FPGA-based USB 2.0 data monitoring and acquisition circuit function and simulation design

Chenhao Duan* and Binhua Li

1 Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming, Yunnan, 650504, China

*Corresponding author’s e-mail: duanch94@foxmail.com

Abstract. This paper builds a system based on FPGA, which is used for USB 2.0 data monitor, simulate and verify system. To build the system, and construction of hardware experimental platforms and logical module programming with HDL are both required. The core of the system is FPGA and ULPI peripherals, which extend two external USB ports for FPGA. By using HDL to package ULPI core, generate data buffer and relays module, and build the connection between ULPI data transceiver. And the function of data monitor and collection will be accomplished. The cooperation simulation by using ISE and ModelSim is using for system analyzation, modification, and verification.

1. Introduction

As digital devices become more and more popular, the data transmission frequency and rate become higher. So, the Universal Serial Bus (USB) protocol become one of the most popular communication agreements between devices, which supports hot-plugging function. But the USB protocol is a type of point-to-point transmission agreement, it does not support data transmission between more than two devices [1]. The signal analyzers, USB protocol analyzers are usually used to collect the data on the USB bus when needed. These methods cost much, require computer software assistance, and do not have good expandability and portability.

To solve these problems, an FPGA-based system is proposed in this paper. This system can monitor and collects data on the USB 2.0 bus. The advantages of this system are: cost less, has good portability, does not require computer assistance, and can collect data individually.

The ULPI agreement is one of USB Physical (PHY) agreements. To the UTMI+ agreement, the ULPI agreement can decrease the number of control and data pins effectively, reduce system development difficulty, and save development time consumption [2].

This paper makes the ULPI core and data transmission of ULPI peripheral to be the system core, design and built a hardware circuit, code FPGA logical modules. The verification of the data monitor and collection is accomplished by analyzing waveforms from the simulation.

2. Design of hardware circuit

At first, the data transmission between the original device and the original host is required integrally. After that, data monitor and acquisition can be engaged. Because of the consideration, the number of the USB ports on the system, the system required at least two ULPI PHY chips. So that the system has two USB ports. One chip communicates with the original host, the other communicates with the original device.
The FPGA chip is the core of the processing system. With the HDL program, the FPGA chip mainly accomplished the ULPI peripheral control and transmission. It also handles data relays, DDR3 control, and other basic functions.

2.1. Design of hardware circuit
The original signal, which is transmitted to the original device, transmits to the FPGA first. Then transmits to a device with ULPI peripheral. When the device returns data, the return-data transmitted to FPGA first. The FPGA program decides the valid data, copies the data and stores to DDR3 temporarily. At the same time, the whole data from the device is transmitted to the host with the other ULPI peripheral. In such a way to make sure that the original data flow is integrated, and the data monitor and collection is accomplished. The temporary storage data is saved to the DDR3 on the FPGA development board. The data will be sent to the backend circuit with other ports when the backend circuit is required.

The hardware circuit structure diagram of this system is shown in figure 1.

2.2. Circuit core chip selection
The FPGA chip named XC6SLX16-FTG256, which belongs to the Xilinx Spartan-6 family, has low power consumption and high performance. Based on the structure of previous generations of Spartan series, this FPGA chip utilizes the mature and reliable Virtex series’ high-end architecture [3].

The USB3300 chip, made by Microchip, is chosen to become the ULPI PHY chip for this system. The chip can connect to ULPI Link with ULPI agreement. And the chip can communicate with a host or a device by using USB protocol. This chip has a broad utilization, less difficult to use, and the ability to meet the USB 2.0 interface expansion and control requirements of this system.

