

Periodic Jitter Injection with Direct Time Synthesis by SPP™ ATE for SerDes Jitter Tolerance Test in Production

Masashi Shimanouchi
NPTest, a Schlumberger Company
150 Baytech Drive, San Jose, CA 95134-2302
mshimanouchi@nptest.com

Abstract

Controlled amount of jitter injection into high speed serial bit stream is required for SerDes jitter tolerance test. While jitter injection by Direct Time Synthesis can be a much more cost effective method than a combination of several instruments, it is not widely used yet. Considering its high potential in high volume production test cost reduction, we have studied the basics of the jitter injection flexibility and accuracy of our test system, and the high end ATE seems to be acceptably accurate for high volume production test of multi-gigabit SerDes. The results of our studies provide a foundation for further exploitation of jitter injection using Direct Time Synthesis method.

1 Introduction

In order to address the rapidly growing needs for testing high speed serial data link devices and circuit blocks in large logic ICs whether for telecom or datacom, several test methods have been proposed in the past few years [1]-[6]. While ATE has been the most cost-effective high-volume production test solution for many kinds of ICs, there has been no complete ATE solution yet for such high speed serial devices and circuit blocks mainly due to their extremely high data rate and the jitter test requirement. Currently available ATE seemed to be just suitable to test lower speed devices like SerDes for OC-3 for which ATE's analog test resources such as Sampler and AWG can be utilized for jitter measurement and injection[5].

For jitter measurement at a much higher data rate in a few Gbps range, various instruments such as BERT[1], oscilloscope[1][2] and TIA[6] have been proposed. On the other hand, jitter injection method has been limited to the serial bit stream retimed by a phase-modulated clock[1][2] combined with a DDJ(data dependent jitter) injection filter[1][3] and a random noise source[1], which is a straightforward implementation described in the compliance test methodology documents for some serial link protocol standards[7][8]. Although this jitter injection method is very accurate and, therefore, currently the necessity for each protocol standard compliance test, device characterization and/or design validation, it is not an economical test solution in high volume production especially for high-density backplane SerDes integrated with large CMOS[1].

There is, however, another method to implement controlled amounts of jitter injection, namely DTS (Direct Time Synthesis). This is a method to generate phase changes on serial bit sequence through digital delay calculation rather than analog delay modulation, and is highly flexible in generating jitter[7]. As this is what an ATE usually does, no special HW-used only for jitter injection will be needed, making it a very cost effective solution if it works as intended. Nevertheless, this method has not been exploited much, and the primary reason seems to be twofold: 1)some ATE architectures allow jitter injection only with very simple data pattern such as 1010, 2)ATE's EPA (Edge Placement Accuracy) may be perceived as being not good enough for accurate jitter injection.

Since flexible timing programming with a complex data pattern can be easily achieved by the ATE with SPP (Sequencer Per Pin) architecture[9], the issue 1) can be solved. On the other hand, the high end ATE's EPA is claimed +/-50ps[10], which is 0.25UIpp for 2.5Gbps bit stream, and 0.3125UIpp for 3.125Gbps bit stream. While this timing accuracy specification may not seem good enough at a glance, actual data receiver, will tolerate much larger jitter. For example, a XAUI(3.125Gbps) receiver shall tolerate the sum of DJ (deterministic jitter) and RJ (random jitter) of at least 0.55UIpp[8]. Another example is that an actual deserializer will tolerate much larger PJ (periodic jitter, frequency sweep component) than the standard's jitter tolerance mask as inferred from MAX3880 (2.488Gbps deserializer) data sheet, which shows that the typical tolerable high frequency PJ is more than 0.4UIpp[11]. Having considered these examples, +/-50ps EPA may not be fatally inaccurate for jitter injection in high volume production test if it means an acceptable level of TJ (total jitter) even though it is not accurate enough for protocol standard compliance test or device characterization.

The purpose of this article is to provide a foundation upon which a cost-effective DTS-based jitter injection method for SerDes production test is to be developed. We discuss three different but closely related subjects. In section 2, having reviewed a typical clock and data recovery mechanism, the meaning of PJ mask is studied. In section 3, it is explained how PJ can be injected in a serial data stream using SPP-based ATE. In section 4, we study the jitter injection accuracy of our modern ATE, and discuss the relationship between EPA and TJ. In the end in section 5, the limitations of this method and additional work needed are discussed.

2 Deserializer Jitter Tolerance

The most critical building block of a deserializer is the CDR (clock and data recovery), and jitter tolerance test is documented in each protocol standard with jitter tolerance mask [7][8]. While DJ+RJ mask could be intuitively understood from setup time and hold time concept, PJ mask might not be obvious. Since PJ injection is essential to test clock recovery circuit, the meaning of PJ mask is studied in detail after a brief review of how a typical CDR works and fails to work.

2.1 Clock and Data Recovery

2.1.1 Principle of Operation [12]

The block diagram of a typical CDR is shown in Fig.1. Since timing information is embedded in serial bit stream, a CDR recovers the timing information as a timing clock signal, and the bit stream is re-timed by the recovered clock. When the binary data stream is at r_b Gbps and in NRZ (non return to zero) format, there is no r_b GHz component in the data stream. Therefore the first stage of a CDR is an edge detector, which generates all the edge transition (both low to high and high to low edge transitions) so that the output of the edge detector has r_b GHz component. The edge detector output is fed into a PLL (Phase Locked Loop). The PLL is to lock to the r_b GHz component, and its output is a recovered clock timing.

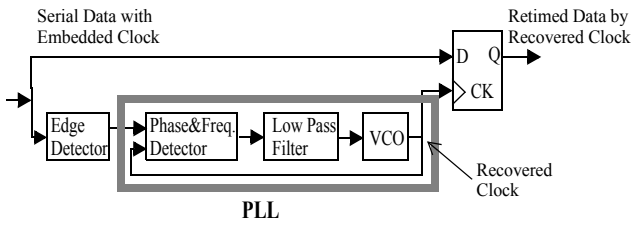


Fig.1 Clock and Data Recovery Circuit

2.1.2 Timing Jitter and Bit Error

The ideal relation between the data and the recovered clock at the input of the re-timing flip flop is shown in Fig.2 case-1, where the clock edge is placed in the center of the data bit. As shown in the case-2, let's assume that the data timing is shifted by J_{data2} , and the recovered clock timing is shifted by J_{clk2} by some reason. If J_{data2} and J_{clk2} are the same amount, the setup time and the hold time are the same as the case-1, and the data will be latched correctly. However, if the recovered clock timing does not exactly follow the data timing shift as shown in the case-3 ($J_{data3} > J_{clk3}$) for example, the data may not be latched correctly violating required setup time.

