A Jitter Education:
Finding a Place for Jitter Analysis in the EET Curriculum

Gene L. Harding
Purdue University

Abstract

Timing jitter has become a major issue in the high-speed electronics industry during the past several years. It is the phenomenon seen when the rising or falling edge of a digital waveform appears before or after the expected time. This paper describes the basics of timing jitter, how to measure and display it, and proposes where and how to incorporate various jitter topics into a four-year electrical engineering technology (EET) curriculum.

Introduction

Timing jitter is the phenomenon seen when a digital waveform's transition appears before or after the expected time. If this time displacement is large enough to place the edge into an adjacent clock cycle, the result is a data error on the bus. With the high speeds of today's bus architectures, jitter that used to be negligible is now very important.

Troubleshooting and correcting jitter problems can be a daunting task. It requires knowledge of the different types of jitter, understanding of their underlying causes, and skillful application of the right test equipment.

This paper is divided into five sections. The first covers the basics of jitter, including the difference between random jitter (RJ) and deterministic jitter (DJ), a brief description of different types of DJ, and typical causes associated with each. The second and third sections discuss the types of jitter measurements and ways to display them, respectively. Several different test instruments can be used to measure and analyze jitter. These are covered briefly in part four, along with some of their strengths and weaknesses. The reader who is already familiar with jitter analysis and the test instruments discussed (real time and sampling scopes, logic analyzers, and bit error ratio testers) may want to skip to part five. This section suggests how to weave jitter topics into a four-year electrical engineering technology (EET) education, as well as which topics might be appropriate for all EET students versus which topics should probably be reserved for digital specialists.

Jitter has two fundamental components: random (RJ) and deterministic (DJ). RJ has a Gaussian distribution. It is usually caused by thermal effects, or other events that are largely beyond the designer's control. Deterministic jitter, on the other hand, is generally caused by phenomena that the designer can control, or at least mitigate. If this DJ can be accurately analyzed and traced to its source, it can often be significantly reduced.

There are three basic types of deterministic jitter: duty cycle distortion (DCD), intersymbol interference (ISI), and periodic jitter (PJ).[1] DCD can be caused by two different conditions: incorrect voltage threshold and asymmetric edge rates. An incorrect voltage threshold changes the pulse width of a waveform, even if the edge rates are symmetric. If the threshold is set too high, it makes the positive pulses narrower and the negative pulses wider. From a jitter perspective, this translates into positive jitter on the rising edges and negative jitter on the falling edges. This is illustrated in Figure 1, which shows the correct output of a transmitter, and the distorted output caused by its threshold being set too high.