2.3. Design of ULPI peripheral circuit
For the chip to work properly, part of the USB3300 peripheral circuit should be designed like figure 2. The crystal oscillator circuit provides a clock signal with a frequency of 24M. A stable 60M clock signal is outputted by the on-chip phase-locked loop (PLL), to guarantee the peripheral works normally.
3. Modification and verification of HDL logical modules

The modification and verification of the HDL program are used to verify data processing on FPGA. It also used to simulate and analyze the design of sub-modules, verify the integrity of sub-modules and system function. The accuracy and correctness of every sub-modules are required. Make sure that the sub-modules are work properly, so the system can work expectedly.

To control two ULPI peripheral modules separately, two ULPI data transceivers and control modules are required. The data buffer module saves the data to DDR3 when it receives valid data. And sends it to the backend circuit when it required.

3.1. Design of HDL logical modules

In this system, the ULPI data transceiver and control modules (ULPI cores) are a package from UTMi+ IP core, which is provided by organization Opencores. The function of data writing and reading with DDR3 is accomplished by using the modified IP core, which is provided by the Xilinx. The FPGA signal processing and coordination module are designed based on the WISHBONE structure. Adding direction arbitration, data replication, and other auxiliary modules to complete the FPGA HDL program. The program structure diagram is shown in figure 3.

Figure 3. HDL logical module structure
To simplify the program design, this paper makes some restrictions on system connections. The ULPI peripheral II in figure 3, connects to the host (computer) as a device only. The connection on ULPI peripheral I is unrestricted. In the ideal situation, the ULPI peripheral I can connect to a computer or other devices with OTG mode. But the system integrates and the connection should be verified first, so the ULPI peripheral I connects to another computer. And the analysis of data transmission and reception will be more conveniently.

In the structure for figure 3, the ULPI data transceiver and control module is a wrapper from the UTMI+ core. FPGA data processing and coordination module design based on the WISHBONE structure, with some auxiliary modules’ assist, the function of data replication, direction judgment is accomplished. The data buffer module and data relays module are modified from the IP core of DDR3.

3.2. Package of ULPI core
The WISHBONE structure is already integrated into the UTMI+ core, which is used to package into a ULPI core. And the UTMI+ IP core is provided for free. The ULPI agreement and the UTMI+ wrapper document reference UTMI+ Low Pin Interface (ULPI) Specification and ULPI Design Guide.

ULPI test generation logic, ULPI control status register, and ULPI interface multiplexer are required for ULPI wrapper. The wrapper will be accomplished with these following modules.

3.2.1. ULPI control status register. The control status register in the UTMI+ core is not available for ULPI wrapper. The ULPI core requires a new control status register, which adapts the characteristics of ULPI control signals and functions. The new control status register can provide control signals, control the state machine correctly, response to incoming data or signals.

3.2.2. ULPI interface multiplexer. The function of ULPI interface multiplexer is data separation/combination with different direction of transmitted data. When the ULPI peripheral sends data to the UTMI+ core, the Package IDs (PIDs) sent into the control status register and bring the state machine into the appropriate state. And the data transmitted into UTMI+ core directly. The data sent to the backend modules when the data unpacking and combination are completed. When the ULPI core sends data to ULPI peripheral, PIDs are package with ULPI control status register, so the ULPI peripheral can jump into the appropriate state. After the data packaged in the UTMI+ core, it sends to ULPI peripheral via ULPI interface multiplexer.

3.2.3. ULPI generation test logic. ULPI generation test logic verifies the incoming data with CRC6 or CRC15. According to the verification result, the next processing is performed. The data transmitted by USB will not be wrong in most situations.

3.3. DDR3 control module
The DDR3 chip we selected can be controlled and connected with the IP core, which is provided by Xilinx. The modification of this IP core is required to adopt the system.

3.4. Simulation and verification
Coding the testbench to simulate modules with ISE and ModelSim cooperate simulation. Comparing simulation waveforms and expected results to check out that the function of logical modules is integrated or not. The sequence of system simulation is shown in figure 4.

