

2 Gbps SerDes Design Based on IBM Cu-11 (130nm) Standard Cell Technology

Rashed Zafar Bhatti
EE- Systems Department
University of Southern California
Marina del Rey, CA-90292, USA
bhatti@isi.edu

Monty Denneau
IBM T.J. Watson Research Center
Yorktown Heights, NY 10598
denneau@us.ibm.com

Jeff Draper
Information Sciences Institute
University of Southern California
Marina del Rey, CA-90292, USA
draper@isi.edu

ABSTRACT

This paper introduces a standard cell based design for a Serializer and Deserializer (SerDes) communication link. The proposed design is area, power and design time efficient as compared to conventional SerDes Designs, making it very attractive for modest budget multi-core and multi-processor ASICs with wide communication buses that are difficult to accommodate within the pin count of commonly available packaging. The design employs a “Statistical Random Sampling Technique” to observe and adjust the synchronization and serialization signals at start up rather than using a resource-heavy PLL or DLL based frequency multiplier/synthesizer and clock data recovery circuits. The serialization and deserialization logic is based on standard cell technology that makes the design highly portable. Multiple serial lines are bundled with a strobe that is used as a reference signal for deserialization. Data-to-strobe timing skew is compensated by adjusting the launch times of strobe and data symbols at the sender side. The edges of the strobe are set within the eye of data symbols to have maximum timing margin, which makes the design inherently tolerant of jitter. Power consumption of the proposed SerDes design is 30 mW per serial link targeted to IBM Cu-11(130 nm) Technology, nearly a 2.5x improvement over the conventional design with a 60% less area requirement.

Categories and Subject Descriptors

B.4.3 [Input/Output and Data Communications]: Interconnections (Subsystems) – *Standard Cell based Serializer and Deserializer circuits for high speed signaling.*

General Terms

Measurement, Design and Theory.

Keywords

Duty Cycle Correction (DCC), Phase detection, SerDes, LVDS, CML driver, PLL, DLL, CDR, jitter and skew compensation.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GLSVLSI'06, April 30–May 2, 2006, Philadelphia, Pennsylvania, USA.
Copyright 2006 ACM 1-59593-347-6/06/0004...\$5.00.

1. INTRODUCTION

Over the past few decades processor designers have been exploiting mainly technology frequency scaling for performance gain, with architectural improvements providing only modest performance adders. However, frequency scaling is reaching its physical limits, and architectures have matured so greatly that any further revolutionary improvement is unlikely. To push forward the conventional performance scaling trend computer engineers are now moving towards a parallel processing paradigm by putting more and more processing cores on a single chip along with different kinds of on-chip memory architectures. DIVA's PIM chips[3], Intel's Pentium D, AMD's Dual Opteron, Sun's Niagara and IBM's Cell are just a few of the many examples of such architectures. Multiple cores on a single chip essentially demand wider interfacing buses that produce a considerable mass of interconnection wires on the chip and require a high number of package pins to connect the silicon die with the rest of the board level circuit. This problem is often tackled by serializing multiple signals to a high-speed data channel at the source side and deserializing them at the target side. A dedicated pair of a Serializer and Deserializer (SerDes) is used for this purpose. SerDes technology has existed for some time, but, until recently, has had little visibility as a true interface in commonly fabricated ASIC's. The reasons that SerDes has not enjoyed widespread adoption are (1) design time inefficiency, (2) high power requirements, (3) channel bit error rate (BER) and (4) silicon area cost. The recent surge in low voltage differential signaling (LVDS) technology and its common-mode versatility has partially solved the problem of channel bandwidth and BER problem [11], but still all known state-of-the-art SerDes designs like IBM High Speed SerDes (HSS)[10] are complex, area-intensive and power hungry which make them not suitable for medium and low-budget projects.

