Methodologies and Challenges in Testing of High-Speed I/O Interfaces

Srikanth Alaparthi, Prasanth Jampani
Texas A&M University, College Station, TX
alaparthi@neo.tamu.edu , jampani@neo.tamu.edu

Abstract

With recent advancements in technology, delivering data at higher speeds plays a crucial role in the performance of the system. Faster serial buses introduce major test challenges as traditional functional testing and simple design-for test (DFT) techniques no longer work. At multiple Gbps data rates, jitter, noise and bit error rate (JNB) testing becomes mandatory to ensure the interoperability of the I/O link. This paper aims at describing the challenges and concepts involved in the testing of high speed I/O interfaces. It also describes some common schemes used like AC Loop back, IBIST and BIST for DDR in detail.

1. Introduction

I/O or input/output refers to the communication between a processing unit, and the outside world – possibly a human, or another processing system. As processor speed continues to increase, I/O has become the bottleneck for system level performance and the system designers have begun increasing I/O performance. This requires a change in the test methodology and in the subsequent DFT techniques. Examples of such I/O performance changes include Intel's changing its processors' front-side bus from common-clock to source-synchronous (SS) signaling and increasing their bus transfer rate from less than 100 MHz to 800 mega transfers/second (1 MT/s = 1 Mbyte/s/pin). On the chipset side, Intel has upgraded its universal serial bus from 48 Mbps to 400 Mbps and has transitioned to the Serial Advanced Technology Attachment (SATA) standard at a 1.25-Gbps data rate. Also, PCI Express at 2.5 Gbps is set to replace the 10-year-old peripheral component interface (PCI) standard.

Today's high performance SoC designs incorporate a large variety of high speed I/O buses and protocols. Recent Intel based PC chipset architecture with a Memory Bridge and an I/O bridge is shown in Figure1 to exemplify mixed I/O types. High speed serial I/O technologies are being rapidly adopted in data storage, telecommunications and personal computer applications. This is because they offer a reliable, low power and high data bandwidth capability using low-cost backplane and connector technologies. Manufacturers of semiconductors used in the high-end computing (advanced microprocessors) and consumer (graphics and gaming chipsets) applications are using these high-speed serial bus interfaces like PCI Express and HyperTransport to deliver data at rates up to 6.4 Gbps. Significant test challenges are emerging as multiple SerDes channels with data rates of between 1 Gbps and beyond 10 Gbps are being integrated into SoC's and FPGA's.

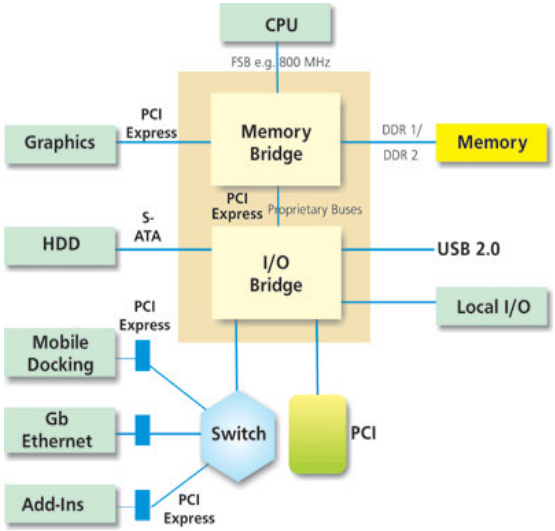

Figure1: Example of a recent Intel-Based PC Chipset Architecture with a memory bridge and an I/O Bridge

IC manufacturers are finding it increasingly expensive and time-consuming to guarantee all of the critical signal integrity parameters of these high speed I/O devices. With frequencies in the multiple-gigabit-

per-second range, the signals can no longer be treated as purely digital. Signal-integrity concerns such as timing jitter and level noise need to be considered to maintain adequate fault coverage and to meet required quality levels. Testing of Bit Error Rate (BER) in the order of 10^{-12} is not practical and so eye diagram plays a critical role in estimating the performance of the I/O link.

