International Journal on Applications in Electrical and Electronics Engineering
Volume 8 : Issue 2 : September 2022, pp 1 – 6 www.aetsjournal.com ISSN (Online) : 2395 - 3527

AN EFFICIENT VLSI IMPLEMENTATION OF EDGE DETECTION ALGORITHM BASED ON MMM DESIGN

S. THAMIZHELAKKIYA , R. RAJINI VISWANITHA , S. LILLYPET

Abstract*—** This paper introduces FPGA-based optimized implementation of Montgomery Modular Multiplier (MMM) architecture. The novel architecture of the proposed design enhanced the maximum frequency of the design and also the occupied area on the targeted FPGA. A Xilinx Spartan FPGA XC3S 200 TQ-144 implementation of the proposed architecture comparing with other related designs revealed that, our design occupies the smallest area, and the efficiency is enhanced in the range better times the efficiency of other relevant designs. The proposed design is implemented as a modular multiplier for lightweight elliptic curve cryptography (ECC) over general (p***). The proposed architecture is targeted the hardware implementation of lightweight cryptographic modules used Image Processing Application. The proposed Sobel edge detection algorithm uses approximation methods to replace the complex operations; the pipelining is employed to reduce the latency. This Design is implemented using Verilog HDL and simulated by Modelsim 6.4 c and synthesized by Xilinx tool.

Keywords— Sobel edge detection, FPGA, ECC, MMM design, Modular Multiplier

I. INTRODUCTION

Edge detection techniques have been successfully used for different applications. In edge detection, the abrupt changes in the pixel intensity are determined. These change in pixel intensities are determined by different techniques, in which different parameters are tuned to refine the edges of salient objects while suppressing the redundant objects from image. The edges obtained by different edge detector are broadly classified into two types: correct edges and false edges. Correct edge represent salient object and false edges are produced due to detector sensitiveness. The edge detection algorithms have three steps: filtering, enhancement and detection. Filtering is normally used to remove the noise from image. Enhancement is used to magnify the pixel intensity values in local area of an image and in detection the strong edges are determined. Recent research in the fields of Artificial Intelligence computer vision and Pattern Recognition reveals that the edge detection is very important in some way or the other. Key point detection is one major part of the process that majorly deals with image edges not only edges but also true edges. Identified key points are then used to describe the feature vectors that are further used in different applications. There is much research going on nowadays on edge detection as it has a key role in almost all upcoming fields.

S. Thamizh Elakkiya , Department of VLSI , Vandayar Engineering College, Pulavarnatham, Thanjavur.
(Email : tamileni1999@gmail.com)
R. Rajini Viswanitha ME , Department of VLSI , Vandayar Engineering College, Pulavarnatham, Thanjavur.
S. Lillypet ME , Department of VLSI , Vandayar Engineering College, Pulavarnatham, Thanjavur.

A) SOBEL EDGE DETECTION OPERATOR

In case of Sobel Edge Detection there are two masks, one mask identifies the horizontal edges and the other mask identifies the vertical edges. The mask which finds the horizontal edges that is equivalent to having the gradient in vertical direction and the mask which computes the vertical edges is equivalent to taking in the gradient in horizontal direction. Sobel masks are given in the bellow Fig.

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

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

Tabel 1.1 Sobel Operator

By passing these two masks over the intensity image the gradient along x direction (Gx) and gradient along the y direction (Gy) can be computed at the different location in the image. Now the strength and the direction of the edge at that particular location can be computed by using the

gradients Gx and Gy. The robust threshold computation method with high accuracy has been introduced, and utilizes more resources with increased computational complexity. The sobel edge detection circuit are applied to an end-user camera equipment should have lower hardware cost. In previous implementation the lower hardware cost is achieved by employing rough calculations to replace the complicated operations and reduction in input data to meet the real-time applications. However, it results in lower accuracy. Thus, there is a trade-off between hardware cost and accuracy. Existing System Technique Robust Threshold Computation Existing system drawbacks are Low Accuracy, High Cost, Low operation speed and more Delay.

