Understanding and Measuring Jitter

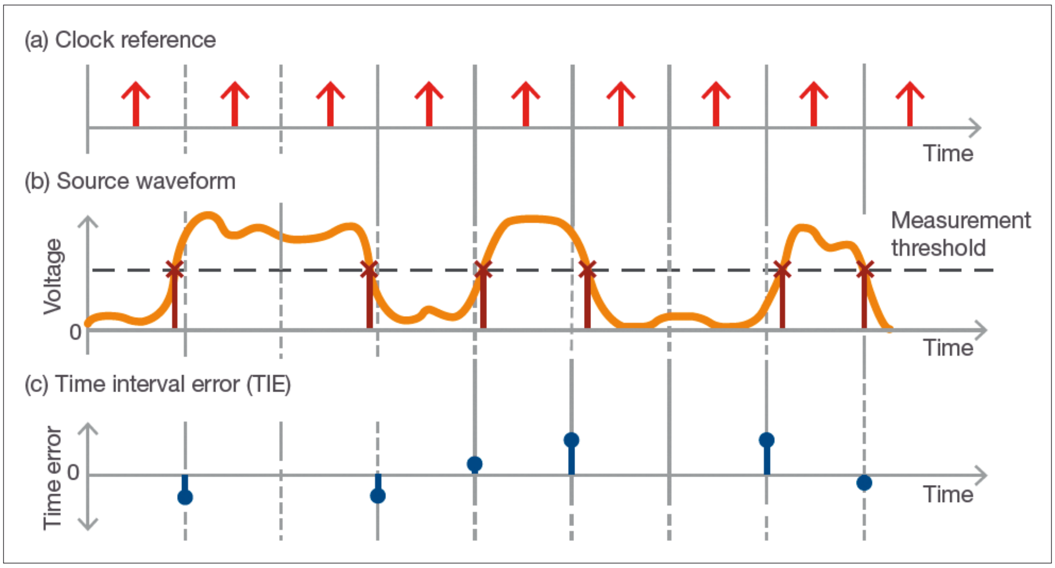

Figure 1. Simultaneous clock, waveform, and resulting TIEs

Clean signals matter. If there's too much jitter on the signal you're measuring, the receiver will end up decoding something completely different from what was transmitted.

Quelle:
White Paper "Left Turn or Lake Front? Understanding and Measuring Jitter" by Keysight 2019

Keysight Technologies, Inc.
www.keysight.com

Imagine riding in the prototype of a self-driving car you're developing the GPS system for the car wasn't tested properly, and unfortunately there is a lot of jitter in the remote GPS's transmission. This may cause the receiver in the car to misinterpret a "turn left" command as a "turn right." Now you find yourself in the middle of a lake rather than safely heading off to your destination. And even worse, your worst nightmare becomes reality – the design doesn't get approved.

What is Jitter?

Today's R&D engineers have to produce cleaner, faster digital designs in less time with expanded validation requirements. A key step along the way is reducing the amount of jitter introduced to your design by understanding it, measuring it properly, and finding the source. The less jitter present, the more likely your design will be stable and pass to the next round in the development cycle.

Fully understanding jitter is critical for creating a stable design. Let's look at what jitter is, why you should care about it, and how to analyze it. You need for example Keysight EZJIT Infiniium oscilloscope software to make automated jitter measurements at the touch of a button.

Jitter is when noise and phase variations occur on edges, causing mistiming in the signal. As a simple example, consider a basic data signal (the orange trace in Figure 1). To analyze an embedded device for serial data applications, a reference clock is derived from the incoming data stream and used jointly with the incoming signal at the receiver to reconstruct the data. The reference clock is produced in the clock recovery circuit and allows the receiver to essentially be "looking" at ideally spaced points in time.

It sees what the voltage of the signal is at each of those points. Based on what it interprets from that process, it reconstructs the data stream, which should ultimately match what the transmitter sent.

However, problems arise if there is a significant amount of jitter introduced into our signal. If the bits in the received signal have a lot of jitter, they will not properly synchronize with the reference clock. This means the receiver could end up seeing the wrong bit in each clock cycle and, therefore, decode the data incorrectly.