The important consideration in testing high speed interfaces is to deliver the data at those speeds. Older testing flows on high-end automated test equipment (ATE) platforms can sometimes offer the most thorough test coverage, but have the disadvantages of higher cost and test programming complexity. Moreover, cost per pin rises with bus speed in high-speed data applications, making traditional functional testing even less viable. The testers have thus turned to loopback techniques in which we use the device to source the test data and receive it back into the device for recognition. This approach gives comprehensive, cost optimized testing of today's high-speed I/O interfaces. Also, the standard BIST approach suffers from its inability to perform mission-mode parametric tests, which are becoming more important with the integration of high-speed I/O interfaces. To manage the cost and the efficiency of-test challenges, new design-for-test (DFT) techniques and alternative test methods are being explored. In particular, AC scan and built-in self-test (BIST)/loopback techniques are increasingly being used to improve fault coverage on timing-related failures for the high-speed portions of the devices.

The rest of the paper is organized as follows. Section 2 provides the basics of the high speed architectures. Section 3 overviews the concept of the Eye diagram. Section 4 details the AC loopback testing approach. Section 5 and Section 6 presents IBIST and BIST for DDR testing methodologies. Section 7 draws the conclusions.

2. High speed I/O architectures

High speed I/O architectures are largely driven by the demand of delivering higher data rates. The important considerations while designing them include clock generation, distribution and test methodologies. The two main kinds of architectures are as follows:

- *Global clock architecture (GC):* A global clock is sent to both the receiver and transmitter. It is limited by the skew and propagation delay for clock and data.
- *Source synchronous architecture (SS):* In this approach, the transmitter sends a strobe signal along with a data signal which is used at the receiving end to capture data. One advantage of this signaling architecture is that common- mode jitter (variations that occur simultaneously in both the signal and the strobe) doesn't impact the interface's performance; only differential jitter (variations that affect the data or strobe differently in a given cycle) affects. There are mainly two physical data path loops, one for the data signal and one for the strobe and both are required for the buffers operation. The data is launched from the driver on a falling edge of Clk and, mid-cycle; the Strobe is launched from its driver based on the rising edge of the same Clk. At the end of the data loop, the output arrives at the receiver input latch and is captured in the latch when the strobe arrives. Figure4 shows SS I/O interface.

3. Eye diagram

In the recent past, short distance chip-to-chip I/O links essentially have adopted serial communication architectures. At multiple-Gbps data rates, a digital waveform appears to be an analog waveform at the receiver input because of the frequency dependent lossy property of the channel. As the data rate keeps increasing, the unit interval(UI) – period during which a digital bit can exist – becomes shorter and shorter, and, as such, the system will be more susceptible to failures resulting from jitter and noise. Because of this failure mechanism, jitter, noise and BER, (JNB) testing becomes necessary for multiple Gbps interfaces.

The bandwidth achievable by a signaling system is limited by attenuation, interference, and jitter. These factors are illustrated in the conceptual *eye diagram* of Figure 2. Constructed by *folding* the data waveform into a symbol time, an eye diagram shows variations of signal amplitude (voltage noise) and timing (jitter) across bit cells. The rectangle in the middle represents the *eye opening*, which must be wider than the receiver jitter plus aperture and taller than the receiver sensitivity. The Eye diagram of the TX and the RX, is electrical Signal Specification of an interface that defines the allowable ranges of timing and voltage output/input of the TX/RX. Figure1 shows the illustration of determining the interconnect jitter and loss based on the eye diagram of the TX and RX. Figure 3 shows eye diagram that is seen in an oscilloscope. Convolution of many bits superimposed looks like this eye shape. The width of the eye specifies the jitter allowed. Better equalization techniques help in widening the eye. Similarly the height of the high signifies the bit strength.