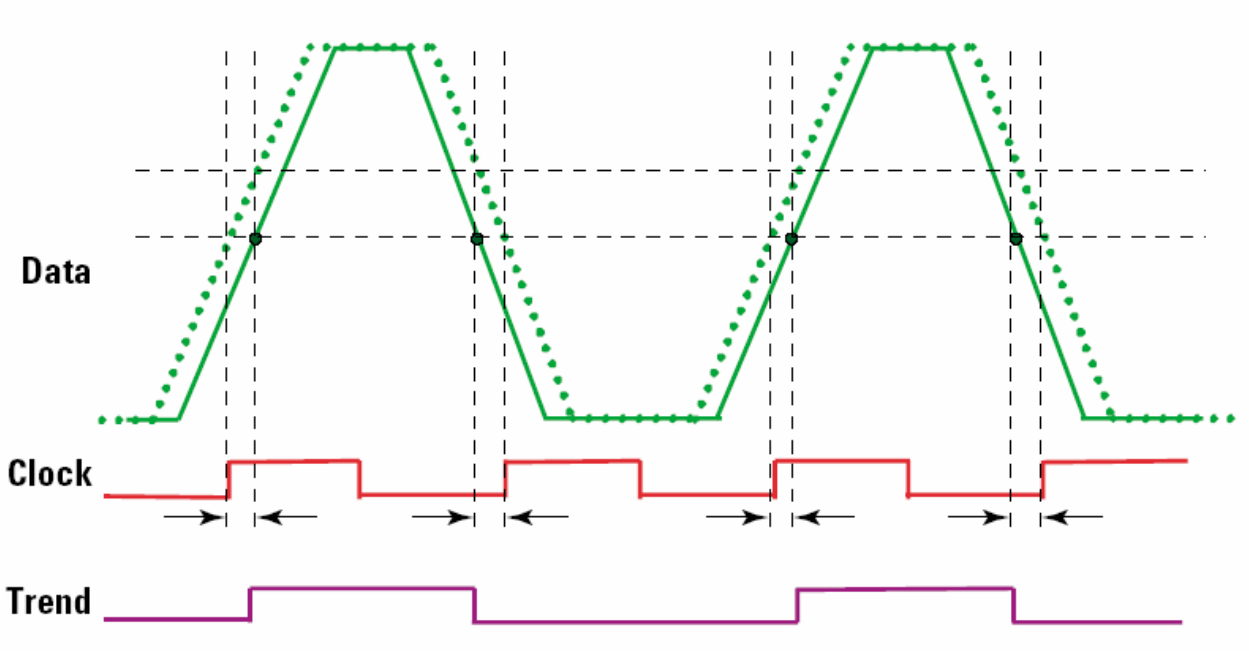

Figure 1: DCD caused by incorrect threshold [1]

Conversely, different edge rates on the rising and falling edges of a digital pulse train can cause DCD jitter, even if the voltage threshold is set correctly.[1] For instance, if the rising edge rate is faster than the falling edge rate, this would widen the positive pulses and narrow the negative pulses. Such a condition translates into negative jitter on the rising edges and positive jitter on the falling edges, as shown in Figure 2.

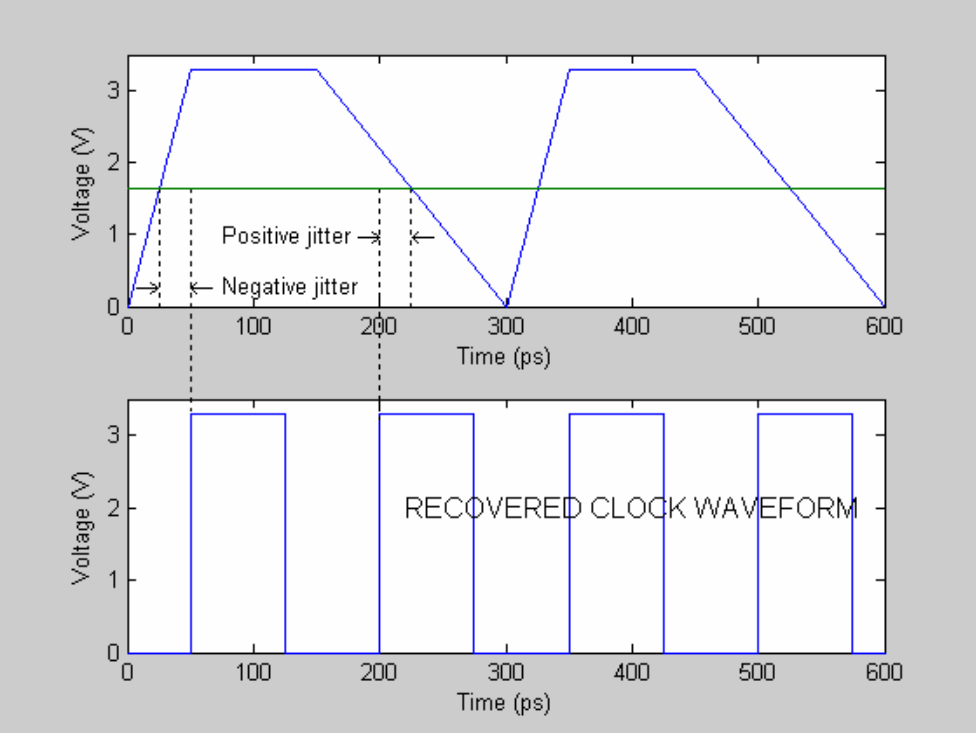

Figure 2: DCD caused by asymmetric edge rates

Intersymbol interference (ISI), also called data-dependent jitter, is caused by a data rate that is too high for the analog bandwidth of a system[1]. This means that, for patterns like 1-0-1-0-1-0, the signal does not have enough rise or fall time to fully traverse the voltage step from low to high (or high to low), which results in earlier edge crossings and negative jitter. Repeating patterns, such as 1-1-1-0-0-0, provide more time for the waveform to reach its steady state value, which causes later edge crossings and positive jitter. Figure 3 illustrates ISI, where the signal with a lower peak value has an earlier transition because it never reaches its steady state value.

