Using Clock Jitter Analysis to Reduce BER in Serial Data Applications

KEYSIGHT
TECHNOLOGIES
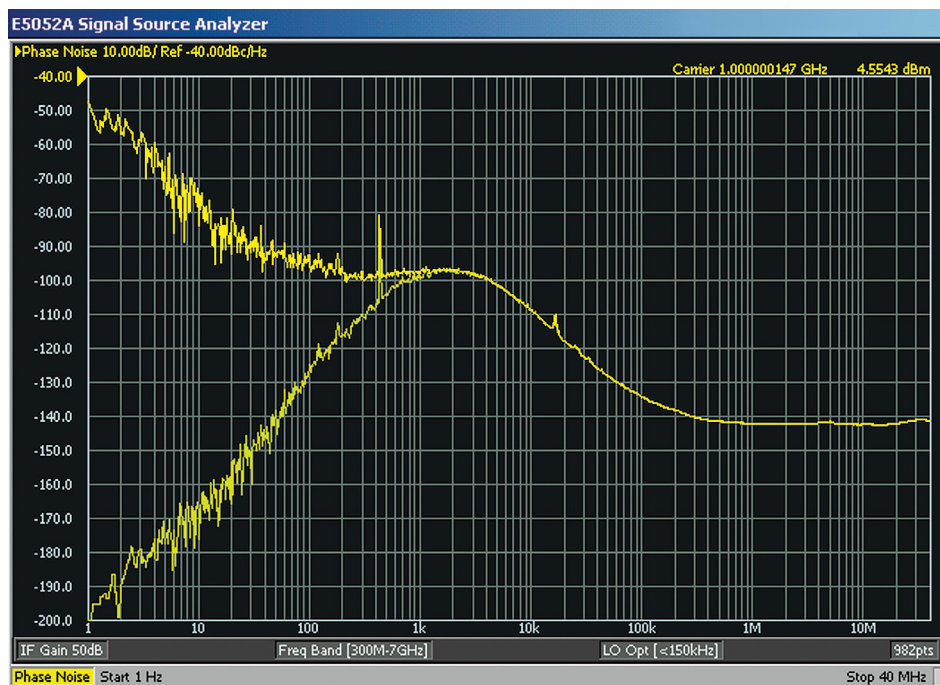
Introduction

The analysis of clock jitter has evolved as data rates have increased. In high-speed serial data links clock jitter affects data jitter at the transmitter, in the transmission line, and at the receiver. Measurements of clock quality assurance have also evolved. The emphasis is now on directly relating clock performance to system performance in terms of the bit error ratio (BER).

This application note discusses clock-jitter issues relevant to serial data systems including: an introduction to jitter-related problems in serial data applications, the role of reference clocks, and how their jitter affects the rest of a system. With the context and issues of reference clocks in place, a review of oscillator and phase noise sets the stage for the discussion of techniques for evaluating clock quality with emphasis on emerging techniques for compliance testing. A survey of jitter analysis equipment completes this note.

The effects of jitter in serial data applications

The NIST definition of jitter[1] is "the short term phase variation of the significant instants of a digital signal from their ideal positions in time. The term "jitter" is typically concerned with non-cumulative variations above 10 Hz. Cumulative phase variations below 10 Hz are usually defined as wander. In serial data applications, since the clock is embedded in the data and reconstructed at the receiver, you rarely need to bother with wander.
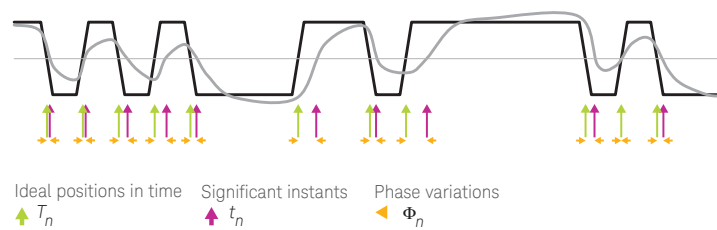

Ideal positions in time   Significant instants   Phase variations
T_n t_n Φ_n

Figure 1. The NIST definition of jitter is "the short term phase variation of the significant instants of a digital signal from their ideal positions in time

In figure 1, the smooth blue line represents the actual analog waveform, the black line represents the ideal digital waveform, and the straight gray line represents the slice-threshold of an ideal receiver. The "ideal positions in time", T_n, are those points where the ideal digital waveform crosses the slice-threshold. The "significant instances", t_n, are those points where the actual analog waveform crosses the slice-threshold. The jitter, or phase variations, T_n, are the difference between the two, $\Phi_n = t_n - T_n$.

Introduction to jitter

It's both natural and accurate to think of the "significant instants" as the logic transition times or edges.

Significant instants are logic transition times

t_1 t_n

$$S(t) = P\ (2\pi f_d t\ +\ \varphi(t))$$
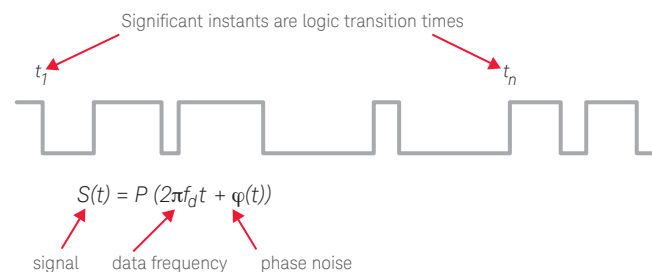
signal   data frequency   phase noise

Figure 2. A data signal showing the significant instances of a data signal.

It is convenient, though not entirely accurate, to think of the "ideal positions in time" as integer multiples of the bit period, T. The phase variations can then be written, $\Phi_n = t_n - nT$; this is also the definition of "phase jitter" which is also called "cumulative jitter" – despite all its names; it's just the jitter of each edge.
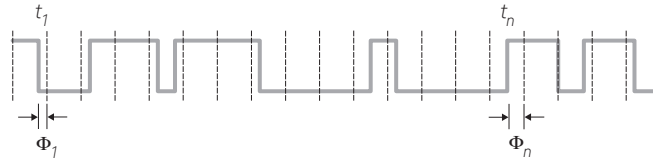

Figure 3. Phase jitter is the jitter at each edge

It is important to notice that jitter is a discrete quantity. If you write down a general function for a signal, $S(t)$, as a pulse-train of logic values, P, then its argument, $2\pi f_d + \phi(t)$, shows how jitter gets in the system. In an ideal system, each edge would be placed according to the data frequency of the signal, fd, and the phase noise term, $\phi(t)$, would be zero for all t. It is the phase noise term that causes jitter. Phase noise, $\phi(t)$, is a continuous function of time, but jitter (or phase jitter or cumulative jitter), Φ_n, is the amount of phase noise at crossing times.

Phase jitter can be written in terms of phase noise (in radians)

$$\Phi_n = \frac{\phi(t_n)}{2\pi f_d}$$

Why is jitter important?
Jitter is important for exactly the same reason as signal-to-noise ratio; a low SNR means a high bit error ratio. Voltage noise causes bit errors when the signal voltage fluctuates vertically across the logic-slice threshold. Similarly, jitter causes errors when the timing of a signal transition fluctuates horizontally across the sampling point. The sampling point is the point in voltage and time, (t, V) where the receiver determines whether a bit is a logic one or zero[2].

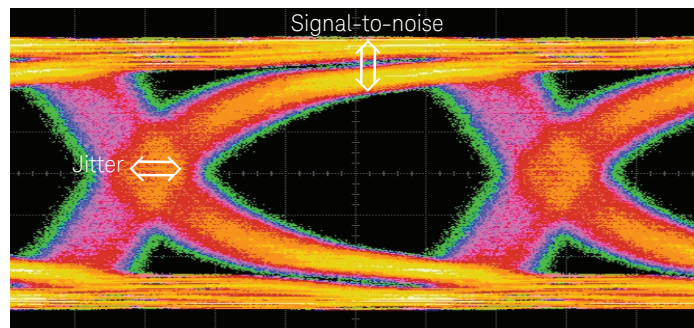The only reason to analyze jitter is to limit the BER[3].

Figure 4. Jitter causes errors when the signal fluctuates horizontally the same way that a low signal-to-noise ratio causes errors when the signal fluctuates vertically.

Jitter caused by phase noise
In a clock signal,

$v_{real}(t) = (v_0 + \Delta v(t))\ sin(\omega t + \phi(t))$,

you can distinguish amplitude noise and phase noise, or amplitude modulation (AM) and phase modulation (PM). Both AM and PM can cause jitter. The phase noise term, $\phi(t)$, translates the periodic function, in this case a simple sinusoid, horizontally and the amplitude noise term, $\Delta v(t)$, vertically.

Jitter caused by amplitude noise

Amplitude noise can also cause jitter. It's a second-order effect in the sense that, if a signal has zero rise/fall time then amplitude noise can't cause jitter. Of course, real signals have finite rise/fall time and, by moving the signal up or down, amplitude noise changes the times of logic transitions.

Figure 5. Example of how jitter can be caused by amplitude noise

Figure 5 shows how vertical translation introduces jitter by changing the crossing points of each edge. The crossing points of those edges with longer rise/fall times experience greater displacement than those with shorter rise/fall times.

In most cases, clock-jitter is dominated by phase noise, but it's important to keep in mind that noise is noise. Jitter and voltage noise, while simple to describe as separate functions, are correlated. It's possible for elements of the voltage noise term, Δv(t), to change the phase noise term, ϕ(t).

The jitter probability density function

The jitter probability density function (PDF) gives the probability for a given logic transition to differ from the ideal by certain amount (between Φ and $\Phi + d\Phi$. One way to measure the jitter PDF is to make a histogram of the crossing point. Each entry in the histogram is the time-position of a given edge with the time-resolution, or bin-width, of the histogram.

Different sources of jitter combine through the mathematical process of convolution[4] to form the resulting jitter PDF. Figure 6 shows the PDF for sinusoidal jitter.
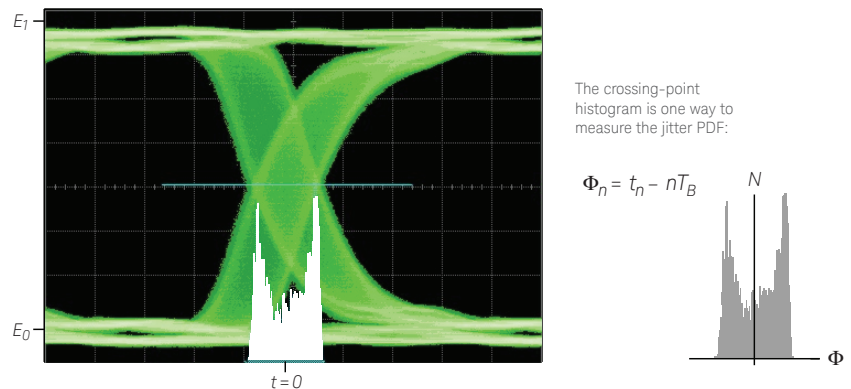

The crossing-point histogram is one way to measure the jitter PDF:

$$\Phi_n = t_n - nT_B$$

Figure 6. Probability density function (PDF) for sinusoidal jitter

Reference clocks

In figure 7, the four major components of the reference clock are represented. The transmitter usually serializes a set of lower rate parallel signals into a serial data stream. The transmission channel through which the signal propagates is a combination of backplanes and cables. The receiver interprets incoming serial data, reclocks it and, usually, deserializes it back into a parallel data stream. In this type of application, the reference clock is considered more of a constituent than a key player, but in high rate serial data systems the reference clock is a key component.

