Original Research Paper

Engineering

IMPLEMENTATION OF USB FOR HIGH SPEED DATA TRANSFER

Kuldeep Sharma

M.TECH SCHOLAR (VLSI) Electronics and Communication Engineering Department Greater Noida Institute of Technology (Greater Noida)-201308, India

ABSTRACT In this paper we design the physical layer of USB3.0 with Super Speed functionality. The physical layer consists of the PCI express and PIPE interface. The design can transfer data serially from transmitter to receiver at 2.5 Gb/s. The design generates a clock that runs at a frequency of 125MHz to transfer data in parallel interfaces. The architecture of the USB3.0 physical layer is proposed in this paper. The architecture is designed and implemented using Verilog HDL in Xilinx Vivado 2017.4

KEYWORDS : USB 3.0, PHY, Physical Layer, PCI Express, PIPE interface.

1.INTRODUCTION

Universal Serial Bus (USB) is an industry standard developed in the 1990s that defines connector protocols used in bus for transmission and communication. It was developed ad invented by Ajay Bhatt.It was designed to standardize the connection of peripherals to personal computer, both to communicate and to supply electric power. It has largely replaced interfaces such as serial ports and parallel ports, and has become commonplace on a wide range of devices. It is a fast,low cost, bi-directional, dynamically attachable interface that meets the requirements of digital systems of present day.

USB 3.0 is the third major version of the USB standard, inarguably the most successful and ubiquitous interface standard ever created.Among other improvements, USB 3.0 adds the new transfer rate referred to as SuperSpeed USB (SS) that can transfer data at up to 5 Gb/s (625 MB/s), which is about 10 times as fast as the USB 2.0 standard. The following areas indicating the other improvement of its features:

- Advanced power consumption features
- Improved bandwidth
- Increased maximum bus utilization
- Backward USB 2.0 compatibility

In USB 3.0, dual-bus architecture is used to allow both USB 2.0 (Full Speed, Low Speed, or High Speed) and USB 3.0 (SuperSpeed) operations to take place simultaneously, thus providing backward compatibility. Connections also permit forward compatibility, which is, running USB 3.0 devices on USB 2.0 ports. The structural topology is the same, consisting of a tiered star topology with a root hub at level 0 and hubs at lower levels to provide bus connectivity to devices. The Structural topology is shown in Fig. 1

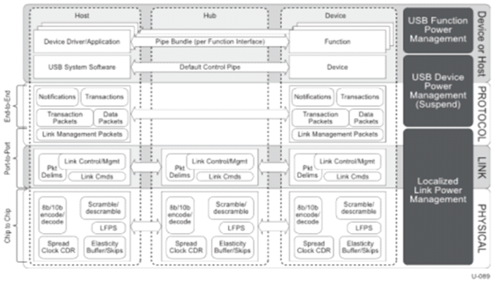

Fig.1 Structural topology of USB 3.0

In this paper we have designed and implemented the physical layer of USB 3.0. The rest of paper is organized as follows: Section II describes the functionality and architecture of the physical layer of USB 3.0. Section III describes the implementation of the transmitter and receiver. The simulation results are provided in Section IV and Section V concludes the work.

2.PHYSICAL LAYER

The physical layer defines the PHY portion of a port and the physical connection between a downstream facing port (on a host or hub) and the upstream facing port on a device. The physical connection comprises of two differential data paths: the transmit data path and receive data path. They use two separate unidirectional paths. The architecture of the PHY transmitter and PHY receiver is shown in Fig.2 and Fig.3 respectively.

After receiving 8 bit data from link layer, physical layer scrambles these data to decrease EMI emission. To transmit over the physical connection, it encodes the 8 bit scrambled data in to 10 bit symbols. These data, then, sent at a speed to keep EMI emission low. The bit stream is recovered by the receiver and then descrambled and decoded from 10 bit data to 8 bit data and for further processing, it is been sent to the Link layer.

