A 8ms delay image transmission system and its applications

著者	Chen H., Nakao M., Miyama Masayuki
journal or publication title	IDW '07 - Proceedings of the 14th International Display Workshops
volume	2
page range	1425-1428
year	2007-01-01
URL	http://hdl.handle.net/2297/9934

A 8ms Delay Image Transmission System and Its Applications

H.Chen, M.Nakao, M.Miyama*
Eizo Nanao Corporation, *Kanazawa University

Abstract

Based on the real-time prototype hardware proposed on SID2007, We've finished developing a real-time image transmission system. This system achieved low system delay(8msec) and high image quality for both desktop images and motion pictures. In this paper, we explained the whole system and CPU control flow in details. System delay was analyzed in encode and decode side. In the end, some possible application fields were introduced.

1. Introduction

The JPEG2000 standard is becoming an increasing popular coding format for a variety of applications that require efficient storage and scalable transmission of images and video, because it offers efficient intra-frame compression, low latency and the amount of compression applied can easily scale to available bandwidth. Unlike competing MPEG formats, there is no motion estimation. That means each frame stands alone, which offers a way to circumvent predicted delays introduced by MPEG which use bi-directional prediction. JPEG 2000's compression output rate is deterministic. With the compression scheme, it's easy to set a threshold, such as ~35Mb/s for IEEE 802.11a/g, and if conditions change making less bandwidth available, smaller file sizes are instantly available.

As our system will be mainly used for PC desktop based applications, quick response is more important than other factors. So we take low delay as first criteria for this system.

On SID2006, we have proposed a new and simple method [1] to support environments which need the following features.

- Low delay and real-time
- Adaptability to change of transmission bandwidth
- Simple architecture

In this method, we discussed to transmit typical computer desktop images including motion pictures of 1280 x 1024 @60fps with low delay at the speed of 35Mbps. The simulation result of our proposed method showed lower delay and higher image quality compared with MPEG-2's

On SID2007, we have proposed a hardware prototype [2] to implement this method.

In this paper, we will introduce detail architecture of our finished system and CPU control flow based on this hardware architecture. After that, system latency (including hardware latency, software latency and network latency) will be analyzed using this system. Two possible applications for this system will be discussed finally.