Typically the reference clock oscillates at a rate much lower than the data rate and is multiplied up in the transmitter which uses the result to define the timing of logic transitions in the serial data stream. The character of the reference clock is included in the data transmitted. At the receiver, two different things can happen. If the reference clock is not distributed, then the receiver recovers a clock from the data stream – using, for example, a phase locked loop (PLL) – and uses that clock to position the sampling point in time. If the reference clock is distributed, then the receiver uses both the data signal and the reference clock to position the sampling point.
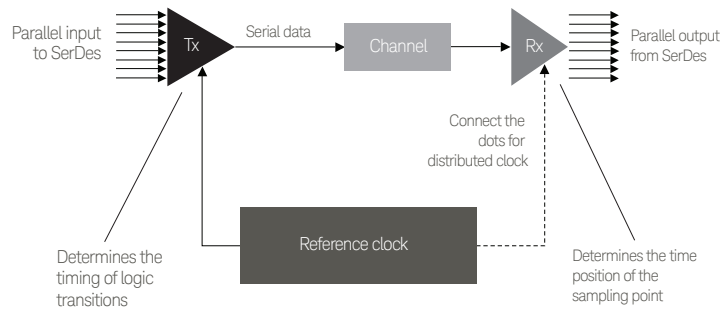

Figure 7. Straw diagram of a serial data system emphasizing the principle components

Random and deterministic jitter, RJ and DJ

In the analysis of serial data systems, it's useful to distinguish two categories of jitter, random jitter (RJ) and deterministic jitter (DJ).

Figure 8 shows an example of how RJ and DJ appear on a signal. DJ determines the trajectory of each logic transition in an eye diagram and RJ smears each occurrence of a particular trajectory. DJ can be associated with a peak-to-peak value, DJ(p-p), and RJ with the width, or rms value, of its distribution, σ.
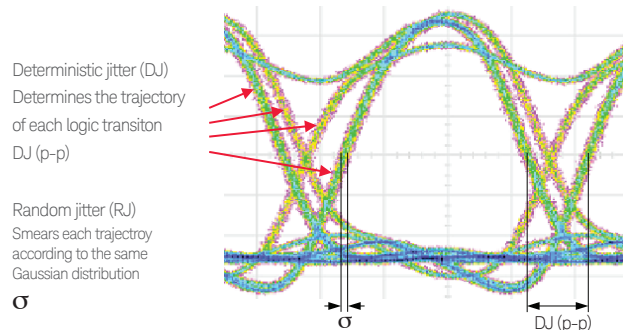

Figure 8. The characteristics of random and deterministic jitter

The characteristics of random jitter – RJ

RJ is caused by the accumulation of a huge number of processes that each have very small magnitudes; for example thermal noise, variations in trace width, shot noise, etc. The central limit theorem of probability and statistics[5] can be used to describe RJ: The PDF of an infinite number of small independent random processes follows a Gaussian distribution,

$$g(x) = \frac{1}{\sqrt{2\pi}\,\sigma}\,\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

As usual in engineering, it's common to ignore some of the formalism. Many of the processes aren't independent and some of them aren't so small.
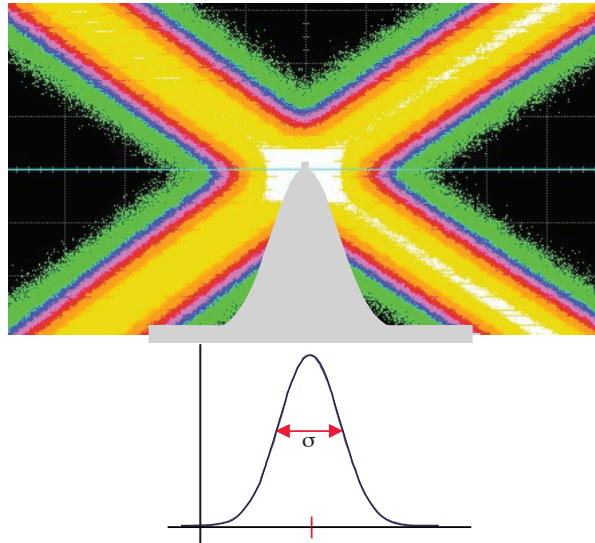

Figure 9. Random jitter doesn't have a well defined peak-to-peak value.

The important thing about RJ is that its PDF is unbounded. There is a tiny probability that RJ could cause a logic transition to occur at some time arbitrarily earlier or later than when it should. That RJ is unbounded means that it doesn't have a well defined peak-to-peak value. Since RJ is described by a Gaussian, the width, or standard deviation, σ, of the distribution, is sufficient to describe the magnitude of RJ.

The characteristics of deterministic jitter – DJ

DJ is caused by a comparatively small number of processes that need not be independent and may have large magnitude, for example electromagnetic interference, reflections and channel frequency response. It's called "deterministic" jitter because, in principle, if you knew everything there is to know about a system, you could accurately predict the jitter of each edge. The important thing about DJ is that its PDF is bounded. Hence, unlike RJ, DJ has a well defined peak-to-peak value, DJ(p-p).
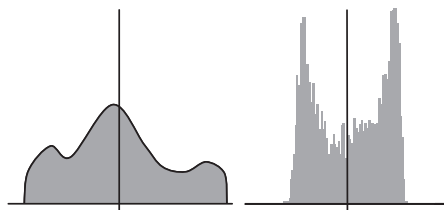

Figure 10. Deterministic jitter combines to form bounded distributions of varying shapes

Total jitter in terms of the BER

The jitter PDF is the convolution of RJ and DJ and, due to the RJ component, it is unbounded. Because it is unbounded, the peak-to-peak value of the jitter PDF is not well defined. In fact, the longer it is measured, the larger it is likely to become.

To compare the performance of different components, including clocks, you need a single quantity that describes the amount of jitter on a signal in a way that is related to the BER contribution of that component. The simple peak-to-peak value of the jitter PDF doesn't meet this need.

Therefore total jitter is defined as Total Jitter at a Bit Error Ratio, TJ(BER).

TJ(BER) is defined as the amount of eye closure at a given BER[6]. It's easiest to describe in terms of a BERTscan or bathtub plot[7]. The bathtub plot is a measurement of the BER as a function of the time-position of the sampling point, x. BER(x) can be measured on a bit error ratio tester (BERT), by scanning the sampling point across the eye-diagram and measuring BER at each point, x. Near the crossing points, BER(x) is large, peaking at $1/\rho^2$ where ρ is the signal transition density. As the sampling point moves toward the eye center, the BER drops very fast. Similarly, but in reverse, as the sampling point approaches the opposite crossing point, BER increases rapidly. The eye opening at a given BER is the distance between the two slopes of BER(x) at that BER. TJ(BER) is the eye closure, which is the width of the eye minus the eye opening.
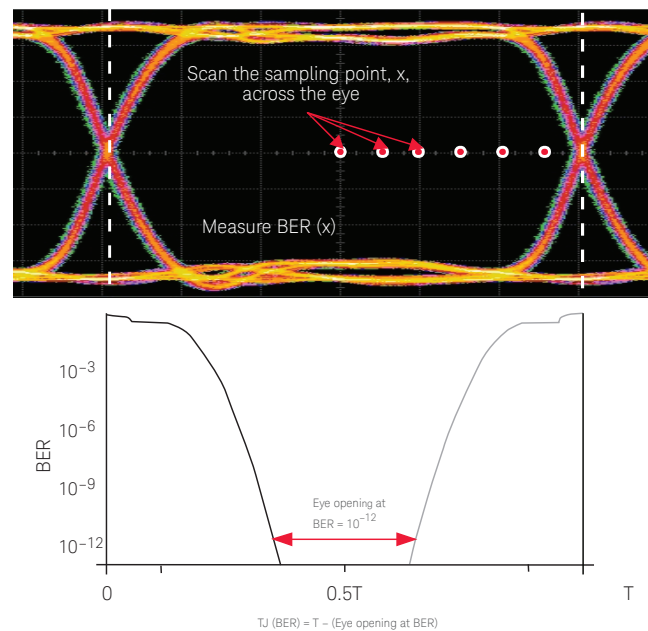
Total Jitter
TJ (BER)

Figure 11. The bathtub plot shows BER as a function of the time-position of the sampling point

Jitter categories

Figure 12 shows a larger breakdown of jitter including the components of DJ and a summary of their interrelationships.

Total jitter at a bit error ratio
Abstract "peak-to-peak"

Random jitter (RJ), σ — Unbounded fluctuations

Deterministic jitter, Jpp^{DJ} — Bounded, peak-to-peak

Thermal noise, shot noise, flicker

Periodic jitter (PJ) — Sinusoidal

Data dependent jitter (DDJ) — Data smearing

Bounded uncorrelated (BUJ) — Crosstalk

Duty cycle distortion (DCD) — Lead/trail edge

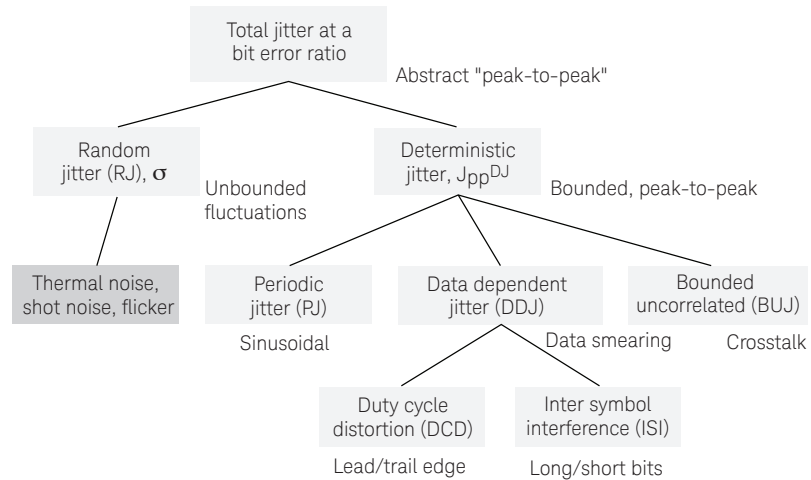Inter symbol interference (ISI) — Long/short bits

Figure 12. The jitter family tree

Jitter analysis technologies

Any jitter measurement can be reduced to the comparison of two clocks. Some sort of golden reference clock is compared to the timing of the transitions of either a test clock or the clock represented by the timing of data transitions on a signal. For jitter analysis on serial data, the golden reference clock may need to behave with a frequency response prescribed by the technology standard with which the system is intended to comply.There are three fundamental technologies used in the analysis of jitter.

Sampling techniques – build a jitter data set by repetitive sampling of a data signal. This is the technology behind the Keysight Technologies, Inc. equivalent-time sampling oscilloscope, the 86100C Infiniium DCA-J.

Real-time techniques – build a jitter data set from a continuous sweep of a signal. There are many approaches that use real-time technology including real-time oscilloscopes, like Keysight's 80000 Infiniium series, and phase noise analyzers like Keysight's E5052A and E5001A SSA-J.

Digital techniques – build a jitter data set on a bit-by-bit basis. For each bit a true/false type datum is acquired – whether or not jitter was observed. This is the technology used for jitter analysis on BERTs, such as Keysight's N4900 series, to assemble a bathtub plot, BER(x).

Jitter propagation on serial signals

To conclude this brief review of jitter in serial data systems, Refer to figure 13 and consider what type of jitter each component is most likely to generate.
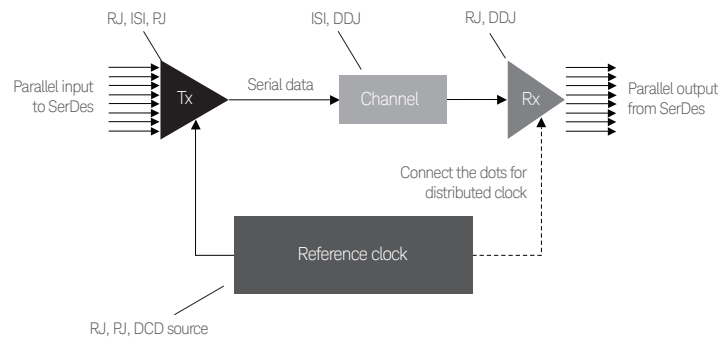

Figure 13. Types of jitter typically generated by the reference clock, transmitter, channel, receiver and other circuit elements

The reference clock – generates primarily RJ from the thermal noise of the oscillator, PJ from spurious sideband resonances of the oscillator, and duty-cycle distortion (DCD) from nonlinearities in the oscillator circuit.

The transmitter – contributes RJ from thermal effects, inter-symbol interference (ISI) from the frequency response of internal transmission lines, and periodic jitter (PJ) from pickup of EMI.

The transmission channel – generates ISI and, if there is duty cycle distortion (DCD) on the incoming signal, data dependent jitter (DDJ), from frequency response and attenuation characteristics.

