Design and Implementation of USB3.0 Data Transmission System based on FPGA

Fenxian Tian [a], Juan Li [b], Xinxin Sun [c] and Jun Yang [d, *]

School of Yunnan University, Yunnan 650504, China.

[a]1878777810@qq.com, [b]1561437500@qq.com, [c]1015462008@qq.com, [d, *] junyang@ynu.edu.cn

Abstract. Aiming at the problems of slow transmission speed and high hardware resources in the traditional USB data transmission system, a USB3.0 data transmission system based on FPGA is designed. The system uses FPGA as the core control chip, and EZ-USB FX3 realizes high-speed data transmission of USB3.0 synchronous Slave FIFO mode through GPIF II general programmable interface. The design scheme has high portability and expandability, can be widely applied to different communication projects. System performance analysis and test results show that the average transmission rate of the system can reach 328MB/S, and the logical resource occupancy rate is less than 1%, realizing high-speed data transmission.

Keywords: FPGA; USB3.0; DDR3; Data Transmission.

1. Introduction

With the continuous development of electronic information technology, massive amounts of data information need to be stored and transmitted, and real-time requirements are more stringent, so high-speed and even ultra-high-speed interfaces must be used for data collection, so as to ensure the timely transmission and accuracy without loss of data [1]. The ideal data transfer speed of USB3.0 interface can be as high as 5Gbps, with its fast transmission speed, small size, flexible interface and mature technology[2], it has been widely supported by everyone, providing a solution for high-speed real-time data transmission. In recent years, with the improvement of chip manufacturing level, high-performance and low-cost field programmable logic gate array (FPGA) has potential advantages in high-speed signal processing in communication interface[3]. FPGA technology can not only improve the parallelism of the system, speed up the processing speed of data instructions, but also enhance the robustness and stability of the system [4], and has many advantages such as large scale, high integration, and flexibility[5].

In this paper, the traditional USB data transmission system has the disadvantages of slow transmission speed and occupation of logic resources. FPGA is used as the main control chip, DDR3 cache is used as the storage unit, and combined with USB3.0 interface technology, an efficient data transmission system based on FPGA and USB3.0 interface is designed. The system provides perfect transmission and storage functions and has broad development space.

2. System Overall Design and Key Technologies

The system is designed with the idea of modularization [6]. It is mainly composed of five parts: signal preprocessing module (including signal conditioning circuit and A/D conversion circuit), DDR3 memory module, FPGA main control module, USB3.0 data transmission module and upper computer display module. This paper will focus on the design of DDR3 storage modules and USB3.0 data transfer modules. The overall design idea of the system is as follows:

(1) Signal preprocessing module mainly realizes signal acquisition and conversion. First, the semaphore is converted into an analog electrical signal by the sensor; then the signal conditioning circuit performs the limiting, bucking, amplification filtering and the like on the weak analog electrical signal [7](to prevent the ADC sampling chip from being damaged); finally, the ADC sampling chip passes the A/D conversion. The converter converts the continuous analog signal into a discrete digital signal.

(2) The DDR3 memory module performs an orderly reading of the data buffered in the FIFO through the controller, and accurately writes the memory unit to realize the cache of the data.

(3) The FPGA main control module mainly realizes the control function of the DDR3 controller and the connection problem of the USB3.0 interface. This module is the core module of the whole system, which controls the timing and logic of the system operation, so that each module works efficiently and orderly.

(4) The USB3.0 data transfer module implements the GPIF II synchronization from the FIFO interface to the FPGA. Synchronous read and write timing from the FIFO interface is the focus of this module design. By verifying the correctness of the read and write timing, writing the state machine, and then configuring the USB firmware file, FPGA and USB data transmission can be realized.

(5) The upper computer module is mainly composed of a data receiving part, a data buffering part and a data display part [8]. The data receiving part uses the design method of asynchronous transmission to store the data transmitted by the USB3.0 lower computer into the buffer, and finally displays it on the upper computer interface to complete the entire data transmission process. FIGURE 1 is a block diagram of the overall system of the system.

