore, the new quantization error $V_{dq}(k)$ will be shifted back to +/- 0.5 LSB. The amount of new quantization noise $V_{dq}(k)$ becomes $V_q + V_d - nLSB$ and n is an integer. In this case, the probability density function (pdf) f_{dq} of new quantization noise V_{dq} is the convolution of the original quantization noise distribution pdf function $f_q(v_q)$ and the pdf of the dither signal f_d . The excessive parts of of 0.5 LSB and -0.5 LSB are needed to be folded back within the +/- 0.5 LSB range.

www.manaraa.com

We analyzed different dithering distribute to change the code and made the average new quantization noise $V_{dq}(k)$ within MSB segment to zero based on above discussion. We assume the initial quantization error V_q falls in [-0.5, 0] LSB.

5.3.2 Uniform Noise Requirement

In order to average the quantization noise to zero, the uniform distribution dithering amplitude should satisfy with $a + b = 0$, $b - a <= 1$. The derivation is as follows.

The uniform dithering follows a continuous uniform distribution $f_q(V_d)$ given by equation 5.8 with the assumption that $V_q < 0$ is given by

$$f_d(V_d) = \begin{cases} \frac{1}{b-a} & a \leq V_d \leq b \\ 0 & otherwise \end{cases} \tag{5.8}$$

Figure 5.7 shows the convolution of the quantization error and pdf of the uniform before folding. The upper bound is $V_q + b$ and the lower bound is $V_q + a$.

The quantization noise V_q >0.5 LSB and V_q < −0.5LSB are folded back to the [−0.5 0.5] for the new quantization noise pdf $f_q(V_{dq})$, thus the range between $V_q + a$ and -0.5 LSB is moved to $V_q + a + 1$ and 0.5 LSB as shown in Figure 5.8.

The expected value in the new quantization noise distribution $E(V_{dq})$ is given by equation 5.9

$$\begin{aligned} & E(V_{dq}) \\ & = \int_{-\infty}^{\infty} f_{dq}(v_{dq}) \cdot (v_{dq}) dv_{dq} \\ & = \frac{1}{b-a} \left[\int_{-0.5}^{v_q+b} v_{dq} dv_{dq} + \int_{V_q+a+1}^{0.5} v_{dq} dv_{dq} \right] \\ & = (v_q + \frac{1}{2}) \cdot (1 - \frac{1}{b-a}) \end{aligned} \tag{5.9}$$

From equation 5.9, the expected value $E(V_{dq})$ depends on the initial quantization noise V_q with a range from -0.5 to 0.5 LSB. To minimize the expected value close to zero, choosing $1/(b-a)$ can be made close to one. When $1/(b-a)$ is exactly equal to one,

www.manaraa.com

thus $V_q + a + 1$ is equal to $V_q + b$. The expected value $E(V_{dq})$ is exactly zero in this case no matter what V_q initial value is as shown in equation 5.10 and Figure 5.9.

$$E(V_{dq}) = \int_{-\infty}^{\infty} f_{dq}(v_{dq}) \cdot (v_{dq}) dv_{dq} = \frac{1}{b-a} \int_{-0.5}^{0.5} v_{dq} dv_{dq} = 0 \tag{5.10}$$

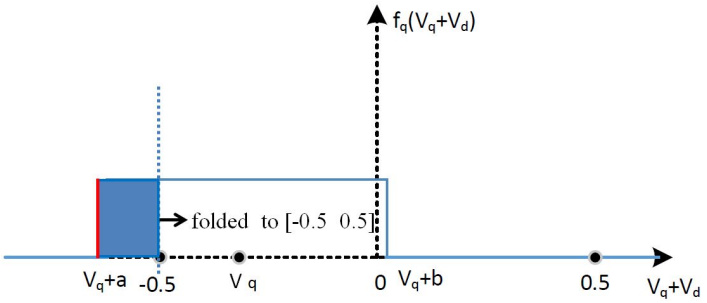

Figure 5.7 Convolution of quantization error function with uniform distribution pdf

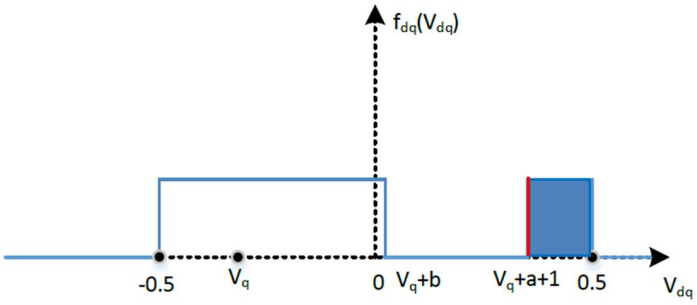

Figure 5.8 Pdf of quantization error after adding uniform noise

5.3.3 Fixed Dithering Pattern

A fixed dithering pattern refers to a series of m kinds of amount dithering signal data and theirs values are repeated with a period of m. Each value has a probability equal to $1/m$.

www.manaraa.com

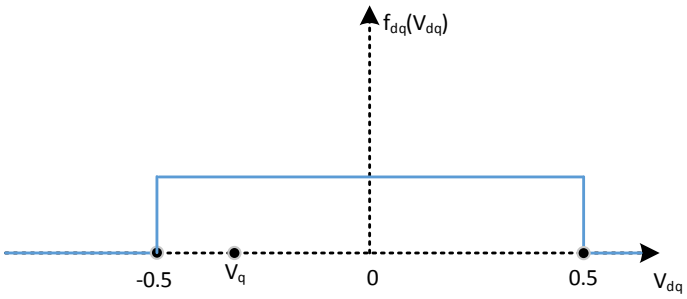

Figure 5.9 Pdf of quantization error after adding uniform noise with exactly 1 LSB width

