A Versatile Scheme for the Validation, Testing and Debugging of High Speed Serial Interfaces

Yongquan Fan and Zeljko Zilic
Department of ECE, McGill University
yongquan.fan@mail.mcgill.ca, zeljko.zilic@mcgill.ca

ABSTRACT

The High-Speed Serial Interface (HSSI) is a cornerstone of the modern communications. To achieve high data rates, sophisticated techniques such as equalization and pre-compensation have now become common in HSSIs. With the concurrent increasing of design complexity and decreasing of the timing budget, the post-silicon validation, debugging and testing of HSSIs are becoming critical. This paper presents a versatile scheme to accelerate the post-silicon validation. Using a novel jitter injection scheme and an FPGA-based Bit Error Rate Tester (BERT), we can validate and test HSSIs without the need of high-speed Automatic Test Equipment (ATE) instruments and Design-for-Test (DFT) features; this scheme also overcomes existing ATE instrument limitations. We can also utilize ATE to provide a more versatile scheme for HSSI validation, debugging and testing.

1. Introduction

The HSSI is becoming a prevalent means of high-speed communication between blocks, ICs and systems. There are numerous HSSI standards, such as SATA, Fiber Channel and XAUI, addressing a wide range of applications. In HSSI, the transmitter (Tx) takes parallel data, converts it into the serial stream with the clock signal embedded, and then drives the transmission media. The receiver (Rx) accepts the high-speed serial data, extracts the clock and restores the data to the original parallel format.

When the HSSI data rate reaches a few Gigabits per second (Gbps), its timing budget is getting very tight. The timing budget of HSSI is reflected in various jitter specifications. On the Tx side, jitter budget defines how much timing deviation the Tx can generate; in the Rx side, jitter tolerance defines how much time deviation the Rx can tolerate. As the traditional "Guaranteed by Design" paradigm cannot be applied anymore, it is becoming imperative to thoroughly validate the tight timing specifications and other parameters *in silicon,* in order to guarantee the design and device quality. Jitter specifications are usually defined at 10^{-12} bit error rate or lower. It is challenging and costly to qualify the timing specifications mainly for three reasons:

The increasing demand for more bandwidth is continuously pushing the data communication rate higher, at a pace faster than the test equipment evolves; systematic HSSI testing solutions on ATE for data rates 6 Gbps and higher are not mature yet [2].

Cost goals set by the marketplace demand competitive test solutions – testing needs to be done as fast as possible using as inexpensive equipment as possible; it is infeasible to use tradition bench solutions in a production environment because it takes hours or days to qualify the jitter and BER performance.

Validating the jitter performance across Process, Voltage and Temperature (PVT) corners is becoming necessary with the continuing scale of the process technology, but the validation is very time-consuming; shortening the validation time (including debugging when necessary) would directly shorten the time-to-market, which provides great competitive advantages in gaining profit and market share.

Motivated by the economic significance of validating, testing and debugging HSSIs, we strive to develop methodologies to address the above challenges. This paper presents a versatile scheme to validate, test and debug HSSIs. Depending on applications, we want to either use ATE high-speed instruments or to perform external loopback testing without ATE. The scheme can also perform the self-calibration and diagnosis. The external loopback scheme also overcomes some of the current ATE limitations. In the remainder of the paper, Section 2 further details the HSSI technology and testing challenges. Section 3 introduces an ATE-based approach. In Section 4, we present the whole versatile scheme, including a novel loopback-based approach.

2. Background

2.1 HSSI Technology Trends

The HSSI protocols are continually evolving to higher speeds to meet the demand for higher bandwidth. One example is the SATA: when the SATA 1.0 Working Group was formed in February 2000 to design SATA for desktops, the target speed was only 1.5Gps; in 2004 it evolved to 3Gps and now SATA 3.0 provides 6Gbps data rate.

978-1-4244-4823-4/09/$25.00 ©2009 IEEE

The increasing bandwidth requirements are forcing silicon vendors to provide HSSIs with higher speeds. In 2002, the highest data rate in Altera FPGAs was only 1.25 Gbps per channel, available in its Mecury devices [3]; now in Altera Stratix IV GT FPGAs, the rate has increased to 11.3 Gbps per channel, with up to 48 HSSIs on each device [4]. Another FPGA provider, Xilinx, provides up to thirty-six 11.2 Gbps HSSIs in its Virtex-6 and Spartan-6 FPGAs, capable of supporting 40G/100G applications [5].

When we push the speed envelope and increase the integration, many signal integrity related issues arise, such as timing jitter, noise and frequency loss. A few key technologies have been developed recently to address these issues [1]. Pre-emphasis and equalization techniques are used to compensate frequency-related losses, especially those related to Printed Circuit Board (PCB) design due to skin effect and dielectric loss. Pre-emphasis is used in the transmitter to boost the high-frequency components of a data signal before it is launched to the transmission medium. Equalization is used in the receiver to re-shape the data signal when it enters the receiver in order to interpret the data correctly.