The receiver – generates RJ from shot noise, and DDJ from internal circuitry. Since the receiver identifies the logic values of the signal, the type jitter it introduces isn't as important as whether or not it can correctly identify the bits.

The EMI of circuit elements – can cause PJ and bounded uncorrelated jitter (BUJ). While it's possible to assign PJ, ISI, DCD, and DDJ well defined mathematical descriptions. BUJ is the repository for other types of bounded jitter. The best example of BUJ is generated by crosstalk from neighboring signals.

The Role of Reference Clocks in Serial Data Applications

In serial data applications, the reference clock is the ultimate source of system timing. It provides the time-base for the transmitter and, in both distributed and undistributed clock systems, the character of the reference clock is reproduced in the clock recovery circuit at the receiver. This section examines how the clock-jitter is propagated through each part of the system.

Effect of clock jitter on data jitter from the transmitter

To define the timing of logic transitions, the transmitter must multiply the reference clock by an appropriate factor to get the data rate. For example, for a 100 MHz reference clock and a 5 Gb/s output signal, the transmitter would use a PLL to multiply the reference clock by a factor of fifty. The PLL multiplier both amplifies the jitter on the clock and introduces its own jitter, primarily RJ from the PLL voltage controlled oscillator (VCO). The effect of frequency multiplication[8] by a factor of n is to multiply the phase noise power to carrier ratio by n^2. The jitter goes up fast!
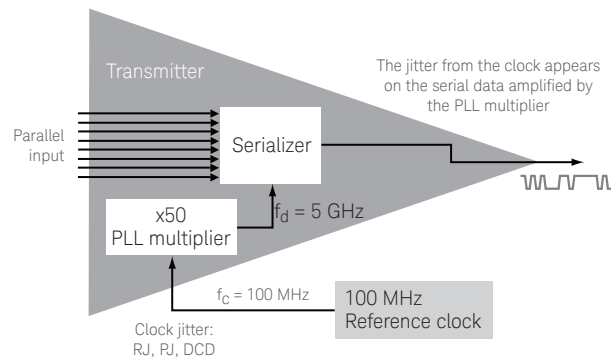
Figure 14. The effect of clock jitter from the transmitter

The PLL multiplier in the transmitter has a certain frequency response[9], typically a second order response like the one shown. The non-uniform frequency response raises an interesting question: What clock-jitter actually matters?

If the PLL were perfect and had zero bandwidth, then it would filter out all the clock-jitter and provide the transmitter with a jitter-free time base. Of course, zero bandwidth means infinite lock time, so you have to compromise, but the narrower the PLL bandwidth, the less jitter from the reference clock makes it into the data. Determining whether or not a clock will function in a system at the desired BER requires careful testing of the jitter frequency spectrum.
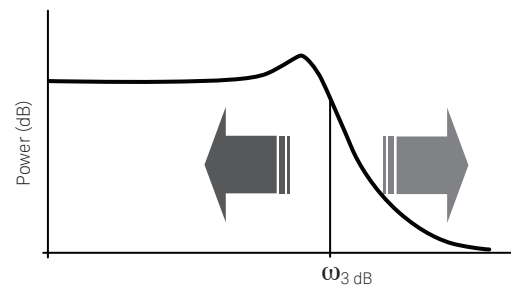

Figure 15. Typical 2nd order PLL frequency response

Effect of clock jitter on data jitter from the channel

Since the transmission channel is passive it generates negligible RJ. However, the resistance of the channel combined with impedance mismatches and the skin effect cause both attenuation and non-uniform frequency response which cause inter-symbol interference[10] (ISI). The jitter originating from the reference clock doesn't have any ISI, after all it has no symbols to interfere, but it may have DCD. The ISI introduced by the channel is affected by DCD. The combination of ISI and DCD is called data-dependent jitter (DDJ) and the amount of DDJ introduced by ISI changes with different amounts of DCD. DDJ = DCD $*$ ISI.

ISI is predominantly caused by the frequency response of the channel. In the absence of DCD, a data signal has only odd harmonics. As DCD is introduced, so are even harmonics. Since, DCD changes the frequency character of the signal, it changes the ISI caused by the channel. It is in this sense that ISI and DCD are correlated. Change one, and the other changes, too.
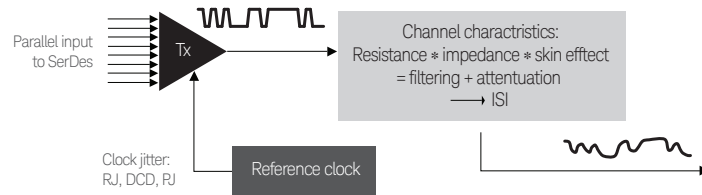

Figure 16. Reference clock DCD introduces even harmonics on the signal and complicates the ISI introduced by the channel

Role of the clock in the receiver – PLL-based clock recovery

The role of the clock at the receiver depends on whether or not the system has a distributed clock. First, consider the case where the reference clock is not distributed. In this case the receiver must obtain all clock information from the data. These systems usually use a PLL-based clock recovery (CR) circuit[11].

By the time it gets to the receiver, the clock jitter has been multiplied and filtered by the transmitter and convolved with the ISI of the channel. From this degraded signal, the receiver must recover a clock that it can use to position the sampling point and accurately identify the signal logic levels.

Like the transmitter multiplier, the clock recovery PLL in the receiver has a certain frequency response. But in the receiver, the narrower the bandwidth the greater the chance that jitter will cause misidentification of a bit and cause a higher BER. Consider the extreme cases. If the bandwidth of the CR circuit were infinite, then the recovered clock would have all the jitter from data. The sampling point would jitter back and forth precisely the same way as the data and there wouldn't be any errors. It is in this sense that the recovered clock tracks data jitter.

In the opposite extreme, a zero bandwidth CR circuit, the sampling point would be fixed at those "ideal positions in time." Jitter at any frequency on the data could cause logic transitions to fluctuate across the sampling point and cause errors.

Of course it's impossible to build either a zero or infinite bandwidth CR circuit. A real CR circuit has finite bandwidth that passes more low than high frequency jitter to the sampling point. The bandwidth of the CR circuit is an important characteristic of the application and should be prescribed by the technology standard.
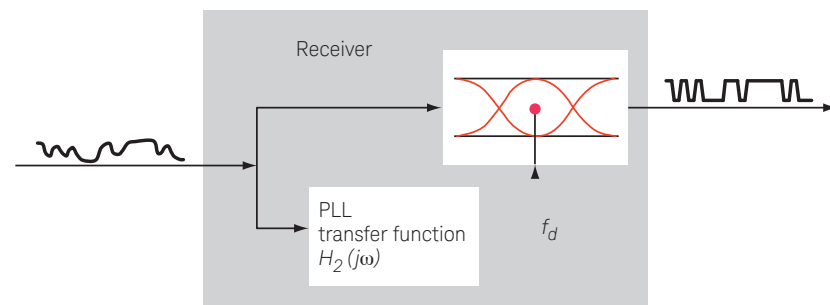

Figure 17. The receiver uses a PLL to recover the clock from the data.

Role of the clock in the receiver – Distributed clock system, phase interpolating clock recovery

In a distributed clock system, the receiver has access to the reference clock. The reference clock is first multiplied up to the data rate, and then aligned with the incoming data by a phase interpolator. Phase interpolators use digital techniques rather than the carefully designed (i.e., expensive) PLL CR circuit used in the undistributed case. The drawback of phase interpolators is that, since they are nonlinear devices, their frequency response isn't as easy to model as that of a PLL. In the absence of specific data, they are usually modeled as PLLs anyway.
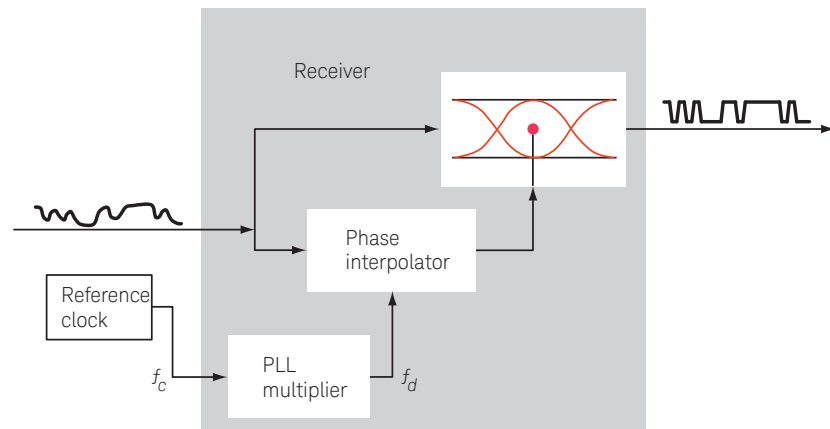
Figure 18. A distributed clock recovery with a phase interpolator

Frequency response of the clock recovery, a typical transfer function

The clock recovery transfer function can be modeled by a second order PLL as shown in Figure 19. The transfer function has two parameters, the natural frequency and the damping factor which combine to determine the bandwidth. The peaking is determined by the damping factor; the greater the damping factor the greater the peaking.

The receiver has a transfer function that is typically modeled by a 2nd order PLL.

$$H(s) = \frac{2s\zeta\omega_n + \omega_n^2}{s^2 + 2s\zeta\omega_n + \omega_n^2}$$

Where
ω_n is the natural frequency
ζ is the damping factor
s is the Laplace variable.

The natural frequency, ω_n, is related to the 3 dB frequency by:

$$\omega_{3dB} = \omega_n\sqrt{1 + 2\zeta^2 + \sqrt{(1 + 2\zeta^2)^2 + 1}}$$
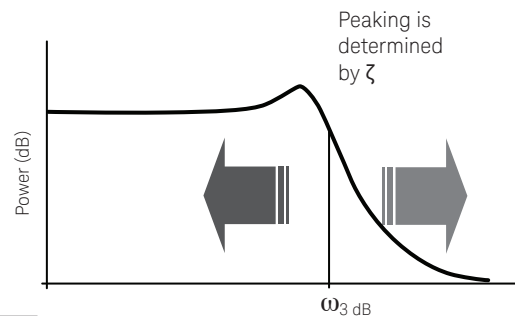
Figure 19. Clock recovery behaves like a jitter filter. The receiver tracks low frequency but not high frequency jitter.

Figure 20 illustrates two key points. First, the CR behaves as a low-pass jitter filter. Consequently, the recovered clock tracks the low frequency jitter on the data, but not the high frequency clock jitter. The result is that the system BER is affected more by high frequency jitter than low. Second, some of the jitter is amplified by transfer function peaking. Since jitter amplified by the CR circuit doesn't track the corresponding data jitter, it also increased the system BER.

The unit interval

You may recall a cryptic statement in the first section of this paper: "It is ... not entirelyaccurate to think of the 'ideal positions in time' as integer multiples of the bit period, T." In the light of the effect that the CR bandwidth has on the ability of the sampling point to track the jitter on the signal, it's important to define "ideal positions in time."

The ideal position in time is any time at which the receiver can set the sampling point that results in an error free system. The naïve definition of a unit interval, a bit period, corresponds to a zero-bandwidth CR circuit but, as shown in this document, a wider bandwidth results in fewer errors. That the receiver can track, and therefore tolerate, jitter is a major advantage; the BER is reduced at the cost of complicating what is meant by unit interval and bit period in the context of TJ(BER).
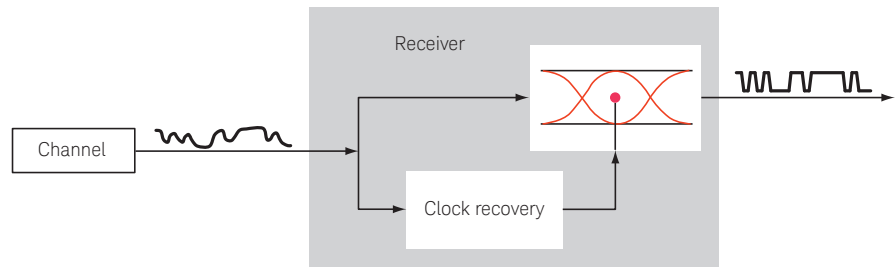

Figure 20. Diagram of a receiver emphasizing that the clock recovery circuit positions the sampling point.