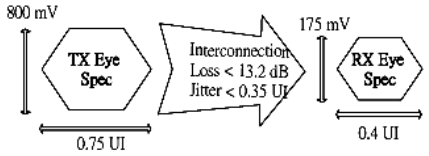

Figure2: Transmitter and Receiver EYE diagrams and Interconnection loss (Shown in [3], PCI-E Interface)

Eye diagram plays key role in testing of the I/O interfaces or pads. It helps in evaluating the JNB response of the system. A good output will have a wide-open eye with large eye-openings in both timing and amplitude axes or equivalently small timing jitter and amplitude noise.

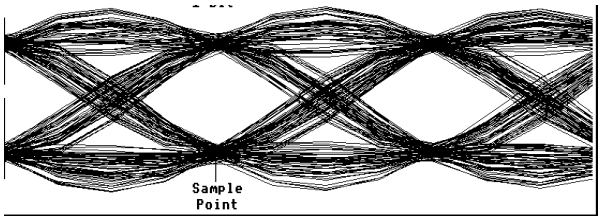

Figure3: Typical EYE Diagram Observed in Oscilloscope

4. AC Loop back testing

I/O Loop back is the method that involves applying a transition fault test methodology to I/O Circuitry. By tying an output to an input, the output data is launched and latched back into the input buffer on the following clock. Figure 4 illustrates a simple I/O loop back scheme for SS Interface. As most signal pads are I/O in nature, the I/O loopback methodology is convenient. The limitation to this method is that, the delay path is tested with clock; the delay cannot be characterized without overstressing the other peripheral circuits.

In [1], Intel introduced a testing scheme that requires only an accurate clock source; it does not require probing individual signals. The method is called AC I/O Loop back as it relies on a loop in the I/O buffer and also it guarantees the AC timing parameters. The complete circuit diagram that is used in Pentium 4 for AC Loop back is explained in [1]. Three additions to the simple loop back scheme are necessary to support AC loopback: Stimulating the loops with controlled data patterns (toggling pattern is preferred), stressing the loop and detecting when the loop is passing and failing.

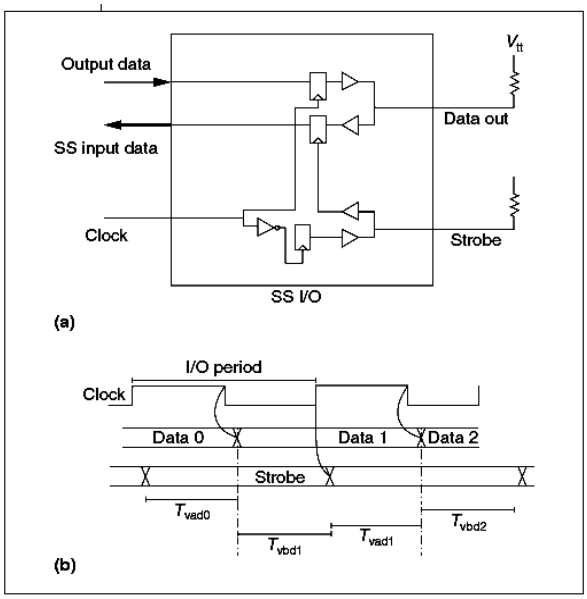

Figure4: SS I/O Interface: (a) I/O loop back (b) Associated timing diagram (Taken from [1])

As we can see from Figure 5, the Pentium4 AC I/O loop back DFT implementation uses two strobe signals; Strobe1 captures even data bits and Strobe2 captures odd data ones. There is a TPG to generate 2-bit patterns and a multiplexer is used to select between data from core and test bits from TPG. The jitter that is introduced by this multiplexer is compensated by its counterpart in the clock path. From the eye diagram that is observed in the oscilloscope, the first fail (FF) point and all fail (AF) point are noted. FF is the minimum delay of the strobe and AF is the maximum delay of the strobe. The Pentium4 stress mechanism included a series of delay elements that could be sequentially added into the strobe clock generation path. So, in their simplest path AF and FF represent the number of added delay elements required for each measurement. AC I/O loopback FF measurements are observed to be worse when compared to functional measurements. This is because of accumulating the delay of different elements.