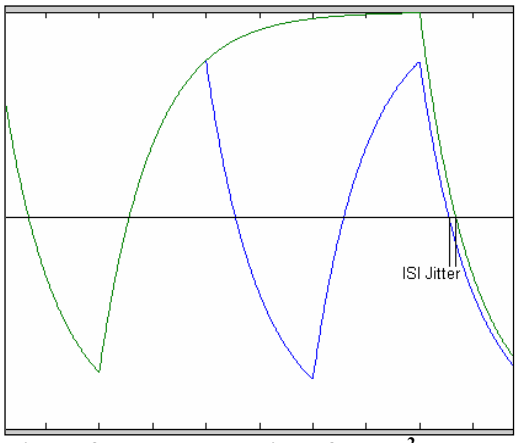

Figure 3: Intersymbol interference[2]

The third type of deterministic jitter is periodic jitter (PJ). It is generally caused by EMI[1], and is called periodic because it repeats in a cyclic fashion[2], based on the coupled signal's characteristics. PJ is sometimes referred to as sinusoidal jitter since any repeating waveform can be broken down into a Fourier series of individual harmonics. A test instrument that can perform a Fourier transform on the signal of interest can be very handy for isolating this type of jitter. There are two types of PJ: correlated and uncorrelated[1]. Correlated jitter comes from EMI that is internal to the system, generally another signal based on the same clock. It can be somewhat tricky to isolate, since its frequency spectrum characteristics overlay with those of the signal of interest. Uncorrelated jitter is caused by signals based on different clocks. These signals can be either internal or external to the system, but are somewhat easier to isolate because they generate frequency (spectrum) spurs apart from those caused by the signal of interest.

So, total jitter is composed of both random (RJ) and deterministic (DJ) components. The three types of DJ are: duty cycle distortion, intersymbol interference, and periodic jitter. DCD can be caused by an incorrect threshold level or asymmetric edge rates, and ISI is caused by system bandwidth limitations. PJ is caused by EMI from other signals, which may or may not be correlated to the signal of interest. This leads us to the types of jitter measurements.

Jitter Measurements

There are three basic jitter measurements that can be done on a single waveform: period jitter, cycle-cycle jitter, and time interval error (TIE).[2] (Note the difference between periodic jitter, which is a type of jitter, and period jitter, which is a type of jitter measurement.)

Period jitter is computed by first measuring each clock cycle in a waveform. The period jitter is the region between the minimum and maximum periods. It can be viewed using a digital storage oscilloscope (DSO) by setting the scope to display a little more than one full clock period and trigger on one edge (either rising or falling). The period jitter can then be seen on the edge that starts the next clock cycle.[2]

The cycle-cycle jitter measurement comes from the period jitter measurement. Cycle-cycle jitter is the *difference* in period between any two *adjacent* clock cycles. It "can be of interest because it shows the instantaneous dynamics a clock-recovery PLL might" need to accommodate.[2] There is also a variation of cycle-cycle jitter called n-cycle jitter, which measures the difference in period between *non-adjacent* cycles. One application of this would be for a system using double-data-rate (DDR) clocking. In this instance, it might be useful to separate jitter of data clocked on the positive edge from data clocked on the negative edge (2-cycle jitter).

