Mathematical Operation of Gray Scale Image Pixels using Xilinx System Generator

Mohammed Vayada [1], Anil suthar [2], Arvind Yadav[3]
[1, 3] Department of Electronics and Communication Engineering, PIET Limda, Baroda
[2]Department of Electronics and Communication, LCIT, Bhandu
Email: - [1] vayada_mohammed@yahoo.com, [2]acsuthar@yahoo.com, [3] aryme@rediffmail.com

***Abstract*- This paper discusses about the new concept of mathematical operation using Xilinx system generator which is embedded in MATLAB®-SIMULINK. Recent advances in synthesis tools for SIMULINK suggest a feasible high-level approach to algorithm implementation for embedded DSP systems. 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 ANSI C can be reused in custom blocks. However, the code generated for DSP platforms is often not very efficient. It is discussed that mathematical operation of Image pixel can be easily made and they can be implemented on VLSI targeted board using system generator and Xilinx Tool.**

Keywords- Digital Image Processing, XSG, Mathematical operation, SIMULINK,

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. We may cite lots 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 neighbourhoods 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 and code can be finally implemented on XSG(Xilinx System Generator)

II. XILINX SYSTEM GENERATOR

Xilinx System Generator (XSG) is an integrator design environment (IDE) for FPGAs, which uses Simulink, as a development environment and is presented 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 developed. 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 was created primarily to deal with complex Digital signal processing (DSP) applications, but it has other application like the theme of this work. 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

In Fig. 1 shows the broad flow design Xilinx System Generator. As already mentioned, you can then move to the configuration file to program the FPGA.

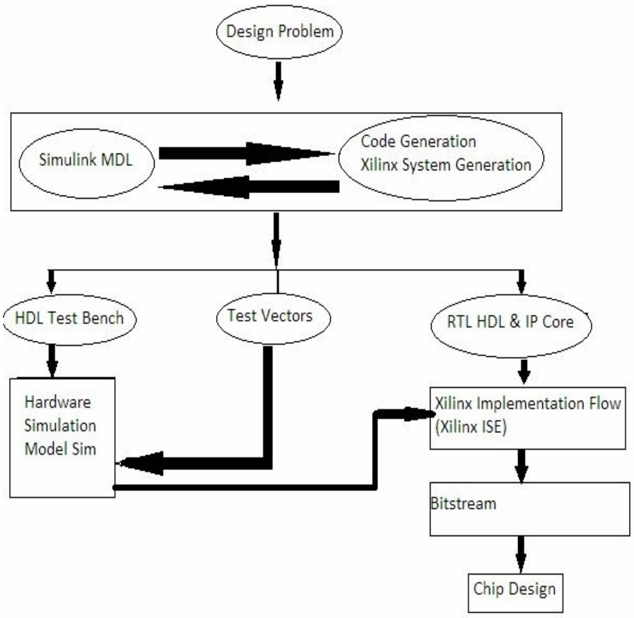


Fig. 1 Design flow in Xilinx System Generator

III. BASIC MODEL BASED DESIGN

Edge detection is the process of identifying sharp changes in image brightness [1]. This is important because it detects physical changes in the objects imaged. The image formation process in many applications such as seismology, photography, radar, microscopy and ultrasound is such that changes in acquired image brightness can relate to severe changes in depth or distance, discontinuities in surface orientation, changes in material properties, variation on scene illumination and boundaries of an object. These identified edges can lead to better image understanding while reducing data volume for more efficient subsequent processing [2].

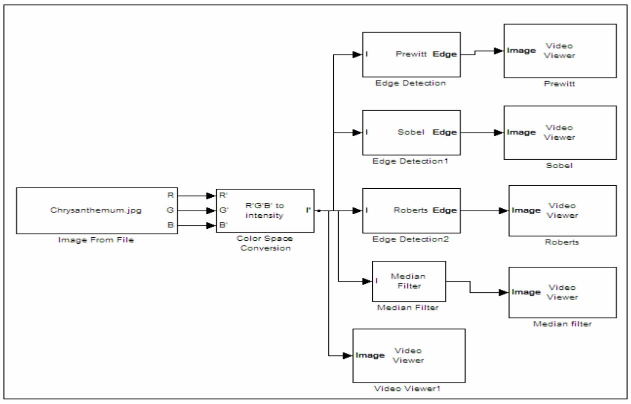
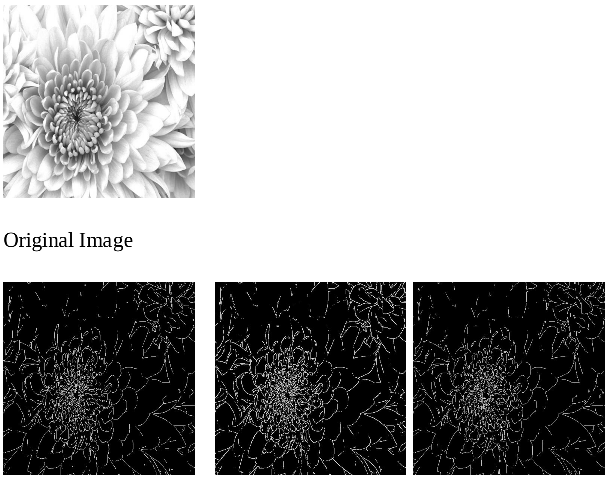