In this paper we introduce a SerDes design that does not employ resource heavy components like conventional SerDes designs. The main emphasis is to speed up the design time and portability of the design across technologies while keeping the overall costs as low as possible. The rest of the paper is organized as following: section 2 is a brief survey of the conventional SerDes designs; Section 3 explains the ideological and theoretical basis of the proposed design. Section 4 shows implementation details of the design. Section 5 briefly analyzes the proposed design and section 6 concludes the paper.

2. CONVENTIONAL SERDES DESIGNS

Figure 1 illustrates the main components of conventional SerDes systems [4]. The serializer on the sender side includes a clock multiplier or frequency synthesizer and a parallel-to-serial multiplexer circuit with an I/O driver at the output, whereas a clock recovery and serial-to-parallel conversion circuits with receiver are the main components of the deserializer at the receiving end. A clock multiplier is one of the very important components of SerDes system. In conventional high-speed SerDes systems like [8], [10], 0 and [5], to achieve maximum bandwidth and pack large number of parallel data lines to one serial link, low-jitter fast-locking PLL based clock multipliers/frequency synthesizers are used to drive the parallel to serial converters. Similarly a clock recovery circuit, on the receiver side, employs sophisticated PLLs or DLLs to recover the clock on the receiver end to capture and deserialize data back to a parallel form. I/O drivers and receivers are used to provide a high bandwidth communication channel between the sender and receiver. Different types of driver and receiver pairs are used depending upon the desired operating parameters. The two most commonly used kinds are low voltage differential signaling (LVDS) and current mode logic (CML) drivers. LVDS drivers generally operate at 100-450 mV swing with DC-coupled signal and provide a speed between 155 Mb/s and 2 Gb/s. CML drivers operate at a higher voltage swing with DC or AC coupled signals and typically provide bandwidth above 2.5 Gb/s[11]. To further enhance the communication channel performance advanced techniques like feed forward equalization (FFE) and pre-emphasis are used on the transmitter side where as on the receiver side equalization is achieved through either passive equalizers, receiver peaking pre-amplifiers or decision feedback equalizers (DFE) [8]. IBM's state-of-the-art 6.4GB/s CMOS SerDes core [10] employs CML drivers and receivers with a 4-tap FFE and 5-tap DFE for channel equalization. Given that frequent usage of LVDS and CML drivers, component libraries for VLSI technologies (e.g. IBM Cu-11, Cu-08 etc) typically provide a reasonable choice of LVDS and CML drivers and receivers.

3. SUBSTRATUM OF THE DESIGN

Many well characterized high-performance IP cores of SerDes systems that are commercially available employ sophisticated components like low-jitter, fast-locking PLLs or DLLs for clock

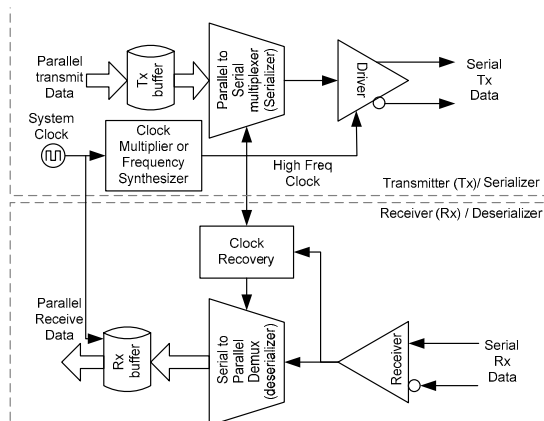


Figure 1. A Conventional SerDes System.

multiplication or frequency synthesis and clock recovery circuits to achieve high-quality communication. However, these characteristics make the overall design extremely resource heavy in terms of silicon area and power consumption. Moreover these IP's are still way too expensive for medium budget projects. On the other hand custom designing a SerDes system with a conventional approach is not design time efficient, especially for systems where SerDes itself is not a primary component of the system. The primary motivation of the proposed work is to come up with a SerDes design approach that can avoid the resource heavy analog components like PLLs and clock data recovery (CDR) circuits.

