



Agilent Technologies

Clock Jitter Analysis with femto-second resolution

presented by:

Agilent Technologies

© Copyright 2008 Agilent Technologies, Inc.

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.

Agenda

- **Jitter in serial data applications**
- **The role of reference clocks in serial data applications**
- **Reference clocks and phase noise**
- **Reference clock quality analysis**
- **The toolset**



In this seminar we concentrate on clock-jitter issues relevant to serial data systems.

After an introduction to the problems that jitter causes in serial data applications, the role of reference clocks and how their jitter affects the rest of a system is covered.

With the context and issues of reference clocks in place, a review of phase noise sets the stage for the discussion of techniques for evaluating clock quality with emphasis on emerging techniques for compliance testing.

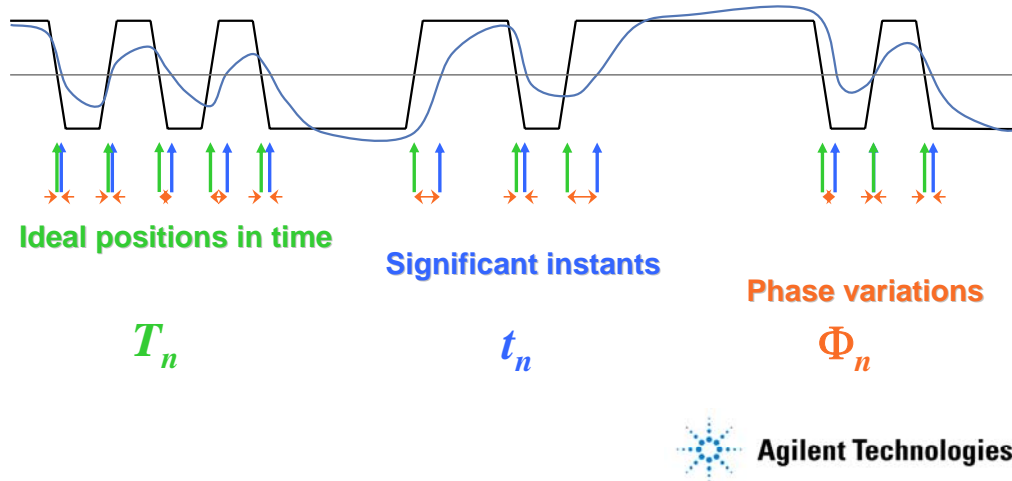
We conclude with a survey of jitter analysis equipment.

Agenda

- **Jitter in serial data applications**
- **The role of reference clocks in serial data applications**
- **Reference clocks and phase noise**
- **Reference clock quality analysis**
- **The toolset**

Jitter

- NIST: Jitter – the short term **phase variation** of the **significant instants** of a digital signal from their **ideal positions in time**.



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, we rarely need to bother with wander.

In the diagram, 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.

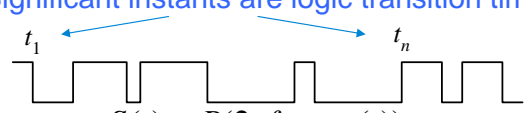
The “significant instances,” t_n , are those points where the actual analog waveform crosses the slice-threshold.

The jitter, or phase variations, Φ_n , are the difference between the two,

$$\Phi_n = t_n - T_n$$

[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.

Introduction to jitter

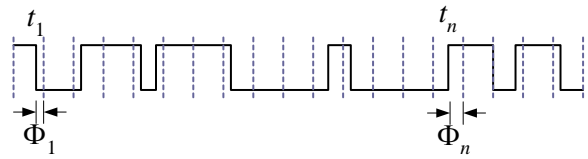
Consider a data signal:  Significant instants are logic transition times

signal $S(t) = P(2\pi f_d t + \varphi(t))$

data frequency

phase noise

If t_n is a transition time, then the **phase jitter** in seconds is: $\Phi_n = t_n - nT$

Phase jitter is the jitter of each edge: 

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

$$\Phi_n = \frac{\varphi(t_n) \text{ [rad]}}{2\pi f_d \text{ [sec]}}$$



It is both natural and accurate to think of the “significant instants” as the *logic transition times* or *edges*. It is convenient, though, as we will see, 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.

It is important to notice that jitter is a discrete quantity.

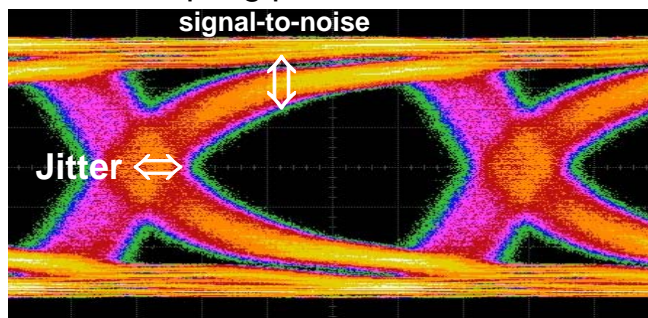
If we write down a general function for a signal, $S(t)$, as a pulse-train of logic values, P , then its argument, $2\pi f_d t + \varphi(t)$, shows how jitter gets in the system.

In an ideal system, each edge would be placed according to the data frequency of the signal, f_d , and the *phase noise* term, $\varphi(t)$, would be zero for all t . It is the phase noise term that causes jitter.

Phase noise, $\varphi(t)$, is a continuous function of time, but jitter (or phase jitter or cumulative jitter – whatever you want to call it), Φ_n , is the amount of phase noise at crossing times. As shown above, jitter can be written in terms of phase noise.

Why do we care about jitter?

- Just as a low **signal-to-noise ratio** causes errors when the signal fluctuates *vertically* across the sampling point...
- **Jitter** causes errors when the signal fluctuates *horizontally* across the sampling point



- The only reason we analyze jitter is to limit the Bit Error Ratio (BER)



We care about jitter for exactly the same reason that we care about 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 [i].

It is useful to always keep in mind that the only reason we analyze jitter is to limit the BER [iii].

[i] 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.

[iii] 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; Agilent Technologies' Application Note, "Jitter Fundamentals: Agilent 81250 ParBERT Jitter Injection and Analysis Capabilities," Literature Number AN-5988-9756EN, 2003.

Jitter caused by phase noise

Consider a clock signal

$$\text{ideal: } v_{ideal}(t) = v_0 \sin \omega t$$

$$\text{real: } v_{real}(t) = (v_0 + \Delta v(t)) \sin (\omega t + \varphi(t))$$

AM noise

PM noise

Phase noise term, $\varphi(t)$ shifts the signal horizontally.

→ phase noise is the primary cause of jitter in clocks

Amplitude noise can also cause jitter..

Clock jitter is dominated by phase noise.



If we consider a clock signal, we can distinguish amplitude noise and phase noise, or *amplitude modulation* and *phase modulation*. Both AM and PM can cause jitter. The phase noise term translates the periodic function, in this case a simple sinusoid, horizontally.

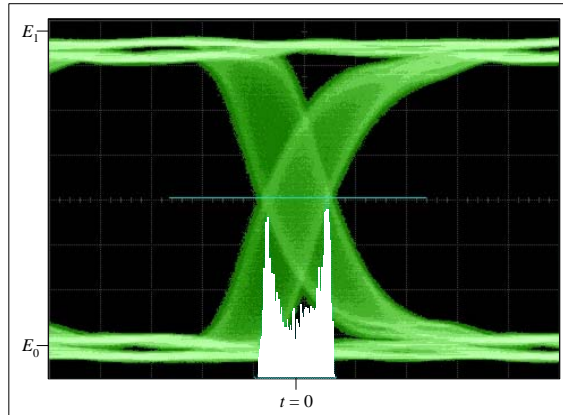
Amplitude noise can cause jitter. It is a second order effect in the sense that, if a signal has zero rise/fall time then amplitude noise can not cause jitter. Of course real signals have finite rise/fall time and, by moving the signal up or down, amplitude noise changes the times of logic transitions.

In most cases, clock-jitter is dominated by phase noise, but it is important to keep in mind that noise is noise. Jitter and voltage noise, while simple to describe as separate functions, are correlated. It is possible for elements of the voltage noise term, $\Delta v(t)$, to change the phase noise term, $\varphi(t)$.

The jitter probability density function

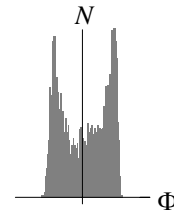
The probability of the timing of a given logic transition being displaced from the ideal time is described by the

Jitter Probability Density Function (PDF)



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

$$\Phi_n = t_n - nT_B$$



The jitter Probability Density Function (PDF) gives the probability for a given logic transition to differ from the ideal by an amount between Φ and $\Phi + d\Phi$.

