How Transmit Jitter Propagates Through the Channel and What is Jitter Amplification

By: Vladimir Dmitriev-Zdorov, Mentor, A Siemens Business

In this detailed study of jitter, Vladimir Dmitriev-Zdorov considers its conversion into vertical noise and the propagation of this noise in a channel. It applies to many, if not all, linear systems. Read on to see his analysis as illustrated in both the frequency and time domains for small and large magnitude jitter.

There is a general understanding that the amount of jitter or time interval error (TIE) observed at the output of a lossy or reflective channel is larger than jitter or TIE of the transmitted signal. Even with an ideally clocked input signal, TIE in the output waveform can appear as jitter in the eye diagram. But what exactly happens if input jitter is increased by a certain value, such as a fraction of the unit interval (UI)? Will the output jitter increase by the same amount or more?

It turns out that in most cases an increment of the input jitter by a certain fraction of the UI (while keeping its distribution and spectral properties unchanged) makes the output jitter increase by a larger amount. This effect is called “jitter amplification.” It has been known for decades and studied in numerous publications.[1-4] Still, this phenomenon remains somewhat of a mystery, because it lacks a general and simple explanation.

https://www.signalintegrityjournal.com/articles/2081-how-transmit-jitter-propagates-through-the-channel-and-what-is-jitter-amplification

Some people believe that jitter propagation through a channel can be studied in the same way as propagation of voltage, current, or waves. If so, the problem can be solved by finding a jitter propagation or transfer function to greatly simplify jitter study, prediction, and possibly, mitigation. Indeed, such characteristics are found for some channels, and specific jitter properties, mostly assuming small jitter magnitude and/or certain types of distribution, spectral, and correlation properties. Unfortunately, no general solution exists because jitter propagation cannot be described by a linear transfer function, even for a perfectly linear time invariant channel.

There are some fundamental reasons why. Most importantly, jitter is not a physical parameter directly associated with energy possessed by a physical system. Unlike jitter, the voltages, currents, or scattering parameters measured at the ports of an electrical circuit are directly related to the power that comes in or out and, therefore, the power that is stored or dissipated in the system.

As power-related parameters, they obey several conservation laws, such as Kirchhoff's current and voltage laws. These laws, together with component relationships (Ohm's law) make it possible to superimpose the values of these physical parameters in linear circuits. However, no similar reasoning applies to jitter. For example, doubling the peak-to-peak or RMS value of jitter at the input does not necessarily translate to a doubling at the output; and, the combination of two jitter types does not necessarily produce the sum of their separate effects.

Our goal is not to find a jitter transfer function (even if it is possible in some cases), but to think about jitter propagation as a combination of at least two transformations. First, input jitter, a timing variation of the transition, creates vertical "noise" which can be found as the difference

between the displaced and non-displaced transitions. Depending on jitter type, this noise could be deterministic or random. Then, the vertical noise at the channel's input (in a form of a voltage, current, or scattering wave) propagates toward the receiver where it converts back into timing variation or output jitter.

The diversity of jitter types, wide multiplicity of digital signals and variety of different channel characteristics make a unified consideration of jitter propagation impossible. This article starts from the simplest and most popular case of a meander input with duty cycle distortion, illustrating the traditional method and the study of jitter-to-noise and noise-to-jitter transformations. Then, the propagation of small uncorrelated Gaussian jitter is considered. Finally, a general approach is described that applies to large magnitude jitter interacting with an irregular or random input pattern in a lossy channel.

PERIODIC CLOCK SIGNAL AND DUTY CYCLE DISTORTION

This is the simplest and most frequently used example to illustrate jitter propagation because the channel's input and output are deterministic periodic signals. The input signal is a periodic meander input with symbol interval T_b and period $T = 2T_b$. The blue waveform in **Figure 1** illustrates an undistorted meander with a duty cycle ratio (DCR) of 0.5; the red one has a

$DCR = (2a / T) = a / T_b < 0.5$

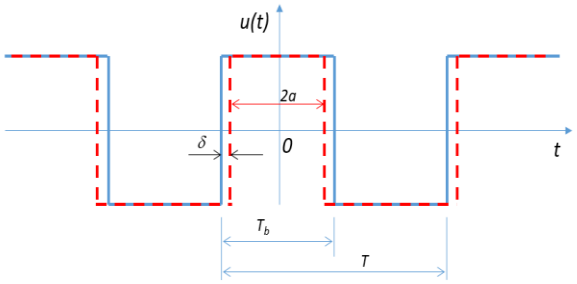


Fig. 1 Periodic meander input with duty cycle distortion.

For any single transition, the timing delta is:

$$\delta = (0.5 - DCR) / T_b .$$

A positive delta creates positive delay for the rising transition and negative for the falling. With the DCR > 0.5, the sign of delta is negative thus reversing the signs of transition delays as well.

What is seen at the output of lossy channels? Consider two channels, providing respectively 3 and 6 dB of loss at the Nyquist frequency with matched terminations (see **Figure 2**). The output waveforms and eye diagrams are shown in **Figure 3**.

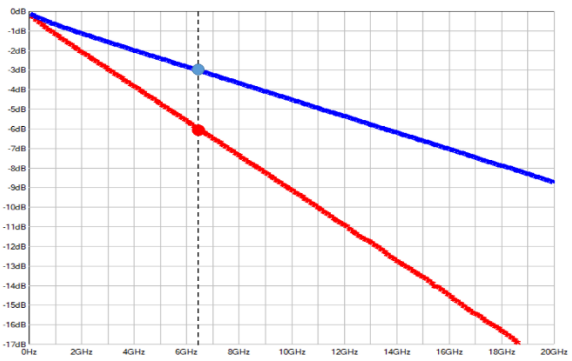


Fig. 2 Channel transfer functions representing 3 (blue) and 6 (red) dB of loss at the Nyquist frequency.

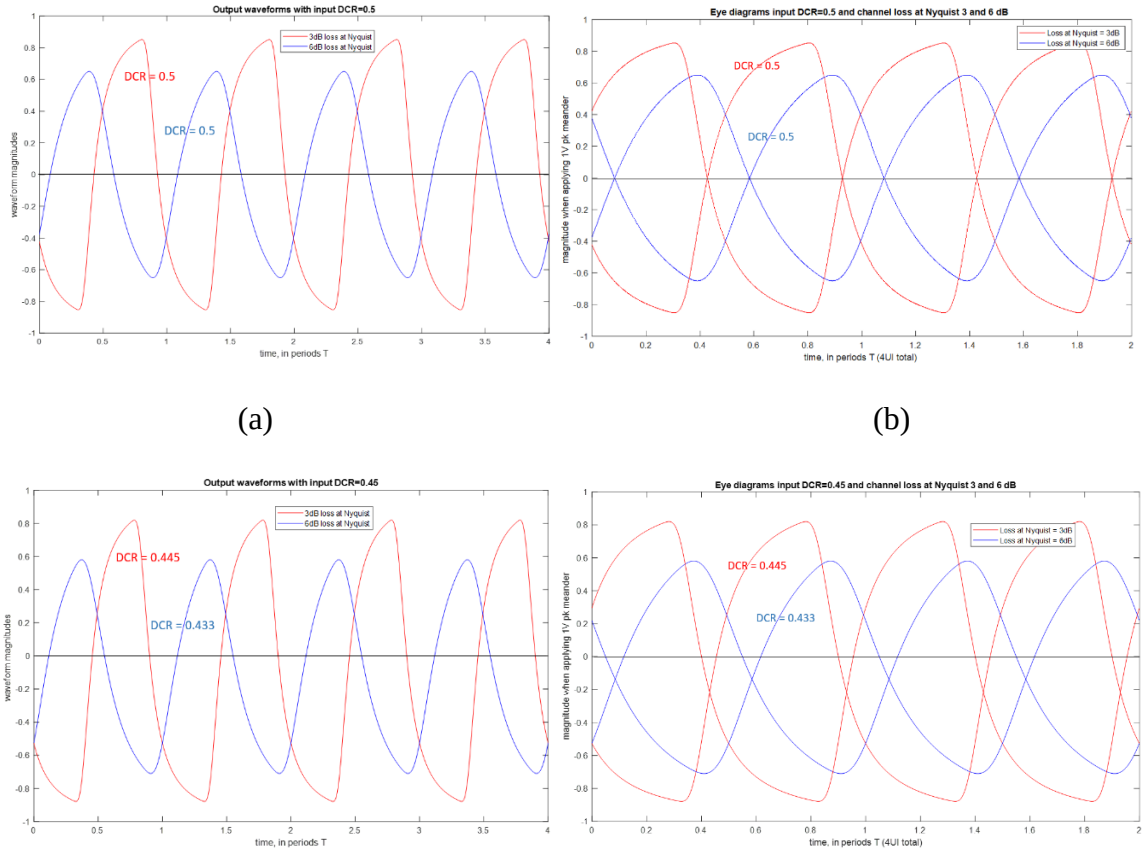


Fig. 3 Output waveforms (a) and eye diagrams (b) with an undistorted input (DCR = 0.5). Similarly, output waveforms (c) and eye diagrams (d) for a moderately distorted input (DCR = 0.45).

