

Embedded Real-Time Video Processing System on FPGA

Yahia Said¹, Taoufik Saidani¹, Fethi Smach², Mohamed Atri¹, and Hichem Snoussi³

¹ Laboratory of Electronics and Microelectronics (EμE),
Faculty of Sciences Monastir, 5000 Tunisia

said.yahia1@gmail.com, saidani_toufik@yahoo.fr,
mohamed.atri@fsm.rnu.tn

² Active Networks

1 rue de Terre Neuve, BP 127 - 91944 Courtaboeuf Cedex - France
smach_fethi@yahoo.fr

³ Université de technologie de Troyes

Institut Charles Delaunay ICD, UMR STMR 6279
BP 2060 - 10010 TROYES Cedex
hichem.snoussi@utt.fr

Abstract. Image Processing algorithms implemented in hardware have emerged as the most viable solution for improving the performance of image processing systems. The introduction of reconfigurable devices and high level hardware programming languages has further accelerated the design of image processing in FPGA.

This paper briefly presents the design of Sobel edge detector system on FPGA. The design is developed in System Generator and integrated as a dedicated hardware peripheral to the Microblaze 32 bit soft RISC processor with the EDK embedded system. The input comes from a live video acquired from a CMOS camera and the detected edges are displayed on a DVI display screen.

Keywords: Sobel Edge detector, Real Time, Microblaze Processor, Field Programmable Gate Arrays (FPGA), Embedded Development Kit (EDK), System Generator (SysGen).

1 Introduction

Computationally Intensive DSP applications such as Image Processing is getting widely used in embedded systems for many applications, such as object detection, space exploration, security or video surveillance.

Reconfigurable hardware in the form of Field Programmable Gate Arrays (FPGAs) has been proposed as a way of obtaining high performance for Image Processing, even under real time requirements [1]. Implementing image processing algorithms on reconfigurable hardware minimizes the time-to-market cost, enables rapid prototyping of complex algorithms and simplifies debugging and verification. Therefore, FPGAs are an ideal choice for implementation of real time image processing algorithms [2].

Edge detection is a fundamental tool used in most image processing applications to obtain information from the frames as a precursor step to feature extraction and object segmentation. This process detects outlines of an object and boundaries between

objects and the background in the image. An edge-detection filter can also be used to improve the appearance of blurred or anti-aliased video streams [3].

FPGAs offer many performance benefits for executing image processing applications. The FPGA design can also reduce the system costs with various verification techniques such as behavioral simulation and post-route simulation. Moreover, Xilinx Embedded Development Kit (EDK) tools make it possible to implement a complete video processing system on a single FPGA using hardware/software codesign methods.

The objective of this work is to develop a real-time edge detection system with an input from a CMOS camera and output to a DVI display and verified the results video in real time.

This paper is organized as follows: Section 2 describes system architecture and functions of each block. Section 3 covers the architecture for edge detector core developed in System Generator. In section 4, experimental results of the proposed system are shown. Finally, a brief conclusion and directions for future work are given in Section 5.

2 System Architecture

The setup for implementation consists of the Spartan-3A DSP FPGA Video Starter Kit (VSK), a development platform consisting of the Spartan-3A DSP 3400A FPGA, the FMC-Video daughter card, and a VGA camera (“Fig. 1”).

The Spartan-3A DSP 3400A Development Platform is built around a Spartan-3A DSP XC3SD3400A device that provides significant resources (for example, 126 embedded DSP blocks) for implementing high performance video processing systems and co-processors.

The VSK includes a VGA camera based upon the Micron MT9V022 image sensor of resolution 720 x 480 pixels delivering serial frames at 60 fps through a FPGA Mezzanine Card (FMC) Daughter card which is an add-on card that augments the video capabilities of the Spartan-3A DSP 3400A Development Platform [4].