One way to measure the jitter PDF, and certainly an easy way to think of it, is to make a histogram of the crossing point. Each entry in the histogram is the divergence of a given edge by an amount determined by the time-resolution, or bin-width, of the histogram.

Different sources of jitter combine through the mathematical process of convolution [1] to form the resulting jitter PDF.

The example shown in the graphic is the PDF for *sinusoidal* jitter.

[1] Eugene Butkov, *Mathematical Physics*, Addison-Wesley, 1968.

The characteristics of jitter

Random and Deterministic Jitter

Deterministic Jitter (DJ)

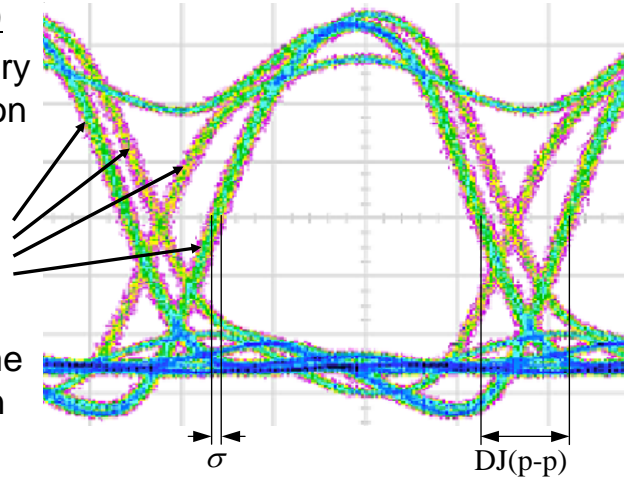
Determines the trajectory
of each logic transition

DJ(p-p)

Random Jitter (RJ)

Smears each trajectory
according to the same
Gaussian distribution

σ



In the analysis of serial data systems, it is useful to distinguish two categories of jitter, Random Jitter (RJ) and Deterministic Jitter (DJ).

This graphic is meant to give a conceptual understanding 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, which we denote, DJ(p-p), and RJ with the width, or rms value, of its distribution, σ (*sigma*).

The characteristics of jitter

Random jitter – RJ

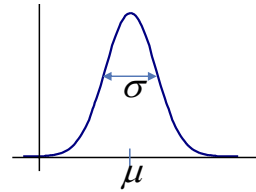
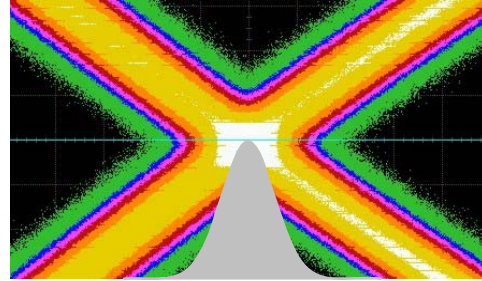
Many small processes

Thermal effects * variations in trace width * shot noise

→ The Central Limit Theorem: “The PDF of an infinite number of small independent random processes follows a Gaussian distribution”

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

$$\text{RJ} = \sigma$$



RJ is caused by the accumulation of a huge number of processes that each have very small magnitudes; things like thermal noise, variations in trace width, shot noise, etc.

The central limit theorem of probability and statistics [i] gives us a handle on how to describe RJ: The PDF of an infinite number of small independent random processes follows a Gaussian distribution.

As usual in engineering, we ignore some of the formalism. Many of the processes are not independent and some of them are not so small and, also as usual in engineering, we get a mathematical description that serves well. Though, of course, we should keep in mind that our mathematics is only as accurate as our assumptions.

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 does not have a well defined peak-to-peak value.

Since RJ is described by a Gaussian, the width, or standard deviation, σ (*sigma*), of the distribution, is sufficient to describe the magnitude of RJ.

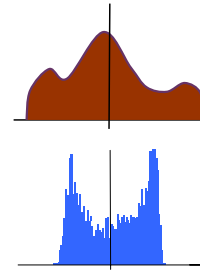
[i] See any standard probability and statistics text, for example, Anthony J. Hayter, *Probability and Statistics For Engineers and Scientists*, 2nd ed. (Brooks/Cole Publishing, 2002).

The characteristics of jitter

Deterministic Jitter – DJ

- Effects of a few processes
 - Electromagnetic interference * reflections * channel frequency response
- Combines to form bounded distributions of varying shapes

$$DJ = DJ(p-p)$$



The **net jitter PDF** is the convolution of RJ and DJ:

$$J = RJ * DJ$$

Therefore, the jitter PDF is unbounded.

→ peak-to-peak jitter is not well defined!



DJ is caused by a comparatively small number of processes that need not be independent and may have large magnitude.

It is called “deterministic” jitter because, in principle, if we knew everything there is to know about a system, we 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).

The (actual or net) 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.

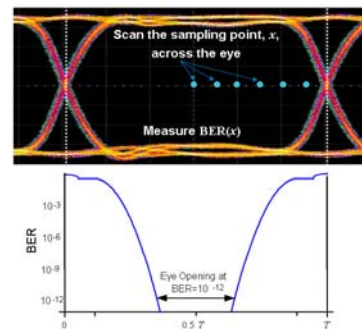
Total Jitter and TJ(BER)

What can we use to compare the jitter performance of different system components?

- Peak-to-peak jitter is not well defined so...

We define

TJ(BER) is the p-p value of the jitter distribution defined in terms of the BER (bit error ratio)



$$TJ(BER) \equiv T - (\text{Eye Opening at } BER)$$



To compare the performance of different components, including clocks, we need a single quantity that tells us 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 does not meet this need. We therefore define Total Jitter at a Bit Error Ratio, TJ(BER).

TJ(BER) is defined as the amount of eye closure at a given BER [i]. It is easiest to describe in terms of a BERTscan or bathtub plot [ii].

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.

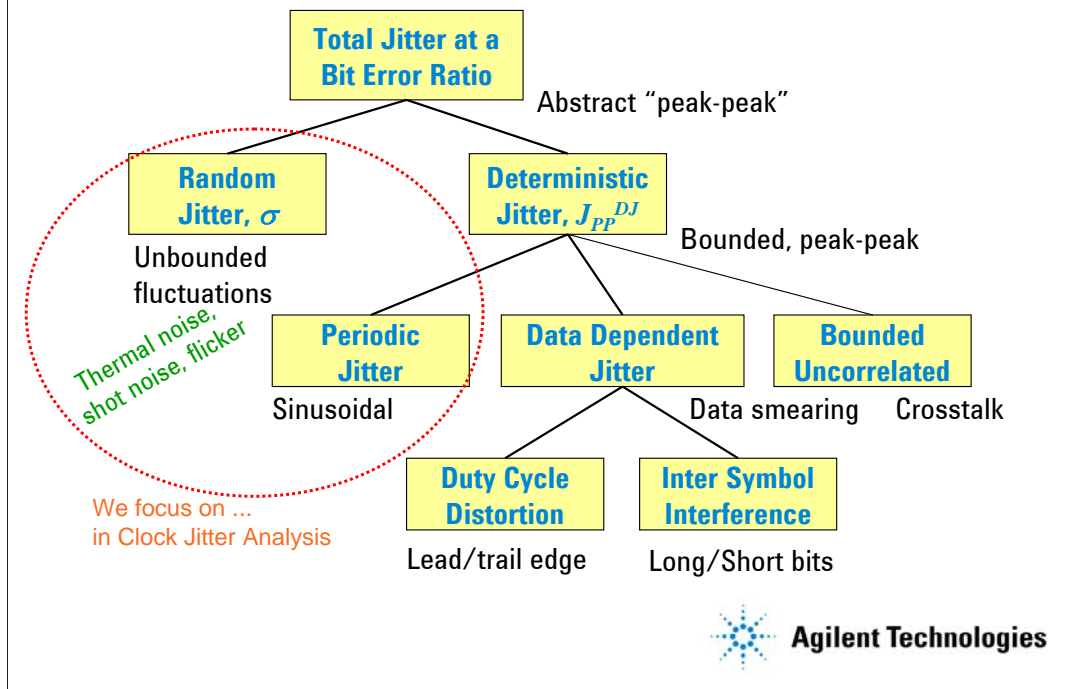
As the sampling point moves toward the eye center, the BER drops very fast. Similarly, but in reverse, as the sampling point approaches the opposite crossing point, BER increases rapidly. The *eye opening* at a given BER is the distance between the two slopes of $BER(x)$ at that BER.