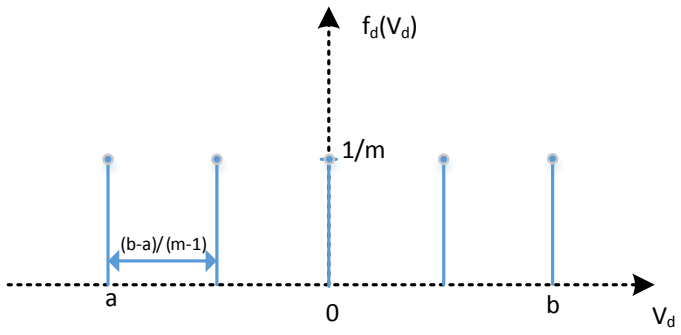

Figure 5.10 Pdf of fixed pattern dithering

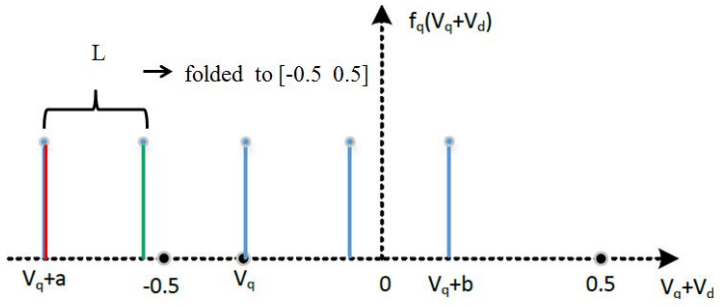

Figure 5.11 Convolution of quantization error function with pdf of fixed pattern dithering

www.manaraa.com

The fixed dithering pattern follows a discrete uniform distribution and we use probability mass function (pmf) for the discrete random variables. For the dither signal V_d with m bins from a to b following the discrete uniform distribution shown in Figure 5.10. The lower bound is a and the upper bound is b. Assume the initial quantization noise V_q falls in [−0.5 0] LSB , the convolution of dithering distribution function f_d and the quantization function f_q is shown in Figure 5.11. The values after dithering lower than -0.5 LSB is folded back to the (-0.5, 0.5) LSB range in Figure 5.12.

Define that there are L values lower than -0.5 and the first value greater than -0.5 is V_x. V_q' is defined as the distance between the -0.5 and V_x so that $V_q' = V_x + 0.5$. The expected value is:

$$\begin{aligned}
&E(V_{dq}) \\
&= \sum_{k=0}^{L-1} \frac{1}{m}(V_q + a + 1 + k \cdot \frac{b-a}{m-1}) \\
&+ \sum_{k=L}^{m-1} \frac{1}{m}(V_q + a + k \cdot \frac{b-a}{m-1}) \\
&= \sum_{k=1}^{L} \frac{1}{m}(V_q' + 0.5 - k \cdot \frac{b-a}{m-1}) \\
&+ \sum_{k=0}^{m-L-1} \frac{1}{m}(V_q' - 0.5 + k \cdot \frac{b-a}{m-1}) \\
&= -0.5 + \frac{b-a}{2} + \frac{L}{m}\left[1 - \frac{m(b-a)}{m-1}\right] + V_q'
\end{aligned} \tag{5.11}$$

In this equation, the value of $-0.5 + \frac{b-a}{2}$ is fixed once a and b are defined. V_q' is between 0 and $\frac{b-a}{m-1}$ which can be minimized by using small value of $b-a$ and large value of m. $\frac{L}{m}[1 - \frac{m(b-a)}{m-1}]$ will depend on the value of L. However, L can be any value between 0 and $\frac{m}{2}$. To make the second term small, the value of $1 - \frac{m(b-a)}{m-1}$ should be minimized which will result in $b-a = \frac{m-1}{m}$ to make it be 0. In this case, the expected value becomes

$$E(V_{dq}) = V_q' - \frac{1}{2m} \tag{5.12}$$

Since V_q' is between 0 and $\frac{1}{m}$, the expected value will be between $-\frac{1}{2m}$ and $\frac{1}{2m}$, thus the

www.manaraa.com

maximum INL estimation error (absolute value) caused by quantization error being $\frac{1}{2m}$.

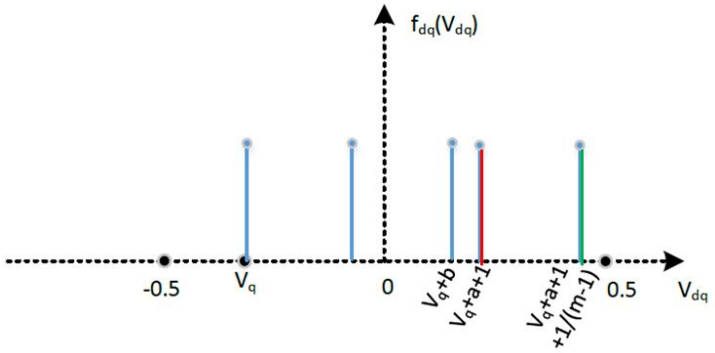

Figure 5.12 Pdf of quantization error after adding fixed-pattern dithering

5.3.4 Gaussian Noise

Gaussian noise following $N(0, \sigma^2)$ is added to an input ramp signal. The convolution of the quantization error with the Gaussian noise is shown in Figure 5.13. The beyond -0.5 LSB part is folded back to [0 -0.5] LSB interval. After summation, the new quantization error probability distribution $f_{dq}(V_{dq})$ is shown in Figure 5.14. The *pdf* of new quantization error depends on the value of V_q and the variance of the noise. Whatever the variance is, the pdf will have non-zero values from $-\infty$ to $+\infty$. For other values beyond $+/-3\ \sigma$, we can treat them as zero since they will hardly change the pdf of the quantization error. When the noise variance is small, the average quantization noise will remain close to the original value. When the noise variance is large, the average quantization is like to be close to 0 since the pdf of quantization is flatter. However, the large noise variance, in turn, will affect the uncertainty of the uSMILE estimation.