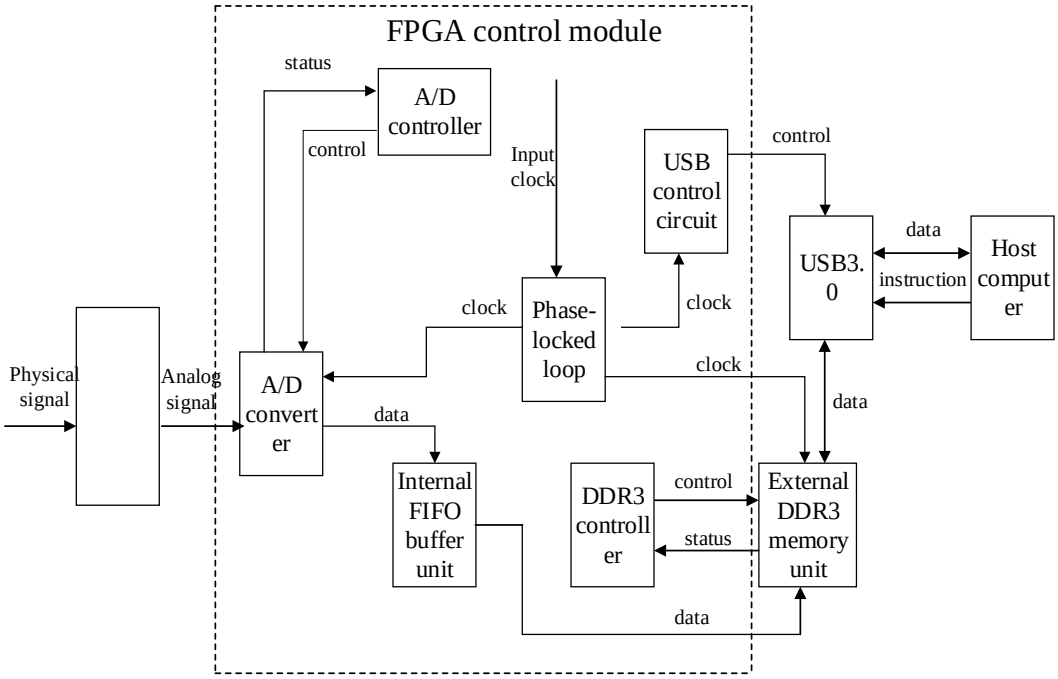


Figure 1. System overall principle block diagram

3. Design of Main Functional Modules

3.1 FPGA and DDR3 Memory Module Design

In this module, DDR3 external high-speed memory is designed to cache large amounts of data to avoid data loss in the FIFO and cause data loss[9]. DDR3 can greatly improve system performance and signal integrity while reducing system power consumption[10]. Currently, DDR3 technology has been widely used in high-speed data transmission equipment.

In practical engineering applications, since DDR3 cannot directly recognize the access request of the processor, it is necessary to design a DDR3 controller to control the reading and writing of DDR3. However, considering the control logic of DDR3 is complicated, it is often difficult to design by itself. It will take a lot of time, so we can directly use the MIG IP core developed by Xilinx to generate DDR3 controller. Its structure is shown in FIGURE 2. It mainly includes three modules: system clock and reset module, user interface module, and MIG IP core controller interface module.