TJ(BER) is the *eye closure*, which is the width of the eye minus the eye opening.

[i] Ransom Stephens, "Jitter analysis: The dual-Dirac model, RJ/DJ, and Q-scale," Agilent Technologies Whitepaper, Literature Number , 5989-3206EN, 2005.

[ii] T11.2 / Project 1230 / Rev 10, "Fibre Channel – Methodologies for Jitter Specification," www.t11.org, 1999.

Jitter categories



Here is a larger breakdown of jitter including the components of DJ and a summary of their interrelationships.

In this seminar we are focusing on Random Jitter and Periodic Jitter since Data Dependent Jitter and Crosstalk are not observed in clock jitter analysis usually.

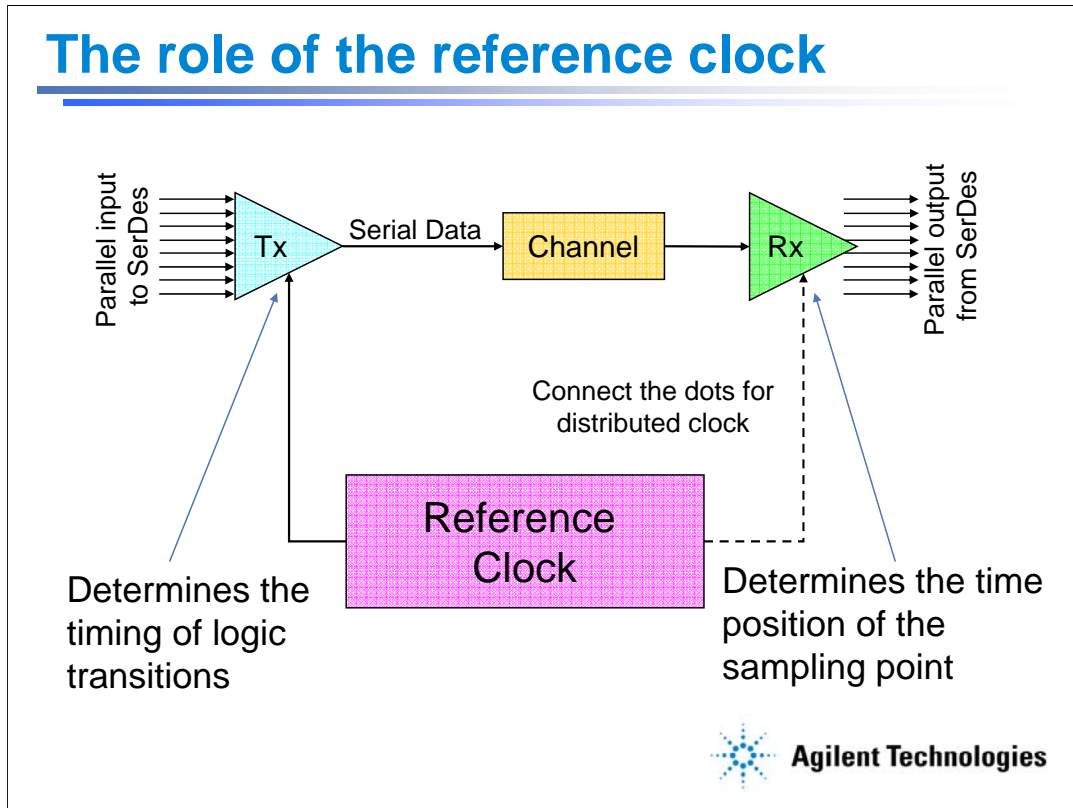
Agenda

- Jitter in serial data applications
- **The role of reference clocks in serial data applications**
- Reference clocks and phase noise
- Reference clock quality analysis
- The toolset



In this section we narrow our focus to the role of reference clocks in serial data applications.

The role of the reference clock



In the system diagram, 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.

In this section, we will examine how the clock-jitter is propagated through each part of the system, particularly in the transmitter (Tx) and the receiver (Rx).

what type of jitter each component is most likely to generate:

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.

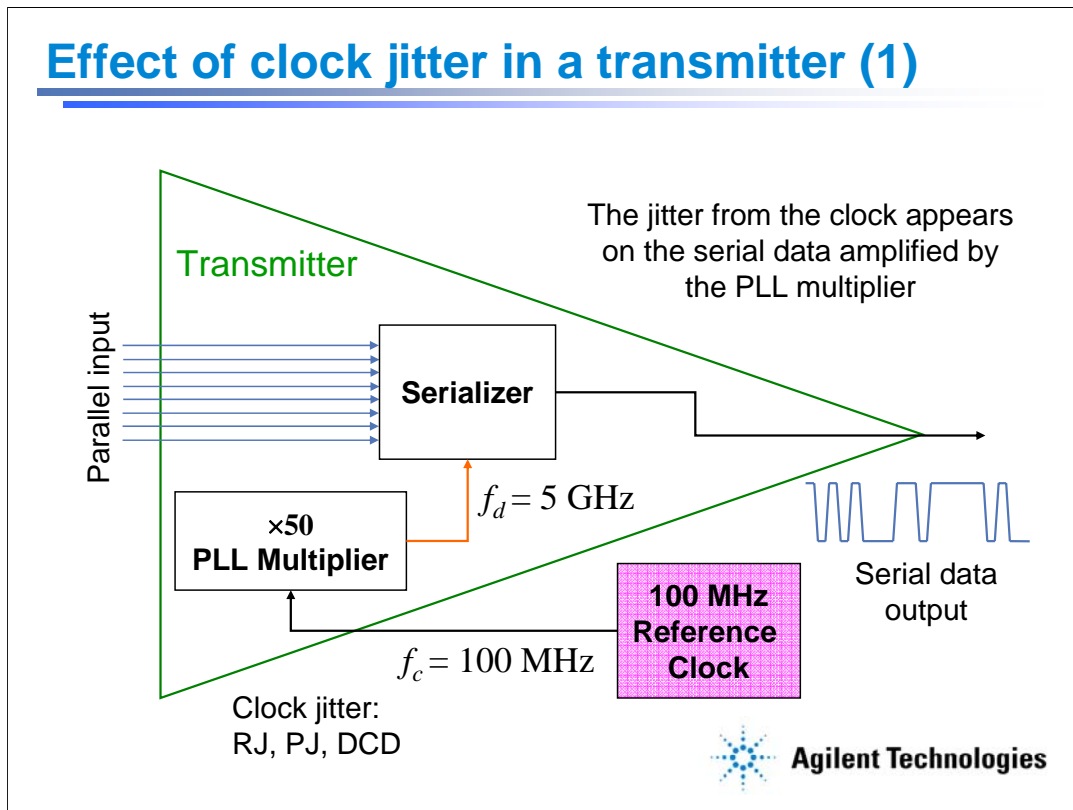
The transmitter contributes RJ from thermal effects, Inter-Symbol Interference (ISI) from the frequency response of internal transmission lines, and PJ from pickup of EMI.

The frequency response and attenuation characteristics of the transmission channel cause ISI and, if there is DCD on the incoming signal, Data Dependent Jitter (DDJ).

The receiver generates RJ from shot noise, and DDJ from internal circuitry. Since the receiver identifies the logic values of the signal, we do not so much care what jitter it introduces as whether or not it can correctly identify the bits.

PJ and, another category called Bounded Uncorrelated Jitter (BUJ) can be introduced through the EMI of other circuit elements. BUJ is the category of DJ that we use to conceal our ignorance. While it is 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.

Effect of clock jitter in a transmitter (1)



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 Gbps output signal, the transmitter would use a PLL to multiply the reference clock by a factor of fifty (x50).

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 [1] by a factor of n is to multiply the phase noise power to carrier ratio by n^2 . The jitter goes up fast!

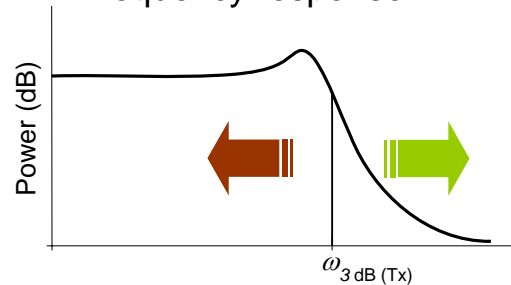
[1] W. P. Robins, *Phase Noise in Signal Sources Theory and Applications*, (Peter Peregrinus Ltd., 1982).

Effect of clock jitter in a transmitter (2)

The PLL multiplier has three important characteristics

1. The multiplying act amplifies the clock jitter.
2. Its VCO introduces more RJ and nonlinearities in its circuitry can introduce more DCD.
3. **There is a bandwidth associated with the PLL.**

Typical 2nd order PLL frequency response:



What clock jitter matters?