5.3.5 Hardware Implementation Comparison

Regarding the implementation, many researchers have proposed efficient and compact design for Gaussian noise generator [23-25], but the circuit complexity and area overhead

www.manaraa.com

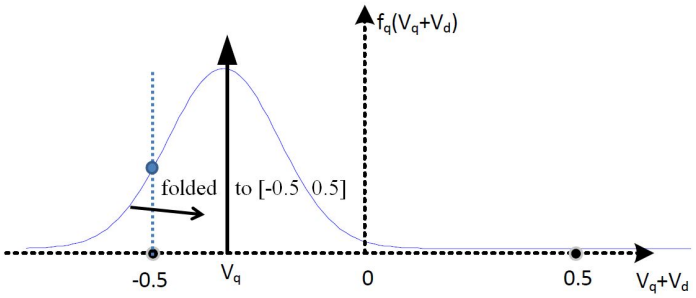

Figure 5.13 Convolution of quantization error with normal distribution probability function

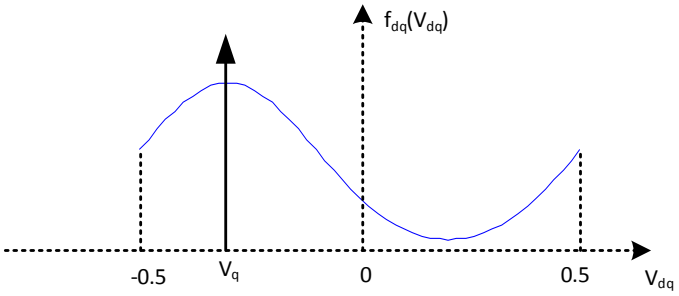

Figure 5.14 Probability density function of the quantization error after adding Gaussian noise

are against the purpose of saving the test cost. The uniform dithering generator provides the best quantization noise averaging capability but requires the probability density function in a continuous form, which is also difficult to implement. The fixed-pattern dithering generator also provides excellent capability of averaging the quantization noise to zero. The implementation is much easier compared with the other two methods since it is easily achieved to generate a finite number of dithering signal. A low cost dithering hardware implementation for the ADC linearity test were previously proposed [26].

The basic idea is shown in a SAR ADC. Figure 5.15 shows a conventional binary-weight N-bit SAR ADC. There are N capacitors and one dummy capacitor C_t which is

www.manaraa.com

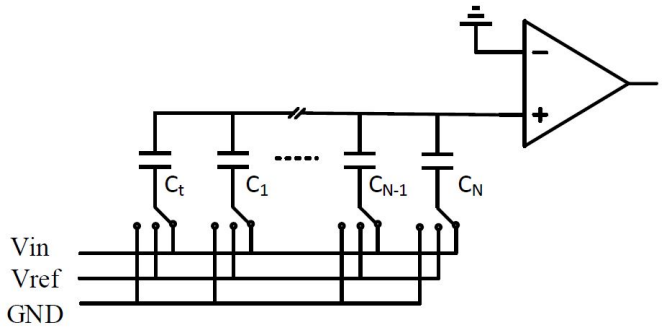

Figure 5.15 Original SAR ADC structure

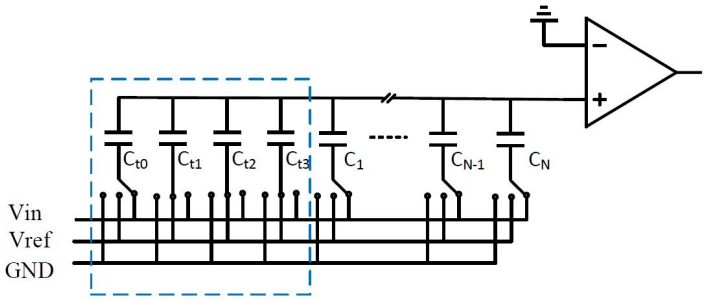

Figure 5.16 The linearity test operation of the modified SAR ADC structure

equal to the value of C_1. In the proposed circuit, the dummy capacitor C_t is modified into a small capacitor array in Figure 5.16. C_{t1} to C_{t3} is a 3-bit capacitor DAC and C_{t0} is a dummy capacitor with a value equal to C_{t1}. Note that the total capacitance of the capacitor array remains the same as in original SAR ADC structure. More switches are added to control each capacitor in the small CDAC array. C_{t1}, C_{t2} and C_{t3} are controlled by a pseudorandom binary sequence (PRBS) pattern generator to generate the fixed pattern dithering.The PRBS 3 generator consists of a XOR gate, and 3-bit shift register. The PRBS 3 generator will update the pattern for every new ADC sample.

www.manaraa.com

5.4 Simulation Results

To demonstrate the validity of the proposed method for uSMILE algorithm in ADC linearity test, extensive simulation has been carried out. A 12-bit SAR ADC is modeled with random capacitor mismatches. For the segmentation, 4-bit MSB, 4-bit ISB and 4-bit LSB are used. Three linear ramp signals with Gaussian noise, uniform noise and fixed-pattern dithering are used as the input of the same ADC. The ramp signal is 1 hit/code with 1 LSB increment each time.