The unit interval in serial data systems – phase jitter

Figure 21 illustrates how it all fits together. First, the transmitter multiplies the clock frequency up to the data rate to define the timing of logic transitions. In so doing, it introduces some jitter on the data from its multiplication circuit and amplifies the clock-jitter below the bandwidth of the multiplication circuit. The channel introduces ISI to the data, the amount of which depends on the magnitude of DCD, resulting in DDJ.

It all comes together at the receiver. The BER is determined by the misalignment of the sampling point with data transitions. The sampling point is not a fixed distance in time from the ideal position of a logic transition – it moves. The elusive "ideal positions in time" are determined by the relative time position of each sampling point and their associated data transitions, but the clock is recovered from data transitions and used to determine the sampling point. It's an interesting web of interdependence.

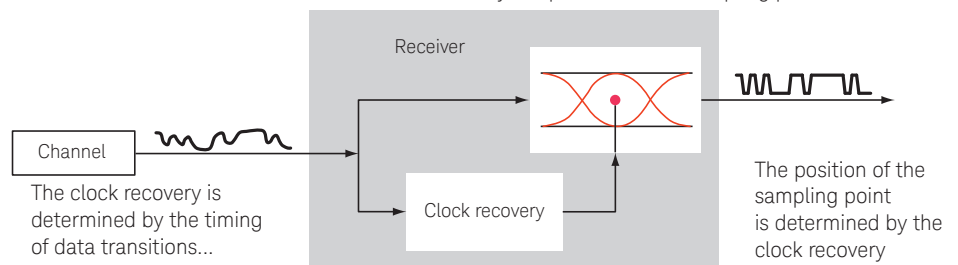The "ideal" time is determined by the position of the sampling point

Figure 21. The interdependence of data transitions, clock recovery, "ideal times," and sampling point positioning.

...then how do you analyze clock jitter?

Clock-jitter should be analyzed under system-specific assumptions about the transfer functions of the transmitter and receiver to determine if a given clock will work in a given system at the desired BER. You do this by:

1. Determining the limiting requirements of the specific system, usually from the standards committee. For example: PCI-Express, FBD, sATA, FibreChannel, etc.
2. Applying the limiting case transmitter and receiver transfer functions to the clock
3. Analyzing the resulting jitter to determine the effect of clock jitter on the BER.

Before diving into analysis, it's important to review what clocks are, how they work, and their characteristics.

Reference Clocks and Phase Noise

This section begins with a review of oscillators emphasizing their noise characteristics which leads naturally to a discussion of phase noise, Bessel functions, the SSB spectrum, another discussion of how amplitude and phase noise are distinguished, and, finally, once again, the relationship of phase noise and jitter.

Oscillators

An oscillator is any system with repetitive dynamics. The dynamics of a low-amplitude pendulum are described by the linear, second order, inhomogeneous, ordinary differential equation with constant coefficients in terms of the vertical rise, y, of the pendulum as a function of time, t. The term D(t) describes how the oscillator is driven. The damping factor – which is the same damping factor that shown earlier in reference to a PLL transfer function – is usually small enough that the oscillator is free to oscillate. The solution to the differential equation is a simple sinusoid whose frequency is determined by the driving term. The natural, or resonant, frequency is the frequency at which the pendulum would oscillate if there were no driving term. It's interesting to note that, in linear systems, the resonant frequency depends only on the geometry of the configuration, not on the initial conditions of the oscillator or the driving term.
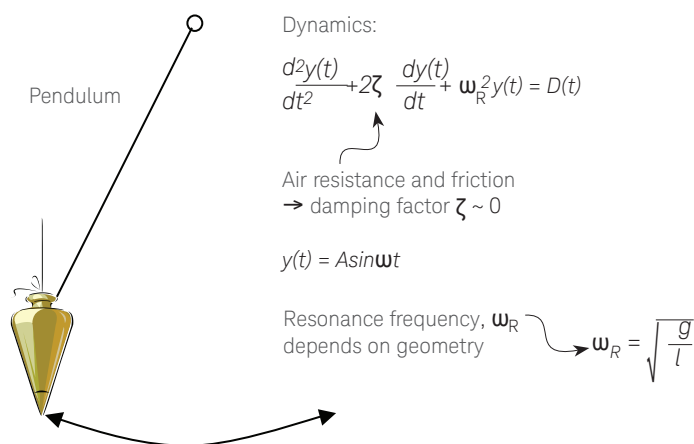
Pendulum

Dynamics:

$$\frac{d^2y(t)}{dt^2} + 2\zeta\frac{dy(t)}{dt} + \omega_R^2 y(t) = D(t)$$

Air resistance and friction
→ damping factor $\zeta \sim 0$

$$y(t) = A\sin\omega t$$

Resonance frequency, ω_R depends on geometry

$$\omega_R = \sqrt{\frac{g}{l}}$$

Figure 22. An oscillator is a system that exhibits periodic variation like a pendulum

Electrical oscillators

An electrical oscillator is easiest, and most generally, described by an LRC circuit.[12] The dynamics are the same as those of a pendulum. Just like air resistance and friction were manifest in the damping term of the pendulum, electrical resistance is manifest in the damping term of the oscillator. The driving term is, of course, given by the applied voltage, which, for clocks, follows a simple sinusoid of frequency ω. A good oscillator – the only oscillators of interest here – has a resistance small enough that it has negligible effect on the resonant frequency. The carrier frequency is usually chosen at the oscillator's resonant frequency.
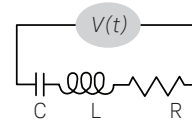
An electrical oscillator:

$$\frac{d^2i(t)}{dt^2} + 2\zeta\,\frac{di(t)}{dt} + \omega_R^2\,i(t) = D(t)$$

$$\frac{d^2i(t)}{dt^2} + \frac{R}{L}\,\frac{di(t)}{dt} + \frac{1}{LC}i(t) = \frac{1}{L}V(t) \qquad\qquad i(t) = \frac{V_m \sin \omega t}{\sqrt{R^2 + \left(\omega L - \frac{1}{\omega C}\right)^2}}$$

$$\omega_R = \sqrt{\frac{1}{LC} - \left(\frac{R}{2L}\right)^2} \approx \frac{1}{\sqrt{LC}} \qquad \text{for a good oscillator}$$

$$\frac{R}{2L} = \zeta$$

Figure 23. An electrical oscillator and its dynamics.

Oscillator parameters

The traditional parameters used to characterize oscillators are shown in figure 24.
The plot shows the current response, i(t), of three different oscillators as a function of driving frequency. The only difference between the three oscillators is their resistance. The oscillator most sharply peaked has 1/3 the damping factor of the next-most sharply peaked and 1/10 that of the least sharply peaked. The bandwidth is proportional to the damping factor and the quality is the ratio of the resonant frequency to the bandwidth, which is inversely proportional resistance. The quality of an oscillator is a measure of how sharply peaked the response curve is. Typically the quality of crystal oscillators used in data and telecom applications is between 10^4 and 10^6. The quality of the sharply peaked oscillator in the graph is 100.

Damping factor $\zeta = \frac{R}{2L}$
$\zeta_1 = (1/3)\,\zeta_2 = (1/10)\,\zeta_3$
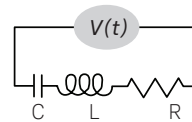
Damped resonance frequency

$$\omega_s = \sqrt{\frac{1}{LC} - \left(\frac{R}{2L}\right)^2} \approx \frac{1}{\sqrt{LC}}$$

$$i(t) = \frac{V_0 \sin \omega t}{\sqrt{R^2 + \left(\omega L - \frac{1}{\omega C}\right)^2}}$$

Bandwidth $\Delta f = \frac{\zeta}{\pi} = \frac{R}{2\pi L}$
$\Delta f_1 = (1/3)\,\Delta f_2 = (1/10)\,\Delta f_3$

Quality $Q = \frac{f_R}{\Delta f} = \frac{1}{R}\sqrt{\frac{L}{C}}$
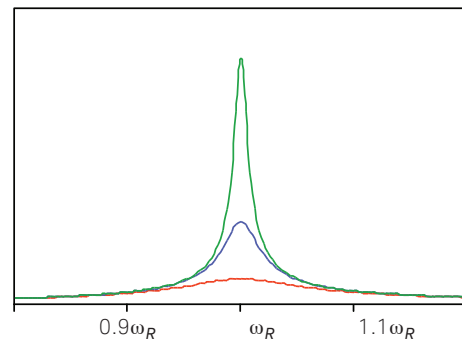$Q_1 = 3Q_2 = 10Q_3$

Figure 24. Oscillator parameters

Oscillators – Example: a crystal oscillator

In most cases a crystal oscillator is the heart of what we've been referring to as a "reference clock." Like a pendulum, the crystal, usually some sort of piezoelectric crystal, such as quartz, will oscillate with the least provocation[13]. To generate a stable oscillation[14], random noise is first applied to the crystal. The crystal responds with largest amplitude at its resonant frequency. The response of the crystal is amplified and returned to the crystal in a feedback loop. The crystal continues to respond primarily at its resonant frequency, increasing the amplitude at the oscillator output at a rate determined by the loop gain and the bandwidth of the crystal oscillator. The amplitude stabilizes when the amplifier gain is reduced either externally or through its own self-limiting nonlinearities and the resulting signal is sharply peaked at resonance.

A nice feature of crystal oscillators is that their resonant frequencies can be adjusted by changing the geometry of the crystal itself.

Oscillator noise

At frequencies closest to the carrier, the primary noise source is the non-zero width of the resonance. A noiseless oscillator would have zero bandwidth and infinite quality which, not coincidentally, implies zero resistance and can only be achieved in superconductors[15]. Far from the carrier, the usual suspects such as power supply feed-through, impedance mismatches, and so forth, from the oscillator feedback loop affect both the phase and amplitude of the oscillator[16].

An ideal oscillator $v_{ideal}(t) = v_0 \sin 2\pi f_c t$

A real oscillator $v_{real}(t) = (v_0 + \Delta v(t)) \sin(2\pi f_c t + \phi(t))$

Amplitude noise   Phase noise

Thermal effects, such as Johnson noise, causes white noise. Temperature and pressure affect the crystal geometry and, consequently, its resonant frequency. Spurious frequencies can be generated, typically tens of kHz above the desired resonance, by vibration of the crystal. In the frequency domain, the spurious frequencies appear at integer multiples of the difference of the vibration and carrier frequencies.

There are two significant practical system level problems caused by oscillator noise. First, the power of the noise is taken from the carrier. Second, as described above, when the oscillator frequency is multiplied up to the data rate, the resulting phase noise is increased by the square of the multiplication factor. That is, the sidebands increase 20 dB for every factor of ten in the multiplier. Unfortunately phase noise cannot be eliminated by a limiting-amplifier and, since so much of the noise is close to the carrier, it can't be eliminated by filtering.

Phase noise as undesirable phase modulation –
Example: sinusoidal phase noise

Sinusoidal phase noise is an important example for many reasons, not the least of which is that, in many discussions of jitter and phase noise, Bessel functions are introduced[8]. Consider

$$v_{real}(t) = (v_0 + \Delta v(t)) \sin(2\pi f_c t + \phi(t))$$

and ignore amplitude noise, $\Delta v(t) = 0$. Let $\phi(t) = A\sin(\omega_j t)$ so that ω_j is the jitter frequency of the phase noise and get

$$v_{real}(t) = v_0 \sin(2\pi f_c t + A\sin(\omega_j t)),$$

Now use a trigonometric identity on $\sin(A + B)$ to get

$$v_{real}(t) = v_0 [\sin(2\pi f_c t) \cos(A\sin(\omega_j t)) + \cos(2\pi f_c t) \sin(A\sin(\omega_j t))].$$

Bessel functions are used to reduce the annoying $\cos(\sin x)$ and $\sin(\sin x)$ terms,

$$\cos(A\sin(\omega_j t)) = J_0(A) + 2 [J_2(A)\cos(2\omega_j t) + J_4(A)\cos(4\omega_j t) + ...]$$

$$\sin(A\sin(\omega_j t)) = 2 [J_1(A)\sin(\omega_j t) + J_3(A)\sin(3\omega_j t) + ...].$$

Plugging the Bessel function expansions into the original expression for vreal(t), gives a sum of sinusoids offset from the carrier frequency by integer multiples of the phase noise frequency,

$$v_{real}(t) = v_0 [J_0(A)\sin(2\pi f_c t) + J_1(A)\sin(2\pi f_c t + \omega_j t) - J_1(A)\sin(2\pi f_c t - \omega_j t) +$$

$$J_2(A)\sin(2\pi f_c t + 2\omega_j t) + J_2(A)\sin(2\pi f_c t - \omega_j t) + ...].$$

Bessel nulls and ratios

The carrier amplitude is modified by the zeroth-order Bessel function evaluated at the phase noise amplitude, $v_0 J_0(A)$, and the first sideband amplitude is given by the product of the first-order Bessel function and the amplitude of the desired signal, $\pm v_0 J_1(A)$.

