A low delay transmission method of multi-channel video based on FPGA

Weijian Fu1*, Baozhi Wei1, Xiaobin Li1, Quan Wang1,2, Xiaofei Hu1
1School of Beihang University, Beijing, China
2State Key Laboratory of Virtual Reality Technology and Systems, Beijing, China

*Corresponding author e-mail: fuwj1993@163.com

Abstract. In order to guarantee the fluency of multi-channel video transmission in video monitoring scenarios, we designed a kind of video format conversion method based on FPGA and its DMA scheduling for video data, reduces the overall video transmission delay. In order to save the time in the conversion process, the parallel ability of FPGA is used to video format conversion. In order to improve the direct memory access (DMA) writing transmission rate of PCIe bus, a DMA scheduling method based on asynchronous command buffer is proposed. The experimental results show that this paper designs a low delay transmission method based on FPGA, which increases the DMA writing transmission rate by 34% compared with the existing method, and then the video overall delay is reduced to 23.6ms

1. Introduction
As the demand for video in life becomes more and more clarifying, for example, Video data is usually transmit by video transmission system in the large business's video monitoring system. Typical video transmission system, including video input, transmission, format conversion and output[1]. When entering multiple video is worth more than 10 channels, data size reached to 3GB/s or more in transmission and conversion[2], and even more delay in the process of transmission also increases as the amount of data. It is difficult to guarantee the real-time performance of video with the traditional transmission method [1-7]. How to reduce the transmission delay of video is still an important problem.

In terms of video format conversion, as a result of multi-channel video in format conversion process takes time, for video frame rate of 25 fps, the conversion time of each frame shall not be more than 40 ms, can appear otherwise lost lead to picture frame phenomenon. At present, some scholars have tried to study the method of low - delay conversion of multi-channel video. Literature[2] to design a YUV422 RGB565 signal conversion system, because of these two kinds of color space frame structure were interlaced storage and storage way line by line, in accordance with the YUV to RGB point transformation formula, and then use line cross the way to solve two kinds of color space storage structure inconsistent problem. Literature[4] based on a kind of special demand, video signal using the divider on the FPGA IP core, 640 * 480 video signal will be cut to 480 * 480 video signal, and use their method of writing to mirror to convert the video frames. All of these methods use FPGA to video resolution or color space conversion[2,4,13,14], because the method is aimed at a specific single channel video format, in the face of the multi-channel video format conversion is still relatively high delay time.
In the transmission aspect, the literature[14] adopts PCI Express Gen1 bus to carry out the transmission of single channel video, and then extracts the effective part of video data from B.T.656 format in FPGA to the video format of YUV420. Literature [9] is the study of the high speed PCI Express bus transport method, adopted a DMA descriptor based on dynamic splicing, effectively improve the DMA read-write speed based on PCIe, but its on video processing, video format conversion is conducted in the CPU, due to its high conversion delay, cannot satisfy the data quantity is greater than 2GB/s multiplex real-time video transmission[9].

Therefore, the problems existing in the transformation and transmission of video are as follows:

(1) Large-size video caused the conversion process to become an important problem, must be designed for the time delay of multi-channel video conversion problem a low latency of the signal processing method.

(2) The quality of video data by packet loss and link to shake hands[9], the influence of time and so on factors, makes the actual transmission bandwidth is less than the theoretical one, cannot meet the demand of more large-scale data transmission.

To this end, this paper presents a multi-channel video transmission method based on FPGA. In this paper, the conversion of color space and resolution in FPGA is adopted, and a DMA scheduling method based on asynchronous command buffer is designed, which can solve the above problems better than traditional methods.

The method is detailed in the following sections. First section 2 of the overall design method is described, and contains two parts, respectively from the above two problems, this paper expounds the FPGA on the multi-channel video format conversion method and the data transmission method based on asynchronous DMA command buffer mode. Section 3 according to the design of experiment, first of all introduces the experimental environment, and then respectively for multi-channel video transmission performance and video delay test. Finally, we give the corresponding experimental results and the conclusion.

