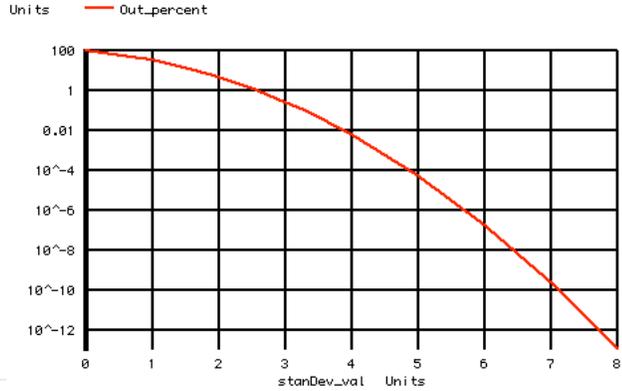
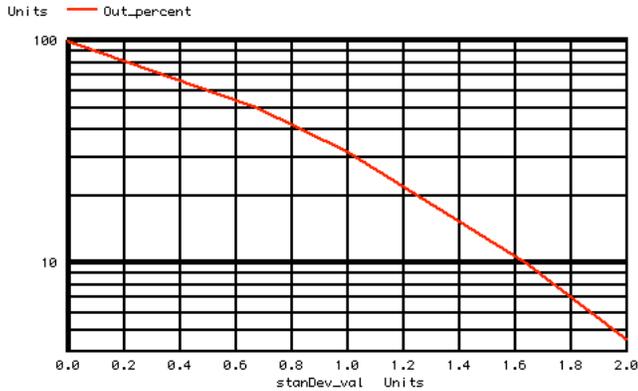
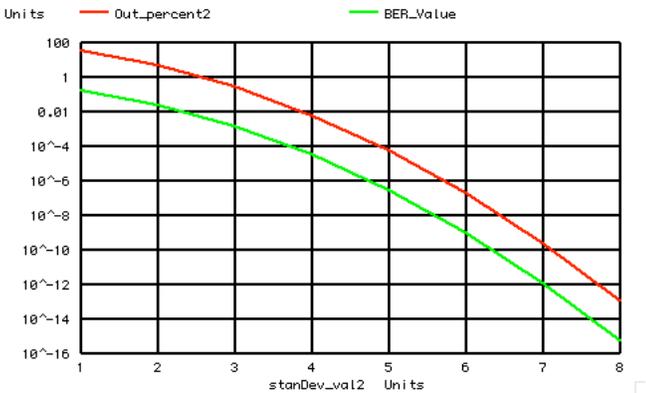


=====**What is needed for a given BER**=====

This spice example just plots standard deviation and BER curves.



At +/- 1 standard deviation, (100-31.73)= 68.27% is within that range.
 At +/- 3 standard deviations, (100-0.27) = 99.73% is within the +/- standard deviation range.



At +/- 7 standard deviation, 2.5600000e-10% is outside that range.
 The BER (Bit Error Rate) should as least be below 2.56e-12/1.414 since a square wave has 1.414 more rms energy.

The BER curve above comes from a Maxim article enclosed below.

```
=====Add Standard Deviation Data=====
plot      OutsideData% vs standard_deviation
=====Add BER Data=====
plot      OutsideData% and BER vs standard_deviation
=====done=====
```

