

Introduction

This application note discusses Timing Signal Jitter, what it is, how it is specified and how it is measured. Jitter specifications herein are approached from a practical point of view and do not rely on complex math to derive the exact meaning of each specification. Also, detailed technical treatments of timing signal jitter are provided for those interested in pursuing this topic beyond this application note. Timing Signal Jitter is specified in a number of different ways using terminology that maybe specific to an application environment. For example, in digital data path synchronization applications jitter is specified in the time domain as Period or Cycle-to-Cycle jitter. However, in data communications applications jitter is specified in the frequency domain as phase noise or RMS jitter. This document will cover both time and frequency domain specifications, relate the two and explain why different applications uses different specifications.

What is Jitter in a Timing Signal?

First, a timing signal is defined as a repetitive digital signal as shown in [Figure 1](#). Timing signals are required to synchronize the transfer of data between two points. In digital chips it may be required to sequence data through a set of processing stages in a pipeline. In an analog application, the timing signal is used to sample a DAC to digitize an audio signal. In communications applications, a timing signals is used to synchronize the transmission of data over high-speed serial links such as LANs and Wide Area Networks. In RF applications, timing signals are used to “front end” convert the modulated analog signal into the digital domain for DSP processing and visa versa. In all applications the performance and reliability of the application can be dramatically affected by the quality of the timing signals used.

A perfect timing signal has a fixed period and duty cycle, which never varies over time and its starting point is fixed in time. But in reality, all timing signals have small variations. These “variations” in phase position, period and duty cycle are generally called “jitter”. To measure jitter in the time domain, the timing signal is typically compared to a “perfect reference signal”. The variations of the timing signal’s rising and falling edges as compared to the perfect reference are defined as “jitter” and often specified as time measured from the ideal signal’s edge locations. See [Figure 2](#). These measurements are normally specified in time such as Pico Seconds, (10^{-12} seconds) or 1 trillionth of a second.

Period Jitter, Cycle-to-Cycle Jitter, or Absolute Period Jitter

There are two ways to look at jitter in the time domain: (1) directly compare the reference signal to the jittering timing signal, and (2) in the frequency domain where the power level of the noise or jitter is compared to the power level of the fundamental signal’s power level.

In practice, these preferred schemes are not often used due to the difficult if finding a perfect reference signal at the desired frequency or having access to the equipment needed to do precise spectrum analysis. So most engineers do the next best thing—they look at the timing signal with an oscilloscope and observe the variations in period-to-period or cycle-to-cycle. If the scope is set up to trigger on every rising or falling edge and set to infinite persistence mode and allowed to trace sufficient cycles it is possible to determine the maximum and minimum periods of the timing signal. Digital scopes can also store a finite number of period durations and post-processing software can analyze the data to find the maximum and minimum periods. See [Figure 3](#). This scheme does have several shortcomings related to detecting, the larges values (due to finite sample time), accumulated jitter and phase shifts relative to the missing perfect reference. These most basic time domain jitter measurements are named as follows: Period Jitter, Cycle-to-Cycle Jitter, or Absolute Period Jitter. In many applications, meeting these specifications are all that is required.

- **Period Jitter**—the maximum jitter observed at the end of a period’s edge when compared to the position of the perfect reference clocks edges. The number of cycle used to look for the maximum jitter varies by application, but the JEDEC specification is 10,000 observed cycles.
- **Cycle-to-Cycle Jitter**—the maximum observed variation between two adjacent cycles periods over a defined number of observed cycles. The number of cycle observed is application dependent, but the JEDEC specification is 1000 cycles.

- **Absolute Period Jitter**—the maximum jitter observed for the test signal when compared to the perfect reference edges in the application environment. Sometimes this is also called the 6-sigma jitter value.

An example of an application where Absolute Period Jitter specs are all that are required is when a timing signal is used to drive a digital chip. Here data is processed and transferred between pipeline stages with the timing signal. If the timing signal has too short of a period (due to jitter), there may not be sufficient time to process the data in the pipeline stage and have it setup and ready for the next stage. Since these digital chips have no error detection or correction capability designed in, once a single stage is corrupted, all further processing stages operate on false data and proceed without detection of the problem. Thus, meeting the minimum absolute period requirement is critical in driving digital chip applications. If the period is too long, things still work but there may be a slight loss in system performance, but the system still works and without undetected errors. In other words, any kind of jitter is acceptable as long as it stays within some absolute bounds. One clock edge out of these bounds can cause an undetected and unrecoverable error. To some extent, the digital chip driving application is least tolerant to exceeding jitter specifications.

Accumulated Jitter, Long Term Jitter, or Phase Jitter

It would appear that jitter is really a simple matter of specifying the period jitter and your done. Unfortunately, many applications have significant additional issues related to the timing signal's jitter. First, we have not considered the effects of the jitter on the timing signals phase with respect to the perfect reference signal and, secondly, what if the jitter is not equally distributed on the + and - side over a span of time? These effects result in what is called Accumulated Jitter or Long Term Jitter. Consider what happens when several successive clock periods have a positive jitter value. At the end of this set of clock periods the timing edges could be significantly displaced in time from their ideal locations. See [Figure 4](#). One way of incorporating this effect in the Period or Cycle-to-Cycle jitter specification is to specify the number of cycles over which the jitter is measured. An example of how this is done is specified in the JEDEC Standard JESD65. Different applications may want to specify a different number of cycles over which the jitter is measured.

Applications where Accumulated Jitter, Long Term Jitter, or Phase Jitter are Issues

The most visual example is to consider a timing signal used to drive a pixel output on a CRT. Here, a small variation in the pixel position is tolerated due to the integration effects of the phosphor and the human eye, therefore fairly large cycle-to-cycle jitter is permitted. However, if the jitter is not random and accumulates over a significant portion of the CRT raster, image distortion or swimming is observed due to improper positioning of the pixels. Consider the case where the jitter is actually modulation from a low frequency source, say from a 60 HZ source, and has a period of 16.7 ms. Thus for approximately 1/4 of the period (~ 4 ms), the jitter increases and in the remaining quarter of the period the jitter decreases. In most CRT raster displays the screen refresh rate is 60HZ or higher, thus the full effect of the jitter is easily visible on the screen. If the jitter had a much higher modulation rate, or was random, the effects would not be as visible.

Digital Designs with FIFOs between Clock Domains

It is common in complex digital systems to have various portion of the design running at different clock rates. For example, data acquisition occurs at one clock rate and data processing at another. However, data must cross between the two clock domains reliably. This is typically done using a memory device called a FIFO (First In First Out). A FIFO memory element takes data in at one clock rate and sends it out at another clock rate in the order it was received. Typically, the depth of the FIFO (amount of data elements it can hold) is fixed in size. The typical operation is that on the higher speed clock side of the FIFO, data is written in short bursts at the higher clock rate filling the FIFO. Once full, the writing is suspended using a FIFO full signal sent to the source. When the receiving side sees that the FIFO has data, it proceeds to empty the FIFO at the slower clock rate. Note that reading of the data does not have to wait until the FIFO is full.

If the exact fill rate and burst size and empty rates are known, the FIFO size needed can be calculated. Since the FIFO size is carefully determined under assumptions of frequencies of the fill and empty clock rates, variations in these clocks can have significant affects, for example, causing the FIFOs to over-run or under-run losing data. If jitter accumulates over the fill or empty time of a FIFO and the FIFO depth is fully used by the applications, it is possible to create FIFO over and under-run conditions. Classic examples of where this has occurred in the past are early PC VGA display adapters. FIFOs are typically used between the display memory and the Pixel serialized to facilitate the clock boundary crossing. When Spread Spectrum clock sources with intentionally jittered (modulated) clock sources were used to reduce EMI, many of these designs experienced problems due to FIFO over and under-run conditions caused by the intentionally added accumulated jitter to the FIFO clocks. One solution to this problem is to add additional depth to the FIFO. Of course, existing designs do not have this option.

FIFOs have one very interesting attribute that is often used in communications applications. A FIFO can detect when an under or over-run condition occurs. This permits the implementation of error recovery strategies. How this is used in data transmission applications and how it affects timing signal requirements and specifications will be discussed later in this document.

ADC/DAC Sampling Clocks

Conversion of analog signals to the digital domain and back are highly dependent on the quality of the data conversion timing signal clock. DSP technology has advanced to the state that most of the signal processing is done in the digital domain. This includes functions such as echo canceling, cross talk reduction, and equalization for signal restoration, modulation and demodulation. All of these digital processing functions are highly dependent on the quality of the “front end” data conversion. In turn, the quality of the data conversion is directly related to the quality of the sampling clock signal. Many modulation/demodulation schemes for data transmission over copper or RF use a combination of Frequency Modulation, Phase Modulation and Amplitude Modulation. These complex schemes are used to improve the data transfer efficiency; get more bits/bandwidth block. Thus if the sampled data stream is using all of these modulation schemes and the sampling clock has jitter in phase and frequency, the sampling clock can actually add its own phase and frequency modulation to the digital data making it more difficult for the DSP to process the converted digital data properly and extract the correct data. If the sampling clock accumulates jitter over a “symbol” time, it will begin to sample the next symbol too early or late and create a problem called Inter Symbol Interference (ISI). This is an oversimplification of all the issues, but does give a feel for the issues related to accumulated and cycle-to-cycle jitter in data acquisition applications in “front ends” to RF, LAN and WAN applications. A whole set of similar problems exist in audio and video analog to digital conversion applications.

