A Parallel Image Processing Platform based on Multi-Core DSP

Guodong Wang[1,2,*]
[1]*School of Computer Science and Engineering, Northwestern Polytechnical University*
Xi'an, PR, China
wwggddd@163.com*

Xiaojian Liu[2]
[2]*AVIC Computing Technique Research Institute*
Xi'an, PR, China

***Abstract*—In order to achieve real-time processing of the image with high frame rate and high resolution, the image processing system has been required to have higher data transmission bandwidth and data processing capacity. Multi-core DSP is used as the key processor to solve the problem of data transmission and processing bottleneck, which the current image processing platforms are facing. The characteristics of FPGA's hardware parallel processing are made use of to realize the preprocessing of multi-channel video. The communication interface of high-speed serial RapidIo is designed to complete real-time transmission of the video. Analyze the multi-core scheduling characteristics to ensure the real-time processing and test the system's performance based on related algorithm.**

Keywords—real-time processing; hardware parallel; high-speed serial; multi-core scheduling

I. INTRODUCTION

With the rapid development of image acquisition equipment, high-resolution images bring a great deal of data processing and a variety of complex algorithms bring large amount of calculation. Therefore, in some embedded applications that have harsh real-time requirements of system image processing speed often becomes a bottleneck affecting system's accuracy and usability. In order to achieve fast and intelligent image processing, image processing system is required to have powerful data processing capabilities and high flexibility. DSP with high-speed and high-performance is the key to achieve rapid processing[1].

This paper designs a real-time image processing platform based on TMS320C6678 multi-core DSP in order to resolve the problem between complexity of image processing algorithm and real-time processing, build parallel framework to realize multi-core performance and promote high-performance image processing platform.

II. C6678 AND ITS STRUCTURE

The TMS320C6678 is an 8-core floating-point DSP in the C66x family that can run at a core speed of up to 1.25GHz.A single instruction cycle can perform 32 fixed-point data operations or 16 floating-point data operations. The entire chip provides the computing capability for 320 GMAC fixed-point and 160 GFLOP floating-point[2].

Each C66x CorePac of the TMS320C6678 device contains a 512KB level-2 memory (L2), a 32KB level-1 program memory (L1P), and a 32KB level-1 data memory (L1D). The device contains a 4MB multi-core shared memory for data interaction and provides DDR3 controller interface that can directly access the 8 GB addressing range.

C6678 has four SRIO high-speed serial ports, two PCIE interfaces, 16-bit external memory EMIF interface, network port, SPI bus, I2C bus, UART, TSIP, GPIO and other interfaces. These interfaces interact data with each processor via on-chip high-speed interconnect bus[3].

978-1-5090-5507-4/17/$31.00 ©2017 IEEE
ICIS 2017, May 24-26, 2017, Wuhan, China

III. SYSTEM STRUCTURE AND FUNCTION

The system consists of C6678 multi-core DSP processor, FPGA coprocessor, DDR3 controller, Ethernet interface, Level converter, I2C and SPI bus interface, RapidIo high-speed serial bus, memory, video input and output module, DSP power supply module, Encoder and Decoder, auxiliary circuits and other functional blocks. System block diagram shown in Fig.1, in which:

Rocket I/O module is embedded in FPGA. The module uses high-speed serial I/O technology that provides 100Mbps~3.2Gbps one-way transferring speed and supports a variety of high-speed serial communication protocols.

The high-speed serial bus technology of RapidIo is a high-performance and low-pin-count crossbar switch interconnect technology based on packet switching technology[4].It supports 250MHz, 500MHz, 750MHz and 1GHz frequency and its transmission performance can be from 1Gbps to 60Gbps. 1x/4x serial interface is suitable for long-distance transmission using clock and data's synchronization technology and 8B/10B encoding mechanism. It supports 1.25GHz, 2.5GHz and 3.125GHz baud rate.

The auxiliary circuits include Reset, Monitoring, Real-time clock, PMBUS debug interface, Power supply and other circuits.

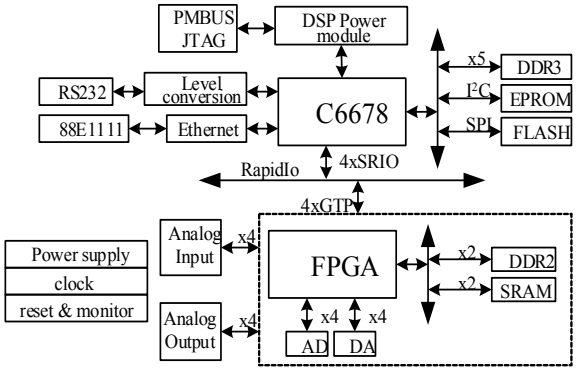

Fig.1. Block diagram of system composition

When system receives external analog signals, FPGA mainly completes the pre-processing work including signal's format conversion and so on and the multi-core DSP processes the back-end data for the front-end's real-time output.