The result is a technique for calculating the response of an oscillator to a type of phase noise that's easy to generate. The ratio of amplitudes of the carrier and first sideband tell us the phase noise amplitude. The phase noise amplitude can be tuned so that the carrier or sidebands have zero amplitude – this is the "Bessel null" technique of calibrating phase noise and jitter sources analyzers.

Clock signal – time domain view

A clock signal is shown in figure 25. In this example, a 2.5 GHz clock signal with a 300 kHz square-wave phase noise term of amplitude –56 dBc (i.e., 56 dB below the carrier). It is a time domain view on a Keysight 86100C DCA equivalent-time sampling oscilloscope showing the sinusoidal envelope of the signal. By zooming in to the slice-threshold, on the right, the expanse of jitter is easy to see.

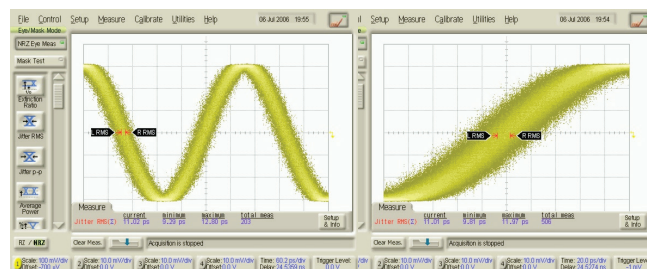$$v_{real}(t) = (v_0 + \Delta v(t)) \sin(2\pi f_c t + \phi(t))$$

Figure 25. A clock signal shown in the time-domain

Clock signal – frequency domain view

This is the same clock signal in the frequency domain. The graphic was taken by a Keysight E4440 Performance Spectrum Analyzer. The frequency spectral density is a measure of the amount of power per unit frequency in the signal. Mathematically, it's nice to think of S(f) as the square of the Fourier transform of the signal. For an ideal signal with neither voltage nor phase noise, the spectrum would yield a delta-function spike. Instead, the sidebands at 300 kHz and integer-multiple offset frequencies caused by the square-wave phase noise term and the ever-present white noise are plainly evident.

$$S(f) = |F[v_{real}(t)]|^2 = |F[(v_0 + \Delta v(t))sin(2\pi f t + \phi(t))]|^2$$

The spectrum analyzer plots the frequency spectral density,

$S(f)$

For an ideal signal, where $\Delta v = \varphi(t) = 0$,
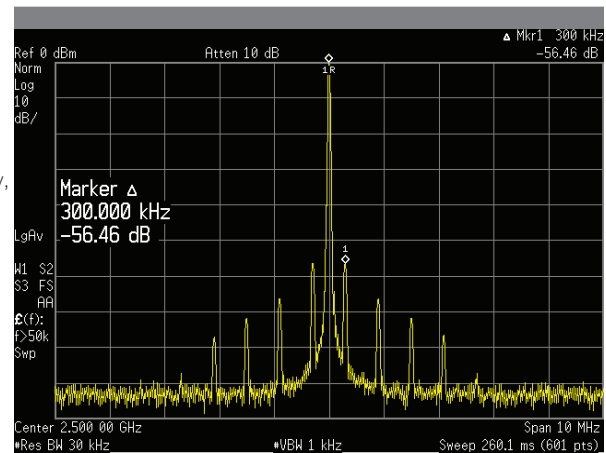
$S(f) = \delta(f - f_R)$

Figure 26. A clock signal shown in the frequency domain

Clock signal – phase noise in the time domain

This is a measurement of the signal phase noise in the time domain. The square wave is plainly evident above the noise.
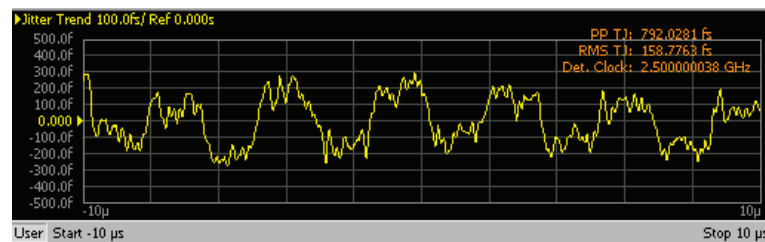
$\varphi(t)$

Figure 27. The phase noise of a clock signal in the time domain

Clock signal – phase noise in the frequency domain

And, finally, the more traditional view of the phase noise, plotted in the frequency domain. Notice that the phase noise frequency domain is distinct from the frequency domain shown on a spectrum analyzer. A spectrum analyzer displays the frequency domain of the signal. A phase noise analyzer displays the frequency domain of the phase noise term. Here's another way of thinking of it. The frequency spectral density, $S(f)$, is the square of the Fourier transform of the signal spectral density. The phase spectral density, $S_\phi(f)$, is the square of the Fourier transform of the phase noise term. They are different frequency domains and different functions. As we'll see below, the phase noise frequency domain, f_ϕ, is related to the signal frequency domain, f, through the offset frequency expression, $f_\phi = f - f_c$. We'll also see that the phase noise spectral density is related to the single-sideband spectrum, $L(f) \approx 1/2\, S_\phi(f)$. Due to this relationship which is significant for many historical reasons, the display shown above is actually $1/2\, S_\phi(f)$, that is, it's 3 dB less than $S_\phi(f)$.

$$S_\phi(f_\phi) = |F[\phi(t)]|^2 = |\phi(f_\phi)|^2 \qquad \tilde{\ }$$

The phase noise analyzer plots the Phase Spectral Density,

$$S_\phi(f_\phi)$$

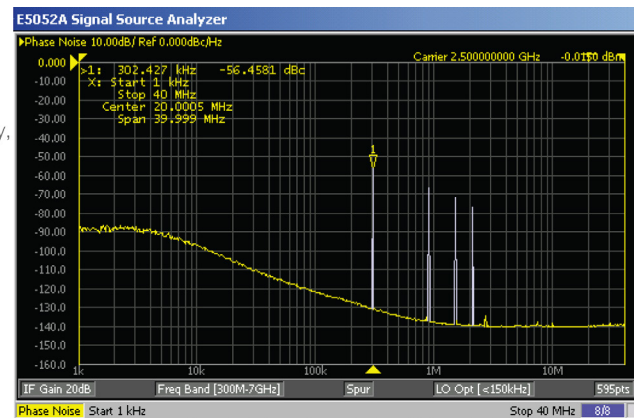The phase noise frequency domain is given by the offset frequency,

$$f_\phi = f - f_c$$

Figure 28. The phase noise of a clock signal in the phase-noise frequency domain

Phase noise

The phase noise continuum[17] can usually be traced to a handful of contributing sources and, in so doing, provide useful diagnostic information. In figure 29 the five common sources of phase noise are illustrated, but in most cases two or three noise processes dominate. Each type of noise is due to a distinct process in the circuit that can be identified by analyzing the phase noise over particular offset frequencies.

Random noise profile $S_\phi(f_\phi) = \sum \dfrac{Constant_n}{f_\phi^{\,n}}$

$1/f_\phi^4$ random walk FM

$1/f_\phi^3$ flicker FM

$1/f_\phi^2$ white FM

$1/f_\phi$ flicker phase noise

$1/f_\phi^0$ white phase noise
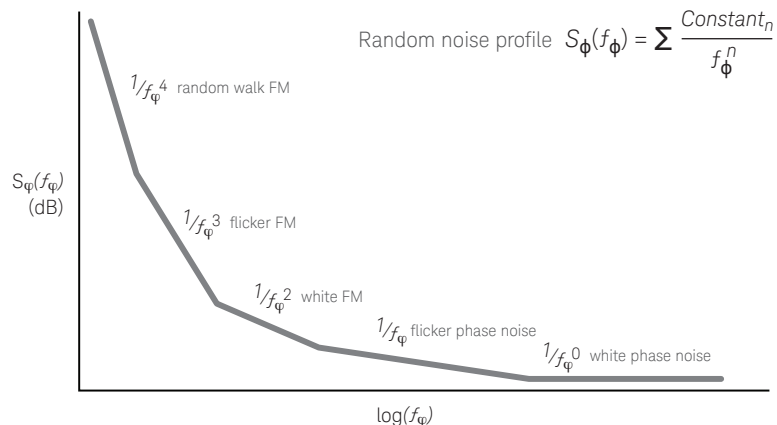
$S_\phi(f_\phi)$ (dB)

$\log(f_\phi)$

Figure 29. Five common sources of phase noise

Random walk frequency modulation (FM) noise follows $f_\phi{}^{-4}$ and so is usually too close to the carrier to measure. In random walk FM dominated systems it's likely that the oscillator itself is subject to mechanical shock, vibration, or temperature variations that can cause the carrier frequency to experience random shifts.

Flicker[18] FM noise follows $f_\phi{}^{-3}$, flicker is a fascinating process that is not well understood but seems to be related to the principal of causality. In practical terms, if flicker FM dominates then there is probably something fundamentally wrong with the oscillator itself. Flicker FM is present in all oscillators, though is usually negligible compared to white noise.

White FM noise follows $f_\phi{}^{-2}$ and is commonly found in clocks that use a client oscillator, like quartz, locked to another resonating device that has the character of a high-Q filter.

Flicker phase modulation (PM) noise follows $f_\phi{}^{-1}$. Like flicker FM, flicker PM can be associated with the physics of the resonator, but is more likely the effect of noisy electronics. It's common in even high quality oscillators due to the standard use of amplifiers to raise the signal level. Flicker PM can also be introduced by a frequency multiplier, such as is commonly used in transmitters. The easiest way to reduce flicker PM is to design low noise amplifiers for oscillator circuitry.

White PM noise is a flat background, $f_\phi{}^{0}$, and is introduced by noisy electronics. It can usually be traced to thermal noise generated in resisters, inductors, amplifiers, diodes, etc. Since it's so broad in frequency, narrowband filtering can reduce the white noise.

Phase noise analysis – Phase spectral density, $S_\phi\,(f_\phi)$

Phase noise analysis[19] is performed by using a phase detector to remove the carrier, leaving the phase noise of the signal. In the figure above, a golden reference clock is mixed with the clock under test. The relative phase of the reference clock is kept in quadrature with the clock under test, $\pi/2$, by a phase shifter. The output that follows the low pass filter is $V(t) \approx K_\phi \sin(\phi(t))$. Within the bandwidth of the mixer, the difference in phase between the reference clock and the clock under test is kept small ($\Delta\phi << 1$ rad) and $K_\phi\sin(\phi(t)) \approx K_\phi(t)$ where K_ϕ is the phase to voltage conversion factor in V/rad. Dividing by K_ϕ leaves the phase noise term $\phi(t)$ as shown above in slide 0. The signal analyzer converts the phase noise into the phase spectral density, $S_\phi(f_\phi)$.
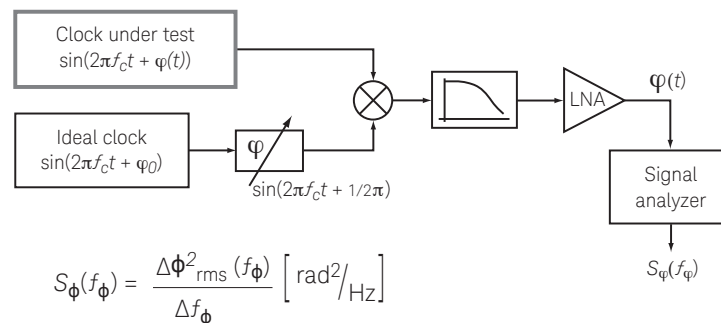

$$S_\phi(f_\phi) = \frac{\Delta\phi^2{}_{rms}\,(f_\phi)}{\Delta f_\phi}\left[\frac{rad^2}{Hz}\right]$$

Figure 30. Phase noise analysis

Phase noise and voltage noise

At this point it's necessary to go back and study the difference between phase noise and voltage noise. First, let $S_v(f)$ be the voltage spectral density,

$$S_v(f) = \frac{\Delta v^2_{rms}(f)}{\Delta f} \left[V^2/Hz \right]$$

The square of the voltage frequency density, S_v is the output of a spectrum analyzer.
It includes both amplitude and phase noise.
Since $S_\phi(f_\phi)$ has only phase noise,

$$S_\phi(f_\phi) = \frac{\Delta\phi^2_{rms}(f_\phi)}{\Delta f_\phi} \left[rad^2/Hz \right]$$

it only affects the distribution of signal power, not the total signal power. In the absence of phase noise, voltage noise is a series of impulse functions at the oscillator harmonics.