Data Transmission Clocking

The topic of transmission of digital data over long distances has a huge body of technical work to draw from and is the subject of many textbooks. Some of these texts are devoted entirely to the topic of jitter in data transmission systems. The point is that the descriptions of the issues and definitions of jitter related to this topic are complex and will only be treated at a very high level with many simplifications.

Most Data Transmission schemes use an encoding scheme to embed the clock in the serial data stream such that it can be recovered at the receiving end and be used to sample the clock/data stream recovering the data only portion. Most receivers have a function called a Clock Data Recovery (CDR) Block. This block is typically a PLL or DLL that need sufficient edges in the data stream to lock to the stream and generate the embedded clock. To ensure that there are sufficient edges in the clock data stream, several schemes are used. For example, when long streams of 0s or 1s are encountered, bits of the opposite polarity are inserted using an agreed upon set of rules understood by both transmitter and receivers. A very common scheme is to convert an 8-bit block of data to a 10-bit block of data. The mapping of the 8-bit codes to the 10-bit codes is such that only 10 bit codes with the highest number of bit transitions are used. Obviously the transmitter and receiver must know the mapping table. This, as with all schemes, adds overhead to the data stream. In the case of the 8/10 coding, a 25% overhead penalty is incurred to ensure reliable clock recovery.

Eye Diagrams

One way of measuring the effects of jitter in a data transmission system is to look at what is called an eye diagram. See [Figure 5](#). First, a stable perfect reference is synchronized to the data/clock stream. Using one edge of the perfect reference as a synch, the data clock stream is displayed on an oscilloscope. As one would expect, the edge changes in the clock/data stream jitter around creating an area that is stable. The eye (stable area between the edges) is said to be closing if the jitter gets great. This makes it difficult to extract a stable clock and center that clock in the eye pattern to correctly sample the data. The eye pattern and its opening size are often used to specify the maximum jitter permitted in a digital data stream. Often a clock source is used to transmit the data stream and the eye pattern is observed at the received end to determine the effect of the clock source on the eye pattern opening size.

Errors Are Tolerated in Data Transmission Schemes

Data Transmission schemes make one very important assumption not present in many other clocking applications—there will be transmission errors. It is assumed that the transmission environment will be hostile. Signal degradation and noise (jitter) will cause errors. Further, it may be possible to build an error free environment, but it may not be economically feasible, therefore errors are tolerated to keep costs low.

With this basic differentiating view, many error detection and correction strategies are implemented to deal with this hostile environment. Basically, it is cheaper to design error detection and recovery than an error free transmission environment. One approach to error recovery is to packetize the data. This approach is ideal for solving many digital transmission problems, and also addresses many clocking issues. The concept is simple—after transmission of the packet, a copy of the packet is held until an acknowledgment of error free reception is received. If it is not received properly it is retransmitted.

Data is typically sent in packets, or small blocks of data. The packet approach is used for a number of reasons. The most basic is that if one is lost, it is relatively easy to detect and retransmit the packet. This is accomplished by adding overhead to the packet, such as a packet sequence number, an address for routing and an error detection code (typically a cyclic redundancy check field CRC). In addition, each packet's boundaries must be detectable. This is typically accomplished by beginning the packet with what is called a code violation—a blatant violation of the clock/data encoding scheme rules that is easily detected. For example, in an 8/10 encoding scheme, picking a 10-bit value that is not a valid 8-bit mapping can be used to create a code violation and detect the beginning of the packet. The end of the packet can be determined by another code violation, agreed fixed length of the packet, or a packet length field in the packet's header.

Figure 6 describes a typical data transmission system which includes a number of repeaters or add drop multiplexers (ADM) unit where data is added or stripped from the transmission line. Both the repeaters and ADM recover the data stream and retransmit it with a new stable clock source removing the noise (jitter) added during the transmission segment. Unfortunately, the new reference timing clock is not in phase with the reference clock or the receive data and will likely have a small frequency difference also.

In reality, each segment of the transmission system is a different clock domain, where the difference may be slight but can have a profound effect on performance. This problem is very similar to the FIFO clock domain crossing issues in complex digital designs discussed earlier. In transmission systems, the FIFO is often called an “Elasticity Buffer”. If the two segments have a fixed offset in clock frequency the elasticity buffer cannot hide or prevent over or under run conditions, eventually they will occur. But as the buffers do reduce the frequency of the errors at the expense of some data delay and since they are really FIFOs, they can detect the error and request retransmission of the packet with the error. The packet approach also allows for the implementation of another strategy to deal with the clocking issues over transmission systems. By inserting a variable small gap between packets, clocks that do not carry data can be inserted and extracted to adjust to the differences in clock frequencies between transmission segments.

When all of the above issues of accepting certain levels of error rates, error recovery strategy, packet size, packet gap size, number of retimed segments, and clocking sources are considered, simple specification of period or cycle to cycle jitter may not provide an acceptable cost or performance solution. For example, large jitter that occurs infrequently is perfectly acceptable, since a robust error recovery strategy has been implemented for infrequent errors. Where as, accumulated jitter over a packet length would introduce a potential error in every packet. This cycle-to-cycle jitter could be very small. But if jitter is accumulated over the packet's time interval sufficiently to cause a bit error in the packet, it could become an issue. This all leads to the need to specify jitter differently for these types of applications. In subsequent sections, discussion of how specifying jitter as RMS power and in the frequency domain as phase noise is more applicable to these applications.

Duty Cycle and Half Period Jitter

In many applications the timing clock signal is used to transfer information on both edges of the clock. In these applications the duty cycle becomes important. However, a duty cycle specification may not be sufficient in some applications. Consider the conditions described in Figure 7, where period jitter can be equally divided between the two half cycles maintaining the 50/50% duty cycle, or the jitter can be all in one $\frac{1}{2}$ of the cycle significantly changing the position of the $\frac{1}{2}$ cycle edge. It is obvious that some additional specification is needed to describe the jitter of the $\frac{1}{2}$ cycle edge that may be used to synchronize or transfer data. This specification is the $\frac{1}{2}$ cycle jitter and describes the jitter from the reference period's $\frac{1}{2}$ cycle edge to the timing signals $\frac{1}{2}$ cycle edge.

Where to Measure Jitter on the Waveform vs. Where the Receiver Senses Jitter

There are a lot of reasons why the specified jitter is different than the observed jitter in an actual design. The first step in resolving these issues is to make sure all users are measuring the jitter at the same point on the waveform and at the same positions in the circuits. Also, if the receiving circuit has a switching point or level different than that at which the jitter is measured, the receiver can sense a different amount of jitter that was observed at the specified waveform level. Once the measurement

approach is agreed to, one can go on to resolve the more important issue of where the jitter is coming from; inherent in the source, due to coupling from other signals, improper termination, power sources, etc. Remember, parts are often specified in a perfect environment and perform differently in real applications.

Time Domain Units of Measure of Jitter: % UI, PS, Degrees, Radians, etc.

When jitter is measured and specified, it is typically in ps (pico seconds) from some reference edge and possibly over some number of specified cycles. However, other units of measure can be used and are common in some applications. One way is to express the jitter as a % of the reference period. In many applications the reference period is called a Unit Interval (UI) and the jitter is specified as a ratio or % of the UI. Other measurements used are degrees where: 1 UI = 360 degrees = 1 period = 2 pie radians. A good way to understand the conversions is to take an example. Consider the OC-3 clock frequency of 155.52 MHz with ± 50 ps of jitter, or a total jitter of 100ps.

$$\text{One Period} = 1/155.52\text{MHz} = 1 \text{ UI} = 6.43\text{ns} = 6430\text{ps} = 360 \text{ degrees} = 2 * 3.1418 \text{ radians}$$

$$\text{Jitter as: ps} = 100\text{ps}$$

$$\text{Jitter as ratio of UI} = 100\text{ps}/6430\text{ps} = 0.01555 \text{ UI}$$

$$\text{Jitter as \% UI} = 100\text{ps}/6430\text{ps} = 0.01555 \text{ UI} = 0.01555 \text{ UI} * 100 = 1.555 \% \text{UI}$$

$$\text{Jitter as Degrees} = 100\text{ps}/6430\text{ps} * 360 \text{ Degrees} = 5.598 \text{ Degrees of Jitter}$$

$$\text{Jitter as Radians} = 100\text{ps}/6430\text{ps} * 2 * 3.1418 \text{ radians} = 0.0977 \text{ radians of jitter}$$

Time Interval Error Measurements (Peak-to-Peak Jitter)—2nd Way to View Jitter in the Time Domain

Another way to view jitter in the time domain is to extract just the jitter component of the signal. This can be done in two ways: (1) Describe the jitter as a time interval error from the reference signal over time. At any timing signal period over time that specific period will have a jitter error in time relative to the perfect reference signal. This can be plotted over time and show the continuous jitter effects (error), see [Figure 8](#). The only problem with this approach is, to see the worst jitter error you would have to observe all of the cycle over the time interval of interest. (2) Assume that the jitter is the result of some modulating waveform and if one can extract and plot the modulation waveform a similar record of the jitter is available. The modulation waveform does give some additional information such as how often jitter of a specific value occurs in the time period of interest. Since both of these time domains give views of jitter, one must observe the entire time record of interest to gain an insight relative to maximum/minimum jitter, accumulated jitter, and frequency of occurrence of jitter in a specific range. When the Time Interval Error (TIE) plot is obtained, the maximum excursion of the error is often called the *Peak-to-Peak Jitter*.