With the integration increase, there is a prominent trend to implement multiple data rates in a single HSSI to accommodate multiple protocols. This requires the HSSI capable of providing multiple rate clock signals. The Phase Locked Loop (PLL) is widely used for clock generation, where a Voltage-Controlled Oscillator (VCO) is a key component. There are two types of oscillators: Ring Oscillator (RO) and LC tank oscillator (LC tank). RO has the advantages of small chip area and wide tunable frequency range, but LC tanks provide lower noise and better jitter performance [6], [7]. Multiple data rates can be implemented by changing divider ratios inside the PLL or by providing additional VCOs. In Altera Stratix IV GT FPGAs, the RO can support data rates from 600Mbps to 10.3 Gbps; two LC tanks are also implemented in this device, one with 4.9~6.375 Gbps optimized for PCIe/CEI-6 compliance and the other with 9.9~11.3 Gbps optimized for XLAUI/CAUI/CEI-11G compliance [8].

A side-effect of implementing multiple data rates is that the jitter performance of the HSSI can vary across the data range. If the same PLL is used at multiple speeds, such as 6 Gbps and 8.5Gbps, one data rate can be susceptible to higher jitter because the VCO can only be optimized at one speed. If different PLLs are used to support different data rates, the performance at one data rate does not relate to another rate. In either case, good performance at a higher data rate does not guarantee better margin at lower data rates, because the PLL characteristics may be different.

2.2 Validation, Testing and Debugging Challenges

With the increasing data rate and the degree of integration, it is challenging to design fault-free electronic products. As a consequence, close to 25% of all design resources at Intel are now spent on post-fabrication validation [9]. It is challenging and expensive to qualify the HSSI devices, especially the jitter performance, including transmitter jitter and receiver jitter tolerance. Jitter is the deviation of a signal from its ideal timing that may cause bit errors. Total Jitter (TJ) consists of Deterministic Jitter (DJ) and Random Jitter (RJ) [21]. Many HSSI standards define jitter performance at the BER level of 10^{-12}, which requires running at least 10^{13} bits. This requirement fundamentally limits test speed: for instance, at 3Gbps, it takes around one hour to run so many bits. With some emerging applications demanding 10^{-14} BER, direct measurements are even further from being practical.

In addition, many settings in the HSSI may affect its jitter and other performance. A few examples include the boost control settings in the equalizer, the bandwidth setting in the PLL and the driver strength settings in the transmitter. These settings are quite common in today's HSSIs. Choosing the optimal settings for the whole HSSI from hundreds or even thousands of available settings is challenging. Debugging a design or device issue is also becoming more challenging because many interactions may exist in a complicated HSSI. It requires tremendous amount of resources in validation, debugging and testing in order to qualify the design, identify issues, and guarantee the device quality.

Because of the long test time, the jitter performance of multi-gigabit HSSI devices is traditionally only evaluated on bench in limited combinations of PVT. Besides the long test time, another reason that jitter is not qualified in production is the availability of ATE instruments, especially for very high-speed applications. For example, to evaluate 8.5GHz FC devices, we prefer the signal generator for the receiver and the digitizer for the transmitter with a bandwidth much higher than 8.5G (such as 15GHz), but they are not commercially mature yet on ATE [2]. Furthermore, there are currently few systematic solutions that can perform HSSI compliance tests accurately and cost-efficiently. Most companies only do loopback tests to check the functionality. Some HSSI parameters, such as Tx jitter and Rx jitter tolerance then need to be guaranteed by design.

Unfortunately, the "Guaranteed by Design" paradigm is no longer valid as we keep increasing the data rate, which results in tightening specifications, such as jitter budget. The devices can increasingly fail just because they do not comply with the jitter specifications. It is hence becoming imperative to develop systematic HSSI validation and testing solutions to ensure the design quality and the device quality, either for use on ATE or standalone.

3. ATE-based Approaches

ATE is traditionally used in production testing. Due to its high throughput, there is a trend in recently years to also use ATE to facilitate validation and debugging. In [10], we propose an

accelerated Rx jitter tolerance testing and validation scheme, where a high-speed Arbitrary Waveform Generator (AWG) is used to generate test signals. In [11], we present a systematic Tx jitter test solution, where a high bandwidth digitizer GigaDig is used to capture the Tx output. Figure 1 shows the test setup. The AWG and digitizer are available on Catalyst/Tiger ATE platforms from Teradyne [12]. Even though [10] and [11] concentrate on jitter testing, the test setup can be used for the whole HSSI validation and testing; jitter testing covers most of HSSI functionality and the other parameters are straightforward to test with this setup.

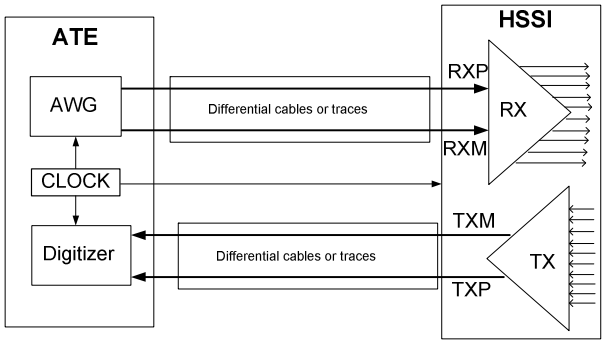

Figure 1. ATE-based HSSI Testing Solution

3.1 Tx Testing Solution

The GigaDig on the ATE is an under-sampling-based digitizer with an analog bandwidth more than 9GHz. Figure 2 shows the digitizer captured waveform of a 6Gbps data signal that repeats a 20-bit pattern with both low and high transition densities -- 00000001111111010101000111.