The main challenge involved in AC I/O loopback scheme is to reduce the test time. One of the ways to accomplish this is to implement the BIST logic on the die. The BIST circuitry can then perform the loop back parameter comparison and simply reports pass or fail. Another way to reduce time is by reducing the time for loading the patterns. This can be done by having default test pattern, like toggling pattern, and using an on chip pattern generation such as IBIST.

Loopback BIST by itself also has problems. It will only provide a pass/fail indication for that particular match-up of transmitter and receiver, operating on a near-ideal loopback connection at a particular voltage and temperature point.

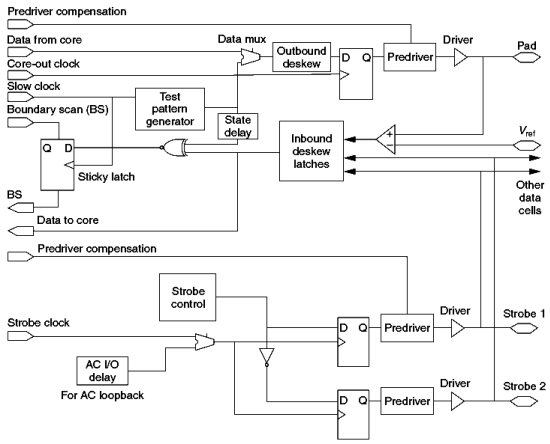

Figure5: AC I/O Loopback Pentium4 (Taken from [1])

5. IBIST

Interconnect built-in self test (IBIST) tests the interface at a high data rate or at-speed with worst-cast patterns. This technology is a design validation and test architecture embedded into many of Intel's processors to enable chip-to-chip interconnect testing and design validation of high-speed buses on a printed-circuit board. IBIST leverages on the boundary-scan IEEE 1149.1 specification as the hardware and software communication methodology for accessing and controlling its embedded on-chip capabilities. IBIST pattern generator along with AC loopback, makes the testing more self contained. It can support more complex patterns, especially those for testing ISI, crosstalk, SSO, and so on. IBIST is based on the premise that high-performance interfaces have to be tested at high speed.

Figure 6 shows an IBIST example with two components as applied for high-speed serial testing at the board level. The two sides of the I/O interface are named master and slave. On the transmitter side, IBIST consists of a programmable pattern generator behind the Tx driver or transmitter of the I/O interface in the high-speed serial link. On the receiver side, IBIST consists of logic to route the received data (Rx) back to their own transmitter as well as error checking logic (XOR gates), which can check the pattern transmitted versus the received pattern. To further reduce the circuit requirement, the slave can simply implement the internal loopback circuit to support only the bounced back mode. With this reduced logic, it cannot independently send a pattern but will support the bounced back mode only. The independent pattern generation helps in testing the interface independent of core logic. The master will have the full control and drives high speed pattern to the slave. Te slave resends all data that it has received immediately through its transmitter. It is bouncing back mode. The master receiver receives it and compares it original transmitted data. Error checking is done in symbol-by-symbol basis. Control registers keep track of the first error detection, transmission of data patterns.

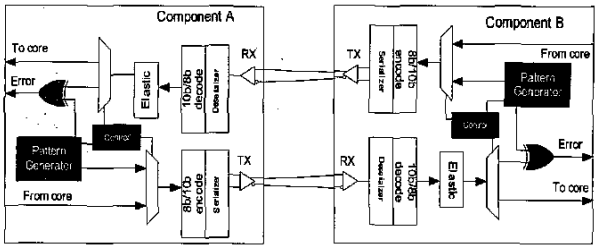

Figure6: IBIST architecture for a differential bus (Taken from [2])

IBIST architectures for PCI Express are explained in [2]. The PCI Express (PCIe) is a point to point serial interconnection that will replace eventually the industry standard PCI and PCIX bus topologies. PCIe uses an isolated path between two components thereby avoiding the shared bus topologies. PCIe has 3 protocol layers: Transaction, Data link and Physical layer. IBIST functionally is fully decoupled from the core and resides in the PCIe physical layer. Since IBIST operates independently, it need not follow the standard PCIe protocol for other two layers. IBIST utilizes the same normal operation component interconnect and timing paths. Data test pattern injection and capture logic are inside the IBIST loop. Dedicated IBIST blocks enable independent testing of the individual lanes within a link. An example of IBIST topology is shown in Figure 6.