In the proposed design we use a system clock and its phases to multiplex the data to the serial link to avoid the requirement of PLL-based high-frequency clock generation to serialize the parallel data. On the transmitter side the two main challenges involved in this approach are (1) generating a clean 50% duty cycle clock from the system clock and (2) producing its exact phase that can be used to multiplex the parallel data lines to serialized symbols of fairly equal duration. We introduce a statistical random sampling technique for both correcting the duty cycle [1] and generating multiple phases of the clock [2]. To avoid complex CDR circuits, multiple serial lines are grouped with a strobe that acts as a reference signal and is used for clocking on the receiver side.

On the receiver side the two main challenges are (1) compensation of the relative skew of the multiple serial links that belong to the same strobe group and (2) de-skewing and aligning the strobe with respect to serial data so that the data capturing edges occur in the eye of that data symbols to have maximum timing margin. We use programmable delay lines at the transmitter side through which the launch time of the serial data and strobe can be controlled.

3.1 Theoretical Basis

Statistical random sampling has long been used to quantitatively estimate a particular attribute of a given population through sampling some correlated observable phenomenon randomly. We applied this technique over high-speed on-chip signals in two manners: (1) to measure the duty cycle of a signal and (2) to measure relative phase of two periodic signals of same frequency and duty cycle. In this technique a random clock [7] is used, which may be generated by a digitally controlled ring oscillator fed with pseudo random numbers generated from a Linear Feedback Shift Register (LFSR). The capricious behavioral characteristics of a ring oscillator together with pseudo-random numbers generated by a LFSR produce a fine random clock for random observation of on-chip signals. Since the speed (average frequency) of the random clock has no direct bearing on the accuracy of the measurement results, this gives another dimension of flexibility in random clock oscillator design.

3.1.1 Duty Cycle measurement

To measure the duty cycle of a signal, its state is repeatedly observed and recorded at random instants of time with the help of a random clock. The duty cycle of the signal is directly related to the probability of capturing a logic high (one) in a particular random observation. A large data sample of premeditated size is gathered and the ratio of the number of ones observed to the total

observation provides the measurement results. The accuracy and confidence level can easily be controlled with the size of the sampled observation[1].

3.1.2 Relative Phase measurement

The measurement of the relative phase of two periodic signals with the statistical random sampling technique requires simultaneous observations of the two signals at random instants of time. For this, the states of the two signals are captured with a random clock and if the leading signal is captured as high and lagging signal is captured as low the observation is counted. If the cycle time of the two signals under observation is T_{cycle} and t_A is the time for which two signals overlap such that the leading signal is high and the lagging signal is low, then for a large enough sample size the ratio of number of counted observations to the total observations becomes quite equal to the ratio t_A/T_{cycle} . This is due to the fact that the joint probability of capturing the two signals in the particular state (when first is high and second is low) is equal to the ratio t_A/T_{cycle} . The observed ratio can be mapped to relative phase using the equation: phase $\phi = 2\pi t_A/T_{cycle}$. Similar to the duty cycle measurement the accuracy and confidence level of the relative phase measurement can easily be controlled with the size of the sampled observation[2]. In the subsequent section we present an implementation instance of a Random Sampling Unit used in the proposed SerDes design approach.

4. IMPLEMENTATION

The SerDes Design presented in this paper has been targeted to IBM Cu-11 standard cell library and is designed for a massive parallel computing architecture with a system clock of 500MHz. Both transmitter and receiver sides are accessible to a common host through which initial configurations, measurements and adjustments can be done at start up before the high speed serial channels are activated. The high speed communication is done through LVDS driver (OLVDS) and receiver (ILVDS) cells, provided in the IO library of IBM Cu-11 technology. The SerDes design is based on 4-to-1 multiplexing serializers (Tx-bit units) and 1-to-4 de-multiplexing deserializers (Rx-bit units) thus having a 2 Gbps link at each serial channel. Seven high-speed bits are grouped with a strobe signal to form a Tx-byte unit. A Tx-byte unit takes a 28 bit packet and transmits it over 8 serial channels including a channel for strobe. The strobe is generated by applying a bit pattern of alternating ones and zeros like 0101 to a tx-bit unit which is the same as used for serialization of data bits. On the receiver side the 8 serial channels are received at a Rx-byte unit which includes 7 Rx-bit units. Rx-bit unit is a 1-to-4 deserializer that uses the edges of the strobe to capture and deserialize the data back to parallel bits. An implementation detail of various components of the proposed SerDes system is provided in the following subsections.