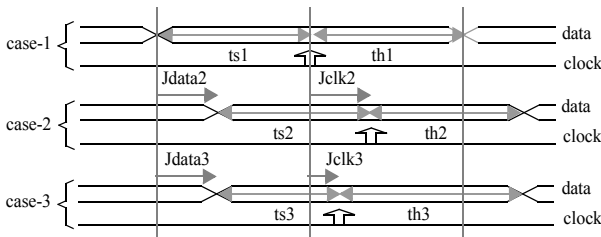


Fig.2 Data Jitter, Clock Jitter and Setup/Hold Time

Since the recovered clock timing does not precisely track instantaneous data timing shift, the timing margin for the setup time and the hold time varies from bit to bit as illustrated in Fig.3. When the setup time or the hold time is violated, bit error occurs and the bit error rate is determined by the PDF (probability density function) of the data timing relative to the recovered clock timing. Therefore how error-freely a CDR can work depends on the two major performance parameters as follow.

- how well the recovered clock edge is positioned around the center of data bits
- how well the recovered clock tracks incoming data timing shift, that is, data timing jitter

In order to guarantee the CDR performance, we have to test these key parameters, and a jitter tolerance test is to be designed for this purpose.

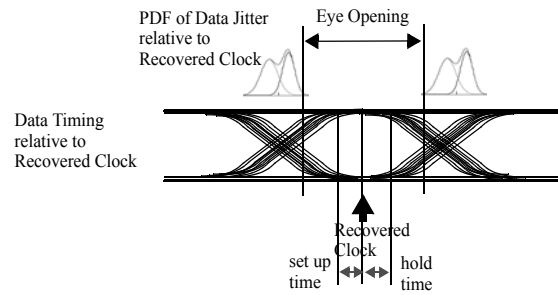


Fig.3 Data Eye Diagram relative to Recovered Clock

2.2 Jitter Tolerance Mask

What kinds of jitter and how much of them a CDR must tolerate are specified in each standard as the jitter tolerance mask with the expectation of 10^{-12} or better bit error rate in general. Jitter tolerance mask consists of various jitter components such as DJ, RJ and PJ attempting to duplicate actual jitter conditions. The jitter tolerance mask specifications for Fibre Channel [7] and XAUI [8] are summarized in Table.1 as examples [1].

	FC [7]	XAUI [8]
DJ	0.38 UI	0.37 UI
RJ	0.22 UI	-
DJ+RJ	-	0.55 UI
PJ	0.10 UI	0.10 UI
TJ	0.70 UI	0.65 UI

Table.1 Examples of Jitter Tolerance Mask

DJ and RJ masks are designed mainly to test how well the recovered clock is positioned around the center of data bits, which is one of the two major CDR performance parameters discussed in the subsection 2.1.2. On the other hand, PJ mask is designed to test another major performance parameter, that is, CDR's jitter tracking capability. The tolerable PJ amounts shown in Table.1 are only for high frequency PJ, and the complete PJ specification is described in a more complicated manner as discussed in the following subsection 2.2.1.

2.2.1 PJ Mask for Jitter Tracking Performance Test

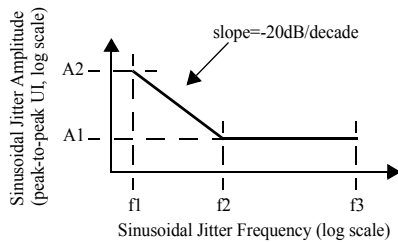
Typical clock recovery circuit is based on a PLL as shown in

Fig.1, and a PLL is basically a high-Q band pass filter whose center frequency is the clock frequency to be recovered [12][13]. Since a PLL is a band pass filter, the phase noise around the ideal clock signal is not completely filtered out. This means that the recovered clock tracks low frequency jitter in the input serial bit stream and the data will be successfully latched. The recovered clock, however, does not track out-of-band high frequency jitter in the bit stream and bit error could occur. Therefore PLL's band pass characteristic is essential to a CDR's performance, and PJ mask is designed to test the PLL's frequency response for this reason. For quantitative discussion, it is convenient to model the timing jitter with a phase modulated sinusoid. Let's express the embedded clock component, $Clock(t)$, by (Eq.1) [14]. $\theta(t)$ is the phase noise, and T_{ideal} is the ideal data bit duration. For PJ mask, sinusoidal jitter is used and is expressed by (Eq.2), where f_m is the injected PJ frequency.

$$\begin{aligned} Clock(t) &= A_c \cos[\omega_c t + \theta(t)] \\ &= A_c \cos\left[\omega_c \left(t + \frac{\theta(t)}{\omega_c}\right)\right] \end{aligned} \quad (\text{Eq.1})$$

$$\begin{aligned} \omega_c &= \frac{2\pi}{T_{ideal}} \\ PJ &= PJ(t) = \frac{\theta(t)}{\omega_c} \\ &= \frac{J_{pp}}{2} \cdot T_{ideal} \sin(\omega_m t) \\ \omega_m &= 2\pi f_m \end{aligned} \quad (\text{Eq.2})$$

Thus complete PJ mask specifies the relation between PJ amplitude and PJ frequency to be injected into the bit stream. Some examples are shown in Fig.4 [7][8][15]. Note that there are two distinctive segments in the PJ mask. As PLL does not track out-of-band high frequency jitter, the use of small and constant amplitude, high-frequency PJ injection can be intuitively understood. On the other hand, the meaning of the lower frequency PJ profile may not be so obvious, where PJ amplitude is inversely proportional to PJ frequency. Since it provides us with a guideline for implementing PJ injection using Direct Time Synthesis and with insight into the bit error mechanism, we study the PJ profile characteristic at lower frequencies in the following subsections 2.2.2 and 2.2.3.



	f1	f2	f3	A1	A2
FC (1.0625Gbps)	42.5kHz	637kHz	5MHz	0.1UI	1.5UI
XAU1 (3.125Gbps)	22.1kHz	1.875MHz	20MHz	0.1UI	8.5UI
OC-48 (2.488Gbps)	100kHz	1MHz		0.15UI	1.5UI

* OC-48 PJ mask extends to lower frequency below f1, but it is not shown here[15].

Fig.4 PJ Mask for Jitter Tolerance

2.2.2 Bit Error with Precise PJ Tracking

Even when a PLL precisely tracks the injected PJ, bit error can occur. The data transition timing in a bit stream is illustrated in Fig.5 with the PJ images as both longitudinal wave and transversal wave. Though it may sound counter-intuitive, the data eye opening becomes small, and therefore bit error is likely to occur when injected PJ amount becomes small, where bit time becomes small as seen in Fig.5.