Fig. 2 Model Based Edge Detection of Image



Original Image

Prewitt Roberts Sobel

Fig. 2a. RESULTS

IV. READING IMAGE AND MATHEMATICAL OPERATION OF IMAGE PIXEL IN XILINX SYSTEM GENERATOR

A. READING IMAGE

Reading the image from the memory is done through image input block. Care should be taken that the size of the image must match with the buffer size, and then the image is converted in to 1D from 2D, because all these procedure in

Xilinx System Generator is done between Xilinx Gateway In and Gateway Out.

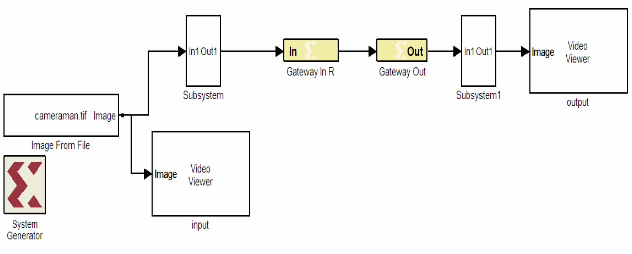
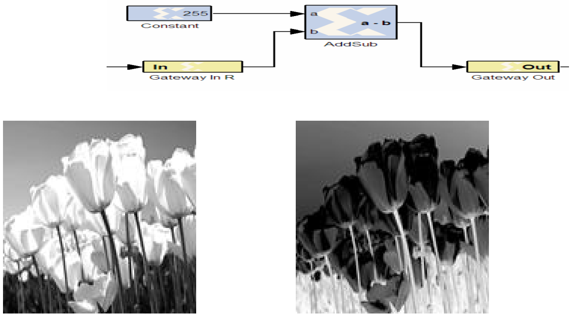


Fig. 3 Image Reading in XSG

The basic model of how to read the image in Simulink, which is further capable of creating VHDL using XSG, is shown in Fig. 3. Image has to be processed through image resize block, 2D to 1D convertor, frame converter and unbuffer block. The output of the unbuffer will be given to the Gateway In and Gateway Out block, which are the Xilinx Boundaries and XSG block is provided. By configuring XSG block we can have VHDL code. The attributes must be matched for the subsequent block.

B. GRAY SCALE IMAGE NEGATIVE



Input image **Output Image Negative**

Fig. 4 Gray scale Image Negative

Fig. 4 shows the model of Gray Scale Image Negative and image results.

C. IMAGE ADDITION

Two images are separately taken in to the Gateway In and Gateway Out block and all the mathematical functions are performed within the Xilinx Boundaries.

Fig. 5 shows the model of Image Addition.

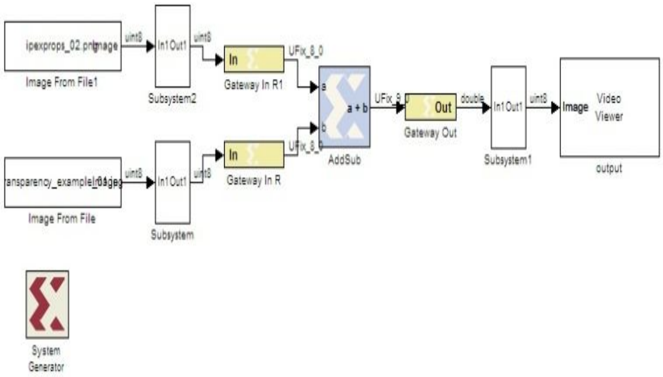
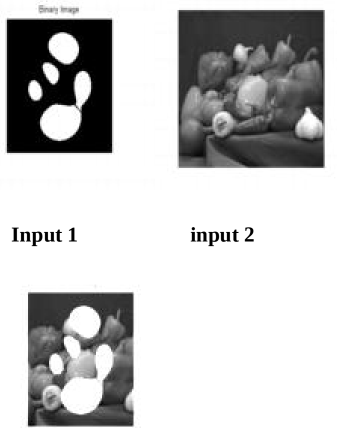