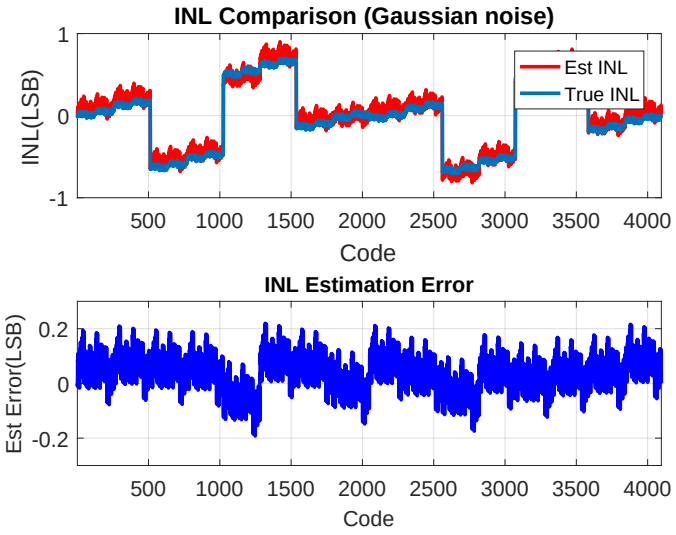

Figure 5.17 INL comparison after adding Gussian dithering

Gaussian noise with 0 mean and 0.5 LSB sigma is added in the simulation. There are 48 variable in the uSMILE method and there are around 4,000 samples in the simulation. The test uncertainty has a variance being $\sigma^2 \times 48/4000 \approx 0.01\sigma^2$, where σ^2 is the Gaussian noise variance. The INL comparison is shown in Figure 5.17. The estimated INL matches with the true INL but with noticeable estimation error. It also shows the difference between the estimated and the true INL. The maximum INL error is around 0.22 LSB. The 3 σ for estimation uncertainty is 0.3 LSB as analyzed earlier. The simulation matches the theoretical analysis.

www.manaraa.com

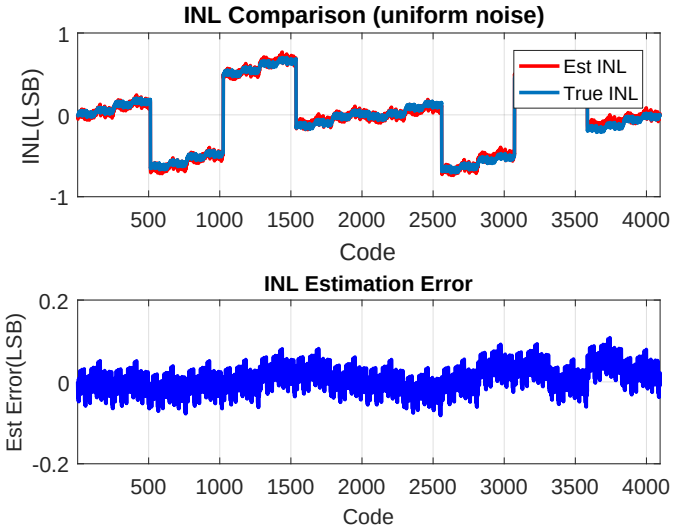

Figure 5.18 INL comparison after adding uniform dithering

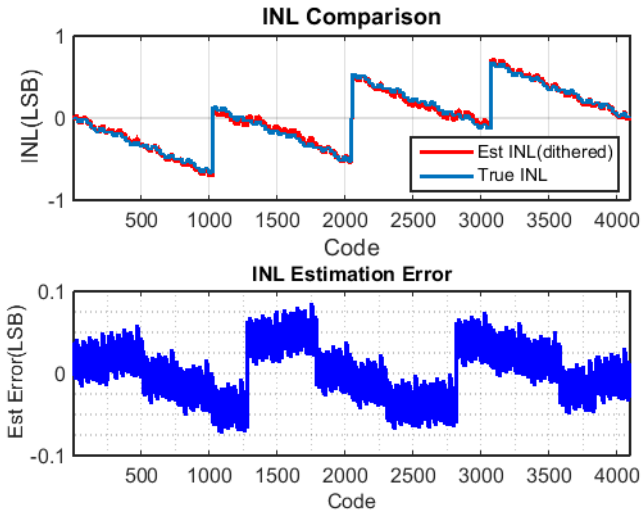

Figure 5.19 INL comparison and uSMILE estimation error after adding fixed dithering

For the uniformly-distributed noise, the interval only needs to be 1 LSB to effectively average the quantization error. In the simulation, 1 LSB wide uniform noise is used. The INL comparison and the estimation error difference is shown in Figure 5.18. The maximum estimation error is less than 0.1 LSB, showing better performance than the

www.manaraa.com

Gaussian noise effect.

The fixed-pattern dithering was implemented using the proposed architecture in section III. In Figure 5.3, the input signal had no noise and the INL estimation produces error as large as 0.5 LSB. Then, a 7-value fixed-pattern dithering was added to the input signal with 7/8 LSB difference between the smallest and largest dithering signal. The estimation is shown in Figure 5.19, which shows less than 0.5 LSB estimation error. In previous analysis, the estimation error can be reduce to $1/2m$ when m equals to 8 which is 0.0625 LSB.

Compare to the Gaussian, Uniform and Fixed pattern dithering, both uniform noise and fixed pattern have very good estimation. Considering the implementation, the fixed pattern dithering is feasible and low-cost one.

5.5 Measurement Results

The proposed method of adding dither to uSMILE algorithm has been validated by MATLAB simulation. However, in real testing, there are more unpredictable factors that cannot be simulated and the ultimate application of this algorithm is experimental testing. This proposed method is validated by measurement data using a 12-bit SAR ADC.

