A Technical Report on

An FPGA implementation of Vision Algorithms using Xilinx System Generator

EC399: MINI PROJECT

DEPT. OF ELECTRONICS AND COMMUNICATION ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY – ANDHRA PRADESH, TADEPALLIGUDEM – 534102, INDIA

Under the guidance of

Dr. Puli Kishore Kumar

SUBMITTED BY

EC3M26

B.Tech 3rd Year

Academic year: 2022-2023

May 3rd, 2023

DECLARATION

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

K KOUSHIK 620144 DATE:03/05/2023

CERTIFICATE

It is certified that the work contained in the thesis titled “**An FPGA implementation of Vision Algorithms using Xilinx System Generator**,” by “K KOUSHIK” bearing Roll No: “620144”, has been carried out under my/our supervision and that this work has not been submitted elsewhere for a minor project.

Signature of Supervisor
Dr. Puli Kishore Kumar
Assistant Professor
Dept. of Electronics and Communication Engineering
N.I.T. Andhra Pradesh
May 3rd, 2023

ACKNOWLEDGMENT

I would like to share our sincere gratitude to our supervisor “Dr. Puli Kishore Kumar” for helping us in completion of this project. During the work we faced many challenges as the field is new and due to our lack of knowledge and experience, but he helped us to get over from all the difficulties and in final completion of our project. We have learnt many new things during the project discussion. We are really glad to be associated with a person like Dr. Puli Kishore Kumar. All of our team is thankful to all department faculty members for their support.

We profoundly thank “Dr.Kiran Kumar Gurrala”, Head of ECE department who has been an excellent guide and also a great source of inspiration to our work. We would like to acknowledge the help for giving us all the required permissions and helping us to complete the project. We would like to thank all the faculty of the ECE department for teaching us and for spending time for the review. We also thank our friends who directly or indirectly helped us in our project work and completion of the report in time. Lastly, we would like to thank our families for their selfless support.

K KOUSHIK

ABSTRACT

This project provides the method of image processing using Xilinx System Generator. Xilinx System Generator has necessary libraries to assist various types of algorithms. It is integrated with MATLAB Simulink environment in this work. Model based design approach is used to implement various kinds of image processing algorithms. Hardware co-simulation is done to verify the results. The different image processing algorithms for RGB to gray scale, algorithm for image negatives, image enhancement, background subtraction, thresholding, erosion, dilation and masking are implemented using available System Generator blocks.

CONTENTS

List of figures

1 Introduction

Image processing has wide applications from medical image processing to computer vision, digital photography, satellite imaging, digital encryption and decryption. The quality of image is considerably increased by image processing algorithms, which helps lot in medical imaging, surveillance and robotics application for target identification and tracking . The need to process the image in real time is time consuming and leads to the only method of implementing the algorithm at hardware level. With FPGA implementations, the logic required by an application is implemented by building separate hardware for each function. Also FPGAs are inherently parallel; this gives the speed to those real time applications while retaining the programmable flexibility of software at a relatively low cost. This paper aims to implement image processing algorithms using Xilinx System Generator. The hardware implementation of the algorithms on FPGAs is done using model based design approach.

1.1 Xilinx System Generator

The Xilinx's Generator is a System-level modeling tool from Xilinx that facilitates FPGA hardware design. It extends Simulink in many ways to provide a modeling environment well suited for hardware design. The software automatically converts the high level System DSP block diagram to RTL. The result can be synthesized to Xilinx's FPGA technology using ISE tools, all of the downstream FPGA implementation steps including synthesis place and route are automatically performed to generate an FPGA programming file. System Generator automates the design process, debugs, implement and verifies the Xilinx-based FPGAs. It provides a high speed HDL co-simulation interfaces which give up to a 1000x simulation performance increase. System Generator also supports a black box block that allows RTL to be imported into Simulink and co-simulated with either Modelsim or Xilinx ISE simulator

1.2 Design flow for image processing using Xilinx System Generator

The algorithms are developed and models are built using library provided by Xilinx Blockset. These models are simulated in Matlab/Simulink environment. The reflected result is viewed on a video viewer. The resultsobtained from System Generator are configured for suitable FPGA implementation. The behavioral model is verified, synthesized and implemented on FPGA. The Xilinx System Generator itself has the feature of generating user constraints file (.ucf), test bench and test vectors for testing architecture.