In both cases, the DCR is measured at the channel output as a portion of the UI where the periodic signal is positive. This value can be easily expressed via the TIE; or, it can be thought of

as a DCR of the digital signal restored from the output waveform by amplification and clamping. Evidently, undistorted input meander creates an attenuated output with the DCR = 0.5 regardless of the channel loss, see Figures 3a and b. With a non-ideal input, however, distortion increases in the lossy channel, and to a larger extent when the loss is higher. This seen in Figures 3c and d. Vertical asymmetry of the eye diagram is also observed when comparing Figures 3b and d. In general, such asymmetry always appears when timing or magnitude distortion is correlated with the digital pattern itself.

What happens if channel loss is increased, or the input DCR is different? This is shown in **Figure 4**, where each curve corresponds to a certain DCR value at the channel's input. Distortion increases with channel loss and grows faster with greater input distortion.

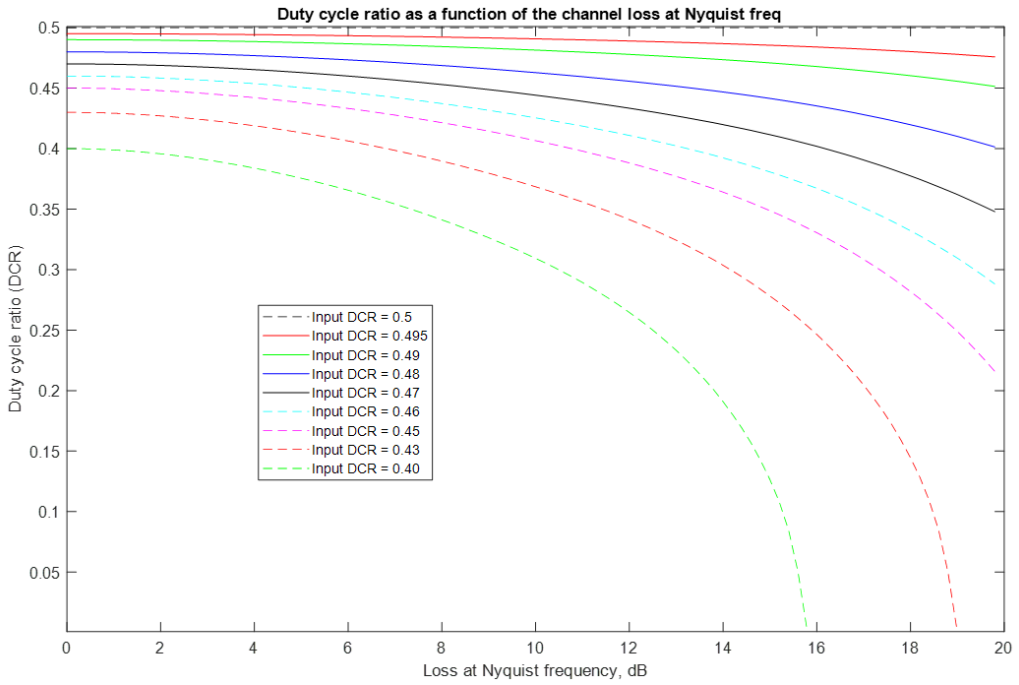


Fig. 4 DCR of the output signal as a function of channel loss at the Nyquist frequency.

Why does the DCR (or peak jitter value) change in the lossy channel. This is explored in two different ways:

1. By analyzing the signal spectrum. Since the input is periodic in time, it can be represented by a Fourier series. By multiplying the magnitudes of the spectral components with the corresponding complex values of the channel's transfer function, the discrete spectrum of the output signal (which is also periodic) is found. The output waveform is recovered by converting the discrete spectrum back into time domain.
2. By considering the problem in time domain. For this purpose, the output signal is represented by a combination of the channel's step responses with different signs and delays.

Explaining Jitter Amplification in the Frequency Domain

A periodic time domain function can be represented by a Fourier series of the form:

$$u(t) = \frac{A_0}{2} + \sum_{n=1}^{\infty}\{A_n \cos(2\pi nt / T) + \sum_{n=1}^{\infty}\{B_n \cos(2\pi nt / T)\} \quad (2)$$

Where:

$$A_n = \frac{2}{T}\int_T u(t)\cos(2\pi nt / T)dt$$

$$B_n = \frac{2}{T}\int_T u(t)\sin(2\pi nt / T)dt$$

By selecting integration limits symmetrical in time $(-T_b, T_b)$ all coefficients B_n are zero and only the cosine components with real magnitudes remain:

$$u(t) = \frac{A_0}{2} + \sum_{n=1}^{\infty} A_n \cos(2\pi nt / T), \text{ where } A_0 = \frac{8a-2}{T} \text{ and } A_n = \frac{4}{\pi n}\sin(2\pi na / T) \quad (3)$$

Magnitudes of the harmonics are shown in **Figure 5**. For the ideal input (DCR = 0.5) the DC component A_0 and the magnitudes of all even harmonics are zero because $sin\ (2\pi na/T)$ equals $sin\ (\frac{\pi}{2}n)$ which is identical zero for even n.

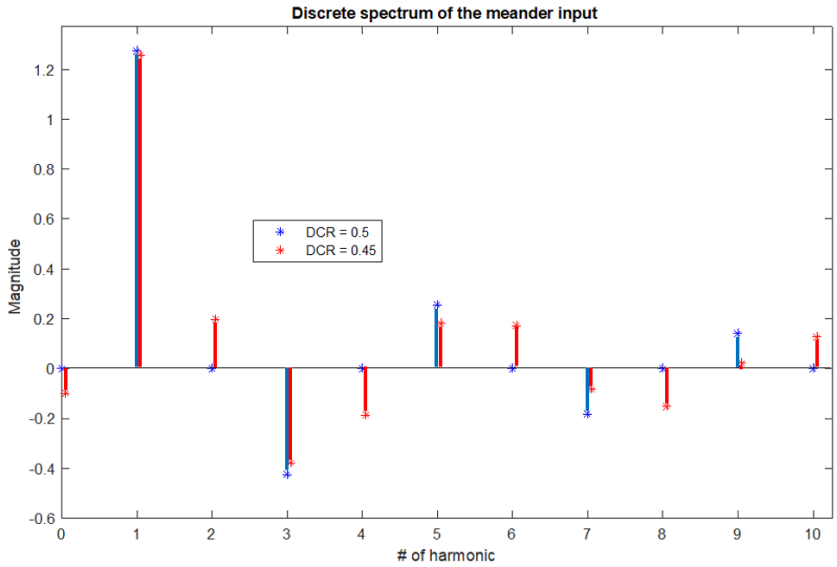


Fig. 5 The spectrum of the ideal meander with the DCR = 0.5 (blue), and distorted meander with the DCR = 0.45 (red). The ideal meander has no even harmonics nor DC in its spectrum. The distorted meander input has non-zero DC and all harmonics.

The output spectrum can be found by multiplying the magnitudes of the input harmonics with the channel transfer function at the corresponding frequencies. The transfer characteristic of a lossy channel is a complex valued function, so the resulted harmonics are complex as well. For convenience, **Figure 6** shows their absolute values.

