Hardware Software co-simulation for Image Processing Applications

A.C.Suthar[1], Mohammed Vayada[2], C.B.Patel[3], G.R.Kulkarni[4]
[1]Research Scholar, Dept. of E.C.E., KSV University, Gandhinagar, Gujarat, India
[1,3]Department of E.C.E., L. C. Institute of Technology, GTU, Mehsana, Gujarat, India
[2]Department of E.C., PIET, Limda, Baroda, Gujarat, India
Principal, CCET, Wadhwan, Gujarat, India

Abstract – ***We proposed the concept of hardware software co-simulation for image processing using Xilinx system generator. Recent advances in synthesis tools for SIMULINK suggest a feasible high-level approach to algorithm implementation for embedded DSP systems. An efficient FPGA based hardware design for enhancement of color and grey scale images in image and video processing. The top model – based visual development process of SIMULINK facilitates host side simulation and validation, as well as synthesis of target specific code, furthermore, legacy code written in MATLAB or ANCI C can be reuse in custom blocks. However, the code generated for DSP platforms is often not very efficient. We are implemented the Image processing applications on FPGA it can be easily design.***

Keywords - **digital image processing; xilinx system generator; matlab**

I. INTRODUCTION

The handling of digital images has become in recent decades a subject of widespread interest in different areas such as medical and technological application, among others. Image processing is used to modify pictures to improve them (enhancement, restoration), extract information (analysis, recognition), and change their structure (composition, image editing) [1]. Images can be process by optical, photographic, and electronic means, but image processing using digital computers is the most common method because digital methods are fast, flexible, and precise.

We may cite lot of examples where image processing helps to analyze infer and make decision. The main objective of image processing is to improve the quality of the images for human interpretation or the perception of the machines independent of the images for human interpretation or the perception of the machines independently. This paper focuses in the processing pixel to pixel of an image and in the modification of pixel neighborhoods and of course the transformation can be applied to the whole image or only a partial region. The need to process the image in real time, leading to the implementation level hardware, which offers parallelism, thus significantly reduces the processing time, which was why decided to use Xilinx System Generator, a tool with graphical interface under the Matlab Simulink, based blocks which makes it very easy to handle with respect to other software for hardware description. In addition to offering all the tools for easy graphical simulation level. This article presents architecture of image processing application generator, which is an extension of Simulink and consists of a bookstore called "Blocks Xilinx", which are mapped architectures, entities, signs, ports and attributes, which script file to produce synthesis in FPGAs, HDL simulation and development tools. The tool retains the hierarchy of Simulink when it is converted into VHDL.

II. XILINX SYSTEM GENERATOR BASED DESIGN

It is requirement of an efficient rapid prototyping system to develop an environment targeting the hardware design platform. Although the Xilinx ISE 12.1 foundation software is not directly utilized, it is required due to the fact that it is running in the background when the System Generator blocks are implemented [2,3]. The System Generator environment allows for the Xilinx line of FPGAs to be interface directly with Simulink. In addition there are several cost effective development boards available on the market that can be utilized for the software design development phase.

Xilinx System Generator (XSG) is an integrator design environment (IDE) for FPGAs, which uses Simulink, as a development environment, it is presenting in the form of block set. It has an integrated design flow, to move directly to the configuration file (*.bit) necessary for programming the FPGA. One of the most important features of Xilinx System Generator is possessed abstraction arithmetic, which is working with representation in fixed point with a precision arbitrary, including quantization and overflow. You can also perform simulation both as a fixed-point double precision. XSG automatically generates VHDL code and a draft of the ISE model being develop. Make hierarchical VHDL Synthesis, expansion and mapping hardware, in addition to generating a user constraint file (UCF), simulation and test bench and test vectors among other things. Xilinx System Generator has created primarily to deal with

complex Digital signal processing (DSP) applications, but it has other application like the theme of this work [6]. The blocks in Xilinx System Generator operate with Boolean values or arbitrary values in fixed point, for a better approach to hardware implementation. In contrast Simulink works with numbers of double-precision floating point. The connection between blocks, Xilinx system generator and Simulink Blocks are gateway blocks. Figure.1 shows the broad flow design Xilinx System Generator. As already mentioned, you can then move to the configuration file to program the FPGA [5].