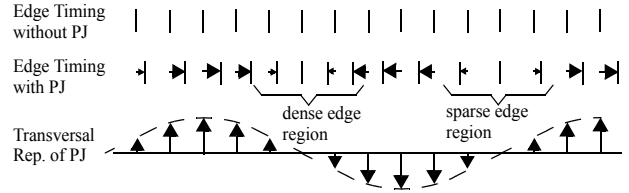


Fig.5 Edge Timing of Data Stream with/without PJ

From (Eq.1) and (Eq.2) with the observation in Fig.5, the minimum bit time T_{min} is expressed by (Eq.3), and approximated by (Eq.4). In order to recover the data bit correctly, the minimum bit time must be greater than a certain amount as expressed by (Eq.5) even with ideally positioned strobe clock. Otherwise the setup time or hold time is violated, and bit error occurs. From (Eq.4) and (Eq.5), the requirement for PJ peak-to-peak amplitude and PJ frequency for successful data recovery is obtained as expressed by (Eq.6), which indicates that injected tolerable PJ peak-to-peak amplitude is inversely proportional to PJ frequency for a given bit error rate.

$$T_{min} = T_{ideal} \left(1 - \frac{J_{pp}}{2} \cdot \sin(2\pi f_m T_{ideal})\right) \quad (\text{Eq.3})$$

$$\cong T_{ideal} \left(1 - \frac{J_{pp}}{2} 2\pi f_m T_{ideal}\right) \quad (\text{Eq.4})$$

$$> \alpha \cdot T_{ideal} \quad (0 < \alpha < 1) \quad (\text{Eq.5})$$

$$\therefore J_{pp} \cdot f_m < \frac{1 - \alpha}{\pi T_{ideal}} \quad (\text{Eq.6})$$

2.2.3 When PLL does not Track Inband PJ

As discussed in [2] and illustrated in Fig.6, the PJ amplitude of the recovered clock $J_{out_{pp}}(f_m)$ is proportional to the PJ amplitude of the injected jitter into the bit stream $J_{in_{pp}}(f_m)$ at a given PJ frequency f_m when the injected PJ amplitude is smaller than a certain value $J_{I_{cpp}}(f_m)$. Beyond this critical point, the linear relationship is broken between the injected PJ amplitude and the recovered clock's PJ amplitude, and it is also shown in [2] that bit error rate suddenly increases with injected PJ larger than this critical PJ amplitude.

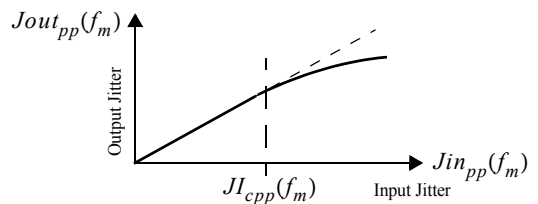


Fig.6 Output Jitter Amplitude vs. Input Jitter Amplitude

Our question here is why bit error rate suddenly increases when injected PJ amplitude becomes larger than a certain value, which varies depending on the PJ frequency that may seem to be small enough for a PLL to track. This phenomenon can be explained as follows.

From (Eq.1) and (Eq.2), the embedded clock component in a PJ injected bit stream is expressed by (Eq.7), and further expanded using Bessel function as expressed by (Eq.8) [13].

$$x(t) = A_c \cos \left[\omega_c \left(t + \frac{J_{pp}}{2} T_{ideal} \sin(\omega_m t) \right) \right] \quad (\text{Eq.7})$$

$$\begin{aligned} \omega_c &= \frac{2\pi}{T_{ideal}} \\ &= A_c \cos[\omega_c t + J_{pp} \pi \sin(\omega_m t)] \\ &= A_c \left(\sum_{n=-\infty}^{\infty} J_n(n, \beta) \cdot \cos(\omega_c + n\omega_m)t \right) \quad (\text{Eq.8}) \end{aligned}$$

$$\beta \equiv J_{pp} \pi : \text{modulation index} \quad (\text{Eq.9})$$

$J_n(n, \beta)$: Bessel function of the first kind

As the embedded clock is expressed by (Eq.8), the clock spectrum is known from Bessel function $J_n(n, \beta)$. In order to study how the embedded clock's spectrum varies depending on the peak-to-peak PJ amplitude, the Bessel function values for various modulation index β and various frequency components n are calculated, and shown in Fig.7.

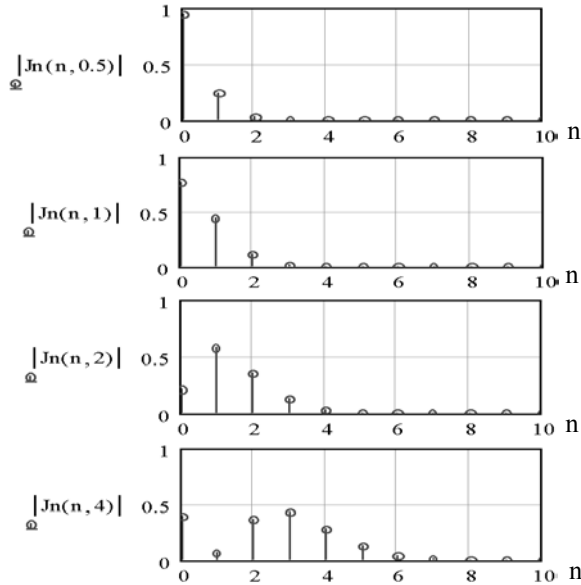


Fig.7 Phase Modulated Signal Spectra vs. Mod. Index

Since the modulation index β is proportional to the peak-to-peak PJ amplitude as defined by (Eq.9), Fig.7 actually shows how the embedded clock spectrum at the input of PLL varies depending on the injected PJ amplitude. It is noticeable that the larger the injected PJ amplitude is, the wider the occupied frequency bandwidth of the embedded clock. Since about 98% of the total energy is within the bandwidth of

$2(\beta+1)f_m = 2(\pi J_{pp} + 1)f_m$, when πJ_{pp} is much larger than 1, the occupied bandwidth of the embedded clock is approximated by $2\beta f_m = 2\pi J_{pp} f_m$, which is denoted (Eq.10).

$$\text{when } \pi J_{pp} \gg 1$$

$$\text{OccupiedBW}[x(t)] \approx 2\pi J_{pp} f_m \quad (\text{Eq.10})$$

Note that a PLL responds not to the jitter frequency but to the input signal frequency. At a given PJ frequency, when the PJ amplitude is so small that the embedded clock's occupied bandwidth is smaller than the PLL's cut off frequency, all the frequency components of the embedded clock are transferred from the input of the PLL to the output with a constant gain. Therefore the injected PJ is fully restored in the recovered clock, that is, the injected PJ is completely tracked by the PLL, and the PJ would not cause bit error. On the other hand, when the PJ amplitude is so large that the embedded clock's occupied bandwidth is larger than the PLL's cut off frequency, the higher frequency components of the embedded clock are more attenuated than the lower frequency components. Thus the injected PJ is only partially restored in the recovered clock, that is, the injected PJ is not completely tracked by the PLL, which would cause bit error. Thus we understand why bit error rate suddenly increases when injected PJ amplitude becomes larger than a certain value and why injected tolerable PJ amplitude is inversely proportional to PJ frequency for a given bit error rate.

3 Principle of PJ Injection by SPP-based ATE

We discuss how PJ of various frequency and amplitude can be injected into high-speed serial bit stream using Sequencer Per Pin architecture ATE.

3.1 SPP Architecture ATE

A very simplified block diagram of SPP architecture ATE is shown in Fig.8. Refer to [9] for more detail.

