APPLICATION NOTE

Finding Sources of Jitter with Real-Time Jitter Analysis

KEYSIGHT

As data rates increase effects of jitter becomes critical and jitter budgets get tighter. Instruments such as real-time oscilloscopes and Bit Error Ratio Testers (BERT) are optimized for determining the total amount of jitter and worst-case eye-opening in your high-speed digital system and can be used to test for compliance based on industry standards. In addition, some real-time instruments can separate random and deterministic jitter components to predict/extrapolate worst-case total jitter (TJ) and eye-opening based on a user-specified Bit Error Ratio (BER), typically 10-12. But when jitter measurements do not meet a particular minimum standard, or if jitter measurement results are “too close for comfort,” then measuring the amount of component or system jitter is just half of the jitter test equation. Determining the root-cause of jitter is the other half of the test equation. The focus of this paper will be to address some practical “tips & tricks” on using real-time oscilloscopes with jitter analysis and high-speed pulse/pattern generators to separate and time-correlate specific deterministic jitter components to help identify, measure, and view sources of systematic timing errors.

Understanding Jitter

Jitter is the deviation of a timing event of a signal from its ideal position. The traditional way to measure jitter is with an eye-diagram using repetitive acquisitions on an oscilloscope as shown in Figure 1. Looking at this composite view, you might assume that you have a band of worst-case jitter equal to the width of the rising and falling edges of the eye-diagram. You might also assume that all edges of your signal are jittering about to the same degree. Both of these assumptions would be incorrect.

Jitter is complex and is composed of both random and deterministic jitter components. Random Jitter (RJ) is theoretically unbounded and Gaussian in distribution. “Unbounded” simply means that if you wait long enough, the peak-to-peak jitter will increase indefinitely theoretically.

This means that an eye-diagram may never show the worst-case condition. If the jitter in your system consisted of just random jitter, then each edge of the data signal would have the same probability of timing error.

Deterministic Jitter (DJ) is bounded and doesn’t follow any predictable distribution. It consists of several different sub-components and is usually caused by a systematic problem in your high-speed digital design. For this reason, DJ is sometimes referred to as systematic jitter. If you were to view each individual edge of the data signal, you would probably see that particular data edges contribute different amounts of timing error. Depending on the data pattern, some edges in the serial pattern will always be shifted to the right (positive timing error), while other edges will always be shifted to the left (negative timing error) relative to their ideal timing locations. And then the random jitter component would cause these individual data bit edges to randomly bounce around the shifted/offset deterministic amplitude of jitter.

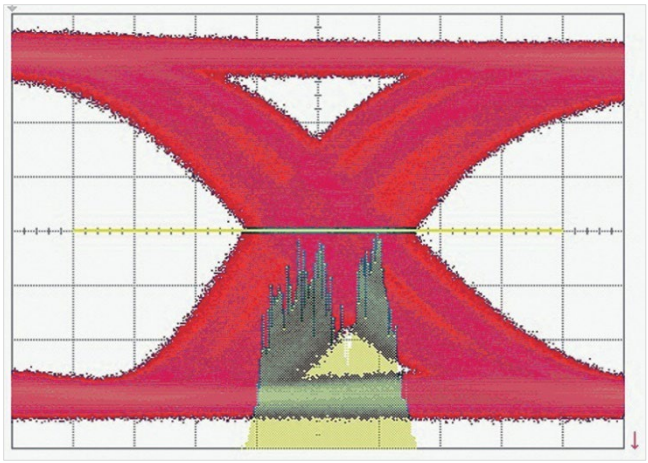

Figure 1. Viewing jitter in an eye-diagram

Using an eye-diagram, you can sometimes quickly determine if your system is overwrought by random or deterministic jitter or the possible combination of the two. Using the oscilloscope's variable intensity or color-graded display capability, you can visually look for the existence of "bright" trace paths within the infinite-persistence display. Referring to the traces in Figure-1, we can see several well-defined paths of brightness. This is a clear indication of deterministic jitter. Some of the data edges consistently occur in some locations while other edges occur in other locations of the eye-diagram. The power of a real-time scope is the ability to view these particular data edges individually as we will see later in this paper.

Another technique to determine the existence of random versus deterministic jitter is to use the scope's histogram feature. By taking a look at a slice of data across the center of the screen, we can see in Figure 1 that the distribution of edge placements is a combination of random and deterministic jitter. If the Probability Distribution Function (PDF) were Gaussian (the classic bell-shaped curve), then the system jitter would probably be overcome by random jitter. The fact that this particular PDF is approximately bi-modal in distribution is another indication of the existence of a significant amount of deterministic jitter.

Real-Time Jitter Analysis

Measuring the total jitter in your system with a Bit Error Ratio Tester (BERT) or real-time RJ/DJ separation techniques will tell you whether or not your system meets a jitter/timing budget specification. And viewing an eye-diagram along with a histogram can give you a good intuitive feel concerning the type of jitter in your system and a rough feel for the amount of total jitter. But neither of these two types of measurements can provide you with much insight into how to identify, view and then reduce specific components of jitter. This is where real-time jitter analysis steps in. The primary contribution of jitter analysis using a real-time oscilloscope is its inherent ability to capture data or clock pulses in a single acquisition with timing measurements on each and every pulse in a long stream of data. The real-time scope and jitter analysis can then be used to time-correlate specific jitter error measurements to specific data bits or other signals in your system that might be contributing to the total system jitter.

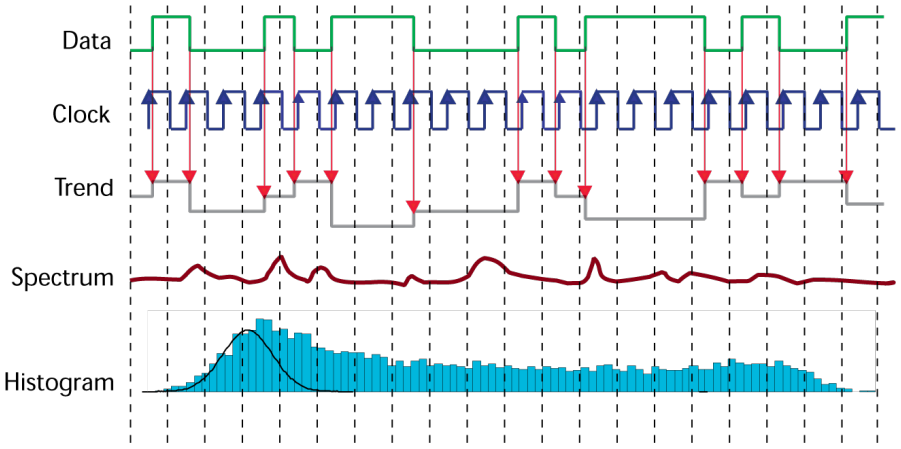

Figure 2. Real-time jitter analysis

Figure 2 illustrates the technique that real-time jitter analysis uses to measure jitter on an NRZ data signal. This type of real-time jitter measurement is typically referred to as a Time Interval Error (TIE) or phase jitter measurement. The real-time scope first captures a deep record of the NRZ data signal and then creates an ideal software-recovered clock based on the captured data.