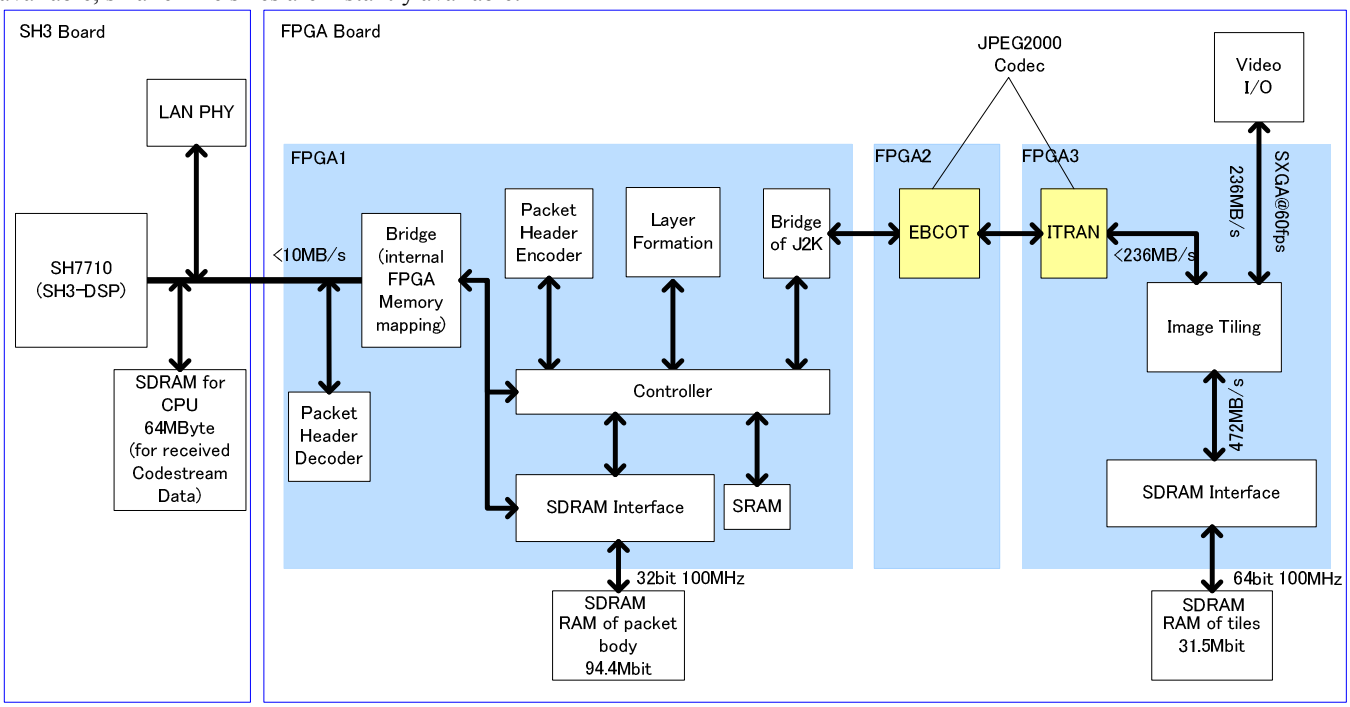

Fig.1 System Block Diagram

2. System Architecture

We have finished developing a real-time system to evaluate performance on FPGA board. The JPEG2000 Codec will finally be implemented in ASIC.

2.1 Hardware Architecture

As is shown in Fig.1, The whole system consists of one Renasas SH3 board and one FPGA board which has three Xilinx XC4VLX100 FPGAs on it.

This system supports two modes: encode and decode mode. The function of these modules in Fig.1 will be explained in details.

- **Bridge of Internal FPGA**

This bridge receives read/write request from CPU at 66MHZ, then convert to internal FPGA mapping, output read/write request at 80MHz for internal FPGA.

- **Control**

Controls encoding and decoding flow to implement enhanced image progressive method which is proposed in SID2006. In encoding mode, it controls Layer Formation and Packet Header Encoder to make layer formation and generate packet headers. And it stores packet headers and bodies to SDRAM. In decoding mode, it read out pass information from SRAM and packet body data from SDRAM, the send it to bridge of J2K (the name of JPEG 2000 codec).

- **Layer Formation**

This module is only used in encode mode. It divides pass data generated from JPEG2000 Codec into several layers using Q value (distortion/rate), each layer represents one image quality.

- **Packet Header Encoder/Decoder**

Make/parsing packet header using/into pass length, pass number, codeblock inclusion information.

- **Bridge of J2K**

In encode mode, read pass data and distortion value for each pass generated from EBCOT. In decode mode, read pass data from SDRAM and send to EBCOT for decoding

- **SDRAM Interface**

SDRAM Interface is a high performance controller for SDRAM access, it support multiple port access

- **ITRAN**

ITRAN is for Image TRANsformation.

In encode mode, color transformation, DWT transformation and quantization[3] are processed. In decode mode, inverse quantization, inverse DWT transformation and inverse color transformation are processed.

- **EBCOT**

EBCOT is for Embedded Block Coding with Optimal Truncation. In encode mode, it generates compressed pass data using quantized image data. In decode mode, it generates image data for inverse quantization using compressed pass data.

- **Image Tiling**

Image Tiling is an interface between frame and tile format.

In encode mode, it receives video signal from Video I/O, calculated CRC32 value for each tile, and compares CRC32 value at same tile position of previous frame, if CRC32 is identical, this tile will not be sent to ITRAN for compressing. This method will reduce processing data for desktop oriented application which has small change between frames.

In decode, it receives tile image data from ITRAN, stores to SDRAM, then reads out in frame format for displaying.

FPGA encoding and decoding flow has been explained in details in [2]. In our system, the whole image transmission system consists of two sub-systems which in encode and decode mode respectively. The two sub systems use wired LAN or wireless LAN to connect each other, which has speed of 35Mbps. In fig.1, the two parts in yellow is JPEG2000 codec, which will be implemented in ASIC this year @0.18μm with the area of 10.7mm^2 160MHz[4]. But in this FPGA version, it only achieved 40MHz because of FPGA speed limitation.

2.2 CPU Control Flow

In this system, CPU mainly controls encoding and decoding flow. Fig.2 shows the encode control flow, Timer Task has a counter to generate frame timescale for buffer status reading. Buffer status is double flags for each tile respectively, if buffer status flags changed compared with previous reading, Q value (distortion/rate for rate control) will be read from FPGA and sorted. After that, JPEG2000 packet header and packet body will be read out from FPGA based on the sorted order. And make our own triplet packet format designed for more efficient TCP/IP transmission. TCP/IP Send Task will send packet if the buffer has new data. Decoding flow is similar with encoding flow.