2 Methodology of implementation of image processing in hardware

All required hardware algorithms are implemented in between image pre-processing and image post-processing as depicted in Fig-2. Image source, image viewer, Image Pre-Processing and Image Post-Processing units are common for the entire image processing applications and they are implemented in Simulink

.

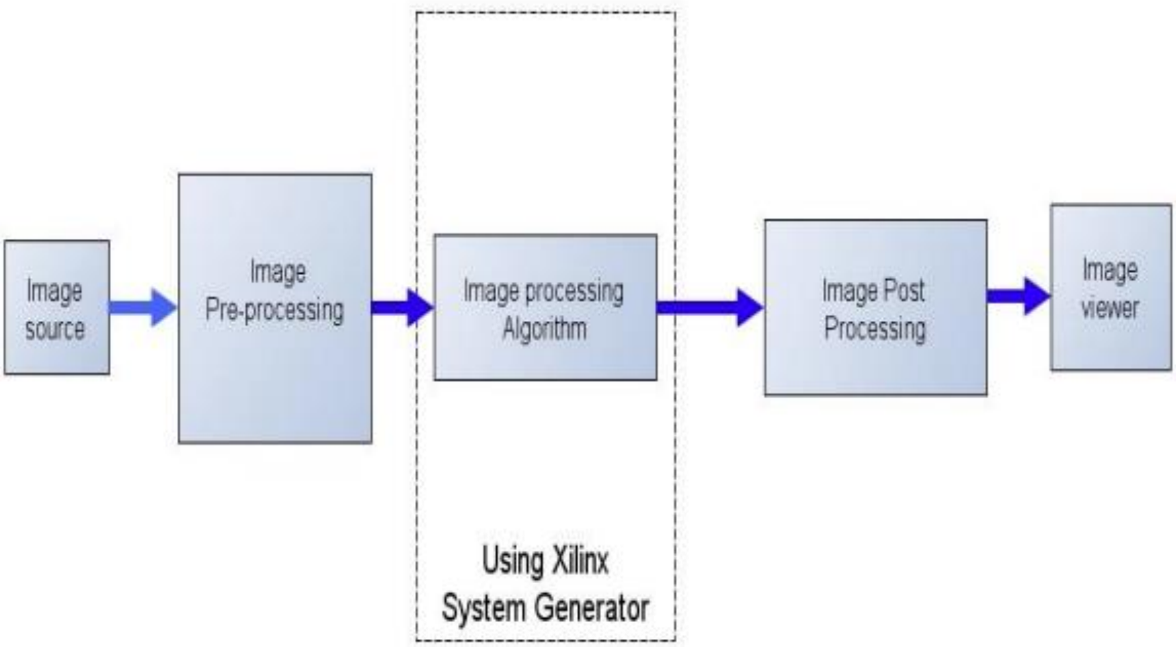

Figure 1: Design flow of hardware implementation of image processing

3 Image Pre-Processing Unit

Image pre-processing in Matlab helps in providing input to FPGA as specific test vector array which is suitable for FPGA bit stream compilation using System Generator. The Image Pre-Processing unit block is shown in Fig-2. Resize, Convert RGB to Gray Scale, 2D to 1D conversion and unbuffer are implemented in this unit. The conversion from 2-D to 1-D data is needed as FPGAs operate in one dimensional data only.

