FPGA Implementation of Reconfigurable Processor for Image Processing

Ms. Payal S. Kadam, Prof. S.S.Belsare

Abstract: In this research paper I have proposed the method of FPGA implementation of reconfigurable processor for image processing application including morphology operations for medical, photoshop and satellite application purpose. Here reconfigurable processor is used for different image processing applications instead of application specific chips or ICs. Field Programmable Gate Array technology has become a suitable method for converting different ideas into reality that can be used for various image processing applications.In this project we are dealing with four applications like Median filter, Edge detection, Erosion and dilation.Here the digital image is taken as input image and we create header file of that with the help of Matlab. In MATLAB, input image is converted into grayscale image and pixel values are calculated. After that we are write C code for Median filter, Edge detection, Erosion, Dilation. We have used the Xilinx Platform Studio (XPS) for embedding c code and header file. For dumping code and pixel values into the FPGA kit we take the help of JTAG and RS232 cable.The purpose of JTAG is to send data parallely to XC3S200 IC. After processing, the processed data is sent back serially with the help of RS232. The primary idea of this project is to use reconfigurable processor for different image processing applications. Field Programmable Array Logic (FPGA) board and Spartan3EDK tool is used so as to increase the capacity and add efficiency.

Keywords: MATLAB, FPGA, Xilinx Plat Form Studio 10.1, Spartan 3EDK, Visual basic window.

——————————   ◆   ——————————

1. Introduction:

Image processing plays an important role in different fields such as medical, security, video processing and motion detection. Digital image consist of finite number of elements which has specific value and location. These elements are called as pixels. By taking these pixels into account we can process image for different applications. In many fields it is common to use application specific chips. Application specific chips are generally used for real time image processing. But these types of application specific chips have limited capacity and are used for intended applications only. It can be advantageous to use general purpose or reconfigurable processor.

The moto of this work is to use a reconfigurable processor for digital image processing. Binary image is a digital image that has two values either 0(low) or 1(high). Binary image is also knowns as bi-level. There are many advantages of using digital image processing over analog image processing techniques which includes noise immunity, less storage requirements, requires less band-width and can be transmitted over long distance.

In this paper I have proposed a reconfigurable processor that can perform filtering, edge detection, erosion and dilation. I have used MATLAB for taking the input image as it provides graphical user interface that can display image that is to be processed. The input image can be any digital image or can be real time image. We can use same image for all the four applications or we can use different images for different applications. This provides more flexibility for processing digital images. The processing of image takes place in FPGA which reduces the time and increases the capacity. FPGS is 'Field Programmable' so it can configure by user after manufacturing and it contains programmable logic blocks.

The main project has following steps:

1. We create a header file of an input image with the help of MATLAB.

2. Then using Xilinx Platform Studio (XPS) and FPGA kit we perform different image processing applications.

3. Output image is displayed on Visual Basic window.

Advantages:

1. Wide application range: As processor is re-configurable we can use same processor for different image processing application.

2. Less expensive: As we are using only one recon-figurable processor the whole system becomes less expensive.

3. Simple structure: The process of all the applications is same only we are changing the c code; this makes structure of the system simple.

The rest of paper is as follows. Description of morphological application such as median filter, edge detection, ero-

————————————————

•*Miss.Payal S. Kadam is currently pursuing masters degree program in electronics engineering (VLSI) in Bharathi Vidyapeeth DeemedUniversity College of Enginnering, Pune, India, PH-9423524715, E-mail: payalskadam94@gmail.com.*

•*Prof.S.S.Belsare, Assistant Professor Electronics and Telecommunication Department in Bharati Vidyapeeth Demmed University College of Engineering, Pune, India, PH-9822824647.*

E-mail:ssbelsare1@gmail.com.

sion, dilation is given in section 2. Block diagram is given in section 3. In the section 4 there are implementation and the results are given. Conclusion, acknowledgment and the references are given the section 5, 6 and 7 respectively.