IV. KEY FUNCTIONAL CIRCUITS DESIGN AND IMPLEMENTATION

A. DSP Power Supply Design

C6678 can work at a speed of up to 10GHZ, so ordinary power supply can't meet the requirement. The design uses 12V power supply, and The DSP core voltage circuit consists of a UCD9222 and a UCD7242 with two voltage output, as shown in Fig.2.

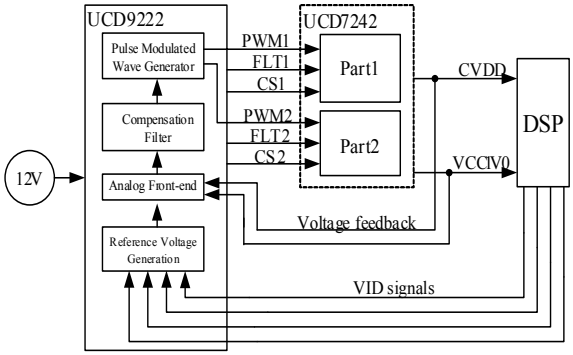

Fig.2. Power structure of DSP core

UCD9222 is a digital PWM controller that can control two independent power outputs[5]. UCD7242 is a driver chip that is fully compatible with the UCD9222, and it can generate two independent power supplies. UCD7242 receives the pulse width modulation wave of UCD9222 as input, by controlling the time of the MOSFET switch's opening and closure to control the output voltage. Then the output voltage returns to UCD9222 through the feedback signal and compares with the internal reference signal of UCD9222 to facilitate that the power control chip can make timely response to output voltage offset. To prevent the feedback voltage from exceeding maximum value of the reference voltage, adding a voltage dividing network to the input port of analog front-end. UCD9222 support VID interface and DSP's VID signals give feedback to the UCD9222 to make real-time adjustment to the voltage.

B. FPGA Logic Design

FPGA is a key device to ensure the realization of the chip's function control and processing algorithms. The Virtex-5 series chip of this system has rich logic resources and unique MGT high-speed interface, which ensures the high performance of complex image processing logic development and high bandwidth video's real-time

Transmission. The function of FPGA mainly includes three parts:

- To complete the C6678's power-on reset timing control;
- To achieve the configuration of video codec chip and the input and output videos' protocol analysis and reconstruction;
- To construct RapidIo controller and realize high-speed real-time transmission of four channels video.

FPGA logic functional block diagram is shown in Fig.3. FPGA detects and analyzes the input video, extracts the Y, Cb and Cr components of the effective pixels, converts them into RGB space and sends the RGB images to the DSP through RapidIo for subsequent processing. At the same time, FPGA initiates a RapidIo read operation as the main device, then reads the processed effective pixel data from the DSP memory according to the video's output timing, performs reverse format conversion and returns to the original digital video timing output.

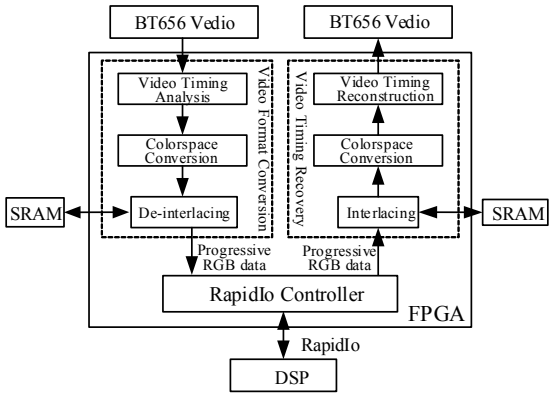

Fig.3. Functional diagram of FPGA logic circuit

C. *RapidIo Transmission Mechanism Design*

This design uses the 4x RapidIo controller in accordance with the SRIO v1.3 and the data transfer at a speed of up to 2.5Gbps to meet the bandwidth requirements of multi-channel video synchronous transmission. In order to realize the real-time reading and processing of PAL video, the 4 buffers mechanism is adopted to process the parallel flow of video stream. Four buffers are allocated to each channel video in the DDR3 memory and each buffer stores one frame of image. Image processing pipeline structure is shown in Fig.4. First, while FPGA is writing the collected image data to buffer 0, DSP is processing image of buffer3 and reading the data that has been processed then output it, and buffer 1 is waiting. Then, when the current frame image has been completely written into the buffer 0, the edge trigger signal is generated by two GPIO control signals to inform DSP. When the next frame image is inputted, the effective pixel is being written to buffer 1, while pixel flow of buffer 3 is being read and DSP is processing the image of buffer 0, and so on. It can achieve the real-time processing for image sequences and give the DSP 40ms to process each frame.