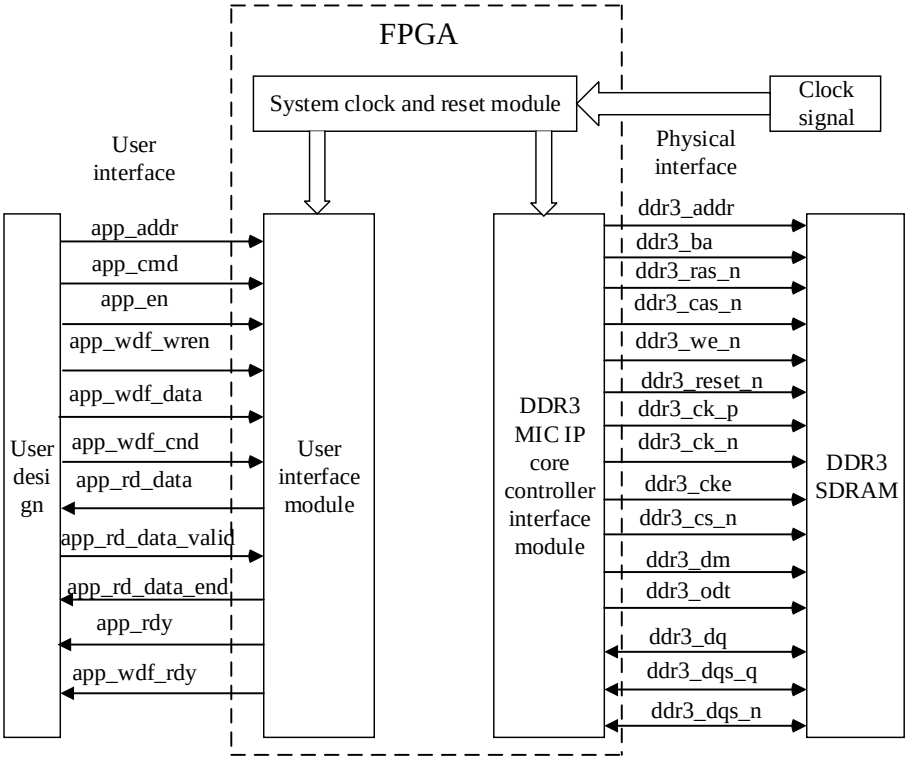


Figure 2. DDR3 controller structure

Among the three modules, the user interface module is the core content of the DDR3 controller, and is also the focus of the design. The user can directly control the system according to the parameters such as the data size, data rate, and cache depth.

The overall block diagram of the user interface design is shown in FIGURE 3, which mainly includes the read data module, the write data module and the control module.