2. Block Diagram:

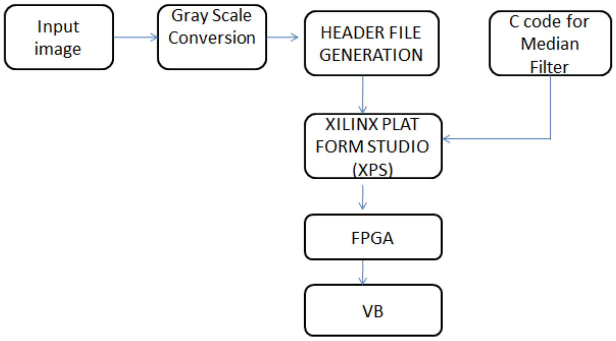


Fig1: Median filter Block diagram

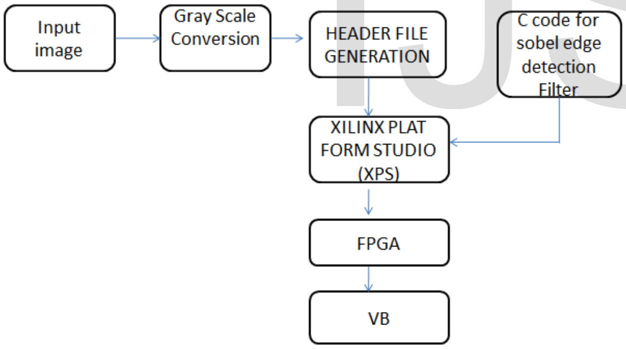

Input image
Gray Scale Conversion
HEADER FILE GENERATION
C code for sobel edge detection Filter
XILINX PLAT FORM STUDIO (XPS)
FPGA
VB

Fig2: Sobel edge detection Block diagram

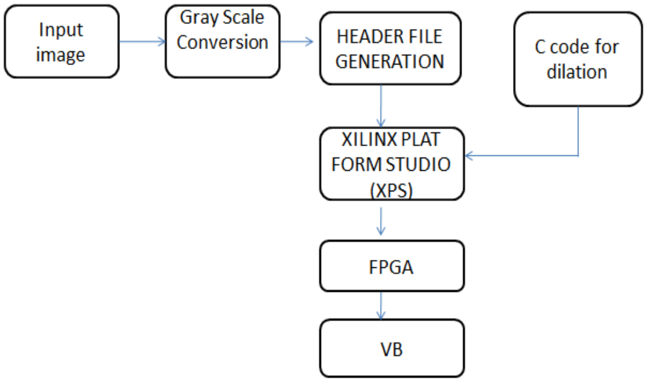


Fig3: Erosion Block diagram

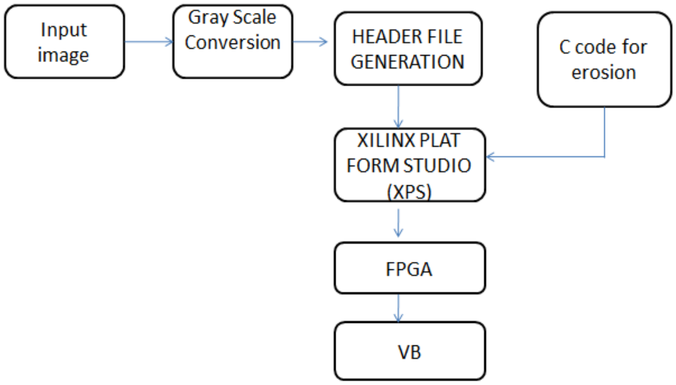


Fig4. Dilation Block diagram

3. Description of applications:

1. Median Filter

Median filtering is a nonlinear filtering process. It used to remove noise from images. While reducing noise it preserves the edges. Median filtering is known for reducing 'salt and pepper' type of noise. For filtering each and every pixel is taken into account. Filtering process includes selection of window of a particular size. Then that window is slided over entire image. The values of pixels in that window are arranged in ascending order and then the middle or median value is finalized. This value is used for replacement.
The median is calculated by first sorting all the pixel values from the window into numerical order, and then replacing the pixel being considered with the middle (median) pixel value.

Example: To illustrate, take a window of size and apply median filtering.

i = [3 81 7 4]
Calculations for new values:
Output [1] – [3 3 81] - middle value is 3
Output [2] - [3 81 7] – sorted matrix [3 7 81] – middle value is 7
Output [3] - [81 7 4] – Sorted matrix [4 7 81] –middle value is 7
Output [4] - [7 4 4] – Sorted matrix [4 4 7] – middle value is 4
Finally, i = [3 7 7 4]

2. Edge detection

When there is drastic change in intensity of pixel value compared to neighbouring pixel value then it is nothing but the edge.It is also known as boundary. It identifies the discontinuities of pixel value in given image. These discontinuities are then located to display the edges.