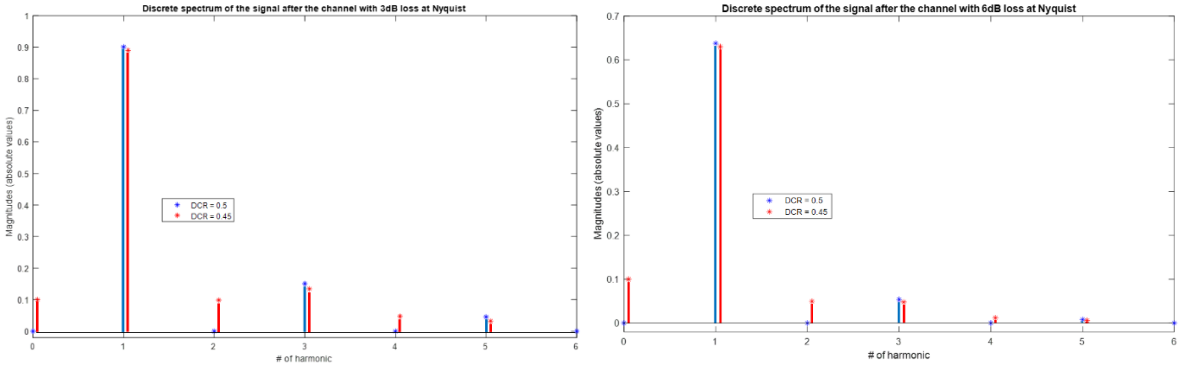


Fig. 6 Output signal spectrum. Channel with 3 dB loss (a) and 6 dB loss at the Nyquist frequency (b).

Figure 6 provides enough data to explain why the DCR (or jitter) is increased by a lossy channel. Note that duty cycle distortion creates a non-zero DC component in the input signal. This can be seen in Figures 5 and 6. The variable part of the signal is defined by its harmonics starting from the first and above. The lossy channel attenuates the signal at higher frequencies but does not significantly affect the DC component.

For example, in the input meander the ratio of the magnitude of the first harmonic to DC is almost 13, but it reduces to 9 and 6.5 respectively by the channels in Figures 6 a and b. Higher harmonics are attenuated more because the channel loss increases with frequency. If higher harmonics are neglected, the output signal becomes a sine function offset by the DC value. No zero crossing is possible if the magnitude of sine wave is below the DC value, but duty cycle distortion can be enormously large even if the DC value is exceeded by just a small amount.

In a nutshell, “jitter amplification” happens in two stages. First, input timing jitter (in this case, duty cycle distortion) is converted into “vertical noise,” which for the given example is a non-zero DC offset. Then, at the output of the channel this DC offset moves a considerably

attenuated “signal” up or down and converts it into a much larger timing variation (output jitter). Both conversions (timing variations into vertical noise and back) are non-linear transformations. Figure 4 illustrates this.

In some cases, linearization around a chosen operating point is possible, which provides some insight into the nature of the jitter transfer function. However, in general, such linearized models suffer from inaccuracy and, if applied incorrectly, may lead to large inaccuracies.

Explaining Jitter Amplification in the Time Domain

The previous analysis, based on Fourier series and attenuation of harmonics appears simple and convincing; however, it only applies to a periodic digital input impaired by a periodic data-dependent jitter. Any irregularity, either in the input pattern, jitter, or crosstalk makes the output waveform non-periodic and requires a different approach. In such cases a proper computational vehicle is the following time domain expression:

$$y(t+\eta_{RX}) = \sum_{k=-\infty}^{n} (b_k - b_{k-1}) S(t - kT_b + \eta_{TX,k}) \qquad (4)$$

The sum in Equation (4) is a convolution of a digital non-return-to-zero (NRZ) input defined by symbol values $b_k \in \{+1, -1\}$ with the channel’s step response. Any change in the input level from b_{k-1} to b_k contributes to the step response times the increment $(b_k - b_{k-1})$, causing the transition. $\eta_{TX,k}$ is a phase jitter at Tx that corresponds to this moment. A positive value of $\eta_{TX,k}$ means that this transition happened early.

For a given moment of observation, t, the transitions from an increasingly distant past (those with smaller k) add to the samples of step response $S(t-kT_b+\eta_{TX,k})$ taken at increasingly later times, because the "tail" of the response caused by these transitions is observed. For now, assume no jitter at the Rx end, i.e. $\eta_{RX}=0$. If necessary, the effect of Rx jitter can be added after the eye diagram is built, by convolving it horizontally with the Rx jitter distribution.

The task is to study the effect of duty cycle distortion, a periodic Tx jitter, on the clock signal, which is a periodic input pattern. Since the pattern and Tx jitter are periodic, so should be the waveform described by Equation (4). Therefore, it should be enough to find the values $y(\tau)$ when parameter τ spans one period (or two unit intervals). The logic level in the meander pattern changes once per symbol time T_b. Hence the multipliers (b_k-b_{k-1}) in Equation (4) make the alternating sequence $+2,-2\ldots$. Jitter values alternate, too, making the sequence $-\delta,+\delta\ldots$. In this case (per Figure 1), the rising edge is delayed, but the falling edge comes early. Therefore, the positive multiplier $+2$ must be accompanied by the jitter value δ with a minus sign. Inversely, the terms with a negative multiplier have δ with a plus sign. Then, Equation (4) becomes

$$y(\tau)=\sum_{k=-\infty}^{2}(-1)^{k+1}2S(\tau-kT_b+(-1)^k\delta) \tag{5}$$

In which the phase time τ changes from 0 to $2T_b$.

To further simplify the explanation, the order of summation is reversed representing Equation (5) as a sum of single "cursors" p_k associated with particular samples of the step response:

$$y(\tau) = -2S(\tau - 2T_b + \delta) + 2S(\tau - T_b - \delta) - 2S(\tau + \delta) + 2S(\tau + T_b - \delta) - \ldots$$
$$+2S(\tau + N_{\max}T_b + \delta) - S_\infty = -[S(\tau - 2T_b + \delta) - 0] + [S(\tau - T_b - \delta) - S(\tau - 2T_b + \delta)] -$$
$$[S(\tau + \delta) - S(\tau - T_b - \delta)] + [S(\tau + T_b - \delta) - S(\tau + \delta)] \ldots \quad (6)$$
$$+[S(\tau + N_{\max}T_b + \delta) - S_\infty] = -p_{-2} + p_{-1} - p_0 + p_1 - p_2 + \ldots$$

The step response $S(t)$ of a causal system equals zero for all $t < 0$. For sufficiently large time, after $N_{\max}$ unit intervals, it approaches a constant value. This means that the number of significant summands in Equation (6) is limited because the values p_k with small and large indices disappear.

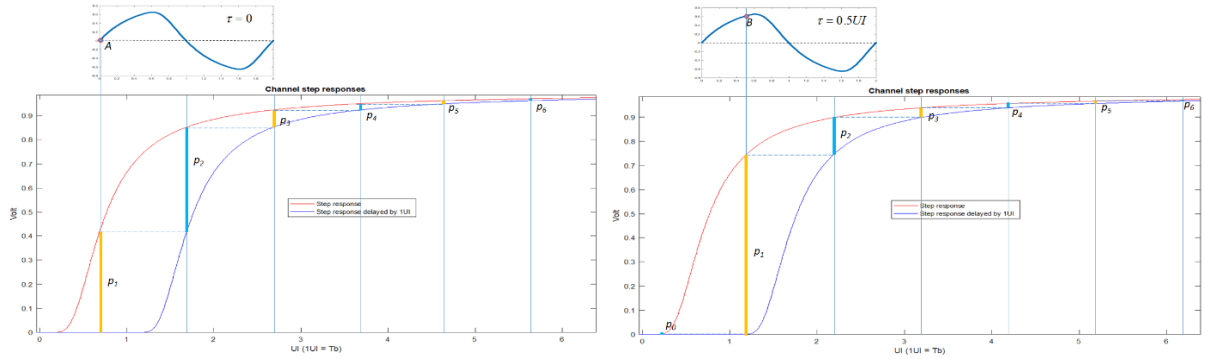
Building the Output Waveform without Tx jitter (DCR = 0.5)

First solve Equation (6) when $\delta = 0$, i.e. without Tx jitter. The method considered here can be used at any observation time, τ, but is shown in detail only for $\tau = 0$ and $\tau = 0.5\ UI$ with an explanation of how the results can be extrapolated. **Figure 7a** is built for $\tau = 0$, which corresponds to point A of the periodic output waveform where it first crosses the median level. The "cursors" p_k are vertical segments showing the difference between the samples of the step response and its copy delayed by 1 UI. The cursors to the left of p_1 and to the right of p_{10} can be neglected. The orange cursors are assigned a '+' sign and the blue ones a '-' sign. For this choice of phase τ, the sum in Equation (6) is zero. The result is formally expressed as:

$$y(0) = -p_{-2} + p_{-1} - p_0 + p_1 - p_2 + \ldots = 0 \quad (7)$$

In **Figure 7b**, the cursors are shifted forward by 0.5 UI and the sum in Equation (6) is a positive value corresponding to point B on the output waveform:

$$y(T_b/2) = -p_{-2} + p_{-1} - p_0 + p_1 - p_2 + \ldots > 0 \qquad (8)$$



(a) (b)

Fig. 7 Cursors p_k defined for the phase $\tau = 0$ creating a zero output value at point A (a) and cursors defined for $\tau = 0.5UI$ with the sum of cursors now positive at point B (b).