- **Peak-to-Peak Jitter**—the maximum observed jitter amplitude in a TIE plot over a specified number of cycles or time.

RMS Jitter

Another way to measure the effects of jitter is to measure its power level. Since we know that the perfect timing signal is a square wave, we know what the RMS power level should be. If we measure the power of the signal with jitter and subtract the power contribution of the perfect reference, the remaining power is the result of the jitter. This gives us a measure of the purity of the signal. Now relate this to the jitter waveform in [Figure 8](#). Analyze this waveform, which is jitter as a function of time, to determine its power content. But first, make some observations about how to calculate power in a waveform. Intuitively, one can see that power is somehow related to the area under the waveform. For simplicity, let's deal with a very specific period waveform first. Consider the Sine wave in [Figure 9](#). Assume this is the AC current of the power delivered to your home. In the first half of the cycle, current flows to your home and in the second half, it reverses and flows to the power company. It would appear that there is no net power delivered. But power is dissipated in a load (delivered) independent of the direction of the current flow. Consider a battery attached to a heating element or a light bulb—the heating element is heated and the light bulb is lit even if the connection of the battery is reversed. This says that power dissipation is independent of current's direction. If the current were constant, like the DC current from a battery, the power is simply given by $P = I^2 R$. One can suspect that for the AC current sine wave there is a value of DC current that delivers the same power as the sine wave. This value is called the **RMS** (Root Mean Square) value of the sine wave and is given as $.707 * (\text{Peak})$ value of the sine wave. Unfortunately, this formula only works for the sine waveform, but the concept of RMS values can be extended to other waveforms. For example, the RMS value of a square wave is $0.5 * (\text{Peak-to-Peak amplitude})$. Referring back to the jitter waveform of [Figure 8](#), there must be some RMS jitter value that represents that same power level as the jitter waveform. But this waveform is not identifiable as a standard waveform and is not periodic. Some assumptions must now be made about the characteristic of the jitter to proceed. One is to assume that the

jitter amplitude has a Gaussian distribution. Remember the bell shaped curve in statistic class. This curve reflects the probability of jitter at a certain magnitude. See [Figure 10](#). For a jitter waveform that has a Gaussian distribution, there is a point on the Gaussian jitter distribution curve that reflects the RMS value of jitter for Gaussian distribution waveforms, just as with the sine wave. It is called the **one sigma value, standard deviation value, or RMS jitter value**.

RMS jitter specifications must make some assumptions as to the characteristics of the jitter to have meaning. Typically, the assumption is that the jitter has a Gaussian distribution. Secondly, RMS jitter spec is a good indication of the purity of the timing signal but does not guarantee period jitter will be absolutely under some value. Consider the case where over 1,000,000 cycles no jitter occurs, but on the 1,000,001th cycle, jitter is 20% of that specific period and this pattern repeats.

This jitter waveform would have a very low RMS jitter value. But if this low RMS value jitter signal were used to drive a Pentium Processor, it would cause errors every few microseconds. In contrast, this signal would work just fine in a data transmission application where this relatively low error rate could be tolerated. Therefore, RMS specifications are typically used in communications applications. Subsequent sections will discuss how RMS jitter is a link between time domain and frequency domain specifications.

RMS Jitter Conversion to Peak-to-Peak Jitter

Often peak-to-peak jitter is specified relative to RMS jitter. If one looks at the jitter distribution curve in [Figure 10](#), other points on the distribution curve can be used to specify jitter. These points are specified as integer multiples of the one sigma value. One specification that is often used is the 6-sigma value. Here, 99.999% of the jitter is accounted for with this value. A common rule used is that the peak-to-peak jitter is 6 times the RMS jitter value. This would be specified as Peak-to-Peak jitter @ 6 Sigma.

$$6 \text{ Sigma Peak to Peak jitter} = 6 * \text{RMS jitter}$$

Jitters Source: Deterministic vs. Stochastic

Jitter is often divided into two types: Stochastic and Deterministic. Deterministic jitter is jitter whose source can be identified and fully described in the time domain. Examples are AC power noise, inductive coupling between two signals, and intentional spread spectrum modulation. Stochastic jitter is jitter, which is entirely random in nature, such as thermal noise in a circuit.

Frequency Domain Specifications

As noted, in the time domain it is difficult to characterize the performance of the timing signal over a long period of time. For example what is the rate of change of the jitter, how often are the peak jitter values reached, what is the maximum accumulated jitter, etc. Additional insight can be obtained by looking at the timing signals characteristics in the frequency domain. But what do we mean by the frequency domain? The following sections discuss some of the basics of frequency domain.

What Is A Sine Wave?

If a ball was placed on a flat surface and rolled at a constant speed, and a point was traced on the circumference of the ball, the waveform generated would be a sine wave. See [Figure 11](#). The radius of the ball and speed of the ball's rotation determine the amplitude and period of the sine wave. The frequency of the sine wave is simply the 1/period. How does this example relate to describing a square wave-timing signal?

Fourier's Series

In 1822, the French mathematician Fourier wrote a paper on heat transfer theory that contained a supposition that all periodic waveforms could be fully described or constructed by summing a series of sine waves of specific frequency and amplitude. This meant that square wave timing signals could be described accurately as a set of frequencies and associated amplitudes, or in the frequency domain. [Figure 12](#) illustrates how summing a few sine waves can approximate a square wave timing signal.

Harmonic Content

If the period of the square wave is assumed to be the fundamental frequency, then integer multiples of this fundamental are called the harmonics. A 100MHz square wave signal has a fundamental frequency of 100MHz, the third harmonic is 300MHz, the 4th is 400MHz, etc. To fully describe a signal, all that is needed is a list of its harmonic content. Mathematically it can be shown that

a perfect square wave-timing signal uses only odd harmonics in every decreasing amplitude. See [Figure 13](#), which is a plot of the frequency spectrum of a perfect square wave. If other harmonics are present, the signal is not a pure square wave.

Or, the timing signals contain jitter resulting in the presence of other harmonic frequencies, both harmonic and sub harmonics. A spectrum analyzer can be used to view the frequency spectrum and display the undesirable harmonic content of the timing signal. Since the harmonic content is the results of both amplitude (wave shape) and phase & period jitter, all real timing signals are rich in harmonic content. In general, we are most interested in the harmonic content resulting in phase and period jitter (frequency) only.

Frequency Spectrum of a Timing Signal

The phase noise specification is a specific form of frequency domains specifications. Phase noise is often considered a very important measurement of the purity of a timing signal, particularly in communications applications. Phase noise is another way of describing the variations (jitter) of the timing signal as compared to a perfect reference signal. [Figure 14](#) shows the *frequency spectrum* plot of a timing signal with jitter. Note that jitter creates a sideband of continuous frequency (power) content around the fundamental frequency and its harmonics. If these sidebands are equal, it indicates that the jitter is largely from phase or frequency jitter vs. signal amplitude jitter.

Spectral Density & RMS Jitter Relationship

If power is extracted from the noise only portion of the spectrum and plotted as a function of frequency, the result is called a *Spectral Density* plot. See [Figure 15](#). This plot is interesting since it shows the distribution of the noise power over the frequency spectrum and the area under the plot is equal to the RMS jitter squared, thus relating the time domain RMS jitter spec to the frequency domain spectral purity.

$$\text{RMS Jitter} = \sqrt{\text{Area under the Spectral Purity Plot}}$$

Spectral Purity & dBc Phase Noise

Returning to the frequency spectrum plot of [Figure 14](#), it is often of interest to define the Spectral Purity in a specific band of frequency. The spectral purity in a band at a specific offset from the fundamental compared to the power of the fundamental is often called the dBc Phase Noise specification at a specific offset from the fundamental. See [Figure 16](#).

To be more precise, phase noise is specified as the ratio of the noise power present in a 1Hz band at a specified offset from the fundamental frequency to the power value of the fundamental or:

$$\text{dBc Phase Noise} = \frac{\text{(Power Level of the 1Hz band at offset X)}}{\text{(Power Level of the fundamental frequency)}}$$

This ratio is often expressed in decibels (dBm), or a ratio of the power in the 1Hz band to the power in the fundamental. When the specific offset is specified, the phase noise is called a dBc value, which simply means dBm at a specified offset from the fundamental. See [Figure 16](#).