The red "x" markings on the green trace in Figure 1 show the timing errors that occur when a signal has jitter. Notice there are some cases where the rising or falling edges occur too soon or too late. This can be seen on a scope using the persistence display mode shown in Figure 2. If a rising edge occurs too late, a receiver will misinterpret the bit. The difference between when the edge crossings actually occur versus when they ideally should occur is called Time Interval Error (TIE).

The difference between when the edge crossings actually occur

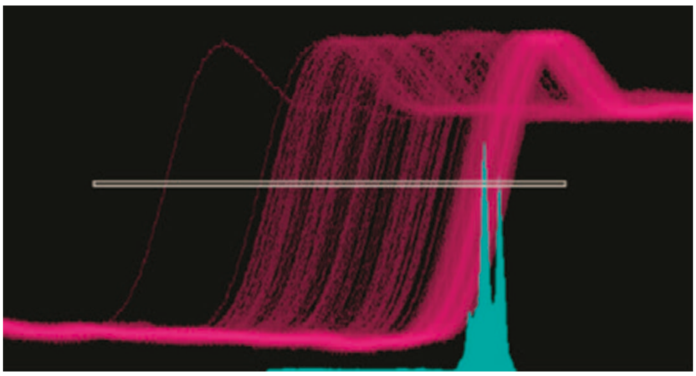

Figure 2. Using the persistence display on a signal shows slight timing errors (known as TIEs)

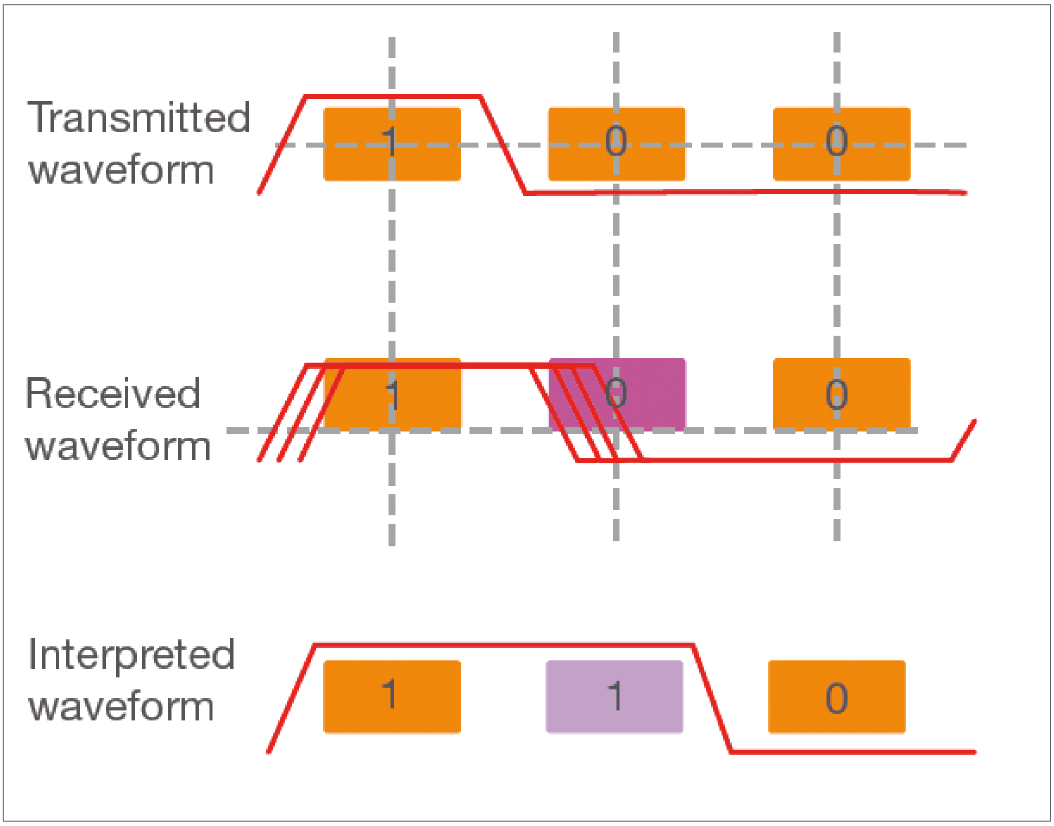

Figure 3. Receiver incorrectly interprets transmitted pulse with jitter

versus when they ideally should occur is called Time Interval Error (TIE).

