EMBEDDED IMAGE PROCESSING SYSTEM ON FPGA

Vo Ky Chau and Truong Quang Vinh

Deparment of Electrical and Electronic Engineering, University of Technology, Viet Nam

ABSTRACT

In this paper, the design of an embedded image processing system (called DIPS) on FPGA is presented. DIPS is based on the Xilinx MicroBlaze™ 32-bit soft processor core and implemented in Spartan-3. Some algorithms are implemented in DIPS, such as image filter (mask size is 3x3), edge detection, forward and inverse wavelets transform. Results from processing on Lena image will also be shown.

1. INTRODUCTION

Today, design of embedded systems can be microcontroller-based, DSP-based, ASIC-based or FPGA-based systems. Xilinx, a FPGA vendor, has provided the MicroBlaze™ 32-bit soft processor core which is licensed as a part of the Xilinx Embedded Development Kit (EDK). The EDK is a complete embedded development solution that includes a library of peripheral IP cores.

In this project, the MicroBlaze™ is used as the heart of the image processing system (called DIPS). Based on the IBM CoreConnect™, an IBM-developed on-chip bus-communications link that enables chip cores from multiple sources to be interconnected to create entire new chips, DIPS includes some available peripheral IP cores such as UART, and memory controller.

2. OVERVIEW OF THE XILINX MICROBLAZE AND STRUCTURE OF DIPS

The MicroBlaze™ soft processor is a 32-bit Harvard RISC architecture. The backbone of the architecture is a single-issue, 3-stage pipeline with 32 general-purpose registers, an Arithmetic Logic Unit (ALU), a shift unit, and two levels of interrupt. This basic design can then be configured with more advanced features such as: barrel shifter, floating-point unit (FPU), caches, exception handling, debug logic, and others. Figure 1 is the diagram of MicroBalze™ core processor.

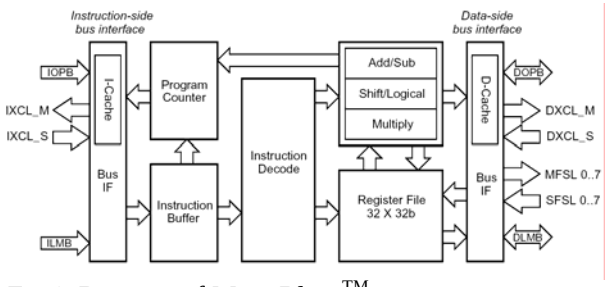


Fig 1. Diagram of MicroBlaze™ processor core

The general purpose processor interface conforms to the IBM CoreConnect™ On-chip Peripheral Bus (OPB) standard.

The MicroBlaze™ processor has two dedicated memory interfaces for improved bandwidth: the Local Memory Bus (LMB) and the Xilinx CacheLink (XCL) interface. The LMB provides low-latency storage such as interrupt and exception handler, while the XCL is a high performance point-to-point connection to an external memory controller.

The Fast Simplex Link (FSL) is a simple, yet powerful, point-to-point interface that connects user-developed co-processors to the MicroBlaze processor pipeline.

Figure 2 shows the diagram of DIPS. As be shown, based on the MicroBlaze™ processor, DIPS includes UART that allowing to send and receive commands and images from PC. DIPS also include four GPIO IP core to control digital input and output such as 7-seg leds, push buttons…

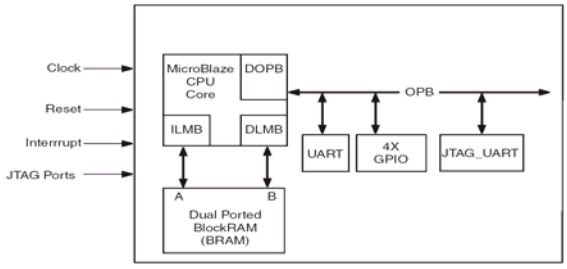


Fig 2. The structure of DIPS

Besides, a memory controller is used to control a 256Kx32 SRAM memory that is used to store source and processed images (not be shown in Figure 2).