2. Proposed Method

This paper is based on the self-developed video transmission system, as shown in figure 1:

![Video transmission system architecture based on FPGA](image)

2.1. Multi-channel video format conversion method based on FPGA

For multi-channel video in the process of format conversion, the format conversion method of video is presented in this paper.

Video format conversion is usually divided into three categories [13]. One is that the input of video is the same as the target video color space, but the resolution is different. There is no need to do a lot of operation in the FPGA. There are already many scholars studying this aspect [2, 14]. The
second is to enter video into RGB color space, which can be converted in a fast format using the IP Core (RGBtoYCrCb Core) inside the FPGA. The third kind is the input video to YUV420, due to the commonly used video DAC is only supported YUV422 format video output[7], in order to make the video processing system can adapt to YUV420 format, need to convert YUV420 to YUV422 in FPGA.

Therefore, this method mainly focuses on the study of the color space conversion of YUV422 on FPGA, and the frame structure of the two color Spaces is shown in fig. 2:

![Figure 2. YUV420 to YUV422 color space conversion diagram](image)

YUV420 frame structure of Y, U, V, listed three color components of each complete pixels Shared component four U, V, Y component used independently, so the U and V regional area to Y a quarter; YUV422 frame structure for two pixels to share a component of the U, V, Y components independently, in accordance with the order of the U, Y, V, Y lined, so figure 2 YUV422 video’s frame width is twice as the width of resolution.

Assuming the top right-hand corner of the video frame address is 0, in view of the input video any one address in each frame \( x \) ( \( x \) for natural Numbers), the corresponding output video address mapping function is \( f(x) \) in the frame, the resolution of the input video frame width is \( W_i \) and high for \( H_i \), according to the principle to select the output video frames is \( W_o \) high is \( H_o \). As shown in formula (1):

\[
 f(x) = \begin{cases} 
 2x + 1 & x < W_i * H_i \\
 W_i + (x - W_i * H_i) * 2 \leq W_o & W_i * H_i \leq x < W_o * H_o \\
 W_o + \left(1 - \frac{5}{4}\right) * 2 \leq W_o & x \geq W_o * H_o 
\end{cases}
\]

For example, the transformation of frame D1 video (YUV420, 704* 576) frames to SD video (YUV422, 720* 576) frames is shown in figure 4:

![Figure 3. YUV420 to YUV422 color space conversion diagram](image)

However, the FPGA cannot detect video format by itself of the input video. In response to this problem, an indirect method for detection of video is designed. PC software can identify a particular
pathway video format according to the camera driver, so we change the first bytes per frame to a FLAG. Each frame of the FPGA will detect the FLAG and perform parity check. According to the common video format and type, the corresponding video resolution, format and frame rate of the flags in detail are shown in Table 1:

| Data Bit | Name      | Value | Function                  |
|----------|-----------|-------|---------------------------|
| BIT[1:0] | RATE_FLAG | 00    | 25 frame per second        |
|          |           | 01    | 25 frame per second        |
| BIT[4:2] | RES_FLAG  | 000   | D1                        |
|          |           | 001   | SD                        |
|          |           | 010   | HD                        |
| BIT[6:5] | CSF       | 00    | YUV420 color space         |
|          |           | 01    | YUV422 color space         |
|          |           | 10    | RGB565 color space         |

FPGA is used to convert video format, and the traditional method [9] of video format conversion on CPU can reduce the delay of video format conversion. Especially for video because more multi-channel video, due to its large amount of data, video format variety [6], the use of FPGA parallel computing feature [8], to be more efficient to complete the video format conversion.

2.2. Data transmission method based on asynchronous DMA command FIFO

In this paper, a DMA scheduling method based on asynchronous command buffer is proposed in this paper [8]. Makes the CPU and FPGA for the DMA operation during preparation for the parallel to the DMA operation, no need to wait for the FPGA again after returning to interrupt for the next step operation, reduce a DMA operation of waiting time, increased the transmission rate indirectly. As Shown in Figure 4:

![Figure 4. DMA scheduling method of transmission based on asynchronous FIFO](image)

Compared with the traditional method [5], this paper avoids the delay of waiting, thus improving the transmission rate of PCIe. The traditional DMA scheduling mode, which adopts the CPU control method, reads and writes the actual bandwidth of [6] only. When the processed data object is video at more than 25 frames per second, this method has a significant amount of time wasted on the time that the interrupt response and wait for the interrupt return [7].
In order to further reduce the latency of the host and FPGA, the speed of PCIe transmission is improved. Method with the structure of the ping-pong Memory access mechanism, respectively in the Memory and DDR3 copy a piece of space, the host and the FPGA can be used directly in a frame is ready to data, to ensure that for the DMA operation, read and write Memory not conflict. This method effectively reduces the waiting time on the CPU side, thus indirectly enhancing the transmission rate of PCIe.

