Measurement-based simulation: increasing IBIS-AMI model accuracy with data from lab measurements

Michael Steinberger, Ph.D., SiSoft

Paul Wildes, SiSoft

Anders Ekholm, Ericsson

Nicke Svee, Ericsson
Abstract

The combination of a spectrum analyzer and specific data patterns can be used to reliably identify different jitter and noise sources in SerDes transmitters and receivers, quantifying the associated impairments precisely. This paper uses data measured on a real system to explain the procedures for measuring clock based DCD, data based DCD, reference clock phase noise, clock leakage, power supply noise, transmission loss and crosstalk in the serial channels at the system level. Once the different impairments are quantified, jitter and noise budgets are extracted than can be used by standard serial link simulation tools. This allows data extracted from silicon characterization to be used to accurately predict operating margins during new system design. This paper will demonstrate the derivation of device jitter and noise budgets along with the correlation of simulated results to the original measured data.

\ 

Author(s) Biography

Michael Steinberger, Ph.D., Lead Architect for SiSoft, has over 30 years experience designing very high speed electronic circuits. Dr. Steinberger holds a Ph.D. from the University of Southern California and has been awarded 14 patents. He is currently responsible for the architecture of SiSoft's Quantum Channel Designer tool for high speed serial channel analysis. Before joining SiSoft, Dr. Steinberger led a group at Cray, Inc. performing SerDes design, high speed channel analysis, PCB design and custom RAM design.

Paul Wildes, Principal Signal Integrity Engineer for SiSoft, is focused on signal integrity analysis for high-end communications and networking platforms. Before joining SiSoft, he spent more than twenty years designing circuits and channels at Cray, Inc. He also developed semi-custom IC's at Bell Labs and performed Unix kernel development at Aeronautics Corporation in Madison, WI. Mr. Wildes received BS and MS degrees in Electrical Engineering from the University of Wisconsin and has been awarded 2 U.S. patents.

Anders Ekholm, Expert Signal Integrity & Timing Analysis for Ericsson AB, started his university education in physics at Umeå University, Umeå, Sweden 1981 and received his M.Sc. from Stockholm University, Stockholm, Sweden 1985. He was involved in postgraduate research at Stockholm University from 1985-1989 in system design. Anders joined Ericsson R&D 1985 working on Power Integrity, EMI/EMC and has more than 20 years’ experience of simulation and modeling of PI,SI and EMI effects in digital systems.

Nicke Svee, HW Designer for Ericsson AB, joined Ericsson AB R&D 1999 working on High Speed serial channels, precision phase lock loops, system level noise budgeting, Signal integrity and timing simulation. He is currently responsible for the performance analysis of combined electrical and optical high speed serial channels. Prior to joining Ericsson Nicke worked for Siemens on medical equipment projects ranging from DC motor controllers to embedded control systems. Nicke received a B.Sc in Electrical Engineering from Royal Institute of Technology in 1996.
1.0 Introduction

The goal of this paper is to produce high speed serial channel models that accurately represent the behaviors and impairments that significantly affect the performance of the system. The models in question are IBIS-AMI models of transmitters and receivers, and lab measurements are used to make sure that the models include all of the relevant impairments. This paper focuses on transmitter models, since their output is directly available for measurement.

One of the challenges is to identify all of the impairments that need to be included in the model. Often, the impairments that are the most important are not the ones that were originally expected. Therefore, it’s important to use experimental procedures and measurement techniques that cover a broad range of phenomena in addition to measurement techniques that are focused on specific impairments.

This paper introduces the spectrum analyzer as a useful general purpose instrument for making broad surveys of signal impairments. Because it operates in the frequency domain rather than the time domain, it offers a view of the system that complements the data collected using time domain instruments such as oscilloscopes and time interval analyzers. As will be demonstrated in this paper, many impairments have easily recognizable signatures in the frequency domain that make it possible to determine which impairments are important in a given transmitter signal. Preliminary estimates of these impairments can be derived from frequency domain data, possibly motivating more dedicated tests to obtain more precise results.

2.0 IBIS-AMI Model Parameters

The IBIS algorithmic modeling interface (AMI) [1] was introduced in 2008 [2] as a way to model the response of the transmitters and receivers used in high speed serial channels. In addition to the classic IBIS analog model, an IBIS-AMI model includes compiled, dynamically loadable library functions that represent the behavior of the equalizers and adaptive loops in the transmitter and/or receiver. The library functions conform to a fixed software interface, making it possible for EDA tools to load and run models that were developed entirely separately from any one EDA tool. This approach has several advantages:

1. The library functions are written using a general purpose programming language such as C, thus giving the model developer a lot of flexibility in choice of model structure or algorithmic content.
2. The implementation details of proprietary equalizer structures and adaptive loops are hidden inside the model and are not visible to the user.
3. A single simulation can contain multiple instances of transmitters and receivers from multiple IP suppliers.
4. The same model can be run without modification in multiple EDA tools.
In addition to the IBIS analog model and the algorithmic model present in the library functions, an IBIS-AMI model uses a separate file to declare the parameters that can be used to control the model. This so-called AMI file usually also contains parameters to be used directly by the EDA tool. These EDA tool parameters must have standardized definitions so that they will be interpreted the same way by multiple EDA tools. These parameters are therefore called “Reserved” parameters.