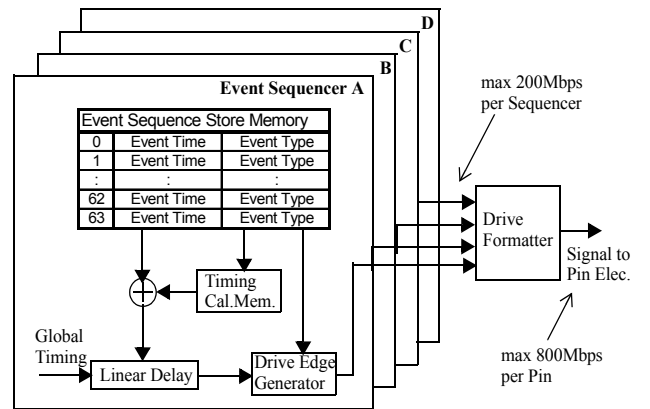


Fig.8 Sequencer Per Pin Block Diagram

An *event* consists of *event time* and *event type*, and an *event* can be specified at any time instance in a *tester period*, whose minimum value is 5ns. An *event* timing information with the *event time* and the *tester period* is called a *time set*. For exam-

ple, to generate input data to a DUT, we specify what drive signal (event type) occurs at what timing (event time). When input *event type* is specified, it is good enough to say it's a drive event, and whether the *event type* value is 1 or 0 is determined by the test vector. Thus we don't need to consume two *events* to specify data=1 and data=0 generated at the same time. As shown in Fig.8, recent ITS9000 family ATEs can define 64 different *events* per event sequencer, any *event* specified in each sequencer can be flexibly combined and repeated, and a drive formatter multiplexes four event sequencers outputs. Since each event sequencer can generate event sequence at up to 200Mbps (one *event* per 5ns), the maximum data rate at a drive formatter output is 800Mbps. In order to generate a serial bit stream at much higher data rate up to 3.2Gbps, a pin electronics driver IC multiplexes four drive formatters outputs as illustrated in Fig.9. This is like using 16 event sequencers multiplexed together to generate a high speed serial bit stream at one tester pin.

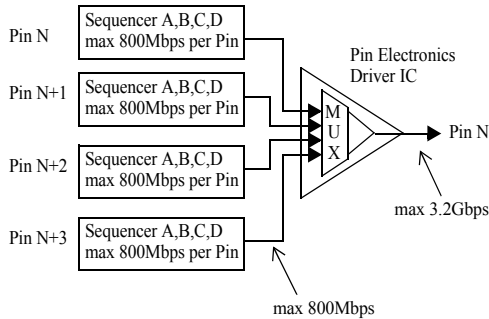


Fig.9 Pin Multiplexing for Max 3.2Gbps Generation

3.2 Periodic Jitter Injection

In order to describe the essence of the PJ injection by Direct Time Synthesis using SPP-based ATE, we examine two examples. One is high frequency PJ injection and another is low frequency PJ injection. The terms of *high frequency* PJ and *low frequency* PJ in this section 3.2 are used only to distinguish the two different programming styles described in the subsections 3.2.1 and 3.2.2 respectively. The two example PJs are expressed by (Eq.11) and (Eq.12) respectively, and they are shown in Fig.10.

$$J_{HF}(t) = A_{HF} \sin\left(\frac{2\pi}{8T_{ideal}}t\right) \quad (\text{Eq.11})$$

$$A_{HF} = 0.8T_{ideal}$$

$$J_{LF}(t) = A_{LF} \sin\left(\frac{2\pi}{64T_{ideal}}t\right) \quad (\text{Eq.12})$$

$$A_{LF} = 2.5T_{ideal}$$

The horizontal axis is normalized with ideal bit time T_{ideal} indicating the event numbers starting from 0. For the sake of brevity, 8 *events* are assumed in one *tester period* (*time set*) in this section 3.2 while the maximum of 16 events can be placed in the minimum *tester period* of 5ns as discussed in section 3.1,

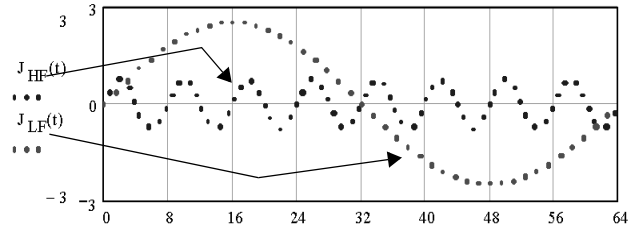


Fig.10 High Frequency and Low Frequency PJs

3.2.1 High Frequency PJ Injection

The *high frequency* PJ example is shown in Fig.11 again. Since the *high frequency* PJ period is only 8 bits long, we can specify all the PJ injected event timings using one *time set* as we have assumed 8 events per *time set* in this example.

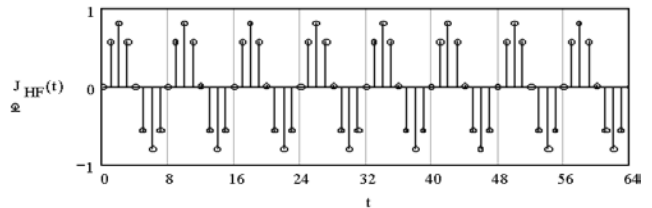


Fig.11 Jitter Amount to be Inject at Each Bit Timing

The *time set* programming is shown in Fig.12. The *event type* is a straightforward drive *event* denoted by "DF", and each *event timing* denoted by " t_n ($n=0..7$)" is to be calculated from the ideal bit time and the injecting PJ. The *tester period* in this *time set* is the PJ period. Once the *time set* is specified, the bit sequence is the repetition of this *time set* as shown in Fig.13.

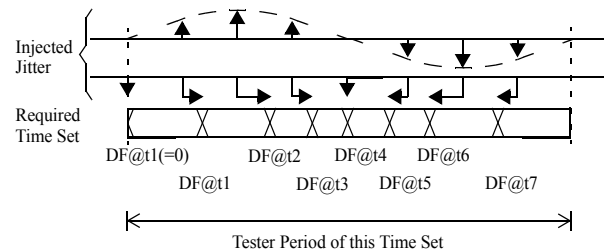


Fig.12 Time Set Programming for Jitter Injection

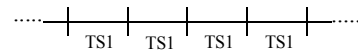


Fig.13 Single Time Set Sequence for High Freq. PJ

3.2.2 Low Frequency PJ Injection

The *low frequency* PJ example is shown in Fig.14 with two other signals $J_{approx}(t)$ and $J_{ts_delta}(t)$. Since the *low frequency* PJ period is 64 bits long, it seems that we need to use 8 different *time sets* at a glance. There are two new things to consider in this example.

- If we are to precisely program all the timings with lower frequency PJ, we may run out of *time sets* because we can use only up to 64 different *time sets*. To overcome this issue, we can approximate sinusoidal PJ by PWL (piece wise linear) signal, which is denoted by $J_{approx}(t)$

in Fig.14. The approximation error is shown in Fig.15. Though the maximum error in Fig.15 may appear to be large, it is because of this extreme example. Since real world applications use much smaller PJ amplitude at this PJ frequency, or use much lower PJ frequency with this PJ amplitude, actual approximation error can be made less than a half of the ATE's fine delay resolution. Combined with another consideration discussed below, the required number of the *time sets* can be reduced by half. Note that PJ may be approximated using the combination of PWL and stair step jitter profile when PJ frequency is very low, and refer to the section 7 Appendix for this method.