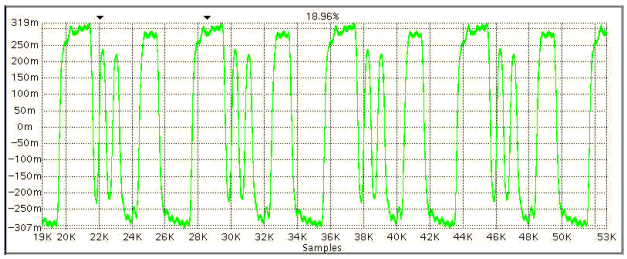

Figure 2. Captured 6G waveform (only 45 bits shown)

By manipulating clock frequencies, we can get the desired effective sampling rate [11]. In Figure 2, each data bit has 400 samples, which gives an effective sampling rate of 2400GHz. We can capture multiple cycles of the 20-bit test pattern and then extract DJ, RJ and TJ from time domain or frequency domain [11]. Figure 3 shows an example of the extracted RJ and DJ at each edge of the 20-bit test pattern. TJ can be calculated by convoluting the RJ and DJ. Figure 4 plots the Probability Density Function (PDF) and Cumulative Density Function (CDF) of the device TJ. Once we get the TJ CDF, we can get TJ peak-to-peak value at a certain BER level by calculating the time difference between t_1 and t_2:

$$TJ_{peak-to-peak@BER} = t_2 - t_1$$

where t_1 and t_2 satisfy

$$TJ\_CDF(t_2) = 1 - BER/2$$
$$TJ\_CDF(t_1) = BER/2$$

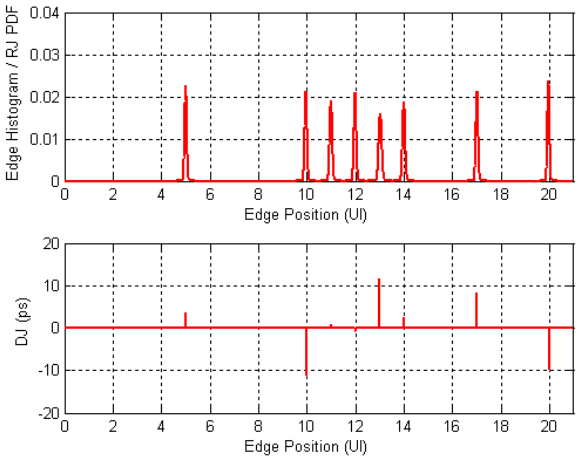

Figure 3. RJ and DJ of all eight edges

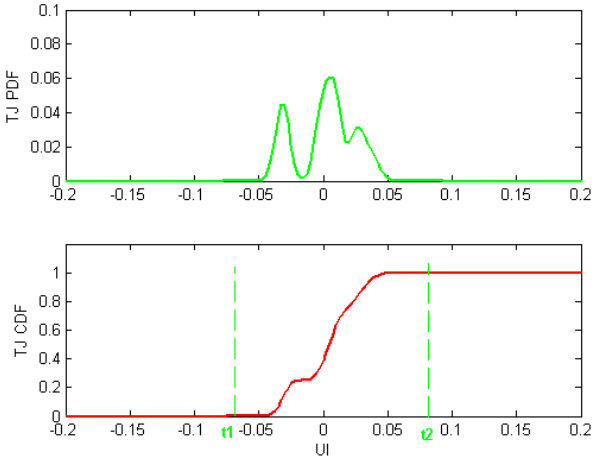

Figure 4. The PDF and CDF of the device TJ

With the above approach, we can characterize the Tx jitter performance in a few tens milliseconds. Once the Tx output waveform is captured, other Tx parameters, such as amplitude, rising/falling time can be extracted easily and quickly. It only takes around 100 milliseconds to validate or test the whole Tx.

3.2 Rx Testing Solution

For Rx validation and testing, jitter tolerance is the most challenging. Firstly we need test signals with controllable amounts of injected jitter. In [13], we propose a scheme to inject controllable amount of PJ though the AWG on ATE. By

controlling the amount of injected PJ, we can control the amount of injected TJ in the Rx test signals. As an example, Figure 5 shows the test signal calibration results.