Some of the Reserved parameters describe jitter and noise characteristics that are to be modeled directly by the EDA tool rather than being implemented inside the algorithmic model [1], [3]. This approach makes it easier to write an AMI model, enables analytic techniques that can be implemented in the EDA tool but not in the AMI model, and assures that a given impairment in models from different IP vendors will be modeled in a consistent way.

An IBIS-AMI model may therefore need values to be supplied for some of the Reserved jitter and noise parameters in order for that model to be complete. Without these parameters, some impairments will be missing and results obtained using these models will be optimistic in a way that is not readily apparent to the model user.

Unfortunately, the IP vendor may not realize that a given impairment is important, and so may not supply values for the associated parameter in the AMI file. Since the AMI file is an ASCII text file, however, the model user or a third party can correct these deficiencies without having to recompile the algorithmic model.

This paper therefore describes laboratory measurements that can be used to make sure that the Reserved parameters in an AMI model are complete and accurate. The AMI parameters addressed in this paper and described in [3] are

- **Tx_Rj**: Transmitter uncorrelated jitter, typically Gaussian distributed
- **Tx_Dj**: Transmitter jitter correlated with the data
- **Tx_Sj**: Transmitter periodic jitter
- **Tx_Sj_Frequency**: Frequency of transmitter periodic jitter
- **Tx_DCD**: Transmitter clock based duty cycle distortion

### 3.0 Introducing the Spectrum Analyzer

The spectrum analyzer is a general purpose radio receiver used to plot the spectral density of a signal over a very wide range of amplitudes, and over either a very broad or very narrow frequency range. It has been an essential piece of test equipment for microwave circuit designers and radio engineers for many decades, but has not typically been used to analyze digital signals.

Figure 1 is a simplified block diagram of a spectrum analyzer.
The signal to be measured passes through a switchable attenuator to a mixer, where it is heterodyned with the output of a swept microwave oscillator. The input attenuator is used to make sure that the mixer is operating in its linear region.

The output frequency of the microwave oscillator is swept across the frequency band of interest. This frequency band can range from one megaherz or less up to many gigaherz. In modern spectrum analyzers, synthesizer techniques are used to sweep the frequency across a broad frequency range while maintaining extremely low phase noise.

The output of the mixer is passed through a low pass filter to obtain a sample that contains only frequency components of the input signal that differ from the oscillator frequency by less than the bandwidth of the low pass filter. That is, at any one time, only the energy in a narrow frequency band around the oscillator frequency is detected. The low pass filter bandwidth is therefore called the resolution bandwidth of the measurement. The resolution bandwidth can be selected in steps that usually go in factors of three between 1 kHz or less to perhaps 3 MHz. The amount of time needed to sweep a given frequency range is inversely proportional to the square of the resolution bandwidth. So larger resolution bandwidths are used to obtain data relatively quickly while smaller resolution bandwidths are used to obtain very low noise, detailed measurements.

The energy at the output of the low pass filter is detected by an energy detector with a very wide dynamic range. That is, the output of the energy detector is an accurate measure of energy from a very low noise floor up to perhaps a few milliwatts. Modern spectrum analyzers typically have a maximum dynamic range of greater than 70 dB.

The output of the energy detector usually drives a logarithmic amplifier to produce a display whose vertical scale is in decibels. The logarithmic scale makes it possible to distinguish all the signals that are present across the entire dynamic range of measurement.
Finally, the sweep oscillator that is used to tune the microwave oscillator is also used to drive the horizontal sweep of the display. Sweep times can range from a few milliseconds up to several minutes, depending on the frequency range being swept and the resolution bandwidth.

4.0 Impairments and Spectral Analysis

This section describes a number of possible impairments that might be present at the output of a transmitter driver, and derives equations for the spectral density to be expected if they are. These equations can therefore be used to quantitatively analyze spectral data from a spectrum analyzer.

4.1 Pseudorandom Data Spectral Density

By definition, a pseudorandom data sequence emulates random data in the sense that each data bit in the sequence is almost completely uncorrelated with respect to the rest of the bits in the sequence. The illusion is not complete, however, in that the sequence repeats, so each bit in the sequence is equal to the bit in the same position in all other iterations of the sequence. Furthermore, a pseudorandom sequence has an odd number of bits, so at each offset time in the autocorrelation function, there will be a net correlation of one bit out of the N bits in the sequence.

Figure 2 illustrates the autocorrelation function $G_{xx}(t) \equiv \int x(t)x(t-\tau)d\tau$ for a pseudorandom sequence of amplitude $\pm v$ containing $N$ bits of duration $T$. Note that since the autocorrelation function is the integral of the signal times itself, the units are volts squared.