Depending on the user's selection, this software-generated clock can be either of a fixed frequency or a PLL-type clock with a specified loop bandwidth. The jitter software then performs a best-fit algorithm to align the NRZ data edges with the "virtual" clock edges. Although Figure 2 shows a software-generated clock waveform (blue trace), this signal is typically NOT shown on the scope's display. It is merely used as a theoretical timing reference for making Time Interval Error (TIE) measurements on each data edge by the scope's computer.

Jitter analysis software in the real-time oscilloscope then performs a series of delta-time measurements between each edge transition in the data record at a specified threshold level relative to edges of the best-fit clock. There is no need to specify a measurement threshold level for the clock since it is just a theoretical time reference. After completing all of the timing measurements in the data record, the results can be viewed in three different formats.

With the jitter "trend" waveform, absolute timing errors for each data edge are plotted on the vertical axis with the horizontal axis based on the scope's timebase. In other words, time error versus time. The "trend" waveform provides a time-correlated view of each timing error relative to the data waveform. In the case of data-dependent jitter, the "trend" waveform can be a very powerful tool to time-correlate specific errors with the data bit that caused the error. In the case of Periodic Jitter (PJ) caused by signal coupling, the "trend" waveform can also be time-correlated to other signals in the system captured on other channels of the oscilloscope. Characteristics of periodic jitter will be discussed later in this paper.

Another way to view the jitter is in the frequency domain. With the jitter “spectrum” waveform, an FFT math function is performed on the TIE trend data to produce a view of the jitter based on repetitive frequency components within the series of delta time error measurements. In this case, data is plotted as amplitude timing error on the vertical axis versus frequency on the horizontal axis. This view can be especially beneficial when looking for uncorrelated Periodic Jitter (PJ) components that are not time-correlated to the data signal.

The “histogram” view will show the Probability Distribution Function (PDF) of the jitter (composite of all TIE measurements in the data record) and is plotted as timing error versus number of hits (N). The results of the real-time histogram should closely correlate with the results of a repetitive histogram produced from an eye-diagram measurement. However, with real-time sampling much more data is collected from a single acquisition of the signal. In other words, you will be able to view a distribution of jitter from a single acquisition. In addition, the real-time jitter histogram will build up with repetitive real-time acquisitions to generate a more accurate and complete PDF.

Typical Characteristics of Individual Jitter Components

In order to interpret measurement results and waveform views performed by real-time jitter analysis, you must first understand the characteristics and likely causes of individual jitter components. There is more to jitter interpretation than just knowing that Random Jitter (RJ) is Gaussian in distribution, and deterministic jitter is non-Gaussian.

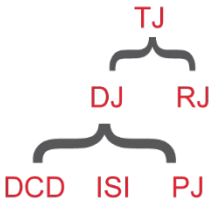

Figure 3. Jitter components

As mentioned earlier, Total Jitter (TJ) is composed of a Random Jitter (RJ) component and a Deterministic Jitter (DJ) component. Random jitter is unbounded, and for this reason (unlimited peak-to-peak) RJ is usually measured in terms of an RMS value. In addition, random jitter is very predictable in terms of distribution. The Probability Distribution Function (PDF) is always Gaussian in distribution. Unfortunately, predicting the cause of RJ is a more difficult task and is not within the scope of this paper. RJ is often caused by thermal effects of semiconductors and requires a deeper understand of physics. However, one piece of advice is to pay close attention to the amount of vertical noise in your system. Random vertical noise will directly translate into random timing jitter.

On the other hand, deterministic jitter is bounded and is always measured in terms of a peak-to-peak value. Although the distribution of deterministic jitter can be very unpredictable, the likely causes and characteristics of the individual sub-components of measured deterministic jitter are very predictable. The sub-components of Deterministic Jitter (DJ) consist of Duty Cycle Distortion (DCD), Inter-Symbol Interference (ISI), and Periodic Jitter (DJ) as shown in Figure 3. Let's now take a closer look at possible causes and characteristics of each of the sub-components of deterministic jitter.

There are two primary causes of Duty Cycle Distortion (DCD) jitter. If the data input to a transmitter is theoretically perfect, but if the transmitter threshold is offset from its ideal level, then the output of the transmitter will have duty cycle distortion as a function of the slew rate of the data signal's edge transitions. Referring to Figure-4, the waveform represented by the dotted line shows the ideal output of a transmitter with an accurate threshold level set at 50% and with a duty cycle of 50%. The solid line waveform represents a distorted output of a transmitter due to a positive shift in the threshold level. With a positive shift in threshold level, the resultant output signal of the transmitter will have less than 50% duty cycle. If the threshold level is shifted negatively, then the output of the transmitter will have greater than 50% duty cycle.

Measuring TIE relative to the software-generated best-fit clock results in a positive timing error on the rising edge of each data bit and a negative timing error on the falling edge of each data bit. The resultant TIE trend waveform will possess a fundamental frequency equal to 1/2 the data rate. The phase of the TIE trend waveform relative to the data signal will depend on whether the threshold shift is positive or negative. With no other sources of jitter in the system, the peak-to-peak amplitude of DCD jitter will be theoretically constant across the entire data signal. Unfortunately, other sources of jitter, such as ISI, almost always exist making it sometimes difficult to isolate the DCD component. But one technique to test for DCD is to stimulate your system/components with a repeating 1-0-1-0… data pattern. This technique will eliminate Inter-Symbol Interference (ISI) jitter and make viewing the DCD within both the trend and spectrum waveform displays much easier. Using the jitter spectrum display, the DCD component of jitter will show up as a frequency "spur" equal to 1/2 the data rate.

Another cause of DCD is asymmetry in rising and falling edge speeds. A slower falling edge speed relative to the rising edge will result in greater than 50% duty cycle for a repeating 1-0-1-0… pattern, and slower rising edge speeds relative to the falling edge will result in less than 50% duty cycle. Although not graphically shown in this paper, the results of jitter analysis and the TIE trend waveform will look similar to the results of the example illustrated in Figure-4.

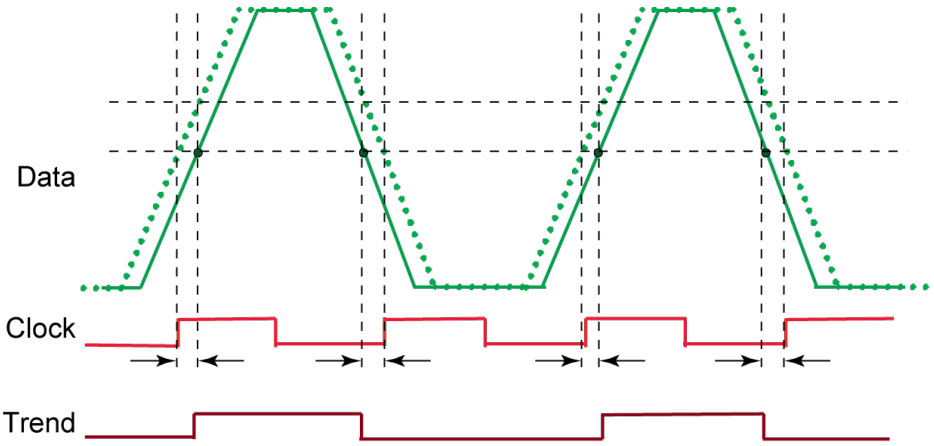

Figure 4. Duty Cycle Distortion (DCD)

Inter-Symbol Interference (ISI), sometimes called data dependent jitter, is usually the result of a bandwidth limitation problem in either the transmitter or physical media. With a reduction in transmitter or media bandwidth, limited rise and fall times of the signal will result in varying amplitudes of data bits dependent on not only repeating-bit lengths, but also dependent on preceding bit states. In addition, improper impedance termination and physical media discontinuities will also result in ISI due to reflections that cause signal distortions. Although we will address these two phenomena (BW limitations and reflections) separately in this paper as contributors to ISI jitter, in reality, waveform distortions due to reflections are also a bandwidth limitation problem.