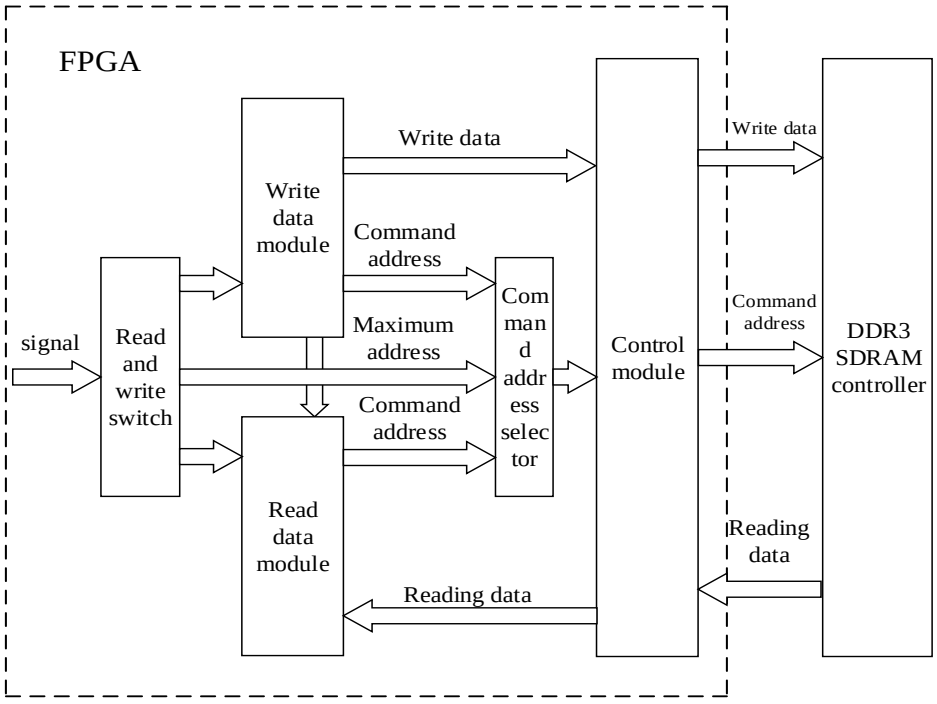


Figure 3. User interface design block diagram

Since DDR3 only contains a set of address structures, the read and write operations cannot be performed at the same time. This article uses the method of separating the read and write data channels to perform time-sharing operations on DDR3. The read and write operations of DDR3 are as follows:

(1) DDR3 Write Operation Process

When writing to DDR3, the write data module generates an address command and a data signal, which are then sent to the control module through the address command selector. After receiving the data, the control module processes the clock and transmits the data to the DDR3 controller for a series of Write operation. In addition, at the time of writing, the write address should be overwritten and registered. When the switching edge of the switching signal is detected, the maximum write data address to be registered is sent to the read data module, after which the read data module will use this address as the maximum address of the reading data.

(2) DDR3 Read Operation Process

When receiving the switching signal and requesting to read DDR3, the read data module will transmit the maximum write address received from the writing data module as the maximum reading address to the control module, which will be processed by the control module and sent to the DDR3 controller to ensure that all data are read out. The read data is processed by the control module and then returned to the read data module for analysis by the user. Through this read-write method, the amount of data of DDR3 can be changed as needed, and the system activity of the controller is improved.

Using the Quartus II 13.0 software platform, the algorithm code is edited using the Verilog HDL language. The integrated module diagram is shown in FIGURE 4.

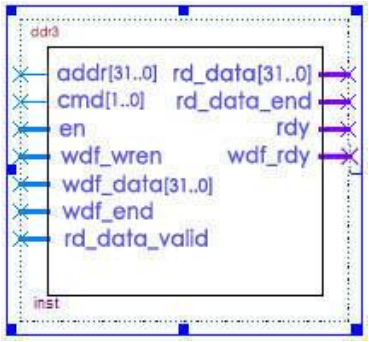


Figure 4. User interface module design

(3) DDR3 read and write test

The DDR3 read and write test is mainly to verify the consistency of the read and write operation timing, write the data into the DDR3, and then read the data in the address. As can be seen from FIGURE 5, the data written and read are consistent and meet the design requirements.

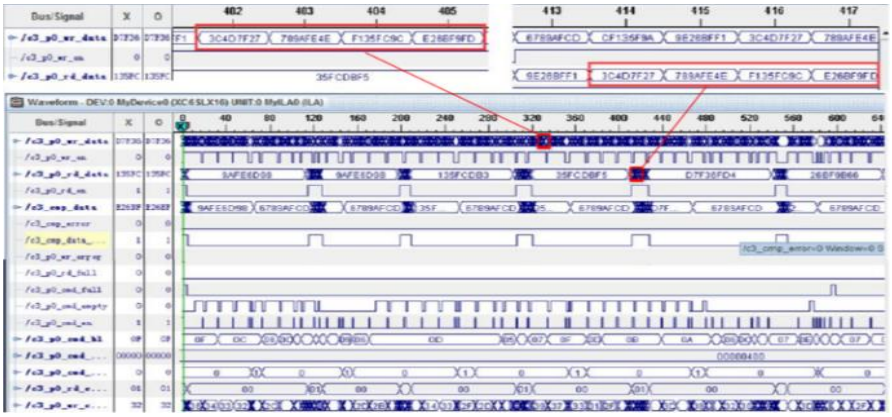


Figure 5. DDR3 data read-write test chart

3.2 USB3.0 Data Transmission Module Design

In this design, the FPGA controller communicates with the EZ-USB FX3 chip through the GPIF II interface in a synchronous Slave FIFO. By analyzing the read and write timing of the Slave FIFO interface, the state machine can be programmed to realize the relationship between the FPGA and the FX3. Interface timing. The signal connection between the FPGA controller and the FX3's Slave FIFO pin is shown in FIGURE 6.