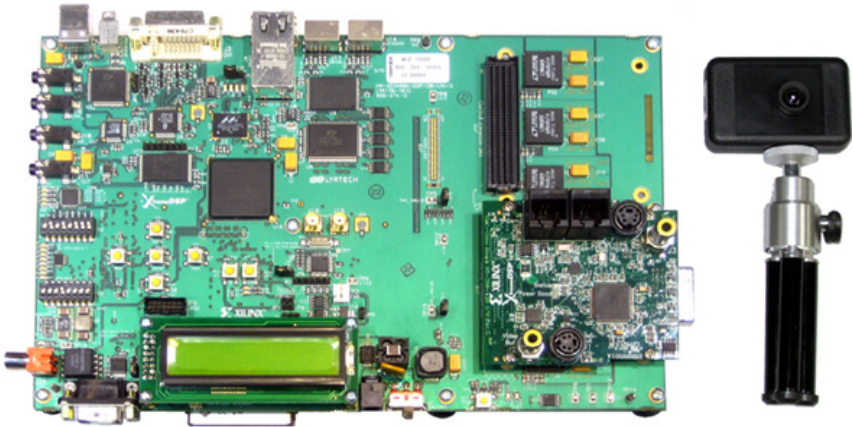


Fig. 1. Spartan-3A DSP 3400 Development Platform, FMC-Video, and Camera

Figure 2 shows a detailed system diagram of the implemented video filtering design. The complete streaming video application includes Video interfaces, a run-time configurable processing blocks, a real-time edge detection filter, and a MicroBlaze embedded processor for embedded control of the video subsystem.

The video processing application is designed as a system on a programmable chip with the help of Embedded Design Kit. The serial video is de-serialized on the FMC-Video card. The resulting parallel data stream is the input to the Camera In block. The Camera PCORE registers the signals, and groups the video signals into a unified bus that is connected to the Camera Processing block, which is included in the camera frame buffer reference designs shipped with the VSK [4], to control brightness, contrast and other parameters. The edge filter is applied on the input signal arriving from the Camera Processing block. The output signal is Gamma corrected for the output DVI monitor and is driven by Display controller to the DVI output monitor. The Video to VFBC core manages the storing of video into frame buffers in external memory. It writes the video data to the VFBC interface on the MPMC memory controller.

The Display Controller core reads video frames out of memory from a VFBC interface of MPMC and displays them to the output screen by applying the correct timing signals. The edge filter core developed in System Generator for DSP will be detailed in the next section.

The video pipeline demonstrated by our design is created using the Xilinx Embedded Development Kit (EDK) [5] and System Generator for DSP [6]. The Embedded Development Kit is a collection of Intellectual Property (IP) cores and tools for building FPGA-based embedded systems. System Generator for DSP enables the use of the Simulink/MATLAB modeling environment for FPGA design by providing a Simulink blockset of over 100 Xilinx optimized DSP building blocks.

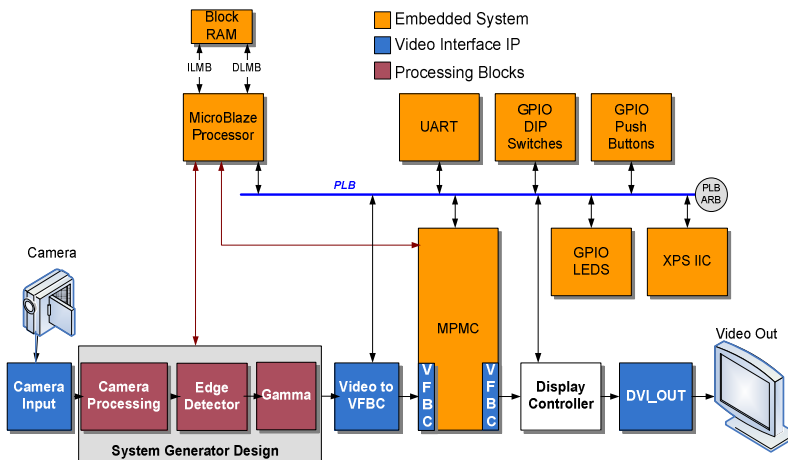


Fig. 2. Video Pipeline with MicroBlaze processor and peripheral

The architecture consists of a set of modules interconnected with buses, as seen in “Fig. 2”. The Camera Processing, Gamma and Edge Detection cores are connected to

the Embedded MicroBlaze processor through Processor Local Bus (PLB). The Processor is connected to dual-port SRAM, called Block RAM (BRAM), using a dedicated Local Memory Bus (LMB). This bus features separate 32-bit wide channels for program instructions and program data, using the dual-port feature of the BRAM. The LMB provides single-cycle access to on-chip dual-port Block RAM.

The MicroBlaze soft processor core [7] provided by Xilinx is central in the system and used as an embedded video controller. It is a reduced instruction set computer (RISC) optimized for implementation in the Xilinx Field Programmable Gate Arrays (FPGAs). Figure 3 shows the block diagram of MicroBlaze.