As phase noise is introduced, the impulse functions are broadened, in a way that reduces their amplitude so that the total power is unchanged. The qualitative result is that the greater the phase noise, the broader the linewidth and the lower the signal amplitude.

The phasor diagram in figure 31 shows how all this fits together. The carrier rotates about the phasor at the carrier frequency. Voltage noise adds to the signal vectorially. Contributions of voltage noise that are parallel to the carrier vector are considered amplitude noise and contributions perpendicular to the carrier vector cause phase noise. Power is only associated, through Ohm's law[12], with voltage, not phase.

Since oscillators are phase noise dominated, except at frequencies far from the carrier and its harmonics, and since phase noise analysis is orders of magnitude more sensitive than voltage noise analysis, the analysis of clocks is generally most precise when carried out in the phase noise, or offset, frequency domain, $f_\phi = f - f$c.
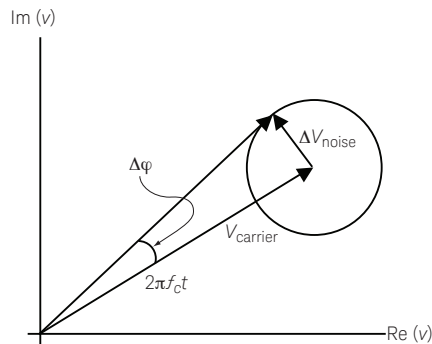

Figure 31. Phasor diagram

Jitter on clocks – single sideband noise spectrum, $L(f_\phi)$

Another way to analyze oscillator noise is to extract the single side band (SSB) noise spectrum, $L(f_\phi)$. Each unit of the voltage spectral density, $S_v(f)$, is divided by the carrier power and is then plotted as a function of the difference between the frequency of that unit and the carrier, $f - f_c$, on a logarithmic scale.

It's not uncommon for a spectrum analyzer to have software that can extract the SSB spectrum as seen in Figure 32.

$$L(f) = \frac{1}{2P_C}\frac{\Delta P(f)}{\Delta f} = \frac{\text{Power density of one phase modulation sideband}}{\text{Carrier power}} \quad \left[\text{dBc}^2/\text{Hz}\right]$$
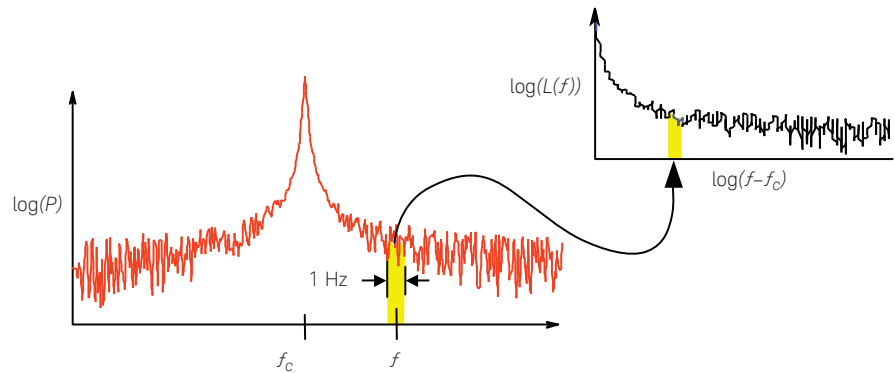

Figure 32. SSB extracted from the frequency domain signal

The SSB spectrum and the phase spectral density

There is a common misunderstanding that the SSB spectrum and the phase spectral density are the same thing. Since $L(f_\phi)$ is derived from $S_v(f)$ which is distinct from $S_\phi(f_\phi)$, the distinction should be obvious, but take a minute to consider how similar $L(f_\phi)$ and $S_\phi(f_\phi)$ are. In Figure 33, Ohm's law is used to convert power to voltage, $P = v2/R$. Common terms cancel and the voltage spectral density, $S_v(f)$, emerges. In the last step, if negligible amplitude noise is assumed, then the voltage noise only contributes to the component perpendicular to the carrier in the phasor diagram. Thus, only in the limit of zero amplitude noise is the SSB spectrum equal to half the phase spectral density.

It's worth pointing out that the dimensions of the terms, $L(f)$, $S_v(f)$, and $S_\phi(f_\phi)$, are the same, even though they don't look like it. Radians and decibels aren't dimensions in the same sense as meters, kilograms, seconds, or Coulombs; dBc and radians are included as reminders, not as dimensions.
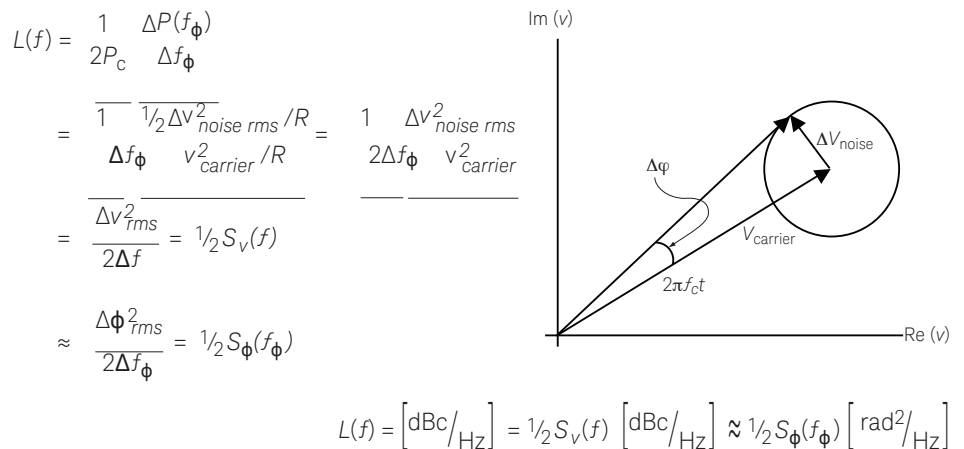
$$L(f) = \frac{1}{2P_C}\frac{\Delta P(f_\phi)}{\Delta f_\phi}$$

$$= \frac{1}{\Delta f_\phi}\frac{\tfrac{1}{2}\Delta v^2_{noise\,rms}/R}{v^2_{carrier}/R} = \frac{1}{2\Delta f_\phi}\frac{\Delta v^2_{noise\,rms}}{v^2_{carrier}}$$

$$= \frac{\Delta v^2_{rms}}{2\Delta f} = \tfrac{1}{2}S_v(f)$$

$$\approx \frac{\Delta\phi^2_{rms}}{2\Delta f_\phi} = \tfrac{1}{2}S_\phi(f_\phi)$$

$$L(f) = \left[\text{dBc}/\text{Hz}\right] = \tfrac{1}{2}S_v(f)\left[\text{dBc}/\text{Hz}\right] \approx \tfrac{1}{2}S_\phi(f_\phi)\left[\text{rad}^2/\text{Hz}\right]$$

Figure 33. The SSB spectrum and the phase spectrum density

Phase noise and jitter

Having exhausted the relationships between voltage, amplitude and phase noise, it is time to show how jitter fits in. Recall that jitter is "the short term phase variation of the significant instants of a digital signal from their ideal positions in time." Phase noise is a continuous function of time that indicates the deviation of a digital signal's phase from the ideal phase, $\phi(t)$. Jitter is the discrete difference between the actual time that a logic signal crosses the slice threshold and the ideal time, Φ_n,

Thus, jitter is proportional to the phase noise at each slice threshold.

$$\Phi_n = t_n - nT$$
$$= \frac{\phi(t_n)}{2\pi f_c}$$

jitter, Φ_n, can also be expressed in seconds, as it is here, and in radians, by removing the carrier frequency in the last line, and in unit intervals, by removing the carrier frequency and the factor of 2π in the last line.

RJ calculation from phase spectral density

A useful result is that you can calculate RJ from the phase spectral density. $S_\phi(f_\phi)$ is the square of the average phase deviations per unit offset-frequency. Thus integrating it over whatever bandwidth is desired and taking the square root of the result yields the width, σ, of the RJ Gaussian distribution.

Jitter is discrete but RJ can be derived from the continuum. By its random nature, the ensemble of phase noise at any point in the waveform is the same as that at the slice threshold. The bandwidth of phase noise analysis is limited by the bandwidth of the phase detector in the phase noise analyzer. In serial data systems, RJ is specified up to the Nyquist frequency of the system.
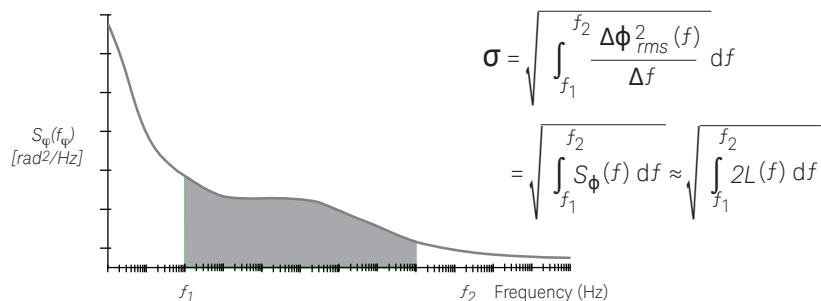

$$\sigma = \sqrt{\int_{f_1}^{f_2} \frac{\Delta\phi_{rms}^2(f)}{\Delta f} \, df}$$

$$= \sqrt{\int_{f_1}^{f_2} S_\phi(f) \, df} \approx \sqrt{\int_{f_1}^{f_2} 2L(f) \, df}$$

Figure 34. RJ is obtained by integrating the continuum phase spectral density over the bandwidth of interest.

Reference Clock Quality Analysis

There is significant historical momentum behind how clocks are evaluated. Many of the established techniques, like phase noise analysis, provide a solid foundation for clock quality analysis in high rate serial data systems.

Reference clock quality analysis

The quality of a clock depends on the point of view. Traditional clock specifications like peak-to-peak phase jitter, period jitter, and cycle-to-cycle jitter indicate clock quality but don't answer the only truly relevant question: Will the clock work in the system you are designing? Only jitter that can cause errors is relevant and, as seen, determining the bands of jitter frequencies that can cause errors in a given system differs for different technologies. The first step in evaluating a clock for a given application is looking at the data sheet.

Historic reference clock specifications

The definitions for phase, period, and cycle-to-cycle jitter are

Phase jitter: The accumulated variation of the phase from an ideal clock,
$$\Delta t_{phase}(n) = t(n) - nT$$

Period jitter: The variation of a clock cycle from the ideal period
$$\Delta t_{period}(n) = [t(n) - nT] - [t(n-1) - (n-1)T] = t(n) - t(n-1) - T$$
$$= \Delta t_{phase}(n) - \Delta t_{phase}(n-1)$$

Cycle-to-cycle jitter: The variation of adjacent clock periods
$$\Delta t_{cyc\text{-}cyc}(n) = [t(n+1)) - t(n)] - [t(n) - t(n-1)] = t(n+1) - 2t(n) + t(n-1)$$
$$= \Delta t_{period}(n) - \Delta t_{period}(n-1)$$

Again, phase jitter, Φ_n, is the same as cumulative jitter and is usually what you call "jitter." In the data sheets, quoted values for these quantities are frequently peak-to-peak values, which, of course, are not well defined quantities. In comparing clock quality they serve as a single value that can be used as a qualitative measure. Often rms values are also quoted, which are well defined and also provide a qualitative measure.