=====**MacSpiceCode**=====

```
Plot BER vs SD
Plot_BER_vs_SD
*=====Create Signal No Reason=====
*V SIN#  NODE_P  NODE_N  DC    VALUE  SIN(  V_DC  AC_MAG  FREQ  DELAY  FDamp)
VIN      VP      0      DC    0      SIN(  0     1     1     )

.control
set      pensize = 2
unlet  stanDev_val
unlet  Out_percent
let      stanDev_val      = vector(14)
let      Out_percent      = vector(14)

echo   "=====Add Standard Deviation Data====="
let     stanDev_val[0]    = 0
let     Out_percent[0]   = 100
let     stanDev_val[1]   = .674
let     Out_percent[1]   = 50
let     stanDev_val[2]   = 1
let     Out_percent[2]   = 31.73
let     stanDev_val[3]   = 1
let     Out_percent[3]   = 31.73
let     stanDev_val[4]   = 1.645
let     Out_percent[4]   = 10
let     stanDev_val[5]   = 2
let     Out_percent[5]   = 4.5500264
let     stanDev_val[6]   = 2.576
let     Out_percent[6]   = 1
let     stanDev_val[7]   = 3
```

```

let      Out_percent[7] = 0.2699796
let      stanDev_val[8] = 3.2906
let      Out_percent[8] = 0.1
let      stanDev_val[9] = 4
let      Out_percent[9] = 0.006334
let      stanDev_val[10] = 5
let      Out_percent[10] = 0.0000573303
let      stanDev_val[11] = 6
let      Out_percent[11] = 0.0000001973
let      stanDev_val[12] = 7
let      Out_percent[12] = 0.0000000002560
let      stanDev_val[13] = 8
let      Out_percent[13] = 1.246e-13
plot     Out_percent vs stanDev_val ylog
plot     Out_percent vs stanDev_val ylog xlimit 0 2
echo     "plot      OutsideData% vs standard_deviation"
echo     "=====Add_BER_Data=====
unlet    stanDev_val2
unlet    Out_percent2
unlet    BER_Value
let      stanDev_val2 = vector(8)
let      Out_percent2 = vector(8)
let      BER_Value = vector(8)

let      stanDev_val2[0] = 1
let      Out_percent2[0] = 31.73
let      BER_Value[0] = 0.16

let      stanDev_val2[1] = 2
let      Out_percent2[1] = 4.5500264
let      BER_Value[1] = 22.8m

let      stanDev_val2[2] = 3
let      Out_percent2[2] = 0.2699796
let      BER_Value[2] = 1.35m

let      stanDev_val2[3] = 4
let      Out_percent2[3] = 0.006334
let      BER_Value[3] = 32u

let      stanDev_val2[4] = 5
let      Out_percent2[4] = 0.0000573303
let      BER_Value[4] = 287n

let      stanDev_val2[5] = 6
let      Out_percent2[5] = 0.0000001973
let      BER_Value[5] = .98n

let      stanDev_val2[6] = 7
let      Out_percent2[6] = 0.0000000002560
let      BER_Value[6] = 1.28p

let      stanDev_val2[7] = 8
let      Out_percent2[7] = 1.246e-13
let      BER_Value[7] = .63f

plot     Out_percent2 BER_Value vs stanDev_val2 ylog
echo     "plot      OutsideData% and BER vs standard_deviation"
echo     "=====done=====
.endc
.end

```

4.4.11_11.07AM
dsauersanjose@aol.com
Don Sauer



Maxim > Design support > App notes > High-Speed Interconnect > APP 1916

Mar 06, 2003

Keywords: jitter, gaussian, random, RJ, deterministic, cycle to cycle, adjacent cycle, duty cycle distortion, pulse width distortion, inter-symbol, bit error rate, BER, ber

Download, PDF format (72kB)

[Related parts](#)

SHARE [?]

APPLICATION NOTE 1916

An Introduction to Jitter in Communications Systems

Abstract: This introduction to jitter presents definitions for various jitter types including the random jitter types: Gaussian, cycle-to-cycle, adjacent cycle; and deterministic jitter types: duty cycle distortion, pulse width distortion, pulse skew and data dependent (pattern) jitter. The application note also discusses the relationship between the various jitter components and system Bit Error Rate (BER).

What is Jitter?

The SONET standard states that "Jitter is defined as the short-term variations of a digital signal's significant instants from their ideal positions in time. Significant instants could be (for example) the optimum sampling instants." The Fiber Channel standard simply defines jitter as "The deviation from the ideal timing of an event."

In short, the term "jitter" describes timing errors within a system. In a communications system, the accumulation of jitter will eventually lead to data errors.

The parameter that is of most value to the system user is the frequency of occurrence of these data errors, normally referred to as the Bit Error Rate (BER). We'll discuss BER in more detail later.

First, some definitions. The basic jitter types and definitions are listed in **Table 1** below. Some jitter types have a number of commonly used terms describing the same measurement. Others have terms describing different measurement methods for the same jitter type. Where multiple terms are used to describe the same jitter type, these are listed together.