The PLL multiplier in the transmitter has a certain frequency response [1], 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.

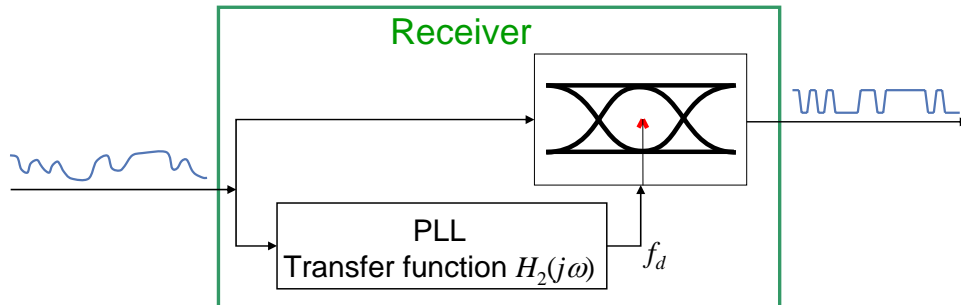
Of course, zero bandwidth means infinite lock time, so we have to compromise, but the narrower the PLL bandwidth, the less jitter from the reference clock makes it into the data.

Determining whether or not a clock will function in a system at the desired BER requires careful testing of the jitter frequency spectrum.

[1] F.M. Gardner, *Phaselock Techniques 3rd Edition*, New York: Wiley & Sons, 2005; Dan Wolaver, *Phase-locked Loop Circuit Design*, Prentice-Hall, 1991.

Role of the clock in a receiver (1)

PLL-based clock recovery



The PLL recovers a f_d clock from the data

→ Defines the sampling point

For zero bandwidth PLL any data jitter could cause errors.

For an infinite bandwidth PLL no data jitter could cause errors.



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 [1].

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 would not be any errors. It is in this sense that we say the recovered clock *tracks* data jitter.

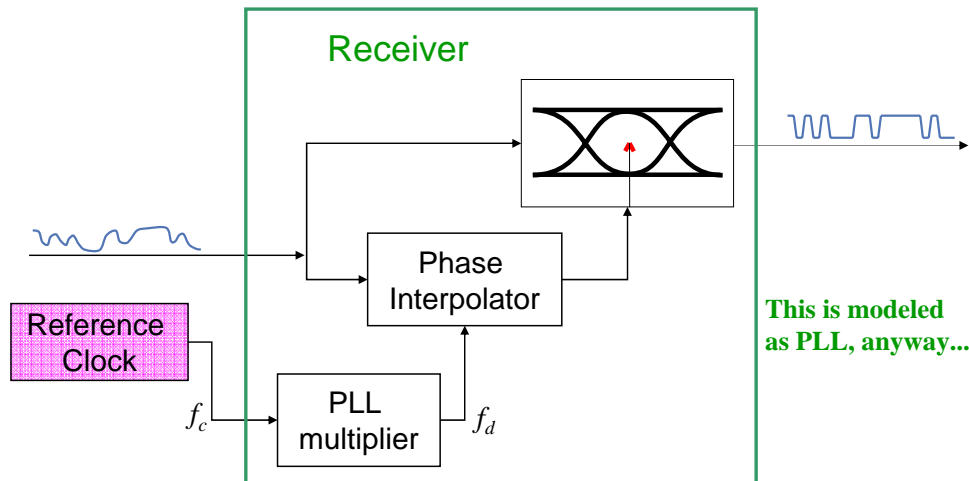
In the opposite extreme, a zero bandwidth CR circuit, the sampling point would be fixed at those "ideal positions in time." Jitter at any frequency on the data could cause logic transitions to fluctuate across the sampling point and cause errors.

Of course it is 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.

[1] Behzad Razavi (editor), *Monolithic Phase-Locked Loops and Clock Recovery Circuits Theory and Design*, Wiley-IEEE Press, 1996.

Role of the clock in a receiver (2)

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 is not as easy to model as that of a PLL.

In the absence of specific data, they are usually modeled as PLLs anyway.

Role of the clock in a receiver (3)

Frequency response of the clock recovery - a typical transfer function -

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

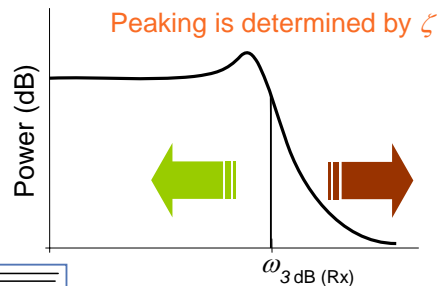
ω_n is the natural frequency

ζ is the damping factor

s is the Laplace variable

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

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



The clock recovery transfer function can be modeled by a second order PLL as shown.

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.

Notice two things in this graphic.

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 does not track the corresponding data jitter, it also increases the system BER.

.. then how do you analyze clock jitter ?

1. Determine the limiting requirements of the specific system.
→ e.g., PCI-Express, FBD, sATA, Fiber-Channel, ..
2. Apply the limiting case transmitter and receiver transfer functions to the clock.
3. Analyze the resulting jitter to determine the effect of clock jitter on the BER.



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.

Before we dive into analysis, we need to review what clocks are, how they work, and their characteristics.

Agenda

- Jitter in serial data applications
- The role of reference clocks in serial data applications
- **Reference clocks and phase noise**
- Reference clock quality analysis
- The toolset



We discuss the SSB spectrum, how to distinguish phase noise from amplitude noise, and, once again, the relationship of phase noise and jitter in this section.

Oscillator noise

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

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

Amplitude noise

Phase noise

- **Sources of Noise**
 - Temperature, pressure, humidity – change frequency
 - Vibration – causes spurs
 - Electromagnetic fields – change the frequency, can cause spurs
- **Properties of Noise**
 - Phase noise cannot be eliminated by a limiter.
 - Noise close to the carrier cannot be eliminated by filtering.



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 [i]. 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 [ii].

Thermal noise, 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.

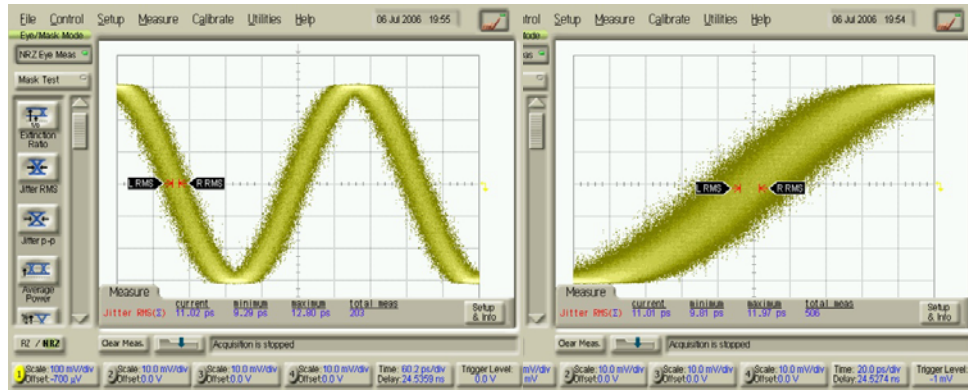
Unfortunately phase noise can not be eliminated by a limiting-amplifier and, since so much of the noise is close to the carrier, it can not be eliminated by filtering.

[i] Alan M. Kadin, *Introduction to Superconducting Circuits*, Wiley-Interscience, 1999.

[ii] Burkhard Schiek, Heinz-Juergen Siweris, Ilona Rolfes, *Noise in High-Frequency Circuits and Oscillators*, Wiley-Interscience, 2006.

Clock signal / Time domain view

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



A clock signal is shown here.

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 an Agilent 86100C DCA equivalent-time sampling oscilloscope showing the sinusoidal envelope of the signal.

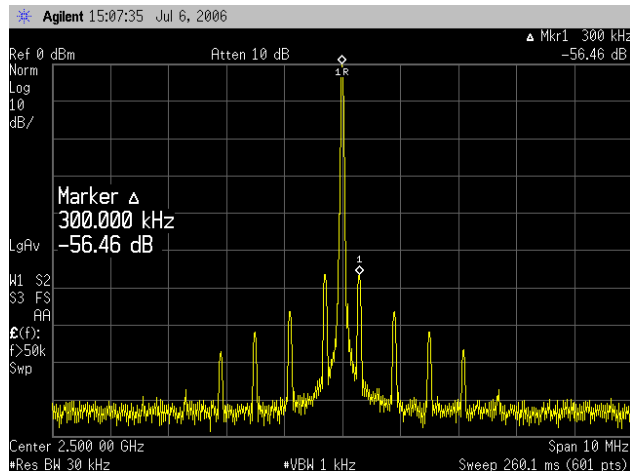
By zooming in to the slice-threshold, on the right, the expanse of jitter is easy to see.