3. ALGORITHMS ON DIPS

3.1 Image Filter

DIPS can filter an image using 3x3 masks. Two main filters are smoothing filter (low-pass filter), and sharpening filter (include high-pass, high-boost, Prewitt and Sobel filters). These masks are shown in Figure 3.

$$\frac{1}{9}\times\begin{bmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{bmatrix}$$

(a) Low-pass

$$\frac{1}{9}\times\begin{bmatrix}-1 & -1 & -1\\ -1 & 8 & -1\\ -1 & -1 & -1\end{bmatrix} \qquad \frac{1}{9}\times\begin{bmatrix}-1 & -1 & -1\\ -1 & w & -1\\ -1 & -1 & -1\end{bmatrix}$$

(b) High-pass (c) High-boost

$$\begin{bmatrix}-1 & -2 & -1\\ 0 & 0 & 0\\ 1 & 2 & 1\end{bmatrix} \qquad \begin{bmatrix}-1 & 0 & 1\\ -2 & 0 & 2\\ -1 & 0 & 1\end{bmatrix}$$

(d) Ox – Sobel (e) Oy - Sobel

$$\begin{bmatrix}-1 & -1 & -1\\ 0 & 0 & 0\\ 1 & 1 & 1\end{bmatrix} \qquad \begin{bmatrix}-1 & 0 & 1\\ -1 & 0 & 1\\ -1 & 0 & 1\end{bmatrix}$$

(f) Ox – Prewitt (g) Oy - Prewitt

Fig 3. Masks in image filter function

Which filter will be used depend on a control byte from PC.

3.2 Edge Detection

This function is performed in 3x3 windows of image by comparing the maximum of differences between horizontal, vertical, and diagonal pixels and a threshold value. If the maximum of differences is greater than the threshold value, the center pixel will be filled with this maximum value, else the center pixel will be zero.

3.3 Wavelets Transform

Wavelets transform is executed by using LeGall 5/3 filter. DIPS uses this filter because of recommendation of JPEG-2000 image compression standard. This is a reversible and lossless transform. Below are the equations for forward and inverse wavelets transform to be used in DIPS.

$$Y(2n+1) = X(2n+1) - \left\lfloor \frac{X(2n) + X(2n+2)}{2} \right\rfloor \quad (1)$$

$$Y(2n) = X(2n) + \left\lfloor \frac{Y(2n-1) + Y(2n+1) + 2}{4} \right\rfloor \quad (2)$$

$$X(2n) = Y(2n) - \left\lfloor \frac{Y(2n-1) + Y(2n+1) + 2}{4} \right\rfloor \quad (3)$$

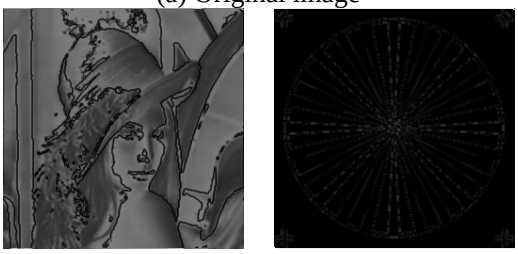
$$X(2n+1) = Y(2n+1) + \left\lfloor \frac{X(2n) + X(2n+2)}{2} \right\rfloor \quad (4)$$

To improve calculating time of this function, wavelets transform unit is designed as an IP core conforms to the IBM CoreConnect™ standard.

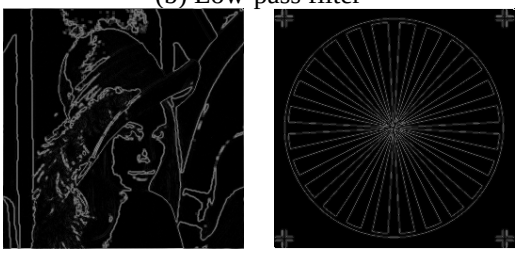
4. RESULTS

Figure 4 shows the results of image filter function in DIPS.