Unlike period jitter and cycle-cycle jitter, time interval error requires knowledge of the ideal transition times of the reference clock. This normally requires some type of clock recovery algorithm, which can be done either with hardware or software. (Note that hardware clock recovery also adds jitter to the system.) The "ideal" clock period is called a unit interval (UI). TIE is calculated by subtracting the time of each ideal transition from the time of the actual signal transition. TIE shows the cumulative effect of jitter. When it reaches ±0.5 UI, the data valid region (also called the "eye") is closed and the system will experience bit errors.[2]

Now that we have looked at what jitter is and how it is measured, the next question is how to display it. There are several ways of displaying jitter, including histogram, trend, frequency spectrum, eye diagram, and bathtub curve.

Jitter is often characterized graphically as a histogram, with the horizontal axis representing time and centered about the "ideal" time for the digital transition. The jitter is considered to be positive if the actual edge occurs after the ideal time, and negative if the edge occurs before the ideal time, as shown in Figure 4.

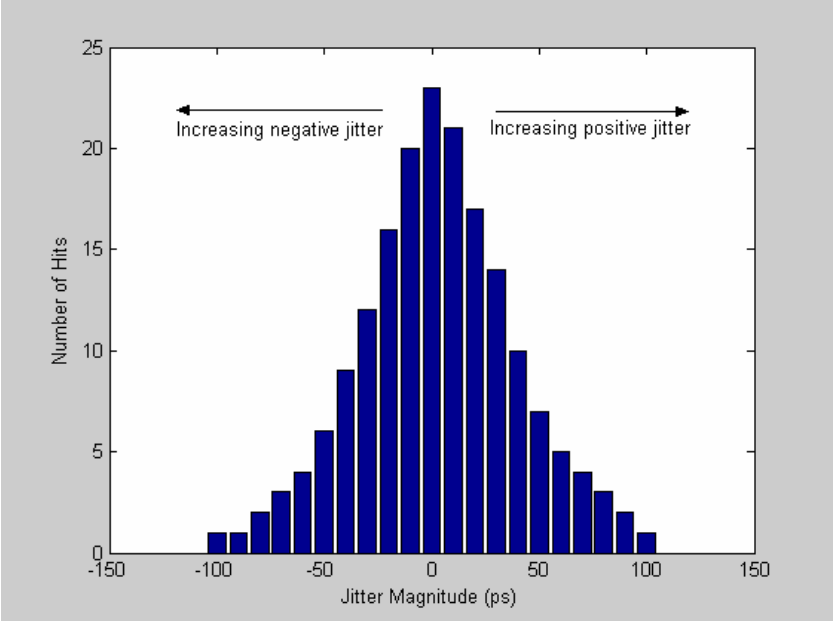

Figure 4: Jitter histogram

The vertical axis of the bar chart represents the number of "hits," or occurrences, of an edge at that moment in time. A measurement of jitter over many transitions that is mapped into such a histogram forms a probability distribution function (PDF). If the measurement is sufficiently large, this PDF is a good indicator of the likelihood of jitter of a given magnitude. Because all jitter contains some random component, the statistical measures of mean and standard deviation are needed to properly characterize it.[2] For deterministic jitter, the additional parameters of max, min, and peak-peak are also used.

Another method of displaying jitter measurements is called a "trend" waveform, which plots jitter magnitude (vertical axis) as a function of time (horizontal axis). Its name probably comes from the fact that one can often spot trends in the jitter (e.g., a jitter peak every four clock cycles) by observing the trend waveform. It is also useful when trying to correlate jitter to other signals.[2] This can be done by displaying the jitter trend and "other" waveform of interest together. If coupling from the other waveform is causing jitter, the trend display will often show maximums and minimums that correspond to transitions from the interfering waveform.

A third way to view jitter measurements is the spectrum, which is just a Fourier transform of the data to show frequency on the horizontal axis and jitter magnitude on the vertical axis. This display can be very helpful when trying to isolate uncorrelated periodic jitter.[1]

A fourth technique for displaying jitter is called an eye diagram. It is created by overlaying many short segments of a waveform so that the ideal edge locations and voltages are aligned, as shown in Figure 5.[2] An advantage of the eye diagram is that it includes voltage level