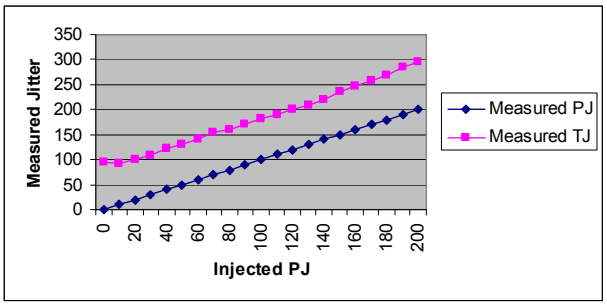

Figure 5. 3Gbps test signal calibration

The Rx jitter tolerance validation and testing are also the most time-consuming because the jitter tolerance is defined at 10^{-12} BER, which requires to run at least 10^{13} bits. In [10], we propose a scheme to accelerate the jitter tolerance qualification using a jitter tolerance extrapolation algorithm we proposed in [13]. We reduce the jitter tolerance qualification time from hours to seconds for characterization and a few tens of milliseconds for production [10].

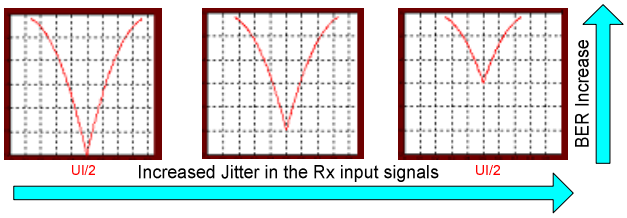

Figure 6. Receiver BER scan

In the jitter tolerance extrapolation, we perform a BER scan as shown in Figure 6 by sweeping the injected PJ in small increments to get several high BER levels, such as 10^{-9}, 10^{-8} and 10^{-7}. These high BER measurements can be obtained quickly. Because the Q factor and the BER are linked by the inverse error function [13], we can transfer the measured PJ vs. BER data into PJ vs. Q factor data. According to our jitter tolerance extrapolation algorithm that the relationship between the Q factor and the PJ is linear [13], we can do a Q factor linear fitting based on the PJ vs. Q factor data. Based on the fitting result, we can return to predict the PJ tolerance at low BER region through the error function. Finally we can translate the PJ tolerance into TJ tolerance according the calibration results shown in Figure 5.

3.3 Limitations

As discussed in section 3.1 and 3.2, the ATE-based solutions greatly speed-up the validation and testing of the Tx jitter and Rx jitter tolerance. However, there are two major limitations in applying the ATE-based approaches for HSSI qualifications.

The first limitation is in the number of ATE instruments available. For each HSSI, we need an AWG and a Digitizer in order to do parallel testing. Nowadays, there exist devices with a few tens or even more than 100 HSSIs. No ATE platform can accommodate so many AWGs and Digitizers. In addition, the AWGs and Digitizers that can handle Gigabit signals are very expensive. The AWG/Digitizer approach is also prohibitive from the cost point of view for devices with multiple HSSIs.

The second limitation is the lack of high-speed ATE instruments for the latest HSSI devices. Even with the fastest AWG on ATE -- AWG6000, we can only perform jitter tolerance testing up to 3Gbps. Even several ATE suppliers have provided production pin-card solutions up to 6Gbps, for higher rates, such as 8.5Gbps and 10Gbps, ATE instruments are not mature yet.

4. New Versatile Validation and Testing Scheme

To overcome these ATE limitations, we propose a new versatile scheme that provides more functionality and flexibility in validation, testing and debugging of HSSIs. Figure 7 shows the block diagram of the proposed scheme. This scheme accommodates the ATE approach discussed in section 3. More importantly, this scheme provides an external loopback approach that does not need any high-speed ATE instruments, such as the AWG or the digitizer. We use a phase delay line to inject controllable amount of jitter, a BERT to generate test patterns and detect bit errors, and high-speed relays to switch signal paths.

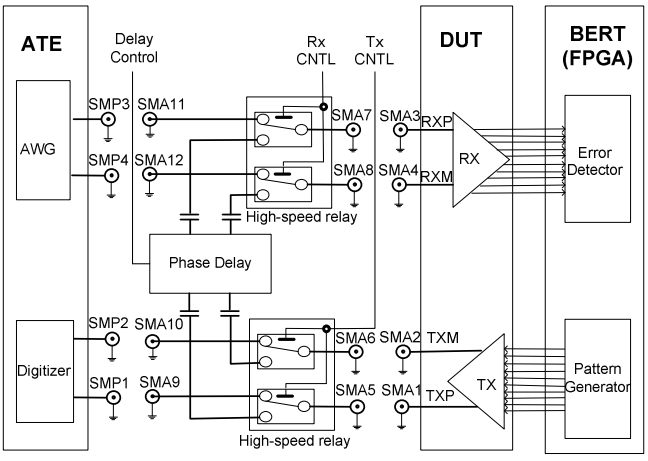

Figure 7. A versatile scheme for HSSI validation and test

4.1 Major Functions of the Scheme

In Figure 7, the phase delay line and relays are incorporated on a testing loadboard with Sub-Miniature type-A (SMA) connectors. By changing the cable connections and controlling

the relays, we can configure the proposed scheme to realize different functions for testing, validation and debugging of HSSIs.

