

A Low-Cost Comparator-Based Method for Accurate Decomposition of Deterministic Jitter in High-Speed Links

Yan Duan ^{ID}, Student Member, IEEE, Hsinho Wu, Member, IEEE, Masashi Shimanouchi ^{ID},
Mike Peng Li, Fellow, IEEE, and Degang Chen ^{ID}, Fellow, IEEE

Abstract—Jitter decomposition is a key tool to identify root causes of jitters in a high-speed digital communication system. It is a huge challenge in balancing the test cost and precision for conventional decomposition methods implemented in instruments where the time interval error (TIE) data are necessary. In this paper, 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 intersymbol 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 or background test.

Index Terms—Comparator network, high-speed data links, jitter decomposition, time interval error (TIE).

I. INTRODUCTION

TO SATISFY the aggressive demand for higher data rate of communication system, the input/output (I/O) speed doubles every two to three years on an average [1]. Using the Optical Internetworking Forum common electrical I/O implementation agreement as an example, its speed has increased from 6.5 Gb/s (Gen1) to 11.3 Gb/s (Gen2) and to 28 Gb/s (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.

Manuscript received January 20, 2018; revised February 28, 2018; accepted March 19, 2018. Date of publication May 2, 2018; date of current version January 20, 2019. (Corresponding author: Yan Duan.)

Y. Duan is with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011 USA (e-mail: yanduan@iastate.edu).

H. Wu is with the Intel Corporations Programmable Solutions Group, San Jose, CA 95134 USA (e-mail: hsinho.wu@intel.com).

M. Shimanouchi and M. P. Li are with the Intel Corporations, San Jose, CA 95134 USA (e-mail: masashi.shimanouchi@intel.com; peng.mike.li@intel.com).

D. Chen is with the Iowa State University, Ames, IA 50011 USA (e-mail: djchen@iastate.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TEMC.2018.2821680

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 the increasing data rate. Currently available jitter measurement techniques require expensive measurement instruments which are limited by the bandwidth.

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 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, shot noise, etc. DJ obeys bounded distributions and can be decomposed into periodic jitter (PJ), data-dependent jitter (DDJ), and bounded uncorrelated jitter (BUJ). PJ is caused by the 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. Nonidealities 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 the frequency-domain-based analysis. The time-domain series of jitter can be analyzed in the frequency domain through the Fourier transform (FT) [3], [4]. In [3], Pang *et al.* use a time lag correlation to decompose jitter components. In [4], Yamaguchi *et al.* separate jitter by performing fast FT on jitter probability distribution function (PDF).

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 TJ 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.

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 time interval error (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 techniques are accurate [15]; and
- 3) low noise;

These requirements can be satisfied with high-precision circuits, and the manufacturing cost for such instrument is very high.

External noise sources can easily affect off-chip measurements especially when the data rate exceeds 1 Gb/s. A lot of on-chip techniques has been developed in recent years as well. However, these on-chip jitter measurement circuits [16]–[18] require a large amount of die area if the jitter histograms have to be collected in real time and the frequency-domain analyses have to be realized; thus, all jitter values are needed. In this case, the clock without jitter are needed as the ideal clock reference that increase the circuit design complexity.

In this paper, instead of using TIE data obtained by instrument, we proposed a low-cost comparator-based jitter decomposition on-chip solution. We mainly focus on describing the algorithm part of the solution in this paper and the circuit design will be discussed in another paper. The decomposition algorithm through a series of digitized 0 s and 1 s from comparator network that is directly related to TIE (called comparator network based method) is developed. The comparator network based method offers several advantages. First, no TIE data are 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 [19], [20] is accurate and more efficient for jitter estimation than the conventional ISI convolution technique [21] 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 [22], [23].

In our proposed algorithm, a comparator network was used to sample the edge of integer periods of pseudorandom binary sequence (PRBS) data and obtain the 0 s and 1 s. Given an initial sampling position that is different from the slicer of receiver, the outputs (0 s and 1 s) of comparators were sent to

the decomposition algorithm based on least squares (LSs) [24], [25]. Then, it obtained the new comparator network sampling position to acquire a new set of output 0 s and 1 s. This iteration process would continue till zero-crossing points were found, where the difference in the number of 0 s and 1 s 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 this paper 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 and Section V concludes the paper.

II. COMPARATOR-BASED DECOMPOSITION 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 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.

A. DJ Modeling

PJ and DCD are modeled by the traditional method [2], [21]. ISI jitter is modeled in time domain.

PJ is a repeating jitter whose frequency is 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)

$$\begin{aligned} \Delta t_{PJ}[n] &= A \sin(2\pi f_0(t - nT) + \phi) \\ &= a \sin\left(\frac{2\pi f_0 n}{f_s}\right) + b \cos\left(\frac{2\pi f_0 n}{f_s}\right) \end{aligned} \quad (1)$$

where $\Delta t_{PJ}[n]$ represents a PJ amount at sampling time nT ; f_0 is the fundamental frequency of PJ; and A is the amplitude of PJ. In a real system, it can be power supply noise or input reference clock of PLL that can be extracted from the data through spectral analysis. In this paper, the PJ from reference clock is considered as an example in simulation; f_s is the frequency of data stream, and ϕ is the phase of PJ. 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 [21]

$$\begin{aligned} \Delta t_{DCD}[n] &= J_{DCD} \times \cos(n\pi) \\ &= [-J_{DCD}, J_{DCD}, -J_{DCD}, J_{DCD}, \dots] \end{aligned} \quad (2)$$

where $\Delta t_{DCD}[n]$ is a DCD at sampling time nT and J_{DCD} is the DCD amplitude that 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

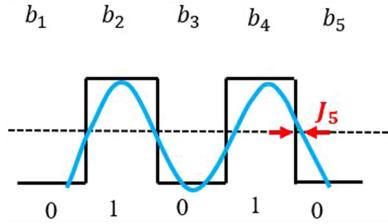


Fig. 1. Example of ISI modeling.

channel, and it occupies the precursor and postcursor samples. The ISI model used here follows that used in [19] and [20] and is more accurate and efficient than conventional modeling based on convolution and first- or second-order low-pass filtering.