Phase Noise Measurement Issues

As with most timing specifications, measurements of phase noise have many issues. The primary issue relates to the capabilities of the equipment. Often the noise floor of the equipment is higher than the level of noise at large offsets. One way to determine the noise floor is to look at the flat part of the plot at very large offsets—this is typically the noise floor. Also with very small offsets, the quality of the local reference used in the equipment makes accurate measurements difficult. Again, the shape of the plot near the fundamental (small offsets) with a perfect source (equipment reference) will not go closer to that certain distance; this is the limitation of detecting close in jitter. See [Figure 18](#).

Conclusions

In regards to frequency domain specifications, the descriptions here are greatly simplified. In actual measurements, effects of phase, frequency and amplitude jitter need to be considered. However, if the signal is digital with amplitude limiting, much of the complexity is removed. Often different groups with different equipment and measurement techniques will come up with widely varied results. If specifications between devices are to be compared, first verify the compatibility of the equipment's capabilities and measurement techniques. One of the most common problems when measuring phase noise relates to the capability of the equipment used to do the measurements. Often spectrum analyzers are used which may result in poor results due to the limitations of the equipment.

By investigating jitter in the frequency domain, we get a better understanding of its effects on the desired application over the entire time record of the signal. It is mathematically possible to calculate an expected bit error rate given a phase noise plot.

Since we have made assumptions concerning the statistical nature of the noise, we cannot specify absolute limits, but only probabilities of jitter in a specific frequency range. As noted earlier, many applications such as communications can tolerate jitter of some frequencies and be very intolerant of others. Frequency domains specifications ease the understanding of the nature of the signals characteristics in these types of applications.

Figure 1. Perfect Timing Signal

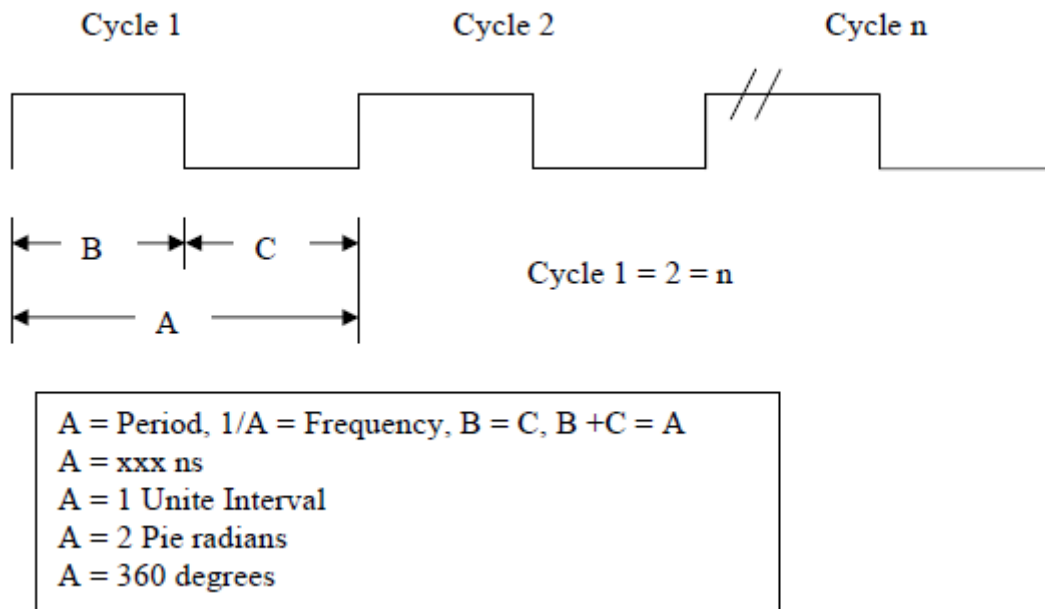
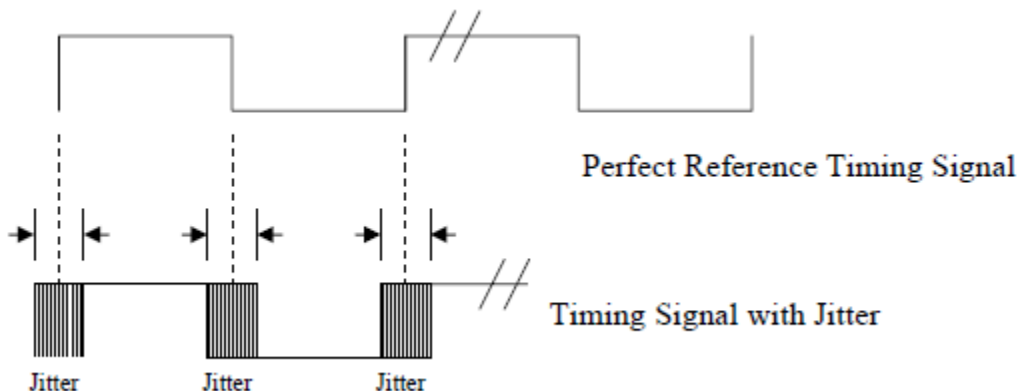


Figure 2. Timing Signal with Jitter



Note: Jitter around each edge can be different and may not be equally distributed around an edge. Also, jitter in the next periods can be different.

- **Period Jitter** is the maximum jitter observed at the end of a period when compared to the position of the perfect reference clocks edge. The number of cycle used to look for the maximum jitter varies by application but the JEDEC spec is 10,000 observed cycles.
- **Cycle-to-Cycle Jitter** is the maximum observed variation between two adjacent cycles periods over a defined number of observed cycles. The number of cycle observed is application dependent, but the JEDEC specification is 1000 cycles.
- **Absolute Period Jitter** is the maximum jitter observed for the test signal when compared to the perfect references edges in the application environment. This is sometimes referred to as the 6-sigma jitter value.

Figure 3. Timing Jitter Viewed With An Oscilloscope

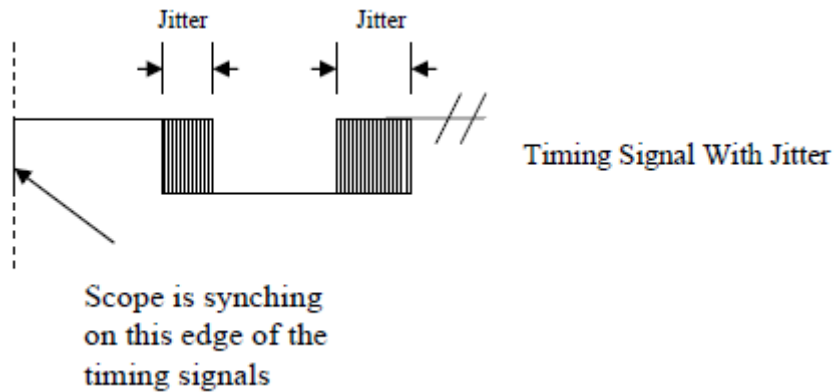
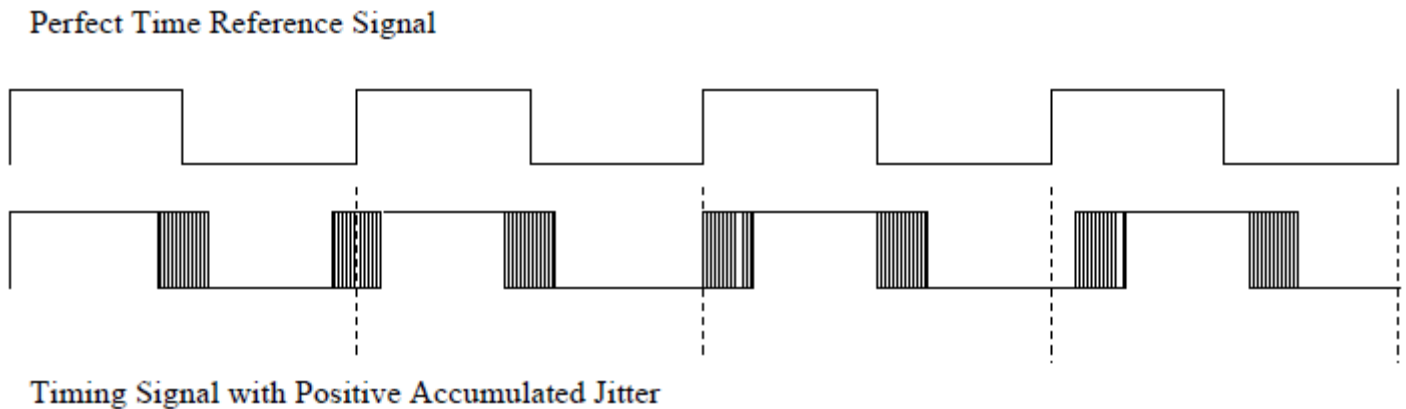
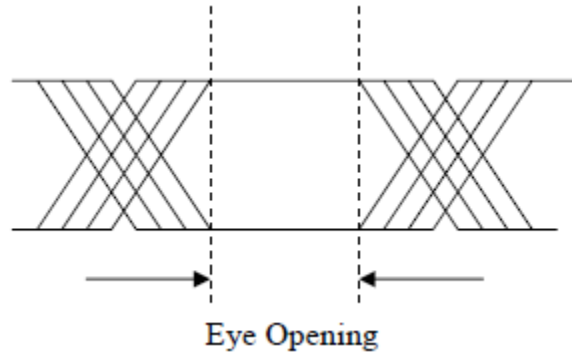


Figure 4. Timing Signal with Accumulated Jitter



In this example the period continues to get larger due to accumulated jitter. The reverse is also possible where the period grows shorter. Spread Spectrum Modulation is an example of intentional accumulated jitter generation.