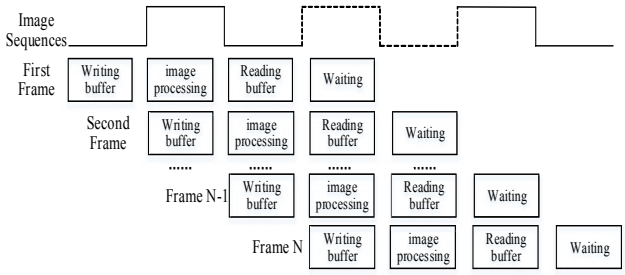

Fig.4. Image processing flow diagram of four buffers

RapidIo's read and write cycles are controlled by GPIO [13:12] that triggers the next operation on the edge. In this design GPIO [13:12]'s read and write relationship with FPGA is shown in Table I. After receiving the GPIO interrupt, the DSP recognizes the interrupt and processes the data of corresponding buffer, so it can realize the loop flow operation for the four buffers by combining with the read and write operation initiated by the FPGA proactively.

TABLE I. GPIO Pins Status Identification

GPIO [13:12]	FPGA Writing	DSP Processing	FPGA Reading	Waiting
00	Buffer 3	Buffer 2	Buffer 1	Buffer 0
01	Buffer 0	Buffer 3	Buffer 2	Buffer 1
10	Buffer 1	Buffer 0	Buffer 3	Buffer 2
11	Buffer 2	Buffer 1	Buffer 0	Buffer 3

V. MULTI-CORE SOFTWARE DESIGN AND PERFORMANCE EVALUATION

A. OpenMP Model

OpenMP is a standard model based on a shared storage system architecture that integrates compiler directives and function libraries. With the traditional C/C++ and Fortran's collaboration, OpenMP has the characteristics of high flexibility, simple and practical and short design cycle. The shared memory model architecture has multiple processors at the bottom and the processors access the same memory, so they can interact with each other through shared variables [6].

The multi-core DSP TMS320C6678 can run OpenMP parallel programming model on SYS/BIOS embedded operating system[7]. The hierarchical relationship between them is shown in Fig.5.

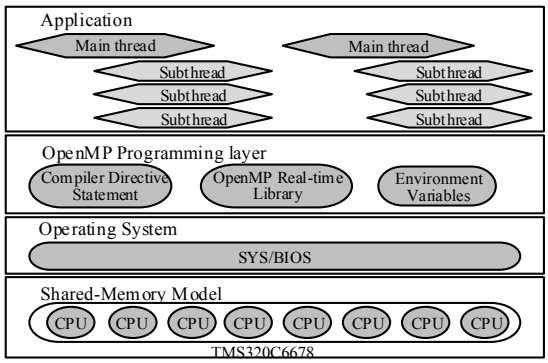

Fig.5. Schematic diagram of the hierarchical relationship between TMS320C6678 and OpenMP

B. Parallel Algorithm Design

In this paper, the iterative bilateral filtering is used to enhance the image and a parallel bilateral filtering scheme is proposed[8]. Iterative bilateral filter is a separable filter that requires a line filter and a column filter to the image. First, parallel processing is performed on the row direction and a region that needs to be filtered is allocated to each core in accordance with the multi-core method, then is sent to each core for parallel processing. After processing the row direction, divide the column direction into parallel, then send to each core for parallel processing to get the final filtered output. It is achieved in CCSv5.5 in accordance with the OpenMP model programming approach. The pseudo code of the parallel process is as follows:

```
omp_set_num_threads(number of cores);
step 1: Parallel row filter
for (total rows divided by number of cores)
    for (each column)
    calculate a;
    y[i]=(1-a)*x[i]+a*y[i-1];
    end
end
step2: Parallel column filter
for (total column divided by core number)
    for (each row)
    calculate a;
    y[i]=(1-a)*x[i]+a*y[i-1];
    end
end
```

After obtaining the base layer by iterative bilateral filtering, the detail layer is obtained in parallel using the multi-core mechanism and OpenMP model, then the corresponding detail enhancement is carried out to obtain the final output image.