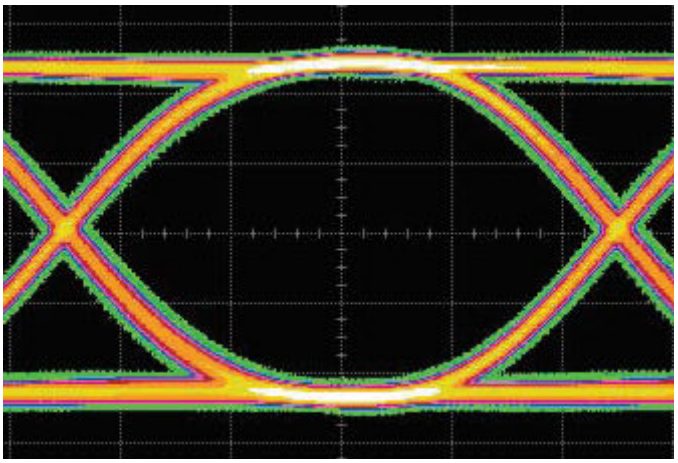

Figure 5: Eye diagram

information. The area in the middle of the eye, at a specified voltage level, is called the data valid window. It is in this region that the data is stable for sampling. Excessive jitter will reduce the data valid window to the point that it does not meet setup/hold requirements, causing data errors in the system. The problem with an eye diagram is that, since all jitter has a random component, a transition will eventually occur through the middle of the data eye. The question is how often that happens. If it happens once in every 1000 bits, it is generally a serious problem. If it happens once in every 10^{12} bits, it may be perfectly acceptable. That is where the Bit Error Ratio Tester (BERT) comes in.

A BERT can characterize jitter indirectly, with a plot of bit errors as a ratio of total bits transmitted. The result is a bathtub plot, as shown in Figure 6. The horizontal axis is one unit

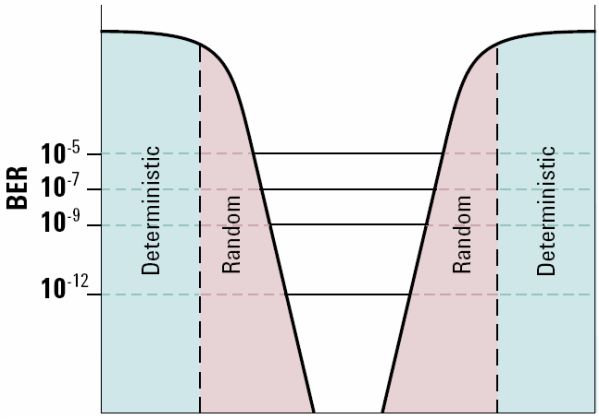

Figure 6: Bathtub curve model

interval; the vertical axis is the ratio of bit errors to total bits transmitted. The distance "across" the bathtub plot at a given ratio is equivalent to the width of the data valid window from the eye

diagram at that data rate.[2] For example, if the bath tub plot shows a width of 0.5 UI for a bit error ratio of 10^{-5}, then a 0.5-UI-wide data valid window on the eye diagram will (statistically) have one transition cross it out of 10^5 transmitted bits.

So, that is a look at what jitter is, how it is measured, and how the measurements are displayed. There are several tools that can be used to measure it.

Jitter Analysis Tools

The measurement tools used to characterize jitter can be divided into three categories: real time, sampling, and indirect. Real time tools, such as standard oscilloscopes and logic analyzers, sample a waveform at many points in order to accurately represent each captured signal. Sampling oscilloscopes have a much lower sampling rate, and depend on a signal's repetitive nature to accurately reproduce it. An instrument that measures indirectly, such as a bit error ratio tester (BERT), measures a different parameter (e.g., bit error rate), but can use that data to derive information about jitter. Each has inherent advantages and disadvantages.

The real time digital storage oscilloscope (DSO) is ubiquitous in the engineering world. Most people doing electronics design and analysis have easy access to a DSO, and are reasonably familiar with how to use one. Moreover, it is a general purpose tool that can be used for many things other than jitter analysis, and is relatively inexpensive compared to some other measurement instruments. Because of its high sampling rate, the DSO is normally pretty fast compared to some of the other instruments listed above. On the other hand, it is also generally lower in bandwidth (BW), and requires a specialized software package to do anything but the most basic jitter analysis.