Table 1. Terms and Definitions Associated with Jitter Measurements

Jitter Term	Definition	Additional Information
Jitter	In addition to the definitions above, jitter is composed of two basic types: random and deterministic.	
Random Jitter (RJ)	Jitter that is not bounded and can be described by a Gaussian probability distribution. Random jitter is characterized by its standard deviation (rms) value.	The principal source is Gaussian (white) electrical noise within system components. Electrical noise interacts with the slew rate of signals to produce timing errors at the switching points.
Random	RJ measurement method. A probability distribution based on the difference in time between an actual clock edge and its ideal (intended) position.	Although two measurements of the same source, random and cycle to cycle jitter are not equivalent. Cycle to cycle jitter has frequency dependant terms and, compared to random jitter measurements, will accentuate high frequency jitter sources while rejecting low frequency sources. The random jitter measurement is independent of frequency.
Cycle to Cycle Adjacent Cycle	RJ measurement method. A probability distribution based on the difference in the period measured between one clock cycle and an adjacent cycle.	
Deterministic Jitter (DJ)	Jitter with a non-Gaussian probability density function. Always bounded in amplitude and with specific causes. DJ is characterized by its bounded, peak-to-peak, value.	Sources are generally related to imperfections in the behavior of a device or transmission media but also may also be due to EMI, crosstalk, grounding problems.
Duty Cycle Distortion Pulse Width Distortion Pulse Skew	DJ component. Deviation in duty cycle value from the ideal (intended) value. In many serial data systems this equates to a deviation in bit time between a 1 bit and a 0 bit. May also defined as the difference in propagation delay between low to high and high to low delay times.	Source is commonly timing differences between rising and falling edges within a system. May also be caused by ground shifts in single ended systems.
Data Dependant Jitter	DJ component. Timing errors that	Primary source is component and

Pattern Jitter Inter-Symbol Interference	vary with the data pattern used. Data dependant and Pattern jitter are used to describe the effect of jitter in the time domain. Inter-Symbol Interference is more commonly applied to frequency domain measurements i.e., the spreading of a signal peak as would be seen on a spectrum analyzer.	system bandwidth limitations. Higher frequency signals have less time to settle than lower frequency ones. This leads to changes in the start conditions for transitions at different frequencies and produces timing errors dependent on the data pattern being applied.
Sinusoidal Jitter Periodic Jitter	DJ component. Jitter that has a sinusoidal (or periodic) form and is related to (correlates to) the data pattern.	Source is interference from signals that are related to the data pattern. Ground bounce and other power supply variations are common causes although the levels of sinusoidal jitter normally encountered are very low.
Uncorrelated bounded Jitter	DJ component. Jitter that is bounded in amplitude and uncorrelated (to the data pattern).	Commonly sinusoidal in nature, source is interference from other signal sources either within the system or external to it. Sources include EMI, capacitive and inductive coupling and power supply switching noise.
Total Jitter (TJ)	The summation (or convolution) of deterministic and random jitter. Total jitter is the peak to peak value obtained.	$TJ = DJ + n \times RJ$ where n = number of standard deviations corresponding to the required BER. This summation is usually applied due to it's simplicity although this method over-estimates the actual BER since the maximum RJ errors will not always coincide with maximum DJ error. A probabilistic (convolved) summation of the two jitter types would produce a more accurate solution though the application of this would require knowledge of the DJ modulation waveform.
Mapping Jitter	System level jitter component of DJ type. Jitter due to mapping of data from one transmission standard to another when bit stuffing has occurred during the mapping process.	Gaps are left in the recovered signal after de-mapping. Phase locked loops (PLLs) are used to smooth the resulting gaps but a certain amount of jitter remains.
Pointer Jitter	System level jitter component of DJ type. Jitter resulting from the application of a SONET signal containing defined sequences of pointer activity to a demultiplexer.	
Wander Jitter	System level jitter component of DJ type. Low frequency timing errors less than 10Hz in frequency (SONET).	Principal source is system temperature variations.
Jitter Transfer Jitter Gain	Ratio of jitter on output signal to jitter applied on input signal.	Used to quantify the jitter accumulation performance of data retiming devices: regenerators, PLLs.
Jitter Tolerance	Amount of input jitter a receiver must tolerate without violating system BER specifications.	Can be split into Random Jitter Tolerance and Deterministic Jitter Tolerance.
Unit Interval (UI)	Time period equivalent to 1 bit time in a serial data stream.	Reciprocal of baud rate. Jitter specifications are often quoted in multiples of UI.