In a high-speed link, since some kind of equalizer is typically used to properly equalize to the precursor, the postursors from the previous k -bit is considered in modeling the ISI effect. The selection of the number of postcursor k depends on the amount of loss of the channel and the data rate, and can be determined during channel characterization. The previous k -bit has 2^k binary combinations. Each binary combination generates a different ISI jitter amount on the current bit (main cursor). For example, as shown in Fig. 1, the blue curve is the actual data sequence due to ISI and the black curve is the ideal data sequence for data sequence $b_1 \dots b_5$. b_5 is the current bit, $b_1 b_2 b_3 b_4$ are preceding four bits. $b_1 b_2 b_3 b_4$ has 16 binary combinations. If $b_1 b_2 b_3 b_4$ is 0101, as shown in Fig. 1, the time deviation of actual b_5 edge and the ideal b_5 edge is defined as ISI-induced jitter J_5 . The index 5 is the decimal representation of 0101. ISI jitter model is described in the following equation:

$$\Delta t_{\text{ISI}}[n] = \sum_{l=0}^{2^k-1} (J_l \times C_l[n]) \quad (3)$$

$$C_l = \begin{cases} 1, & \text{if binary to decimal } b_{(n-k)} b_{(n-k+1)} \dots b_{(n-1)} = l \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

where $\Delta t_{\text{ISI}}[n]$ 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 $b_{n-k} b_{n-k+1} \dots b_{n-1}$. J_l is the jitter value of the l th binary combination to the current bit b_n . C_l is a corresponding sign that 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 nonidealities. J_l is the estimation parameter in the algorithm.

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

$$x[n] = d_n \times [\Delta t_{PJ}[n] + \Delta t_{DCD}[n] + \Delta t_{\text{ISI}}[n]] \quad (5)$$

where $x[n]$ is the TIE amount at sampling time nT . Parameters a, b, f_s, f_0, C_l, J_l , and J_{DCD} have been mentioned in the

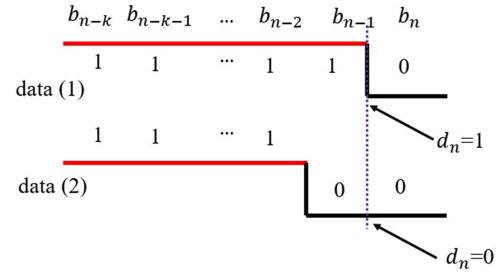
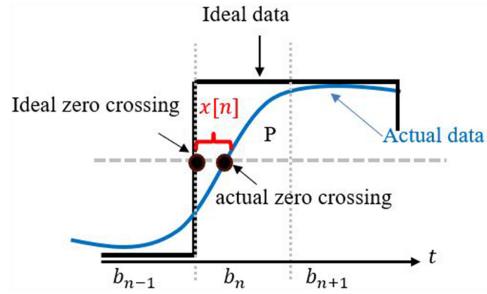
Fig. 2. Data transition sign d_n .

Fig. 3. Definition of TIE.

previous section. d_n is data transition sign used to indicate the existence of 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 Fig. 2. d_n is 0 when there is no data transition from bit $n - 1$ to bit n , as shown in data (2) of Fig. 2. When d_n is 0, no jitter exists for the current bit n .

B. Replacing TIE Sequence 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 Fig. 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.

For simplicity, let us assume that there is no overshooting, undershooting, ringing, and other signal integrity problems in the data edges, i.e., the data edges are monotonic both at rising and falling edges, as shown in Fig. 3. This is not a serious limitation since the proposed method focuses near zero crossing, and monotonicity 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 areas as a function $g(n)$. As seen in Fig. 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., $x[n] \approx 0$, the gray and green areas are approximately the same and $g[n] \approx 0$, as seen in Fig. 4(b). When the actual data

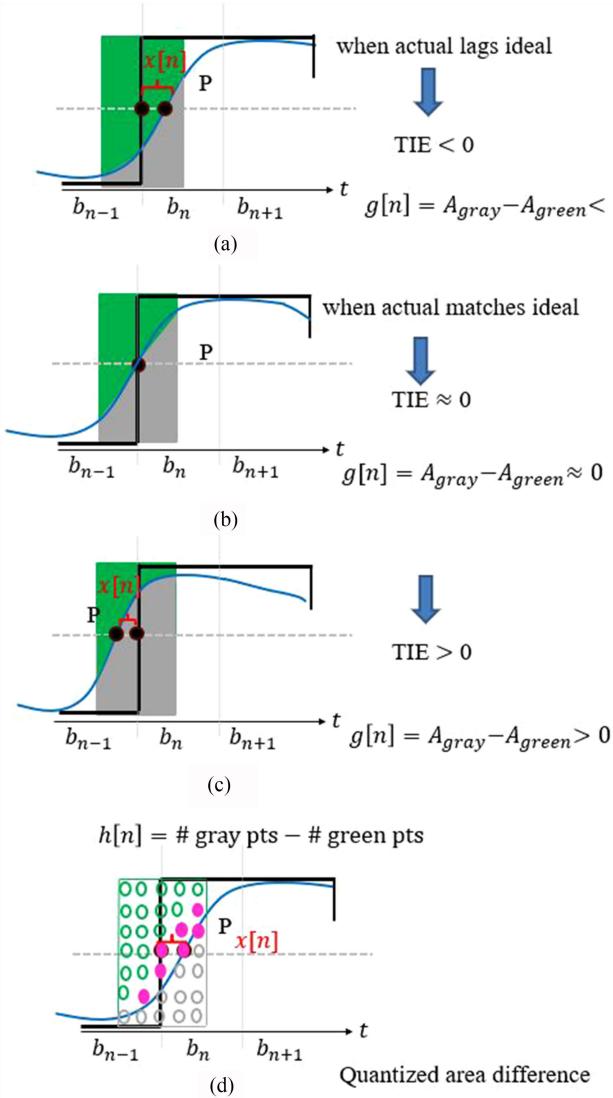


Fig. 4. (a)–(c) Converting finding zero-crossing point to area difference function $g(n)$. (d) Converting finding the area function $g(n)$ to quantized area difference function $h(n)$.

leads the ideal data, i.e., $x[n] > 0$, the gray area will be more than the green area and we have $g[n] > 0$, as seen in Fig. 4(c). Therefore, $g[n]$ is locally proportional to $x[n]$, and could be used as a candidate for replacing the TIE $x[n]$.

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 as

$$\begin{aligned} h[n] &= (\# \text{green dots} - \# \text{gray dots})_n \\ &= \text{quantized}(A_{gray} - A_{green})_n. \end{aligned} \quad (6)$$

This is illustrated in Fig. 4(d).