For edge detection we take the help of operators. Operators are used to calculate the gradients. These gradients are observed and gradients with high value are nothing but the edges. Different types of operators are used for edge detection.

Sobel Edge Detection

Here for edge detection we are using sobel operator. The sobel operator that we have used here has 3 rows and 3 columns. Sobel operator has two gradients. One gradient is in X-direction and other one is in Y-direction.

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

Derivatives are calculated with the help of gradients in x-direction as well as in y-direction. The final value of gradient is calculated using following equation.

$$G = \sqrt{G_x^2 + G_y^2}$$

Following are the steps of edge detection:
(1) Smoothing: reduce the noise by preserving the edges.
(2) Enhancement: sharpen the image by improving the quality of image.
(3) Detection: recognize the real discontinuity by avoiding the noisy pixels.
(4) Localization: determine the boundary between two discontinuities. This boundary is nothing but edge.

3. Dilation

Dilation is one of the morphological operations. Dilation is done on digital binary images. Some dilation techniques can also be applied to grayscale images. Here we are using dilation for filling the gaps between the broken alphabets. White pixels are called as foreground pixels. After applying dilation on the image of broken alphabets the gap gets filled as white part gets extended.

Steps for dilation-
1. Take any digital image or an real time image as an input image.
2. Convert image into grayscale image and calculate the pixel values. This is nothing but creation of header file using MATLAB.
3. By taking these values into account the grayscale image is converted into black and white image. For that pixel value equal to or greater than 128 is changed to 255 and below 128 to 0.
4. Then if pixel value is above 128 then replace neighbouring pixels with 0.
5. Print the output image with modified pixel values.

4. Erosion

Erosion is one of the morphological operations. It is some what analogous to dilation described above. Here also we are considering white pixels as foreground pixels. After applying erosion the white part reduces. For easy visualisation and understanding I have taken same input image that is used as an input image for dilation. By comparing the output of dilation and erosion we can easily get cleared with the concept.

Steps for performing Erosion-
1. Take any digital image or a real time image as an input image.
2. Convert image into grayscale image and calculate the pixel values. This is nothing but creation of header file using MATLAB.
3. By taking these values into account the grayscale image is converted into black and white image. For that image value equal to or less than 128 is changed to 255 and below 128 to 0.
4. Then if pixels values are less than 128 then replace neighbouring pixels with 0.
5. Print the output image with modified pixel values.

5. Implementation and results:

The images shown in fig5 is the how to create GUI file with the help of MATLAB, fig6 is the input image, fig7 is the header file creation, fig8 is the header file.