What happens with phase τ equal to 1, 1.5 and 2.0 UIs. With $\tau = 1UI$ all cursors from Figure 7a move right by 1 UI. The new cursor p_0 appears, taking the place of p_1, p_1 takes place of p_2 and so on. The sum of the cursors becomes zero, because it contains the same summands as for $\tau = 0$, but taken with opposite signs:

$$y(T_b) = -y(0) = +p_{-2} - p_{-1} + p_0 - p_1 + p_2 - \ldots = 0 \qquad (9)$$

With a 1.5 UI shift, the picture mostly repeats what is shown in Figure 5b for $\tau = 0.5\,UI$ except that the cursors with same indices (and signs) are moved 1 UI up and the new cursor p_0 appears on the left. Again, both cases use the same summands, but the values are taken with opposite signs. Instead of the positive waveform value at point B, it is negative with the same magnitude.

$$y(1.5T_b) = -y(0.5T_b) = +p_{-2} - p_{-1} + p_0 - p_1 + p_2 - \ldots < 0 \qquad (10)$$

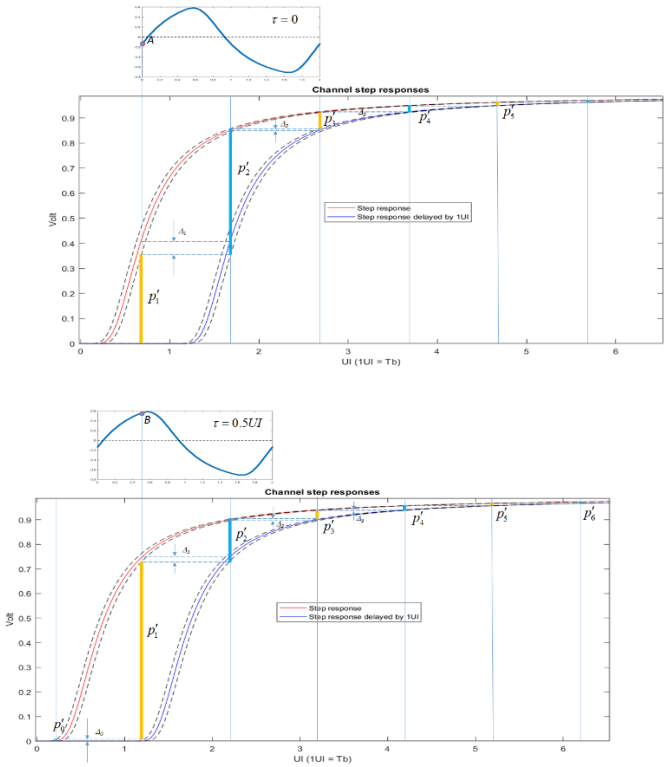
After one period, the same result is obtained as with $\tau = 0$. The new cursors p_{-1} and p_0 replace p_1 and p_2 while preserving the signs, and pushing others higher by 2 UI, repeating Figure 7a with incremented indices.

Output Waveform with Tx DCD (DCR = 0.45)

Let the input meander be affected by duty cycle distortion (DCD). In agreement with Figure 1, assume that all rising transitions are delayed by time δ, a certain fraction of the UI, and all falling ones come early by the same amount of time. This means that all "positive" (orange) cursors should be the difference between the sample of the rising step response delayed by δ and the sample of the next falling transition which comes ahead of time by δ. Likewise, all "negative" (blue) cursors should be measured between the sample of the leading (falling) transition, which comes early, and the next (rising) transition, which comes late.

Figures 8a and b illustrate how to find the value of the waveform at points A and B. Assume that DRC = 0.45, which makes $\delta = \frac{0.5-0.45}{T_b} = 0.05\ UI$. For each of the two major step responses, red and blue, their twins are added, shifted by $\pm\delta$ in both directions. The latter are shown by dash lines.

Take the cursor p_1' from Figure 8a. Its upper limit is now defined by the delayed step response, therefore it occurs lower by Δ_1 than it would without DCD. The next (blue) transition has not materialized yet, otherwise the cursor p_1' would have been offset from the bottom, as well. Next comes the cursor p_2'. Compared to its non-jittered version p_2 in Figure 7a, it adds to its length Δ_1 at the bottom and Δ_2 at the top. The next positive cursor p_3' is trimmed from the bottom by Δ_2 and from the top by Δ_3 and so on.



(a) (b)

Fig. 8 Cursors p_k defined for the phase $\tau = 0$ create point A of the output waveform (a). With the DCR = 0.45 this value is negative. Cursors defined for $\tau = 0.5$ UI and the corresponding point B on the output waveform (b).

The output value at $\tau = 0$ can be defined as follows:

$$y'(0) = \ldots - p'_{-2} + p'_{-1} - p'_0 + p'_1 - p'_2 + \ldots = \ldots - (p_{-2} + \Delta_{-3} + \Delta_{-2}) + (p_{-1} - \Delta_{-2} - \Delta_{-1}) - \ldots$$
$$(p_0 + \Delta_{-1} + \Delta_0) + (p_1 - \Delta_{-0} - \Delta_1) - (p_2 + \Delta_1 + \Delta_2) + \ldots = \quad (11)$$
$$\ldots - p_{-2} + p_{-1} - p_0 + p_1 - p_2 + \ldots - (2\Delta_{-3} + 2\Delta_{-2} + 2\Delta_{-1} + 2\Delta_0 + 2\Delta_1 + \ldots) = y(0) - \sum |2\Delta_n|$$

The same approach works for any τ, although the values of particular summands in (11) could be different. It is interesting that, regardless of τ, all "deltas" add up and create a negative offset $\sum|2\Delta_n|$ in the output value at $\tau = 0$. One can verify this by comparing Equation (11) to Equation (7). This vertical offset moves the waveform zero crossing point to the left from the point A, thus creating TIE or jitter at the channel's output.

The sign of the vertical offset changes if the DCR becomes larger than 0.5. This changes this sign of edge delays: the rising transition will happen ahead of time and the falling – later. As a result, the positive cursors will gain and negative decrease, thus adding a positive vertical offset to the waveform. This result agrees with expectation: the waveform with the DCR = 0.55 should be a mirror copy of the one with the DCR = 0.45.

What happens if channel loss increases? The step response becomes longer, the number of the meaningful summands p_n and Δ_n in Equation (11) increases while their values get smaller. This further diminishes the sum of alternating terms (i.e. signal magnitude), but will not affect much the sum of increasingly many small deltas, all of the same sign. In the end, there will be much larger distortion at the channel's output, seen as "jitter amplification."

We conclude this section by reiterating our observation about jitter/noise/jitter transformation. As seen from Figures 8 a and b, timing variation of the transition time shifts the channel's step response right or left and creates vertical increments denoted by Δ_n. Since jitter values are correlated with input bit polarity, it occurs that all these noise-related deltas have the same sign and create a considerable vertical offset. In contrast, the "useful signal" is the sum of the alternating terms which partially compensate each other.

Will the eye diagram change if the same amount of jitter is applied at the Rx side?

Yes, the eye diagram of the output signal becomes quite different if the same amount of DCD is applied to the sampler clock instead of the digital input pattern. In this case jitter amplification is not seen because it does not originate at Tx in the first place. Also, the signal propagating through the lossy channel would not have a DC component and so the output waveform would not "sink" under the sampling threshold. With jitter applied at the Rx side, the resulted eye diagram can be found by taking the one without jitter (see Figure 3b) and summing two copies of it shifted to the right and left by the timing error δ.

Can a lossy channel "reduce" the amount of jitter?