A simple way to count dots is to place a comparator at each grid point, with both varying threshold voltages and triggering

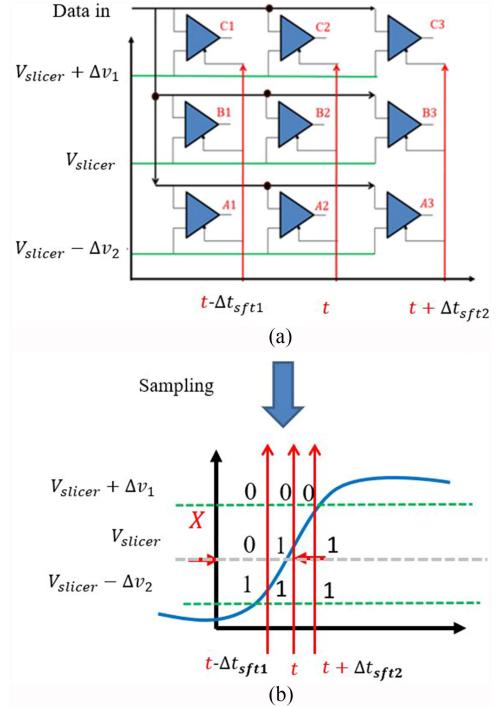


Fig. 5. (a) 3×3 comparator network. (b) Output of comparator network sampling.

times, as shown in Fig. 5(a) ($3 \times 3 A1, \dots, C3$ 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 Δv . 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 Δt . In the Fig. 5(a), there are two voltage intervals (Δv_1 and Δv_2) and two time-delays (Δt_{sft1} and Δt_{sft2}). The Δv and Δt_{sft} do not have to have the same values. The voltage intervals 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. For simplicity, we assume that the voltage intervals Δv between each row are the same and so are the time delay Δt_{sft} between each column. The nine comparators in Fig. 5(b) produced nine Boolean outputs. The number of 0 represents the green area and the number of 1 represents the gray area. The difference of 0 s and 1 s represents the area difference function $h(n)$, as shown in Fig. 5(d), and is given by equation

$$h[n] = (\# 1 - \# 0)_n = \text{quantized}(A_{gray} - A_{green})_n. \quad (7)$$

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

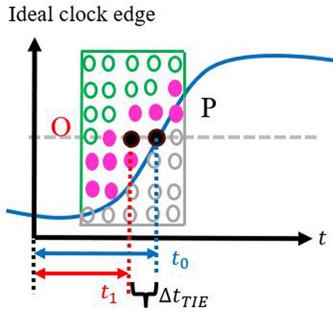


Fig. 6. Relative TIE definition.

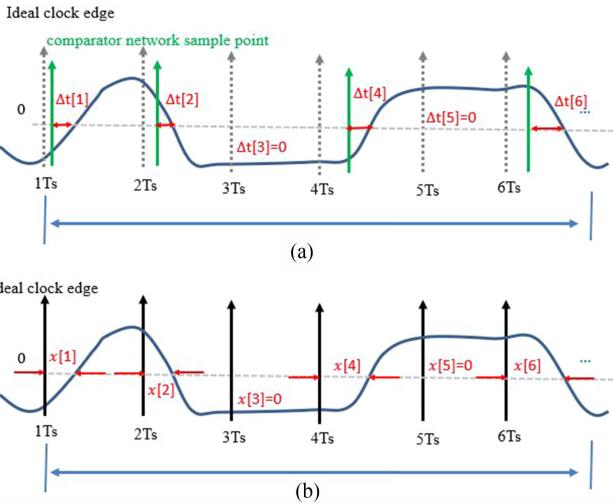


Fig. 7. (a) M-bit relative TIE. (b) M-bit absolute TIE.

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 $\alpha \Delta h[n] \approx x[n]$.

To approximate α , we move the center of the grid point box by a time interval Δt_{TIE} to O from P, as shown in Fig. 6. Some of the comparators output will change from 1 to 0. This will cause a change in the value of the quantized area function $\Delta h(n)$. To a first-order approximation, we will have

$$\Delta t_{\text{TIE}} \approx \alpha \Delta h[n] = \alpha(\Delta \# 0 - \Delta \# 1)_n. \quad (8)$$

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 paper, we measure the S parameter of channel and use a simulation to obtain because 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 Δt_{TIE} represents the deviation of a given reference point and the zero-crossing point. For example, every edge in a data stream shown in Fig. 7(a) 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 ($\Delta t_{\text{TIE}}[1], \Delta t_{\text{TIE}}[2], \dots, \Delta t_{\text{TIE}}[M]$) of M bit data stream shown

in Fig. 7(a) can be expressed by a series difference of 1 and 0 in each sampling time $1T, 2T, \dots, MT$, respectively, which is given in the following:

$$\begin{bmatrix} \Delta t_{\text{TIE}}[1] \\ \Delta t_{\text{TIE}}[2] \\ \vdots \\ \Delta t_{\text{TIE}}[M] \end{bmatrix} = - \begin{bmatrix} \alpha_1(\Delta \# 0 - \Delta \# 1)[1] \\ \alpha_2(\Delta \# 0 - \Delta \# 1)[2] \\ \vdots \\ \alpha_M(\Delta \# 0 - \Delta \# 1)[M] \end{bmatrix}. \quad (9)$$

C. Jitter Decomposition by LSs

Using LS can estimate the PJ and ISI in the proposed model if absolute TIE is known, which was proved by our previous works [19], [20]. Equation (6) shows that TIE is a linear equation. For a linear time invariant system, LS estimation overcomes the convergence problem [24], [25] 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 $[a, b, J_{\text{DCD}}, J_0, J_1, \dots, J_{(2^k-1)}]$.

Define that M bits absolute TIE sequence is shown in Fig. 7(b). The absolute TIE in each bit is $x[1], x[2], \dots, x[M]$ taken at sampling time $1T, 2T, \dots, MT$, respectively. Then the absolute TIE sequence Z_M can be expressed by the following:

$$Z_M = \begin{bmatrix} x(1) \\ x(2) \\ \vdots \\ x(M) \end{bmatrix} = H_M \theta + V_M \quad (10)$$

where V_M is RJ vector (denoted as $[[1], [2], \dots, [M]]^T$) for the TIE sequence, θ representing $[a, b, J_{\text{DCD}}, J_0, J_1, \dots, J_{(2^k-1)}]^T$ 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

$$H_M = [A \ B \ C]$$

$$A = \begin{bmatrix} d_1 \times \sin\left(\frac{2\pi f_0}{f_s}\right) & d_1 \times \cos\left(\frac{2\pi f_0}{f_s}\right) \\ \vdots & \vdots \\ d_M \times \sin\left(\frac{2\pi f_0 M}{f_s}\right) & d_M \times \cos\left(\frac{2\pi f_0 M}{f_s}\right) \end{bmatrix}$$

$$B = \begin{bmatrix} d_1 \times \cos(\pi) \\ d_2 \times \cos(\pi) \\ \vdots \\ d_M \times \cos(\pi) \end{bmatrix} \quad C = \begin{bmatrix} d_1 \times C_{01} & \cdots & d_1 \times C_{2^k-11} \\ d_2 \times C_{02} & \cdots & d_1 \times C_{2^k-12} \\ \vdots & \ddots & \vdots \\ d_M \times C_{0M} & \cdots & d_M \times C_{2^k-1M} \end{bmatrix} \quad (11)$$