The most useful measure of clock quality that has historically been provided in data sheets is the phase noise spectrum. A simple way to determine whether or not a clock is adequate for an application is to apply a mask test to the phase noise spectrum. By requiring the phase noise fit under a mask, the amount of phase noise in different offset frequency bands can be limited.

In response to the use of RJ and DJ to estimate TJ(BER) in serial data systems, some clock vendors have started quoting RJ and DJ. Similar in spirit to applying a mask test to a phase noise spectrum, RJ and DJ can be used to quickly estimate a maximum TJ(BER) contribution from the clock for a given application. Clock DJ is a combination of periodic jitter (PJ) and Duty-cycle distortion (DCD). In most cases, the reference clock doesn't exhibit DCD so, PJ is the only relevant form of DJ.

In any case, though, to judge the utility of a clock in a specific application you should apply models of the specified worst-case response of the transmitter and receiver clock recovery functions.

Quantities quoted on clock data sheets

Table 1 shows typical value ranges given in data sheets for clocks used in serial data applications of 1 Gb/s and higher.

Table 1. Typical values ranges in clock data sheets

Quantity	Typical Values (varies by application)
Cycle-to-cycle jitter	30 to 150 ps
Phase jitter	30 to 80 ps
Peak-to-peak jitter (without specifying number of cycles measured)	20 to 50 ps
rms of whole jitter distribution	2 to 5 ps
rms random phase jitter in named bandwidths	0.3 to 4 ps
Phase noise relative to carrier (at named offset frequencies)	−100 to −80 dBc/Hz
rms RJ	0.3 to 2 ps
DJ (doesn't specify DJ(p-p) or DJ(dd))	0.1 to 1 ps
TJ(10-12)	3 to 40 ps

TIE analysis on a real-time oscilloscope

Real-time oscilloscopes are the best tool for assembling the time interval error (TIE) data set. First a signal is captured, the top trace in the above diagram, then the values of that signal at the voltage slice level are assembled giving the TIE data, { t_n }. The actual data is acquired by extremely fast ADCs so it is not truly an analog trace. The precise crossing times must be interpolated from each set of two data points that bracket the slice level. If the bandwidth of the oscilloscope is sufficient (three times the data rate is usually adequate) the interpolation shouldn't introduce appreciable uncertainty.

With the TIE data in hand, the phase jitter histogram is easy to extract – a measure of the PDF – and the jitter trend, $\varphi(t_n)$ can be plotted. Notice the distinction between the discrete jitter trend, $\varphi(t_n)$, and the continuous time-domain representation of the phase noise $\varphi(t)$. The jitter spectrum can be calculated by using the usual trick of padding the discrete data set, { t_n }, with zeros and applying a discrete Fourier transform[20]. Again, notice the distinction between the jitter spectrum, which is the Fourier transform of the crossing times and the phase noise spectrum which is the Fourier transform of the phase noise.

Figure 35. TIE analysis on a real-time oscilloscope

Jitter analysis on a time interval error data set

It's easy to extract the phase, period, and cycle-to-cycle jitter from the TIE data set, { t(n) },

Δt_{phase} (n) = max (or rms) { t(n) – nT }
Δt_{period} (n) = max (or rms) { [t(n) – nT] – [t(n–1) – (n–1)T] = t(n) – t(n–1) – T }
$\Delta t_{cyc-cyc}$ (n) = max (or rms) { [t(n+1) – t(n)] – [t(n) – t(n–1)] }

as well as RJ and DJ[21].

Time interval error analysis

The power of the TIE data in jitter analysis is tremendous. Given the worst case transfer characteristics of the transmitter and receiver, the techniques of digital signal processing[22] (DSP) can be used with impunity. For example, the second-order PLL transfer function,

$$H(s) = \frac{2s\zeta\omega_n + \omega_n^2}{s^2 + 2s\zeta\omega_n + \omega_n^2}$$

$$\omega_{3dB} = \omega_n\sqrt{1 + 2\zeta^2 + \sqrt{(1 + 2\zeta^2)^2 + 1}}$$

can be applied to the TIE data to determine the RJ and DJ that the clock will contribute to the TJ(BER) of the system.

In practice, the TIE data must be provided by an oscilloscope with sufficient bandwidth to represent the signal and sufficient memory depth to provide enough data to assure accuracy.

The biggest drawback to use of TIE techniques is the signal integrity of real-time oscilloscopes. While they are without question the most flexible tool in your lab, they can rarely compete with the fidelity of an equivalent-time sampling oscilloscope and can't approach the sensitivity of a phase noise analyzer.

TIE analysis on a real-time oscilloscope – example: spread spectrum clock

Many standards allow the use of spread-spectrum clocking to avoid concentrating electromagnetic interference at specific frequencies. Spread-spectrum clocking (SSC) is simply low frequency modulation of the clock.

Figure 36 shows a 2.5 Gb/s signal with 33 kHz triangle-wave modulation. The top two diagrams show the effect of the spread spectrum clock on the signal. The crossing point of the eye-diagram, on the top left, is smeared, and the jitter trend, on the top right, shows triangle-wave modulation. By applying a model of the receiver PLL to the TIE data, on software within the scope[23], the bottom two diagrams show that the recovered clock tracks the jitter induced by the spread spectrum clock yielding the appearance of an open eye.

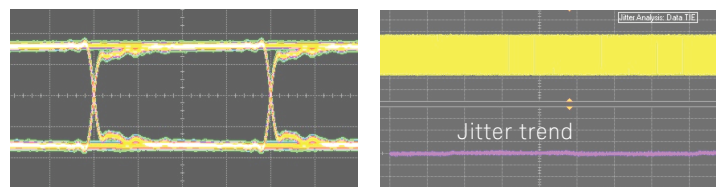Signal with spread spectrum clock
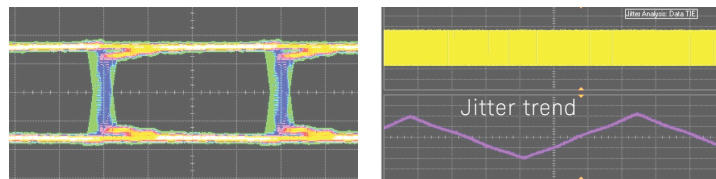
+ PLL transfer function

Figure 36. TIE analysis on a real-time oscilloscope

Phase noise analysis

Thorough analysis of a clock signal requires femtosecond accuracy which can only be achieved by a phase noise analyzer. Phase noise analysis provides two key measurements, frequency domain, $S_\phi(f_\phi)$ and time domain, $\phi(t)$, which harbor all the phase information of the clock up to the limit of the phase detector bandwidth.

RJ analysis on a phase noise analyzer

Two important goals can be achieved by analyzing RJ on a phase noise analyzer. First, by integrating the RJ spectrum, the width of the corresponding RJ Gaussian distribution is extracted within the bandwidth of interest,

$$\sigma = \sqrt{\int_{f_1}^{f_2} S_\phi(f)\,\mathrm{d}f}$$

Second, as described above in reference to slide 0, the major causes of RJ can be isolated by analyzing the power-series behavior of $S_\phi(f_\phi)$,

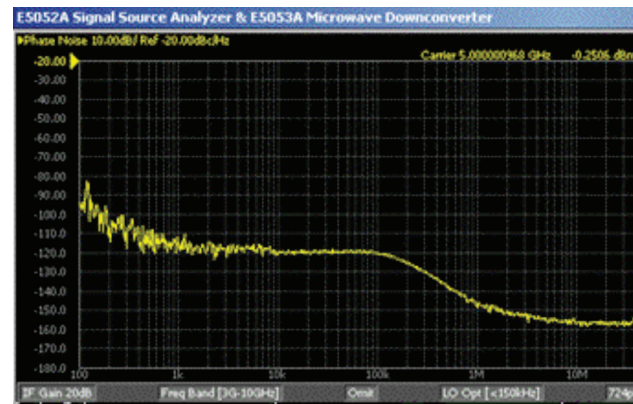$$S_\phi(f_\phi) = \Sigma \ \frac{Constant_n}{f_\phi^n}$$

Figure 37. RJ analysis of a phase noise analyzer

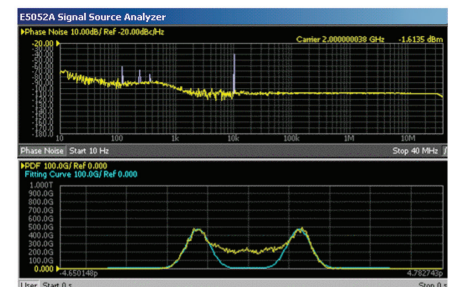## PJ on a phase noise analyzer

PJ causes sharp spurs in the phase noise spectrum. Knowledge of the PJ frequencies is a terrific tool for diagnosing problems. The time domain view shows how the combination of RJ and PJ smear the crossing point and cause errors. It also allows extraction of the clock DJ which is required for compliance by some specifications.

In the offset-frequency domain:
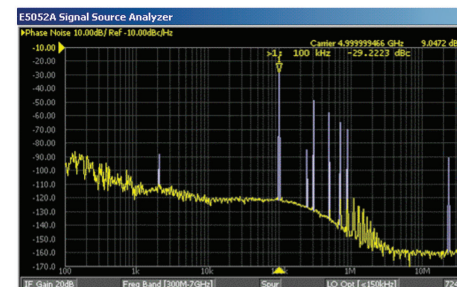
In the time domain:

Spurs in $S\phi(f)$
$\rightarrow$Isolate PJ frequency and amplitude

Jitter histogram from $\phi(t)$
$\rightarrow$Evaluate effect of PJ on TJ(BER)

Figure 38. PJ on a phase noise analyzer

Bandwidth limitations of phase noise analysis

The bandwidth of the phase noise analyzer, and hence $S_\phi(f_\phi)$, is limited to the bandwidth of the phase detector. This means that the RJ measurement cannot include RJ contributions out to Nyquist which is required in most serial data compliance specifications. This limitation is something you should be aware of, though it usually doesn't pose a problem. As is obvious from its inverse power-law nature, the phase noise drops precipitously with offset frequency. The white noise above the phase detector bandwidth integrates to a negligible fraction of the RJ closer to the carrier. In any case, if the white noise is substantial out to the bandwidth-limit of the phase detector, then the RJ generated would exceed any compliance specification.

The same problem occurs with PJ spurs. The phase detector bandwidth is certainly sufficient to observe spurs due to problems with the oscillator itself such as shock, vibration, etc, but may not be sufficient to detect all spurs that could be caused by the clock circuitry or pickup from, for example, power switching.
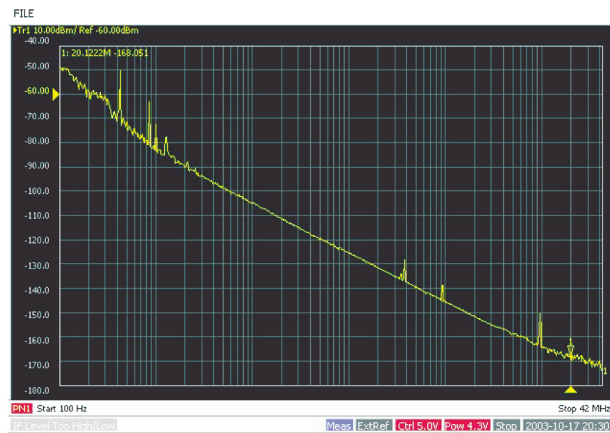

Figure 39. Bandwidth is limited to the bandwidth of the phase detector

Emulate the PLL response

Figure 40 illustrates the effect of a PLL response function applied directly to the phase noise signal, $\phi(t)$. The jitter transfer function is what's left over after the clock recovery response is applied. If H(s) is the clock recovery transfer, then $1 - H(s)$ is the jitter transfer function. By applying the jitter transfer function to the phase noise spectrum, we're left with just that phase noise which can affect the system. You can see how the low frequency jitter is suppressed. The ability to analyze just that phase noise which can affect the BER is a powerful tool.
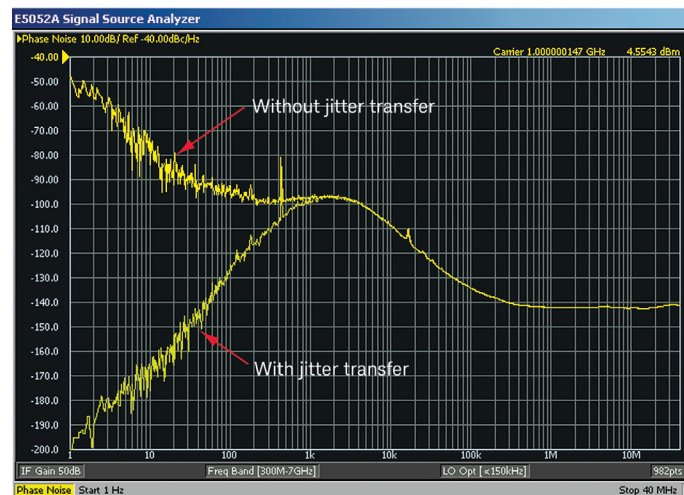

Figure 40. PLL response function applied directly to the phase noise signal, $\phi(t)$

Pros and cons of TIE and phase noise analyses

There are several other advantages to phase noise analysis. First, RJ can be analyzed over different bandwidths and its sources identified. Second, Periodic jitter is easy to identify as spurs in the phase noise spectrum. And, third, the transmitter and receiver response to the clock can be observed by applying mathematical filters directly to the phase noise signal, $\phi(t)$.