Figure 5. Eye Diagrams



Jitter in the signal closes the “eye” and makes data and clock recovery more difficult.

Figure 6. Data Transmission System with Repeaters

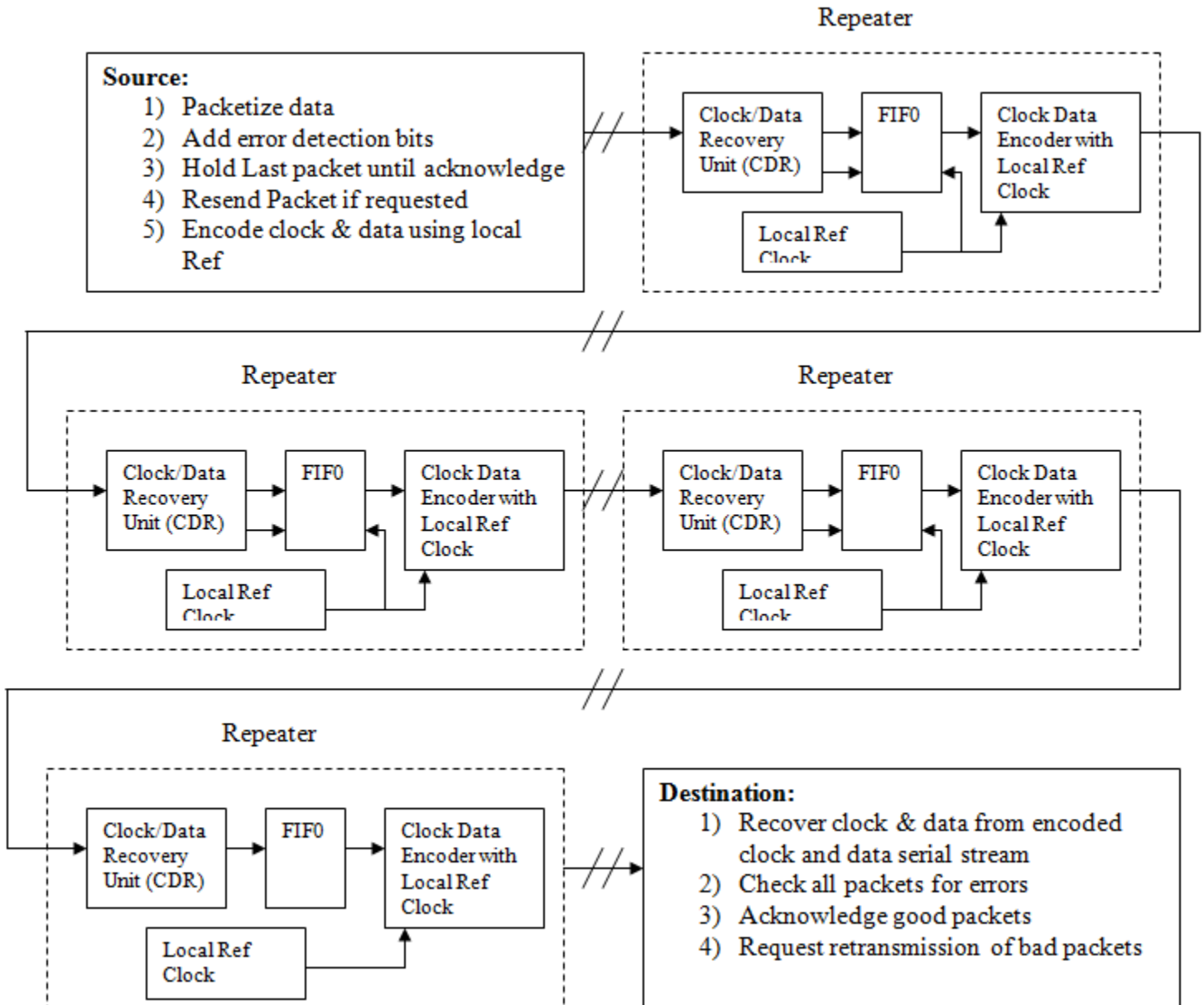


Figure 7. Duty Cycle And 1/2 Period Jitter

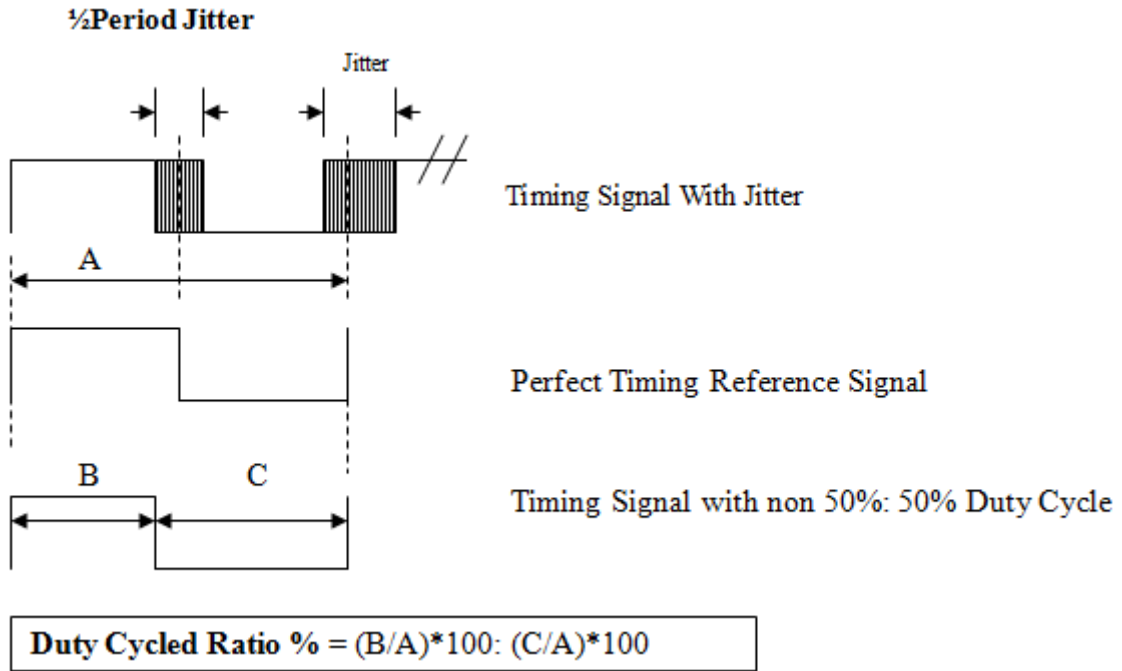


