International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 05 Issue: 07 | July - 2021 ISSN: 2582-3930

Design and Simulation of High-Speed CMOS SERDES Using CADENCE for High-Speed Communications

Chandupatla Tejaswi
CMR Engineering College
Electronics and Communication Engineering

Dr. K Ravi Kiran
Senior Physical Verification Engineer
Adept Chips Pvt Ltd

Abstract: *Input/output (I/O) has always played a crucial part in modern high-speed applications. As integrated circuits (IC) become smaller size and faster speed, traditional parallel communication is not suitable due to cross-talk, data-skew, and other problems related to electronic packaging and signal integrity. Serial I/O has the advantage of faster speed, less interference between adjacent links, fewer pin counts and thus lower packaging costs. A Serializer/Deserializer (SerDes) is such a tool that takes the parallel link input and condenses it into fewer lines of serial stream which might then deserialized and output because the original recovered parallel data. SerDes is extremely beneficial because it solves the issues of the many traditional parallel data links and reduces the number of I/O pins and price for connectors and cables. Designing a strong, lower power SerDes that functions properly at high speed is extremely challenging and requires knowledge from several different areas. As a result, this Project serves as an introduction to SerDes for beginners also as a tutorial of mixed-signal microcircuit design, using an example of a Serializer circuit. Fundamental concepts and major components of SerDes are covered, also because the design flow of a Serializer from unit block design in Cadence Virtuoso to simulation, using a 45nm CMOS process.*

I. Introduction

Nowadays, data rates on serial interfaces are increasing rapidly as the technology continuing to advance. The input/output (I/O) performance has become the bottleneck of the overall system performance. Traditional parallel communication like PCI and PCI-X, however, cannot meet the quality for high-speed links for inter-Integrated Circuits (IC) data transmission. In parallel communication, the difference in time of arrival of simultaneously transmitted data is usually referred as skew. The tolerance of data skew between parallel signals is approaching the practical limit, because of the increasing operating frequency of the high-speed data links, and data skew can cause critical problems such as phase difference. In addition, the cross-talk, which refers to the interference between adjacent parallel data links, is causing more problems as data rates going higher and better. What is more, the number of circuits which will be manufactured on a chip is increasing year by year, as is predicted by Moore's Law, and thus extra pins associated with parallel links would cause higher packaging costs.

To circumvent the performance limitation of the normal parallel communication, point- to-point serial digital communication is one among the possible solutions. Serial data transfer requires fewer lines, which reduces board area. The cross-talk and data skew problems are much easier to be solved during a serial link comparing to a parallel link. A device called SerDes (Serializer/Deserializer) provides a mean to convert an n-bit parallel data bus to one serial stream with equivalent bandwidth. [1].

II. Purpose

The purpose of this paper is to know the concepts and theories behind SerDes as a system also because the components inside, so as to style and implement a SerDes within our research group for further signal integrity analysis. This thesis details the basics and basic concepts of SerDes, as wells as design and analysis of a serializer circuit as an example. It will also function a tutorial for the electronic design automation (EDA) tools like Cadence Virtuoso and HSPICE (SPICE stands for Simulation Program with microcircuit Emphasis). This thesis documents a step-by-step procedure for designing and simulating a mixed-signal design using both Cadence Virtuoso and HSPICE through an example.

III. SerDes Architecture Design

Several of the core components, major blocks and circuit basics of high speed SerDes were

introduced in the previous chapter. Here in Figure 1 [2], one of the classic high speed SerDes core overview is shown.