Figure 5 shows an example of ISI due to bandwidth limitation problems. Limited bandwidth produces limited edge speeds, and limited edge speeds will result in varying pulse amplitudes at high-speed data rates. Varying pulse amplitudes will then result in transition timing errors. Let's now examine this more closely.

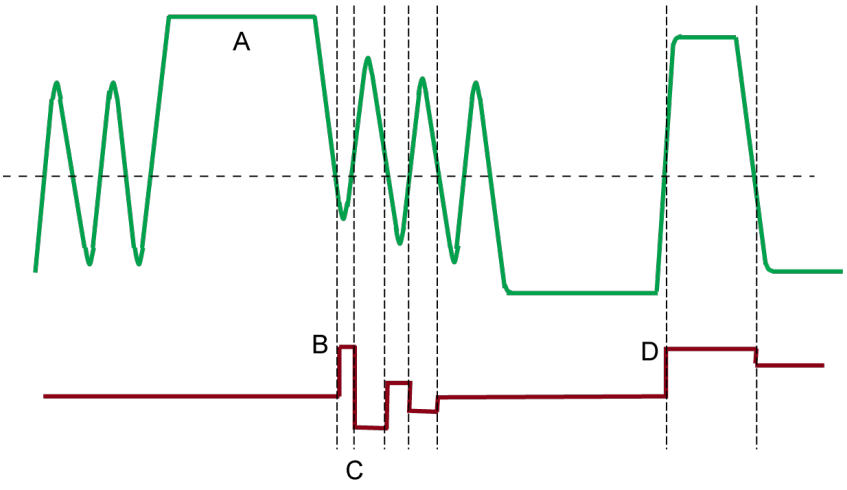

Figure 5. Inter-Symbol Interference (ISI) due to BW problem

With a long series of repeating “1’s,” the amplitude of the data signal will eventually rise to a full steady-state high level as illustrated by the long-high pulse at point A in Figure-5. When the state of the data changes to a “0,” the signal will have a relatively long transition time to reach the threshold level, resulting in a positive timing error. This will be manifested as a positive peak of timing error in the jitter trend waveform at point B. Note that this point on the jitter trend waveform is time-aligned with the negative data crossing point on the data signal.

The negative peak amplitude of the next “0” bit preceded by a long string of “1’s” will be attenuated for two reasons. First of all, the preceding long string of “1’s” means that the signal will take longer to transition to a true low level since the data signal starts from a higher initial level. Secondly, the following “1” bit causes the signal to reverse direction before it even reaches a solid low level. This reduction in signal amplitude will produce a negative timing error on the next transition to a “1” since the signal has a very short distant to travel to reach the threshold level. This is illustrated at point “C” on the jitter trend waveform.

The positive timing error illustrated at point “D” on the jitter trend waveform follows the same logic as the positive timing error at point “B” previously discussed. With a long string of “0’s” the data signal has sufficient time to settle to a full steady-state low level. When this signal then transitions back to a high level, it again has a longer transition time to reach the threshold level, and hence produces a positive timing error. Once you understand how bandwidth limitations produce Inter-Symbol Interference (ISI) timing errors, it becomes more intuitive to understand the unique “signature” of the jitter trend waveform due to ISI and how it relates to the time-correlated serial data signal under measurement.

Another common cause of inter-symbol interference besides bandwidth limitations is signal reflections due to improper terminations or impedance anomalies within the physical media. Signal reflections will produce distortions in the amplitude of the data signal as shown in Figure-6. Depending on physical distances between impedance anomalies, reflections produced by one pulse may not show up on a high-speed data signal until several bits later in the serial pattern. Notice which pulse the arrows begin on and the where pulse distortion (reflection) occurs as illustrated by the end of each of the arrows in Figure-6.

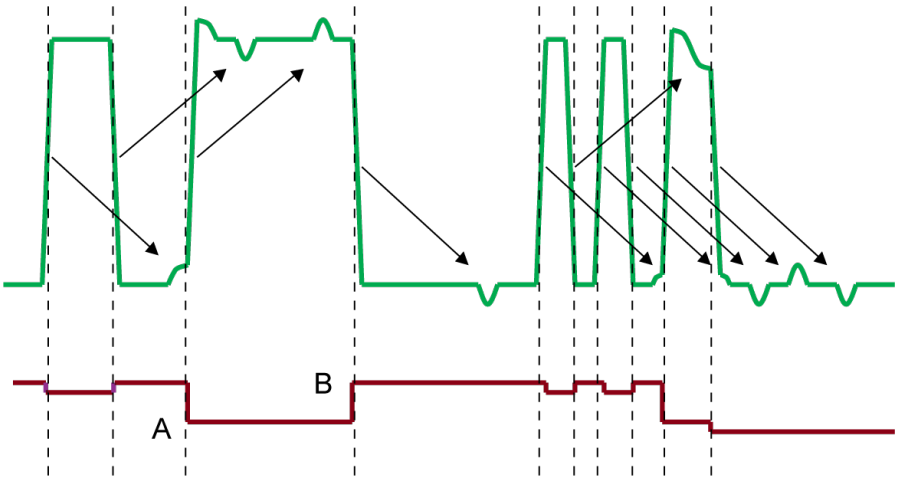

Figure 6. Inter-Symbol Interference (ISI) due to BW problem

If the amplitude of the signal becomes distorted on or near a data transition edge due to reflections, then a timing error may occur. If a signal reflection causes signal attenuation near the data edge, then a negative timing error will be detected since the signal will have less distance to travel when transitioning to the threshold level. This is illustrated at point “A” on the jitter trend waveform in Figure-6. If a signal reflection causes a boost in signal amplitude, then the result will be a positive timing error since the signal will have farther to transition to reach the threshold level. This is illustrated at point “B” on the jitter trend waveform. Inter-symbol interference due to signal reflections can be very difficult to isolate and interpret. But if you have signal reflection problems in your system, you probably also have a bandwidth limitation problem.

Periodic Jitter (PJ) is usually the result of a cross-coupling or EMI problem in your system and can be either correlated or uncorrelated to the data signal. An example of uncorrelated PJ would be signals from a switching power supply coupling into the data or system clock signals. It is considered to be uncorrelated because it is not time-correlated with either the clock or data signal since it would be based on a different clock source. An example of correlated PJ would be coupling from an adjacent data signal based on the same clock or a clock of the same frequency.