- When different *time sets* need to be cascaded as in this example, care must be taken when *event time* is specified in each *time set*. The absolute event timing of an *event* in each *time set* with respect to the timing of the first event of the bit stream during one PJ cycle is known from the ideal bit time and the PJ amplitude and frequency. However the *event time* being specified in each *time set* is not the absolute event timing but the difference between the absolute event timing and the jitter time amount that has already passed until this new *time set* is to be used. This time difference is added to the ideal event time as we did for high frequency PJ programming when *event time* is specified in each *time set*. The set of these time differences is denoted by $J_{ts_delta}(t)$ and shown in Fig.14.

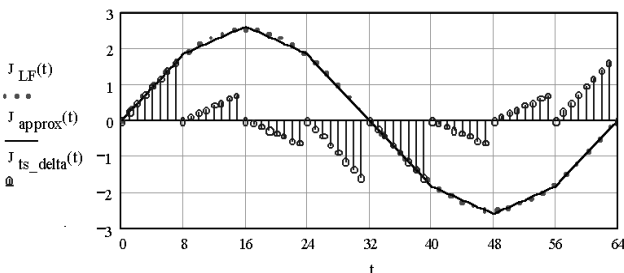


Fig.14 PJ Approximation by PWL Jitter Profile and Jitter Amount to be Injected at Each Bit Time in Various Time Sets

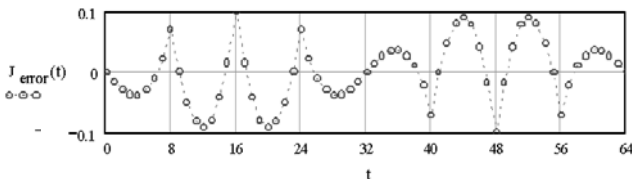


Fig.15 Approximation Error by PWL Jitter Profile

Thus four different *time sets* are needed in this example, and they are sequenced as shown in Fig.16 requiring 8 *time sets* for one PJ cycle injection. The group of the 8 *time sets* is repeatedly used to generate long bit stream.

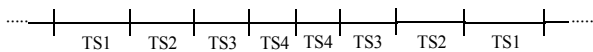


Fig.16 Multiple Time Sets Sequence for Low Freq. PJ

4 SPP-based Jitter Injection Accuracy

The most critical question for the jitter injection method discussed in the section 3 is whether or not this method can inject jitter accurately enough for SerDes jitter tolerance test. When

PJ to be injected is properly approximated, PJ injection accuracy becomes the same as the ATE's timing accuracy because of the nature of Direct Time Synthesis. Therefore we study the timing accuracy of the SPP-based ATE from the view point of high speed serial communication device test.

4.1 EPA versus TJ

Timing accuracy of digital ATE is usually described by OTA (overall timing accuracy) [17][18] and/or EPA (edge placement accuracy) [18]. While OTA is used for the combined accuracy of drive side and strobe side, EPA is used for either drive side or strobe side accuracy, and drive side EPA is of our interest in this article. EPA/OTA is often discussed concerning yield loss and escape [17][19][20], and the coverage of the random error in EPA/OTA is about six times of the standard deviation σ ($\pm 3\sigma$), which is either explicitly mentioned [10] or inferred from the data sample numbers (hundreds to thousands) of the measurements used to estimate EPA/OTA [18]. On the other hand in high speed serial communication device test, TJ (total jitter) is used to describe data timing uncertainty and/or degradation, and their primary concern is BER (bit error rate) at the order of 10^{-12} [7][8]. In short, if the PDF (probability density function) of an ATE's timing uncertainty were to be modeled by a Gaussian distribution (standard deviation σ , mean 0), the EPA with $\pm 3\sigma$ would mean about 10^{-3} BER while the BER of 10^{-12} would mean about $\pm 7\sigma$.

Shown in Fig.17 is an eye diagram of 3.2Gbps PRBS15 bit stream generated by our ATE, which was measured by a sampling oscilloscope at the pogo pin.

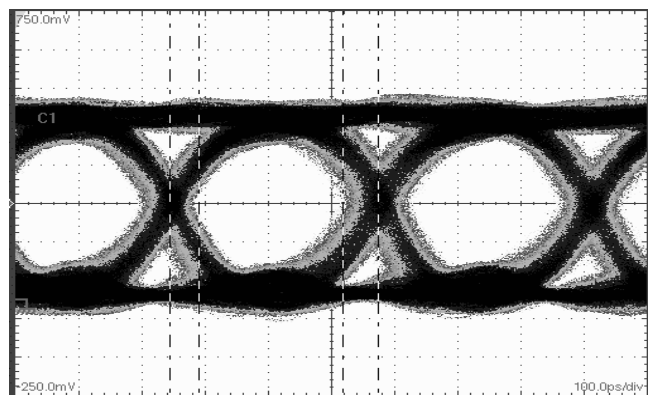


Fig.17 Eye Diagram at 3.2Gbps with PRBS15

From the eye diagram in Fig.17, and the associated oscilloscope readings of the 14.9ps rms jitter (σ_{meas}) and the 92.0ps peak-to-peak jitter, this ATE's EPA may be considered ± 50 ps. Since the measured peak-to-peak jitter (92.0ps) is very close to the six times value ($\pm 3\sigma_{meas}$) of the measured rms jitter (89.4ps = 6×14.9 ps), the jitter PDF seems to be close to a Gaussian distribution. Our question here is if the 14 times value ($\pm 7\sigma_{meas}$) of the measured rms value (208.6ps = 14×14.9 ps) is a good estimate of the ATE driver output's TJ. In other word, does the $14\sigma_{meas}$ include all but 10^{-12} of the jit-

ter population? In order to answer this question, we study the drive side timing accuracy in the next section 4.2, which resulted in the jitter performance shown in Fig.17.

4.2 Timing Accuracy Analysis of Modern ATE

In order to analyze our ATE's timing accuracy from the view point of high speed serial communication device test, we utilize the jitter analysis methodologies which have been used for high speed serial communication protocol standards [7][8][21]. We use *jitter* and *timing error* interchangeably in this section. Compared with the conventionally-used algebraic sum of all the timing error components for EPA/OTA estimations [10][18], the distinctive features of these jitter analysis methodologies are the following.

- What matters is BER, and TJ can be correlated to BER through its PDF. Therefore TJ value (peak-to-peak total jitter) varies depending on the BER (10^{-3} , 10^{-12} , etc.) of our interest.
- TJ consists of DJ and RJ. DJ is bounded while RJ is not bounded. RJ's PDF is to be approximated by Gaussian distribution by definition.
- DJ and RJ are convolved, resulting in TJ. When DJ and RJ consist of subcomponents, and there is no correlation among the subcomponents, they are convolved, resulting in the total DJ and the total RJ.

We start from identifying various jitter (timing error) sources. The timing critical building blocks of the ATE are shown in Fig.18 along with the jitter caused by each building block. As discussed in the section 3.1, four event sequencers are multiplexed per pin, and four pin-timing-resources are multiplexed to generate high speed serial data at up to 3.2Gbps.