Clock signal / Frequency domain View

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

Spectrum analyzer
plots the
Frequency Spectral
Density, $S(f)$

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



This is the same clock signal in the frequency domain.

The graphic was taken by an Agilent E4440 Performance Spectrum Analyzer. The frequency spectral density is a measure of the amount of power per unit frequency in the signal.

Mathematically, it is 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.

Clock signal

Phase noise in the frequency domain

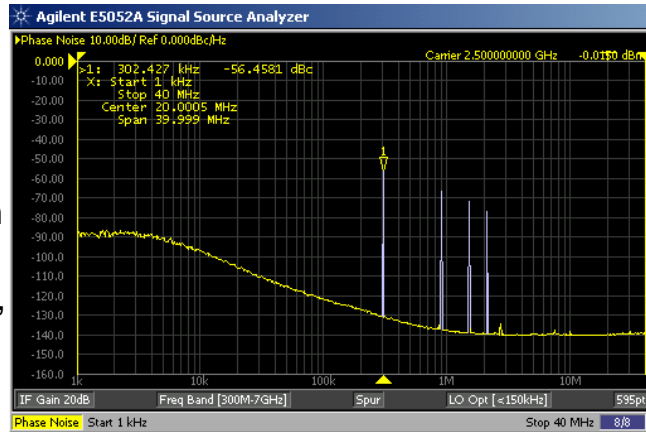
Phase noise analyzer plots the

Phase Spectral Density, $S_\varphi(f_\varphi)$

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

$$f_\varphi = f - f_c$$

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



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 is 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, $S\varphi(f)$, is the square of the Fourier transform of the phase noise term. They are different frequency domains and different functions.

As we will see below, the phase noise frequency domain, f_φ , is related to the signal frequency domain, f , through the offset frequency expression, $f_\varphi = f - f_c$. We will also see that the phase noise spectral density is related to the single-sideband spectrum, $L(f) \approx \frac{1}{2} S\varphi(f)$.

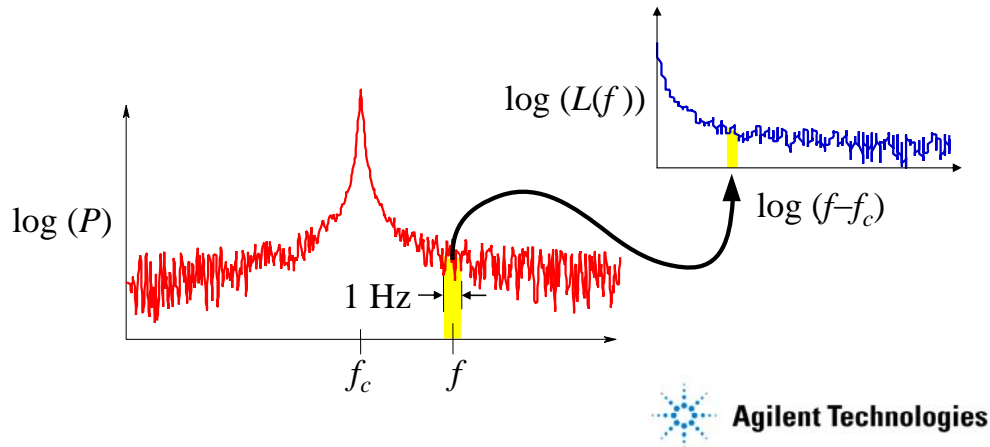
Due to this relationship which is significant for many historical reasons, the display shown above is actually $\frac{1}{2} S\varphi(f)$, that is, it is 3 dB less than $S\varphi(f)$.

Jitter on clocks

Single side band noise spectrum, $L(f)$

SSB is extracted from the frequency domain signal:

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



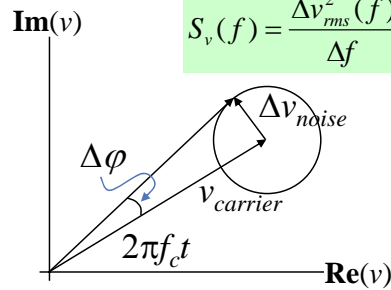
Another way to analyze oscillator noise is to extract the Single Side Band (SSB) noise spectrum, $L(f\phi)$.

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 is not uncommon for a spectrum analyzer to have software that can extract the SSB spectrum.

The SSB spectrum and the phase spectral density

$$\begin{aligned}
 L(f) &= \frac{1}{2P_C} \frac{\Delta P(f_\phi)}{\Delta f_\phi} \\
 &= \frac{1}{\Delta f_\phi} \frac{\frac{1}{2} \Delta v_{Noise\ rms}^2 / R}{v_{Carrier}^2 / R} \\
 &= \frac{1}{2\Delta f_\phi} \frac{\Delta v_{Noise\ rms}^2}{v_{Carrier}^2} \\
 &= \frac{\Delta v_{rms}^2}{2\Delta f} = \frac{1}{2} S_v(f) \approx \frac{\Delta \phi_{rms}^2}{2\Delta f_\phi} = \frac{1}{2} S_\phi(f_\phi)
 \end{aligned}$$



Voltage Spectral Density

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

Phase Spectral Density

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

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

(if $|\Delta\phi|$ is small enough.)



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 lets have a closer look anyway so we can see just how similar $L(f_\phi)$ and $S_\phi(f_\phi)$ are.

First, use Ohm's law to convert power to voltage, $P = v^2/R$. Then cancel common terms and identify the voltage spectral density, $S_v(f)$.

In the last step, if we assume negligible amplitude noise, 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.

Radians and decibels are not dimensions in the same sense as meters, kilograms, seconds, or Coulombs. In the bottom equation, dBc and radians are included as reminders.

Equating the different terms is not a contradiction.

Phase noise and jitter

- Clock signal: $v(t) = (v_0 + \Delta v(t)) \sin(2\pi f t + \varphi(t))$
- Jitter is “the short term **phase variation** of the **significant instants** of a digital signal from their **ideal positions in time.**”
- Phase noise is continuous $\varphi(t)$
- Jitter is discrete Φ_n
- Jitter is the phase noise at each slice threshold

$$\begin{aligned}\Phi_n &= t_n - nT \\ &= \frac{\varphi(t_n)}{2\pi f_c}\end{aligned}$$



Recall that jitter is “the short term phase variation of the significant instants of a digital signal from their ideal positions in time.”

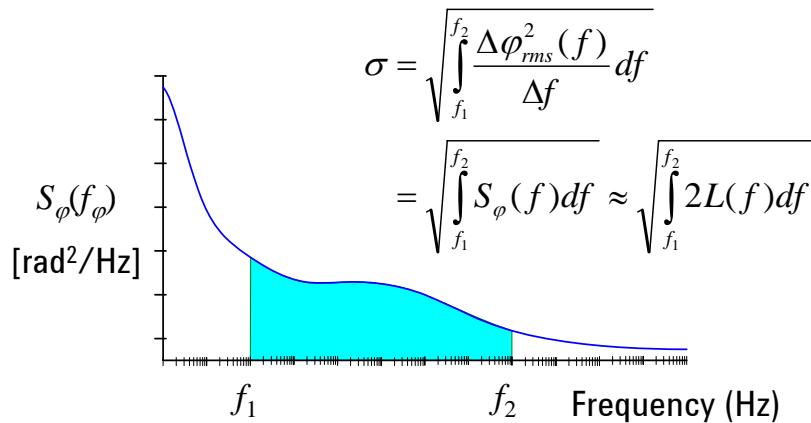
Phase noise is a continuous function of time that indicates the deviation of a digital signal’s phase from the ideal phase.

Jitter is the discrete difference between the actual time that a logic signal crosses the slice threshold and the ideal time. Thus, jitter is proportional to the phase noise at each slice threshold.

We should add that jitter, Φ_n , can be expressed in seconds, as it is here, 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

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



A useful result is that we can calculate RJ from the phase spectral density. $S_\varphi(f_\varphi)$ 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 we can derive RJ 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. We will revisit this issue shortly.

Agenda

- Jitter in serial data applications
- The role of reference clocks in serial data applications
- Reference clocks and phase noise
- **Reference clock quality analysis**
- The toolset



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

- Traditional clock specifications
 - Phase, period, cycle-to-cycle jitter
 - .. do not answer the fundamental question:
Will the clock work in the system?
 - Only care about jitter that can cause errors
 - Need to analyze clock-jitter under application-specific specifications
 - Transmitter: multiplier bandwidth
 - Receiver: clock recovery bandwidth
- How do we read a clock data sheet?