Figure-7 shows an example of a “corrupter” signal (upper trace) capacitively coupling into our serial data signal (middle trace). This coupling will result in amplitude distortions on the data signal. Just like inter-symbol interference due to reflections, if these amplitude distortions occur at or near a data signal transition, a timing error may occur.

Since most Periodic Jitter (PJ) will be uncorrelated to the data signal, any attempts to time-correlate the jitter trend waveform to the data waveform may turn into a futile attempt. As we will show later in this paper, uncorrelated periodic jitter can often be detected using the jitter spectrum view.

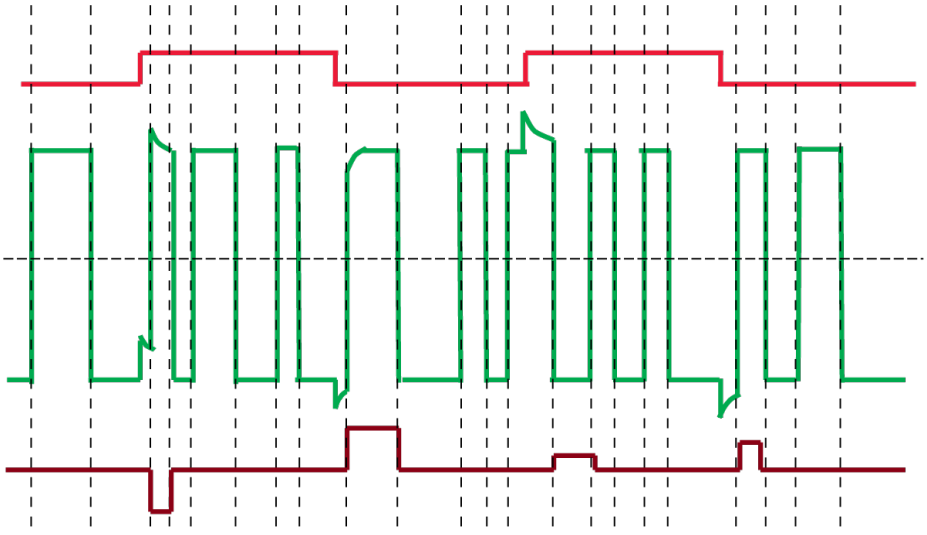

Figure 7. Periodic Jitter (PJ) caused by capacitive coupling

Isolating Jitter Components in the Real World

In the examples that we have just illustrated, we have shown how individual and isolated jitter components are theoretically manifested in the jitter trend and spectrum views. But in the real world, jitter components are rarely isolated. If you have multiple jitter components in your system contributing to the total system jitter, these various jitter components become convoluted and you end up viewing composite results, which can be difficult to interpret. If you are familiar with the arcade game called “whack-a-mole,” perhaps you can relate to this analogy. In this arcade game the “moles” pop their heads up one at a time, and it’s your job to “whack” them on their heads with a mallet to push them back down. This is exactly what we would like to do with the various jitter components. It would be nice if the jitter “moles” (jitter components) would appear individually so that we could identify them and then “whack” them down. Unfortunately, in a live system all of the jitter moles may have their heads popped up simultaneously. This makes it difficult to identify their source and to decide which “mole” to whack first.

But there are some novel stimulus-response techniques you can employ in order to isolate, measure, and then view individual jitter components. Once you successfully isolate individual components, you can then often time-correlate worst-case peaks of jitter to specific data bit transitions and then use common-sense debug techniques to solve your jitter problems… whacking them down one component/mole at a time.

The primary tool to isolate jitter components is a real-time oscilloscope with responsive and interactive jitter analysis such as Keysight Technologies, Inc. 6-GHz 54855A, shown in Figure 8. In addition, a high-speed pulse/pattern generator such as Keysight’s 3.35-Gb/s 81134A, shown in Figure-9 can be very useful for generating known serial patterns of high-speed differential stimulus. Let’s begin with some real jitter measurement examples using these two measurement tools.

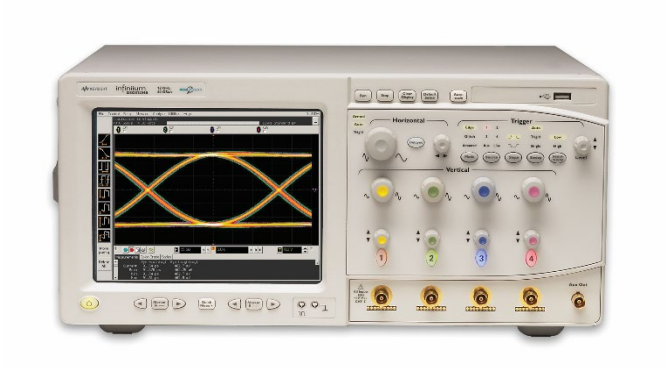

Figure 8. Keysight’s 80000 12 GHz real-time oscilloscope

Figure 9. Keysight’s 81134A 3.35-Gb/s pulse/pattern generator

Isolating and Measuring Duty Cycle Distortion (DCD)

One technique mentioned earlier to isolate and measure Duty Cycle Distortion (DCD) is to stimulate your system/component with a repeating 1-0-1-0… serial pattern. This stimulus pattern will eliminate most of the Inter-Symbol Interference (ISI). Although ISI is eliminated with this repeating pattern, Random Jitter (RJ) and any Periodic Jitter (PJ) will still be present in the signal, which will contribute to convoluted measurement results. But there is a measurement technique to also eliminate both random jitter and uncorrelated periodic jitter in the jitter measurement results. Figure-10 shows the capture of a repeating 1-0-1-0… serial pattern (green trace (top)) with jitter analysis results showing the TIE trend waveform (orange trace (bottom)). To eliminate the random components (RJ and PJ), we have used waveform math to average the jitter trend waveform. Before averaging, the TIE trend waveform would be “bouncing” vertically due to the random components with repetitive acquisitions. But averaging has eliminated the random jitter components to show a very stable trend waveform with an approximate constant level of peak-to-peak amplitude of jitter from cycle-to-cycle. We can then use the scope’s manual markers or the scope’s automatic parametric measurement capability to measure the peak-to-peak amplitude of duty cycle distortion. In this case, we measured approximately 10-ps of DCD.

In addition to determining the level of DCD, we can also glean additional information about our measurement results. With the time-correlated display of the jitter trend waveform and data signal, we can see that the trend waveform is in-phase with the data signal. This is an indication that the duty cycle of our pulse is less than 50%. Rising edges always occur late (+error), and falling edges always occur early (-error). Perhaps our transmitter threshold level is too high, or perhaps the output of the our transmitter generates slower rising edge speeds as compared to faster falling edge speeds. At this point if we believe the peak-to-peak level of DCD is excessive, we can setup additional characterization tests to measure the duty cycle of each pulse in the data stream using jitter analysis. In addition, if we suspect that the duty cycle distortion is caused by asymmetry in the rising and falling edge speeds, we can also setup the instrument and jitter analysis to characterize each rising and falling edge in the data stream.