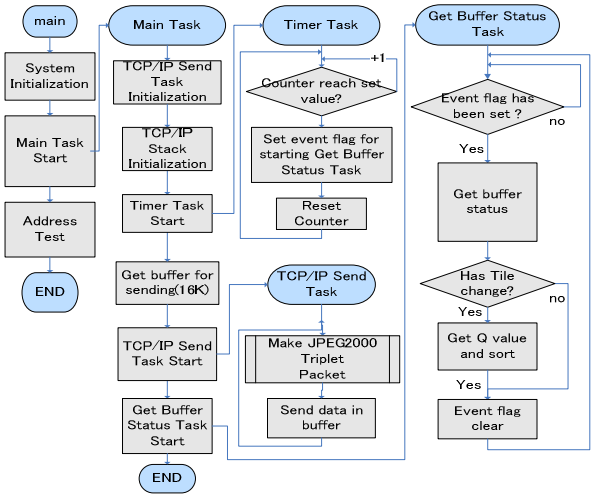

Fig.2 Encode Control Flow

3. Result

3.1 Hardware Implementation Result

JPEG2000 Codec LSI is design using 0.18μm process. As is shown in Tab.1, it has a gate count of 832k, operating frequency is 160MHz. Peripheral circuit means the logic except JPEG2000 Codec which is shown in Fig.1. It is implemented in Xilinx XC4VLX100 FPGA. As JPEG2000 Codec LSI is still under manufacturing now, so we replaced this part using FPGA for whole system verification. The operating frequency is reduced to 40MHz because of FPGA speed limitation. So input data rate should also be reduced consequently. Here we have made input video date rate configurable. In this system, sample frame rate is set to 1/4 of original frame rate or frame size is set to 1/4 of original frame size.

	JPEG2000 Codec LSI	Peripheral Circuit
Gate count	832k @0.18μm process	14k LUT (14% of Xilinx XC4VLX100)
Frequency	160MHz	40MHz

Tab.1 Hardware Information

3.2 System Delay

	Delay
Hardware processing of transmitter (video input, image tiling, JPEG2000 encode, packet header making)	2.5ms
Software processing of transmitter	<1.5ms
Network transmit	3ms
Software processing of receiver (packet header decoding and packet data writing to FPGA)	0.2ms
Hardware processing of receiver (JPEG2000 decode, tile update and video output)	0.2ms
Total	<8ms

Tab.2 FPGA Utilization

As is shown in Tab.2, the total system delay is including:

(1)Hardware processing of transmitter

This delay is video input, image tiling, JPEG2000 encode, packet header making, as hardware is parallel processing, it is about 0.15 frame time(2.5ms@60fps)

(2)Software processing of transmitter

As is shown in Fig.3, 70% CPU processing load is TCP/IP transmitting, so the other processing time of software is 3ms x 30%/70%<1.5ms,

(3)Network transmit

As is shown inFig.4, network transmit delay is less than 3 ms

(4)Software processing of receiver

Software processing of receiver is including packet header decoding and packet data writing to FPGA.

From the real time processing test, average delay for one packet is 4.4us, there are less than 90 packets in one tile, so delay for one tile will be less than 0.1ms. average time for writing one packet data to FPGA is 7.0us, so delay for one tile will be less than 0.1ms.

(5)Hardware processing of receiver

This delay is including JPEG2000 decode, tile update and video output, the average delay is 1/80 frame time(0.2ms@60fps).