The disadvantage is that the phase noise analysis is band-limited. While the input signal bandwidth of a phase noise analyzer can be much higher than is available for a real-time oscilloscope, the offset frequency is limited by the bandwidth of the phase detector, typically 50 MHz. The TIE data set covers bandwidths up to Nyquist, but down to an offset frequency that depends on the memory depth for a real-time oscilloscope, or the sampling rate for an equivalent-time oscilloscope.

It's useful to keep in mind that phase noise analyzers can only be used on clock signals. The only way they can be used to analyze data signals is if the signal is first passed though a clock recovery circuit with a wide bandwidth and extremely flat jitter transfer.

Phase noise analysis has by far the lowest noise floor
  – TIE has nearly unlimited flexibility
  – Both can measure and identify PJ
  – Phase noise measures RJ in different frequency bands
  – TIE measures RJ up to the Nyquist frequency
  – Phase noise measures RJ down to 1 Hz
  – Both can apply transmitter/receiver response models
  – Phase noise can only be applied to a clock signal

The tools for clock-jitter analysis

Below is a list of specific equipment for clock–jitter analysis with particular emphasis on a comparison of phase noise analyzers and oscilloscopes.

Phase noise analyzers - SSA
  – Keysight E5052A signal source analyzer with precision clock jitter analysis software, E5001A, SSA-J
Real-time oscilloscopes
  – Keysight 80000 Series Infiniium oscilloscopes with E2688A serial data analysis and EZJIT+ software
Equivalent-time sampling oscilloscopes – DCA
  – Keysight 86100C Infiniium digital communications analyzer, DCA-J
Spectrum analyzers
  – Keysight E4440 series PSA

Capabilities of different clock-jitter analysis equipment

The fundamental difference between measurements performed on oscilloscopes and phase noise analyzers, in addition to the bandwidth issues discussed above, are the noise floor and dynamic range.
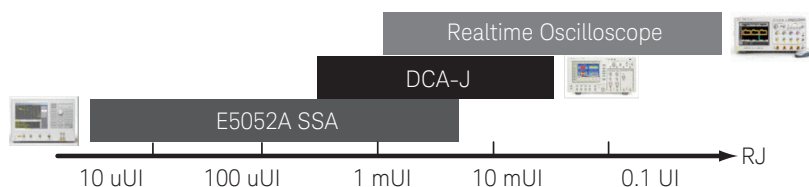

Figure 41. Range for jitter measurements by different types of equipment

The phase noise analyzer, Keysight's E5052A Signal Source Analyzer (SSA), has by far the lowest jitter noise floor[24]. At tens of femtoseconds it is an order of magnitude lower than the Keysight 86100C DCA[25] which, in turn, has a noise floor lower than the Keysight DSO81304B real-time oscilloscope[21].

SSA ~ 10 fs < equiv-time scope ~ 300 fs < real-time scope ~ 2 ps

For a variety of reasons, the dynamic range, or jitter ceiling, has the opposite order. The dynamic range of a real-time oscilloscope is nearly arbitrarily large. The DCA is limited to jitter that is an appreciable fraction of a UI because of technique limitations, and the phase noise analyzer, SSA range, is limited by the stability of its internal VCO, almost 10 mUI.

real-time scope ~ multi-UI > equiv-time scope ~ 1/4 UI > SSA ~ 8 mUI

Conclusion clock jitter

The embedded clock used in serial data systems reduces the effect of jitter on the BER by first reducing the jitter of the transmitted data by using a small bandwidth clock-multiplier at the transmitter and, second, by using a wide bandwidth clock recovery circuit at the receiver. The result is that the receiver tracks much of the jitter on the data.

For serial data applications the primary goal of clock–jitter analysis is to determine the effect that the jitter of the reference clock has on the bit error ratio (BER) of the system. The most accurate approach is to apply the transfer functions of the worst case transmitter and receiver for the application to the clock and measure the resulting clock RJ and DJ. These can be combined with the RJ and DJ of the other system components to estimate the maximum likely system TJ(BER) as well as to budget the jitter to the four major system components: the transmitter, the transmission channel, the receiver, and, indeed, the reference clock.

Jitter propagation on serial signals – the tools

Keysight Technologies provides an exhaustive set of tools for jitter analysis on serial data systems.

Figure 42. Jitter analysis tools for serial data systems
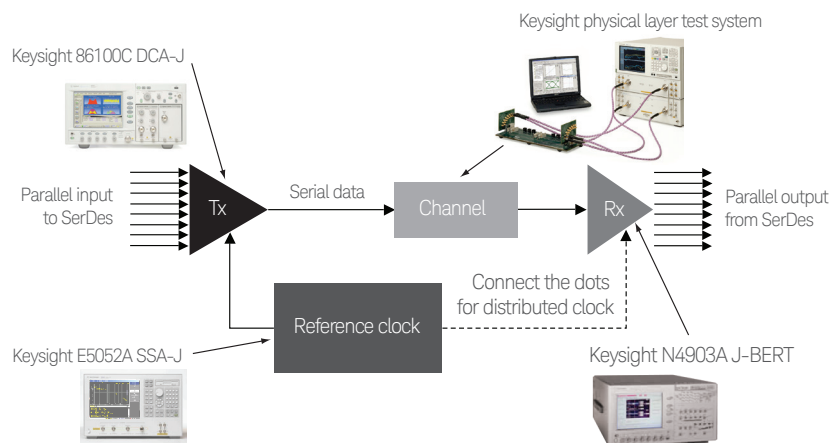
References

1. NIST Technical Note 1337, "Characterization of Clocks and Oscillators," edited by D.B. Sullivan, D.W. Allan, D.A. Howe, F.L. Walls, 1990.
2. For a general reference, see Howard Johnson, *High speed signal propagation: advanced black magic*, Prentice Hall, 2003; H.W. Johnson and M. Graham, High Speed Digital Design, Prentice Hall, 1993.
3. Ransom Stephens, *The rules of jitter analysis*, http://www.analogzone.com/nett0927.pdf, 2004; Ransom Stephens, *Analyzing Jitter at High Data Rates*, IEEE Communications Magazine, vol. 42, no. 2, February 2004, pp. 6-10; Keysight Technologies' Application Note, *Jitter Fundamentals: Keysight 81250 ParBERT Jitter Injection and Analysis Capabilities*, Literature Number 5988-9756EN, 2003.
4. Eugene Butkov, *Mathematical Physics*, Addison-Wesley, 1968.
5. See any standard probability and statistics text, for example, Anthony J. Hayter, *Probability and Statistics For Engineers and Scientists*, 2nd ed. Brooks/Cole Publishing, 2002.
6. Ransom Stephens, *Jitter analysis: The dual-Dirac model, RJ/DJ, and Q-scale*, Keysight Technologies Whitepaper, Literature Number, 5989-3206EN, 2005.
7. T11.2 / Project 1230 / Rev 10, *Fibre Channel - Methodologies for Jitter Specification*, www.t11.org, 1999.
8. W. P. Robins, *Phase Noise in Signal Sources Theory and Applications*, Peter Peregrinus Ltd., 1982.
9. F.M. Gardner, *Phaselock Techniques 3rd Edition*, New York: Wiley & Sons, 2005; Dan Wolaver, Phase-locked Loop Circuit Design, Prentice-Hall, 1991.
10. William R. Eisenstadt, Bob Stengel, Bruce M. Thompson, *Microwave Differential Circuit Design Using Mixed-Mode S-Parameters*, Artech House Publishers, 2006.
11. Behzad Razavi (editor), *Monolithic Phase-Locked Loops and Clock Recovery Circuits Theory and Design*, Wiley-IEEE Press, 1996.
12. John David Jackson, *Classical Electrodynamics 3rd Edition*, John Wiley & Sons, Inc., 1999.
13. L.D.Landau and E.M.Lifshitz, *Statistical Physics, Part 2, 3rd ed.*, Pergamon Press, 1980.
14. Emad Hegazi, Jacob Rael, Asad Abidi, *The Designer's Guide to High-Purity Oscillators*, Springer, 2004; George D. Vendelin, Design of Amplifiers and Oscillators by the S-Parameter Method, John Wiley and Sons, 1990;
15. Alan M. Kadin, *Introduction to Superconducting Circuits*, Wiley-Interscience, 1999.
16. Burkhard Schiek, Heinz-Juergen Siweris, Ilona Rolfes, *Noise in High-Frequency Circuits and Oscillators*, Wiley-Interscience, 2006.
17. J.A. Barnes and D.W. Allan, *A Statistical Model of Flicker Noise*, Proc. IEEE, vol. 43, pp. 176-178, February 1966; D. Halford, *A General Mechanical Model for |f|a Spectral Density Random Noise with Special Reference to Flicker Noise 1/|f|*, Proc. of the IEEE, vol. 56, no. 3, March 1968, pp. 215-258; Sullivan, D. B., et al. (eds.), *Characterization of Clocks and Oscillators*, National institute of Standards and Technology Technical Note 1337, National institute of Standards and Technology, Boulder, CO 80303-3328; R.B. Blackman and J.W. Tukey, *The Measurement of Power Spectra*, New York: Dover, 1958.
18. J.A. Barnes and D.W. Allan, *A Statistical Model of Flicker Noise*, Proc. IEEE, Vol. 43, pp. 176-178, February 1966; Chase P. Hearn, *The role of amplitude-to-phase conversion in the generation of oscillator flicker phase noise*, NASA technical memorandum, 1985.
19. Kroupa, V. F. (ed.), *Frequency Stability: Fundamentals and Measurement*, IEEE Press, New York, 1983; Hewlett-Packard, *Precision Frequency Measurements*, Application Note 116, July 1969; P.F. Panter, Modulation, Noise, and Spectral Analysis, McGraw-Hill, New York, 1965.
20. William H. Press, et al., *Numerical Recipes in C++ The Art of Scientific Computing*, Cambridge University Press, 2002.
21. Keysight Technologies Application Note, *Analyzing Jitter Using Keysight EZJIT Plus Software*, Literature Number AN-5989-3776EN, 2005.
22. Emmanuel Ifeachor and Barrie Jervis, *Digital Signal Processing: A Practical Approach*, Addison-Wesley, 1993.
23. Keysight Technologies Data Sheet, *High Speed Serial Data Analysis and Clock Recovery Software for Infiniium Oscilloscopes*, Literature Number 5989-0108EN, 2006.
24. Keysight Technologies Technical Overview, *Keysight E5052A Signal Source Analyzer with E5001A SSA-J Software Provides Precision Cock Jitter Analysis with Femtosecond Resolution*, Literature Number 5989-5040EN, 2006.
25. Keysight Technologies Product Note, 86100C-1, *Precision Jitter Analysis Using the Keysight 86100C DCA-J*, Literature Number 5989-1146EN, 2006.

Learn more at: www.keysight.com

For more information on Keysight Technologies' products, applications or services, please contact your local Keysight office. The complete list is available at:

www.keysight.com/find/contactus

KEYSIGHT
TECHNOLOGIES