In (11), C_{li} can be extracted from the data stream and store in lookup table. For instance, if binary combination of postcursor of the i th 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 that the f_0 is a known parameter in this paper. f_s is the data rate. The solution $\hat{\theta}$ for length of absolute TIE M bits is

$$\hat{\theta} = [H_M^T H_M]^{-1} H_M^T Z_M. \quad (12)$$

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

$$\hat{\theta} = [\hat{a}, \hat{b}, \hat{J}_{DCD}, \hat{J}_0, \hat{J}_1, \dots, \hat{J}_{2^k-1}]. \quad (13)$$

The amplitude of estimation PJ is

$$\hat{A} = \sqrt{\hat{a}^2 + \hat{b}^2}. \quad (14)$$

The ISI is calculated by

$$\begin{aligned} \widehat{ISI}_{pk-pk} &= \max(\hat{J}_0, \hat{J}_1, \dots, \hat{J}_{2^k-1}) \\ &- \min(\hat{J}_0, \hat{J}_1, \dots, \hat{J}_{2^k-1}). \end{aligned} \quad (15)$$

Since we are unable to know the ideal clock edge as a reference point and absolute TIE data are 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 $[a_{old}, b_{old}, J_{DCDold}, J_{0old}, J_{1old}, \dots, J_{(2^k-1)old}]$.

As mentioned in part B of this section, the difference of 0 and 1 of comparator output is approximately proportional to relative TIE Δt_{TIE} of position O and zero-crossing point P shown in (9). Therefore, the M bits relative TIE sequence can be expressed as the difference of 1 and 0 by the following:

$$\begin{aligned} \Delta T[M] &= \begin{bmatrix} \Delta t_{TIE}[1] \\ \Delta t_{TIE}[2] \\ \vdots \\ \Delta t_{TIE}[M] \end{bmatrix} \approx -\alpha \begin{bmatrix} (\Delta \#0 - \Delta \#1)[1] \\ (\Delta \#0 - \Delta \#1)[2] \\ \vdots \\ (\Delta \#0 - \Delta \#1)[M] \end{bmatrix} \\ &= H_M \Delta \theta + V_M. \end{aligned} \quad (16)$$

$\Delta \theta$ represents estimate parameters $[\Delta a, \Delta b, \Delta J_{DCD}, \Delta J_0, \Delta J_1, \dots, \Delta J_{2^k-1}]$ and we combine the (12) and (13), the estimated $\Delta \theta$ is obtained as

$$\hat{\Delta \theta} = -[H_M^T H_M]^{-1} \alpha [\#0 - \#1]_M \quad (17)$$

$$\hat{\theta}_{new} = \theta_{old} + \hat{\Delta \theta} \quad (18)$$

where $\hat{\theta}_{new}$ is the estimate jitter parameters $[a, b, J_{DCD}, J_0, J_1, \dots, J_{(2^k-1)}]$.

D. Block Iteration

In reality, we performed the estimation many times since one time estimation is not accurate. 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 Fig. 8). The incoming PRBS data stream with PJ, DCD, and ISI jitter was divided into several blocks. Each block had the same integer periods of PRBS data (M bits) and should cover at least 1 period PJ.

First, in block 1, given an initial PJ, DCD, and ISI jitter θ_0 and calculated the initial sample position S_0 (shown in Fig. 8) based on (12), 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 (17) and (18), the estimation distance of the initial sample position S_0 to zero-crossing point $\Delta \theta_0$ can be obtained by the

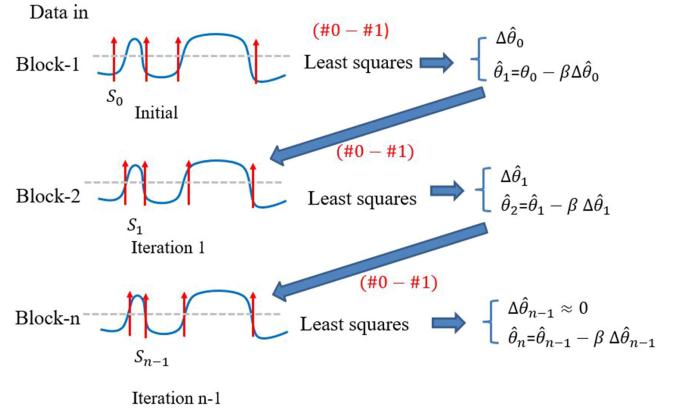


Fig. 8. Whole iteration process.

following equation:

$$\hat{\Delta \theta}_0 = [H_M^T H_M]^{-1} \alpha [\#0 - \#1]_M. \quad (19)$$

Then, the new jitter estimated $\hat{\theta}_1$ can be obtained based on the previous block θ_0 and $\hat{\Delta \theta}_0$ in the following equation:

$$\hat{\theta}_1 = \theta_0 + \beta \hat{\Delta \theta}_0. \quad (20)$$

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

Second, in block 2, the new sampling position 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 estimate θ_1 and $\Delta \theta_1$.

This iteration continues to different blocks until $\#1 - \#0$ approaches zero, which is the zero-crossing point. The final estimation $\hat{\theta}_n$ contains the PJ, DCD, and ISI information. The PJ is obtained by (14) and ISI peak-to-peak (pk-pk) is calculated by (15).

The flow chart of the proposed method is given in Fig. 9. First, given initial jitter θ_{old} for the data block 1 and calculated the comparator network sampling instance based on (12), every data block length is M bits. Second, modeled the ISI matrix based on data pattern and obtained the postcursor 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 for each edge and converted relative Δt_{TIE} sequence. Fourth, used the LS to the relative Δt_{TIE} sequence and obtained the estimated $\Delta \theta$ and $\hat{\theta}_{new}$ based on (17) and (18).

If $\hat{\Delta \theta}$ is less than a threshold value ξ , calculated the final PJ, DCD, and ISI estimation based on (14) and (15). Otherwise, the $\hat{\theta}_{new}$ is considered as the θ_{old} of the next block iteration until the $\hat{\Delta \theta}$ less than a threshold value ξ .