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 do not answer the only truly relevant question: Will the clock work in the system I am designing?

We only care about the jitter that can cause errors and, as we have 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.

Quantities quoted on clock data sheets

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

The ranges of values in this table are typical for data sheets of clocks used in serial data applications of 1Gbps and higher.

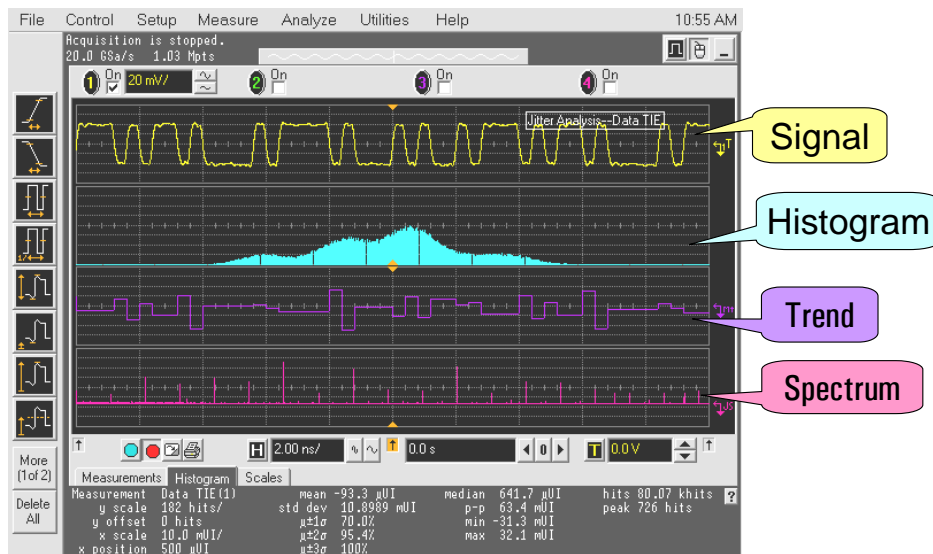


The ranges of values in this table are typical for data sheets of clocks used in serial data applications of 1 Gbps and higher.

Note that shaded lines need femto-second resolution in jitter measurement.

TIE analysis on a real-time oscilloscope

TIE



(TIE: Time Interval Error)



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 and so is not a truly 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 should not 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 [1].

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.

[1] William H. Press, et al., *Numerical Recipes in C++ The Art of Scientific Computing*, Cambridge University Press, 2002.

Jitter analysis on a Time Interval Error data set

TIE

Accumulate the crossing-point times, $\{ t(n) \}$, on a real-time oscilloscope, then...

Measure peak-to-peak or rms phase, period, cycle-to-cycle jitter:

$$\Delta t_{phase}(n) = t(n) - nT$$

Determine max or rms $[\Delta t_{phase}(n)]$

$$\Delta t_{period}(n) = [t(n) - nT] - [t(n-1) - (n-1)T] = t(n) - t(n-1) - T$$

Determine max or rms $[\Delta t_{period}(n)]$

$$\Delta t_{cyc-cyc}(n) = [t(n+1) - t(n)] - [t(n) - t(n-1)]$$

Determine max or rms $[\Delta t_{cyc-cyc}(n)]$



The TIE data set can be used to extract all of the phase, period, and cycle-to-cycle jitter as well as RJ and DJ [i].

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

Time Interval Error analysis

TIE

Model the performance of a clock in a system with TIE data

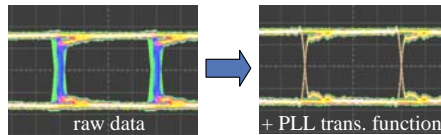
- Recall the second order PLL transfer function

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

- Use DSP techniques to apply $H(s)$ to the data
 - clock performance in the system!
 - RJ, DJ, and TJ(BER) contributions of the clock

Details.. The oscilloscope must have

- Sufficient Bandwidth to represent the signal frequency components
- Memory depth to acquire enough data that the DSP techniques are accurate
- Low noise



example: signal with spread spectrum clock



Agilent Technologies

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 [i] (DSP) can be used with impunity.

For example, the second-order PLL transfer function 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 not approach the sensitivity of a phase noise analyzer.

[i] Emmanuel Ifeachor and Barrie Jervis, *Digital Signal Processing: A Practical Approach*, Addison-Wesley, 1993.

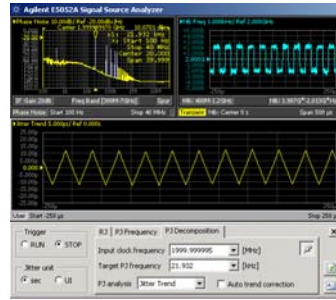
Phase Noise Analysis

P.N.

Clock analysis for high data rate systems requires
femto-second accuracy

Which can only be delivered by a phase noise analyzer
(or the Signal Source Analyzer)

Analyze in both the
frequency domain, $S_{\phi}(f)$
and time domain, $\phi(t)$



→ RJ, PJ, diagnostics
.. up to the phase detector bandwidth



Thorough analysis of a clock signal requires femto-second accuracy which can only be achieved by a phase noise analyzer (or signal source analyzer).

Phase noise analysis provides two key measurements, $S_{\phi}(f)$ and $\phi(t)$, 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

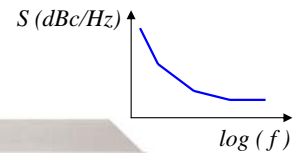
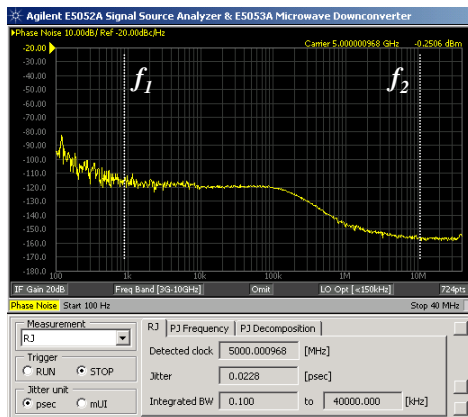
P.N.

Measure RJ_{rms} in desired bandwidths:

$$\sigma = \sqrt{\int_{f_1}^{f_2} S_{\varphi}(f) df}$$

Determine impact of Flicker, shock, vibration, temperature, noisy electronics, ...

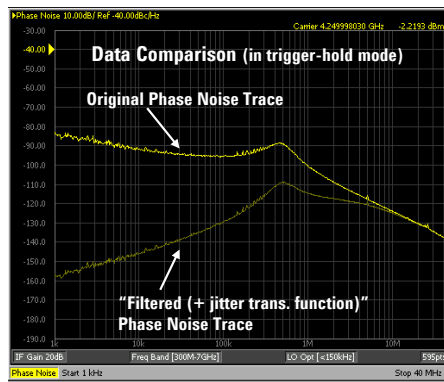
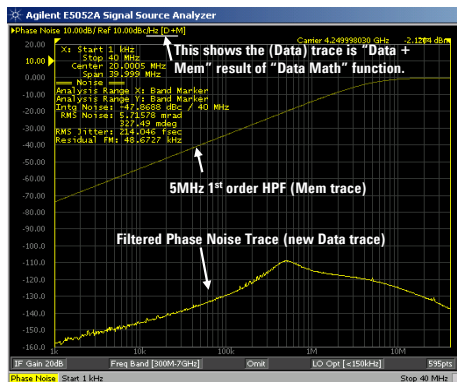
Random Noise Profile $S_{\varphi}(f_{\varphi}) = \sum \frac{Constant_n}{f_{\varphi}^n}$



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. Second, the major causes of RJ can be isolated by analyzing the power-series behavior of $S_{\varphi}(f_{\varphi})$.

Emulate jitter transfer function

P.N.



If $H(s)$ is a PLL transfer function (of a receiver or clock recovery circuits), then $1-H(s)$ is a jitter transfer function, JTF. In the offset frequency domain, it is very easy to emulate a specific JTF by filtering measured phase noise (spectral density) in a real time mode.



This is an illustration of the effect of a PLL response function applied directly to the phase noise signal, $\phi(t)$.

The jitter transfer function is what is left over after the clock recovery response is applied.

If $H(s)$ is the clock recovery transfer function, then $1 - H(s)$ is the jitter transfer function.

By applying the jitter transfer function to the phase noise spectrum, we are left with just that phase noise which can affect the system.