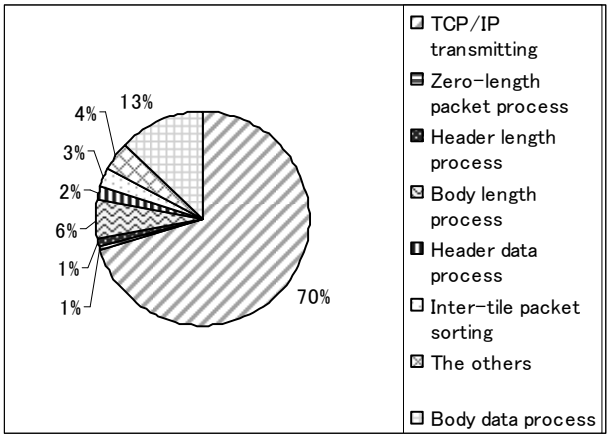

Fig.3 CPU Processing Load

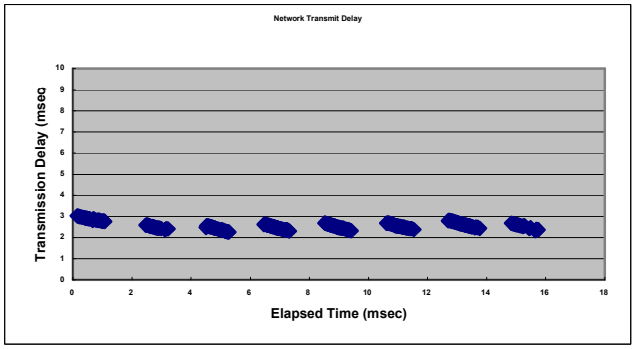

Fig.4 Network Transmit Delay

So the total system delay is less than 8ms from video input to video output, much better than MPEG-2 case, which spend at least 4 frame time by bi-directional prediction or 2 frame time by mono-directional prediction. The image is separated and processed tile by tile and the delay is minimum, so it is suitable for the remote desktop applications. As is shown in Tab.3, the result of the image quality using software simulation model was shown at SID2006[1] and was better than MPEG-2.

	MPEG-2	Our method
Image Quality	~35dB	~40dB
System Delay	> 100ms	~8ms

Tab.3 System Delay and Image Quality Comparison

4. Applications Fields

Recently, long distance (>20m) display demand is increasing, the normal solution is special DVI cable using optical fiber, but it is very expensive and not easy to drive multiple display. Our system can be used for this kind of application by connecting PC and monitor using LAN cable or USB cable which is very popular and cheap in many places. The cost will be reduced and simple for multiple display connection.

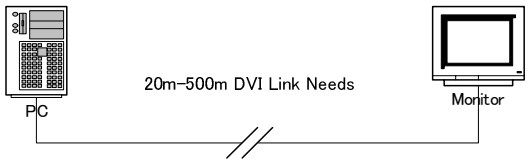

Fig.5 Application of Long Cable

Another possible application of our system can be cable free presentation system using wireless network as is shown in Fig.6

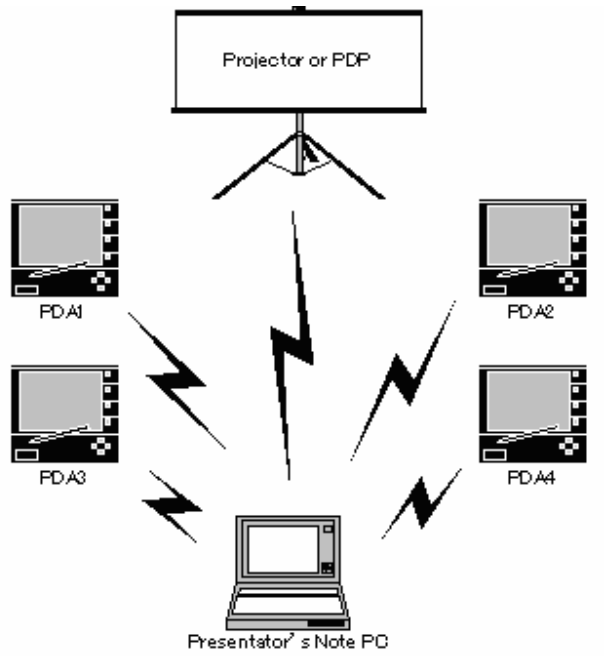

Fig.6Application of cable free Presentation System

5. Conclusion and Future work

In this paper, we have introduced a low delay image(8ms) transmission system with high image quality compared with MPEG. System delay is analyzed in details. And some application fields for this system are introduced.

In future, we will implement a JPEG2000 codec chip to reduce system cost and make these applications possible for product. And we are planning to use inter-frame compression to achieve better image quality.

6. Acknowledgments

The authors wish to acknowledge the aid and encouragement of Prof. Matsuda, Dr. Miyama and the students of Kanazawa University, Kanazawa, Japan.

7. References

[1] M.Nakao, et al, "High Quality Image Compression System for Low Delay and Real-Time Wireless Transmission", SID2006, Jun. 2006.

[2] M.Nakao, et al, "Low delay and Real-time Image Transmission Hardware for Remote Desktop", SID2007, Jun.2007

[3] "Information Technology; JPEG 2000 Image Coding System – Part 1: Core Coding System", ISO/IEC 15444-1, Aug. 2000.

[4] M.Miyama, et al, "A 158MS/s JPEG2000 Codec with a Bit-plane and Pass Parallel Embedded Block Coder", PSC 2007, Nov., 2007