E. Algorithm Realization Analysis

1) *Consideration of comparator network design:* The requirements of comparator network in this algorithm are not stringent. First, the comparator speed is the data rate of I/O rather than the requirement that three times higher than data rate in a real-time testing instrument. Second, the sampling clocks $t - \Delta t_{sft1}$, t , and $t + \Delta t_{sft1}$ shown in Fig. 5(a) are not required to be ideal or

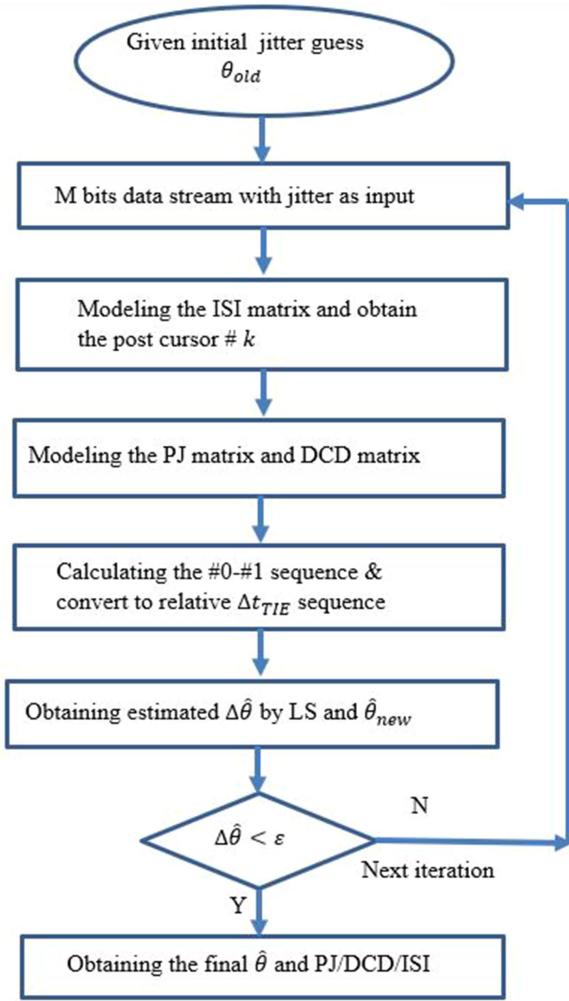


Fig. 9. Flow chart of the proposed algorithm.

jitter-free. Third, the time interval Δt_{sf1} and Δt_{sf2} are not required to be equal. The voltage intervals Δv_1 and Δv_2 between rows are not required to be equal.

However, there are some challenges that might cause potential impacts to the receiver after adding the comparator network to the receiver, such as if the total input capacitance of comparators network will affect the signal under test and cause the jitter characteristic or not, the offset, and others. The detailed proof of how time interval Δt and voltage interval Δv requirements are relaxed and other design issues and consideration will be discussed in another paper.

2) *The data length M bit in each iteration selection:* The data length M should include the integer periods of PJ at least one period and covers integer 2^k bits.

3) *The data pattern selection:* PRBS-n pattern is commonly used in the jitter test. In order to obtain each accurate ISI code jitter, the data pattern 2^n should cover at least two times of 2^k . For example, the n should be larger than 10 if the k is 10.

III. SIMULATION

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

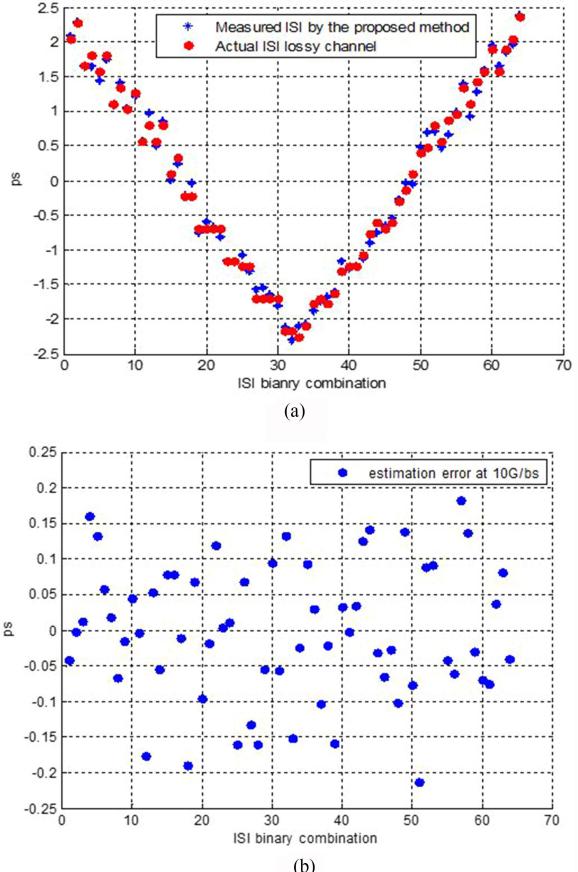


Fig. 10. (a) Comparison of ISI TIE binary combination and estimation result at 10 Gb/s. (b) Estimation error.

analysis and the data rates are 10 and 25 Gb/s in the simulation, respectively. The PJ was a sine wave with 100-MHz frequency at 10 Gb/s and 250-MHz frequency at 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 10 Gb/s and $\mu = 1$ ps, $\sigma = 0.5$ ps in 25 Gb/s, respectively. The vertical voltage interval μ is 10% supply voltage, $\sigma = 20$ mV for 10 and 25 Gb/s if supply voltage is 1 V, α was roughly guessed as 0.3 in 10 Gb/s and 0.5 in 25 Gb/s based on the observation of data edge waveform and $\beta = 1$.

A. 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 3.5 dB at 5 GHz and 7.5 dB at 12.5 GHz was used to generate the ISI TIE sequence. We classified the ISI jitter sequence to 2^k binary combinations. The postcursor number k of the transmission line is 6, which was obtained from the channel pulse response. These 6 bits postursors have 64 binary combinations from 000000 to 111111. The corresponding jitter amount are from J_0 to J_{63} . The red dots in Figs. 10(a) and 11(a) represent the TIE ISI jitter in each ISI binary combination, which shows that different binary combinations correspond to different ISI jitter at 10 and 25 Gb/s, respectively. The eye diagram of PRBS 7 only with ISI in Fig. 12

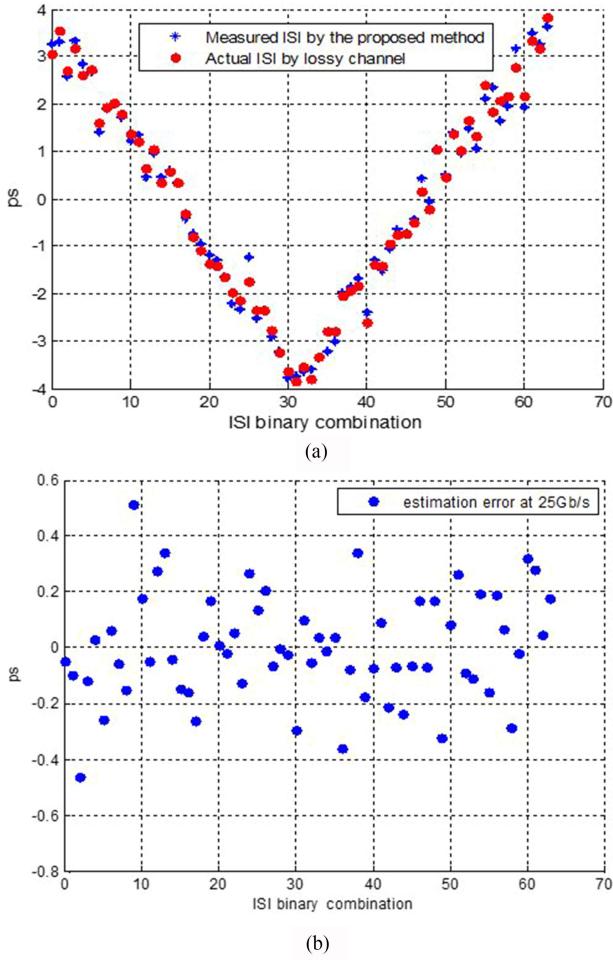


Fig. 11. (a) Comparison of ISI TIE binary combination and estimation result at 25 Gb/s. (B) Estimation error.