One critical element to perform this particular measurement using waveform averaging is that the oscilloscope’s waveform-capture and jitter analysis capability must be very responsive with a fast display and measurement update rate. Some jitter analysis packages are geared more for single-shot/static measurements. Averaging requires multiple acquisitions at a fast rate. Keysight’s MegaZoom technology provides the fastest display update rates and measurement processing time on deep memory records in the oscilloscope industry.

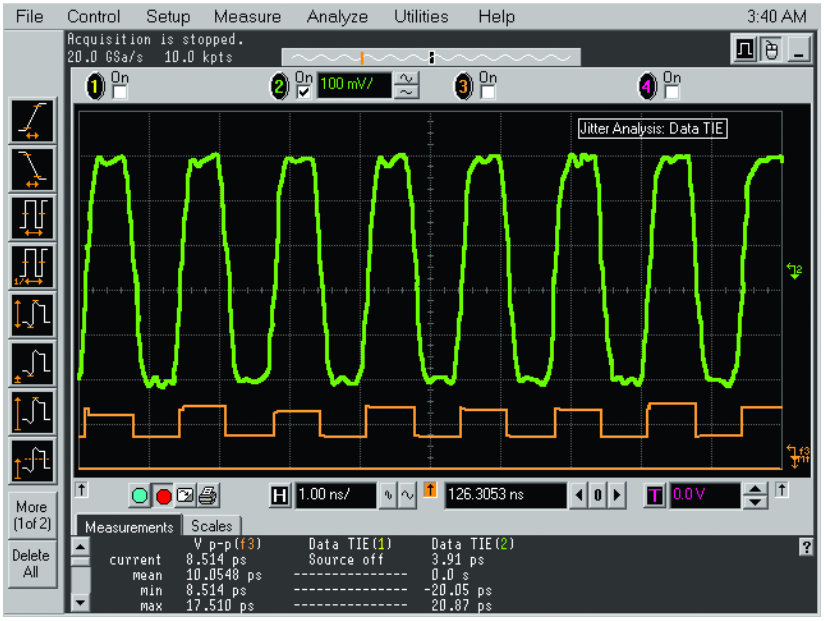

Figure 10. Isolating DCD

Isolating and Measuring ISI

Figure 11 shows a measurement example of isolating inter-symbol interference due to a system bandwidth limitation problem. Using a high-speed pulse/pattern generator, stimulate your selected devices under test with a repeating PRBS signal at the full-specified clock rate. You may apply the stimulus at either the input of a transmitter circuit, the input of a receiver, or possibly at the input of a transmission line in front of a receiver. Using one of the scope's channels, capture the data signal at the desired point and setup jitter analysis to perform a TIE measurement with the jitter trend waveform view. Running real-time acquisitions repetitively with a synchronous trigger, you will observe the TIE trend waveform (orange trace (bottom)) randomly bouncing up and down in amplitude. This "bouncing" is primarily due to Random Jitter (RJ). But if the scope's update rate is sufficiently fast, you can usually see a repetitive pattern of the deterministic components. To eliminate the random components, simply use waveform math to average the TIE trend waveform. With sufficient cycles of acquisition and averaging, the resultant TIE trend waveform will become very stable and consist of just the Deterministic Jitter (DJ) components.

Once you have a stable jitter trend waveform, you can then time-correlate specific peaks of jitter back to the serial data signal. The TIE trend waveform will have a unique "signature" of DJ dependent on the serial data pattern. If your system/components exhibit bandwidth limitation problems, you can observe and time-correlate inter-symbol interference jitter as evidenced by positive peaks of jitter coincident with the end of long strings of "1's" or "0's." And you will probably observe negative peaks of jitter coincident with the end of short "1's" or "0's."

To measure the amount of deterministic jitter, simply select to perform an automatic peak-to-peak measurement on the averaged TIE trend waveform. With minimal correlated Periodic Jitter (PJ) in the system, this measurement gives the total peak-to-peak deterministic jitter consisting of just Duty Cycle Distortion (DCD) and Inter-Symbol Interference (ISI) jitter components. In this example, we measured approximately 44-ps of deterministic jitter. From the previous measurement where we determined DCD using a repeating 1-0-1-0… pattern, we can compute the approximate ISI component as 44-ps – 10-ps = 34-ps.

In addition to measurement responsiveness, another critical element of being able to perform jitter measurements such as these on high-speed serial data signals is having an oscilloscope and differential active probes with sufficient bandwidth. Poorly designed probes or long cables can contribute a significant amount of inter-symbol interference jitter to your measurements. You definitely don't want to wind up measuring jitter components caused by waveform distortions contributed by your scope, probes, and cabling.

Figure 11. Isolating Inter-symbol Interference (ISI)

Figure 12 shows an example of viewing inter-symbol interference generated by both bandwidth limitation problems and reflections. It is almost impossible to separate termination problems from bandwidth limitation problems when making jitter measurements. Both of these contributors will result in ISI jitter for the same reason, which is modulation of peak amplitudes of pulses at the receiver. These modulations will ultimately generate timing errors because of variations in transition times to data crossing threshold levels. But there are some clues that you can look for to identify impedance termination problems that result in signal reflections.

If termination problems do not exist, then leading-edge pulse shapes of similar width pulses should look similar since each pulse is generated by the same transmitter and travels down the same physical media. But if reflections do occur, you will probably be able to see dissimilar pulse shapes. In Figure-12 we show an example of two high-level pulses (A & B) consisting of a series of “1’s.” But notice the differences in their pulse shapes near the leading edges. The first long series of “1’s” (A) is not immediately preceded by other pulses. But we can see a reflection (an “up” bump) approximately 1-ns after the leading edge of this pulse. The second long series of “1’s” (B) is immediately preceded by two short “1’s.” We can see distortion in this pulse (another “up” bump) approximately 1 ns after the preceding short “1” pulse. Due to termination problems, this preceding pulse is causing a delayed reflection onto this later pulse very close to its leading edge. Depending on the data pattern, reflections will also be present on some of the shorter pulses, but these are much more difficult to identify because reflections on narrow pulses will be “masked” due to bandwidth limitations on short “1” and “0” pulses.

Measuring the total effect of ISI jitter including termination problems is exactly the same as just previously discussed. Simply establish a synchronous trigger and then average out the random jitter components from the TIE trend waveform. The net result is the total Deterministic Jitter (DJ), which consists of a combination of ISI and DCD. For this example we measured approximately 47 ps of peak-to-peak deterministic jitter. Then simply subtract out the DCD component based the DCD measurement using the repeating 1-0-1-0… serial pattern. If your ISI measurement results appear to be excessive, you can then easily time-correlate worst-case jitter peaks with specific data pulses.

Figure 12. ISI due to reflections

Figure 13. Testing for reflections at the transmitter side

To verify that the waveform distortions you are viewing on the data signal are really caused by reflections and not by some other effect, such as probing resonance, probe your signal at the transmitter side of the transmission line and reduce the data rate of your signal. As you can see in Figure 13, the reflections will become more predominant at the transmitter side. And with a slower data rate, it will be easier to determine which pulse edge causes each waveform distortion and then you can easily measure the time delay between the cause and effect to help identify the location of your impedance discontinuities. In this particular example, we have identified that our transmission line has both a capacitive and inductive discontinuity as evidenced by the positive peak reflections (low impedance) and the negative peak reflections (high impedance) respectively.