4.1.1 Testing, Validation and Debugging on ATE

In this configuration, the Tx output is connected to the digitizer on the ATE and the output of the AWG on the ATE is connected to the Rx input. Cables are used to:

Connect SMA1 to SMP1, and SMA2 to SMP2
Connect SMA3 to SMP3, and SMA4 to SMP4

This is the test setup discussed in Section 3. The whole HSSI functionality and most design specifications can be qualified in less than one second in production [10] [11]. In this approach, we can accurately control the parameters in the Rx test signal, such as injected jitter, amplitude and test patterns. We can also capture the Tx output waveform and extract Tx parameters in a few tens milliseconds.

The ATE-based solution can also drastically facilitate the HSSI validation and debugging process. For example, if we find Rx jitter tolerance is too low, we can quickly debug it by measuring the jitter tolerance at different conditions, such as varying the amplitude of the Rx input signal, changing the equalizer and PLL settings, turning on/off the Tx block, sweeping supplying voltages, etc. All the measurement results are stored automatically and can be analyzed using ATE software. Using the ATE-based configuration, these kinds of debugging and validation procedures can be done more than 100 times faster than traditional bench approaches [10].

4.1.2 External Loopback with Jitter Injection

Loopback has been widely used to check the functionality of HSSIs. Traditional loopback approaches do not have the capability to qualify design parameters, such as Tx jitter and Rx jitter tolerance. Recently, there was some research that used DFT features or special modules to verify design parameters through external loopback [14] [15]. Our approach does not rely on any DFT features or special instruments; it only needs a few extra components that can fit into a testing loadboard. The approach is especially attractive for multiple-lane HSSIs or HSSIs with data rates above 6Gbps, where systematic ATE solutions are not available.

In the loopback configuration, Rx CNTL and Tx CNTL are set to low (the relays switch to lower throw). Delay Control can be connected to a digital channel on ATE or another resource to control injected jitter. Cables are used to:

Connect SMA1 to SMA5, and SMA2 to SMA6
Connect SMA3 to SMA7, and SMA4 to SMA8

Figure 8 shows the block diagram of the loopback configuration. In this approach, we inject controllable amount of jitter to the output of the Tx signal using a phase delay line and then loop the signal back to the input of the Rx. When the injected jitter is below a certain level, the Rx should be able to recover the transmitted data. Otherwise, the device is defective.

In the above loop-back testing scheme, we use an FPGA-based BERT. The Pattern Generator in the FPGA provides parallel data to the transmitter. The Error Detector compares the recovered data with the transmitted data and records errors. The FPGA may also need to provide some glue logic. We can still detect bit errors using the digital channels on ATE [13] or DFT features if they are available. The details of the BERT and the phase delay line are discussed in later.

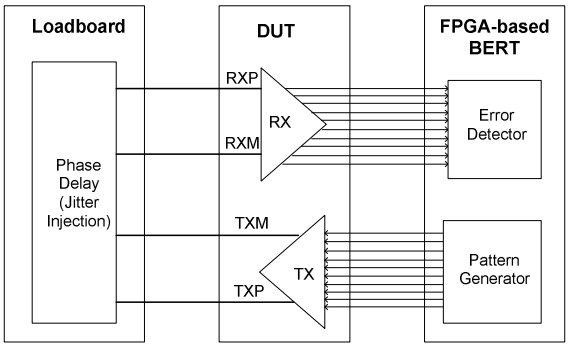

Figure 8. Loopback-based jitter testing

In the loopback approach, one thing we need to consider is the amount of jitter that needs to be injected. In [10], we set the amount of injected jitter using a jitter extrapolation algorithm based on calibrated test signals. In the loopback approach, the Tx jitter may vary from device to device. To minimize the possibility of skipping bad devices or failing good devices, we need to characterize devices in order to set a proper amount of injected jitter. We can achieve a more accurate testing by using a gold device or by measuring the Tx jitter and then setting the injected jitter accordingly. In either case, the test setup and the test program become complicated.

In the proposed versatile validation and testing scheme as shown in Figure 7, the AWG can still be used to provide other function tests, such as input level sensitivity and wakeup tests. In addition, the AWG and digitizer can be used to debug Rx and Tx independently. This is necessary if we need to trace down an issue because the loopback test can only qualify the whole HSSI – it does not provide detail information about the Rx or Tx if the test fails.

4.1.3 Other Configurations

1) External loopback without jitter injection: the DUT Tx output is directly connected to the input of the Rx. Cables are used to connect SMA1 to SMA3, and SMA2 to SMA4. This provides a quick way to check the functionality of the HSSI.

2) Characterize Relay and Delay line using the Digitizer:

Set Rx CNTL and Tx CNTL to low
Connect SMA1 to SMA5, and SMA2 to SMA6
Connect SMA7 to SMP1, and SMA8 to SMP2

3) Characterize relays only using the digitizer:
Set Rx CNTL and Tx CNTL to high
Connect SMA1 to SMA5, and SMA2 to SMA6
Connect SMA9 to SMP1, and SMA10 to SMP2

4) Characterize relays only or both the relay and the delay line using external instruments: instead of connecting to the Digitizer in 2) and 3), we can connect bench instruments to calibrate the relay and delay line.