You can see how the low frequency jitter is suppressed.

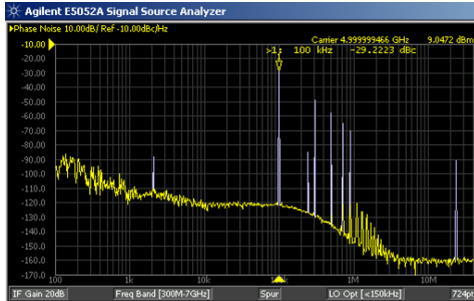
The ability to analyze just that phase noise which can affect the BER is a powerful tool.

PJ on a phase noise analyzer

P.N.

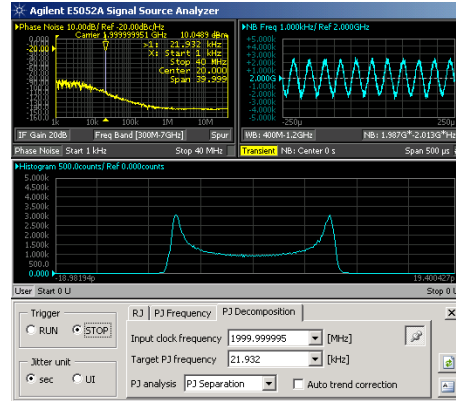
In the offset-frequency domain:

In the time domain:



Spurs in $S_{\varphi}(f)$

→ Isolate PJ frequency and amplitude



Jitter histogram from $\varphi(t)$

→ Evaluate effect of PJ on TJ(BER)



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.

Pros and cons of TIE and PN analyses

TIE

P.N.

Feature	TIE analysis	PN analysis
Noise floor	Good	Lowest floor
PJ measurement	Both can measure and identify PJ	
RJ measurement	Measurable up to the Nyquist frequency	Measurable in different frequency bands. Measurable down to 1Hz offset frequency
Transfer function	Both can apply Tx/Rx response models	
Application flexibility	Nearly unlimited flexibility	Only be applied to a clock signal



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, $\varphi(t)$.

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 50~100 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 is 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 through a clock recovery circuit with a wide bandwidth and extremely flat jitter transfer.

Agenda

- Jitter in serial data applications
- The role of reference clocks in serial data applications
- Reference clocks and phase noise
- Reference clock quality analysis
- **The toolset**



We now turn to specific equipment for clock-jitter analysis with particular emphasis on a comparison of phase noise analyzers and oscilloscopes.

The tools for clock-jitter analysis

Phase noise analyzers – SSA (Signal Source Analyzer)

- Agilent E5052B Signal Source Analyzer with precision clock jitter analysis software, E5001A (SSA-J)

Real-time oscilloscopes

- Agilent 80000 Series Infiniium oscilloscopes with E2688A Serial Data Analysis and EZJIT+ software

Equivalent-time sampling oscilloscopes – DCA

- Agilent 86100C Digital Communications Analyzer, DCA-J

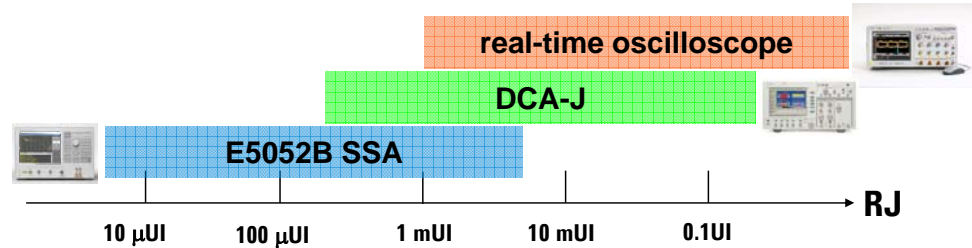
Spectrum Analyzer – PSA

- Agilent E4440 series Performance Spectrum Analyzers, PSA



Capabilities of different clock jitter analysis equipment

Range for jitter measurements by different types of equipment



Fundamental differences

Noise floor

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

Dynamic range

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



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.

The phase noise analyzer, Agilent E5052B Signal Source Analyzer (SSA), has by far the lowest jitter noise floor [i]. At tens of femto-seconds it is an order of magnitude lower than the Agilent 86100C DCA [ii] which, in turn, has a noise floor lower than the Agilent DSO81304B real-time oscilloscope [iii].

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.

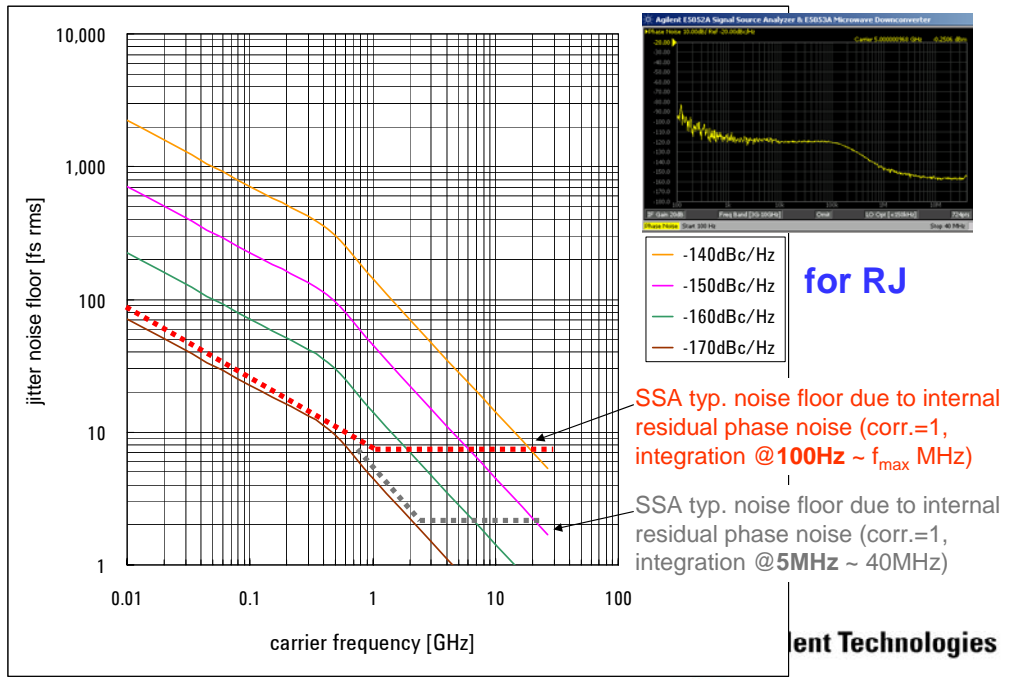
The DCA is limited to jitter that is an appreciable fraction of a UI because of technique limitations, and the phase noise analyzer, SSA range, is limited by the stability of its internal VCO, almost 10 mUI.

[i] Agilent Technologies Technical Overview, "Agilent E5052A Signal Source Analyzer with E5001A SSA-J Software Provides Precision Clock Jitter Analysis with Femto-second Resolution," Literature Number 5989-5040EN, 2006.

[ii] Agilent Technologies Product Note, 86100C-1, "Precision Jitter Analysis Using the Agilent 86100C DCA-J," Literature Number 5989-1146EN, 2006.

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

Jitter noise floor vs. Carrier frequency (1)

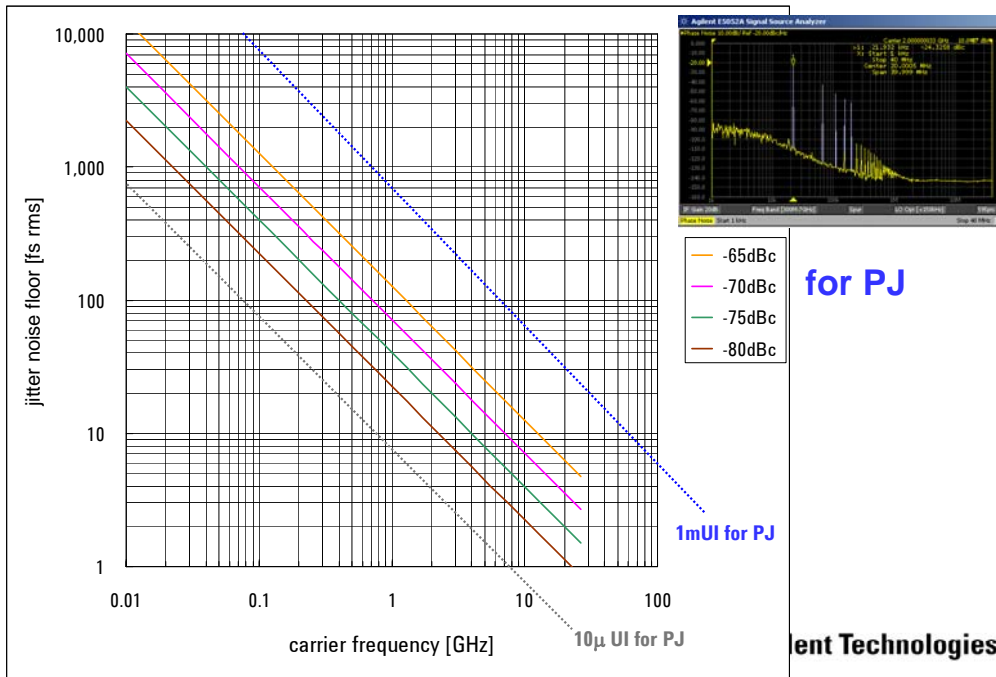


The relationship between jitter noise floor and carrier frequencies with several levels of random jitter (-dBc/Hz) is shown in the graph.