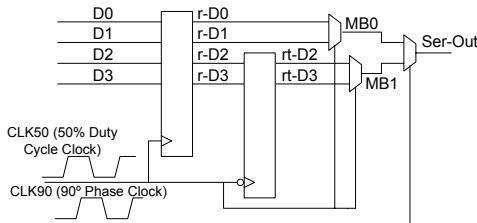


Figure 2. Tx-bit unit: A 4 to 1 serializer circuit module.

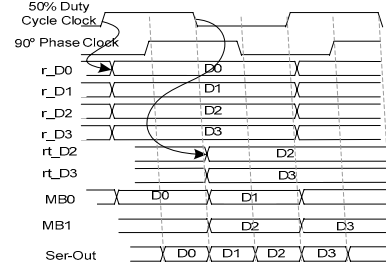


Figure 3. Timing diagram of Tx-bit unit.

4.1 Transmitter Side (Serializer)

4.1.1 Tx-Bit Unit

Figure 2 shows the standard cell based tx-bit unit circuit. All four input data bits D0~D3 are captured at the edge of the clock; bits D2 and D3 are recaptured at negative edge of the clock. Three balanced multiplexers are used to multiplex the data to a single serial output. The timing diagram of the Tx-bit unit is shown in Figure 3. It takes a symmetric (50% duty cycle) clock (CLK50) and its 90° lagging phase (CLK90) to multiplex and serialize four parallel data bits. In the proposed design these two signals are generated from the system clock. On large chips, the system clock is distributed through large clock distribution trees made up of clock buffers and interconnects of appropriate sizes to minimize skew and end-to-end delay. A noticeable degradation in duty cycle can occur at the terminal ends of the signal distribution network. This is due to the slight mismatch in the drive strengths of pull-up and pull-down networks of the CMOS gates/buffers and non-uniformity in the distribution of wiring capacitance. To conserve the portability of the design across different technologies a standard cell based duty cycle corrector (DCC) circuit is designed to fix the problem of duty cycle degradation.

4.1.2 Duty Cycle Corrector (DCC)

The proposed DCC circuit is shown in Figure 4. It employs a statistical random sampling technique to measure and correct the duty cycle of the clock. The circuit uses a small programmable delay line that provides a delayed version of the input signal. The delayed signal is ORed with the original input to stretch and ANDed with the input to chop the input signal as illustrated in Figure 5. This DCC can adjust an input signal with input duty cycle of 30% to 70% to wide achievable values within this range. The duty cycle corrected symmetric clock is then passed through a digitally controlled delay line to produce its 90° lagging phase that can be fed to Tx-bit unit as CLK50 and CLK90. The relative phase between CLK50 and CLK90 is adjusted with the help of a random sampling unit (RSU).

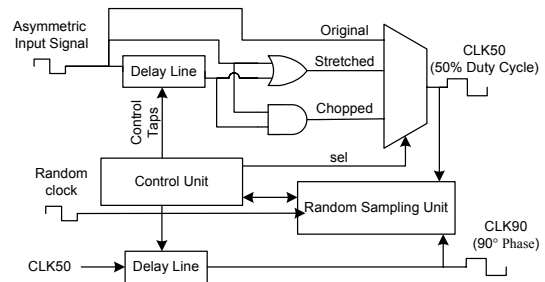


Figure 4. Duty Cycle Corrector (DCC) and Phase Generator.

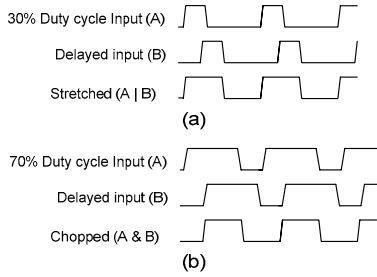


Figure 5. Timing diagram of duty cycle correction (a) Stretching (b) Chopping.