Logic analyzers, although much less common than DSOs, are nevertheless widely used in digital design. They are more specialized instruments, tend to be more expensive, and are limited in bandwidth like DSOs. But, they can have some utility for analyzing jitter, depending on the speed of sampling and time base accuracy. Eye Finder™ and Eye Scan™ from Agilent Technologies, although only available on higher end analyzers, are surprisingly effective for characterizing jitter in digital systems.

Next is the sampling oscilloscope, which operates differently from a real time DSO. These are available with very high analog BW, in the tens of GHz. They do not, however, sample fast enough to reconstruct a waveform from a "single shot" data capture, and are thus only effective at analyzing repetitive signals. The lower sampling rate makes them significantly slower than a traditional DSO, and they can be pretty expensive. For extremely fast signals, however, the sampling oscilloscope is one of the few tools that can do the job at all.

Another tool that will work in the high-BW realm (tens of GHz) is the bit error ratio tester (BERT). It is arguably less intuitive than an oscilloscope or logic analyzer, but has the distinct advantage of being able to perform compliance tests that other instruments simply cannot do. Some of today's digital communications standards specify errors on the order of one in every 10^{12} bits transmitted. BERTs are made for this type of testing, and can do jitter analysis along

the way. They can, however, be slow. BERTs can "project" bit error ratios based on smaller data sets and statistical analysis, but actually sampling 10^{12} bits for a rigorous test takes hours. Finally, although BERTs are expensive, specialized instruments, they are also powerful, effective, and necessary for some application testing.

So, the real time DSO, logic analyzer, sampling oscilloscope, and BERT can all be used to measure jitter. The final question is how to integrate jitter basics, measurements, displays, and tools into an EET curriculum.

Jitter in the EET Curriculum

In-depth jitter analysis is much too complex to cover all at once in an undergraduate curriculum. The basics, however, are suitable for first and second year students. It makes sense to introduce basic topics in the first two years, then ramp up to more technically challenging subjects in the third and/or fourth years of an EET program. Moreover, the early topics are worthwhile for all students, not just those planning to specialize in the digital arena. For instance, RF system designers must contend with phase jitter in transmitted/received signals. By including jitter basics in the underclassman portion of the curriculum, all EET students would be exposed to those concepts. More advanced topics could be reserved for junior and senior digital classes, with a focus on analysis of those applications. The following paragraphs suggest some topics to cover within an eight-semester undergraduate EET curriculum, and when might be appropriate times to address each one. Subjects in the first six semesters are necessary to adequately and thoroughly address jitter analysis. The advanced topics in the final two semesters would be nice, but due to the expense of the lab equipment, may not be practical for many programs. The information contained in the following paragraphs is summarized in table 1, at the end of this section.

The first semester is arguably too early to introduce the subject of jitter. It is better at this point to concentrate on the basics of analog and digital electronics. It is appropriate, however, for students to begin learning about an important tool for jitter analysis: the real time oscilloscope. Whether DSO or an older analog scope, first semester students should see and use a real time scope, and become familiar with its basic operation.

At our school, basic timing issues are discussed during the second semester, along with the concepts of sampling and aliasing for DSOs. This would be an appropriate time to introduce jitter and wander, and to explain the difference between random and deterministic jitter. If available, students should use DSOs in the lab.

The third semester digital class (e.g., microprocessors) would be a good time to discuss the different types of deterministic jitter: duty cycle distortion, intersymbol interference, and periodic jitter. Intersymbol interference leads naturally to 8B/10B encoding. Although such encoding would not necessarily have to be covered in great detail, the idea behind it could certainly be covered at a high level. Additionally, it would be fitting to introduce eye diagrams and the DSO's infinite persistence mode.

By the fourth semester, if not before, students should be taught the basics of logic analyzers. This would include asynchronous vs. synchronous modes; probing considerations; inverse assembly and source code correlation; cross-triggering and time correlation; and jitter analysis technologies (such as Agilent's Eye Finder™ and Eye Scan™). The topic of probing considerations should be expanded to include oscilloscope probing. Simple passive probes are not acceptable to capture signals at speeds over a few hundred MHz, and it is important that designers understand the issues involved with high-speed active probing.