There will naturally always be some amount of jitter in your signal. In fact, most designs have a specification for jitter tolerance. So, we turn now to the important question: What happens if you exceed that specification?

Why Should You Care About Jitter?

As we mentioned before, if your signal isn't synchronized to the reference clock and the jitter exceeds the tolerable amount, the receiver will end up interpreting bits incorrectly. As a basic example, see how this might happen in Figure 3. The transmitted data was binary 100. However, there was some jitter in the received waveform, and this caused the second bit to appear as a 1 to the receiver when in reality the transmitted bit was a 0. So, the receiver decoded 110.

Consider the example of a drone. Imagine the serial packet at the top of the graphic tells the drone to fly to home base. However, due to jitter, the bits were misinterpreted by the receiver, and the decoded command tells the drone to fly east. Your precious design may end up flying through a window rather than landing at the designated location. Having such an error prevents your design from ever being approved to move forward in the development process.

No matter how good your design may be, there will always be a certain number of bits received incorrectly. The ratio of incorrect bits to total bits sent is called the Bit Error Ratio (BER). You obviously want this number to be as low as possible but certainly at a minimum below the specific standard's stated goal; for instance, USB 3.0 is 1E12 BER. To limit the BER, you must understand the different types of jitter that contribute to these bit errors.

Types and Components of Jitter

To ensure the jitter in your signals doesn't reach harmful

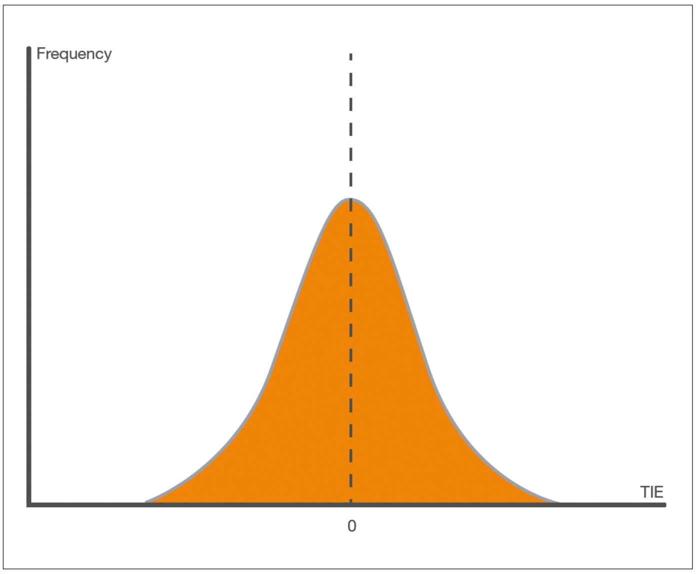

Figure 4. Histogram with Gaussian distribution, caused by RJ