How Does Jitter Lead to Data Errors?

Information is extracted from serial data streams by sampling the data signal at specific instants. Ideally these sampling instants would always occur at the centre of a data bit time, equidistant between two adjacent edge transition points. The presence of jitter changes the edge positions with respect to the sampling point. An error will then occur when a data edge falls on the wrong side of a sampling instant.

As has been stated in Table 1, the total jitter can be expressed as the sum of deterministic jitter and a number of standard deviations of random jitter at any particular error probability value. Random jitter is defined above as jitter which can be described by a Gaussian probability distribution. Gaussian distributions are symmetrical about a mean value. One standard deviation (1σ) is defined as the

window which contains 68.26% of a population to one side of the mean. **Table 2** lists multiples of σ with the proportion of total population applicable to each.

Table 2. Proportion of Total Population vs. Standard Deviation in a Gaussian Distribution

Limit	Proportion of Population Within Limits
$\pm 1\sigma$	68.2689%
$\pm 2\sigma$	95.45%
$\pm 3\sigma$	99.73%
$\pm 4\sigma$	99.99367%
$\pm 5\sigma$	99.9999427%
$\pm 6\sigma$	$100 - 1.973 \times 10^{-7}\%$
$\pm 7\sigma$	$100 - 2.5596 \times 10^{-10}\%$
$\pm 8\sigma$	$100 - 1.24419 \times 10^{-13}\%$
$\pm 9\sigma$	$100 - 2.25718 \times 10^{-17}\%$
$\pm 10\sigma$	$100 - 1.53398 \times 10^{-21}\%$

The result of the summation of deterministic and random jitter is another probability distribution, an example of which is shown in **Figure 1**. The distribution plots probability against timing error magnitude and is characterized by a having a centre portion, which represents the deterministic jitter content, and outer portions which are the tails of the (random jitter) Gaussian distribution. The shape of distribution shown is referred to as a bimodal response.

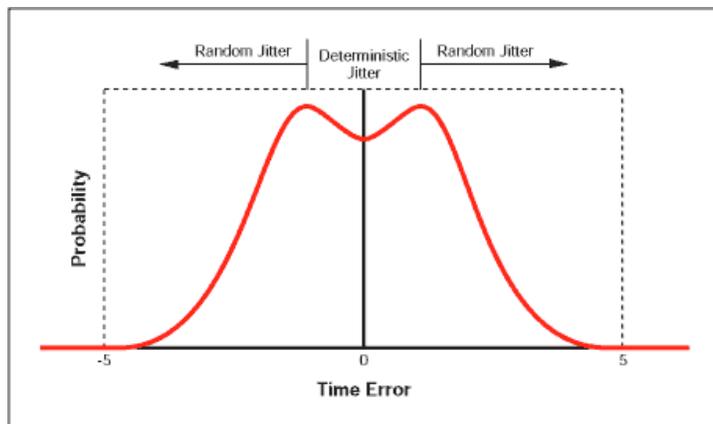


Figure 1. Probability histogram showing deterministic and random components.

Adding the jitter probability distribution of Figure 1 to a data stream effectively modulates the data edge positions with respect to the sampling instant. This is illustrated in **Figure 2**, which shows an ideal eye diagram with probability histograms superimposed on the data transition points. The probability of a data error associated with the sampling instant is the sum of the probabilities that either the first data transition will arrive too late or the second data transition will arrive too early. This probability is denoted by the shaded portion under the curves at the sampling point in Figure 2.

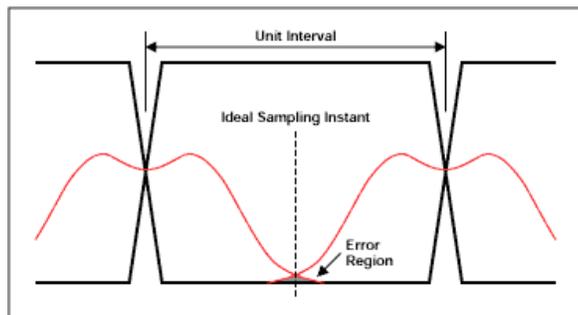


Figure 2. Ideal eye diagram with data transition time probability histograms.