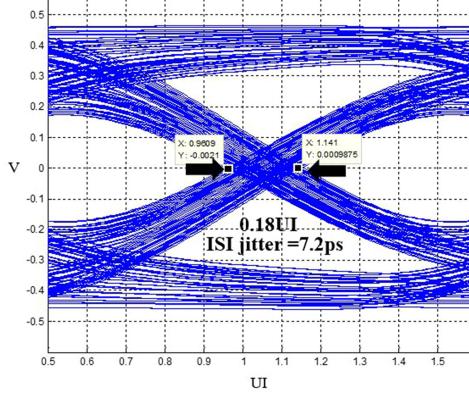


Fig. 12. Eye diagram of PRBS-7 only with ISI at 25 Gb/s.

shows ISI jitter (pk-pk) in this transmission line is about 7.2 ps at 25 Gb/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 Fig. 10(a) for 10 Gb/s and in Fig. 11(a) for 25 Gb/s. Fig. 11(a) and (b) shows the error between the actual ISI and estimated ISI at 10 and 25 Gb/s, respectively.

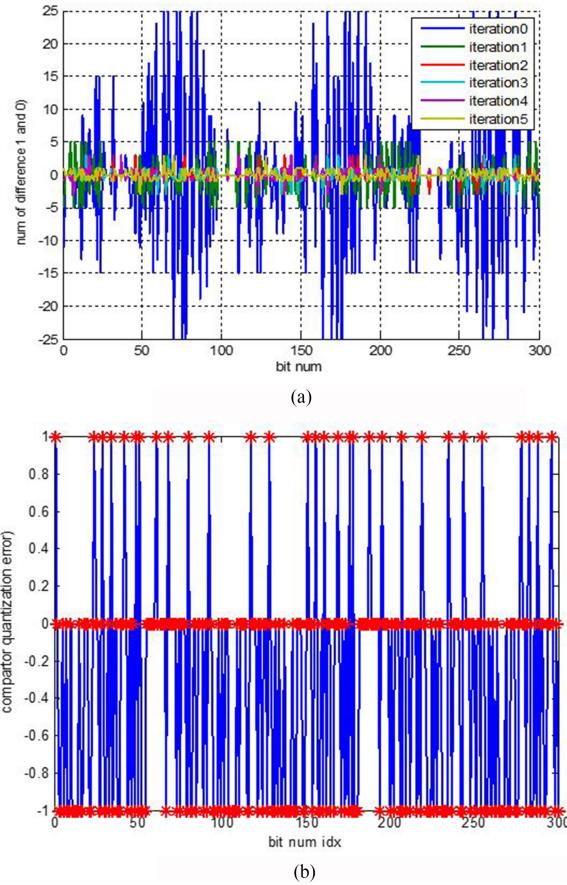


Fig. 13. Validation of convergence of the algorithm. (a) Iteration process at 25 Gb/s. (b) Last iteration result.

They show that the estimated ISI and actual ISI are very close both at 10 and 25 Gb/s. The estimated ISI (pk-pk) value is about 4.6 ps at 10 Gb/s and 7.2 ps at 25 Gb/s, which is very close to the pk-pk jitter obtained from the eye diagram in Fig. 12.

We also used the feature selective validation (FSV) method [26], [27] to compare the ISI TIE jitter sequence and estimated ISI jitter at 10 Gb/s. Two quality factors GRADE and SPREAD for amplitude difference measure are 3, which means the result is good in FSV. The GRADE and SPREAD for feature difference measure are 4, which shows fair. The GRADE for global difference measure (GDM) is 4, representing fair, and the SPREAD for GDM is 3, meaning good.

B. Validation of Convergence of the Algorithm

Given an initial PJ (10 ps), ISI (0 ps for each binary combination), and DCD (0 ps) at 25 Gb/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 LSs. In each recursive iteration, the LS fitting error can be computed. Fig. 13(a) 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 Fig. 13(b). It is clear that the errors have been reduced to within 1, which is the quantization error of the

TABLE I
COMPARISON OF THE PROPOSED METHOD AND JNEYE AT 10 Gb/S (UNIT: PS)

Channel	Added jitter			JNEYe (state-of-the art)			The proposed method				
	DCD	PJ	ISI	Sample data (bit)	DCD	PJ	ISI	Sample data(bit)	DCD	PJ	ISI
Channel A	4	15	unknown	8k	4.3	15.5	4.4	1.27k	4.5	15.3	4.6
	10	10	unknown	8k	10.6	10.5	4.4	1.27k	10.7	10.6	4.7
	8	25	unknown	8k	8.2	25.3	4.4	1.27k	8.6	25.2	4.6
Channel B	0	0	unknown	8k	0	0	3.5	1.27k	0	0	3.6
Channel C	0	0	unknown	8k	0	0	9.2	1.27k	0	0	9.3

TABLE II
COMPARISON OF THE PROPOSED METHOD AND JNEYE FOR CHANNEL A AT 25 Gb/S (UNIT: PS)

Added jitter			JNEYe (state-of-the art)			The proposed method				
DCD	PJ	ISI	Sample data (bit)	DCD	PJ	ISI	Sample data(bit)	DCD	PJ	ISI
0	0	unknown	8k	0	0	7.58	1.27k	0.02	0.07	7.28
0	8	unknown	8k	0	8.36	7.58	1.27k	0	7.99	7.65
2	1.5	unknown	8k	2.2	1.56	7.58	1.27k	2.34	1.63	7.61
4	4	unknown	8k	4	4.48	7.58	1.27k	4.06	3.75	7.24

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.

C. Validation of PJ, DCD, and ISI Jitter Estimation

In the 10 Gb/s simulation, PRBS-7 data stream with different PJ (10-, 15-, and 25-ps pk-pk values), DCD (4-, 8-, and 10-ps pk-pk values), and ISI jitter caused by channel A (4.6-ps pk-pk values) was sent to the algorithm. The simulation results are summarized in the third column group of Table I. It shows that the estimated jitter is very close to the added jitter.