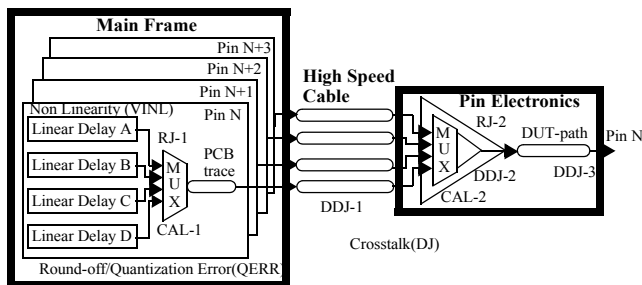


Fig.18 Various Jitter Sources in SPP ATE Drive-side

There exists intrinsic noise in electronic systems such as thermal noise, shot noise, flicker noise etc., and their PDFs are Gaussian distributions [22]. The RJs due to these intrinsic noises in Fig.18 are denoted by RJ-1 in the main frame and RJ-2 in the pin electronics. The phase noise of the ATE's master clock is a part of RJ-1. The bounded timing error (DJ) consists of data/pattern dependent error (DDJ-1, DDJ-2, DDJ-3), fine-delay/timing-vernier non linearity (VINL), round-off/time-quantization error (QERR), and calibration error (CAL-1, CAL-2). There is no noticeable amount of PJ observed in this ATE. These timing error components are summarized in Fig.19.

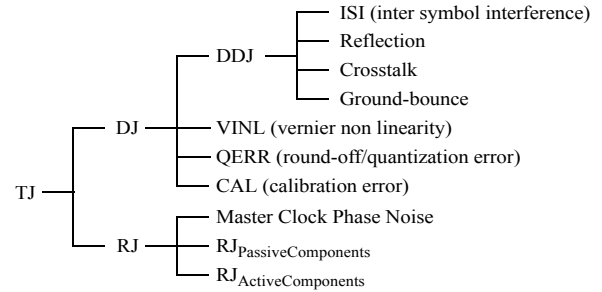


Fig.19 Jitter Categories and Jitter Sources

Now that all the jitter (timing error) sources of the ATE are identified, the next step is to estimate their amounts. The convolution of two PDFs is defined by (Eq.13) [13]. RJ-1 and RJ-2 have their own Gaussian distributions, and they are convolved as expressed by (Eq.14) resulting in the total RJ. We use the 5ps standard deviation for the total RJ in the following discussion since 4ps to 5ps rms jitter was measured at a pogo pin in our test system using 1010... signal and avoiding VINL, QERR and CAL.

$$p_x(t) * p_y(t) \equiv \int_{-\infty}^{\infty} p_x(t-\tau) \cdot p_y(\tau) d\tau \quad (\text{Eq.13})$$

$$p_{RJ}(t) = p_{RJ1}(t) * p_{RJ2}(t) \quad (\text{Eq.14})$$

For DJ, we assume that the PDFs for DDJ, VINL, QERR and CAL are to be approximated by uniform distributions as shown in Fig.20. Since actual PDFs for these jitter (timing error) sources are much more complicated than uniform distributions, we will have to improve our PDF models for more accurate analysis in the future. Uniform distribution, however, would not under-estimate TJ considering that a random variable tends to concentrate around its mean value such as Gaussian distribution. The studies of DJ PDF being convolved in TJ in datacom application began only recently [21]. The impact on TJ estimation of three different DJ PDFs (triangle, rectangular/uniform, double-delta) is found in [21].

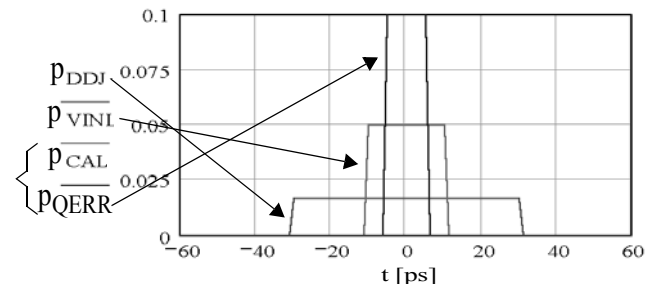


Fig.20 PDFs of Various Error Components of the ATE

As the fine delay resolution of the ATE is 10ps, the +/-5ps PDF is used for both CAL and QERR. From the characterization results of the timing-vernier linearity error [23], the +/-10ps PDF is used for VINL. The +/-30ps PDF is used for DDJ, and it is worth discussing more in detail about DDJ. As shown in Fig.19, DDJ consists of several subcomponents. ISI (inter symbol interference) is the primary contributor [16],

and jitter due to reflection (impedance discontinuity) [24] is the secondary one. The physical locations of the DDJ sub-components are distributed over the entire timing-critical signal paths as illustrated in Fig.18. Apart from ISI, since there is almost no correlation among the DDJs at different locations and among different sub-components, their PDFs are to be convolved, resulting in Gaussian like PDF. This means that most of the jitter population is around zero and the probability to have very large jitter is small. On the other hand, the ISIs at different locations but in the same single signal path have strong correlation among them (same data pattern propagates through them), and therefore, ISI accumulation is more like an algebraic sum than a geometric sum, which makes ISI the primary DDJ contributor in our current generation ATE. Thus the PDF of the total DDJ is a function of DDJ-1 through DDJ-3 as expressed by (Eq.15), but not a simple convolution among them. Then the PDF of the total DJ is obtained as the convolution of the PDFs of DDJ, VINL, QERR and CAL as expressed by (Eq.16).

$$p_{DDJ}(t) = Function(p_{DDJ1}(t), p_{DDJ2}(t), p_{DDJ3}(t)) \quad (Eq.15)$$

$$p_{DJ}(t) = p_{DDJ}(t) * p_{VINL}(t) * p_{QERR}(t) * p_{CAL}(t) \quad (Eq.16)$$

The total DJ PDF $p_{DJ}(t)$ and the total RJ PDF $p_{RJ}(t)$ are shown in Fig.21. As expected, the total DJ PDF is not a uniform distribution, which means that the worst case DJ does not often happen.

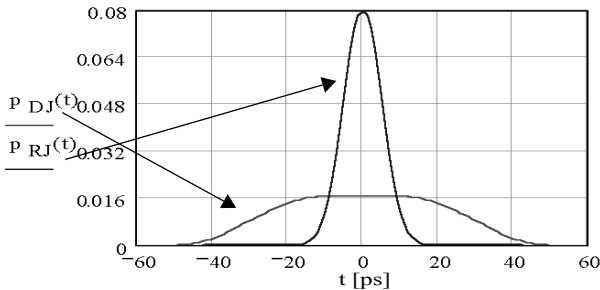


Fig.21 PDFs of DJ and RJ of an ATE

By convolving the total DJ PDF and the total RJ PDF, we can obtain the TJ PDF $p_{TJ}(t)$ as expressed by (Eq.17), and it is shown in Fig.22 with $p_{DJ}(t)$. Since the positive side and the negative side of $p_{TJ}(t)$ and $p_{DJ}(t)$ are symmetric, only the positive side of them is shown.

$$p_{TJ}(t) = p_{DJ}(t) * p_{RJ}(t) \quad (Eq.17)$$

Since the difference between the two PDFs is very small as noticed in Fig.22, it is very difficult to accurately estimate $p_{DJ}(t)$ and $p_{RJ}(t)$ only from $p_{TJ}(t)$ measurements (solving inverse problem or inverse modeling), and therefore forward modeling is useful and helpful as shown in this article in order to analyze ATE's timing accuracy in detail.