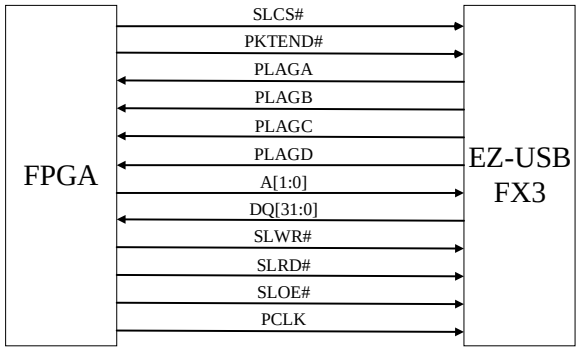


Figure 6. Pin connection diagram of FPGA controller and FX3

(1) Synchronous read operation timing from FIFO

When the FPGA reads data through the GPIF II interface, it first sends the synchronous slave FIFO address, which is refreshed when the rising edge of PCLK arrives. Then the chip select signal SLCS# is pulled low to make it active, and finally the output enable signal is pulled low. SLOE# and the read strobe signal SLRD# can initiate a read operation from the FIFO. During the read process, data is transferred from the newly addressed FIFO to the data bus DQ[31:0], providing a new value after the t CO transmission delay time (from the rising edge of PCLK).

Synchronous slave FIFO write operation timing process is relatively simple, and will not be described here.

Also use the Quartus II 13.0 software platform to use the Verilog HDL language to edit the algorithm code. The integrated module diagram is shown in FIGURE 7.

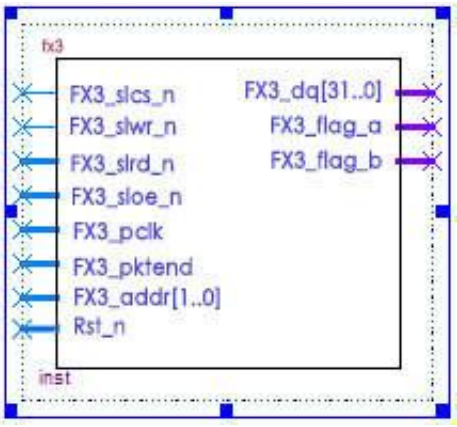


Figure 7. FX3 Controller integrated module diagram

The simulation can verify that the interface timing of the Slave FIFO controller is correct, which guarantees the reliable transmission of data. FIGURE 8 and FIGURE 9 show the timing simulation of the Slave FIFO controller.