4.1.3 Random Sampling Unit (RSU)

The RSU shown in Figure 6 includes three event counters. At any transition of control signal “Sample”, “Counter 1” is loaded with “Desired Sample Size (n)” and the other two counters are reset. Cascaded flip flops are used to address the metastability issues due to the asynchronous nature of a random clock. At an active edge of the random clock, “Counter 1” is decremented, whereas “Counter 2” is incremented only when the captured state of “Signal 1” is high. Similarly “Counter 3” is incremented only when “Signal 1” is high and “Signal 2” is captured Low. When “Counter 1” decrements to zero, further sampling is stopped, at this point “Counter 2” provides the measurement of the duty cycle of Signal 1 and “Counter 3” shows the phase difference of the two input signals. The size of the counters used depends upon the required accuracy and confidence level. Our design space exploration shows that a design with 16-bit counters could be implemented within a modest area (1745 cells of size $0.4\mu\text{m} \times 4.8\mu\text{m}$ in IBM Cu-11, 130nm technology) and provides 99% accuracy with a 99.9999% confidence level. To make the correction process faster, coarser measurements can be done in the beginning with smaller sized samples, and more accurate measurements can be performed with large sized samples towards the end of the measurement and adjustment process.

4.1.4 Tx-byte unit

Figure 7 shows the block diagram of a Tx-byte unit that integrates 8 Tx-bit units to generate 7 high speed data streams and a reference strobe. A signal DCC and Phase generator in conjunction with RSU is used to supply CLK50 and CLK90 to all 8 Tx-bit units. A reference strobe sent from the transmit side

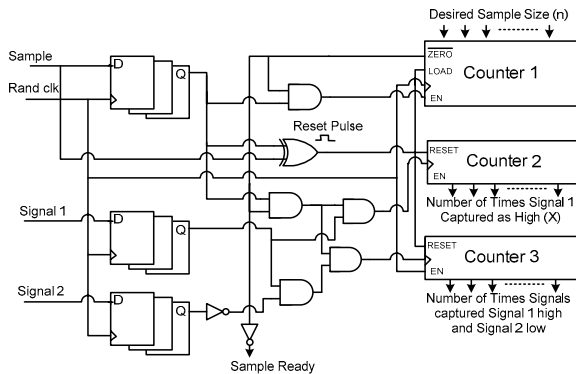


Figure 6. Random Sampling Unit (RSU)

negates any need for CDR on the receiver side. This considerably reduces circuit complexities of the de-serializer at the cost of a slight pin count increase (for strobe signal) as compared to conventional SerDes systems. To compensate the skew of the data with respect to the strobe, due to the routing and connecting wires to the receiver, the Tx-bit units are fed with CLK50 and CLK90 through controlled delay lines. By adjusting these delay lines, the launch time of serialized data or the strobe can be varied. Multiplexers at the input of each Tx-bit unit allow a selection between the regular data and alignment pattern.

4.2 Receiver side (Deserializer)

The receiver side contains the Rx-bit units and ring buffers to deserialize and align the received bits of a data packet. The de-

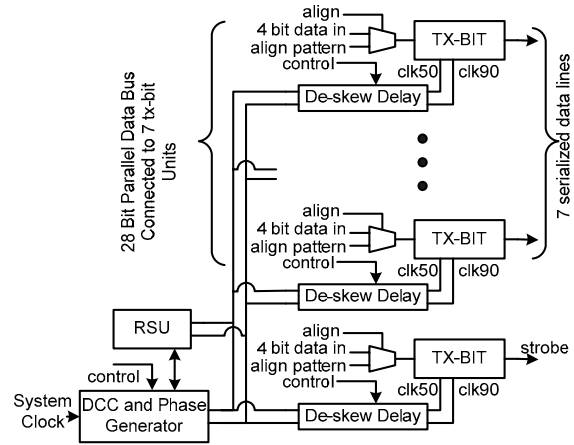


Figure 7. Tx-Byte unit.

serialized data is deposited into the ring buffers. After reset, the buffer reading process is kept on hold until all buffers have received at least one data set.