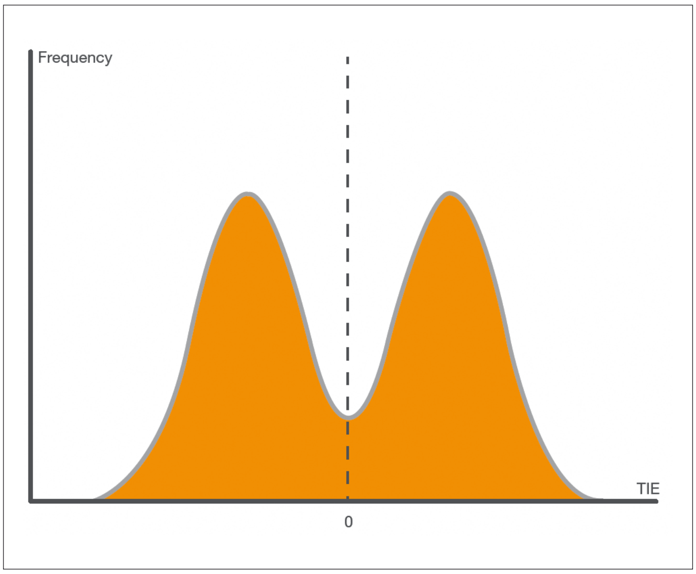

Figure 5. Histogram with bimodal distribution, caused by DJ

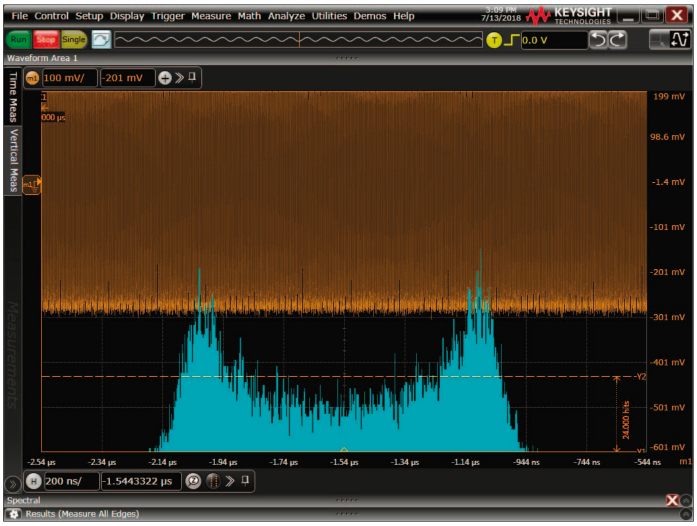

Figure 6: Keysight's EZJIT application enables a histogram view of a partially Gaussian and partially bimodal distribution due to presence of both RJ and DJ components

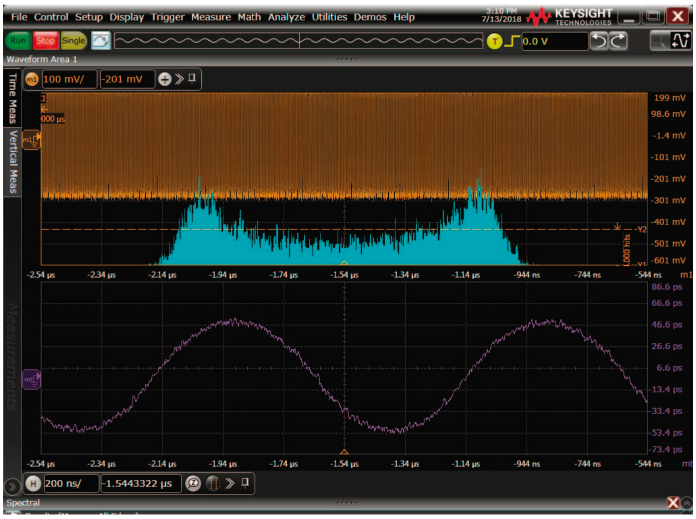

Figure 7. TIE trend plot generated using the EZJIT application shows sinusoidal modulation of TIEs

levels, it's important to understand the various kinds of jitter and where they originate.

Note the model referenced here is the Dual Dirac model of jitter, and please remember this is just a model. This model identifies two types of jitter: Deterministic and Random jitter, along with the assumption that you can identify the deterministic peak-to-peak jitter and that the random jitter deviates from that. The most important thing for you to take away here is the understanding of these types of jitter.

With this model, two main types of jitter make up the total jitter (TJ) in a signal:

1. Random jitter (RJ): This is inherent jitter that is harder to eliminate. This is caused by factors like thermal noise, shot noise, and pink noise. There are some steps you can take to reduce this in your device, but you will never completely get rid of it.

2. Deterministic jitter (DJ): This is caused by your device's design flaws and physical limitations. This includes Duty Cycle Distortion (DCD), InterSymbol Interference (ISI), Sinusoidal (or periodic) jitter (PJ), crosstalk, impedance mismatch, and more. You are more likely to be able to control and eliminate this kind of jitter.

Now that you know the types of jitter and their various components, how can you useyour oscilloscope to analyze this jitter and develop solutions to problems that mightbe causing it?