Measuring reflections at the transmitter side of the transmission line with a reduced data rate signal is very similar to a Time Domain Reflectometry (TDR) measurement. In fact, if your reflection problems are serious enough, it might be advisable to use a real TDR instrument such as Keysight's 86100A, or a Vector Network Analyzer (VNA) to more accurately characterize your physical media for impedances and distances between physical anomalies.

Isolating Uncorrelated Periodic Jitter (PJ)

Periodic jitter (PJ) is typically caused by cross-talk, signal coupling, or EMI. In most cases, PJ will be uncorrelated to the data signal, meaning the “corrupter” signal has no time relationship to the serial data signal. The best tool to use to identify the presence of uncorrelated PJ is usually the jitter spectrum display. Uncorrelated PJ will exhibit unique frequency “spurs” in the spectrum display. Within the spectrum display you will probably see lots of spurs. Most of these spurs will be harmonics and sub-harmonics of the data signal, which indicates the presence of pattern dependent jitter such as DCD and ISI. When testing for uncorrelated periodic jitter, it will be your job to look for “odd ball” frequency spurs unrelated to the data signal frequency.

Referring to Figure-14, we have captured a long string of serial data (yellow trace (top)) using deep memory and then performed jitter analysis on each data edge. The purple trace (middle waveform) shows the jitter trend waveform while the orange trace (bottom waveform) shows the frequency spectrum of jitter. Looking at the TIE jitter trend waveform, it is impossible is visually detect any low frequency modulation in this measurement. However, the jitter spectrum view (orange trace (bottom)) clearly shows a low frequency component of modulation. In this example, we have measured a “spur” on the spectrum display at exactly 400-kHz. This frequency is unrelated to the data signal’s 2.5-Gb/s data rate and is an indication that we have just discovered an uncorrelated periodic jitter component.

Note that if the periodic jitter is sinusoidal in nature, then the spectrum display will generate a single frequency spur associated with the fundamental frequency of this coupling component. However, if the periodic jitter is non-sinusoidal, then there will probably be several frequency spurs associated with this component of coupling. But all of these particular spurs on the spectrum display will be harmonics of one another. To help identify the source of digital periodic jitter, look for the fundamental frequency component amongst the cluster of spurs.

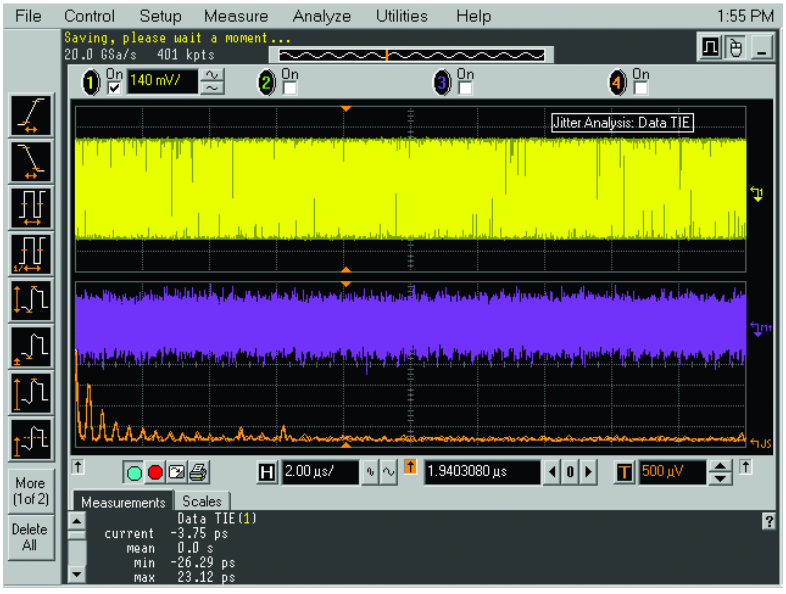

Figure 14. Isolating uncorrelated PJ with a jitter spectrum measurement

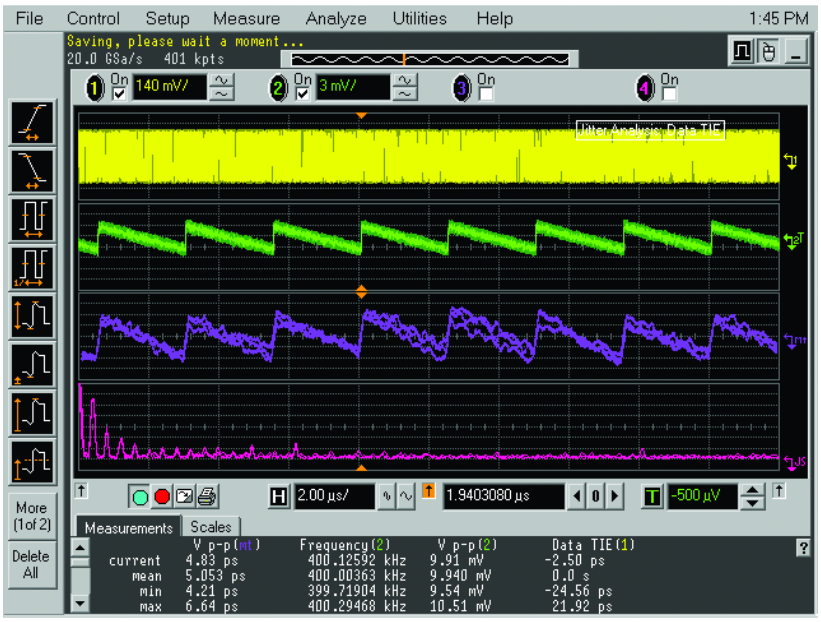

Figure 15. Verifying that switching power supply coupling causes periodic jitter

Using either the scope's automatic measurements or markers, we can now easily measure the peak-to peak amplitude of this jitter component. In this example, we measure approximately 5-ps peak-to-peak of uncorrelated PJ caused by coupling from the witching power supply, which is an insignificant amount of jitter. But if the amplitude of this jitter component is excessive, then perhaps better shielding, filtering, or improved board trace layout techniques are needed to reduce the amount of this component of periodic jitter in our system.

Having sufficient acquisition memory depth in your oscilloscope is a critical measurement element to detect relatively low frequency periodic jitter. In this example, we have captured eight periods of low frequency modulation using 400-k points of acquisition memory with the A-to-D converter running at a sample rate of 20 GSa/s. A scope with less memory could have made this measurement by capturing a minimum of one period of uncorrelated PJ. However, with deeper memory acquisition you can obtain higher resolution measurements in the spectrum display. In addition, deeper memory will enable you to measure lower frequency components of jitter. With the 90000 Series scopes 1 Gpts of acquisition memory, you can detect jitter components as low as 20 Hz.