4.2.1 Rx-bit unit and Rx-byte unit

The Figure 8 shows the Deserializer circuit with a ring buffer. The strobe is passed through an equal delay fork to generate a true (*stb*) and inverted (*stb_bar*) version of the received strobe signal. The “*stb*” and “*stb_bar*” signals are then used to deserialize the received data back to four parallel bits. The two input flip flops capture the state of the serial signal in every cycle of the strobe. A “write” signal is generated by dividing the “*stb*” signal to capture the de-serialized data into the ring buffer when the output of the Rx-bit unit is valid. Figure 9 shows the timing diagram of the deserialization process of Rx-bit unit and the write signal. Rx-byte unit is a simple integration of seven Rx-bit units and RSU’s to observe the serial signals and strobe. Figure 10 shows the Rx-byte unit.

4.3 Start up and De-skewing

Putting all the pieces of the design together, some initial configuration to start high-bandwidth data communication is required for setup, or configuration. It is important to mention that even highly sophisticated SerDes designs require some time before they can start high-speed data communication. During this time their PLL’s and CDR circuit are stabilized and reach a

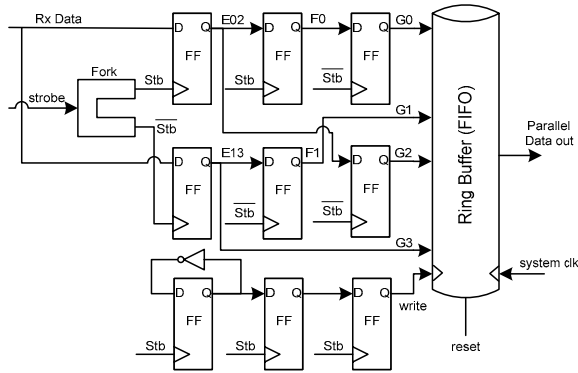


Figure 8. Rx-bit unit (1 to 4 De-serializer) with Ring Buffer.

locked state. In the proposed design at startup, the first DCC and phase generator is adjusted to generate a 50% duty cycle signal CLK50 and its 90° lagging phase CLK90. The second function is de-skewing data and strobe. The delay lines through which CLK50 and CLK90 are fed to the tx-bit unit provide a way to place the edges of the strobe exactly in the middle of the data eye to achieve maximum timing margin. To de-skew the data and strobe all tx-bit units are fed with a known periodic bit pattern (alignment pattern) that generates a periodic signal of a lower frequency than regular operating speed (1 GHz as in this design) on all serial lines of data and strobe alike. The resulting patterns are observed on the receiver side with RSU's on each serial channel and relative skews are measured. The skews are then adjusted by controlling the delays of CLK50 and CLK90 signals fed to the corresponding Tx-bit units. The lower frequency of the alignment signals is used to avoid locking with incorrect cycle in the case when skew is larger than a regular operating cycle.

5. ANALYSIS OF DESIGN AND RESULTS

The proposed SerDes is a unique design as compared to conventional SerDes designs in many ways. Timing skew is compensated during the configuration stage as a closed-loop system at boot time, whereas an open-loop timing convention is employed at run-time for high-speed data communication. Effects of jitter are tolerated by having a large timing margin. Looking at the overall picture, the proposed SerDes design is based on some

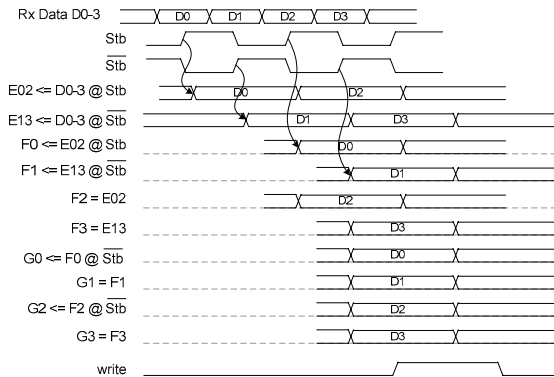


Figure 9. Timing diagram of 1 to 4 de-serializer (Rx-Bit unit).