Figure 8. Time Interval Error Jitter Plot & Peak-to-Peak Jitter

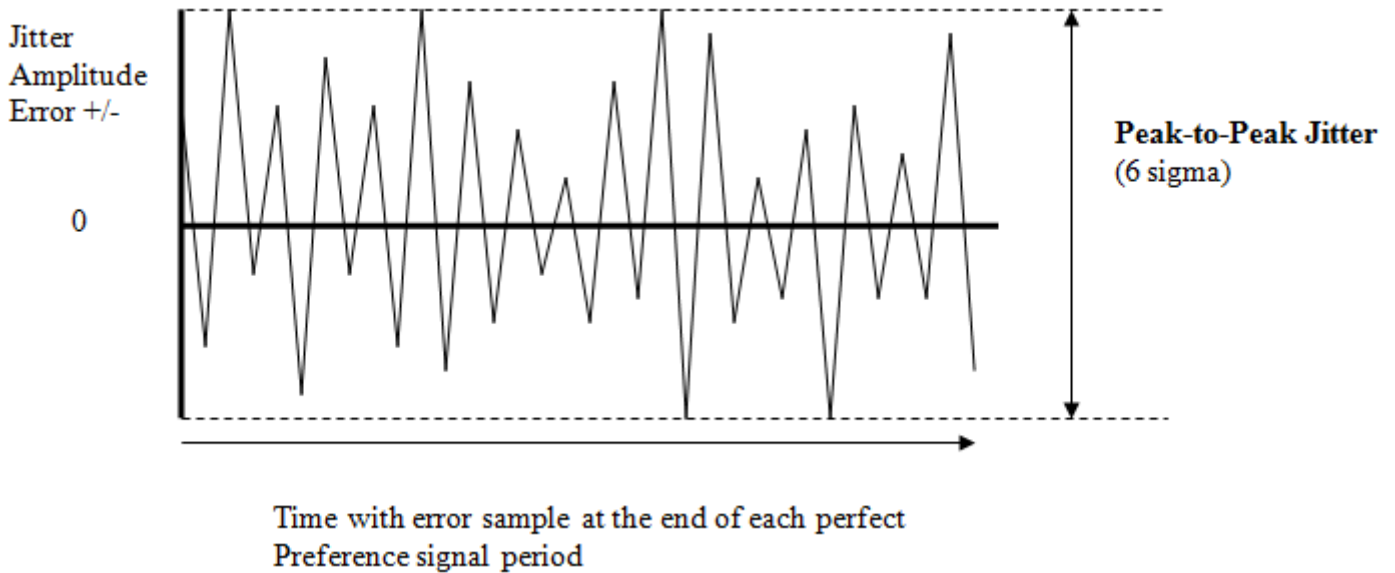


Figure 9. Sine Wave & RMS Level

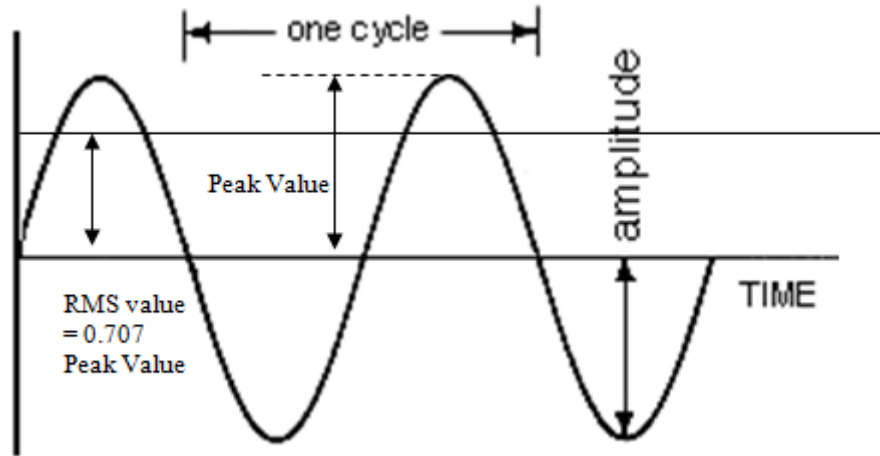


Figure 10. Gaussian Jitter Distribution

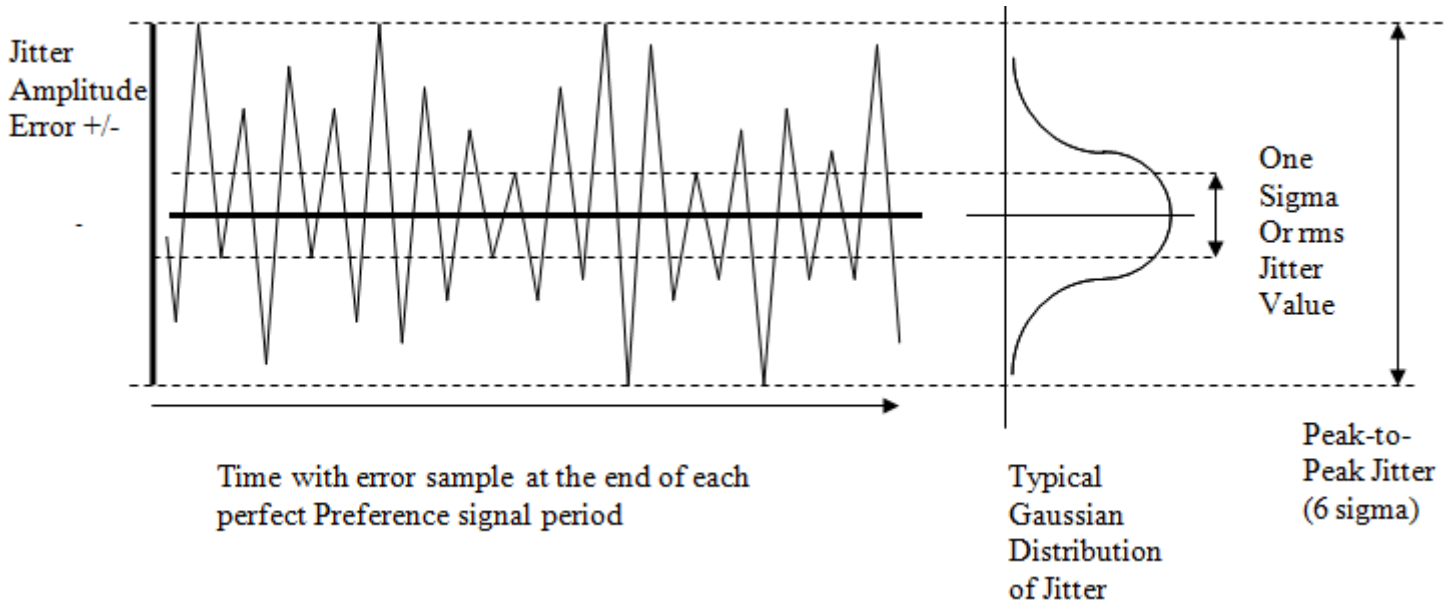


Figure 11. Sine Wave Generation

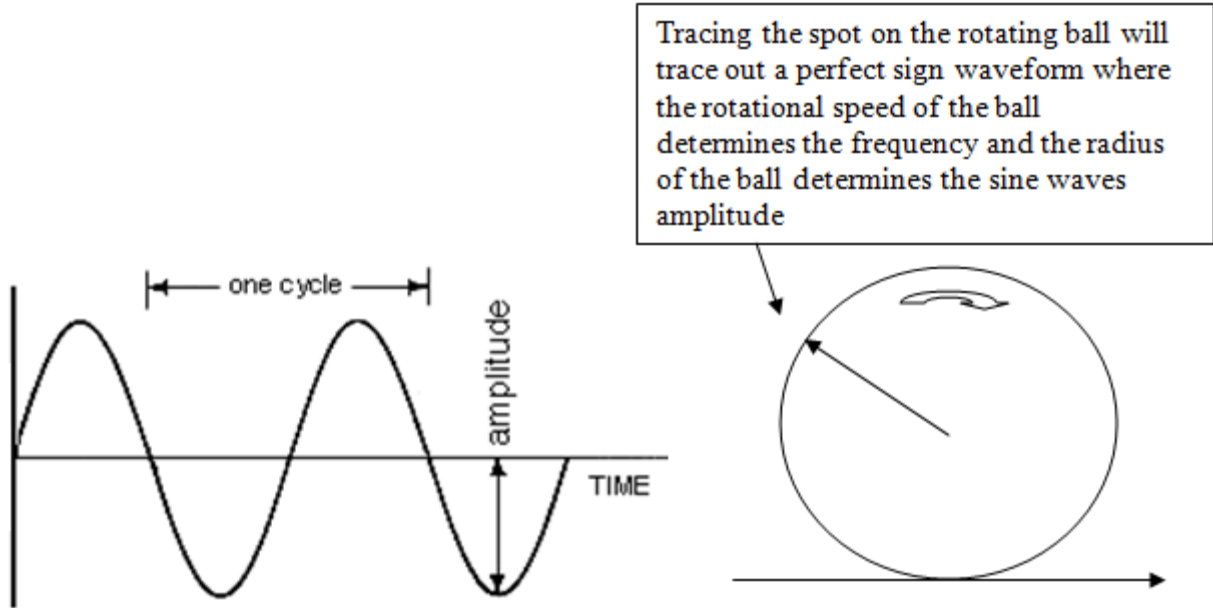