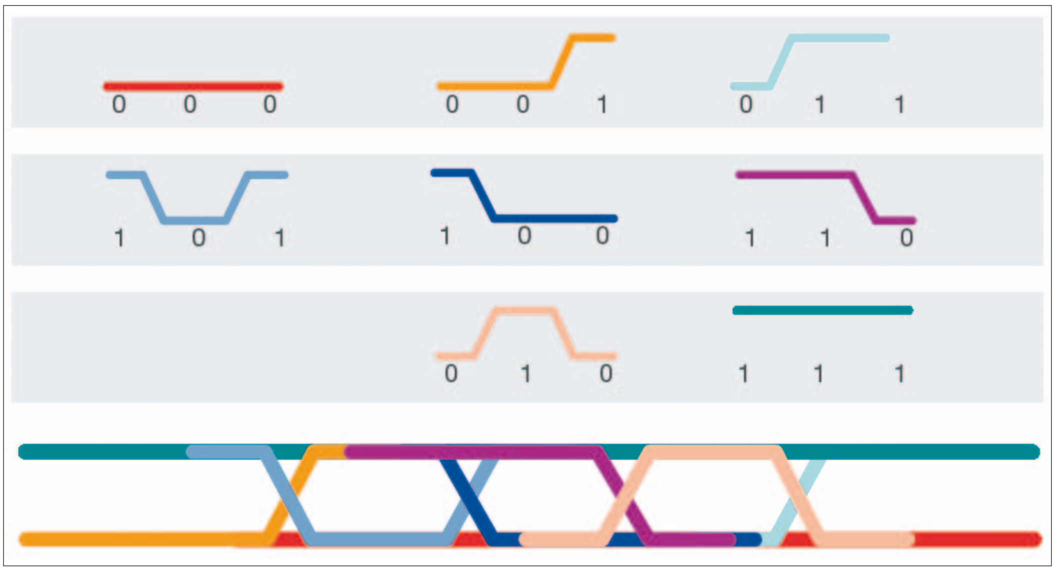

Figure 8. The eight-bit transition combinations and their superimposition forming an eye diagram

How Do You Measure and Understand Jitter?

There are several different ways to visualize and measure jitter using an oscilloscope. The first step in measuring jitter is to adjust the oscilloscope's time base to capture multiple periods of your signal. The oscilloscope then compares these periods to the reference clock. Remember the reference clock is produced in clock recovery.

This reference clock provides the ideal bit rate, which the oscilloscope needs to determine whether a signal is in line with what is ideal or where there are errors. From this comparison, the oscilloscope derives the TIE values, which can then be viewed in various formats.

The ideal bit rate can be completely calculated by the scope using an estimate you enter, or manually entered as the exact specification. The latter is the most accurate, while the first option is the least accurate but easiest to set up.

Once the bit rate is set, the analysis begins. Keysight's EZJIT application for Infiniium oscilloscopes makes it easy to measure and analyze jitter in a variety of graphical formats.

Histogram

The histogram displays the TIE timing values on the x-axis with the frequency those values occur in the signal on the y-axis. The most important function of the histogram is that it helps you determine whether the jitter in your signal is random or deterministic and in roughly what proportions.

If a signal has only RJ, the distribution appears Gaussian, as in Figure 4. This is because random jitter typically has a Gaussian distribution, in which case, the TIEs will center around zero. Most of the TIE values are near zero, and there are fewer values the farther away we get (in other words, there will be fewer very large errors). This means the probability that RJ causes highly noticeable jitter is small.

On the other hand, if a signal has a significant amount of DJ, the distribution appears different from the Gaussian distribution. One example is a bimodal distribution, similar to what you see in Figure 5. The histogram centers around two distinct points rather than just one. This is because, for this example, DJ has sinusoidal modulation or periodic jitter, as you will see when we discuss TIE trend plots. Rather than centering around zero, the TIE values center around the high/low peak values. Because of this, DJ is more likely to affect the signal you're working with.

Jitter is never completely random or completely deterministic. In almost all cases, the histogram has both Gaussian and non-Gaussian characteristics, as shown in Figure 6. But when you look at the histogram, you can determine whether the shape appears more Gaussian (indicative of RJ, which you can't do much to fix) or more non-Gaussian (indicative of DJ, which you might be able to fix).

TIE Trend Plot

Another format for displaying jitter is a TIE trend plot, which displays the TIEs on the y-axis and the points in time at which they occur on the x-axis. This shows the TIE trends over time, which enables you to see if there is any modulation or a repeating pattern of errors. You can see a TIE trend plot in Figure 7 as the purple trace. In this example, the trend plot is displayed alongside the corresponding oscilloscope trace (orange). You can now clearly see there is in fact sinusoidal modulation of the TIEs, caused by DJ.