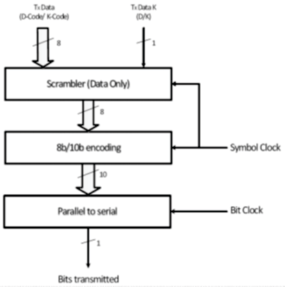

Fig.2 PHY Transmitter architecture

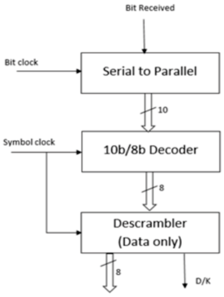

Fig.3 PHY Receiver architecture

3.IMPLEMENTATION

PHY TRANSMITTER:

The PHY transmitter shown in Fig.2 consists of three parts. Scrambling is done to reduce the EMI noise generated by sequence. It helps to eliminate repetitive patterns which result in large amounts of energy in a small spectrum of frequency. Scrambling helps to spread the energy over a frequency range. Scrambling is achieved by XORing the data with the output of a linear feedback shift register (LFSR). The LFSR is realized using the polynomial equation:
G(X) = X16+X5+X4+X3+1.

The encoder converts the 8bit scrambled data into a 10bit data. The original 8-bit data is broken into two blocks, 3 most significant bits (y) and 5 least significant bits (x). Encoding of x and y is then done with the help of a 3b/4b encoder and a 5b/6b encoder. The 3 bits (y) is encoded into 4 most significant bits and the 5bits (x) is encoded into 6 least significant bits. Encoding is done in such a way that we obtain a DC balanced output signal for each symbol. That is, the number of ones in the entire output stream is equal to number of zeroes. A running disparity register is used to keep track of the difference in number of ones and zeroes.

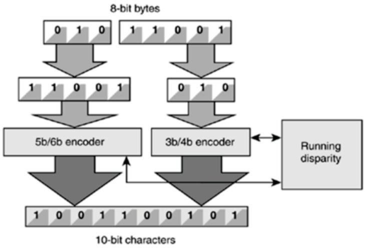

Fig.4 Encoding scheme

The parallel to serial converter is used to convert the 10bit data in serial format to enable transmission over the channel. The converter runs on a different clock proportional to the bit rate as compared to the other two blocks which run on a clock proportional to the symbol rate.

PHY RECEIVER:

The PHY receiver shown in Fig.3 also consists of three parts. The serial to parallel converter converts the serial data back into a 10b data. It is clocked by a clock proportional to bit rate.

The 10b symbols are converted back to 8b characters by the 8b/10b Decoder. The 8b/10b Decoder also looks for errors in the incoming 10b symbols. For example, error detection logic can check for invalid 10b symbols or detect a missing Start or End character.

The De-Scrambler (Optional) reproduces the de-scrambled packet stream from the incoming scrambled data stream. The De-Scrambler implements the inverse of the algorithm implemented in the transmitter Scrambler.

4.SIMULATIONRESULTS

The simulation waveform of the PHY transmitter and the PHY receiver is shown in Fig.5 and Fig.6 respectively. The encoder will generate positive and negative disparity code on consecutive clock cycles in order to maintain the DC balance. As we seen in simulation design, clock divider which is used to divide the BIT RATE clock by 10. This clock is used to hold the 10bit encoded data until each bit is serially transmitted using parallel to Serial conversion. In the transmitter side the parallel to serial converter used to convert the 10bit encoded parallel data into 1bit serial data. The waveform of transmitter and receiver section is given below:

1.Transmitter Section:

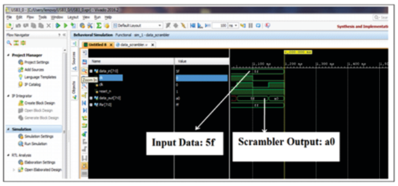

Figure: 4.2 Simulation Waveform of Scrambler of transmitter

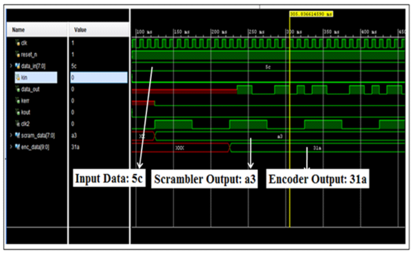

Figure 4.5: Simulation Waveform Showing Overall Transmitter Output

2.Receiver Section:

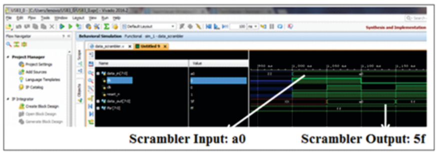

Figure 4.8: Simulation Result of Descrambler of Receiver

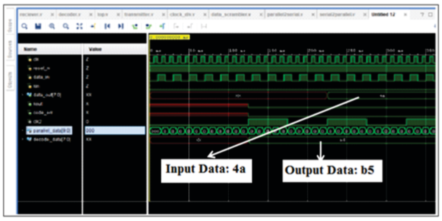

Figure 4.9: Simulation Result Showing Overall Receiver Output

5.CONCLUSION

This design explains the concept of USB 3.0 and Serial data communication. This design is able to transfer data on 2.5GHz clock. This design helps to understand the Super speed features of USB 3.0. Due to data bursting capability, USB 3.0 provide far more speed than USB 2.0 as device does not have to wait for the hubs acknowledgment. It also helps to understand 8b/10b encoder and decoder and how they provide a dc balanced data. And this design is designed and verified using Verilog HDL in Xilinx Vivado 2017.2.

REFERENCES

1. Hardik Trivedi, Rohit Kumar, Ronak Tank, "Implementation of USB 3.0 SuperSpeed Physical Layer using Verilog HDL", International Journal of Computer Applications (0975 – 8887), Volume 95– No.24, June 2014
2. Rakesh Roshan Mandel, Philemon Daniel, "A study to implement super speed plus USB 3.1 physical layer" International Journal of Electrical, Electronics And Data Communication, ISSN: 2320-2084 Volume-4, Issue-2, Feb.-2016
3. A B M Najmul. Karim, Mohammad. Anas, Tashfia. Afreen, and Iqbalur. Rahman Rokon, "FPGA Implementation of USB 3.0 (Super-Speed Bus) Function IP Protocol using Verilog HDL", 2nd International Conference on Power Electronics, Systems and Applications (ICPESA'2013) Sept. 25-26, 2013
4. "Universal Serial Bus 3.0 Specification", Revision 1.0, November 12, 2008.
5. "Data Manual Texas Instruments", Literature number: SLLSE16E, June 3, 2011
6. A.X. Widmer and P.A. Franaszek, "A dc-balanced, partitioned block, 8B/10B transmission code, "IBM journal of Research and Development, vol.27, no.5, pp.440-451, Sep 1983.
7. "Universal Serial Bus 2.0 Specification", Revision 1.0, March 13, 2006.
8. "PHY interface for the PCI Express and USB 3.0 Architecture", March 11, 2009.
9. "Lattice Semiconductor Corporation 8b/10b Encoder/Decoder", February 2012.
10. Ching-Che Chung, Chen-Yi Lee, "An All-Digital Phase Locked Loop for high speed clock generation", Febr 6, 2003.
11. Clifford E. Cummings and Peter Alfke, "Simulation and Synthesis Techniques for Asynchronous FIFO Design with Asynchronous Pointer Comparisons", SNUG 2002.
12. Ravi Budruk, Don Anderson & Tom Sanely, 2004. "PCI Express System Architecture", Mindshare Inc., pp 419-434.
13. Thatcher, Jonathan (1996-04-01). "Thoughts on Gigabit Ethernet Physical", IBM Retrieved on 2008-08-17.
14. Jenming Wu & Yu-Ho Hsu, "8B/10B Codec for Efficient PAPR Re-duction in OFDM Communication Systems", International technology roadmap for Semiconductors (ITRS).