In semester five, students should lay the foundation for in-depth jitter analysis. This begins with an elementary course in probability and statistics, so they become familiar with probability distribution functions, the Gaussian distribution, and the statistical terms of mean, variance, and standard deviation. (These concepts could be used in semester six.) It also includes an introduction to the topics of clock recovery and the unit interval. Although not directly related to jitter analysis, a logic analyzer lab would be nice, if the equipment is available (especially if the plan is to eventually do jitter measurements with the analyzers).

Semester six is the time to really dig into jitter analysis. With the topics covered in the previous five semesters, students should be ready to discuss jitter measurements (period, cycle-cycle, and n-cycle jitter, as well as time interval error) and ways to display them (histogram, trend, spectrum, eye diagram, and bathtub curve). To discuss the bathtub plot requires knowledge of the bit error ratio tester (BERT), so this would be a perfect time to introduce the BERT and sampling oscilloscope. It would also be the time to discuss jitter analysis software, at least for real time oscilloscopes. Agilent Technologies, Amherst Systems, LeCroy, and Tektronix all sell such packages. Ideally, the course would compare the different vendors' packages. More realistically, students could use one of the packages (depending on which scopes are in the lab, etc.) in the context of learning display methods. Finally, they would learn techniques for separately quantifying the different types (both RJ and DJ). This would culminate in a lab where the students are presented with a signal containing various types of jitter, and would be required to determine what types of jitter are present, along with the magnitude of each.

Semesters seven and eight could include an optional lab (one each) for the sampling oscilloscope and BERT, if available. Because these instruments are so expensive, a demo using an overhead projector might be the only practical way to do each lab.

The topics discussed in this section are organized by semester in table 1, below.

Table 1: Jitter topics by semester

Semester	Topic(s)
1	Real time oscilloscope introduction
2	Real time DSO topics: sampling, aliasing Jitter vs. wander Random vs. deterministic jitter Use DSOs in lab, if available
3	DCD vs. ISI vs. PJ, and 8B/10B encoding Infinite persistence with a DSO, and eye diagrams
4	Intro to logic analyzers: asynchronous vs. synchronous modes, inverse assembly & source code correlation, cross-triggering & time correlation, jitter analysis technologies Probing considerations (including oscilloscopes and logic analyzers)
5	Probability and statistics (probability distribution functions), Gaussian distribution, mean, standard deviation Clock recovery, unit interval Logic analyzer laboratory (if available)
6	Jitter measurements (period, cycle-cycle, n-cycle, TIE) Jitter measurement displays (histogram, trend, spectrum, eye diagram, bathtub curve) Real time DSO jitter software (Agilent, Amherst, LeCroy, Tektronix) Distinguishing and quantifying different types of DJ Introduce BERTs and sampling oscilloscopes Jitter analysis laboratory
7	Laboratory with a sampling oscilloscope (optional)
8	Laboratory with a BERT (optional)

Conclusion

The first four parts of this paper covered the basic types of jitter, its measurements, how the measurements are displayed, and the tools used. The final section proposed where to integrate various jitter and related topics into a four-year EET curriculum, by semester. Although there is much detail left to develop the actual curriculum materials, jitter analysis could be effectively covered in the first six semesters. Perhaps the toughest challenge, however, will be deciding what topics in a given curriculum are dated enough to drop in favor of the jitter analysis topics.

[1] Finding Sources of Jitter with Real-Time Jitter Analysis, Agilent Technologies Application Note 1448-2, 25 Jun 2004.
[2] Understanding and Characterizing Timing Jitter, Tektronix primer, Oct 2003.

GENE L. HARDING is an assistant professor of electrical and computer engineering technology at Purdue University. He worked with logic analyzers and oscilloscopes for three years at Agilent Technologies, and has over 17 years experience with the U.S. Air Force doing electronic warfare, wide- and metropolitan-area network management, technical intelligence analysis, and missile defense system software support. He holds MSEE and BSEE degrees from Rose-Hulman Institute of Technology.