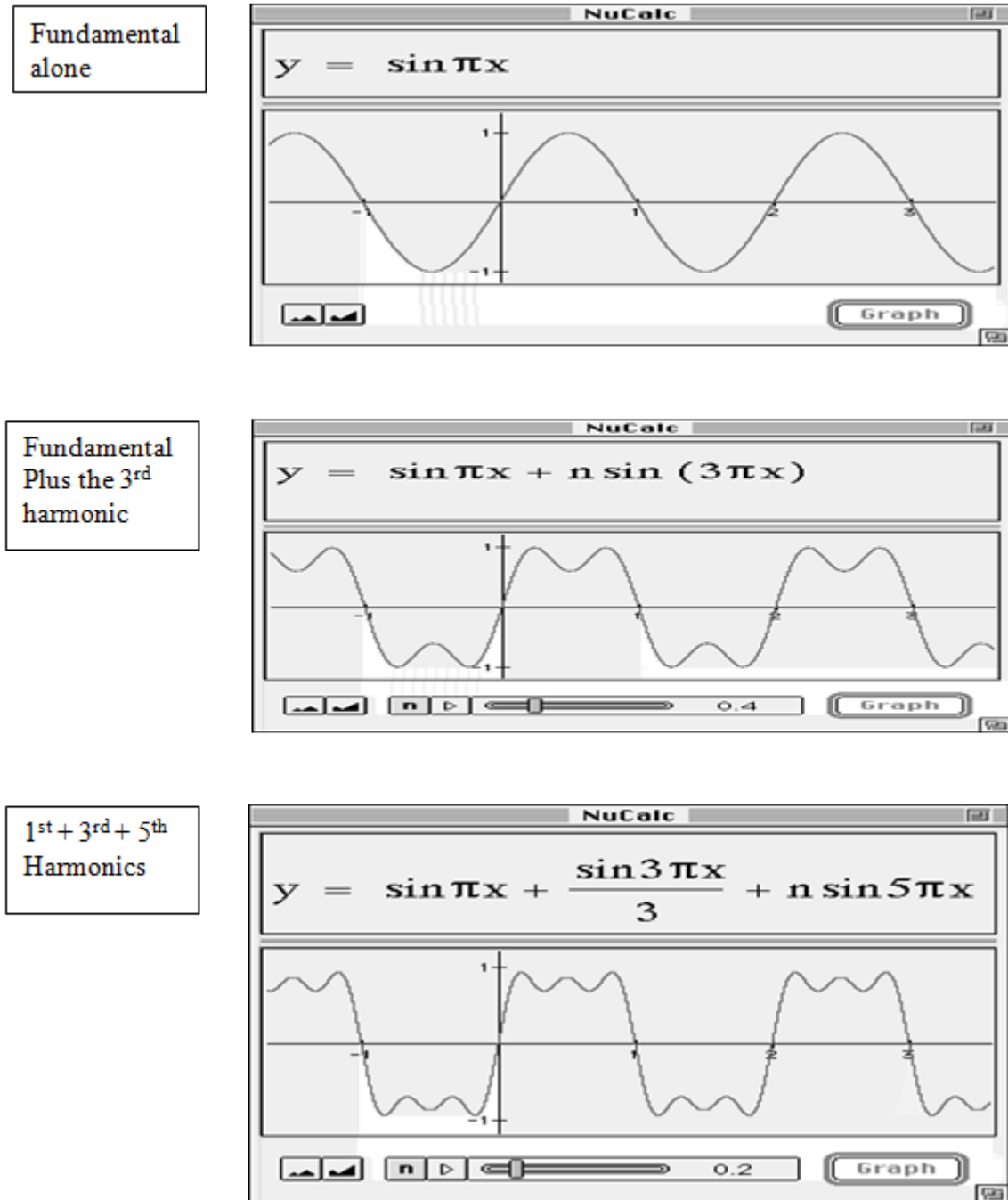
Figure 12. Fourier Series Of A Square Wave


Figure 13. Spectral Content of a Perfect Square Wave

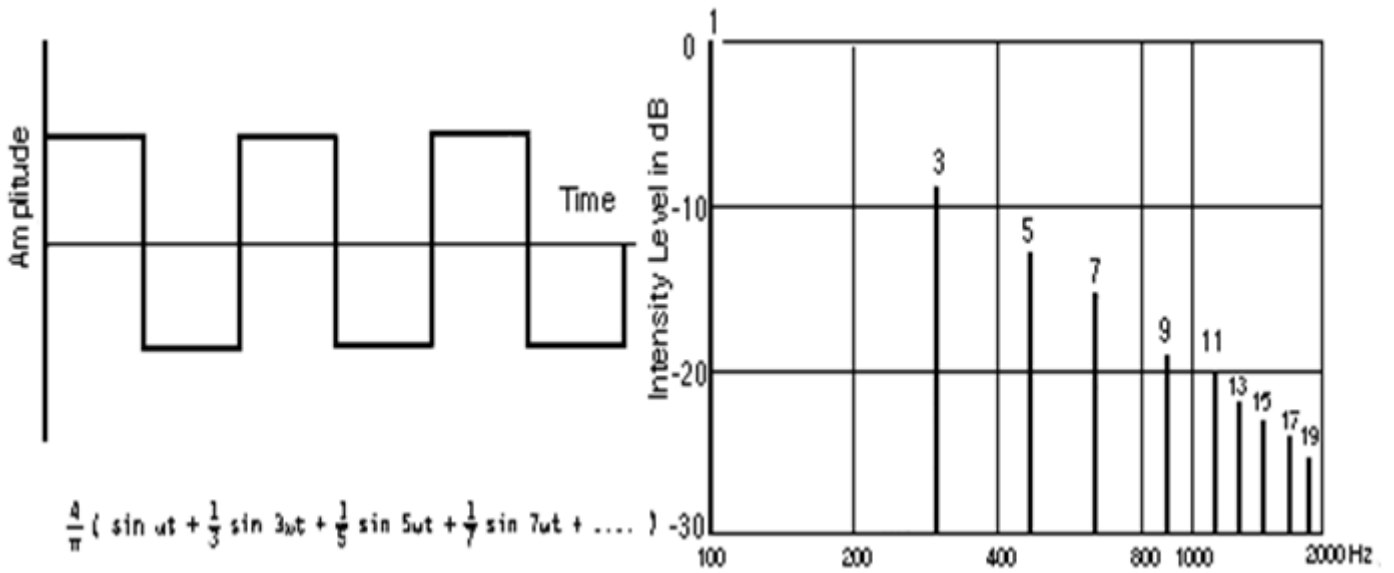


Figure 14. Spectral Content of a Realistic Timing Signal

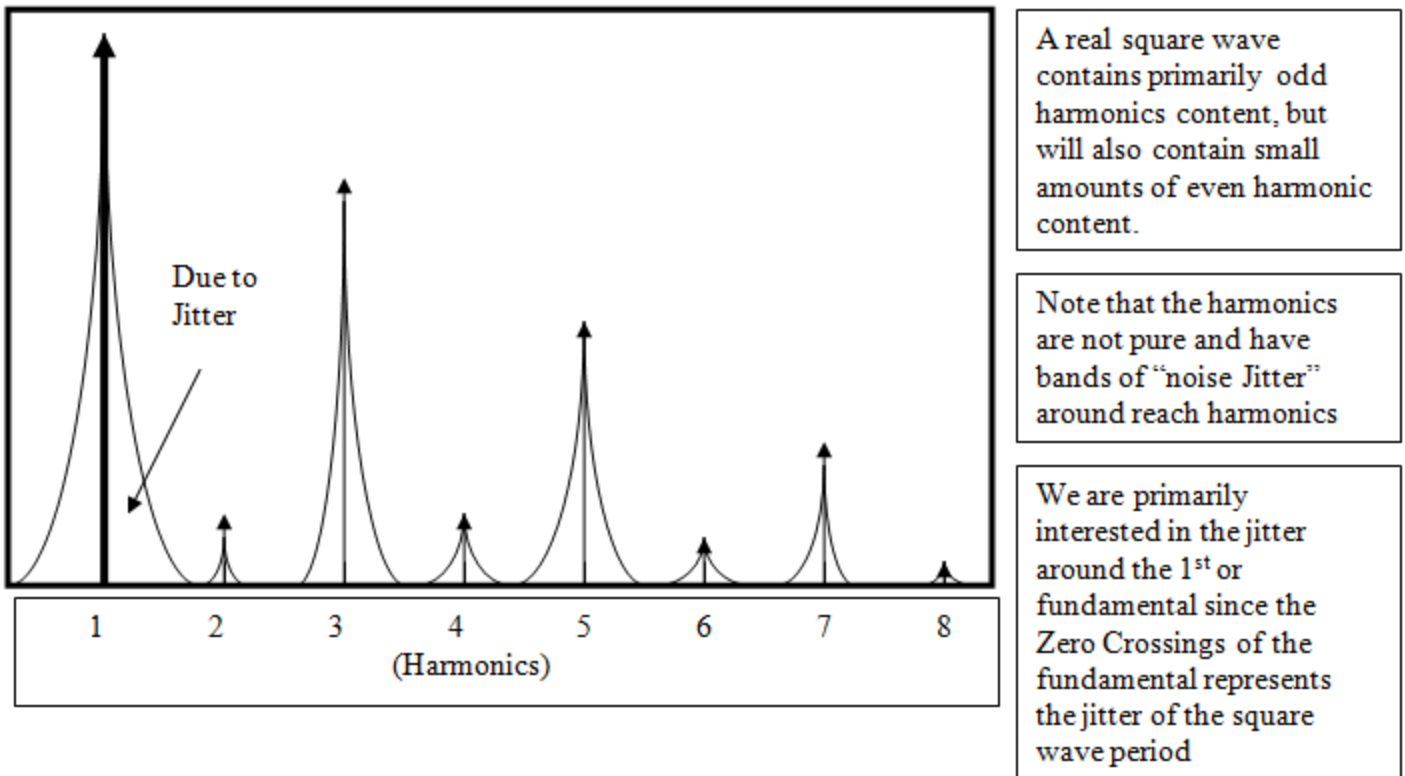
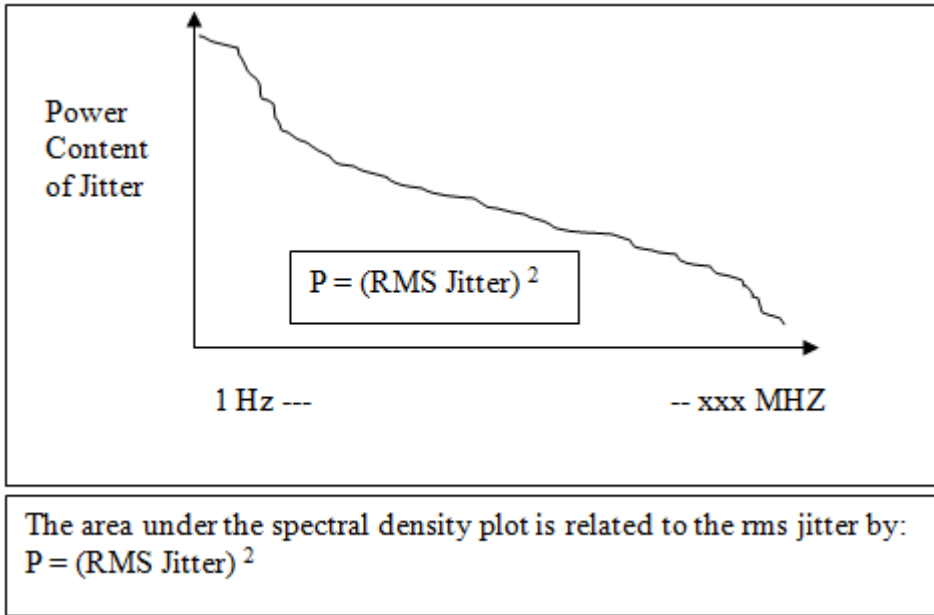