Although it sounds counter-intuitive, this is possible, as well, but in very special cases of data-dependent jitter. For example, assume that a periodic pattern '001001001…' is applied to the lossy channel, and the rising edge always comes early while the falling edge is always late. For this type of input jitter, the length of the positive symbols becomes larger than 1 UI. However, as previously seen, the lossy channel reduces all spectral components with respect to DC. For the pattern with negative imparity (where the number of logical zeros dominates) the DC component is negative. At the channel's output, the waveform sinks deeper, thus reducing the time it remains in positive territory. With the right amount of loss, jitter could be reduced or even eliminated.

JITTER AMPLIFICATION IN GENERAL LOSSY AND REFLECTIVE CHANNELS, SMALL MAGNITUDE CASE.

https://www.signalintegrityjournal.com/articles/2081-how-transmit-jitter-propagates-through-the-channel-and-what-is-jitter-amplification

A clock signal with DCD is perfectly deterministic, which makes its analysis very simple. However, when input symbols are irregular and input phase jitter is random, statistical methods and other instruments must be employed.

Assume an eye diagram for a channel whose step response has a certain finite duration, up to $N_{\max}$ unit intervals, after which it becomes perfectly flat. Each trajectory is defined by a unique combination of symbol values preceding the observation interval. For a NRZ channel without jitter, the eye diagram can be composed of up to $2^{N_{\max}}$ different trajectories or filaments. Each trajectory is a solid line of 1 UI length, one of many creating the "body" of the eye diagram. Usually, the number of such trajectories is very large, and, even without jitter, they merge into a fuzzy cloud. Assume a high resolution picture, however, in which individual trajectories can be distinguished in the vicinity of the median level, and this is not the case in the absence of jitter. (see **Figure 9a**).

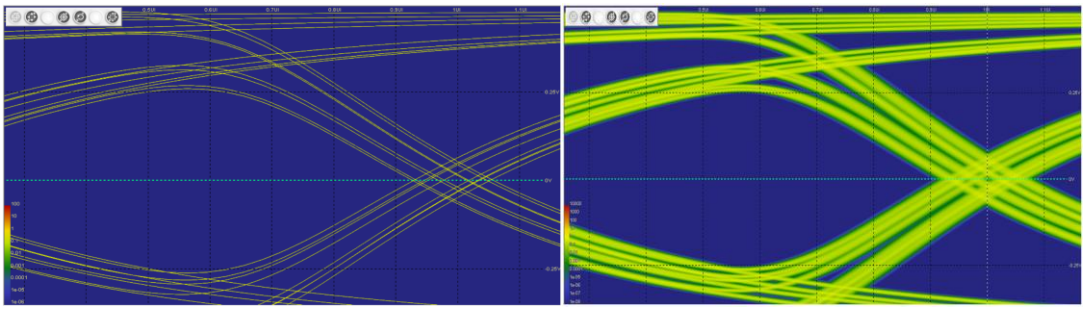


Fig. 9 Trajectories creating an eye diagram in absence of jitter (a) and after adding a small amount of Gaussian uncorrelated Tx jitter.

Introduce time phase as a modulus of time divided by the symbol interval, or $\tau = mod\ (t, T_b)$, with $\tau \in [0, T_b]$. Then, Equation (4) becomes:

$$y(\tau) = \sum_{k=-\infty}^{n} (b_k - b_{k-1}) S(\tau - kT_b + \eta_{TX,k}) \quad (12)$$

where $\eta_{TX,k}$ is small input phase jitter. As previously mentioned, the number of meaningful summands in Equation (12) is equal to the length of the channel step response expressed in symbol intervals. The factors $(b_k - b_{k-1})$ are increments of the digital input signal and could be –2, 0, or +2. If we ignore cases with $(b_k - b_{k-1}) = 0$ (no transition), the rest must be alternating, as do rising or falling transitions in the NRZ input pattern. All possible combinations of the weight factors in Equation (12) create the multiplicity of trajectories that make up the eye diagram. The slope of each trajectory can be found from:

$$\frac{y(\tau)}{d\tau} = \sum_{k=-\infty}^{n} (b_k - b_{k-1}) \frac{S(\tau - kT_b)}{d\tau} = \sum_{k=-\infty}^{n} (b_k - b_{k-1}) H(\tau - kT_b) \quad (13)$$

where $H(t)$ is the channel's impulse response, a derivative of the step response.

Now add a small amount of random uncorrelated Gaussian jitter $\eta_{TX,k}$ with standard deviation σ. The trajectories in the eye diagram become fuzzy as in **Figure 9b**. It appears that different trajectories acquire an unequal amount of vertical noise, and even a single trajectory seems to change its fuzziness as it progresses in time.

This can be explained as follows. Small random jitter at Tx causes a random variation of the moment when the corresponding step transition starts; which, when observed at a later time $t = \tau - kT_b > 0$, creates random vertical noise. If jitter is small enough, the observed random noise is also Gaussian, and its standard deviation can be found as a product:

$$\sigma \frac{S(\tau - kT_b)}{d\tau} = \sigma H(\tau - kT_b) \tag{14}$$

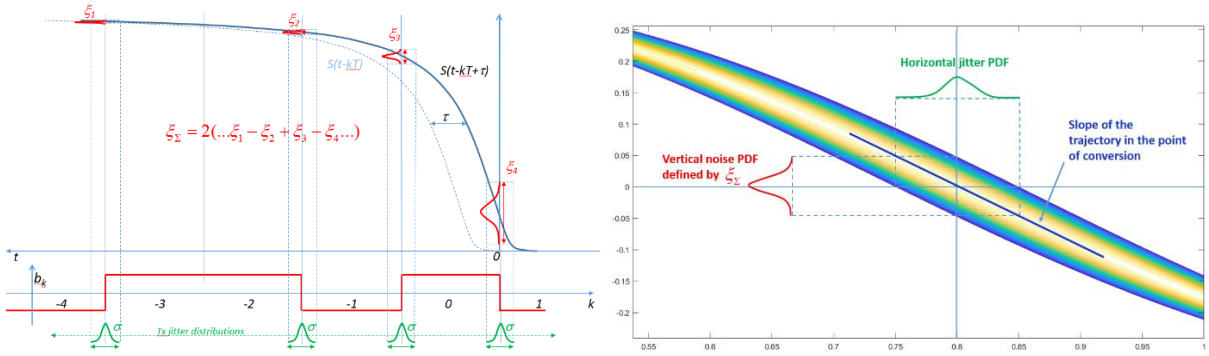
When a particular noisy trajectory is observed at a given phase τ, its vertical cross-section is a probability distribution of noise ξ_Σ which is a sum of the noise components ξ_k originating from several preceding transitions, as illustrated by **Figure 10a**. If the input jitter is uncorrelated, the variance of the vertical noise ξ_Σ around each trajectory can be found as the sum of the variances of all noise components:

$$\sigma^2_{noise} = \sum_{k=-\infty}^{n} [(b_k - b_{k-1})\sigma H(\tau - kT_b)]^2 \tag{15}$$

This expression results in the "transfer function" of the input jitter to vertical noise ξ_Σ observed at phase τ of the trajectory:

$$K_{TxJitToNoise} = \sqrt{\sum_{k=-\infty}^{n} [(b_k - b_{k-1})H(\tau - kT_b)]^2} \tag{16}$$

Note that σ_{noise} and $K_{TxJitToNoise}$ are functions of the phase τ and factors $(b_k - b_{k-1})$ that define the sign and positions of transitions creating a particular trajectory. The first makes the noise dependent on exact position within each UI and the second explains why different trajectories show different amounts of vertical noise.



https://www.signalintegrityjournal.com/articles/2081-how-transmit-jitter-propagates-through-the-channel-and-what-is-jitter-amplification

Fig. 10 Input jitter converts into vertical noise and then back into output jitter. Uncorrelated jitter creates vertical noise components (a); the sum vertical noise makes the trajectory fuzzy and converts back into horizontal output jitter (b).