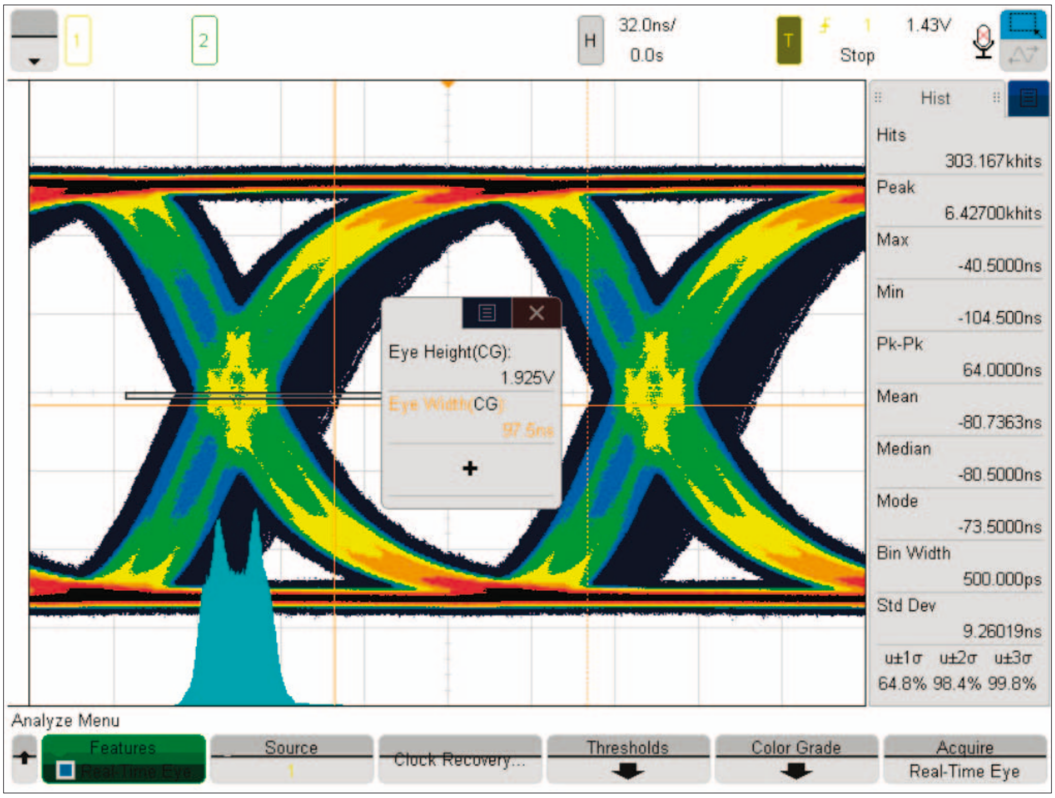

Figure 9. Keysight's Real-Time Eye software sets up an automatic eye diagram of your signal. Note that the colors of the screen are inverted for easier viewing in this white paper. The eye diagram accumulates data over time, which is why it seems to "fill in" the longer it is left to acquire pulses. You may see the opening of the eye gradually shrink over time simply because more instances of jitter are being captured. However, if you have a very stable design, you shouldn't see much change

Eye Diagram

One of the most popular formats for viewing jitter is an eye diagram. As the signal comes through, it is sliced into bit transition combinations (or series of three bits), and these individual traces layer on top of each other. There are eight possible bit transition combinations in total, seen in Figure 8.

Eye diagrams use color grading to show how frequently/infrequently signals come through different areas of the diagram, providing another way to view the TIE's frequency. Measuring the opening of the eye gives you a glimpse into how much jitter you have. The wider the eye, the less jitter in the signal. The more closed it appears, the more jitter present.

An example of this can be seen in Figure 9. This eye shows significant timing errors with the coloring in the histogram making it clear roughly how often they occur. This can be set up automatically on the instrument, making it easy to see how RJ and DJ (or PJ) are affecting your device.

Key Takeaways

You've learned a lot of information about jitter, but there are four high-level points to remember:

- Jitter is defined as unwanted timing errors in a signal. This causes edge crossings to occur a little early or a little late (TIE).
- Excessive jitter results in incorrect interpretation of bits on the receiving end, ultimately causing potential system errors.
- When you see a Gaussian distribution on a histogram, this indicates random jitter, and there is not much you can do about this. However, if you see a non-Gaussian distribution, this indicates deterministic/periodic jitter, which is more easily controlled and reduced.
- Use your oscilloscope's histogram, TIE trend plot, and eye diagram to fully understand how much jitter is on your signal and whether it is within the acceptable range. If not, you need to tweak your design. ◄