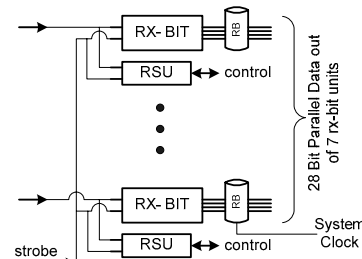


Figure 10. Rx byte unit.

unusual tradeoffs as compared to conventional SerDes designs. In this section we analyze different aspects of the design.

5.1 Design Time efficiency

The fast pace of technology scaling and reduced time to market require design time efficient circuits and systems. To make the system design time efficient, the proposed design methodology suggests the use of well-characterized library components instead of custom designing components that may involve long and tedious characterization phase before they can be used in the actual system. This design rule places a very hard constraint over design dimensions and takes off a lot of design flexibility. In the proposed design this constraint restricts the possibilities of using any custom designed high-speed line driver and receiver pair. This practically imposes an upper limit on the achievable communication speed by the SerDes designed under this rule, but advanced component libraries greatly mitigate this limit. Typical standard cell libraries and IO libraries are becoming fast enough to meet the speed requirements, e.g. the LVDS drivers provided by in the IBM Cu-11 IO libraries are sophisticated enough to handle a data rate of about 2~3 Gbps.

5.2 Portability of Design

The proposed SerDes is strictly a standard cell based design that does not require any custom design component. This characteristic makes this design practically portable to any standard cell ASIC or FPGA technology. The requirement of resource heavy components like a PLL or DLL for frequency synthesis and CDR circuits is negated by using a standard cell based RSU that can provide a comparable performance when used with an appropriate sample size. The trade-off here is the startup time of SerDes that it requires to get ready before it can start its high-speed data communication. In high end SerDes systems this time is reduced by using fast-locking PLL's and CDR's. The setup time of the proposed SerDes design may range from 500ms to a few seconds depending upon average frequency of the random clock driving the RSU's and convergence characteristics of search algorithms employed to adjust the delay line taps. If the system has to run for a considerable time before it is restarted then the setup time of a few seconds becomes insignificant.

5.3 PIN count reduction ratio

The serializer employs the tx-bit unit which is a 4-to-1 multiplexer unit, but quantitative analyses can easily indicate that the package pin count reduction ratio is not exactly 1/4. This is due to a strobe signal which is grouped with every 7 high-speed bits, i.e., 28

parallel bits are communicated over 8 high-speed channels, thus the actual pin count reduction ratio of this design is 2/7 instead of 1/4.

5.4 Area and Power Efficiency

From an area and power perspective, in general the most resource heavy components of a conventional SerDes system are a frequency multiplier or frequency synthesizer, Driver/Receiver pair and CDR circuit. The proposed design does not use these components, resulting in an overall design that is area and power efficient. The tx-byte unit occupies an area of 3257 cells excluding and 148409 cells including the LVDS drivers (where one cell is $0.4\mu\text{m} \times 4.8\mu\text{m}$ in IBM Cu-11, 130nm technology). The rx-byte unit requires 31535 cells excluding and 176687 including LVDS receivers. The result is 46442 cells ($89168.64 \mu\text{m}^2$) per high speed bit (2 Gbps channel). The power required to drive a single high-speed channel at 2Gbps is around 30 mW. For a quick comparison to a highly sophisticated SerDes design e.g. IBM High Speed SerDes (HSS)[10], the proposed design is 60% smaller in area and requires 62% less power.