The spectral density of the signal is obtained by taking the Fourier transform of the autocorrelation function. Ignoring the DC component of the autocorrelation function, the result is
where the comb function defines a series of Dirac delta functions, with a delta function at each frequency for which the argument assumes an integer value.

Note that this is the spectral density at the output of the driver and does not include the effects of either equalization (de-emphasis) or the channel loss incurred between the driver output and the point at which the spectrum is measured.

4.2 Insertion Loss

To estimate the insertion loss at a given frequency such as half the data rate, one can compare the spectral density for a primary spectral component near DC to the spectral density at the desired frequency, accounting for the de-emphasis equalization introduced at the driver. The calculation assumes that the waveform at the output of the transmitter driver closely approximates a saturated data signal (e.g., square wave) and therefore has a \( \frac{\sin(x)}{x} \) characteristic. Given the spectral density (energy) \( S(\omega) \) and de-emphasis voltage gain \( G(\omega) \), the channel voltage insertion loss \( H(\omega) \) is

\[
H(\omega) = \frac{\omega T}{2} \frac{S(\omega)}{\sin\left(\frac{\omega T}{2}\right)} \frac{1}{G(\omega)}
\]

(EQ 2)

4.3 Clock Based DCD

We refer to the duty cycle distortion produced by imperfect clocks as clock based DCD. The most common case occurs when both the rising edge and falling edge of a half data rate clock are used, usually in an architecture with two parallel data paths, each operating at one half the data rate.

Clock based DCD in either the transmitter or the receiver can significantly affect serial channel performance, and therefore should be represented accurately in the jitter parameters associated with an IBIS-AMI model.

The analysis and discussion in this section only addresses the case of a half data rate clock. While the techniques presented here can be extended to more general cases, such an extension is beyond the scope of this work.

In this section, we will demonstrate that clock based DCD has a very distinct spectral signature, and therefore can be reliably identified and measured using a spectrum analyzer.
The result can be confirmed by applying specialized procedures using a time interval analyzer. The result from spectral analysis provides a useful preliminary estimate of the clock based DCD while the time interval analyzer method is more direct and therefore produces a more precise result.

Both measurement procedures assume that the data is a repeating data pattern with an odd number of bits, such as a PRBS7 or a PRBS9 pseudorandom bit sequence. If such a data pattern is modulated by clock based DCD, then one repetition of the data pattern will end on a different phase of the half rate clock than it started. That difference will be evident in the signal because of the DCD. The only way to get a pattern that repeats itself precisely is to concatenate two iterations of the same data pattern. Thus, the clock based DCD will be evident as a characteristic that appears when two iterations of the data pattern are concatenated and does not appear when only single iterations of the data pattern are averaged together.

When a data signal with a repeating data pattern is analyzed using a spectrum analyzer, the primary spectrum of the signal will have discrete spectral components spaced by the pattern repetition rate. However, when such a pattern with an odd number of bits is modulated by clock based DCD, there will be additional spectral components spaced at the repetition rate of a double data pattern, or in other words one half the frequency spacing of the primary spectral components. The mathematics of the spectral components generated by clock based DCD is presented later in this section.

As demonstrated to us by Daniel Chow of Altera, the observation that clock based DCD doubles the length of the pattern that actually repeats can be applied to time interval analyzers as well. In normal operation, a time interval analyzer automatically identifies the repeating data pattern, locks to it and accumulates an average over all repetitions of the data pattern. It is also possible, however, to set the time interval analyzer to expect a data pattern which is twice as long as the actual data pattern. For example, even though a PRBS9 data sequence is 511 bits long, it is possible to set the time interval analyzer to expect a data pattern which is 1022 bits long. When configured in this way, the time interval analyzer will record threshold crossings times for both phases of the clock for each bit in the sequence. Comparing the threshold crossing times for edges which are exactly one data pattern apart therefore provides a direct measure of clock based DCD.

### 4.3.1 Clock Based DCD Spectral Analysis

To estimate the magnitude of the spectral components, the clock based DCD is treated as an instance of phase modulation and analyzed using communications analysis techniques developed for radio systems. Note that the following analysis is performed at the output of the transmitter driver. The effect of any measurement channel will be added at the end of the analysis.

Suppose that, in the absence of clock based DCD, the signal at the output of a transmitter is the sum of some number of discrete spectral components:
\[ x(t) = \sum_{k} a_k \cos(\omega_k t + \phi_k) \]  \hspace{1cm} (EQ 3)

Then when clock based DCD of peak magnitude \( \tau \) seconds and wave shape \( v_{ck}(t) \) is applied to the signal, the output of the transmitter is

\[ x(t) = \sum_{k} a_k \cos(\omega_k(t + \tau v_{ck}(t)) + \phi_k) \]  \hspace{1cm} (EQ 4)

Note that \( v_{ck}(t) \) is a square wave at half the data rate, with values \( \pm 1.0 \).