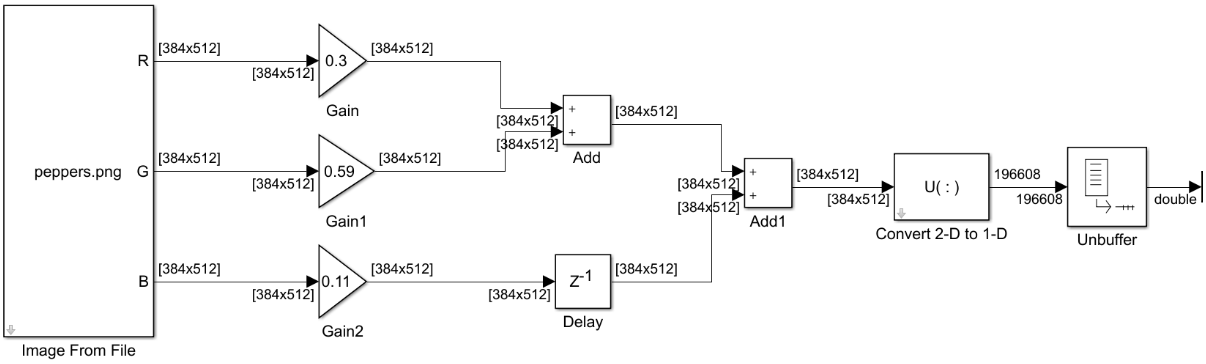

Figure 2: Image pre-processing Unit

4 Image Post-Processing Unit

Image post-processing helps in recreating image from 1-D array. It consists of three blocks: Buffer, Convert 1-D to 2-D, and video viewer. The first block converts scalar samples to frame output at lower sampling rate. Second block converts 1-D image signal to 2-D image matrix. The last block is used to display the output image back on the monitor. Fig-3 depicts the Image Post-processing steps in block diagram form.

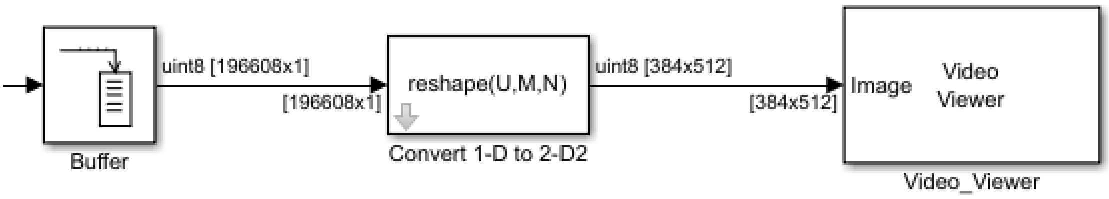

Figure 3: Image Post-Processing Unit

5 Algorithm for gray scale image negative

Negative image is obtained by simply inverting the image matrix. Such produced image looks like the negative of the film. In MATLAB, this is obtained by inverting the image source with NOT gate or by using Addsub block, subtracting one input by constant 255. Both the Addsub and NOT gate is available in Xilinx System Generator library which make steps simpler to use. The algorithm used is shown in Fig-4.