3. Experimental Results

This article uses the embedded multiplexed multi-channel video conversion and transmission system is based on the design of hardware interface card, the card on the integrated piece of FPGA, 8 pieces of DDR3 and 16 video DAC, FPGA uses Xilinx company K7 series, model is XC7K325T. The hardware card is connected to the high performance PC via the PCIe Gen2 X4 interface. The video in the upper computer software is 16, including 2-channels HD (1080 * 720), 4 SD (720 * 576) and 10 D1(704 * 576). In addition to the image compression sampling format used by 4-channels SD video for YUV422, the other 12 channels video adopts the compressed sampling format of YUV420.

Based on the above experiment setting, the following are respectively the DMA write transmission rate test and video delay test:

3.1. DMA write transmission rate test

DMA operations include DMA reads and DMA writes in two ways. Since the method used in this paper only involves PCI master equipment (PC) to write video data from the device (FPGA), we only test the transfer rate of DMA writing. In the test, the host repeatedly sends a different number of commands to the FPGA and records the execution time. The DMA write transmission rate $V_w$ is calculated by formula (2). In (2), $N$ represents the number of commands to execute, and $B_s$ is the block size in a command. The total execution time of the command can be calculated by multiplying the CPU clock cycles used by the clock cycle $t_c$.

$$V_w = \frac{N * B_s}{n * t_c}$$

(2)

The peak transmission rate of this method is 3262 MB/s, which is 85.4% of PCIe theoretical bandwidth. Table 2 shows the experimental results of the comparison of DMA writing transmission rate between proposed method and others:

|                  | Literature[15] | Literature[8] | Proposed method |
|------------------|----------------|---------------|-----------------|
| DMA write rate/(MB·s⁻¹) | 2156           | 2622          | 3262            |
| Bandwidth occupancy | 51%            | 65%           | 85.4%           |

Literature[15] and literature [8] uses the DMA scheduling of CPU master, including literature [15] uses a PCIe Gen1X2 interface, the DMA write rate of 539 MB/s, used in the literature [8] PCIe Gen1 X4 interface, the DMA write rate of 1311 MB/s. Compared with Gen2 X4, the method of writing the above three kinds of writing rate is compared with the traditional method, and the transmission rate is improved by 34% compared with the traditional method.

3.2. Video delay test

The method of video transmission and conversion is designed in this paper. According to the asynchronous DMA command method, the overall delay of video can be divided into three parts: CPU processing delay, PCIe transmission delay and FPGA processing delay. We shall have a PC start dealing with the frame of the video time for $t_s$, PC sends the timing of the DMA requests for $t_d$, PC player put into FPGA to return to the timing of the interrupt for $t_e$, the three time points can be achieved by PC assistant program is added in the driver interface, shown in figure 5, according to the DMA operation process, the overall delay for single frame video was $t_e - t_s$, PCIe transmission time for $t_e - t_d$. Table 3 compares this method with other literature methods.
Table 3. Video delay test comparison

|                     | Literature[9] | Literature[15] | 本文方法 |
|---------------------|---------------|----------------|--------|
| CPU processing delay| 28.8ms        | <1ms           | <1ms   |
| PCIe transmission delay | 36ms      | 40.5           | 26.8ms |
| FPGA processing delay | <1ms       | 1.0ms          | 1.8ms  |

According to proposed method, the application functions as write 25 frame transmission delay and record the results, the average for single frame video always delay to an average of 28.6 ms, the average transmission time for PCIe 26.8 ms, less than 40 ms frame period, will not affect video fluency. Since this method does not do any video processing on the CPU, the CPU delay can be ignored.