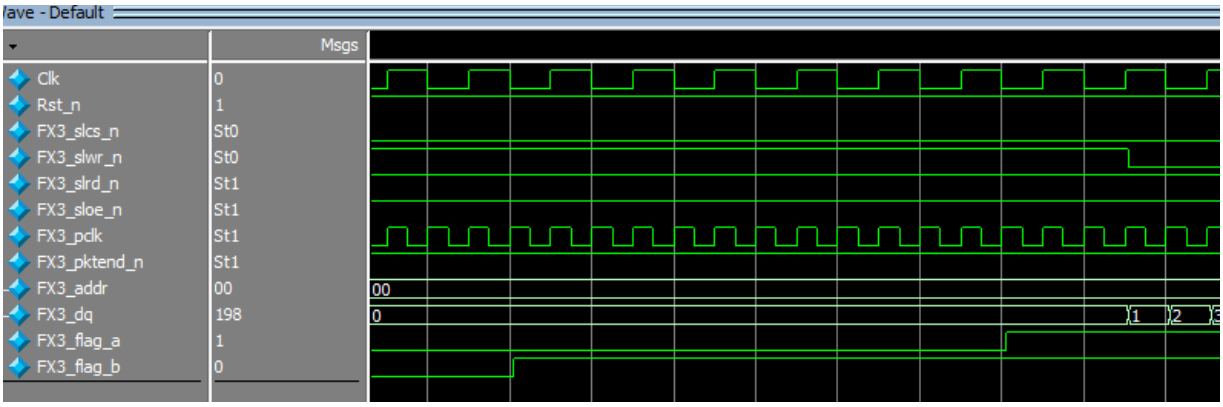


Figure 8. Slave FIFO timing simulation start waveform

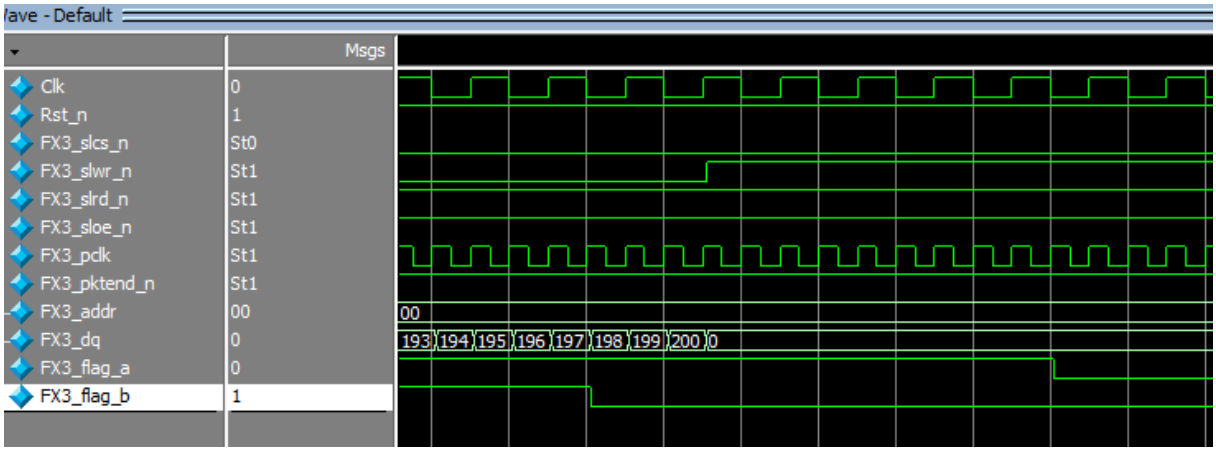
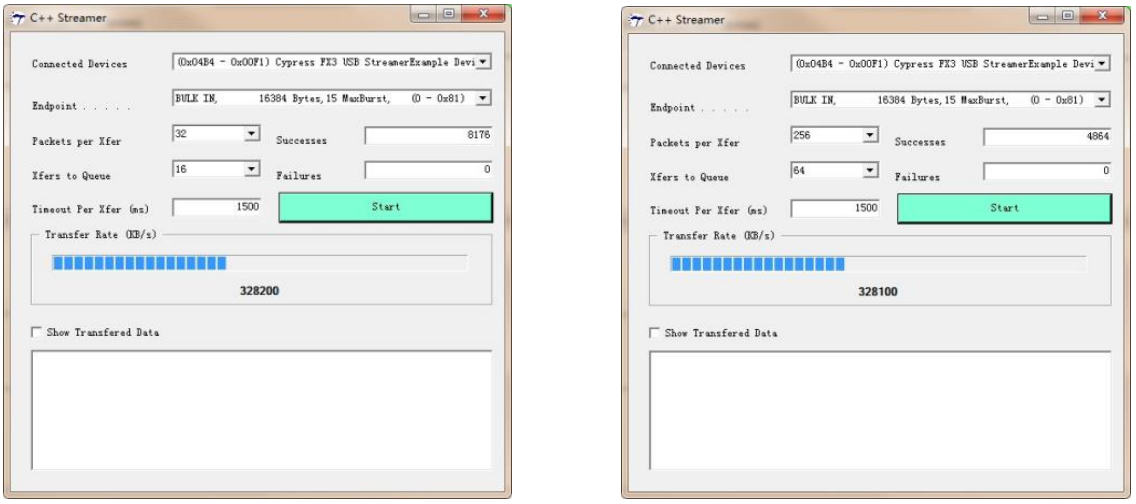