Isolating Correlated Periodic Jitter (PJ)

Detecting the presence of correlated Periodic Jitter (PJ) is usually the most difficult type of jitter to uncover. Fortunately, this type of jitter is usually the least likely source of jitter contributing to a system's total jitter. The fact that it is correlated means that this jitter component is synchronous with the serial data stream that it is corrupting. A possible source of correlated PJ is coupling from an adjacent lane of serial data, such as multi-laned InfiniBand data traffic. In this case, adjacent lanes of data would be based on a common clock.

Since correlated periodic jitter is time-correlated to the serial data signal under test, it may be difficult to detect this component of jitter using the jitter spectrum display. Correlated PJ will definitely contribute to specific frequency spurs in the spectrum display, but these spurs may land on top of or between other data dependent spurs making it hard to discriminate between ISI and correlated PJ frequency components in the jitter spectrum display.

One technique that may prove successful for you is to set up your system to generate serial data traffic on just one data lane as shown in Figure-16. Using the scope's deep memory, engage TIE jitter analysis with the spectrum display turned on and average the results using waveform math and repetitive acquisitions (bottom trace). Averaging the spectrum display will eliminate the Random Jitter (RJ) component. And since only one lane of data is being generated with all other lanes "quiet," the averaged spectrum display will not include any correlated periodic jitter due to coupling from adjacent lanes of traffic. You should see several frequency spurs indicating the presence of ISI, DCD, and possibly uncorrelated PJ. Store this averaged spectrum display for later reference.

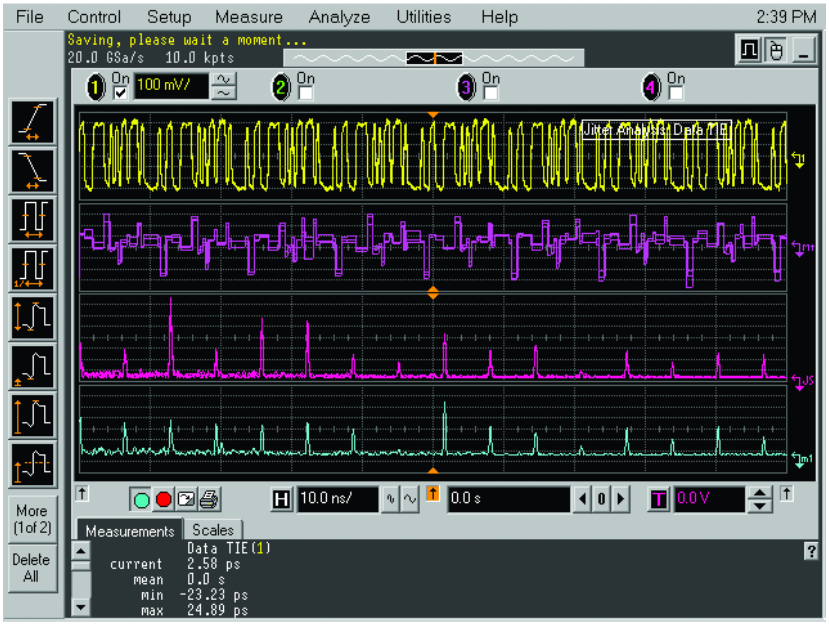

Figure 16. Detecting correlated Periodic Jitter (PJ)

Now turn on all lanes of traffic in your system using the same pattern length for each lane of traffic and perform the same averaged jitter spectrum measurement as just described. Jitter contribution from correlated periodic jitter will add to the height of the ISI & DCD spurs relative to the previous measurement that did not include correlated periodic jitter (second trace up from bottom). If there is a significant difference in spur heights from these two measurements, then this is an indication of the presence of correlated periodic jitter.

To measure the peak-to-peak amplitude of correlated PJ, use the techniques previously discussed to measure the amount of DJ with the system generating just a single lane of data traffic, and then perform the same DJ measurement with the system generating multi-lanes of data traffic. The peak-to-peak difference between these two measurements is the amount of peak-to-peak correlated PJ in your system. To compute the total amount of PJ (correlated and uncorrelated), simply add both PJ components.

Isolating Random Jitter (RJ)

In several of the previous examples of testing for inter-symbol interference, we have shown how you can eliminate random components of jitter (RJ + uncorrelated PJ) from the jitter trend waveform by establishing a synchronous trigger on the data signal and then averaging the jitter trend waveform using waveform math and repetitive acquisitions. To isolate the random components, simply use a chained waveform math function as shown in Figure-17 to subtract the averaged trend waveform (green trace (second from bottom)), which represents the DJ component, from the real-time trend waveform (purple trace (third from bottom)), which represents TJ. The results will be a trend waveform (pink trace (bottom)) consisting of both Random Jitter (RJ) and uncorrelated periodic jitter. Since these components are random relative to the data signal, it will be impossible to time correlate these results to the data waveform.

Figure 17. Measuring system Random Jitter (RJ)

At this point you can measure the contribution of these jitter components by performing an ac RMS measurement on this waveform. Note: Random Jitter (RJ) is always measured in terms of an RMS value since it is theoretically unbounded. In this example we have measured approximately 5.7 ps rms of random jitter. This amount of jitter may not sound like much, but when converted into a peak-to-peak value based on a BER of 10-12, 5.7 ps rms RJ translates into approximately 80-ps of peak-to-peak jitter.

If you believe that the uncorrelated periodic jitter component is significant, you can use techniques previously discussed to detect and measure the uncorrelated periodic jitter components. But rather than measuring the peak-to-peak amplitude of uncorrelated PJ component as previously described, measure the ac RMS value over a single period. You can then use the square root of the sum of the squares formula to separate out the uncorrelated periodic jitter component from the RMS measurement that includes both random components (RJ + uncorrelated PJ). As a first-order approximation, the results will be the RMS value of just the random jitter (RJ) component.

Analyzing a Spread Spectrum Clock

Jitter on a data signal is basically unwanted phase modulation. However, some serial data standards actually specify intentional modulation of the serial data with Spread Spectrum Clocking (SSC). You can use jitter analysis to measure the shape and frequency of this designed-in modulation.

Figure-18 shows an example of measuring the spread spectrum clock when performing jitter analysis on a 2.5-Gb/s data stream. By setting up the jitter analysis to display the jitter trend waveform based on a fixed-frequency clock, the results of the jitter measurement will be largely driven by the spread spectrum clock as shown by the purple trace (bottom) in this screen-shot. We can easily measure the frequency of this clock using either the scope's markers or automatic parametric measurements. In this case, we measured 32.9-kHz.