The IBIST architecture offers two modes of operation: FIXED and OPEN. In FIXED mode, IBIST is simply a push-button. No programming from the test host is allowed. Whereas OPEN mode is designed for more interactive testing required in test development and debug. It provides the user with a complete complement of debug features allowing for custom test pattern definition and special execution options. FIXED primarily is useful in high volume

manufacturing. The pattern generator in IBIST, is implemented as a state machine for constructing meaningful pattern permutations. It supports both the modes. In FIXED mode, the user is only required to set the start bit in the control register. In OPEN mode, the user can specify the contents of the pattern buffer, loop count, skip insertion count, delay symbol, usage etc. In [2], both these modes are discussed in detail.

IBIST uses PCIe "Skip Ordered Sequences" (SOS) and "Electrical Idle Ordered Sequences"(EIOS) to frame IBIST test data. These minimal sequences of operations are needed for the basic testing methodology. By default, IBIST operation is loop back enabled. Programmatic configuration of the IBIST test slave differentiates the IBIST supported loopback mode from other forms of external loopback operation within PCIe standard.

Though IBIST is mainly explained for PCIe, it is easily extendable to other types of I/Os. IBIST methodology addresses the expanded fault spectrum associated with next generation IO topologies and is one facet of Intel's test methodology for the future. IBIST has become a standard Intel high performance IO test and validation feature.

5. BIST for DDR

Testing of memory arrays is very critical for the success of an SOC chip as the embedded memory will occupy 94% of the whole SOC area [4]. Little research has been done on the widely-used DDR SDRAM where in it allows transferring two data words per execution of the memory cycle. There are mainly two challenges for DDR SDRAM testing [4]. First one is the high data rate architecture. In the current approach, the pipeline strategy is used together with several design techniques to achieve high speed requirement. The second one is the variable execution cycles for the commands in DDR SDRAM. To provide the capability of the at-speed technique, the BIST also has to transfer two data words per memory cycle which means that BIST must run at twice the clock rate as that of the DDR.

The access data is available after the CAS latency (CL) for READ and WRITE operation. The number of accessed addresses depends on the burst length (BL). The DDR SDRAM BIST must function correctly for a variety of READ and WRITE access times [4]. Besides this it also has to take command to command latencies into account [4] which means that a subsequent command cannot be issued until a specified latency after the issue of its previous command.

The BIST architecture is shown in Figure 7. It mainly consists of six functional blocks. Test instruction generator (TIG) is used to generate the March elements of the required test algorithms. Test vector generator (TVG) translates the March elements to intermediate codes which will be efficiently decoded into the physical signals for the DDR SDRAM by the next stage. TVG solves the problem of different execution cycles of a command in a different execution mode. Address generator (AG) generated the address signals for the memory. The comparator (CMP) block compares the responses from the memory and the golden patterns from PSG block to give out pass/fail. Extra pipeline registers are inserted into both TVG and TIG.