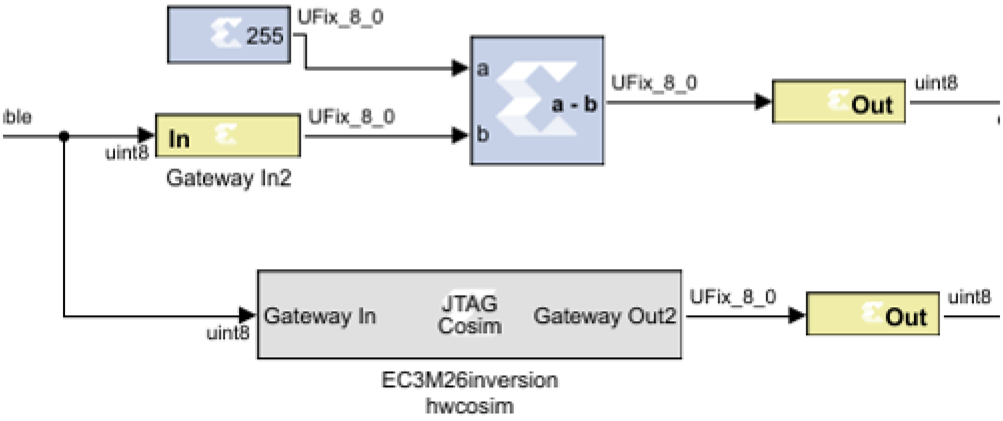

Figure 4: Algorithm for Image Negative using Addsub Block for Gray scale Image

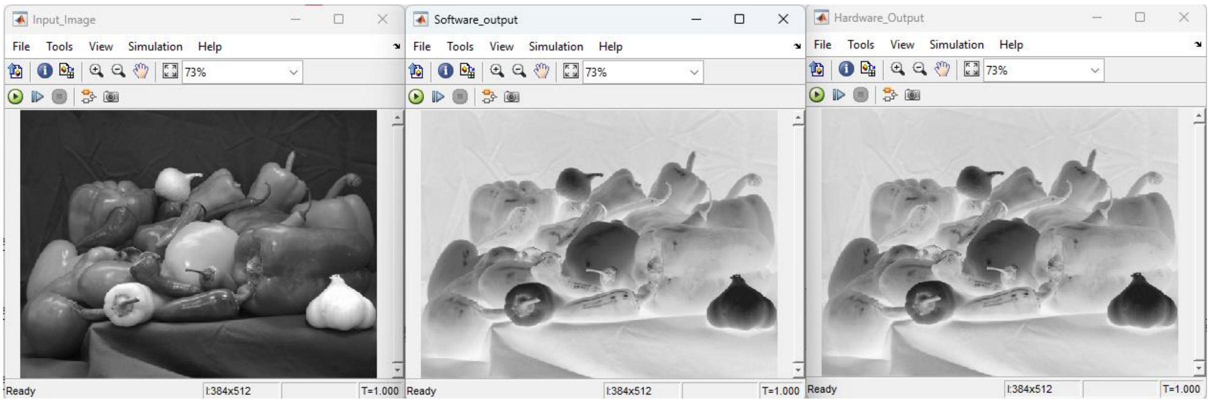

Figure 5: Result obtained from Grayscale Negative

6 Algorithm for Image contrast stretching

Grayscale image is stretched according to the equation. New_ pixel =3(old pixel-127) +30 where, New_pixel is its result after the transformation.