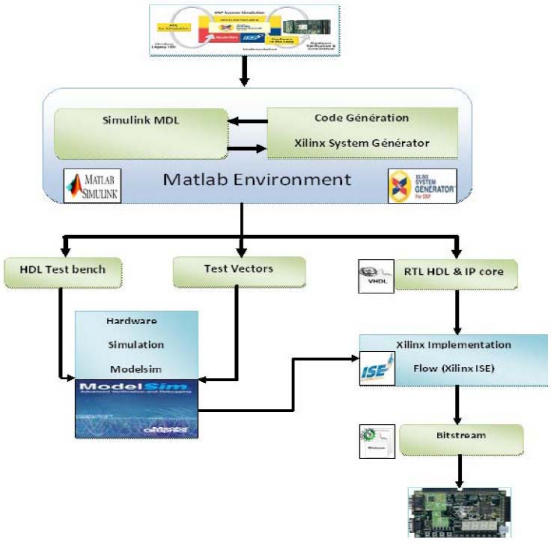


Figure .1 Design flow in Xilinx System Generator using matlab

III BASIC MODEL BASED DESIGN

The very fundamental part of our work is how to read and write image through Xilinx system generator with matlab platform. In this section, we shows figure 2 and 3 the reading of image using XSG. We represent how image is read from the directory and process in simulink. All the image processing is doing between FPGA boundaries gateway in and gateway out. Image read is in Simulink is basic block for reading the image from the directory and gateway out the image output as per the block specified under Xilinx FPGA.

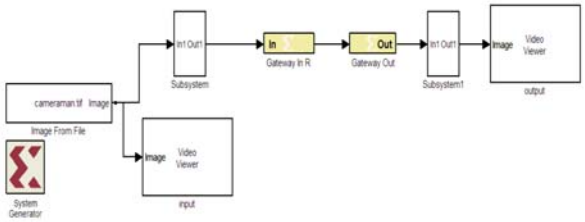
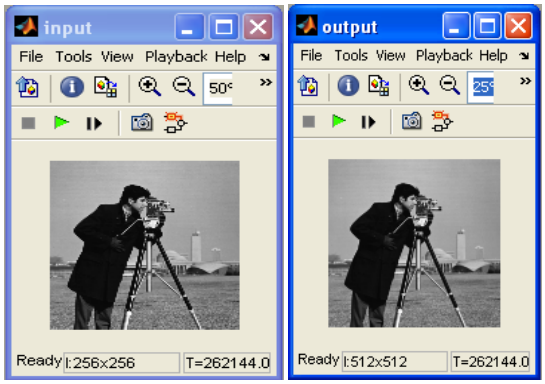


Figure .2 Image Read operations using in XSG



(a) Input Image (b) Output Image

Figure .3 (a) Input and (b) output of basic image read model

IV. OPERATIONS OF IMAGE PROCESSING

A. Negative of Image

Figure 4 and 5 shows gray scale and colour image respectively the operation of negative of image.

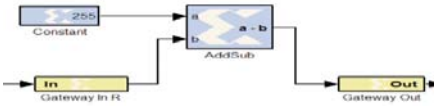
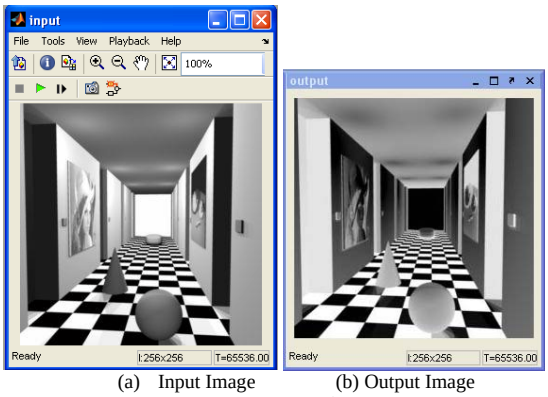