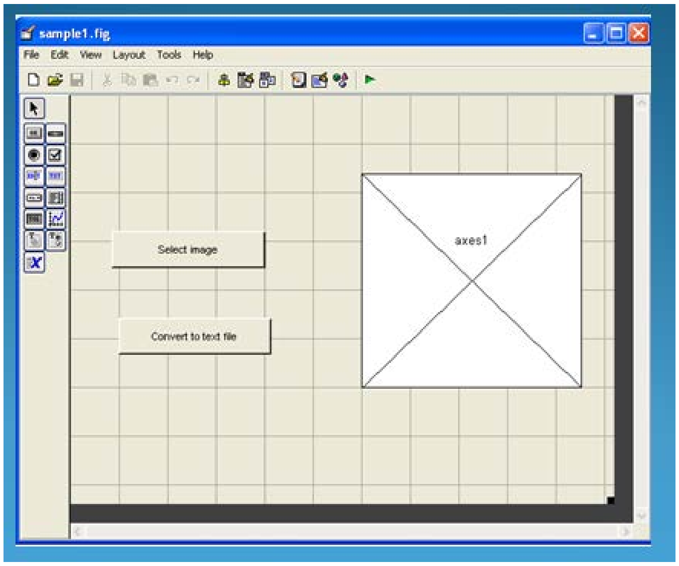


Fig5: GUI file of output window

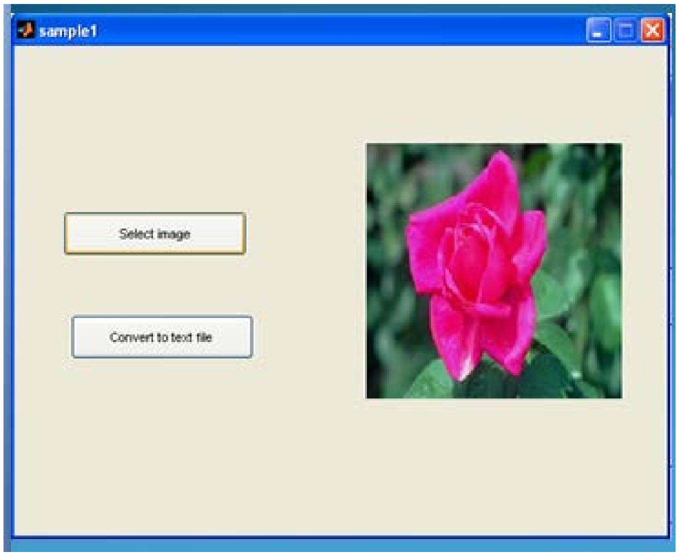
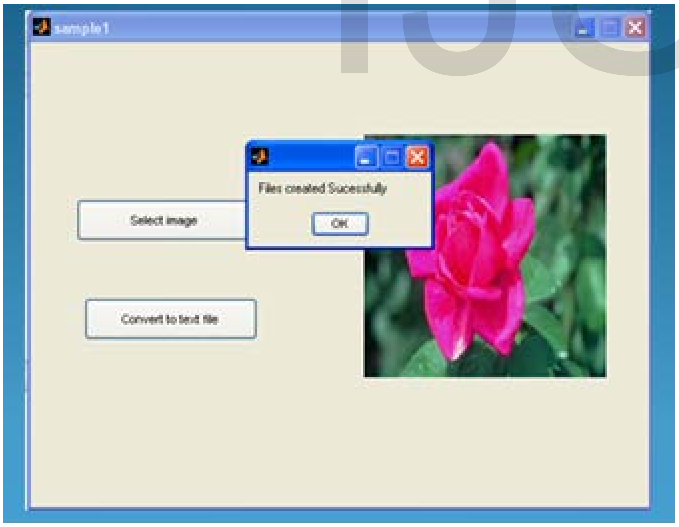
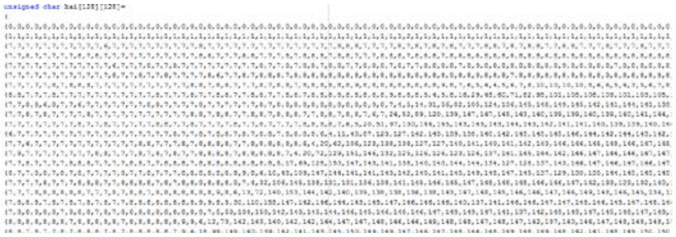


Fig6: Selection of input image



Files created Sucessfully

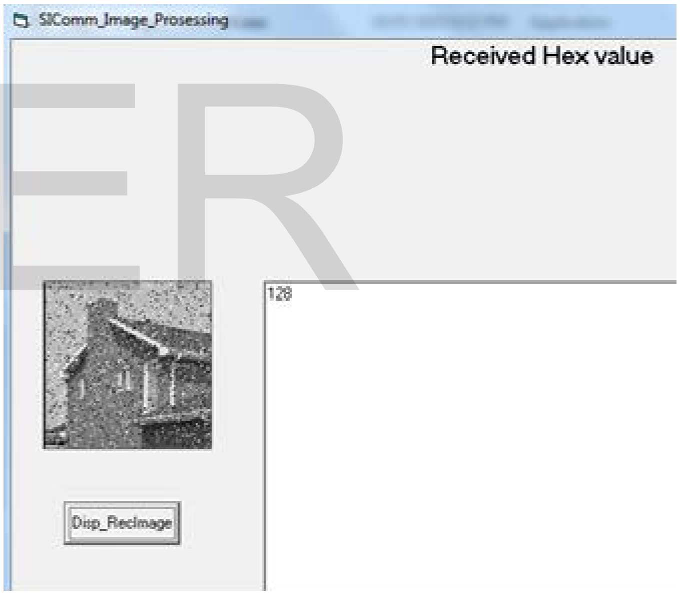
Fig7: Header file creation



```
unsigned char hai[128][128]=
{
[illegible]
```

Fig8: Header file

In this section results of various image processing applications are compared. Fig9 is input for median filter. Fig10 is the output after applying median filter. Fig11 is the input for sobel edge detection. Fig12 is the output after applying sobel edge detection. Fig13 is the input for dilation. Fig14 is the output after applying dilation. Fig15 is the input for erosion. Fig16 is the output after applying erosion.