In the above configurations, RF cables are used in order to maximize the configuration flexibility for validation, testing, debugging and calibration. This is very beneficial when we are in the debugging stage for a validation or test solution. Once the most suitable solution is finalized, many cables can be replaced by PCB traces.

4.2 FPGA-based Bit Error Detection

In [13], we use high-speed digital channels on ATE to compare the Rx recovered parallel data with expected data and then record bit errors. There are two limitations for this approach. First, it is very difficult to achieve synchronization between the DUT and ATE, as it depends on some special macros from the ATE vendor. In addition, it requires many high-speed digital channels, which are expensive and sometimes might not be available. Even though some BIST-based approaches almost do not need any ATE instruments, they need extra silicon area and design efforts to implement. In addition, we do not have the ability to change the test pattern once the design is finished.

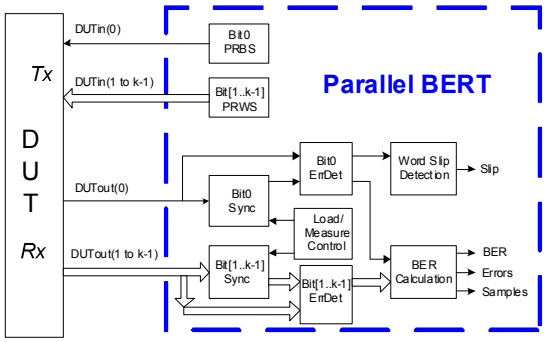

Figure 9. Block diagram of the BERT

To overcome these limitations, we explore FPGA-based approaches. To detect the bit errors in HSSIs, we can re-use the FPGA-based parallel BERT presented in [16]. Figure 9 shows the block diagram of the BERT. A Pseudo-Random Bit Sequence (PRBS) and a *k*-1 bits Pseudo-Random Word Sequence (PRWS) form a pattern generator, where *k* is the width of the parallel data of the HSSI; the other blocks form an error detector. The sequence for bit0 is used for synchronization and word slip detection [16].

The FPGA-based solution does not need any ATE instrument or any DFT features. It can be used to test almost any HSSI and the user has the freedom to set the test pattern. To demonstrate the BERT functionality, we used the BERT to test the HSSIs in the Altera Mercury FPGA device [3]. The FPGA is also used to implement the BERT. Figure 10 shows the test setup.

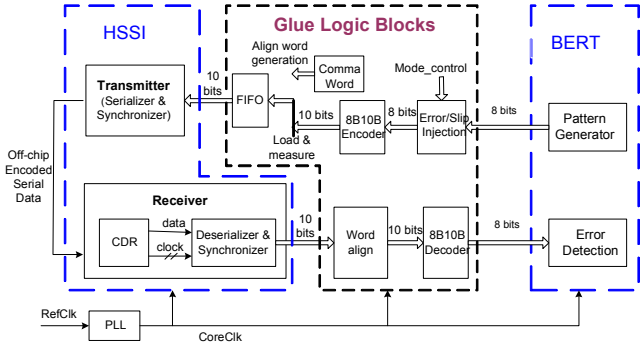

Figure 10: Testing Setup to verify BERT functionality

In the testing setup, the data width of the BERT is 8 bits. The glue logic is developed to interface the BERT and the HSSI. The Error/Slip Injection block inserts errors or word slips for the purpose of demonstrating the BERT functionalities. The 8B10B encoder encodes the 8-bit sequences to 10-bit sequences to ensure enough bit transitions in the serial data for date recovery [17]. A FIFO is used to ensure that there is always data ready for transmission after the testing begins. Comma words are inserted at the start of the testing for word alignment. The 8B10B Decoder recovers the 8-bit PRWSs sent by the BERT.

The testing setup is implemented in VHDL, targeting the EP1M120F484C7 device using Quartus II software. The synthesized results are downloaded onto an Altera Mercury Demo board. The outputs of the transmitter are connected to the inputs of the receiver by two SMA cables. We obtained zero BER both from simulations and from running real tests in the board when no error or slip was injected. The zero BER experiment results demonstrate the functional correctness of the HSSI and the BERT.

4.3 Delay-line based Jitter Injection

4.3.1 Jitter Injection

In [13], we present a scheme to inject controllable amount of jitter using the AWG. This approach can successfully test the jitter tolerance performance for data rate up to 3Gbps. For data rates above 3Gbps, we can not use the AWG-based jitter injection approach because currently the maximum sampling rate of commercially mature AWGs is limited to 6 Giga

samples per second. In this section, we present a phase delay line based jitter injection scheme. It can inject controllable amount of jitter for data rates up to 12.5 Gbps. In addition, it only needs a few components, which can easily fit into a testing loadboard.

A delay line is a device where the input signal reaches the output of the device after a known period of time has elapsed. Delay lines or phase delay lines have been widely used in electronics and derivative fields such as telecommunications and testing. Early delay lines were implemented with a RC-based ramp generator and a comparator that transitioned the delay line output when the ramp generator reached a certain voltage level. More sophisticated delay lines then were developed using a voltage-controlled delay line (VCDL) [18].