The minimum noise floor of the SSA (Signal Sour Analyzer) in each case depends on a noise bandwidth set up in measurement.

Assuming that your clock signal is 5GHz with -140dBc/Hz white noise of phase fluctuation, then you can see about 30 femto-second rms jitter (from the above graph) in random jitter measurement by the SSA.

Jitter noise floor vs. Carrier frequency (2)

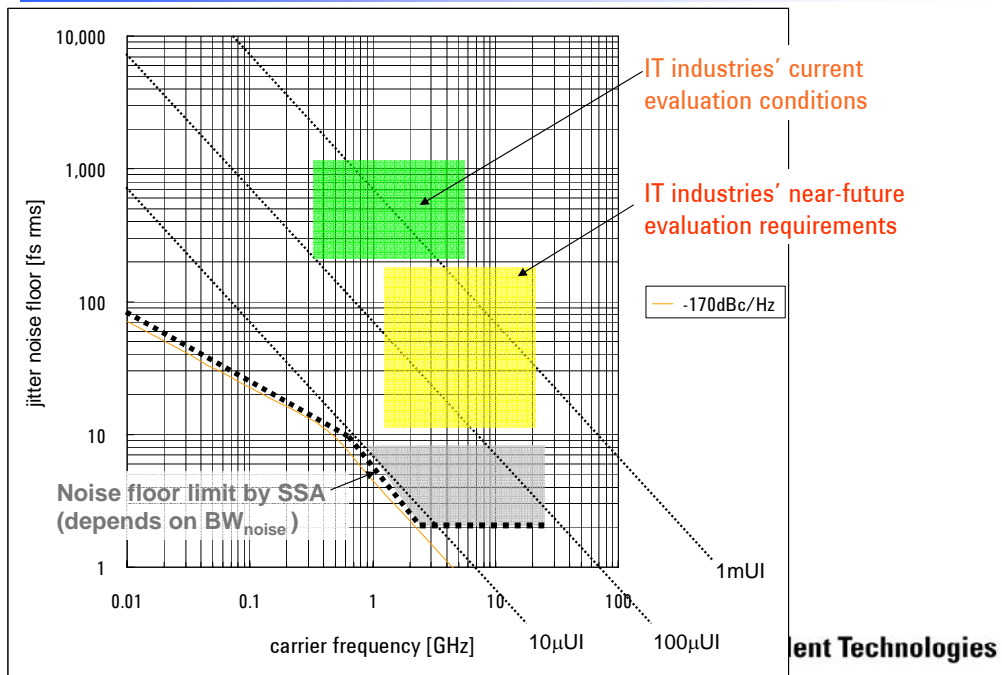


This graph shows the relationship between periodic jitter noise floor and carrier frequencies for several spur levels (-dBc).

Actual measurement conditions are mostly located between 1mUI and 10µUI (Unit Interval).

Assuming that your clock signal is 5GHz with -65dBc sideband spur, then you can see about 25 femto-seconds rms jitter (from the above graph) in periodic jitter measurement by the SSA.

Jitter noise floor vs. Carrier frequency (3)



This graph shows clock-jitter measurement requirements in IT industries for high-speed serial data communication applications. The area below 200 fs and above few GHz is becoming more important for quiet clock evaluation. Only a phase noise analysis method can achieve such ultra-low clock jitter measurement.

Conclusion

Clock jitter analysis

The goal is to

Determine the effect of Clock jitter on the BER of a system

- * Clock-jitter is amplified by the transmitter multiplier,
- * Has a frequency-dependent response to Clock Recovery.

If the jitter on the data is the same as the jitter of the sampling point → no errors!

Small bandwidth on Transmitter PLL limits data-jitter.

Large bandwidth on Receiver Clock Recovery limits affect of jitter on BER.

>> Appropriate response (transfer function) models are necessary for accurate evaluation of clock signal quality.



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 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.

Conclusion

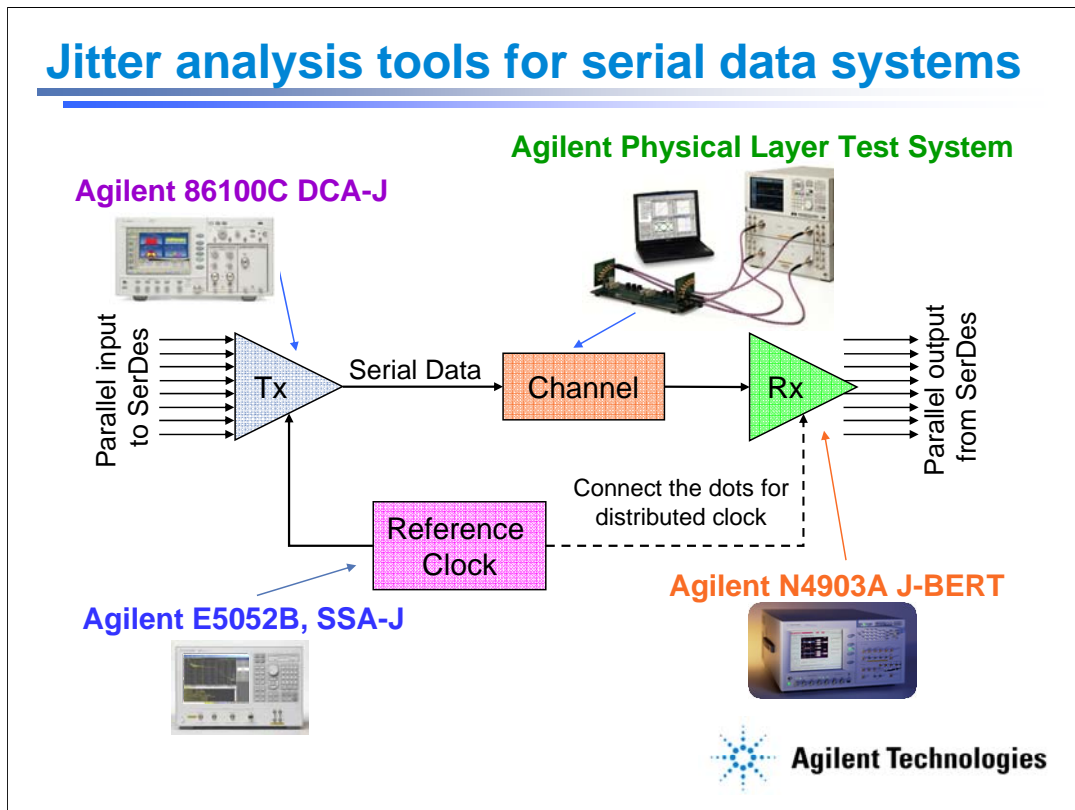
Clock jitter analysis tool

Sub-pico-second accuracy and **femto-second resolution** with **appropriate response (transfer function) models** in clock jitter measurement is desired for recent high-speed data communication systems.

Phase noise analysis is one of the best promising ways to take, for accurate clock jitter measurement.



Jitter analysis tools for serial data systems



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

Q & A



Q&A

- What clock frequency can the SSA measure?
10MHz to 7GHz (E5052B) or 26.5GHz (E5052B+E5053A)
- What is SSA's jitter frequency bandwidth?
100MHz
- How can I emulate jitter transfer functions with the SSA?
Special utility software is available. Contact Agilent Sales.
- Can the SSA measure DCD (Duty Cycle Distortion)?
No.
- Can the phase noise analysis measure cycle-to-cycle jitter?
No.
- How can I find out more about the SSA and SSA-J?
Refer to: www.agilent.com/find/ssa
www.agilent.com/find/ssaj