As explained in [1], Equation 4 can be expanded into a sum of Bessel function coefficients times spectral components represented as cosine functions. In principle, these spectral components can occupy a very broad bandwidth. However, since the amplitude of the clock based DCD is usually a small fraction of a symbol time, we will use only the low modulation, first order terms of this expansion, also as explained in [1].

Furthermore, both for the sake of clarity and because it determines how the spectral data is to be analyzed, we will consider each term in Equation 4 separately.

Expanding \( v_{ck}(t) \) into its Fourier transform,

\[ v_{ck}(t) = \sum_{n=0} \frac{(-1)^n}{(2n+1)\pi} \cos\left(2\pi(2n+1)\frac{1}{2T}(t + t_0)\right) \]  \hspace{1cm} (EQ 5)

where \( T \) is the symbol duration and \( t_0 \) is a fixed time offset.

Note that since \( v_{ck}(t) \) is a square wave, the phase deviation is always \( \pm \omega_k \tau \). Therefore, define a mean squared phase deviation and an amplitude multiplier which is a function of the mean square phase modulation

\[ D_{\phi k} = (\omega_k \tau)^2 \]  \hspace{1cm} (EQ 6)

\[ A_{ck} \approx e^{-D_{\phi k}/2} \approx 1 \]  \hspace{1cm} (EQ 7)

Then the low modulation, first order approximation for one term in Equation 4 is

\[ a_k \cos(\omega_k(t + \tau v_{ck}(t)) + \phi_k) \]

\[ = a_k A_{ck}\left(\cos(\omega_k t + \phi_k) + \frac{\omega_k \tau}{2\pi} \left(\cos(\omega_k t + \phi_k + \frac{t + t_0}{2T}) + \cos(\omega_k t + \phi_k - \frac{t + t_0}{2T})\right)\right) \]  \hspace{1cm} (EQ 8)

Dropping all the offsets and second order terms for the sake of clarity,
In other words, in addition to the original spectral component \(a_k \cos(\omega_k t)\), there are modulation sidebands \(a_k \frac{2\omega_k \tau}{\pi} \cos(\omega_k t + 2\pi \frac{t}{2T})\) and \(a_k \frac{2\omega_k \tau}{\pi} \cos(\omega_k t - 2\pi \frac{t}{2T})\) separated from the original spectral component by a frequency equal to one half the data rate. Thus, for example, a spectral component at the data pattern repetition rate \(\frac{1}{NT}\) will have modulation sidebands at the pattern repetition rate plus half the data rate \(\frac{1}{NT} + \frac{1}{2T}\) and at the pattern repetition rate minus one half the data rate \(\frac{1}{NT} - \frac{1}{2T}\). For a PRBS7 data pattern at 10 Gb/s, the data spectral component at 79 MHz will have modulation sidebands at 5.079 GHz and -4.921 GHz while a data spectral component at 5.039 GHz will have modulation sidebands at 39 MHz and 10.039 GHz.

Note that if the original spectral component is near DC, then \(\omega_k\) will be relatively small, and the modulation sidebands will also be relatively small. This situation is illustrated in Figure 3. Also, since the signal is real-valued, every spectral component at a positive frequency is matched by a corresponding spectral component at the negative of that frequency, and vice versa.

\[
a_k \cos(\omega_k (t + \tau v_c(t))) \approx a_k \left( \cos(\omega_k t) + \frac{2\omega_k \tau}{\pi} \left( \cos(\omega_k t + 2\pi \frac{t}{2T}) + \cos(\omega_k t - 2\pi \frac{t}{2T}) \right) \right)
\]

(EQ 9)

**FIGURE 3. Modulation sidebands for low frequency spectral component**

Similarly, a spectral component at just over half the data rate will have modulation sidebands at that frequency minus half the data rate (or in other words near DC) and at that frequency plus half the data rate (or in other words slightly above the data rate).
To illustrate the effect of clock based DCD, we ran a simple computational experiment in which we generated an ideal PRBS7 with 0.25 UI peak to peak of clock based DCD. The result is shown in Figure 5.

One can estimate the level of the clock based DCD using Equation 9. That is, at least before the signal encounters any transmission loss, the clock based DCD is indicated by the ratio of the amplitude $a_{DCD} \equiv a_k \frac{2\omega_k \tau}{\pi}$ of the modulation sideband to the amplitude $a_k$ of the associated spectral component.

$$DCD_{clock} = \frac{2\tau}{T} = \frac{2\pi}{2\omega_k T} \frac{a_{DCD}}{a_k}$$

(EQ 10)
The case of \( \omega_k > \frac{2\pi}{2T} \) by only a small amount is a useful case in practice because in this case one can use the approximation

\[
DCD_{\text{clock}} \approx \frac{a_{\text{DCD}}}{a_k} \quad \text{(EQ 11)}
\]

In other words if the spectral component frequency slightly is greater than half the data rate, then the modulation sidebands, and particularly the modulation sidebands near DC, are a direct indication of the amount of clock based DCD in the signal at the output of the transmitter driver.