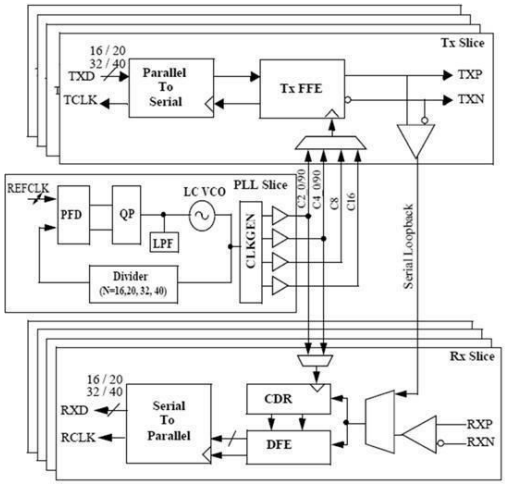

Figure 1 Overview of the SerDes core

Here the Phase-locked loop (PLL) slice ensures that the clock signals for the transmitter and receiver slice have low jitter. The transmitter (TX) slice performs parallel-to-serial conversion through a Serializer circuit. The serialized data is then fed to a feed forward equalizer (FFE) to make sure that the receiver input may be a clean waveform. The receiver (RX) slice also requires equalization after the serialized data being transmitted through the channel. a choice feedback equalizer (DFE) is required to enhance the bit error rate (BER). After the signal is equalized, the serial stream is driven through the Deseriazlier to perform the serial-to-parallel conversion. the rest of this chapter describes the planning of Serializer in additional detail.

A. Serializer

Serializer operation performs the parallel-to-serial conversion as shown in Figure 3.1. A simplified schematic of a 2:1 Serializer is presented here as an example. [Figure 3.2]

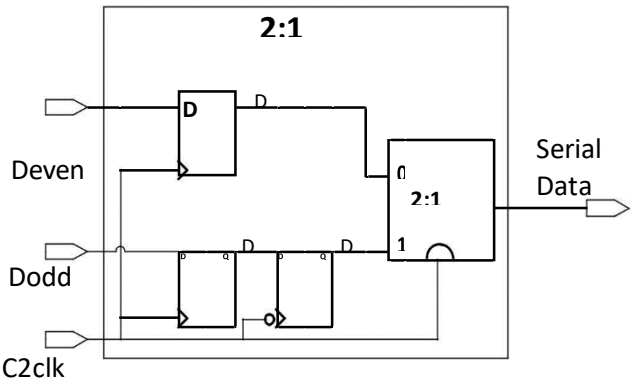

Figure 2 A 2:1 Serializer Circuit

Assuming the two bits of parallel data, Deven and Dood, are time-aligned into the Seralizer and are synchronized to the half-rate C2 clock signal. The Parallel Deven and Dodd signals are captured by the first two D-latches, which create the De and Do outputs on the rising fringe of the C2 clock signal. The Do's signal is generated by resampling the Do signal on the falling fringe of the C2 clock signal. The select input of the 2:1 MUX is controlled by the C2 clock signal, in order that when the clock is low De input is chosen, and when clock is high Do's is chosen .

B. D-latch design

A latch may be a crucial component within the development of several major blocks in high speed SerDes, including the Serializer block, Differential Driver block, Phase Detector block, Deserializer block, etc… A positive latch may be a level sensitive circuit that passes the D input to the Q output when the clock signal is high and it's said to be within the transparent mode. When the clock is low, the input file sampled on the falling fringe of the clock is held stable at the output for the whole phase, and therefore the latch is claimed to be within the hold mode. Similarly, a negative latch passes the D input to the Q output when the clock signal is low. A register, however, is an edge- trigged component contrary to the level-sensitive latches. A latch is an important component within the construction of an edge-triggered register. A flip-flop generally refers to any bistable component, formed by the cross coupling of gates. Often in some textbooks, an edge-triggered register is mentioned as a flip-flop also . [7] Shown in Figure 3 is that the transistor-level implementation of a positive MUX-based D- latch built by using transmission gates.

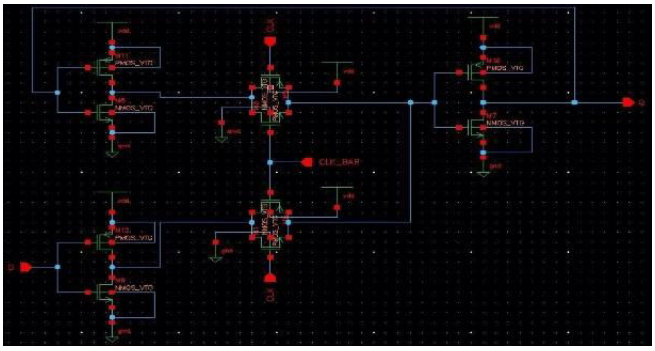

Figure 3 A positive MUX-based D-latch using transmission gates

When the CLK signal is high, the lowest transmission gate is on, and thus the latch is transparent – input D is copied to Q. During this point, the highest transmission gate is off. When the CLK signal is low, the lowest transmission gate is off while the very best is on. The feedback ensures the output is held as long because the CLK signal is low.