Today, ultra wideband phase delay lines have been developed using the InGap or InP Heterostructure Bipolar Transistor (HBT). InGap HBT is a proven reliable technology that has been widely used in large volume wireless applications. It exhibits characteristics such as high cut off frequency, high linearity and temperature stability, suitable for ultra wideband device design. InP HBT has the highest cutoff frequency among the IIIV available technologies [19].

One unique product in the market is the phase delay line iT4036. It was developed by GigOptix using high speed HBT Emitter Coupled Logic (ECL) topology realized in InP [20]. The phase delay line is ultra-wideband, operating at speeds up to 12.5Gbps for data signals and 11.7GHz for clock signals. It can provide tunable phase delay up to 120ps in a single device [20]. Figure 11 shows the relation between the delay control signal and the phase delay. The delay control bandwidth is up to 1GHz. Its output amplitude is 400mVpp in single-ended mode and 800mVpp in differential mode. Inspired by these unique features, we proposed to use this product for jitter injection and asked GigOptix to provide us an evaluation board in 2007 to do jitter inject experiments for HSSI testing. Later on GigOptix also suggested using iT4036 for jitter injection [22].

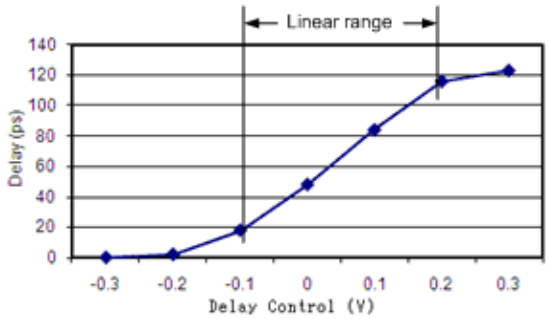

Figure 11. Delay vs. delay control voltage [20]

In traditional delay line applications, a constant voltage is applied to the delay control to generate a constant phase delay. Because of its unique performance, we use the iT4036 to inject jitter by applying a clock signal to the delay control input. The delay for each data edge may be different, depending on the amplitude of the control signal at the instance of each edge. This is equivalent to injecting deterministic jitter to the input signal. Because the delay control bandwidth is up to 1GHz, we can inject DJ with frequencies up to a few hundreds MHz, which is suitable for the HSSI jitter testing and characterization. As shown in Figure 11, there is a linear region where the relationship between the delay control voltage and the phase delay is linear, so we can conveniently control the amount of injected DJ by adjusting the amplitude of the clock signal to the delay control input.

4.3.2 Experimental Results

We demonstrate the delay line based jitter injection technique on an iT4036 evaluation board. In our experiments, we connect the Tx output of a 6 Gbps HSSI to the input of the delay line through cables and then capture the output of the delay line using the digitizer available on ATE. The delay control signal is provided by a digital channel on the ATE. The digital channel sources a clock signal and the injected jitter can be adjusted by changing the *Voh* and *Vol* levels of the digital channel. The injected jitter frequency can also be adjusted according to test requirements.

Figure 12 plots the captured waveform of a 6 Gbps NRZ data signal from the output of the delay line with 400 samples each bit. As we can see the output signal is very clear. The rise time and falling time are very short as specified in the delay line specification [20]. The plot shows the output amplitude around 900mV, which is also close to the delay line specification 800mVpp.

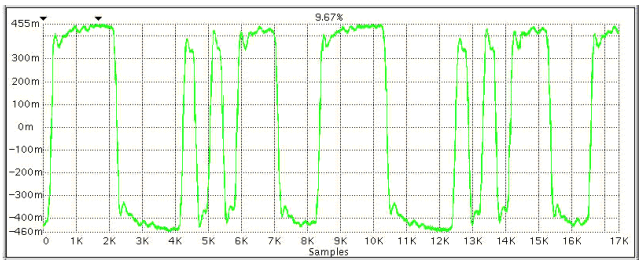

Figure 12. The Tx output waveform after the delay line

Figure 13 plots the extracted DJ profile from the captured data signal using the Tx jitter extraction scheme proposed in [11]. As we can see, the DJ profile is close to a square waveform, which is the profile of the jitter source we injected to the data signal. There is minor distortion at the second half of the high level, which is most likely caused by the intrinsic PJ of the device. However, we are not concerned about the detail DJ profile because the HSSI specification only defines the peak-to-peak value. In the 400-bit 6Gbps data signal, the DJ profile repeats twice. Therefore, the DJ dominant frequency is

30MHz, which correlates the injected jitter source – a 30MHz clock signal.

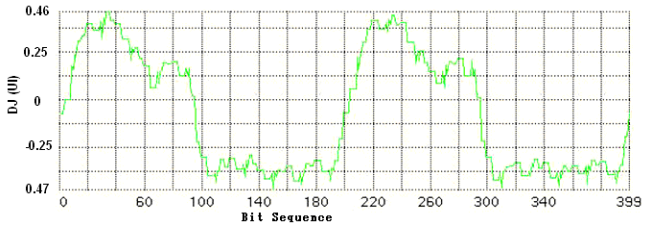

Figure 13. Extracted DJ profile

Experiments show a linear relationship between the DJ and the amplitude of the delay control signal. We can calibrate the DJ and TJ of the test signal at different delay control amplitudes using either bench equipment or the digitizer on ATE. According to the calibrated test signal, we then can set the injected jitter level for jitter tolerance validation and testing using the same approach as discussed in [10].