When interpreting these results, however, it’s important to include the loss of the interconnect network between the transmitter driver output and the measurement point. Consider that Equation 10 uses the ratio of the spectral densities at two very different frequencies. For example, \( a_{\text{DCD}} \) may be the square root of the spectral density at a frequency near DC while \( a_k \) is the square root of the spectral density at a frequency slightly greater than half the data rate. Therefore at the measurement point, the spectral component used to determine \( a_k \) may have encountered significantly more attenuation than the attenuation encountered by the spectral component used to determine \( a_{\text{DCD}} \).

4.4 Data Based DCD

We refer to the duty cycle distortion produced when the data rise time is different from the data fall time as data based DCD. High speed serial channels almost always use differential electrical interfaces which are designed to be completely symmetrical; so this effect should normally be small. Nonetheless, some asymmetry can be present, due primarily to on-chip variations.

Data based DCD occurs every time there is a data transition. For example, if the rise time is faster than the fall time, then the onset of a “one” will occur early and the onset of a “zero” will occur late. In either case, the distortion will be a very short positive pulse which is synchronous with the clock, as illustrated in Figure 6. For rising edges, the duty cycle distortion immediately precedes the clock edge while for falling edges, the duty cycle distortion immediately follows the clock edge. An analogous set of conditions occurs when the rise time is longer than the fall time.
While the dependence on data pattern makes it difficult to calculate most of the frequency domain effects of data based DCD, the synchronization of the distortion with the clock generates a spectral component at the data rate which is unmistakable and relatively easy to calculate. Given a clock period \( T \), a peak magnitude of \( \tau \) for the data based duty cycle distortion and peak amplitude \( v \) for the data signal, the Fourier coefficient at the data rate frequency for a single transition at bit \( k \) is either

\[
V_k(\omega_c) = \frac{\tau}{T} \int_0^T 2v \cos(\omega_c t) dt = \frac{2v}{T} \left( \sin(\omega_c \tau) \right) = \frac{2v}{\pi} \sin(\omega_c \tau) \tag{EQ 12}
\]

or

\[
V_k(\omega_c) = \frac{-\tau}{T} \int_0^T 2v \cos(\omega_c t) dt = \frac{2v}{T} \left( 0 - \sin(-\omega_c \tau) \right) = \frac{2v}{\pi} \sin(\omega_c \tau) \tag{EQ 13}
\]
If $V_k(\omega_c)$ is averaged over all bits, those with transitions and those without, the spectral component at the data rate frequency is

$$V(\omega_c) = \frac{2\nu n}{\pi N} \sin(\omega_c \tau) \tag{EQ 14}$$

where $N$ is the total number of bits and $n$ is the number of transitions. For a pseudorandom sequence, $n = \frac{N+1}{2}$. Thus, for a pseudorandom sequence, the spectral density (energy) at the data rate frequency is

$$S(\omega_c) = V(\omega_c)^2 = \left(\frac{\nu(N+1)}{\pi N} \sin(\omega_c \tau)\right)^2 \approx \left(\frac{\nu(N+1)2\tau}{N}\right)^2 \tag{EQ 15}$$

Using Equation 1 to calculate the spectral density $S(0)$ for a spectral component near DC,

$$DCD_{data} = \frac{2\tau}{T} = \frac{1}{N+1} \sqrt{\frac{S(\omega_c)}{S(0)}} \tag{EQ 16}$$

As is the case when interpreting any spectral data, the effects of the equalization (de-emphasis) and channel loss must be accounted for using an equation such as Equation 2 above.

illustrates the results of a computation experiment with 0.25 UI of data based DCD, analogous to for clock based DCD.

![FIGURE 7. Results of computational experiment with 0.25 UI of data based DCD](image)
4.5 Clock Leakage

Section 4.3 on page 7 (Clock Based DCD) exemplifies the modulation of the delay of clock and/or data paths by some impairment; and communications analysis of the resulting phase modulation is the appropriate approach for such impairments. It is also possible, however, that a signal such as an internal clock leaks directly into the output signal as an added spectral component.

There are several ways in which additive noise might be included in the performance analysis of a high speed serial channel. The most direct and accurate approach would be to model the noise as a form of crosstalk and use one of the several available methods (see, e.g., [5]). Unfortunately, the parameters either currently defined in IBIS [1] or currently proposed [2] do not offer any parameters that would facilitate this form of analysis. The best alternative right now would be to treat the clock leakage as a form of periodic jitter using the Tx_Sj AMI parameter.