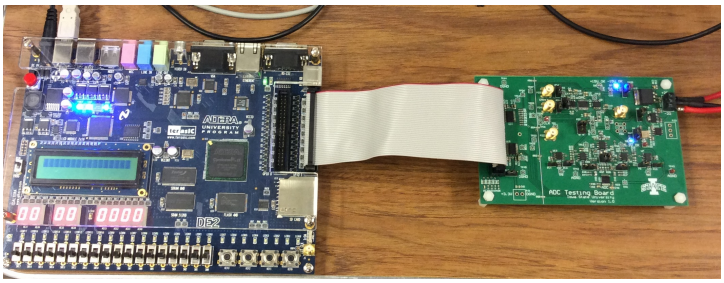

Figure 5.20 PCB Board Measurement Setup

The test setup is show in Figure 5.20. FPGA board was used to control the timing,

www.manaraa.com

control signal, and data storage. The test stimulus was generated from a 20-bit DAC AD8756. The ADC under test was ADS7253 (a 12-bit commercial ADC with 73.5dB SNR). The 20-bit DAC has a very good linearity, which can be used as histogram test to obtain a relatively accurate INL results. Since the resolution of the DAC is high, the DAC code can be programmed to generate a fixed pattern dithering. The fixed pattern added to the DAC code was a PRBS-3 code. The dither voltage range corresponds to -3/8 LSB to 3/8 LSB of ADC.

By applying the stimulus to the SAR ADC, the output of ADC was collected and sent to the uSMILE. The INL of ADC was also tested by 64 hits/code histogram ramp test. The DNL/INL comparison of uSMILE before and after adding dithering with histogram test is shown in Figure 5.21. For the INL/DNL comparison, the blue curve is the result of 64 hits/code histogram ramp which is treated as a standard. The red curve is result of 1 hit/code uSMILE before adding dithering. The cyan curve is the one with 1 hit/code uSMILE after adding dithering. Figure 5.22 shows the INL estimation error in the uSMILE before and after adding dithering. In the test results, the two ends comparison are not shown due to saturation near top and bottom for the histogram. From the result, it shows that the INL estimation error is as large as 0.5 LSB without a dithering technique. The INL estimation error is less than 0.2 LSB with the fixed pattern dithering.

In the measurement results, there are clearly some segmented errors in the INL estimation which is caused by the unwhitened quantization noise. After applying the dithering, the segmented error shape was significantly reduced and the INL estimation error was reduced from 0.6 LSB to 0.2 LSB.

www.manaraa.com

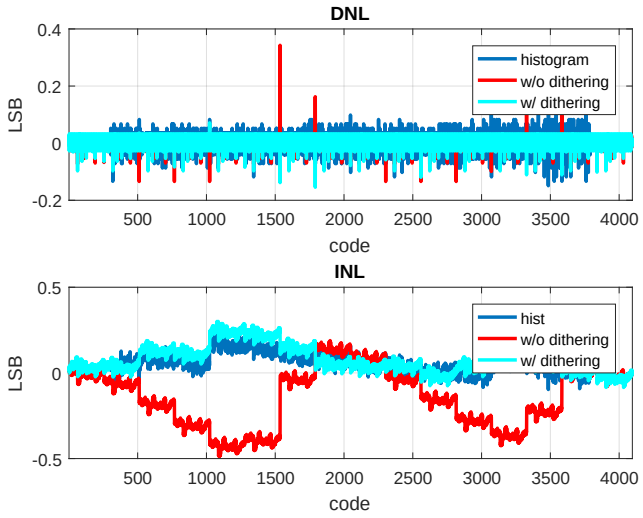

Figure 5.21 INL comparison before and after adding dithering (measurement)

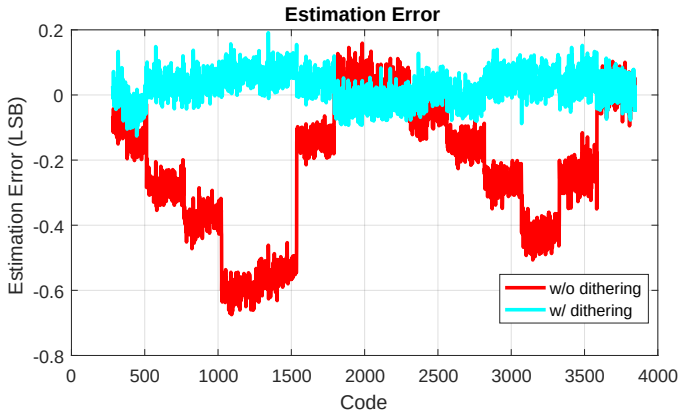

Figure 5.22 Estimation Error Comparison

5.6 Conclusion

The uSMILE algorithm was developed for accurate ADC linearity test with significantly reduced data acquisition time. However, in a low noise testing environment, the quantization error causes up to +/- 0.5 LSB INL estimation error. In this chapter, we analyze the root cause of the quantization error-induced estimation error. To overcome this

www.manaraa.com

issue, different dithering methods including Gaussian, uniform, fixed-pattern dithering were compared and evaluated. It has been shown that proper dithering can significantly improve the estimation accuracy. The fixed-pattern dithering method was proven to the most-efficient and cost-effective method. The proposed methods were validated by both simulation and measurement in hardware setup. The simulation and measurement results show that the INL estimation error can be reduced to 0.2 LSB with fixed-pattern dithering. Therefore, with the proposed fixed pattern dithering, the uSMILE algorithm can be effectively used in low-resolution ADC with reduced number of sampling, thus saving the test cost. Considering the high volume in around 12-bit ADCs, the cost reduction becomes significant.

www.manaraa.com

REFERENCE

[1] International Technology Roadmap for Semiconductors, 2011 edition, [Online]. Available: http://public.itrs.net

[2] J. Doern berg, H.-S. Lee, and D.A. Hodges, "Full Speed Testing of A/D Converters," IEEE J. Solid State Circuits, SC-19, pp. 820-827, December 1984

[3] J. Blair, "Histogram measurement of ADC nonlinearities using sine waves," IEEE Trans. Instrum. Meas., vol. 43, pp. 373383, Jun. 1994.