To determine how vertical noise converts back into output jitter, consider the vicinity of the point where a particular noisy trajectory crosses the threshold level (see **Figure 10b**). Vertical noise converts back into timing variations by a factor that is inversely proportional to the slope of the trajectory defined by Equation (13):

$$K_{NoiseToRxJit} = 1 / \left| \sum_{k=-\infty}^{n} (b_k - b_{k-1}) H(\tau - kT_b) \right| \tag{17}$$

For convenience, vector $\chi = \{b_k\}$ is introduced, which defines the digital input creating a certain trajectory. Then, jitter-to-noise and noise-to-jitter transfer functions that correspond to a particular point at a given trajectory become $K_{TxJitToNoise}(\chi, \tau)$ and $K_{NoiseToRxJit}(\chi, \tau)$. The jitter amplification factor found for a particular point τ on the selected trajectory is defined as the product:

$$K_{JitAmpl}(\chi, \tau) = K_{TxJitToNoise}(\chi, \tau) K_{NoiseToRxJit}(\chi, \tau) \tag{18}$$

Although the factors $K_{TxJitToNoise}$ and $K_{NoiseToRxJit}$ are composed from the same set of coefficients $\chi = \{b_k\}$ and employ the same values of the channel's impulse response $H(\tau - kT_b)$, it is hard to predict how their product behaves. For that reason, Monte Carlo analysis is conducted, in which

a large ensemble of step responses is created by combining a number of normalized step responses from lossy channels and low-loss channels with reflections.

The first group creates monotonic step responses for which the samples of impulse response H_k remain non-negative. In the second group, the values of the impulse responses can have different signs. Simulation of input patterns is reduced by selecting the samples of H_k alternatively with different signs or skipping some of them, governed by a pseudo random bit sequence (PRBS5) started from different positions.

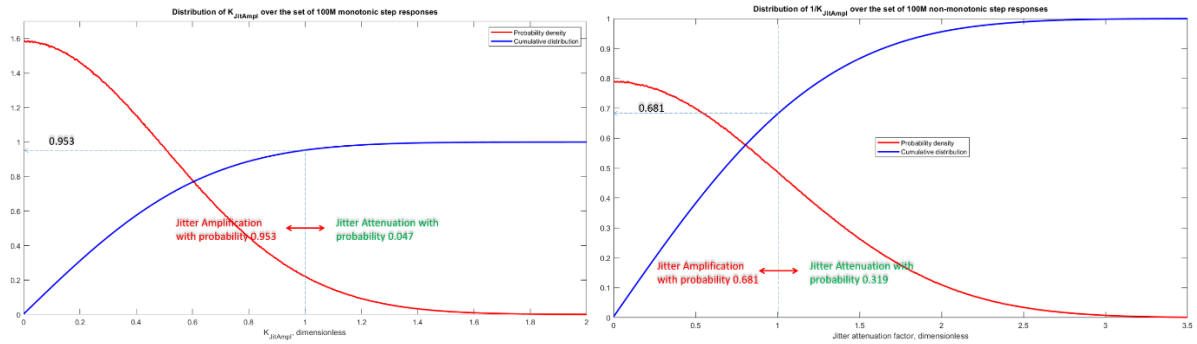


Fig. 11 Probability density (red) and cumulative distribution of "jitter attenuation factor" (blue) found by Monte Carlo analysis over an ensemble of 100 M step responses. Lossy channels, monotonic step responses (a); reflective channels, non-monotonic step responses (b). For cumulative distribution the vertical coordinate is probability; for probability density it is the derivative of probability by the "jitter attenuation" factor.

The histogram of the *jitter amplification factor* $K_{JitAmpl}$ does not converge well enough to define a meaningful probability density function (PDF) and cumulative distribution function (CDF) function. The reason is the unbounded nature of this parameter. However, *jitter*

attenuation $K_{JitAttn}$, inverse to amplification, behaves quite well for both types of channel step responses.

It is possible to build a very smooth histogram (approximate PDF), then integrate it to find the corresponding CDF (see **Figure 11**). Since $K_{JitAttn} < 1$ means jitter amplification, the largest number of cases fall into this category. However, for lossy channels with monotonic step responses, the probability that a particular trajectory will demonstrate jitter amplification is 0.953, and it is less probable for reflective channels (only 0.681).

One scenario that causes large jitter amplification is the case where the slope of the step response in the data transition points (see Figure 10a) is approximately equal. Since transitions have opposite directions, the summands in Equation (13) come with opposite signs, which makes the sum, and hence the slope of the trajectory in the observation point small. The factor $K_{NoiseToRxJit}$, an inverse to the slope, becomes large. However, the summands in Equation (16) originating from uncorrelated jitter values do not cancel each other, thus creating large factor $K_{TxJitToNoise}$. Taken together, the two factors make jitter amplification large.

Inversely, jitter attenuation happens when the derivatives of the step response taken at transition points alternate but remain close by absolute value. In this case, taken with opposite signs, they sum up in concert and make the slope of the trajectory large which reduces the factor $K_{NoiseToRxJit}$.

While separate trajectories may demonstrate jitter amplification or attenuation, in the eye diagram comprising increasingly many trajectories, the general trend is jitter amplification. One reason is the higher probability for any trajectory to amplify input jitter rather than reduce it.

Second, even with an equal number of the trajectories increasing or decreasing jitter, one does not diminish the effect of the other.

WHY JITTER EVALUATION BASED ON THE SYMBOL (PULSE) RESPONSE MIGHT BE INACCURATE

So far, evaluation of the effect of transmit jitter and the associated vertical noise has been based on Equations (4), (5) and (12) that describe the output waveform as a superposition of transition edges (or channel step responses). But why not use the symbol responses $P(t)$ instead? A symbol (or pulse) response is the response of an undisturbed channel to the isolated input symbol. It equals the difference between the step response and its copy delayed by one symbol interval, i.e. $P(t) = S(t) - S(t - T_b)$. Using this notation, Equation (12) can be rewritten as:

$$y(\tau) = \sum_{k=-\infty}^{n} b_k P(\tau - kT_b) \tag{19}$$

Many channel evaluation tools rely on this formula. In particular, all COM specifications, from Mellitz et al.[5] to the most recent ones, use Equation (19) not only to determine intersymbol interference (ISI) and crosstalk noise, but also to evaluate the effect of transmit jitter. They define the small-signal jitter-to-noise transfer factor by taking the time derivative of (19):

$$\frac{y(\tau)}{d\tau} = \sum_{k=-\infty}^{n} b_k \frac{dP(\tau - kT_b)}{d\tau} \tag{20}$$

One problem with Equations (19) and (20) is an assumption that the shape of the symbol response $P(t)$ is unaffected by jitter; however, uncorrelated phase jitter transforms into period jitter thus changing the distance between adjacent transitions and therefore the shape of the "symbol response." Hence, "modification" of individual pulse responses due to input jitter is itself the mechanism that explains its conversion into vertical noise, and therefore should not be ignored.

The second problem is that, according to these equations, input jitter creates vertical noise, even when input logical levels do not change and the channel is silent. Compare this for example to Equation (12), where summands disappear if logical levels are identical.

Take, for example, a lossy channel with a NRZ pattern in the presence of small Gaussian uncorrelated input jitter having standard deviation σ. Its step responses (original and delayed), together with the symbol or pulse response is shown in **Figure 12**.

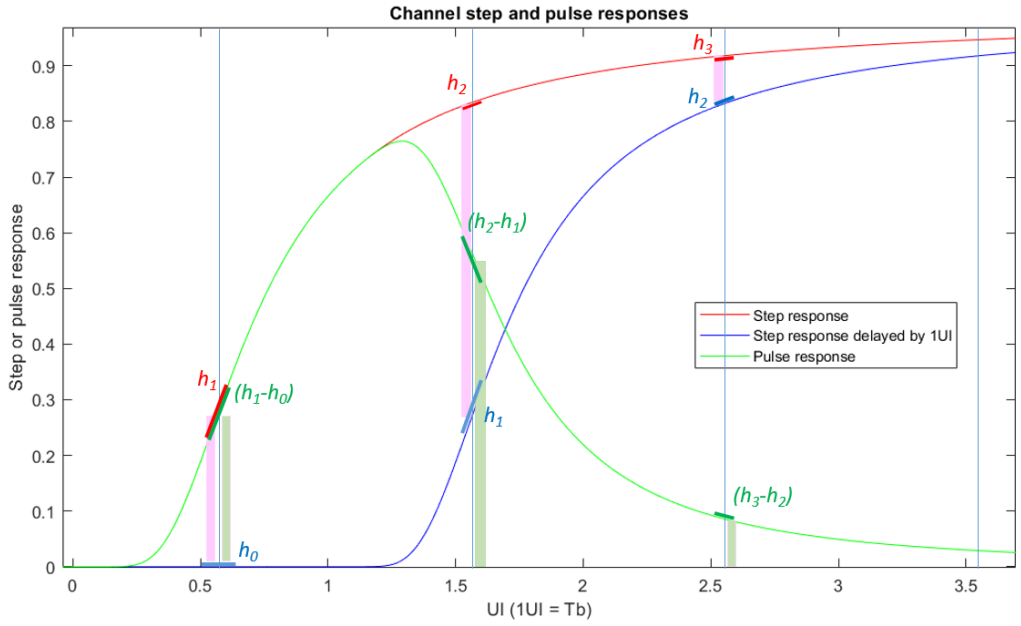


Fig. 12 Channel step response (red), its copy delayed by 1 UI (blue) and the symbol response (green). The input jitter-to-noise transformation is proportional to the steepness of the response curves.