5. Conclusions

We have proposed a versatile scheme for HSSI post-silicon validation and debugging, as well as for production testing. Depending on applications and cost goals, we can use either the ATE with high-speed instruments or the external loopback approach without ATE. The scheme can be used to test HSSIs with data rates up to 12.5 Gbps. Calibrations and debugging of the components used in the scheme have also been considered.

6. Acknowledgements

The authors would like to thank the following for providing technical support and help: Bernhard Lanchisky, Yi Cai and Liming Fang from LSI Corporation, Carlo DiGiovanni, Jim Emerich, Jwo Cheng and Terry Cummings from GigOpix.

References

[1] P. Noel, F. Zarkeshvari and T. Kwasniewski, "Recent advances in high-speed serial I/O trends, standards and techniques", 18th Canadian Conference on Electrical and Computer Engineer, CCECE05, May 2005

[2] ITRS. *The International Technology Roadmap for* Semiconductors, 2007 Edition

[3] Altera *Corporation*. *Mercury Programmable Logic* Device *Family Data Sheet.* San Jose, California, January 2003.

[4] Altera Corporation. Stratix IV Device Datasheet, http://www.altera.com/literature/hb/stratix-iv/ stx4_5v4_01.pdf, December, 2008

[5] Xilinx, Inc. Introducing Virtex-6 and Spartan-6 FPGA *Families*, http://www.xilinx.com/products/v6s6.htm

[6] A. Hajimiri and T. H. Lee, "The Design of Low Noise Oscillators", Kluwer Academic Pulbishers, 1999

[7] T. Miyazaki, M. Hashimoto and H. Onodera, "A Performance prediction of clock generation PLLs: A ring oscillator based PLL and an LC Oscillator based PLL", IEICE Trans. on Electronics 2005 E88-C(3): 437-444

[8] Altera Corporation. "Innovating with a Full Spectrum of 40-nm FPGAs and ASICs with Transceivers", whitepaper

[9] P. Patra, "On the Cusp of a Validation Wall," *IEEE Design & Test*, vol. 24, n. 2, Mar.-Apr. 2007.

[10] Y. Fan and Z. Zilic "Accelerating Jitter Tolerance Qualification for High Speed Serial Interfaces", *Proceedings of the 10th International Symposium on Quality Electronic Design*, ISQED'09, March. 2009

[11] Y. Fan, Y. Cai and Z. Zilic, "A High Accuracy, High Throughput Jitter Test Solution on ATE for 3 Gbps and 6 Gbps Serial-ATA", *Proceedings of IEEE International Test Conference, ITC'07, Oct. 2007*

[12] http://www.teradyne.com

[13] Y. Fan, Y. Cai, L. Fang, A. Verma, B. Burcanowski, Z. Zilic and S. Kumar, "An Accelerated Jitter Tolerance Test Technique on ATE fro 1.5GG/s and 3GB/s Serial-ATA", *Proceedings of IEEE International Test Conference, ITC'06, Oct. 2006*

[14] A. Meixner, A. Kakizawa, B. Provost, S. Bedwani, "External Loopback Testing Experience with High Speed Serial Interfances", *Proceedings of IEEE International Test Conference, ITC'08,* Oct. 2008

[15] W. Fritzsche, A. Haque, "Low cost testing of multi-GBit device pins with ATE assisted loopback instrument", *Proceedings of IEEE International Test Conference, ITC'08,* Oct. 2008

[16] Y. Fan and Z. Zilic, "Bit Error Rate Testing of Communication Interfaces", *IEEE Transactions on Instrumentation and Measurements*, Vol. 57, No. 5, May 2008, pp. 897-906.

[17] Altera Corporation, *8B10B Encoder/Decoder MegaCore Function User Guide*.

[18] http://www.maxim-ic.com/appnotes.cfm/an_pk/209, "How delay lines work", Application notes

[19]"Semiconductior Technologies for Optical Interconnection, from Ultra Long Haul to Very Short Reach"http://www.gigoptix.com/pdf/whitepapers/Optical_Interconnection_rev6.pdf, white paper by GigOptix

[20] "iT4036 Wideband Phase Delay Datasheet", hikari-trading.com/opt/gigoptix/file/it4036/it4036_data.pdf

[21] Y. Cai, B. Laquai, and K. Luehman, "Titter Testing for Gigabit Serial Communication Transceivers", *IEEE Design & Test of Computers*, Vol. 19, Issue 1, Jan, 2002

[22] "GigOptix's Broadband Phase Delay Opens Numerous Application Opportunities," Press release, April 28, 2008, available at http://www.gigoptix.com/newsroom/release/ 2008_04_01_archive.php