[4] IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters, IEEE Std. 1241-2010, Jan. 2011.

[5] M. Burns and G. W. Roberts, An Introduction to Mixed-Signal IC Test and Measurement. New York: Oxford Univ. Press, 2000.

[6] T. Kuyel, "Linearity Testing Issues Of Analog To Digital Converters," ITC International Test Conference, pp. 747-756, 1999.

[7] J.L. Huang, C.X. Ong, K.T. Cheng, "A BIST scheme for on-chip ADC and DAC testing," Proc. of the Design, Automation and Test in Europe Conference and Exhibition, 2000, pp 216-220.

[8] L. Jin, et al, "Code-Density Test of Analog-to-Digital Converters Using Single Low-Linearity Stimulus Signal," IEEE Transactions on Instrumentation and Measurement, Vol. 58, No. 8, pp. 2679-2685, August 2009.

www.manaraa.com

[9] L. Jin, K. Parthasarathy, T. Kuyel, D. Chen and R. L. Geiger, "Accurte Testing of Analog-to-Digital Converters Using Low Linearity Signals With Stimulus Error Identification and Removal," IEEE Trans. Instrum Meas., vol. 54, pp. 1188 1199, June 2005.

[10] L. Jin, et al, "SEIR Linearity Testing of Precision A/D Converters in Nonstationary Environments With Center Symmetric Interleaving," IEEE Transactions On Instrumentation And Measurement, Vol. 56, No. 5,pp. 1776-1785, October 2007.

[11] F. Adamo, et al, "FFT Test of A/D Converters to Determine the Integral Nonlinearity," IEEE Trans. On Instrumentation and Measurement, Vol. 51, No. 5, pp. 1050-1054, October 2002.

[12] F. Attivissimo, et al, "INL reconstruction of A/D converters via parametric spectral estimation," IEEE Trans. Instrum. Meas., vol. 53, no. 4, pp. 940946, Aug. 2004.

[13] Z. Yu, D. Chen, R. Geiger, and Y. Papantonopoulos. "Pipeline ADC linearity testing with dramatically reduced data capture time," Proc. IEEE Int. Symposium on Circuits and Systems, pp. 792-795, 2005.

[14] L. Jin, et el, "Linearity Test of A/D Converters Using Kalman Filtering," IEEE International Test Conference, Paper 28.3, Santa Clara, CA, pp. 1-9, Oct. 2006.

[15] B. Vasan, et al, "Linearity Testing of ADCs Using Low Linearity Stimulus and Kalman Filtering," IEEE International Symposium on Circuits and System, pp. 3032-3035, June 2010.

[16] S. Goyal, et al, "Test Time Reduction of Successive Approximation Register A/D Converter By Selective Code Measurement," International Test Conference, Nov, 2005.

www.manaraa.com

[17] Z. Yu and D. Chen, "Algorithm for Dramatically Improved Efficiency in ADC Linearity Test," IEEE International Test Conference (ITC), pages 1-10, 2012.

[18] R. M. Gray and J. G. Stockham, Jr., "Dithered quantizers," IEEE Trans. Inform. Theory, vol. 39, pp. 805812, May 1993.

[19] L. C. Roberts, "PCM television bandwidth reduction using pseudorandom noise," M.S. thesis, Mass. Inst. Technol., Cambridge, Feb. 1961.

[20] W. H. Jeffrey, "Reduction of the subjective effects of quantization noise in PCM speech transmission," M.S. thesis, Univ. Maine, Orono, Aug.1994.

[21] B.N. Suresh Babu and H.B. Wollman, "Testing an ADC linearized with pseudo-random dither," IEEE Transactions On Instrumentation And Measurement, Vol. 47, No. 4,pp. 839-847, AUGUST 1998.

[22] Mahmoud Fawdy Wagdy and Michael Goff, "Linearizing Average Transfer Characteristics of Ideal ADCs via Analog and Digital Dither," IEEE Transactions On Instrumentation And Measurement, Vol. 43, No. 2,pp. 146-150, APRIL 1994.

[24] Lee, D.-U.; Villasenor, J.D.; Luk, W.; Leong, P.H.W., "A hardware Gaussian noise generator using the Box-Muller method and its error analysis," in Computers, IEEE Transactions on , vol.55, no.6, pp.659-671, June 2006

[24] Lee, D.-U.; Luk, W.; Villasenor, J.D.; Guanglie Zhang; Leong, P.H.W., "A hardware Gaussian noise generator using the Wallace method," in Very Large Scale Integration (VLSI) Systems, IEEE Transactions on , vol.13, no.8, pp.911-920, Aug. 2005

[25] Alimohammad, A.; Fard, S.F.; Cockburn, B.F.; Schlegel, C., "A Compact and Accurate Gaussian Variate Generator," in Very Large Scale Integration (VLSI) Systems, IEEE Transactions on , vol.16, no.5, pp.517-527, May 2008

www.manaraa.com

[26] Schuchman, L., "Dither Signals and Their Effect on Quantization Noise," in Communication Technology, IEEE Transactions on , vol.12, no.4, pp.162-165, December 1964

www.manaraa.com

CHAPTER 6. SUMMARY

In this dissertation, jitter decomposition challenges in high-speed links is discussed. Algorithms with a function of accurate and fast jitter decomposition were presented.