Consider the right side of the eye diagram, therefore the sampling phase is chosen slightly to the left of point **A** shown Figure 5a; at 0.55 UI versus 0.65 UI. The values of the step and pulse responses at points designated by thin vertical lines contribute to the value of the waveform. More precisely, when finding the value of the waveform by Equation (12), the heights of the pink bars are summed, taken with alternating signs or skipped. Every such value is the difference between the samples of the step response and its delayed copy.

Why the difference between the two? Each symbol value b_k participates in two summands of Equation (12) where it has opposite signs. Equation (19) yields the sum of the pulse response cursors (green bars), also taken with alternating signs. According to the definition of the pulse response, all the summands in the first and second case are equal, and so are the waveform values. However, it does not work the same way in presence of random jitter.

Assume that the last transition happens 1 UI prior to the observation time. This corresponds to the value of the step response shown in red (see Figure 12) at 1.6 UI. Due to input jitter, the starting point of the response experiences small random variations. The same horizontal variations observed at 1.6 UI translate into vertical noise with RMS $\sigma|h_2|$, where h_2 is time derivative of the step response at 1.6 UI. However, according to Equation (12), this value should be taken with the factor 2 because of the difference $|b_k - b_{k-1}|$ between unequal NRZ symbol

values. As a result, the contribution to the variance of the vertical noise from this transition becomes $\left(2\sigma|h_2|\right)^2 = 4\sigma^2 h_2^2$.

When using pulse response representations, Equation (20), the corresponding contribution into vertical is defined by the slope of the pulse response, which is $|h_2 - h_1|$ thus adding $\sigma^2 \left(h_2 - h_1\right)^2$ to its variance.

Assuming that digital input is a clock meander, the total noise variance found by the two methods becomes $V_{noiseS}(\tau) = 4\sigma^2 \sum_{k=-\infty}^{n} h_k^2$ and $V_{noiseP}(\tau) = \sigma^2 \sum_{k=-\infty}^{n} (h_k - h_{k-1})^2$ respectively. It turns out that in most cases – not necessary with meander input – the step response-based formula gives a larger sum variation, which then translates into a larger output jitter. More detailed analysis can be found in the work of Dmitriev-Zdorov et al.,[6] showing that for a lossy channel with a monotonic step response, the method based on pulse response, Equations (19) and (20), underestimates the effect of input jitter.

THE GENERAL CASE: LARGE JITTER, NON-GAUSSIAN DISTRIBUTION, COMBINATION OF RANDOM AND DETERMINISTIC JITTER

How is the eye diagram or bit error rate (BER) determined in more general cases, when jitter is non-Gaussian, or combines different jitter types, and its magnitude is not small such that response linearization cannot be applied? One obvious solution is to run a long time domain

simulation based of Equation (4) or its time-wrapped version, Equation (12), and accumulate as much statistics as possible. Unfortunately, this approach is not computationally feasible considering that BER values as low as 1e-15 are often of interest, which would necessitate running tera- to peta-symbols.

Statistical analysis may be a good alternative, even though its ability to account for a channel's non-linearity is limited. Statistical analysis can be based on Equation (4), as well, but with some assumptions. Let the input pattern be stationary (i.e. its statistical properties do not change over time), with jitter random stationary as well. Then, the channel's output becomes a random value whose probability distribution is periodic in time phase $\tau \in (0, T_b)$, with period T_b:

$$y(\tau + \eta_{RX,k}) = \sum_{k=-\infty}^{\infty} (b_k - b_{k-1}) S(\tau - kT_b + \eta_{TX,k}) \tag{21}$$

The task is to build a probability distribution of $y(\tau + \eta_{RX,k})$, for sufficiently many phase values covering one period, e.g. from tens to hundreds, considering given statistical properties of the pattern magnitudes b_k, and Tx jitter samples $\eta_{TX,k}$. Typically, at this stage $\eta_{RX,k}$ can be added later when the entire eye diagram is built, by convolving the eye horizontally with probability distribution of $\eta_{RX,k}$.

The approach, summarized here, is described in detail by Dmitriev-Zdorov.[7] For a given phase τ the distribution of the output value $y(\tau)$ is a convolution of distributions, or PDFs of all summands in Equation (21). As previously mentioned, the required number of summands does not need to exceed the duration of the non-stationary portion of the channel's step response

measured in symbol intervals. In the presence of random transmit jitter, the sampled value of the step response $V_{k,\tau} = S(\tau - kT_b + \eta_{TX,k})$ is random, too, with its vertical PDFs denoted as $v_{\tau,k}(\mu)$.

Each summand in Equation (21) is a product of the discrete random value $\beta_k = (b_k - b_{k-1})$ and $V_{k,\tau}$ which, depending on Tx jitter, may have a continuous or discrete distribution. Distribution of the product of the two random values $(b_k - b_{k-1})$ and $S(\tau - kT_b + \eta_{TX,k})$ is a convolution of their probability distributions. However, since the first is discrete, PDF $u_{\tau,k}(\mu)$ of the product $\beta_k V_{k,\tau}$ is a scaled version of $v_{\tau,k}(\mu)$. Notice that values β_k on subsequent steps are not independent, because they involve both current and previous symbols and therefore create a discrete-time Markov chain. This process can be described with a corresponding flow diagram (see **Figure 13**).

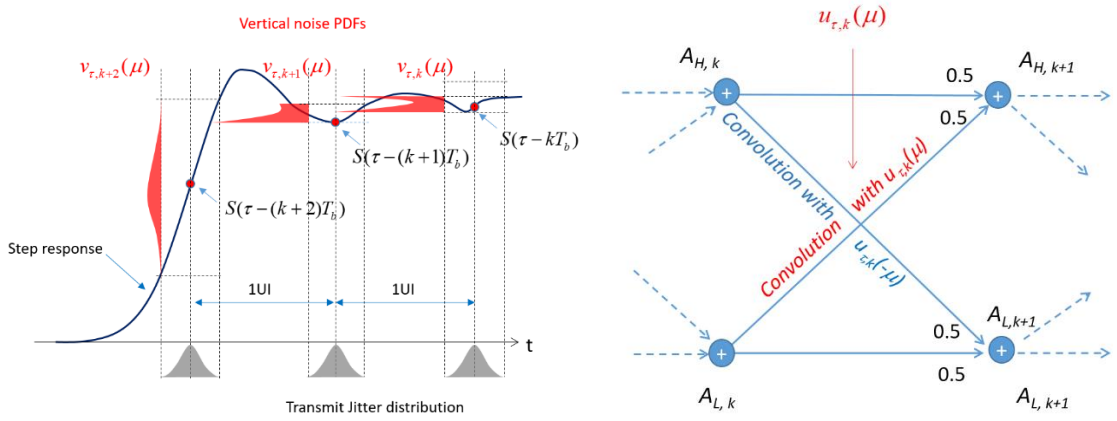


Fig. 13 Conversion of Tx jitter into vertical noise (large jitter case) (a). One step in the statistical analysis – adding k+1-th summand in Equation (21) (b).

Figure 13a illustrates how transmit jitter converts into vertical noise in general. Instead of linearization of the step response over a number of points, its shape must be considered, exactly.

Once the jitter PDF is known (not necessary Gaussian), it can be converted into noise PDFs $v_{\tau,k}(\mu)$ and its scaled versions $u_{\tau,k}(\mu)$. One step of the solution process is shown in Figure 13b. The nodes $A_{H,k}$ and $A_{L,k}$ designate two separate PDFs that are the "upper" and "lower" portions of the vertical cross-section of the eye diagram on step *k*.