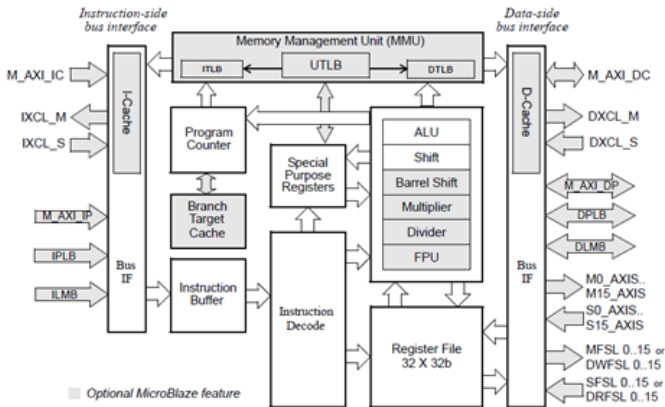


Fig. 3. MicroBlaze Core Block Diagram

3 Sobel Edge Detector

Edge detection is the process of localizing pixel intensity transitions. The edge detection has been used by segmentation, motion analysis, object recognition, target tracking, and many more [8]. Therefore, the edge detection is one of the significant techniques in the field of image processing.

The most well known technique for edge detection is gradient-based. The gradient method looks the edges by finding maximum and minimum in the first derivative of the image. Sobel is gradient based edge detection algorithm which performs a 2-D spatial gradient measurement on the video data. It uses a pair of 3X3 convolution masks, one estimating gradient in x-direction and other in y-direction. Then the value of the gradient magnitude is computed from the above two gradients.

First, RGB data are converted into grayscale to obtain image intensity, using the following equation:

$$I = (0.2989 \times R + 0.5870 \times V + 0.1140 \times B) \tag{1}$$

Then horizontal and vertical gradient are calculated as shown in “(2)”.

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * I \quad \text{and} \quad G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * I \quad (2)$$

The magnitude and orientation are obtained as follow:

$$G = |G_x| + |G_y| \quad \text{and} \quad \theta = \text{Arctan} \left(\frac{G_y}{G_x} \right) \quad (3)$$

We build the sobel edge detector as a video processing accelerator, using System Generator for DSP and Simulink. The design of our filter is shown in “Fig. 4”.

System Generator supports hardware in-the-loop co-simulation using the Spartan-3A DSP 3400A development platform, which can accelerate the performance of Simulink simulations up to 100x. This acceleration enables video algorithm development and debug using real-time video streams read into Simulink using The Mathworks’ Data Acquisition Toolbox [9].

System Generator for DSP can automatically generate accelerator blocks in the form of a custom peripheral for the embedded video application that allows the MicroBlaze processor to read and write shared memories in the accelerator block. This includes an automatically generated hardware interface for the PLB bus, a software C driver file, and software documentation for using the DSP co-processor.

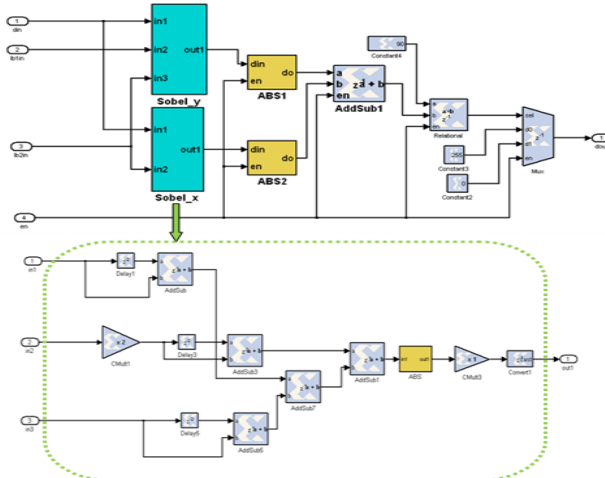


Fig. 4. Sobel architecture with system generator

The System Generator design contains an EDK Processor block that can be exported as an EDK pcore using the EDK Export Tool compilation target. The export process creates a PLB-based pcore, which is integrated to the Microblaze 32 bit soft RISC processor with the Xilinx Platform Studio (XPS) [6].

4 Experimental Results

In the system setup a DVI display shows the output edge from the camera. Experimental setup for implementation of sobel edge detection is presented in “Fig.5”.

The total resource usage for the system, including the MicroBlaze, bus structure, the sobel edge core and peripherals, is 9094 slices, equaling 38% of the Spartan 3A DSP 3400. The system was implemented to run at 62.5MHz. It is possible that higher frequencies are attainable, up to a limit of around 125MHz. The MicroBlaze has a maximum frequency of 125MHz on the Spartan 3A DSP 3400, and the sobel core has a post-synthesis maximum estimate of 68.432MHz.