VI. EXPERIMENTAL RESULT

The platform used in the experiment is the C6678 image processing platform designed in this paper. The running frequency of the processor is set to 1GHz and the compiling environment is CCSv5.5 of TI Company. Using the above iterative bilateral filtering algorithm to deal with visible images of different resolutions in the case of 1, 2, 4, 8 cores respectively in parallel and the run-time is shown in Table II. Table III shows the multi-core speedup ratio of processing efficiency.

Table II. Image multi-core parallel processing time

Time (ms)	1 core	2 cores	4 cores	8 cores
1280*1024	156.454	82.250	45.153	24.526
1024*768	110.878	58.458	30.266	15.490
800*600	47.741	25.864	14.476	8.295
640*480	31.048	17.275	9.611	5.672

It can be seen from Table II and Table III that the iterative bilateral filtering algorithm has higher computational efficiency on various resolutions. As the number of processing cores increases, the overall processing time is decreasing. The speedup ratio of parallel execution increases linearly by the increase of the number of processors relative to serial execution. When the number of DSP increases, the processing speed is exponentially increased. But the communication between the cores is more frequent, the inter-core interaction and memory access bring more costs and the parallel potential of the algorithm for tapping is also declining, so the processing speed of the growth is gradually slow.

Table III. Speedup ratio of images' multi-core processing

speedup ratio	1 core	2 cores	4 cores	8 cores
1280*1024	——	1.902	3.465	6.379
1024*768	——	1.897	3.663	7.158
800*600	——	1.846	3.298	5.755
640*480	——	1.797	3.230	5.474

As shown in Table II, for the commonly used images the image processing time can be of less than 25ms using 8-cores computing in parallel, which fully meet the requirements of real-time processing. Especially for high-definition images of 1280*1024 resolution, single-core, dual-cores and 4-cores algorithms' processing time are more than 40ms, which cannot achieve real-time processing for image. After 8-cores acceleration, the calculation time is about 24ms that meets the real-time requirements. It can be seen that the multi-core parallel framework greatly improves the processing performance of image processing systems. In addition, for the general standard images, single-core or dual-cores processing can achieve real time requirement and get a better result. As a result, the performance of the multi-core DSP image processing system is also related to the computational efficiency of the algorithm, the optimization of the algorithm and the optimization of the execution code.

Based on the above analysis, the multi-core DSP real-time image processing platform has achieved the performance requirements of the platform and solved the problem of performance bottleneck of the existing real-time image processing platform.

VII. CONCLUSION

In this paper, by analyzing the architecture limitations of the current real-time image processing platform, a real-time image processing platform based on multi-core DSP+FPGA is designed and implemented. The system uses the powerful computing capability of DSP to process the high-frame-rate and high-resolution images in real time. It meets the interactive bandwidth demand of the high-speed data between processor chips and the requirement of processing performance on the complex algorithms. This system possesses good reliability, stability and functions of expansion, which is easy to be improved and upgraded. On the basis of this platform, the follow-up work that transplanting and optimizing the image defogging algorithm, DDE algorithm and other image processing algorithms will be carried out to achieve real-time processing.

REFERENCES

[1] J.Yu. Hardware design of the multi-core DSP image processing system[D].Beijing: University of the Chinese Academy of Sciences, 2013:pp.1-7.

[2] Taxas Instruments Inc.KeyStone Architecture Multicore Navigator UserGuide[EB/OL].http://www.ti.com/lit/ug/sprugr9d/sprugr9d.pdf,2 2011-12.

[3] J.J.Zhang,D.P.Chen,X.L.Chai.Implementation of rapidIo interconnect system under VxWorks[J].ComputerEngineering,2011,Engineering, 2011,37(3):pp.236-240.

[4] Taxas InstrumentsCorp.UCD9222 Data Sheet[EB/OL].http://www. ti.com/product/ucd9222.pdf.p. 2012-02.

[5] H.Wu, J.Y.Xiao, H.Q.Fan. TMS320C6678 multi-core DSP communication on method between nuclear[J]. Eectronic Technology, 2012, 38(9):pp.2-3.

[6] X.J.Gong,L.M.Zou,Y.X.Hu. Research on Parallel Program Design Method of Multi-core System Based on OpenMP[J].Journal of NanhuaUniversity(Natural Science Edition), 2013, 27(1): pp.64-68.

[7] Taxas Instruments.OpenMP Programming for Keystone Multicore Processors [EB/OL].http://www.ti.com, 2012.

[8] Q. Yang, Recursive bilateral filtering, in Computer Vision–ECCV 2012. 2012, Springer. p. 399-413.