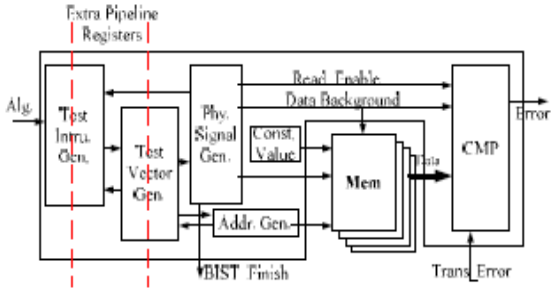

Figure7: BIST for DDR architecture (Taken from [4])

In operation encoding two different schemes are followed [4]. "Command pairing" in which two commands are co-considered by TVG which has the advantage of single Hardware component in the PSG to generate different test sequences for command pairs with different command-command latencies. Also we can generate more compact test sequence. The second scheme is to generate control and data signals separately. This gives the advantage of WRITE and READ operations to be overlapped as shown in the Figure 8. Each March element will be first converted to an operation code called pattern code [4] which is later encoded into a single or command pairs.

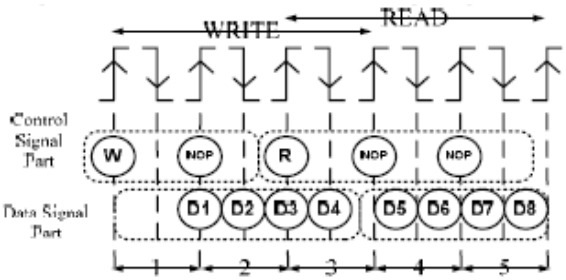

Figure8: Generation of separate control and data signals (Taken from [4])

6. Conclusions

In this paper we have talked about different types of I/O interfaces and high-speed I/O architectures in detail. The challenges involved in testing of high-speed I/O interfaces are described and some of the standard testing techniques are introduced. AC loop back used for Intel Pentium4 processor, IBIST for PCI Express(PCIe) architecture and BIST for high-speed DDR are analyzed and presented in detail.

7. References

[1] T.M. Mak, Mike Tripp, and Anne Meixner, "Testing Gbps Interfaces without a Gigahertz Tester", *Design & test of Computers*, *IEEE*, 2004, pp. 278-286.

[2] J.J. Nejedlo, "IBIST (Interconnect BuiIt-in-Self-Test) Architecture and Methodology for PCI Express", *Proc. Int'l Test Conference* (ITC 03), vol.2, IEEE Press, 2003, pp114-122.

[3] http://www.amsat.org/amsat/articles/g3ruh/109.html

[4] Sheng-chih shen, Hung-Ming Hsu, Yi-Wei and Kuen-Jong Lee, "A High Speed BIST Architecture for DDR-SDRAM Testing", *Proc. IEEE International Workshop on Memory Technology, Design, and Testing* (MTDT'05), 2005.

[5] Dragan Topisirovic, "Advances in VLSI Testing at MultiGb per Second Rates", *Serbian Journal of Electrical Engineering*, Vol.2, No. 1, May 2005, pp. 43-55.

[6] T.M. Mak, Angela Krstic, Kwang-Ting (Tim) Cheng, and Li-C. Wang, "New Challenges in Delay Testing of Nanometer, Multigigahertz Designs", *Deep-Submicron IC Testing/ IEEE CS and the IEEE CASS*, May - June 2004.

[7] C. F. Wu, C. T. Huang, K. L. Cheng, and C. W. Wu, "Fault Simulation and Test Algorithm Generation for Random Access Memories," *IEEE Trans. on CAD*, Vol. 21, No. 4, pp. 480-490, April 2002.

[8] S.Y. Huang, D.-M. Kwai, and C. Huang, "A High-Speed Architecture For At-Speed DRAM Testing," *Journal of Chinese Institute of Electrical Engineering*, Vol. 8, No. 4, pp. 387-394, Nov. 2001.

[9] Mike Tripp, et. Al., "Elimination of Traditional Functional Testing of Interface Timings at Intel", *Proc. Int. Test Conf* 2003, pp .1014-1022

[10]http://www.eeproductcenter.com/article/printableArticle.jhtml?printable=true&articleID=18902491&printable=true

[11]http://www.eetimes.com/article/showArticle.jhtml?articleId=16502180

[12] L.T. Wang, C.E. Stroud and N.A.Touba, "*System on chip test architectures*", Morgan Kaufmann Publishers, USA, 2008.