Figure 15. Spectral Density Plot & RMS Jitter



Note that **Spectral Density** is the spectral Power content of the NOISE or Jitter by itself. In other words if the jitter is the results of a modulation waveform, the Spectral Density plot is the Power spectrum of this modulation waveform, not the spectrum of the timing signal.

Figure 16. Spectral Purity, dBc Phase Noise Values from Frequency Spectrum Plot

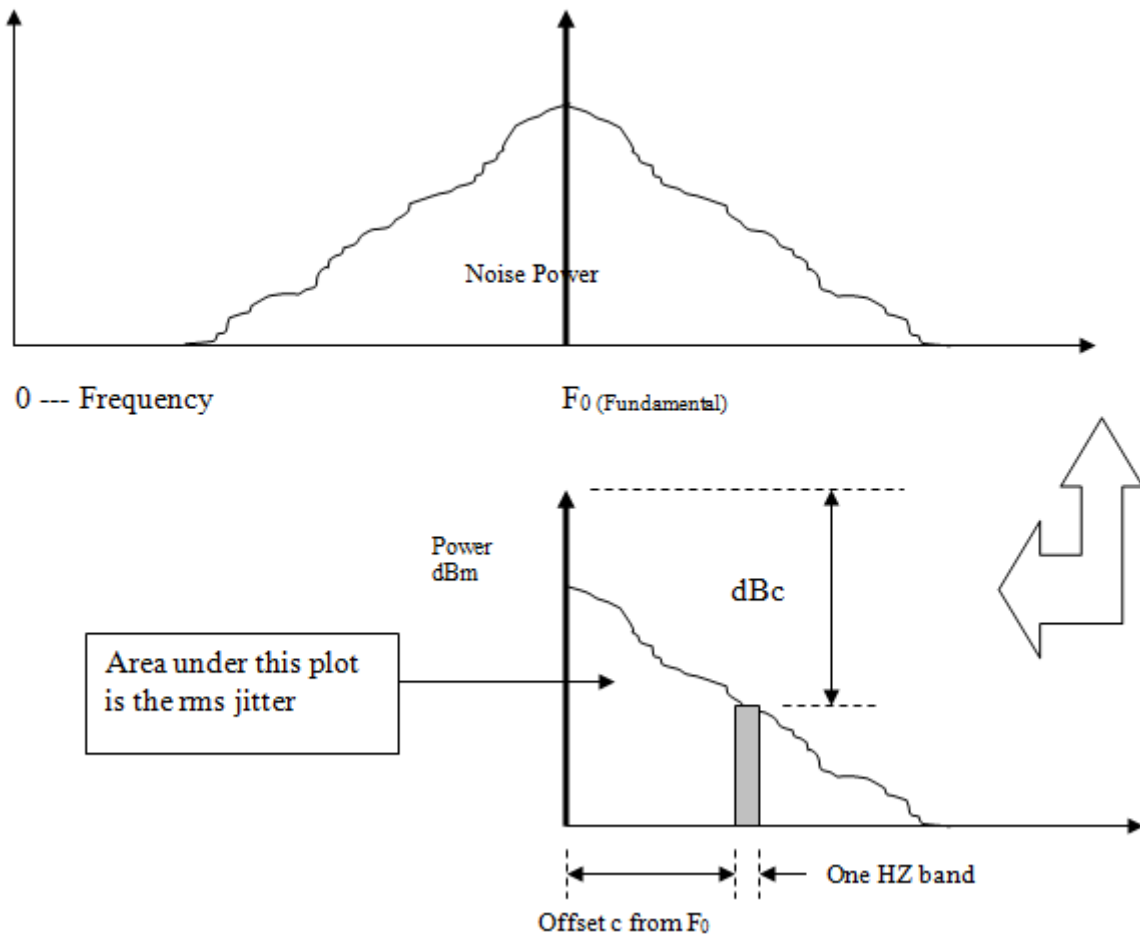


Figure 17. dBc Phase Noise Plot with Mask

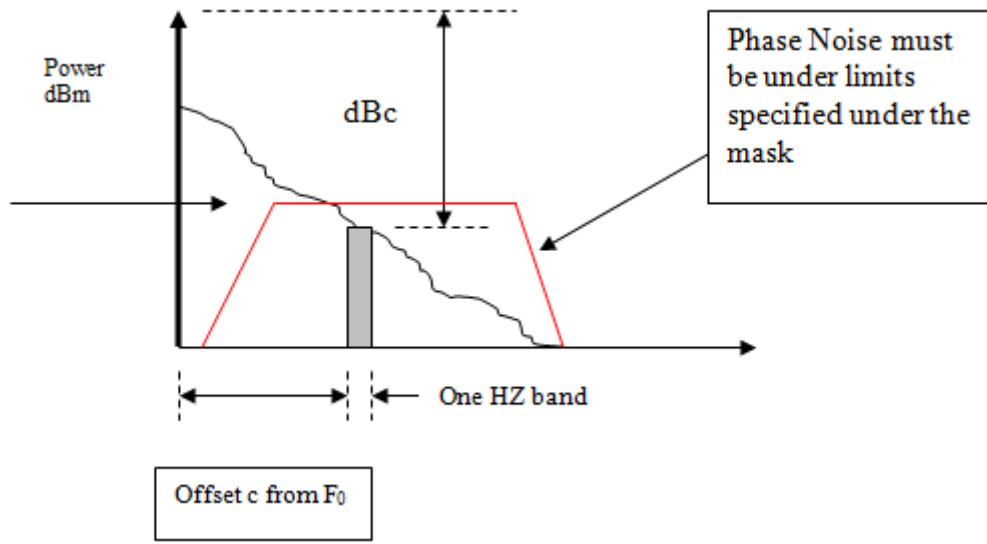
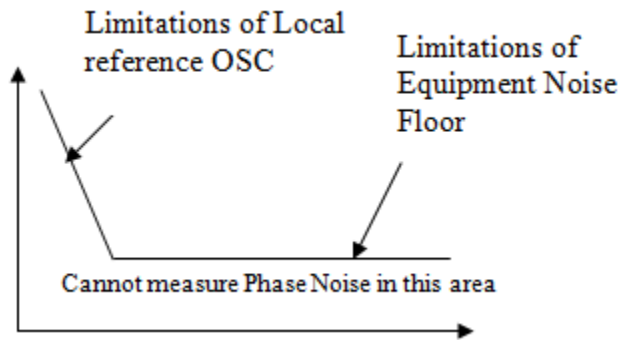


Figure 18. Phase Noise Measurement Limitations





Corporate Headquarters
6024 Silver Creek Valley Road
San Jose, CA 95138 USA

Sales
1-800-345-7015 or 408-284-8200
Fax: 408-284-2775
www.IDT.com

Tech Support
email: clocks@idt.com

DISCLAIMER Integrated Device Technology, Inc. (IDT) and its subsidiaries reserve the right to modify the products and/or specifications described herein at any time and at IDT's sole discretion. All information in this document, including descriptions of product features and performance, is subject to change without notice. Performance specifications and the operating parameters of the described products are determined in the independent state and are not guaranteed to perform the same way when installed in customer products. The information contained herein is provided without representation or warranty of any kind, whether express or implied, including, but not limited to, the suitability of IDT's products for any particular purpose, an implied warranty of merchantability, or non-infringement of the intellectual property rights of others. This document is presented only as a guide and does not convey any license under intellectual property rights of IDT or any third parties.

IDT's products are not intended for use in applications involving extreme environmental conditions or in life support systems or similar devices where the failure or malfunction of an IDT product can be reasonably expected to significantly affect the health or safety of users. Anyone using an IDT product in such a manner does so at their own risk, absent an express, written agreement by IDT.

Integrated Device Technology, IDT and the IDT logo are registered trademarks of IDT. Product specification subject to change without notice. Other trademarks and service marks used herein, including protected names, logos and designs, are the property of IDT or their respective third party owners.

Copyright ©2014 Integrated Device Technology, Inc.. All rights reserved.