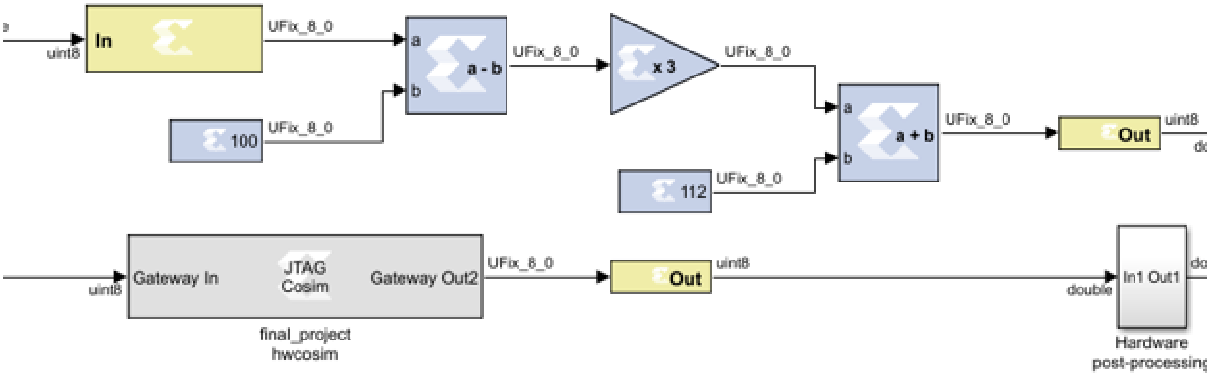

Figure 6: Algorithm for Image contrast stretching

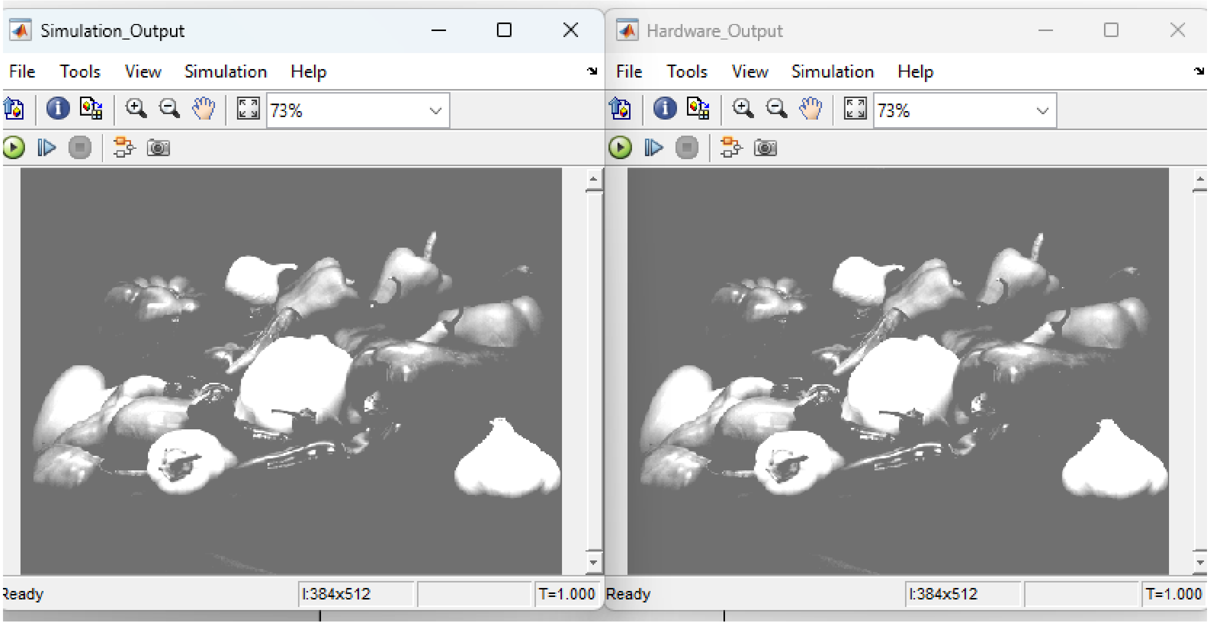

Figure 7: Result obtained from Grayscale Contrast stretching

7 Algorithm for Image Thresholding

Thresholding an image is the method of replacing each pixel in an image with a black pixel in the image intensity if it is less than some fixed constant value or white pixel if the image intensity is greater than that constant. Fig-8 shows the basic block diagram of thresholding while Fig-9 depicts the result that we implemented in the FPGA.