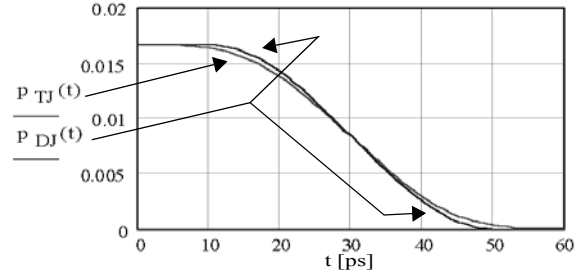
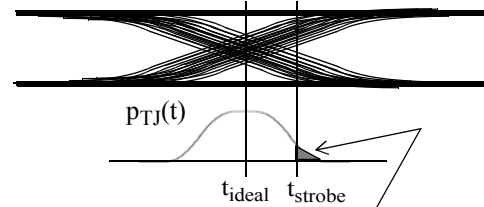


Fig.22 Positive Halves of TJ and DJ PDFs of an ATE

As briefly mentioned in the beginning of this subsection, TJ value (peak-to-peak total jitter) varies depending on the BER (10^{-3} , 10^{-12} , etc.) of our interest. TJ PDF $p_{TJ}(t)$ is related to BER by (Eq.18) [7], and the physical meaning of (Eq.18) is illustrated in Fig.23. Actual data bit timing has a distribution around its ideal edge timing, whose PDF is expressed by $p_{TJ}(t)$. If the bit stream is strobed at a certain timing with respect to the ideal bit timing, bit error occurs when the data bit arrives after strobe timing, and the probability of the bit error occurrence is determined by the probability of the data bit arrival time and the strobe timing as expressed by (Eq.18).



$$\frac{1}{2}BER(t_{strobe}) = \int_{t_{strobe}}^{\infty} p_{TJ}(t) dt \quad (Eq.18)$$

Fig.23 BER, $p_{TJ}(t)$ and Strobe Timing

Using (Eq.18), BER for our $p_{TJ}(t)$ is calculated as a function of strobe timing, and the positive half of the BER is plotted in Fig.24 with its vertical axis in log scale.

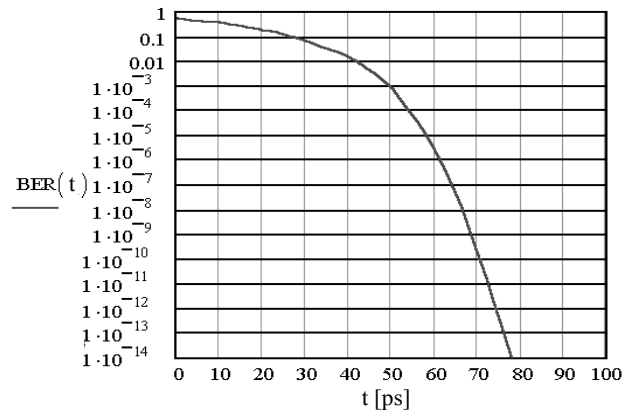


Fig.24 Positive Half of an ATE's TJ vs. BER

The BER curve in Fig.24 can be interpreted as follows.

- When the bit stream is strobed 50ps after the ideal bit timing, the BER will be the order of 10^{-3} . This means that

about 99.9% of the jitter population is within ± 50 ps. Since 99.7% of the population is included within $\pm 3\sigma$ of a Gaussian distribution (its standard deviation is σ), the EPA of this ATE may be considered ± 50 ps, and this is what we are likely to observe in the eye diagram measured by an oscilloscope with hundreds to thousands of samples.

- With the expectation of $BER=10^{-3}$, the TJ of this bit stream is 100ps ($=2*50$ ps).
- With the expectation of $BER=10^{-12}$, the TJ of this bit stream is 150ps ($=2*75$ ps).

Thus we have developed a methodology to understand the relation between an ATE's EPA and TJ for a given BER. Note that the 60ps peak-to-peak DDJ is at 3.2Gbps, and DDJ at lower data rate is smaller than this.

5 Further Work Areas and Limitations

Our work in this article provides only a foundation for further exploitation of jitter injection by Direct Time Synthesis. A couple of critical further work areas and limitations are pointed out below.

5.1 Extensive Measurements and Flexibility Study

The PJ injection method has been discussed mainly from theoretical and conceptual points of view in this article. We need to carry out extensive measurements to further study and validate the accuracy of injected PJ frequency and magnitude over the entire frequency specification range such as the ones shown in Fig.4 with small frequency step at various data rates. Because of the limited number of the *time sets*, there are some PJ frequencies that cannot be generated. Theoretical work is needed to figure out all the PJ frequencies which cannot be achieved, and an algorithm needs to be developed to generate proper *time sets* for any achievable PJ frequency and amplitude.

5.2 Jitter Calibration and Tester Guard-band

While the calibration for jitter injection, which may be best performed by BERT(bit error rate tester) Scan[1][7][8], is required for our method too, the jitter injection by Direct Time Synthesis using our current generation ATE is not accurate enough for compliance test and/or design validation. Our intention is to apply this method to high volume production test. Though actual devices will have a certain amount of margin by design for the jitter mask required in each protocol standard, some samples will not work as well as expected due to manufacturing defect and/or process variation. Therefore another kind of calibration will be required regarding how to set a proper guard band for production test, where yield loss and escape need to be considered.

5.3 Data Dependent Jitter Injection

Since Direct Time Synthesis method directly manipulates bit timing, not only PJ but also DJ can be conveniently injected. For example, since the DDJ due to the limited frequency bandwidth of a data transmission medium can be predicted

when the loss characteristic of the medium is known [5][16], the expected DDJ can be injected into the bit stream for jitter tolerance test. There are two kinds of DDJ injection setups resulting in two very different eye shapes [1]. One is the Fibre channel recommended setup[7], which uses a limiting amplifier after DDJ injection filter/cable to shape the waveform. Another is the XAUI recommended setup[8], where the output of DDJ injection filter/cable is directly fed to the DUT. What can be achieved with Dynamic Time Synthesis is Fibre channel-type eye shape. DDJ filter/cable needs to be placed somewhere between the pin electronics driver and the DUT for XAUI type DDJ injection.

5.4 Non-ideal DUT Signal Path

While we intend to inject controlled amount of jitter, additional DJ is inevitably induced by non-ideal signal path between ATE's driver IC and DUT input, and between DUT output and ATE's comparator IC [24]. The additional DJ is caused by multiple reflections among various impedance discontinuities, signal loss due to lossy transmission lines on both ATE itself and device loadboard, DUT socket used for production test, etc. In order to improve the timing accuracy for high speed device test, we have to keep improving DUT signal path quality from various points, and need to come up a way to accurately estimate the inevitable DJ amount and calibrate it out.