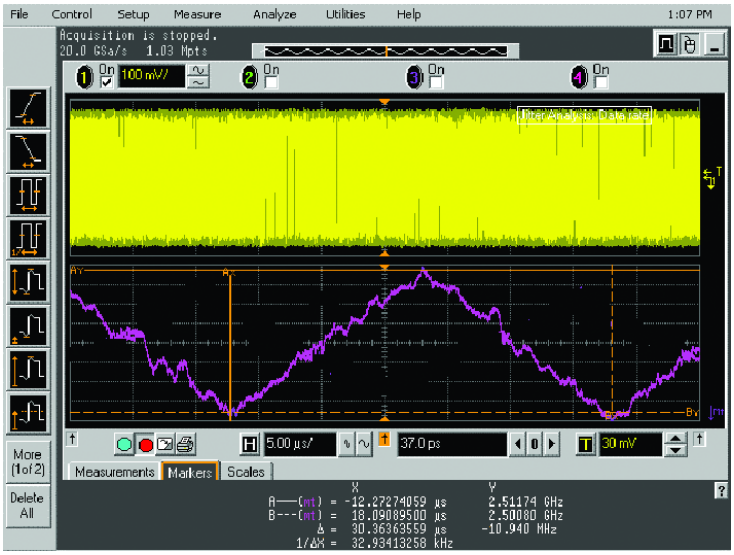

Figure 18. Measuring the Spread Spectrum Clock

Just as memory is important for detecting low-frequency Periodic Jitter (PJ), having sufficient memory is also important for performing spread spectrum clock measurements. Capturing a minimum of one period of a 30 kHz spread spectrum clock at a sample rate of 20 GSa/s requires a minimum of 667-K of acquisition memory.

You should be cautioned that not all real-time jitter analysis is capable of measuring the spread spectrum clock. When timing errors exceed one Unit Interval (UI), which is equivalent to one period of the software-generated reference clock, some jitter analysis will “snap-back” and measure the timing error of the data edge relative to the nearest clock edge. Measuring the SSC requires that time error measurements be able to exceed one UI. The jitter analysis software must be able to track UIs in the serial data pattern and then measure the timing error relative to the appropriate clock edge... not the nearest clock edge.

Although we used a fixed-frequency software generated clock as a reference for this particular SSC measurement, if you wanted to analyze the unwanted modulation (jitter) in this 2.5-Gb/s data stream, you would use a “golden” PLL-type clock with a specific loop frequency in order to filter-out the intended low frequency modulation of the spread spectrum clock.

If your system is based on an embedded PLL clock, then to measure system jitter (not SSC) you should always use a PLL-type clock as your reference, regardless of whether your system uses a spread spectrum clocking scheme or not. This means that low-frequency periodic jitter (PJ) will also be filtered out of the jitter measurements. But this is okay. The hardware PLL clock in your system will also filter out any low frequency period jitter. When making jitter measurements with a software-generated clock, we want our software-generated reference clock to emulate the system’s embedded hardware clock.

Conclusions

Some real-time jitter analysis packages give answers in terms of the amount of total jitter that may be present in your system. This can be important for determining if your high-speed digital system meets a particular worst-case jitter and eye-opening specification. But knowing how much random jitter and deterministic jitter is in your system usually doesn’t give you a clue as to where it is coming from. The key to finding sources of jitter lies in the ability to time-correlate jitter measurement results with high-speed serial data signals, as well as other possible sources of uncorrelated periodic jitter. A real-time oscilloscope with jitter analysis along with the stimulus-response techniques described in this paper meet that critical time-correlation requirement to relate jitter trend measurement results to measured signals. Once you are able to time-correlate particular real-time timing error measurements to particular bits within a serial data pattern, it usually becomes a routine troubleshooting task to solve your deterministic jitter problems.

Features

- 13 GHz bandwidth real-time oscilloscopes with 40 GSa/s sample rate on all four channels simultaneously
- Up to 1 Gpts MegaZoom deep memory at all sample rates
- Electronic attenuators eliminate the reliability and repeatability concerns associated with mechanical attenuator relays
- Trigger jitter as low as 500 fs rms
- Easy-to-use, easy-to-understand jitter analysis option
- E2688A Serial Data Analysis/Mask Testing with Clock Recovery
- Serial ATA Signal Quality Compliance Test
- Each InfiniiMax probe amplifier supports both differential and single-ended measurements for a more cost-effective solution
- Unrivaled InfiniiMax probing accessories support browsing, solder-in, and socket use models at the maximum performance available
- The 13 -GHz InfiniiMax 1169A probe and E2668A/E2669A connectivity kits (all sold separately) are recommended for the Infiniium DSO91304A oscilloscope
- USB, SATA, PCIE, DDR compliance test options
- Communications Mask Test Kit
- VoiceControl Option

54850 Series Scopes and 1130 Series Probes

The highest-performance end-to-end measurement system available

Experienced scope users know that their measurements are only as good as their probing system. And as bandwidth increases, it is increasingly important to make sure you are measuring your circuit, not your scope probe. Nothing is more frustrating than chasing down an apparent design problem, only to find that it was caused by an inferior scope probe. Together, the newest Infiniium scopes and the breakthrough InfiniiMax high performance probing systems offer an end-to-end measurement solution with unmatched performance, accuracy and connectivity. The results are measurements you can trust and better insight into circuit behavior.

InfiniiMax high-performance active probe system

The innovative InfiniiMax probing system provides either differential or single-ended probing solutions for the most demanding mechanical requirements, without sacrificing performance. A flat frequency response over the entire probe bandwidth eliminates the distortion and frequency-dependent loading effects that are present in probes that have an in-band resonance.

Performance-enabling technology

When you need to make multi-channel measurements on projects that use subnanosecond logic, you need powerful instruments. These new Infiniium oscilloscopes maintain their full sampling performance of 40 GSa/s on all channels, so you can make critical timing measurements at the full performance of the oscilloscope.

Figure 19. Keysight's 90000 12 GHz real-time oscilloscope

81133A and 81134A 3.35 GHz Pulse/Pattern Generators

The need for pulse and pattern generation is fundamental to digital device characterization tasks. The ability to emulate the pulse and pattern conditions to which the device will be subjected is essential. This emulation should include both typical and worst case conditions. Accurate emulation requires superlative signal integrity and timing performance along with full control over parameters that allow specific worst case testing.

Setting standards

The Keysight 81133A and 81134A 3.35 GHz Pulse Generators provide programmable pulse periods from 15-MHz (66.6-ns) to 3.35-GHz (298.5-ps) pulse capability on all channels. At these frequency ranges the transition time performance becomes critical; less than 60-ps is specified and with a RMS jitter of 1.5-ps typical signal quality is assured. The Delay Control Input and the Variable Crossover Point functionalities allow emulation of real world signals by adding jitter to clock or data signals or by distorting the ‘eye’ for eye diagram measurements.

Features

- Frequency range from 15-MHz to 3.35-GHz
- Low jitter
- LVDS applications can be addressed with output levels from 50-mV to 2.00-V
- PRBS from 25-1 to 231-1
- Fast rise times (20%-80%) < 60-ps
- Delay Modulation (jitter emulation)
- Variable Cross-Over Point (eye deformation)
- 8-kBit data pattern memory, RZ, NRZ, R1, Burst capability
- 12-MBit extended pattern memory
- Graphical User Interface
- All Inputs and Outputs are SMA connectors

Figure 20. Keysight’s 81134A 3.35-Gb/s pulse/pattern generator

Related Literature

Publication title	Publication type	Publication number
Jitter Solutions for Digital Circuits	Brochure	5988-8427EN
Jitter Measurement Solutions	CD	5988-9350EN
Measuring Jitter in Digital Systems	Application Note 1448-1	5988-9109EN
Signal Integrity Solutions	Brochure	5988-5405EN
Signal Integrity Solutions	CD	5988-6915EN