5.5 Results

The proposed SerDes design targeted to IBM Cu-11 is laid out for a 1600 ball grid array (BGA) pin package using area pad drivers (OLVDS) and receivers (ILVDS) of IBM IO libraries. The design is then tested through HSPICE simulations for lossy transmission lines connecting wire models of 50Ω co-axial cable, wires on FR4 and NELCO13 boards of un-equal lengths for data and strobe lines. Lengths of the wires are set such that the delay lines to compensate the strobe-data skew should emulate the worst case scenario. Figure 11 shows an eye diagram of the data signal received at the pads of an LVDS receiver with respect to edges of the strobe signal for an FR-4 board wires. The simulations include

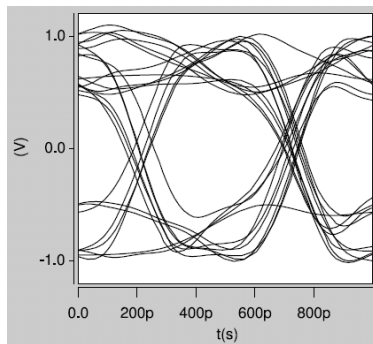


Figure 11. Simulation results for the complete system that includes line loss, crosstalk, reflections and jitter.

full loss, crosstalk, power supply reflections, and jitter. In spite of the jitter in the strobe the wide opening of the eye shows a timing margin that corresponds to a jitter tolerance of around $\pm 200\text{ps}$ to other sources of noise like crosstalk etc.

6. CONCLUSION

The increasing trend of multi core processors and embedded systems demands large signal count. SerDes are a direct solution to reduce package pin count by practically time multiplexing multiple signals on to the same pin. However the conventional designs of SerDes systems come with large die area, operating power and financial cost. The proposed SerDes uses a unique design technique to avoid resource heavy components used in conventional SerDes designs. The design is based on Standard cell technology and is portable to any other technology. The power consumption of the proposed SerDes design is 30 mW per serial link targeted to IBM Cu-11(130 nm) Technology, nearly a 2.5x improvement over the conventional design within 60% less area.

7. REFERENCES

- [1] R. Bhatti, et al, "Duty cycle measurement and correction using a random sampling technique", IEEE International Midwest Symposium on Circuits and Systems 2005.
- [2] R. Bhatti, et al, "Phase Measurement and Adjustment of Digital Signals Using Random Sampling Technique", IEEE International Symposium on Circuits and Systems 2006.
- [3] Jeff Draper, et al, *The Architecture of the DIVA Processing-In-Memory Chip*, Proceedings of the International Conference on Supercomputing, June 2002
- [4] C. Dryden, "Survey of design and process failure modes for high-speed SerDes in nanometer CMOS", 23rd Proceedings of IEEE VLSI Test Symposium 2005.
- [5] T. Geurts et al, "A 2.5 Gbps - 3.125 Gbps multi-core serial-link transceiver in $0.13 \mu\text{m}/\text{m CMOS}$ ", Proceeding of the 30th European Solid-State Circuits Conference 2004. Page(s):487 – 490.
- [6] K. Iniewski et al, "SerDes technology for gigabit I/O communications in storage area networking", 4th IEEE International Workshop on System-on-Chip for Real-Time Applications Proceedings 2004, Page(s):247 – 252
- [7] S. Maggioni, et al, "Random sampling for on-chip characterization of standard-cell propagation delay", Fourth International Symposium on Quality Electronic Design 2003.
- [8] E. Matoglu et al, "Design and verification of multi-gigabit transmission channels using equalization techniques", Proceedings of Electronic Components and Technology 2005.
- [9] H. Partovi et al, "A 62.5 Gb/s multi-standard SerDes IC", Proceedings of Custom Integrated Circuits Conference 2003.
- [10] M. Sorna et al, "A 6.4Gb/s CMOS SerDes core with feedforward and decision-feedback equalization", IEEE International Solid-State Circuits Conference, 2005.
- [11] E. Suckow, "Basics of High-Performance SerDes Design", <http://www.analogzone.com>
- [12] S. Sunter, A. Roy and J. F. Cote, "An automated, complete, structural test solution for SerDes", Proceedings of International Test Conference 2004.