Literature [9] is used in the CPU for video format conversion, the method of 8-core 2.8 GHz frequency according to the total volume of 2.18 GB/s of every frame image conversion need to consume 28.8 ms, PCIe transmission takes 36 ms, FPGA side don't do processing delay is negligible, the total delay of 64.8 ms. The document [15] uses the CPU controlled DMA transmission method, PCIe transmission takes up 40.5 ms, FPGA terminal processing delay is 1.0 ms, and the total delay is 41.5 ms. The above two methods are based on the experiment setting, and the delay is larger than the frame cycle 40ms. Compared with traditional methods, video delay can be reduced by proposed method.

4. Conclusion
In this paper, a low-delay transmission method based on FPGA is proposed. In this way, the low delay transmission of video is able to support data size greater than 2GB/s. At the same time, a transformation method of video color space on FPGA was proposed, and the parallel computing power of FPGA was used to effectively reduce the delay of video conversion process. The experimental results show that the method adopts the asynchronous command buffer FIFO mechanism, improves the write transmission rate of DMA effectively, which is 36% higher than the traditional method. In the case of the same multi-channel video input source, the low-delay multi-channel video transmission method proposed is reduced to 23.6ms compared with the traditional method. But now, only YUV420 and RGB color space conversion to YUV422 can be supported, even more in the video monitoring application scenario, the multi-channel video usually contains more color space, this compatibility for multi-channel video transmission system is also very important, we will continue our studies for multiple variety transmission of video format compatibility.

Acknowledgments
This work was financially supported by The Natural Science Foundation of China fund(61272347).

References
[1] Wang Y, Su B, Qiu W. Super-resolution video restoration system based on ZedBoard[J]. Journal of Applied Optics, 2015, 36(4):537-542.
[2] Bensaali F, Amira A. Design and Efficient FPGA Implementation of an RGB to YCrCb Color Space Converter Using Distributed Arithmetic[J]. Lecture Notes in Computer Science, 2004, 3203:991-995.
[3] Zhu E J, Liang H L, Zhang T. Research on Video Format Conversion Algorithms[J]. Video Engineering, 2006.
[4] Zhou X, Chen J J, Dou L. Study of processing and displaying for video signal with special resolution[J]. Optoelectronic Technology, 2016(3):195-199.
[5] Zhang T. Video format transform based on DSP and FPGA[J]. Foreign Electronic Measurement Technology, 2013.
[6] Gao J, Zhang Z. Design of Digital Video Processor System on FPGA[J]. Video Engineering, 2011.

[7] Qian hua H E, Tang Z H, Tiao L I. FPGA-based Video Conversion System[J]. Microelectronics & Computer, 2003.

[8] Kan X B. High speed image acquisition system design based on PEX8311 of PCIe bus[J]. Electronic Design Engineering, 2011.

[9] Li S L, Jiang H X. Design of DMA controller for multi-channel transmission system based on PCIe[J]. Journal of Computer Applications, 2017, 37(3):691-694.

[10] Chen-Yang L I, Zhi-Jun X U. Based on PCI main line’s DSP video frequency gathering actuation design[J]. Manufacturing Automation, 2010.

[11] Jiang H X, Liu T S, Li H Y, et al. A High-efficiency Data Transmission Method Based on SRIO in FPGA+DSP Heterogeneous Video Processing System[J]. Chinese Journal of Computers, 2015.

[12] Lecroy T. Teledyne LeCroy Announces New TeleScan PE Utility supporting PCI Express 3.0 Specification[J]. Teledyne Lecroy.

[13] D. L. Donoho. Compressive sampling. IEEE Trans. IT, 52 (4):1289–1306, April 2006.

[14] Jia J F. Design of real-time video gathering system based on PCI Express bus[J]. Electronic Design Engineering, 2010.

[15] Gong J, Wang T, Chen J, et al. An efficient and flexible host-FPGA PCIe communication library[C]// Field Programmable Logic and Applications (FPL), 2014 24th International Conference on. IEEE, 2014:1 - 6. van der Geer, J.A.J. Hanraads, R.A. Lupton, The art of writing a scientific article, J. Sci. Commun. 163 (2000) 51-59.