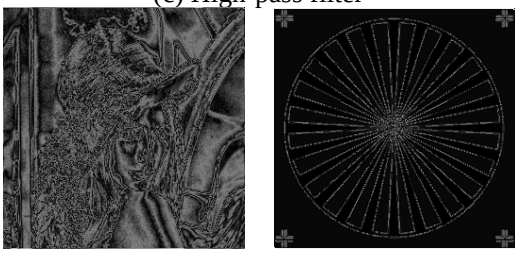
(a) Original image



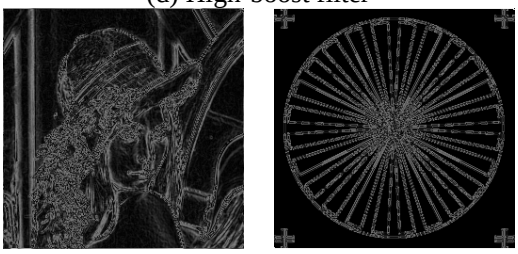
(b) Low-pass filter



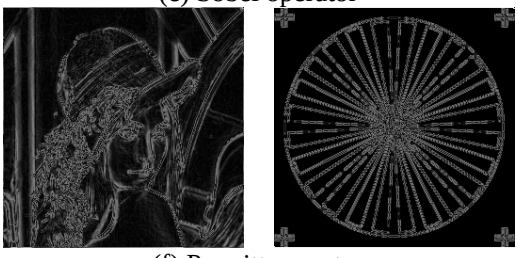
(c) High-pass filter



(d) High-boost filter



(e) Sobel operator

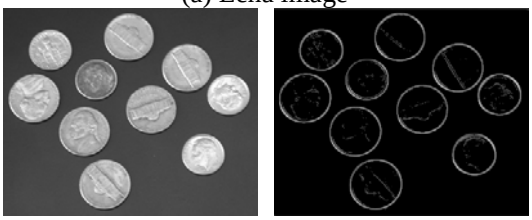


(f) Prewitt operator

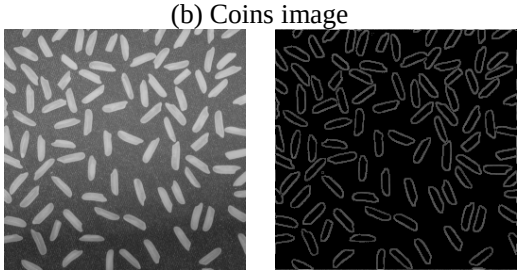
Fig 4. Image filter function

Figure 5 presents the result of edge detection function in DIPS.

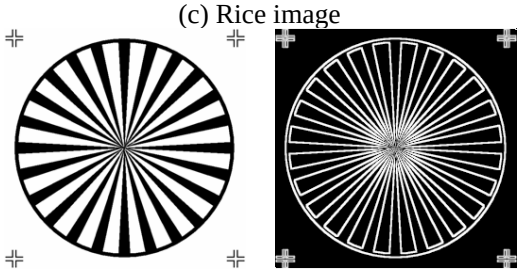
(a) Lena image



(b) Coins image



(c) Rice image

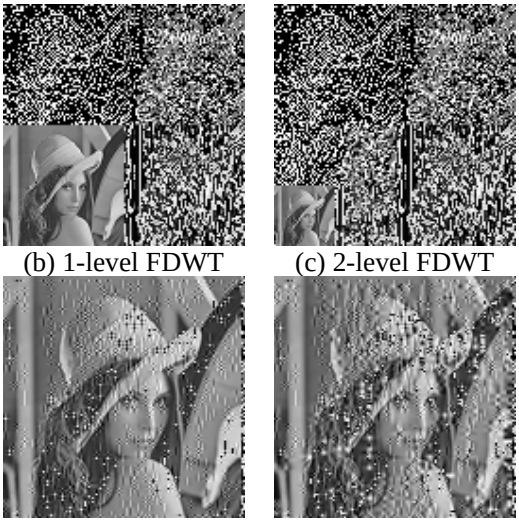


(d) Pat image

Fig 5. Edge detection function

Figure 6 shows the results of forward and inverse discrete wavelets transform in DIPS.

(a) Original image



(b) 1-level FDWT (c) 2-level FDWT

(d) 1-level IDWT (e) 2-level IDWT

Fig 6. Forward and Inverse Wavelets Transform

As shown, inverse wavelets transform has error because of truncating the number of bits in its structure.

5. CONCLUSIONS AND FUTURE WORK

This writing introduces a method to implement an image processing system in FPGA. The MicroBalze™ is used to make easily in designing embedded systems. This 32-bit microprocessor is strong enough to support image processing applications, and easy enough to learn quickly.

Besides the above functions, DIPS also can upgrade to compress an image (in designing phase). If image input and output devices are supplied, DIPS can be used as a platform to developing many applications in image processing field.

REFERENCES

1. Bernd Jahne, Digital Image Processing, Springer (2002).
2. Michael D. Adams, The JPEG-2000 Still Image Compression Standard, Dept. of Electrical and Computer Engineering, University of Victoria (25-12-2002).
3. Michael D. Adams, Jasper Software Version 1.700.0, Dept. of Electrical and Computer Engineering, University of Victoria (09-02-2003).
4. Digilent Inc., S3Board reference manual.
5. Xilinx, MicroBlaze Processor Reference Guide, (9-2004).