Fig. 5 Image Addition

Fig. 6 shows the Result of Image Addition



Input 1 **input 2**

Fig. 6 Output Image

D. GRAY SCALE IMAGE SUBTRACTION

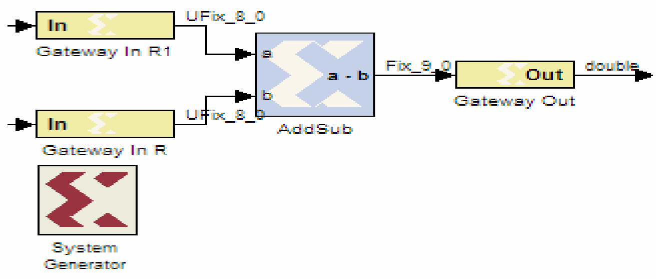
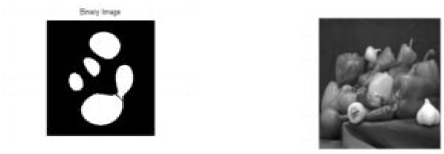
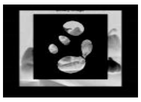


Fig. 7 Image Subtraction

Fig. 7 shows gray scale image subtraction model. This model is same as image addition except that, this time subtraction of two images is calculated and output of this model is shown in Fig. 8.



Input Images



Output Image

Fig. 8 Result of Image Subtraction

E. GRAY SCALE IMAGE MULTIPLICATION

Fig. 9 shows the model of Image Multiplication.

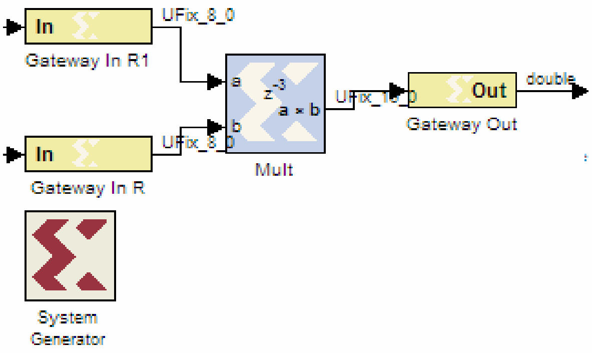
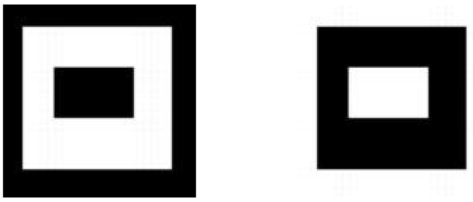
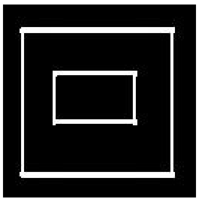


Fig. 9 Image Multiplication

Fig. 10 shows the Result of Image Multiplication.



Input Image

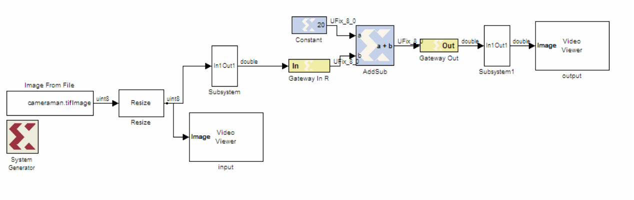


Output Image

Fig. 10 Result of Image Multiplication

F. IMAGE ENHANCEMENT

Fig. 11 shows Image Enhancement Model and Result using XSG.



Input Image

Output Image

Fig. 11 Model and Result for Image Enhancement

V. FUTURE WORK

Our future work is oriented for multimedia authentication system in which we are going to apply the combination of DCT, DWT, and SVD Transformation using XSG. We will test all these possibilities and compare the results with individual Transformation. Than the (.mdl) file will be converted to (.hdl) using XSG and this authentication system will be implemented on the VLSI target board [5].

VI. CONCLUSION

The Xilinx System Generator tool is a new application in image processing, and offers a friendly Design environment for the processing, because processing units are designed by using blocks. This tool supports software simulation, but the most important part is that it can be synthesized on FPGAs hardware, with the parallelism, robustness and speed, these features are essentials in image processing. In this paper we have presented the basic idea about how mathematical image processing can be done in model based approach. We have demonstrated some of the image processing application implemented using SIMULINK /XSG.

VII. REFERENCES

[1] R. Gonzalaz, R. Woods, "Digital Image Processing". New Jersey: Prentice-Hall 2002.

[2] DSP System Generator User guide release 10.1 March 2008

[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