The problem of such MUX-based D-latch design using transmission gate is that it requires both CLK and CLK_bar signal, which could lead on to clock overlap and eventually cause race condition to happen. a real Single Phase Clocked (TSPC) Latch [8] can overcome the matter caused by clock overlap. Figure 4 shows the transistor implementation of a TSPC latch.

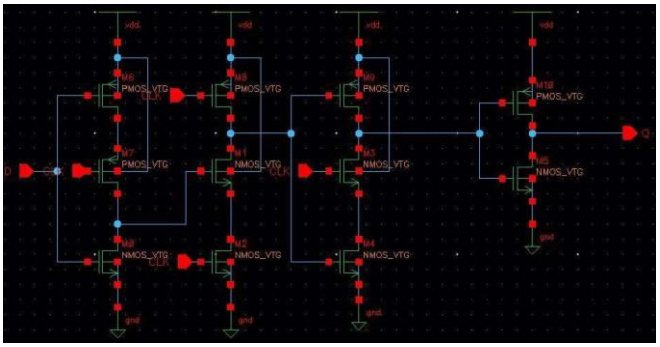

Figure 4 a real Single Phase Clocked Latch

For the positive TSPC latch shown above, when the CLK is high, the latch is within the transparent mode, and corresponds to 2 cascaded inverters. When the CLK is low on the opposite hand, both inverters are off and therefore the latch is in hold mode.

A slightly different configuration of a TSPC with split out latch is employed within the final design of the Serializer as is shown in Figure 5. The advantage is that fewer transistors are needed during this design, and thus lower power consumption of the general system. Also smaller propagation delay leads to faster speed of the whole circuit. Again, no inverted clock signal is required during this design, therefore the circuit is freed from clock skew issue.

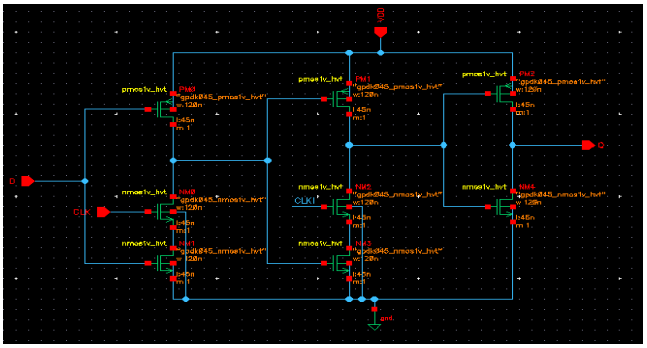

Figure 5 a real Single Phase Clocked Latch with split output

C. **Multiplexer design**

The transistor-level schematic of a transmission-gate multiplexer is shown in Figure 6. the thought behind this circuit is to use two transmission-gates as simple switches to propagate either input A or input B on to the output. an additional inverter is required to get the inverted select signal S_bar. While the upper transmission-gate is activated by S, the lower transmission-gate is activated by S_bar, thanks to the wiring of their control inputs. When S is low, only the lower transmission-gate is conducting (because S_bar is connected to its n-channel and S to its p-channel transistor gate inputs), while the upper transmission-gate is non-conducting. As a result, the worth of B is skilled to the output of the multiplexer. When S is high, the upper transmission- gate is activated, while the lower transmission-gate is non-conducting. Therefore, the worth of A is skilled to the multiplexer output.