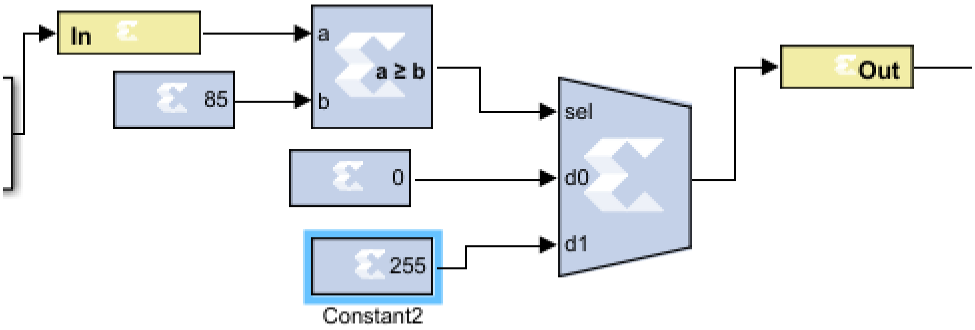

Figure 8: Algorithm for Grayscale image thresholding

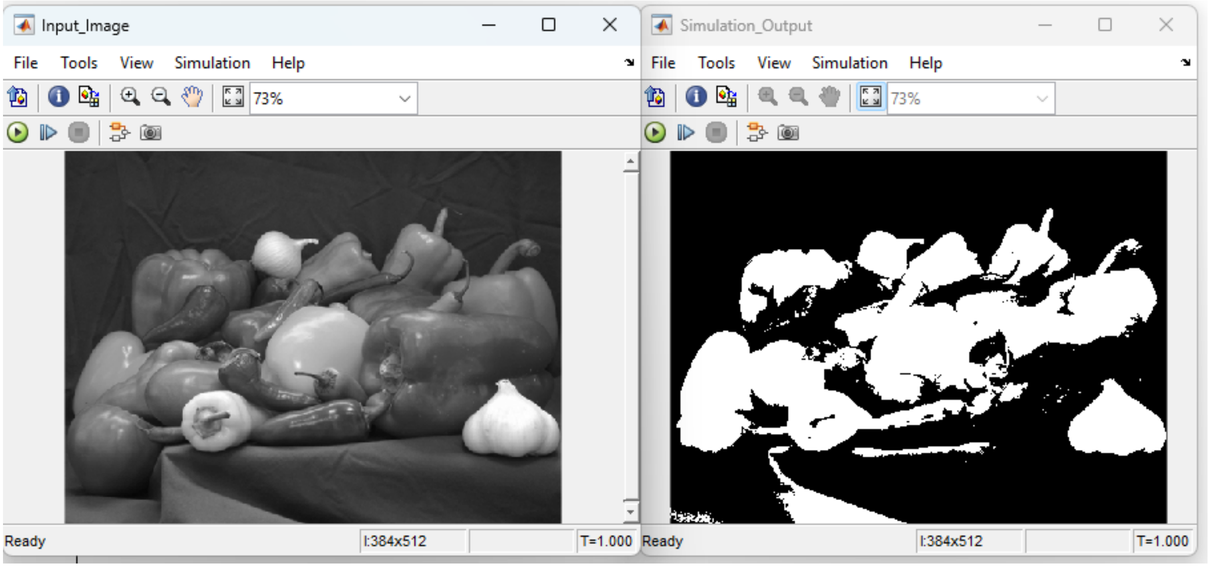

Figure 9: Result of Grayscale Image thresholding

8 Hardware/Software Co-Simulation in System Generator

System generator provides hardware co-simulation making it possible to incorporate a design running on an FPGA directly into Simulink simulation. When the system design is simulated in Simulink, result for the compiled portion are calculated in actual FPGA hardware, often resulting in significantly faster simulation time while verifying the functional correctness of the hardware.

9 Conclusions

Xilinx System Generator is integrated with Matlab Simulink for the real time image processing algorithms. Hardware co-simulation is used during the FPGA verification.

10 REFERENCES

[1] S. S. D. Praveen Vanaparthy, "FPGA Implementation of Image Enhancement Algorithms for Biomedical Image Processing," International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering , vol. 2, no. 11, pp. 5747 - 5753, November 2013.

[2] N.P.A.A.Gokhale, “FPGA Implementation for Image Processing Algorithms Using Xilinx System Generator”, IOSR Journal of VLSI and Signal Processing, vol.2(4),pp.26-36.

[3] E. Kuhn, "Grayscale Conversion of a Color Image Using Simulink and Xilinx Blocks," 2010.

[4] Xilinx Inc., *Synthesis and Simulation Design Guide*. June 2008.

[5] R.G.R. Woods, “Digital Image Processing”, New Jersey; Prentice-Hall, 2008