As channel loss results in both ISI and jitter amplification, DCD amplification is a big concern in high-speed links. The DCD was well analyzed for both clock channels and data channels in this dissertation. This dissertation presents a general formula to calculate the data channel DCD amplification based on peak distortion analysis and statistical jitter modeling methodology. The presented methodology was validated by time-domain simulation on different lossy channels. The comparison among formula calculation, worst case pattern, and PRBS time-domain simulation in the system demonstrated the accuracy and fast simulation.

In the TIE-based decomposition method, a new time-domain ISI model was applied in the algorithm which is more realistic, accurate, and faster than the conventional ISI models. This method utilized Least Squares (LS) estimation which simultaneously separates ISI, RJ, and PJ. This algorithm obtained the estimated individual jitter component values with fine accuracy by using less data samples compared with the conventional methods. The efficiency and accuracy of the proposed method were demonstrated by simulation and hardware experiments.

In the comparator-based decomposition algorithm, instead of using TIE jitter sequence to decompose, it used Boolean output from a network of simple low-cost comparators to identify the deviation of current sampling position from the ideal sampling position. The new method simultaneously separated ISI, PJ and DCD. Simulation and

www.manaraa.com

measurement results demonstrated that the proposed method can estimate the ISI, PJ and DCD with sufficient accuracy using significantly fewer data samples.

In addition, the low cost and simple dithering method which improved the test of linearity of ADC was proposed. In this thesis, we proposed three types of distribution dithering methods adding to the ramp input signal to reduce the estimation error when uSMILE was applied in low resolution ADCs. Fixed pattern was proved to be the most efficient and cost-effective method by comparing it to the Gaussian, uniform, and fix-pattern distributions. The simulation results indicate that the estimation error could be significantly reduced dithering. Furthermore, a hardware measurement with commercial ADC products was used to validate the effectiveness of the fix-pattern dithering method.Our measurement showed the INL estimation error could be reduced to less than 0.1 LSB. This method could be applied as built-in-self-test (BIST) in the future.

Compared to the existing methods, these algorithms have significant advantages and benefits to reduce the test cost, to improve efficiency and accuracy. First, as the channel loss results in both ISI and jitter amplification, the DCD amplification in data channel was not as well understood and clearly quantized as clock channel DCD amplification. The DCD amplification analysis for data channels in this dissertation provided a fast and accurate mathematical equation to calculate the DCD amplification factor only using post-cursor and pre-cursor coefficient of channel impulse response. The worst-case pattern was a simple and fast pattern to analyze the DCD amplification using fewer data than than PRBS pattern, therefore reducing the simulation time greatly.

Secondly, the new time domain ISI modeling in the TIE-based and comparator-based jitter decomposition had obvious advantages over the conventional ISI models. A conventional ISI model was to model the channel as a first-order and a second-order low-pass filter or based on the convolution technique. However, such a model was too simple to represent the real channel and became invalid because of discontinuities. This model was very time-consuming since the cursor was usually 100-bit long. The ISI jitter

www.manaraa.com

model in this dissertation was simpler than the conventional ISI cursors convolution technique and more accurate and realistic than the low pass filter model which made the simulation fast and reduce the simulation time. This model did not assume any linearity or superposition requirements on the ISI jitter as a function of the previous k-bit, nor does it rely on linearity in the conversion from data waveform voltage errors to timing errors near zero crossing, thus makes the ISI model more robust to channel non-idealities. Both the TIE-based and comparator-based methods can provide ISI analysis in detail for each ISI binary combination.

Lastly, the data samples in the TIE-based and comparator-based method were fewer in number than the state-of-the art requirement. In the TIE-based method, the data samples could be reduced by 50X as shown in the measurement with the comparable accuracy. Currently available jitter measurement techniques require expensive measurement instruments but they do not guarantee sufficient test quality. The jitter analysis algorithms in these instruments usually use by using the histogram method or spectral test. All these algorithms require large samples of TIE jitter data. TIE data must be measured by an instrument with: 1) sufficient bandwidth (three times the data rate is usually adequate) to represent the signal; 2) sufficient memory depth to acquire enough data so that the digital signal processing (DSP) techniques are accurate; 3) low noise. These requirements could be satisfied with high precision circuits such as extremely fast ADC and ideal PLL. Unfortunately, the manufacturing cost for such an instrument is huge and the instrument design also remains a big challenge when the data rate is very high.

In the comparator-based method, no TIE data is needed, meaning that no extremely high precision circuit is necessary which greatly reduces the complexity of circuit design and test cost. The requirements of comparator network in this algorithm were not stringent. The speed was the data rate of I/O rather than the requirement that was 3 times higher than data rate in a real-time testing instrument. The sampling clocks of

www.manaraa.com

the comparator network did not need to be ideal or jitter-free. The jitter existing in sampling clocks helped to generate the different time intervals thus no extra precision circuits were needed to generate the exact time interval. All these relaxed requirements of the comparator network design made the comparator-based method applied in BIST less complex with lower costs, which has a great potential to reduce the whole test cost. However, some state-of-the-art commercial simulation platforms or instruments with different jitter decomposition methods are only applied in off-chip simulation or testing. Instruments are unable to lend themselves to fast parallel testing of devices with a large number of high-speed interfaces due to their hardware complexity, cost and scalability limitations.