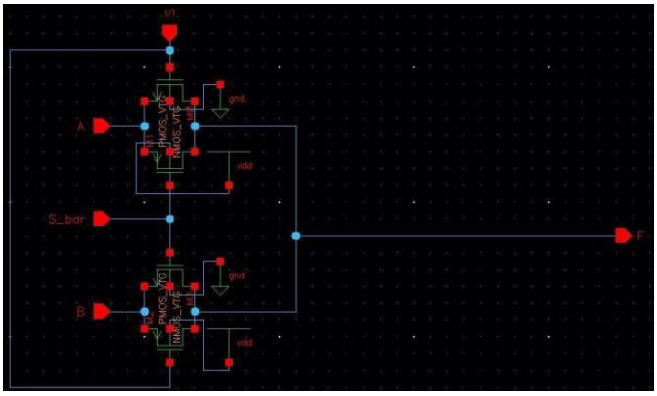

Figure 6 Transmission-gate multiplexer

However, traditional transistor sizing method and logical effort can't be applied to the present transmission-gate multiplexer deisgn, and thus it's hard to seek out out the optimal transistor size for max speed theoretically. Also, so as to lower the equivalent resistance Req, the transmission gate must be made wide. The capacitance of the gates, however, also will be increased, leading to no reduction within the time constant of the transmission-gate multiplexer. As a result, another design called Current Mode Logic (CML) Multiplexer is adopted and is shown in Figure 7. The CML circuits are widely utilized in GHz range high speed bipolor driver or multiplexer implementations.

The differential selects signals, S and S_bar, select which of the 2 data-input A and B to be connected to the output. When the select signal, S is high (and S_bar is low), A directly affect the output while B is disconnected. When the select signal, S goes low (and S_bar high) B are going to be connected to the output. Thus, both levels of the clock are going to be wont to multiplex the info. The advantage of this CML circuit is that it's higher operating speed with constant power consumption independent of operation frequency.

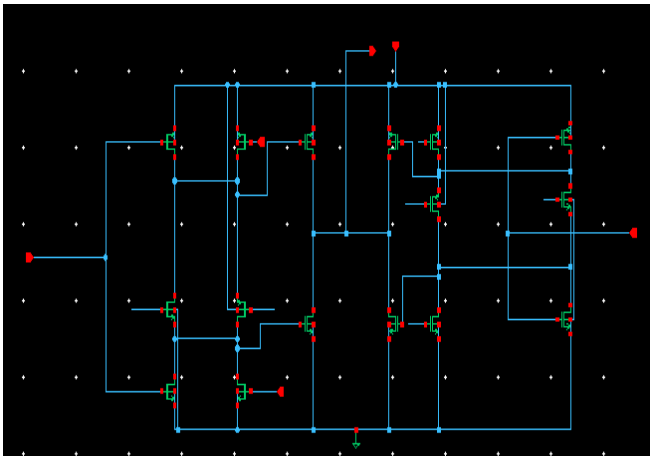

Figure 7 Test bench for the 1:2 Deserializer circuit

IV. Conclusion

In summary, the work presented during this paper laid down a path necessary to realize knowledge of designing and building analog and digital circuits. an easy Serializer and deserializer circuit is meant, created and simulated using Cadence Virtuoso. The step-by-step "cookbook style" tutorial of mixed-signal circuit design and simulation was shown, and will be applied to other transistor level circuit design projects also. The Serializer/Deserializer circuit presented during this project isn't only a key component during a SerDes system, but also is an example for one to know the serialization process. Other core components of a SerDes, including Equalizers, Clock and Data Recovery (CDR), Differential driver and receiver, and Phase-locked loop (PLL) were also discussed.

V. Reference

[1] Lattice Semiconductor Corporation, "Serdes Introduction," China Distributor FAE Training, January 2003.

[2] D. Stauffer, J. Mechler, K. Dramstad, C. Ogilvie, A. Mohammad, J. Rockrohr and M. Sorna, High Speed Serdes Devices and Applications, New York: Springer, 2008.

[3] Y. Shim, W. Lee, E. Song, J. Cho and J. Kim, "A Compact and Wide-Band Passive Equalizer Design," *IEEE MICROWAVE AND WIRELESS COMPONENTS LETTERS,* vol. 20, no. 5, pp. 256-258, May 2010.

[4] D. Chen, "SerDes Transceivers for High-speed Serial Communications," Carleton University, Ottawa, Jan 2008.

[5] J. Schutt-Ainé, "Signal Integrity," Class notes for

ECE451, Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, 2009.

[6] T. Van Roon, "PLL Tutorial," 19 October 2010. [Online]. Available: http://www.sentex.ca/~mec1995/gadgets/pll/pll.html. [Accessed 30 June 2013].

[7] J. Rabaey, A. Chandrakasan and B. Nikolic, Digital Integrated Circuits: A Design Perspective, Prentice Hall, 2003.

[8] J. Yuan and C. Svensson, "High-Speed CMOS Circuit Technique," *IEEE Journal of Solid- State Circuits,* vol. 24, no. 1, pp. 62-70, 1989.