For the low levels of clock leakage typically encountered, one can approximate the Tx_Sj by assuming that the impairing signal is shifting the signal at the edge of the eye diagram. Therefore, the slope of voltage vs. time at the edge of the eye diagram can be used to convert the peak clock leakage voltage $v$ to an equivalent Tx_Sj amplitude.

\[ Tx_Sj = \frac{v \frac{1}{dv}}{dt} \]  

(EQ 17)

To convert from the dBm (dB, milliwatts) to voltage,

\[ dBm = 10 \log \left( 1000 \frac{v^2}{2Z_0} \right) \]  

(EQ 18)

\[ v = \frac{2Z_0}{1000} 10^{dBm/20} \]  

(EQ 19)

4.6 Reference Clock Phase Noise

Spectrum analyzers are widely used in microwave engineering to measure the phase noise of oscillators. In a high speed serial channel and associated IBIS-AMI models, the oscillator phase noise is a truly Gaussian process that is independent of the data, and should therefore be characterized as Rj.

Figure 8 is an example of a spectrum analyzer sweep across a frequency band which is much smaller than the data rate. Note the much narrower sweeps than this, and with lower resolution bandwidth, are routine. In this example, a dashed red line is used to suggest what the spectrum would look like if the noise floor were lower. The energy under the dashed red curve can be integrated to determine the phase noise of the oscillator. In fact, many spectrum analyzers can perform this integral automatically.
If there is a need to determine the oscillator phase noise precisely, the most effective technique is to use as short a data pattern as possible, with an alternating “1/0” data pattern being ideal. That way, the signal energy will be concentrated in fewer spectral components, thus bringing those spectral components further out of the noise.

The oscillator phase noise suggested in Figure 8 is too small to be a significant impairment for a high speed serial channel. If the oscillator in a system has excessive phase noise, that will be clearly evident in a spectrum analyzer scan of the transmitted signal.

#### 4.7 Power Supply Noise

There are two ways in which power supply noise might affect a high speed serial channel: it could either be coupled directly into the transmitted signal, similar to the effect of clock leakage as described in Section 4.5 on page 14, or it could modulate the delay of the clock and data paths, similar to the effect of clock based DCD as described in Section 4.3 on page 7. A spectrum analyzer will detect either of these two effects.
In order to distinguish the effects of power supply noise from other effects in the system, one must know what spectral characteristics to look for. The characteristics of the high performance logic designs typically associated with high speed serial channels make it easy to know what spectral characteristics to look for.

High performance logic designs are necessarily highly pipelined. That is, they use a lot of flip flops to make sure that timing margins are met. A flip flop necessarily draws a pulse of current at exactly the same time on every clock pulse. Furthermore, the pulses of current driving the flip flops must be supplied by a network of clock distribution amplifiers. The net result is that the current drawn by the flip flops and clock distribution amplifiers is fully synchronous with the IC’s core clock. The current consumed by the flip flops and clock distribution network typically consumes over 70% of the current in the device, and all of this current occurs at the core clock frequency and its harmonics. It turns out that the timing of the remaining combinatorial logic is only loosely correlated with the core clock, and can be accurately modeled as a shot noise process that generates a broadband Gaussian current load.

Thus, while it will usually be impractical to detect the effect of the current consumed by the combinatorial logic, a spectrum analyzer can easily detect the effect of the current consumed by the flip flops and clock distribution network.

In summary, the effect of power supply noise will be evident in a transmitted signal as modulation sidebands at plus or minus the core clock frequency. The most effective way to separate these sidebands from other phenomena in the system is to use a very short data pattern such as an alternating “1/0” pattern and look for modulation sidebands at one half the data rate plus or minus the core clock frequency.

4.8 Crosstalk

System crosstalk can be detected by driving the potential aggressors with a different data pattern than the victim. Since the crosstalk should be at a lower amplitude than the desired signal, it would be good to make the aggressor data pattern shorter than the victim data pattern so that its spectral components are more prominent. For example, the aggressors might be driven with a PRBS7 data pattern while the victim is driven with a PRBS15 data pattern or real system traffic.

5.0 Experimental Results

5.1 Clock Based DCD

The experimental investigation of clock based DCD started with the initial spectrum analyzer scan shown in Figure 9.
The transmitters in the system under evaluation were configured to generate a PRBS7 pattern, and a full spectral sweep was made. The results showed that while several of the transmitters exhibited the spectral characteristics of a fairly clean PRBS7 waveform, several of the transmitters exhibited spectral characteristics very similar to those in Figure 5, and therefore indicating the presence of significant amounts of clock based DCD. Figure 9 is the most extreme of these cases.