The TIE data is still required in the TIE-based method, which means the implementation with the same drawbacks existing in the state-of-the-art decomposition methods. The comparator-based method is a great improvement for the TIE-based method with same accuracy without using TIE data. The comparator-based method requires the circuit design and can be integrated in the receiver which can be automatically adjusted the slicers sampling position and reduce the BER. Our future work is to implement the circuit in the receiver.

www.manaraa.com

APPENDIX

Chatper5 - Proof of Equation 4 Derivation

Let N_{MSB}, N_{ISB} and N_{LSB} be the number of bits in MSB, ISB and LSB bits. Then, the number of segments are $2^{N_{MSB}}$, $2^{N_{ISB}}$ and $2^{N_{LSB}}$ for MSB, ISB and LSB segments respectively. Define E_M as a column matrix of all the error terms of MSB E_M terms. E_I and E_L are defined similarly.

$$E_M = \begin{bmatrix} E_M(0) \\ \vdots \\ E_M(2^{N_{MSB}} - 1) \end{bmatrix} \tag{A.1}$$

Define three matrices H_M,H_I and H_L. H_M is a $k * 2^{N_{LSB}}$ matrix with each term being a boolean value either one or zero. k is the total sample number of input data.Each row represents each sample falling in which MSB error term E_M. If the MSB error term of the j_{th} sample data corresponds to E_{MX}, then the X_{th} column of the H_M is one in j_{th} row. It is the only one 1 in each row and all the others are zeros in corresponding row. H_I and H_L are defined in the same way for ISB and LSB bits.

$$H_M = \begin{bmatrix} C_{MSB}(1) == 0 \; C_{MSB}(1) == 1 \cdots C_{MSB}(1) == 2^{N_{MSB}} - 1 \\ \vdots \; \ddots \; \vdots \\ C_{MSB}(k) == 0 \; C_{MSB}(k) == 1 \cdots C_{MSB}(k) == 2^{N_{MSB}} - 1 \end{bmatrix} \tag{A.2}$$

www.manaraa.com

Then, the estimated INL for the whole sample can be expressed as equation A.3.

$$\left[C_{exp} - C + N_{noise}\right] = \left[H_M \ H_I \ H_L\right] \begin{bmatrix} E_M \\ E_I \\ E_L \end{bmatrix} \tag{A.3}$$

In the $(i+1)_{th}$ column in the matrix, we multiple both sides by the transpose of this column matrix.

$$\begin{aligned} &\left[C_{MSB}(1) == i \cdots C_{MSB}(k) == i\right]\left[C_{exp} - C + noise\right] \\ &= \left[C_{MSB}(1) == i \cdots C_{MSB}(k) == i\right]\left[H_M \ H_I \ H_L\right] \begin{bmatrix} E_M \\ E_I \\ E_L \end{bmatrix} \end{aligned} \tag{A.4}$$

In this matrix $\left[C_{MSB}(1) == i \quad \cdots \quad C_{MSB}(k) == i\right]$, only the location where the corresponding MSB bit being i will be 1s and all the other all 0s. Therefore, equation A.5 can be obtained.

$$\begin{aligned} &\left[\textstyle\sum_{C_{MSB}==i}(C_{exp} - C) + \sum_{C_{MSB}==i}(N_{noise})\right] \\ &= \sum_{C_{MSB}==i} E_M(i) + \sum_{j=0}^{2^{N_{ISB}}-1}\left[\textstyle\sum_{C_{MSB}==i \&\& C_{ISB}==j} E_I(j)\right] \\ &\qquad \sum_{j=0}^{2^{N_{LSB}}-1}\left[\textstyle\sum_{C_{MSB}==i \&\& C_{LSB}==j} E_L(j)\right] \end{aligned} \tag{A.5}$$

If the total number of samples is k, within each MSB segment, the number of samples is approximately equal to $k/2^{N_{MSB}}$. Within each MSB segment, the number of samples for each ISB segment is approximately equal to $k/2^{(N_{MSB}+N_{ISB})}$. Within each MSB segment, the number of samples for each LSB segment is approximately equal to $k/2^{(N_{MSB}+N_{ISB})}$. Then, A.5 can be approximated by A.6.

www.manaraa.com

$$\begin{aligned}\left[\sum_{C_{MSB}==i}(C_{exp}-C)+\sum_{C_{MSB}==i}(N_{noise})\right] \\ \approx \sum_{C_{MSB}==i} E_M(i) \\ +\frac{k}{2^{N_{MSB}+N_{ISB}}}\left[\sum_{j=0}^{N_{ISB}-1}(E_I(j))\right] \\ +\frac{k}{2^{N_{MSB}+N_{ISB}}}\left[\sum_{j=0}^{N_{LSB}-1}(E_L(j))\right]\end{aligned} \tag{A.6}$$

In this equation, $\sum_{j=0}^{N_{ISB}-1}(E_I(j))$ and $sum_{j=0}^{N_{LSB}-1}(E_L(j))$ are close to 0 and their coefficients are also much smaller compared with the number of C_{MSB} equal to i so that the last two terms are almost 0 which can be discarded (A.7).

$$\left[\sum_{C_{MSB}==i}(C_{exp}-C)+\sum_{C_{MSB}==i}(N_{noise})\right] \approx \sum_{C_{MSB}==i}(E_M(i)) \tag{A.7}$$

Divide both sides by the number of MSB bits being $(\#C_{MSB}==i)$.

$$\begin{aligned}E_M(i) \approx \frac{1}{\#C_{MSB}==i}\sum_{C_{MSB}==i}(C_{exp}-C) \\ +\frac{1}{\#C_{MSB}==i}\sum_{C_{MSB}==i}(N_{noise})\end{aligned} \tag{A.8}$$

www.manaraa.com