Figure 9. Slave FIFO simulation end waveform

4. System Ransmission Performance Analysis and Testing

Finally, the system performs data transmission performance tests. In this paper, the Streamer software developed by Cypress C++ is used to test the transmission speed of the system. By observing FIGURE 10, the average transmission rate of the system is kept at 328MB/s regardless of the transmission of small packets or large data packets. There is no packet loss, which is a significant improvement compared to the speed of the previously used USB 2.0 interface. At the same time, as can be seen from FIGURE 11, the system has a small logical resource occupancy. Overall, the system is designed to meet the needs of high-speed data transmission.



(a)Small packet transmission rate (b)Big packet transmission rate test

Figure 10. Data transmission rate test

Family	Cyclone II
Device	EP2C35F672C6
Timing Models	Final
Total logic elements	17 / 33,216 (< 1 %)
Total combinational functions	17 / 33,216 (< 1 %)
Dedicated logic registers	8 / 33,216 (< 1 %)
Total registers	8

Figure 11. System logical resource utilization diagram

5. Summary

This paper designs a high-speed data transmission system based on FPGA and USB3.0 synchronous Slave FIFO mode. FPGA is used as the core control chip of the system, and the timing of A/D conversion module, DDR3 data storage module and USB3.0 data transmission module are respectively performed. Logic control makes the entire system run efficiently and orderly. The design part of the DDR3 controller and the design part of the USB3.0 interface in this system are the key points and difficulties of the design. Compared with the general data transmission system, the system greatly increases the data storage capacity and increases the transmission rate. After multiple data transmission rate tests, the system has a transmission rate of 328MB/s and a logical resource occupancy of less than 1%, which meets the real-time requirements for high-speed data transmission.

References

[1]. Qian Y J, Cui K. Design of high-speed CCD data acquisition system based on FPGA and USB3.0 [C] //International Conference on Information & Communication Technology Convergence. 2015.

[2]. Jin L, Chen S, Yang Y, et al. Design and implementation of high-speed data transmission system of printing machine[J]. Microcomputer & Its Applications, 2014.

[3]. Huang J, Wang Y, He D, et al. Research of aerial imaging spectrometer data acquisition technology based on USB 3.0[J]. Proceedings of the Spie, 2016, 26.

[4]. Gong Y, Yu F. Design of high-speed real-time sensor image processing based on FPGA and DDR3[C]// IEEE International Conference on Computer & Communications. 2018.

[5]. Wang P, Wu C. Design of Dual-Channel High-Speed Data Acquisition Card Based on USB 3.0 and FPGA[J]. Advanced Materials Research, 2012, 472-475:2315-2319.

[6]. Guo-Ren L I, Zeng B, Yao-Yao D U, et al. PET System Based on USB3.0 High-speed Data Transmission[J]. Nuclear Electronics & Detection Technology, 2013, 33(1):99-78.

[7]. Janßen B, Hübner M, Jaeschke T. An AXI compatible cypress EZ-USB FX3 interface for USB-3.0 SuperSpeed[C]// International Conference on Reconfigurable Computing & Fpgas. 2015.

[8]. Bei Y, Sun Y. High-Speed Data Transmission System Design and Implementation Based on FPGA[C]// International Conference on Information Engineering & Computer Science. 2009.

[9]. Nikolaos Terzopoulos, Costas Laoudias, Fotis Plessas, George Souliotis, Sotiris Koutsomitsos, Michael Birbas. A 5-Gbps USB 3.0 transmitter and receiver linear equalizer[J]. International Journal of Circuit Theory and Applications,2015,43(7).

[10]. Fei Tao, Yue Tang, Xiaofu Zou, Qinglin Qi. A field programmable gate array implemented fibre channel switch for big data communication towards smart manufacturing[J]. Robotics and Computer Integrated Manufacturing,2019,57.