3.5. Simulation and verification of ULPI core
The sequence of the simulation of the ULPI core is shown in figure 5. Functional simulations of sub-modules are first carried out. The functional simulation of the whole system is next.
The simulation of wake up should be simulated at first. The system should quit the low power mode at the beginning, then the system is activated. Active the control signal high for 5 periods, and drop it low in the following period to wake up the system. The low power state machine will keep watch over the control signal DIR.

After the system woke up, the system will identify the PIDs. There are two types of PIDs, the DATA and the control PIDs. According to different types of PIDs, the data behind the PID will be processed differently. The control signals simulation required the system can identify every control type signal. The PID identification state machine should be able to jump into the appropriate state with different control signals. The data of DATA will be transmitted into the UTMI+ core. UTMI+ core combine 4 period of 8-bit data as a 32-bit data, and sends to backend modules by WISHBONE structure.

The simulation waveform of data combination and transmission is shown in figure 6. The four periods of data are 01010100, 01010110, and two periods 00100001. They are combined as 32-bit data. The data transmitted by WISHBONE shows that the simulations, which are mentioned in figure 5, are all properly functioning. And the ULPI core can finish the function of ULPI peripheral control.

Figure 4. System simulation and modification process
Figure 5. Simulation and verification sequence
Figure 6. Send data to WISHBONE
3.6. Simulation and verification of system
The WISHBONE structure connects two ULPI cores and DDR3 control module. The random data of DATA is sent into the system by one ULPI peripheral. After data combination, 32-bit data transmitted to other ULPI core and split into four 8-bit data. The split data transmitted to another ULPI peripheral by another ULPI core. At the same time, the valid data be sent to DDR3 for temporary storage. The data monitor and collection are accomplished.

4. Conclusion
The system is accomplished with hardware circuit and HDL logical modules’ design. And the system functional simulation is completed. Using ISE and ModelSim co-simulation to verify the functions of logical modules. The results of the comparison between state machines' states and output waveforms are used to analyze and modify the system. These allow the system to perform monitoring and acquisition of data on the USB 2.0 bus eventually.

Acknowledgments
This work is supported by the National Natural Science Foundation of China (11673009). I thank Professor Binhua Li for experimental guidance and comments. Opencores organizations for provides open source UTMI+ core. Lei Shu and Jiajia Zheng for paper modification. The fellows of the laboratory for comments and assistance of modification of HDL logical modules. Classmates at KUST who helped to conduct the experiments are also acknowledged.

References
[1] Hewlett-Packard, Intel, Lucent, Microsoft, NEC, Philips USA. (2000) Universal Serial Bus Specification. Revision 2.0. www.usb.org.
[2] Philips, Motorola, TransDimension, Cypress, ARC, SMSC, Conexant, Synopsys, Mentor, ISI, Texas Instruments, Qualcomm USA. (2004) UTMI+ Low Pin Interface (ULPI) Specification. Revision 1.1. www.ulpi.org.
[3] Xilinx USA. (2001) Spartan-6 FPGA Family. https://www.xilinx.com/products/silicon-devices/fpga/spartan-6.html.
[4] Microchip USA. (2016) Hi-Speed USB Host, Device or OTG PHY with ULPI Low Pin Interface. http://www.microchip.com.
[5] Opencores USA. (2019) USB 2.0 Function Core. https://opencores.org/ projects/usb
[6] Xilinx USA. (2019) DDR3 Controller. https://www.xilinx.com/products/intellectualproperty/ddr3.html#overview
[7] SMSC, MICROCHIP USA. (2012) ULPI Design Guide. Revision 1.0. https://www.microchip.com/
[8] Mentor Graphics USA. (2019) ModelSim – Leading Simulation and Debugging. https://www.mentor.com/products/fpga/model/.
[9] Rudolf Usselmann USA. (2007) USB Function IP Core. www.opencores.org.
[10] Opencores USA. (2010) WISHBONE System-on-Chip (SoC) Interconnection Architecture for Portable IP Cores. https://opencores.org/howto/wishbone