This is for NRZ signaling, more than two may be needed if the number of signal levels is higher. Initially, PDFs contained in $A_{H,k}$ and $A_{L,k}$ are Dirac pulses corresponding to the value of the step response at sufficiently large time, but with opposite signs. At every step, the PDF of the vertical noise is evaluated, moving toward the beginning of the step response, as shown in Figure 13a. Then, a scaled version of this PDF is created that accounts for the factor 2, and its flipped copy. The PDF from $A_{L,k}$ is then convolved with $u_{\tau,k}(\mu)$ and the result is summed with the PDF stored in $A_{H,k}$ (both with weighting factor 0.5) and placed into $A_{H,k+1}$.

Similarly, the PDF from $A_{H,k}$ is convolved with $u_{\tau,k}(-\mu)$ and summed with the content of $A_{L,k}$, both halved, which makes $A_{L,k+1}$. The process repeats until the start of the step response is reached and PDFs stop changing. The process repeats for the next time phase value τ until the entire eye diagram is created, column by column.

The outlined framework is quite generic. With minor modifications it works for multilevel signaling in channels with asymmetrical transition waveforms for cases where input symbols are not uncorrelated. It can also be used to simulate a channel where input jitter contains random and deterministic components. For example to account for DCD jitter, it is enough to assume that the distance between adjacent step response samples in Figure 13a alternate.

To simulate sine jitter, this distance would be modulated by a sine function of a certain frequency, and so on. **Figure 14** illustrates some of these cases. Figures 14a and b illustrate eye diagrams built for a channel with edge rise/fall asymmetry using statistical and time domain simulation respectively. The statistical solution agrees well with the time domain eye, but gives better insight into low probability effects. Figures 14c and d show the results of a statistical simulation for a PAM-4 channel. The former assumes an identical shape for all transitions (save for magnitude scaling) and the latter considers unequal, individually measured transitions between different signal levels.

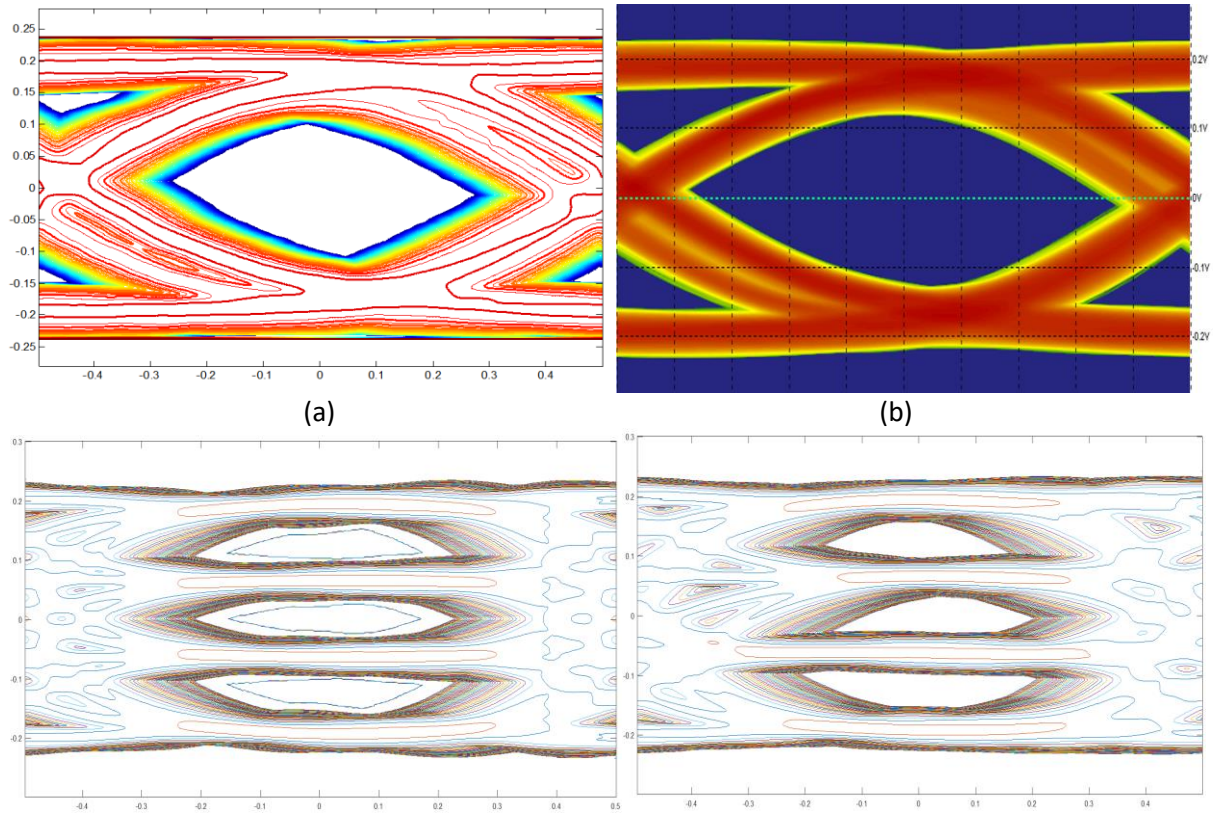


Fig. 14 Examples of statistical eyes determined by the above method. Statistical eye built for a NRZ channel with asymmetrical rise/fall edges and Gaussian input jitter. Eye contours cover

20 orders of magnitude (a). The same channel and input jitter, but the eye diagram is built in the time domain over 10 M symbols, without extrapolation to low probability (b). Note the wider eye opening in the second case. Statistical eye for a 4-level pulse amplitude modulation (PAM-4) multilevel signaling channel, all transition edges identical, no input jitter (c). The same channel with different transition edges and Gaussian jitter added (d).

What about jitter amplification when Tx jitter is not "small?" As seen from Figure 13a, the vertical noise is always bounded, and cannot exceed the magnitude of the step response; however, even half of it is enough to close the eye completely. The general trend is the same, by increasing input jitter, an even faster increase of output jitter is observed, and this seems to work for large jitter magnitudes as well. Analysis is considerably complicated, however, with jitter noise having arbitrary non-Gaussian distributions.

CONCLUSION

A jitter study that considers its conversion into vertical noise and propagation of this noise in the channel is intuitively clear and simple, universal, and physically sensible. It applies to many – if not all – linear systems. This phenomenon is illustrated in both the frequency and time domains for small and large magnitude jitter. The method of accurate statistical analysis is described based on this understanding.

Lossy reflective channels are expected to increase jitter over an ensemble of eye diagram trajectories and the transition edges the channels may produce. To improve the accuracy of serializer/deserializer (SERDES) compliance analysis/simulation, both time domain and

statistical simulations must be based on superimposing step (edge) responses. Pulse/symbol responses cannot adequately represent transmit jitter because they assume a fixed distance between two adjacent transitions.

References

1. C. Madden, S. Chang, D. Oh and C. Yuan, "Jitter Amplification Considerations for PCB Clock Channel Design," *IEEE 16th Topical Meeting on Electrical Performance of Electronic Packaging*, October 2007.
2. S. Chang, D. Oh and C. Madden, "Jitter Modeling in Statistical Link Simulation," *IEEE Electromagnetic Compatibility Symposium*, August 2008.
3. F. Rao, V. Borich, H. Abebe and M. Yang, "Rigorous Modeling of Transmit Jitter for Accurate and Efficient Statistical Eye Simulation," *DesignCon*, January 2010.
4. M. Miller, "Understanding Apparent Increasing Random Jitter with Increasing PRBS Test Pattern Lengths," *DesignCon*, January 2013.
5. R. Mellitz, A. Ran, M. Peng Li and V. Ragavassamy, "Channel Operating Margin (COM): Evolution of Channel Specifications for 25 Gbps and Beyond," *DesignCon*, January 2013.
6. V. Dmitriev-Zdorov, C. Filip, C. Ferry and A. Neves, "BER- and COM-Way of Channel Compliance Evaluation: What are the Sources of Differences?" *Signal Integrity Journal*, March 2017.

7. V. Dmitriev-Zdorov, “Accurate Statistical Analysis of SERDES Links Considering Correlated Input Patterns, Data-Dependent Edge Transitions and Transmit Jitter,” *DesignCon*, January 2015.