Figure .4 Gray Image Negative operations



(a) Input Image (b) Output Image

Figure .5 (a) Input and (b) Output of Gray Scale Negative

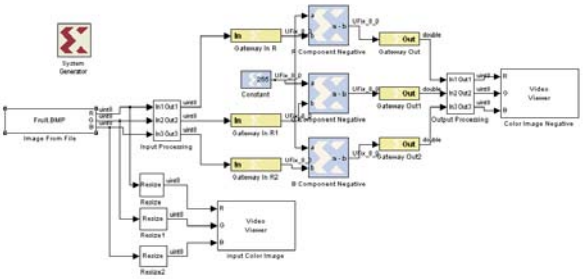


Figure .6 Color Image Negative operations

(a) Input Image (b) Output Image

Figure .7 (a) Input and (b) Output of Gray Scale Negative

boundaries. In this section we represents how gray scale image is inverted using simple block insertion which reflect image negative at output side, note that the block required to calculate negative is connected within the FPGA Gateway 888IN/OUT boundaries. Section C shows the color image can be extract in to its R-G-B components. Section D represents the required block to calculate color image negative, each color component is treat as individual signal and individual image negative blocks required to make all component negative and output of each is given to the R-G-B video viewer. The result shows in figure 6 and 7 for two input test images.

B. Enhancement of Image

In this in we shows that how image can be enhanced by adding a constant to each pixel values.[3,4] Image filtering can also be done using model based design different filtering architecture can be defined and Xilinx block can be created. Figure 8 and 9 show the simulation model with its result.

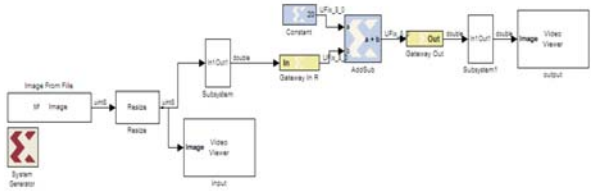
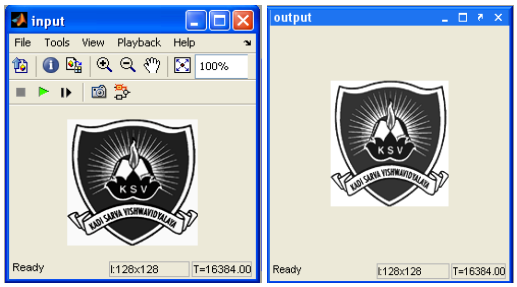


Figure .8 Image Enhancement Simulation Block



(a) Input Image (b) Output Image

Figure .9 (a) Input and (b) Output of Gray Scale Negative

V CONCLUSION

Xilinx system generator is a very useful tool for developing computer vision algorithms. It could be described as a timely, advantageous option for developing in a much more comfortable way than that permitted by VHDL or Verilog hardware description languages (HDLs).The Xilinx System Generator tool is a new application in image processing, and offers a friendly environment design for the processing, because processing units are designed by blocks. This tool support software simulation, but the most important is that can synthesize in FPGAs hardware, with the parallelism, robust and speed, this features are essentials in image processing.

In this paper we have presented the basic idea how image processing can be done in model based approach, we have demonstrated some of the image processing application which is done under SIMULINK and this can be implement using Xilinx System Generator (XSG). In this paper, we have shown how image read and enhance of image like, gray scale or color images, R-G-B component extraction from color image and color image negative very efficiently and we have taken two test images for the color image negative to give better idea.

VI REFERENCES

[1] R. Gonzalaz, R. Woods, “Digital Image Processing”. New Jersey: Prentice-Hall 2002.
[2] DSP System Generator User guide release 12.1.
[3] Xilinx System Generator User's Guide, www.Xilinx.com.
[4] Matlab website,http:// www.mathworks.com.
[5] White paper: Using System Generator for Systematic HDL Design, Verification, and Validation WP283 (v1.0) January 17, 2008
[6] J.C.Moctezuma, S.Sanchez, R.Alvarez, A. Sánchez “Architecture for filtering images using Xilinx system generator” World Scientific Advanced Series In Electrical and Computer Engineering, Proceedings of the 2nd WSEAS International Conference on Computer Engineering and Applications. Pages 284-289.2008.