In the 25-Gb/s simulation, PRBS-7 data stream with different jitter component was generated. The data stream with different PJ (0-, 1.5-, 4-, and 8-ps pk-pk values), DCD (0-, 2-, and 4-ps pk-pk values), and ISI jitter caused by channel A (7.2-ps pk-pk values) was sent to the algorithm. The simulation results are summarized in Table II. 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 ones caused by the jitter amplification due to channel loss.

IV. MEASUREMENT RESULT

To verify the ISI (pk-pk) estimation and the accuracy and efficiency of the algorithm, a hardware test bench (shown in Fig. 14) was used to measure the jitter components for 10 Gb/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 the simulation part A was used to generate the ISI jitter. All experiments were done at a data rate of 10 Gb/s.

We also used the result of JNEYe with conventional decomposition algorithms as a 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

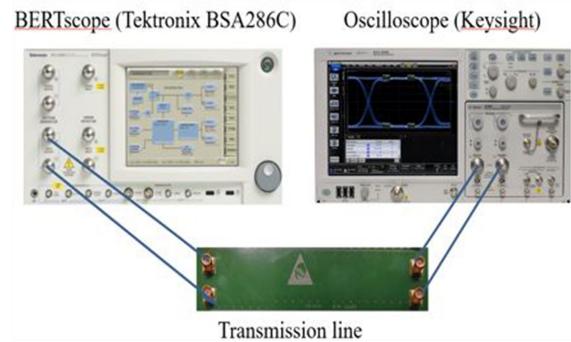


Fig. 14. Experimental test bench of 10 Gb/s.

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 [28]. All tests for comparison were done at 10 and 25 Gb/s.

A. 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 III. 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 4.4 ps. The one in the proposed method is 4.6 ps. All values are very close. Therefore,

TABLE III
COMPARISON OF THE PROPOSED METHOD/INSTRUMENT/JNEYE FOR CHANNEL A AT 10 Gb/S (UNIT:PS)

Added jitter			Oscilloscope(state-of-the art) (sample data: 8k bits)			JNEYe (state-of-the art) (sample data: 8k bits)			The proposed method (sample data: 8k bits)		
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	4.6
0	20	0	0.6	20.5	3.2	0	20.1	0	0	20.2	0
0	20	unknown	0.8	20.3	6.5	0	20	4.4	0	20.1	4.58

the ISI modeling is validated. DCD in the simulation is 0 ps, whereas the one in the measurement is 0.5 ps due to the instrument noise. PJ in the proposed method is 0 ps (pk-pk) while the oscilloscope result is about 1.4 ps.

B. Comparison of the Accuracy and Sample Data

In the hardware experiment, PRBS-7 data stream at 10 Gb/s with different PJ (0 and 20 ps 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 was sent to the proposed method and JNEYe.

Table III 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, whereas in the proposed algorithm the estimation is about 0 ps. The DCD measurement result is 0.5 ps, whereas 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.

We also modeled channel B with 3-dB loss and channel C with 5-dB loss at 5 GHz 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 I. It shows that both of two methods have the same estimation accuracy with the same sample data at 10 Gb/s.

For the 25 Gb/s comparison, PRBS7 data stream with different PJ (1.5, 4, and 8 ps pk-pk), DCD (0, 2, and 4 ps), and ISI jitter generated from channel A was sent to JNEYe and the proposed method.

Table II 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.

V. CONCLUSION

An efficient and accurate comparator-based method is presented that simultaneously extracts PJ, DCD, 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 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. However, there are still limitations in the proposed method. The proposed method cannot decompose the BUJ when the crossing-talk existing and the RJ estimation is not accurate. 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. Some limitations in comparator network design should be considered in real-circuit implementation.

ACKNOWLEDGMENT

The authors would like to thank Intel for providing JNEYe hardware validation in this paper.

REFERENCES

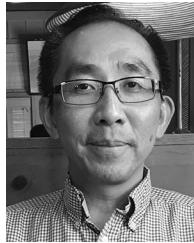
- [1] ITRS roadmap, ITRS, 2011. [Online]. Available: www.itrs.net/Links/2011ITRS/Home2011.htm
- [2] M. P. Li, *Jitter, Noise and Signal Integrity at High-Speed*. London, U.K.: Pearson Education, 2007
- [3] H. Pang, J. Zhu, and W. Huang, "Jitter decomposition by fast Fourier transform and time lag correlation," in *Proc. IEEE Int. Conf. Commun., Circuits Syst.*, Jul. 2009, pp. 365–368.
- [4] T. Yamaguchi *et al.*, "An FFT-based jitter separation method for high-frequency jitter testing with a 10x reduction in test time," in *Proc. IEEE Int. Test Conf.*, Oct. 2007, pp. 1–8.
- [5] M. Li, J. Wilstrup, R. Jessen, and D. Petrich, "A new method for jitter decomposition through its distribution tail fitting," in *Proc. IEEE Int. Test Conf.*, 1999, pp. 788–794.
- [6] G. Hnsel, K. Stieglbauer, G. Schulze, and J. Moreira, "Implementation of an economic jitter compliance test for a multi-gigabit device on ATE," in *Proc. IEEE Int. Test Conf.*, Oct. 2004, pp. 1303–1312.
- [7] D. Hong and K.-T. Cheng, "An accurate jitter estimation technique for efficient high speed I/O testing," in *Proc. IEEE Asian Test Symp.*, Oct. 2007, pp. 224–229.
- [8] J.-L. Huang, "A random jitter extraction technique in the presence of sinusoidal jitter," in *Proc. IEEE Asian Test Symp.*, Nov. 2006, pp. 318–326.
- [9] R. Stephens, *Separation of Random and Deterministic Components of Jitter*, U.S. Patent 71 49 638, Dec. 12, 2006.