128

Fig9: Original input image for median filter

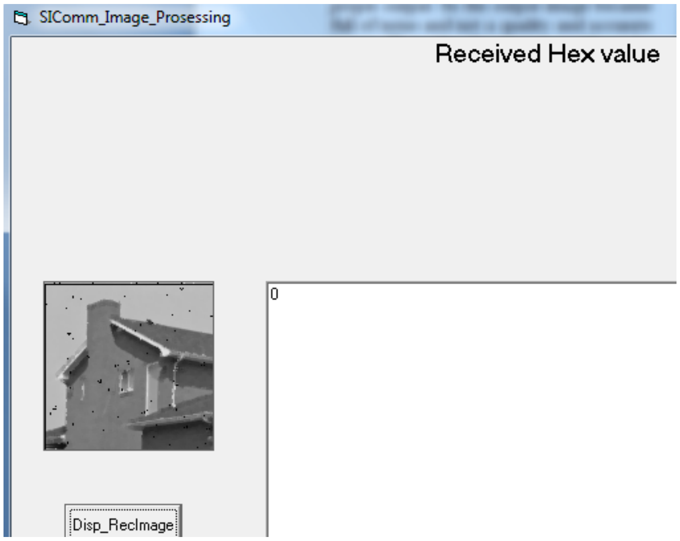


Fig10: Output image for median filter

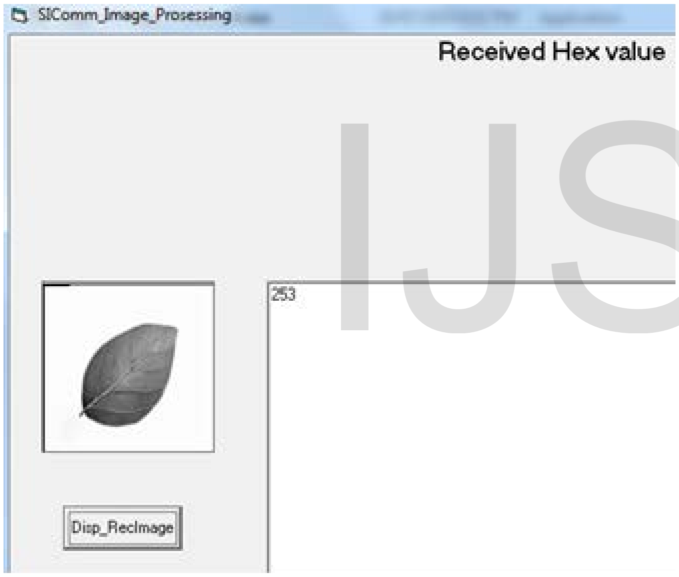


Fig11: Input image for sobel edge detection

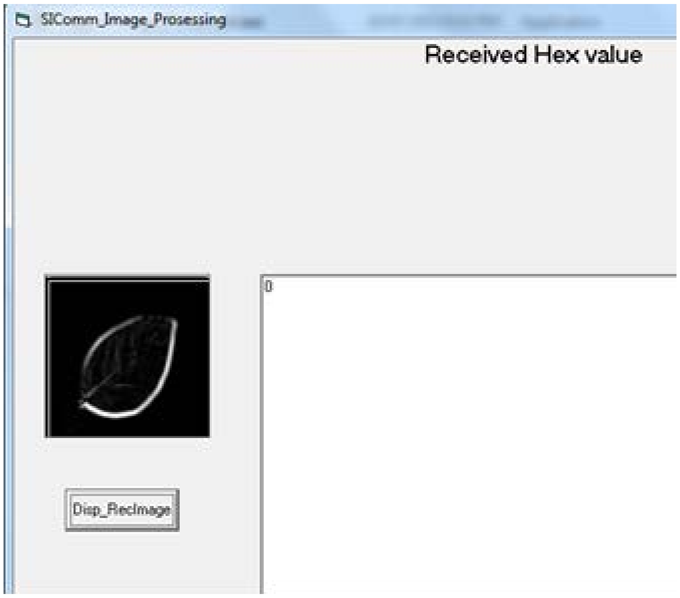


Fig12: Output image for sobel edge detection

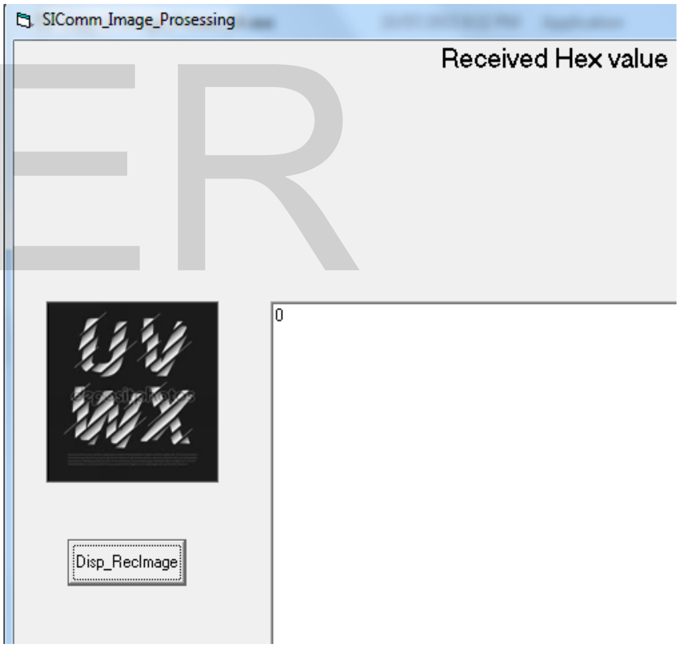


Fig13: Input image for dilation

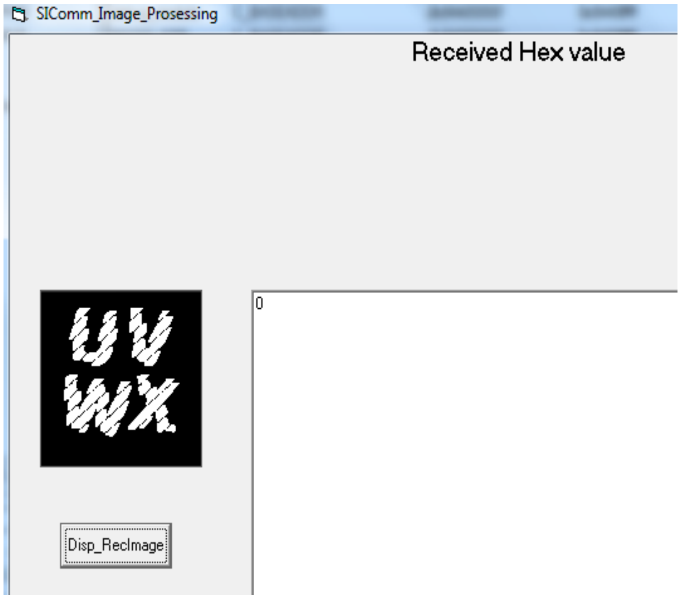


Fig14: Output image for dilation

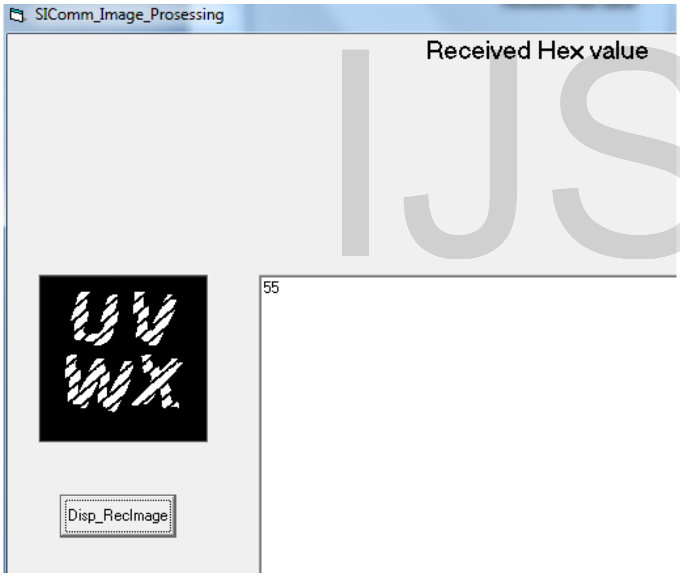


Fig15: input image for erosion

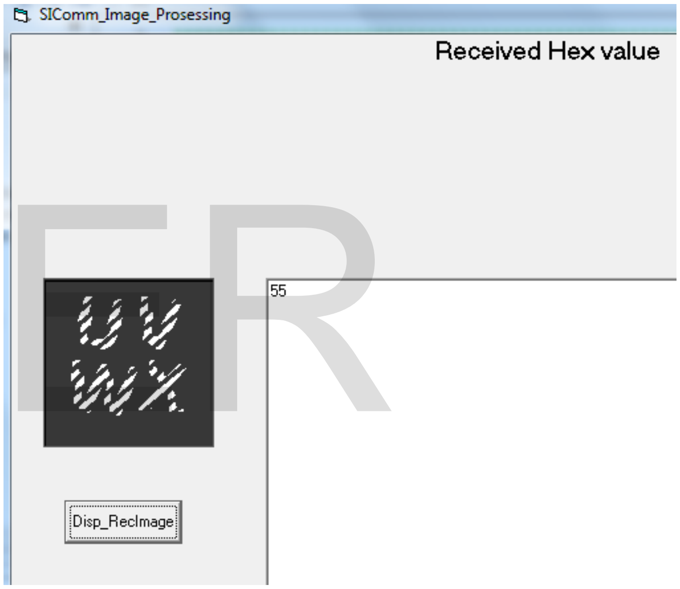


Fig 16: Output image for erosion

6. Conclusion

In this project we have proposed a reconfigurable processor that can be used for different image processing applications. For that we are considering the pixel values and converting them depending on the application to be per-formed. We are creating C code for different applications and processing is done in FPGA which