6 Conclusion

In order to understand the requirements for multi-gigabit SerDes jitter tolerance test, we have discussed how a clock and data recovery circuit works and fails to work, and studied the meanings of PJ mask specified in high speed serial link protocol standards.

As a cost effective high volume production test solution, we have proposed a jitter injection method based on Direct Time Synthesis using a Sequencer Per Pin architecture ATE. This method can inject periodic jitter of various frequency and amplitude, and does not require any specialized ATE hardware resource or any additional component on a DUT loadboard.

In order to assess ATE's timing accuracy for jitter injection, we have applied a new methodology to study the relationship between traditionally used EPA and bit error rate dependent TJ, with which ± 50 ps EPA of our test system is translated into 150ps TJ for $BER=10^{-12}$.

7 Appendix: PJ Approximation by PWL and Stair Step Jitter Profile

As understood from the discussion in the subsection 4.2, the jitter injection accuracy is basically the same as normal data generation accuracy if the jitter to be injected is not approximated or the approximation error is smaller than half of the fine timing delay resolution. We have discussed in the subsection 3.2.2 how to approximate low frequency PJ using

PWL jitter profile. When PJ frequency is very low, we will need to approximate the PJ using the combination of PWL and stair step jitter profile, which is illustrated in Fig.25. In this case, the PJ to be injected is approximated by PWL first. Since the slope of each segment of the PWL jitter profile is very small because of the very low jitter frequency, the quantization error caused by approximating a PWL segment with stair steps can be made smaller than a half of the fine delay resolution, that is, less than ± 5 ps in the ATE analyzed in the subsection 4.2.

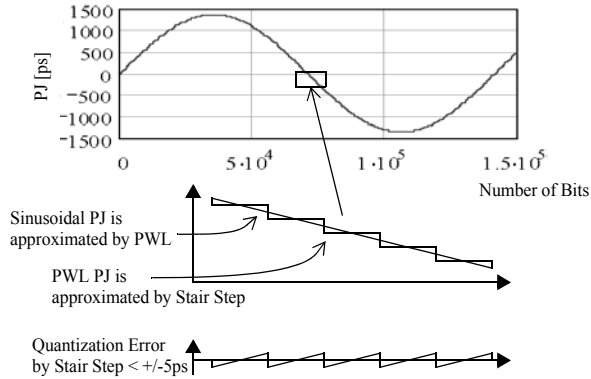


Fig.25 Quantization Error due to PJ Approximation by Stair Step Jitter Profile

For example, the PJ mask for XAUI requires 8.5UIpp sinusoidal jitter injection at 22.1kHz[8]. The largest bit timing shift due to the injected PJ during 64 bits duration is less than 4ps. This means that the quantization error with stair step jitter profile for 128 bits duration is less than ± 4 ps.

8 Acknowledgements

We thank Howard Maassen for useful discussion about Sequencer Per Pin-based jitter injection. We appreciate our colleagues, Curt Meyers, Kai Chan, Robert Glenn, and Neil Ridgway for their timing calibration enhancement and timing accuracy characterization of our test systems.

References

[1] Y.Cai, S.A.Werner, G.J.Zhang, M.J.Olsen, R.D.Brink, "Jitter Testing for Multi-Gigabit Backplane SerDes", ITC Proceedings, pp700-709, 2002

[2] T.Yamaguchi, M.Soma, M.Ishida, H.Musha, L.Marlarsie, "A New Method for Testing Jitter Tolerance of SerDes Devices Using Sinusoidal Jitter", ITC Proceedings, pp717-725, 2002

[3] B.Laquai, Y.Cai, "Testing Gigabit Multilane SerDes Interfaces with Passive Jitter Injection Filters", ITC Proceedings, pp297-305, 2001

[4] Y.Cai, T.P.Warwick, S.G.Rane, E.Masserrat, "DIGITAL SERIAL COMMUNICATION DEVICE TESTING AND ITS IMPLICATIONS ON AUTOMATIC TEST EQUIPMENT ARCHITECTURE", ITC Proceedings, pp600-609, 2000

[5] Udaya Shankar.N, "TEST CHALLENGES FOR SO-

NET/SDH PHYSICAL LAYER OC3 DEVICES AND BEYOND", ITC Proceedings, pp502-511, 2001

[6] J.Wilstrup, "A METHOD OF SERIAL DATA JITTER ANALYSIS USING ONE-SHOT TIME INTERVAL MEASUREMENTS", ITC Proceedings, pp819-823, 1998

[7] National Committee for Information Technology Standardization, T11.2/Project 1316-DT/Rev 4.0, "Fibre Channel -Methodology for Jitter and Signal Quality Specification", November 12, 2001

[8] IEEE Draft P802.3ae, D4.3, Supplement to Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method & Physical Layer Specifications, April 2002

[9] B.West, T.Napier, "Sequencer Per Pin Test System Architecture", ITC Proceedings, pp355-361, 1990

[10] L.Sartori, B.West, "THE PATH to ONE-PICOSECOND ACCURACY", ITC Proceedings, pp619-627, 2000

[11] Maxim data sheet, MAX3880 2.488Gbps SHD/SO-NET 1:16 Deserializer with Clock Recovery, 1999

[12] B.Razavi, MONOLITHIC PHASE-LOCKED LOOPS AND CLOCK RECOVERY CIRCUITS, Wiley-IEEE Press, 1996

[13] R.E.Ziemer, W.H.Tranter, "Principles of Communications", Houghton Mifflin Company, 1976

[14] M.Shimanouchi, "An Approach to Consistent Jitter Modeling for Various Jitter Aspect and Measurement Methods", ITC Proceedings, pp848-857, 2001

[15] Bellcore technical reference: TR-NWT-000253, Issue 2, December 1991

[16] M.Shimanouchi, "New Paradigm for Signal Paths in ATE Pin Electronics are Needed for Serialcom Device Testing", ITC Proceedings, pp903-912, 2002

[17] INTERNATIONAL TECHNOLOGY ROADMAP FOR SEMICONDUCTORS, TEST AND EQUIPMENT, 2001

[18] SEMI G80-0200, "TEST METHOD FOR THE ANALYSIS OF OVERALL DIGITAL TIMING ACCURACY FOR AUTOMATED TEST EQUIPMENT"

[19] B.West, "Accuracy Requirements in At-Speed Functional Test", ITC Proceedings, pp780-787, 1999

[20] Wajih Dalal, Song Miao, "The Value of Tester Accuracy", ITC Proceedings, pp518-523, 1999

[21] M.P.Li, J.B.Wilstrup, "On the Accuracy of Jitter Separation From Bit Error Rate Function", ITC Proceedings, pp710-717, 2002

[22] H.W.Ott, "Noise Reduction Techniques in Electronic Systems", Bell Telephone Laboratories Inc., 1976

[23] A.R.Syed, "R4X/D4X - Formatters for Flexible Test System Architecture", ITC Proceedings, pp885-893, 2002

[24] T.P.Warwick, "What a Device Interface Board Really Costs: An Evaluation of Technical Considerations for Testing Products Operating in the Gigabit Region", ITC Proceedings, pp555-564, 2002