- [10] R. Stephens, *Separation of a Random Component of Jitter and a Deterministic Component of Jitter*, U.S. Patent 71 91 080, Mar. 13, 2007.
- [11] S. Wisethanichkij and K. Dejhan, "Jitter decomposition by derivatived gaussian wavelet transform," in *Proc. IEEE Int. Symp. Commun. Inf. Technol.*, Oct. 2004, vol. 2, pp. 1160–1165.
- [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. Electron. Testing, Theory Appl.*, vol. 25, no. 6, pp. 337–342, Dec. 2009.
- [13] Q. Dou and J. Abraham, "Jitter decomposition by time lag correlation," in *Proc. IEEE Int. Symp. Qual. Electron. Des.*, Mar. 2006, pp. 1–6.
- [14] J. Zhu and W. Huang, "Jitter analysis and decomposition based on EMD/HHT in high-speed serial communications," in *Proc. IEEE Int. Conf. Testing Diagnosis*, Apr. 2009, pp. 1–4.
- [15] Agilent Technologies Application Note, "Analyzing jitter using agilent EZJIT plus software," Literature Number AN-5989- 3776EN, 2005
- [16] 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. Very Large Scale Integr. Syst.*, vol. 19, no. 8, pp. 1325–1335, Aug. 2011.
- [17] 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, p. 1278–1285, May 2008.
- [18] 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. Very Large Scale Integr. Syst.*, vol. 17, no. 12, pp. 1698–1708, Dec. 2009.
- [19] Y. Duan and D. J. Chen, "Accurate jitter decomposition in high speed link," in *Proc. IEEE Int. Symp. Very Large Scale Integr. Test Symp.*, Apr. 2017, pp. 1–6.
- [20] Y. Duan and D. J. Chen, "Fast and accurate jitter decomposition of deterministic jitter components in high speed link," *IEEE Trans. Electromagn. Compat.*, pp. 1–9, to be published.
- [21] D. Oh and X. Yuan, *High Speed Signaling: Jitter Modeling Analysis, and Budgeting*, Englewood Cliffs, NJ, USA: Prentice-Hall, 2011, Ch. 3, pp. 44–64.
- [22] V. K. Sharma, J. N. Tripathi, R. Nagpal, S. Deb, and R. Malik, "A comparative analysis of jitter estimation techniques," in *Proc. IEEE Int. Conf. Electron., Commun. Comput. Eng.*, 2014, pp. 125–130.
- [23] J. Buckwalter, B. Analui, and A. Hajimiri, "Predicting data dependent jitter," *IEEE Trans. Circuits Syst. II*, vol. 51, no. 9, pp. 453–457, Sep. 2004.
- [24] H. Liu and Z. He, "A sliding-exponential window RLS adaptive algorithm: Properties and applications," *Signal Process*, vol. 45, pp. 357–368, 1995.
- [25] J. M. Cioffi and T. Kailath, "Fast recursive least squares transversal filters for adaptive filtering," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-32, no. 2, pp. 304–337, Apr. 1984
- [26] *Standard for Validation of Computational Electromagnetics Computer Modeling and Simulation—Part 1*, IEEE Standard P1597, Feb. 2008.
- [27] A. Orlandi, A. P. Duffy, B. Archambeault, G. Antonini, D. E. Coleby, and S. Connor, "Feature selective validation (FSV) for validation of computational electromagnetics (CEM). Part II assessment of FSV performance," *IEEE Trans. Electromagn. Compat.*, vol. 48, no. 3, pp. 460–467, Aug. 2006.
- [28] M. Li, M. Shimanouchi, and H. Wu, "Advancements in high-speed link modeling and simulation," in *Proc. IEEE Custom Integr. Circuits Conf.*, 2013, pp. 1–8.

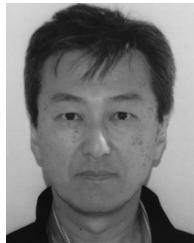


Yan Duan (S'13) received the B.S. and M.S. degrees in electrical engineering from Hunan University, Changsha, China, and the Ph.D. degree from the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, USA in 2017.

From 2006 to 2012, she was a Senior Hardware Engineer with Huawei, Shenzhen, China, a Serdes Architect Intern with Altera, San Jose, CA, USA, in 2014, and an Analog Circuit Design Intern with Huawei, Santa Clara, CA, USA, in 2015. She is currently an Analog Circuit Engineer with Intel, San Jose, CA, USA. Her research interests include ADC design and jitter analysis in high-speed link.



Hsinho Wu (M'11) received the Ph.D. degree in electrical engineering from the University of Florida, in 1999. He is a Design Engineer with Intel Corporations Programmable Solutions Group (formerly Altera), San Jose, CA, USA. He currently works on high-speed communication systems of FPGA products. His development and research interests include signal integrity, clock and data recovery, equalizations, device and system modeling, simulation techniques, and software architecture.



Masashi Shimanouchi received the B.S. degree in physics from the Keio University, Japan, in 1986. He is a Design Engineer with Intel Corporation, Santa Clara, CA, USA. His work on high-speed serial links of FPGA products includes link system and component architecture, modeling, characterization, and link jitter and BER simulation tools development with expertise in measurement metrology, signal processing, signal integrity, and jitter area.



Peng (Mike) Li (F'12) received the B.S. degree in space physics from the University of Science and Technology of China, Hefei, China, in 1985, the Ph.D. degree in physics, the M.S.E. degree in electrical and computer engineering, and the M.S. degree in physics from the University of Alabama, Huntsville, AL, USA, in 1991, 1991, and 1987, respectively.

He is an Intel Fellow and the Technologist for high-speed I/O and interconnects with Intel Corporation, Santa Clara, CA, USA. He serves as the Corporate Technical Expert and Adviser in high-speed link

technology; standards; SerDes architecture; electrical and optical signaling and interconnects; silicon photonics integration; optical field-programmable gate arrays; and high-speed simulation, debug, and test for jitter, noise, signaling, and power integrity. He was the Chief Technology Officer for Wavecrest Corporation from 2000 to 2007. He is an Affiliated Professor with the Department of Electrical Engineering, University of Washington, Seattle, WA, USA. He was a Postdoctoral and then a Research Scientist of high-energy astrophysics with Space Sciences Laboratory, University of California, Berkeley, from 1991 to 1995. He has more than 110 publications on refereed journals and conferences and more than 30 granted patents.



Degang Chen (F'16) received the B.S. degree in instrumentation and automation from Tsinghua University, Beijing, China, in 1984, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of California, Santa Barbara, CA, USA, in 1988 and 1992, respectively.

From 1984 to 1986, he was with the Beijing Institute of Control Engineering, a space industry R/D institute. From March to August 1992, he was the John R. Pierce Instructor of electrical engineering with the California Institute of Technology. After that, he

joined Iowa State University, Ames, IA, USA, where he is currently an Associate Professor. He was with the Boeing Company in summer 1999 and was with Dallas Semiconductor-Maxim in summer 2001. His research experience includes particulate contamination in microelectronic processing systems, vacuum robotics in microelectronics, adaptive and nonlinear control of electromechanical systems, and dynamics and control of atomic force microscopes. His current teaching and research interests include the area of analog and mixed-signal VLSI-integrated circuit design and testing. In particular, he is interested in low-cost high-accuracy test and built-in-self-test of analog and mixed-signal and RF circuits, and in self-calibration and adaptive reconfiguration/repair strategies for enhancing the performance and yield of such circuits.

Dr. Chen is the recipient of the Best Paper Award at the 1990 IEEE Conference on Decision and Control and the Best Transaction Paper Award from the ASME Journal of Dynamic Systems, Measurement, and Control, in 1995. He became an A.D. Welliver Faculty Fellow with the Boeing Company in 1999.