To find the probability of a data error occurring, the sum of the probabilities of either data edge being in error must be multiplied by the probability of a transition actually occurring. The latter is represented by the average transition density and assumed to be equal to 50% for a typical data stream.

By way of an example, consider a data stream with 0.3UIp-p total deterministic jitter (which includes all non-Gaussian timing error sources) and 0.05UI rms random jitter. The maximum allowable jitter is 1UIp-p; this is the amount of jitter an ideal receiver would tolerate before an error occurs (see note below). Using the expression of equation 1

$$DJ(pk) + n \times RJ(rms) = TJ(pk)$$

and substituting $TJ = 0.5UI(pk)$, $DJ = 0.15UI(pk)$ and $RJ = 0.05UI(rms)$, we obtain $n = 7$. This is the number of standard deviations (σ) of random jitter that will produce a data error. For a Gaussian distribution, $1.28 \times 10^{-10}\%$ of samples lie outside a 7σ limit to one side of the mean. The total error rate (BER) is then given by equation 2.

$$BER = (1.28 \times 10^{-10}\% + 1.28 \times 10^{-10}\%) \times 50\% = 1.28 \times 10^{-10}\%$$

The result of equation 2 corresponds to a BER of 1.28×10^{-12} .

The bit error rates corresponding to random jitter limits from $\pm 1\sigma$ to $\pm 10\sigma$ are tabulated below in Table 3.

Table 3. BER as a Function of Number of Standard Deviations

Limit	BER
$\pm 1\sigma$	0.16
$\pm 2\sigma$	2.28×10^{-2}
$\pm 3\sigma$	1.35×10^{-3}
$\pm 4\sigma$	0.32×10^{-4}
$\pm 5\sigma$	2.87×10^{-7}
$\pm 6\sigma$	0.98×10^{-9}
$\pm 7\sigma$	1.28×10^{-12}
$\pm 8\sigma$	0.62×10^{-15}
$\pm 9\sigma$	1.13×10^{-19}
$\pm 10\sigma$	0.77×10^{-23}

Note: The maximum jitter allowable at a particular BER value is normally provided by the system specifications or by the communications standard which the system is required to be compatible with. The maximum allowable jitter is normally specified at a level lower than 1UI.

Related parts

MAX9310	1:5 Clock Driver with Selectable LVPECL Inputs and LVDS Outputs	Free samples
MAX9311	1:10 Differential LVPECL/LVECL/HSTL Clock and Data Drivers	Free samples
MAX9315	1:5 Differential LVPECL/LVECL/HSTL Clock and Data Driver	Free samples
MAX9316	1:5 Differential LVPECL/LVECL/HSTL Clock and Data Driver	Free samples
MAX9317	Dual 1:5 Differential Clock Drivers with LVPECL Inputs and LVDS Outputs	
MAX9320	1:2 Differential LVPECL/LVECL/HSTL Clock and Data Drivers	Free samples
MAX9322	LVECL/LVPECL 1:15 Differential Divide-by-1/Divide-by-2 Clock Driver	
MAX9360	LVTTTL/TTL/CMOS-to-Differential LVECL/ECL Translators	
MAX9374	Differential LVPECL-to-LVDS Translators	Free samples
MAX9380	Single-Ended-to-Differential LVECL/LVPECL 2:1 Multiplexer	
MAX9384	ECL/PECL Dual Differential 2:1 Multiplexer	Free samples
MAX9386	Differential 5:1 or 4:1 ECL/PECL Multiplexers with Single/Dual Output Buffers	Free samples
MAX9389	Differential 8:1 ECL/PECL Multiplexer with Dual Output Buffers	Free samples
MAX9400	Quad Differential LVECL/LVPECL Buffer/Receivers	

Automatic updates

Would you like to be automatically notified when new application notes are published in your areas of interest? [Sign up for EE-Mail™](#).

Your comments

[Login](#) or [register](#) to post a comment.

 [Download, PDF format \(72kB\)](#)

APP 1916: Mar 06, 2003

AN1916, AN 1916, APP1916, Appnote1916, Appnote 1916

[Contact Us](#) | [Rate This Page](#) | [Mail This Page](#) | [Privacy Policy](#) | [Legal Notices](#)

Copyright © 2011 by Maxim Integrated Products