The subsequent experimental study was performed under the following variations:

- Two channels: The best channel (Channel A) and the worst channel (Channel C). The interconnect for these two channels is essentially identical, as demonstrated from measured data not shown here, and both channels were driven from the same device. The difference in channels is therefore primarily due to on-chip variation of the transmitter drivers.
- Two levels of transmit de-emphasis: 6 dB and 0 dB.
- Three measurement methods:
  - Time interval analyzer
  - Eye diagram measurement
  - Spectral analysis
- PRBS9 data pattern
- IC core clocks disabled (minimal power supply noise)

Figure 10 shows the results accumulated using a time interval analyzer. The difference was calculated between the time interval error for a given bit transition in the first iteration of the data pattern to the time interval error for the same bit in the second iteration of the data pattern. The differences for the even numbered bits were accumulated into a histogram, and the differences for the odd numbered bits were accumulated into a separate histogram.
FIGURE 10. Time interval analyzer: Difference in time interval error between first and second iteration of data pattern

From Figure 10, the clock based DCD is essentially zero for Channel A and $6.0 \pm 0.1$ pS for Channel C. These values appear to be independent of choice of transmitter de-emphasis.

Figure 11 shows the eye diagrams measured for Channel A and Channel C with 6 dB transmitter de-emphasis. Whereas the crossing region at the edge of the eye is extremely clean for Channel A, the crossing region for Channel C exhibits a diamond pattern that is characteristic of clock based DCD. While this measurement is not as precise as the time interval error measurement, it does confirm the value of 6 pS DCD for Channel C.

Note that this measurement was only possible because the 6 dB de-emphasis was nearly optimal for this channel, resulting in an extremely clean eye crossing region. This measurement was not possible when the de-emphasis was 0 dB.
Spectrum analyzer measurements of Channel A and Channel C were made using a PRBS7 pattern and a 10kHz resolution bandwidth. The spectrum analyzer measurements were single ended. The spectrum was scanned for any spectral components that were significantly above the noise floor. This scan therefore detected not only clock based DCD but several other phenomena as well. The recorded data points are given in Table 1. Only a couple of the spectral components associated with the data pattern or the clock based DCD were recorded.

**TABLE 1. Spectrum analyzer measurements using PRBS7 data pattern**

<table>
<thead>
<tr>
<th>Frequency</th>
<th>Ch A 0dB (dBm)</th>
<th>Ch A 6dB (dBm)</th>
<th>Ch C 0 dB (dBm)</th>
<th>Ch C 6 dB (dBm)</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>38.8 MHz</td>
<td>-89.7</td>
<td>-84.2</td>
<td>-66 (use -62.9)</td>
<td>-62.4</td>
<td>Clock based DCD</td>
</tr>
<tr>
<td>77.4 MHz</td>
<td>-33.6</td>
<td>-42.7</td>
<td>-33.9</td>
<td>-39.3</td>
<td>PRBS7 spectral component</td>
</tr>
<tr>
<td>116 MHz</td>
<td>-94.05</td>
<td>-83.9</td>
<td>-62.9</td>
<td>-61.7</td>
<td>Clock based DCD</td>
</tr>
<tr>
<td>983.09 MHz</td>
<td>-74.4</td>
<td>-83.66</td>
<td>-81</td>
<td>-82.2</td>
<td>Parallel clock leakage</td>
</tr>
<tr>
<td>4.9152 GHz</td>
<td>-56.7</td>
<td>-59.1</td>
<td>-59.2</td>
<td>-61.3</td>
<td>Half data rate clock leakage</td>
</tr>
<tr>
<td>4.9538 GHz</td>
<td>-52.4</td>
<td>-46.6</td>
<td>-45.2</td>
<td>-46.16</td>
<td>PRBS7 spectral component</td>
</tr>
<tr>
<td>9.8304 GHz</td>
<td>-60.2</td>
<td>-61.7</td>
<td>-57.2</td>
<td>-60.2</td>
<td>Data rate clock leakage</td>
</tr>
<tr>
<td><strong>Estimated insertion loss at 4.95 GHz (dB)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td><strong>-14.8</strong></td>
</tr>
</tbody>
</table>

**FIGURE 11. Eye diagrams for Channel A and Channel C with 6dB transmitter de-emphasis**
Table 1 also shows the estimated insertion loss at 4.95 GHz, as estimated using Equation 2. Although there is considerable variation in the estimates, an average value of -7.4 dB seems reasonable. This conclusion is also consistent with the fact that 6 dB of de-emphasis appears to be nearly optimal for these channels.

Finally, Table 1 shows the DCD calculated using Equation 10 and the gain compensation from Equation 2. Since the measured spectral density at 38 MHz for the Channel C 0 dB case is so different from all the other measurements for Channel C, the calculation uses the spectral density for 116 MHz instead.

As can be seen comparing Table 1 to Figure 10, the results obtained using the spectrum analyzer are consistent with the measurements subsequently made using a time interval analyzer.

For the IBIS-AMI transmitter model used to simulate this system, the parameter value should be

$$\text{Tx\_DCD} = 0.06 \text{ UI}$$

This parameter value was added to the AMI model and the modified model was used in a time domain simulation of the system under test.

Figure 12 compares the simulation results to the measured data already presented in Figure 9. Note that in the simulation results, a threshold has been drawn to indicate the approximate noise floor of the measurement.
5.2 Data Rate Clock Leakage

We know the transmitter we measured has a half rate clock architecture. Therefore it is unlikely that a second harmonic of the internal clock would have significant energy. Fur-
thermore, we observed some minor but consistent differences between the wave shape at the true and complement pins. We therefore assume that most of the spectral content at the data rate frequency (9.8 GHz) is due to data based DCD.