Table 1 shows the amount of logic used for the sobel edge module. A maximum of 5% of the FPGA’s total resources are used by this module. The post-synthesis resource usage of the MicroBlaze processor is 1531 slices. The synthesis results of the overall system are given in Table 2.

From the synthesis results of our system, we can see that few resources of the FPGA are used; hence space is available for other complex image and video processing applications.

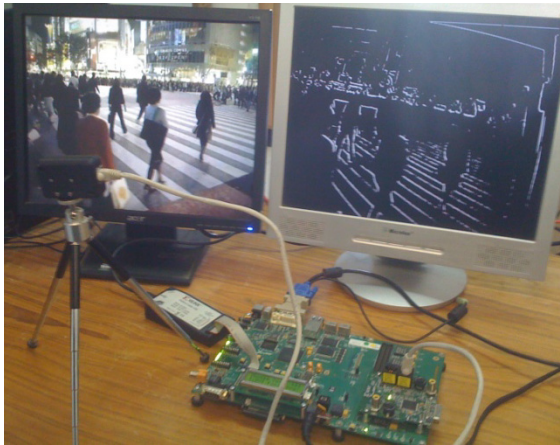


Fig. 5. Experimental setup for implementation of edge detection. Input is from CMOS camera and the output is on a DVI display.

Table 1. Post-Synthesis device utilization for the Sobel edge module implemented on the Spartan 3A DSP 3400

Resource Type	Used	Available	%
Slices	1284	23872	5%
Slice Flip Flops	1745	47744	3%
4 input LUTs	1713	47744	3%
bonded IOBs	0	469	0%
BRAMs	5	126	3%
DSP48s	4	126	3%
Maximum Frequency		68.432 MHz	

Table 2. The synthesis results of the overall system

Resource Type	Used	Available	%
Slices	9094	23872	38%
Slice Flip Flops	11451	47744	24%
4 input LUTs	12883	47744	27%
bonded IOBs	78	469	17%
BRAMs	69	126	55%
DSP48s	7	126	6%
Maximum Frequency	88.547MHz		

5 Conclusion

Continual growth in the size and functionality of FPGAs over recent years has resulted in an increasing interest in their use as implementation platforms for image processing applications, particularly real-time video processing [10].

In this paper, we propose a design for real-time video processing system on a Spartan 3A DSP FPGA. Sobel edge detector was implemented at a rate of 60 fps for an input image of resolution 720x480.

The implemented system architecture has 88.547MHz maximum frequency and uses 9094 CLB slices with 38% utilization, so there is possibility of implementing some more parallel processes with this architecture on the same FPGA.

System Generator for DSP enables the use of Simulink for Xilinx FPGA designs by providing a rich set of DSP building blocks, optimized for Xilinx devices. DSP designs captured in System Generator can be converted into custom peripherals for Platform Studio and connected to the embedded system using the processor local bus.

Future works include the use of the Xilinx System Generator and EDK development tools for the implementation of other blocks used in computer vision like feature extraction and object detection on Xilinx Programmable Gate Arrays (FPGA).

References

1. Crookes, D.: Design and implementation of a high level programming environment for FPGA-based image processing. *IEE Proceedings on Vision, Image, and Signal Processing* 147(4), 377 (2000)
2. Rao, D.V., Patil, S., Babu, N.A., Muthukuma, V.: Implementation and Evaluation of Image Processing Algorithms on Reconfigurable Architecture using C-based Hardware Descriptive Languages. *International Journal of Theoretical and Applied Computer Sciences* 1(1), 9–34 (2006)
3. Neoh, H., Hazanchuk, A.: Adaptive Edge Detection for Real-Time Video Processing using FPGAs. *Global Signal Processing* (2004)
4. Spartan-3A DSP FPGA Video Starter Kit user Guide, <http://www.xilinx.com>
5. Xilinx Inc. Embedded System Tools Reference Manual, <http://www.xilinx.com>

6. Xilinx System Generator user Guide, <http://www.xilinx.com>
7. MicroBlaze soft processor, <http://www.xilinx.com>
8. Senal, H.G.: Gradient Estimation Using Wide Support Operators. *IEEE Transaction on Image Processing* 18(4) (April 2009)
9. Mallet, J.: Updated Starter Kit Speeds Video Development. *Xell Journal* (67), 18–21 (2009)
10. Hutchings, B., Villasenor, J.: The Flexibility of Configurable Computing. *IEEE Signal Processing Magazine* 15, 67–84 (1998)