improves the capacity. Here we have used recofigurable processor for applications such as filtering, edge detection, dilation and erosion.This reduces the cost and can be used for wide application range.

7. Acknowledgement

I am immensely grateful to Prof. S.S.Belsare who provided support and expertise that greatly assisted the research.

8. References

[1] J. Velten and A. Kummert, "Implementation of a high-performance hardware architecture for binary mor-phological image processing operations," in Proc. 47th IEEE Int. Midwest Symp. Circuits Syst., Jul. 2004, pp. 241–244.
[2] H. J. Park, K. B. Kim, J. H. Kim et al., "A novel motion detection pointing device using a binary CMOS image sensor," in Proc. IEEE Int.Symp. Circuits Syst., May 2007, pp. 837–840.
[3] H. Yang, A. C. Kot, and S. Rahardja, "Orthogonal data embedding for binary images in morphological transform domain: A high-capacity approach," IEEE Trans. Multimedia, vol. 10, no. 3, pp. 339–351, Apr. 2008.
[4] M. Wu and B. Liu, "Data hiding in binary image for authentication and annotation," IEEE Trans. Multimedia, vol. 6, no. 4, pp. 528–538, Aug.2004
[5] T. Ikenaga and T. Ogura, "A fully parallel 1-Mb CAM LSI for real-time pixel-parallel image processing," IEEE J. Solid-State Circuits, vol. 35, no. 4, pp. 536–544, Apr. 2000
[6] M. R. Lyu, J. Song, and M. Cai, "A comprehensive method for multilingual video text detection, localization, and extraction," IEEE Trans. Circuit Syst. Video Technol., vol. 15, no. 2, pp. 243–255, Feb. 2005.
[7] W. Miao, Q. Lin, W. Zhang et al., "A programmable SIMD vision chip for real-time vision applications," IEEE J. Solid-State Circuits, vol.43,no. 6, pp. 1470–1479, Jun. 2008.
[8] H. Yang and A. C. Kot, "Binary image authentication with tampering localization by embedding cryptographic signature and block identifier, IEEE Signal Process. Lett., vol. 13, no. 12, pp. 741–744, Dec. 2006.