For the purposes of this analysis, we will assume that the number of dB’s of channel insertion loss is proportional to the frequency. This is, at best, an approximation, albeit a reasonable one in most cases.

Table 2 presents the estimates of the data based DCD using both the time interval analyzer and the spectrum analyzer. Equation 16 was used to reduce the spectrum analyzer data.

Table 2. Estimates of data based DCD

<table>
<thead>
<tr>
<th>Method</th>
<th>Ch A 0dB (pS)</th>
<th>Ch A 6dB (pS)</th>
<th>Ch C 0 dB (pS)</th>
<th>Ch C 6 dB (pS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time interval analyzer</td>
<td>0.13</td>
<td>0.004</td>
<td>0.19</td>
<td>0.21</td>
</tr>
<tr>
<td>Spectrum analyzer</td>
<td>0.21</td>
<td>0.25</td>
<td>0.31</td>
<td>0.20</td>
</tr>
</tbody>
</table>

For both methods and all cases, the estimated data based DCD is quite low; so it’s not clear that any of these estimates is particularly accurate. Nonetheless, the results are almost all the same order of magnitude, so a value of approximately 0.2 pS data based DCD for these drivers seems reasonable.

Since the clock based DCD is much larger than the data based DCD, it would not be a good idea to attempt to combine the data based DCD with the clock based DCD. It might be reasonable, however, to consider the data based DCD as being similar to DJ in the sense that it is directly correlated to the data. One might therefore add the following to the IBIS-AMI model of the transmitter:

\[
\text{Tx}_\text{Dj} = 0.002 \text{ UI}
\]

5.3 Half Data Rate Clock Leakage

Since the data pattern that was measured was an even number of bits long, it is possible to analyze the time interval analyzer data to extract the half rate clock leakage. The method is to combine the data for the rising edges of even numbered bits with the falling edges of odd numbered bits, and vice versa.

The spectral data was reduced using Equation 17 and Equation 19. The slope at the edge of the eye was derived from measured waveform data.
As was the case with the data based DCD results, none of these results looks particularly accurate. The results from the spectrum analyzer appear to be perhaps twice those of the time interval analyzer. One possibility is that since the spectrum analyzer measurements were single ended, some of the half data rate clock leakage could have been in common mode rather than differential mode. This hypothesis could be checked by repeating the spectral measurements using a microwave differential coupler.

Leakage of the half data rate clock is a form of periodic jitter, so it could be added to the IBIS-AMI model of the transmitter as

\[ \text{Tx}_{\text{SJ}} = 0.002 \text{ UI} \]
\[ \text{Tx}_{\text{SJ}} \text{ Frequency} = 4.915\times 10^9 \]

### 5.4 Parallel Clock Leakage

Since the time interval analyzer assumes that the signal is a repeating data pattern, and the pattern length is not an integer multiple of ten, the time interval analyzer will average out the effects of any leakage of a clock at one tenth of the data rate. Therefore, there is no time interval analyzer result to compare to for this case.

The data reduction method was the same as for the half data rate clock leakage above.

<table>
<thead>
<tr>
<th>Method</th>
<th>Ch A 0dB (pS)</th>
<th>Ch A 6dB (pS)</th>
<th>Ch C 0 dB (pS)</th>
<th>Ch C 6 dB (pS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time interval analyzer</td>
<td>0</td>
<td>0.24</td>
<td>0.17</td>
<td>0.20</td>
</tr>
<tr>
<td>Spectrum analyzer</td>
<td>0.66</td>
<td>0.56</td>
<td>0.53</td>
<td>0.44</td>
</tr>
</tbody>
</table>

In this case, the spectrum analyzer was sensitive enough to clearly detect an impairment that does not appear to be large enough to significantly affect system performance.

As far as the IBIS-AMI transmitter model is concerned,

**No action**
6.0 Conclusions

This paper has introduced the spectrum analyzer as a useful tool for detecting and identifying specific impairments in a transmitted signal, especially during initial evaluation of a system. Equations and analytic methods are provided for making preliminary estimates of these impairments, and experimental results are presented as a demonstration of these methods. The results obtained are consistent with those obtained using more specialized methods applied after the spectrum analyzer results indicated that they were needed.

This paper has shown how experimental results using the spectrum analyzer and other methods can then be reflected in improved IBIS-AMI models so that subsequent simulations match the measured data, making the performance analysis results more reliable.

7.0 Acknowledgements

The authors gratefully acknowledge Davood Khoda at Ericsson for taking many of the measurements used in this paper, Daniel Chow of Altera for informative discussions and fruitful collaboration in the laboratory, and Todd Westerhoff from SiSoft for helping to submit the proposal for this paper.

8.0 References