II. RELATED WORKS

1)A low-cost real-time embedded stereo vision system for accurate disparity estimation based on guided image filtering, 2015, C. Ttofis, C. Kyrkou, and T. Theocharides,

This paper presents the design and implementation of a hardware-based stereo matching system able to provide high accuracy and concurrently high performance for embedded vision devices, which are associated with limited hardware and power budget.

2)A novel approach for edge detection based on the theory of universal gravity Pattern Recognition , 2007 G. Sun, Q. Liu, Q. Liu, C. Ji, and X. Li

A new edge detection algorithm is presented. The method is based on the law of universal gravity**.**

3) Guided image filtering, 2012, K. He , J. Sun, and X. Tang

Novel type of explicit image filter - guided filter.

4)Efficient image sharpening and denoising using adaptive guided image filtering, 2014, C. C. Pham and J. W. Jeon

The author's proposed adaptive GF (AGF) integrates the shift variant technique, a part of ABF, into a guided filter to render crisp and sharpened outputs.

III. PROPOSED EDGE DETECTION DESIGN

This paper purposed Edge Detection using Sobel Operator in Digital Image Processing and implementation using Verilog HDL. Firstly, a jpg image is inputted and converted into binary image with the help of MATLAB. Acquire a jpg image, which is by default in an RGB color space and convert this RGB image to grey level image. Now convert the grey level image into the binary image. This binary image is very large, so it is resized and written into a text file shown in the figure8.Further implementation is done on the Xilinx ISE and Modelsim. The Sobel operator is used commonly in edge detection. At each point in the image, the result of the Sobel operator is the corresponding norm of this gradient vector. The Sobel operator only considers the two orientations which are 0°and 90°convolution kernels. The operator uses the two kernels which are convolved with the original image to calculate approximations of the gradient. As given above, the gradients are calculated along with the magnitude in Verilog HDL synthesis and then it is simulated and checked out with respect to the design summery and timing analysis. Now with the help of Xilinx ISE read the text file generated by the MATLAB into the memory and store it into the RAM, then extract the Image window.

1)PROPOSED MULTIPLIER DESIGN

This paper introduces a novel and optimized design of the MMM algorithm, with higher efficiency, enhanced maximum frequency, and lower area costs, which is an essential aspect in the embedded systems architecture. The proposed algorithm is an improvement of the radix-2 MMM structure. No subtraction, no multiplication operation is performed in the proposed design, with one pre computed addition operation, and only one addition operation in the computation process to reduce the critical path delay and increase the maximum frequency. The previous modification of MMM to remove the final subtraction presented a new set of parameters with longer operands and more iterations; this could seriously reduce the efficiency.

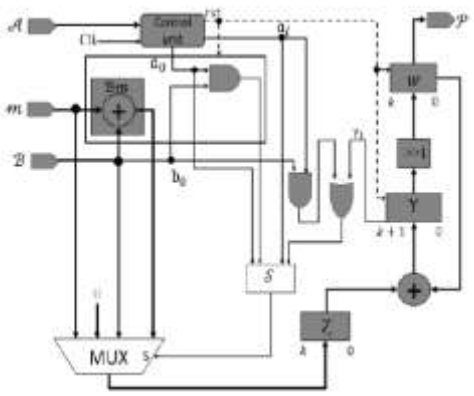


Figure 1 : Proposed Multiplier Design

MAIN BLOCK DIAGRAM

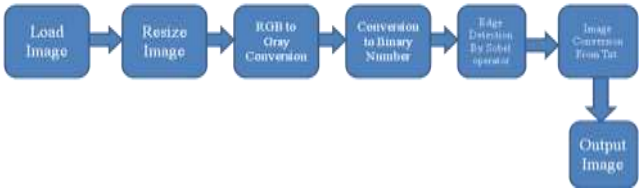


Figure 2 : Process Diagram

2)PROPOSED DESIGN FLOW CHART

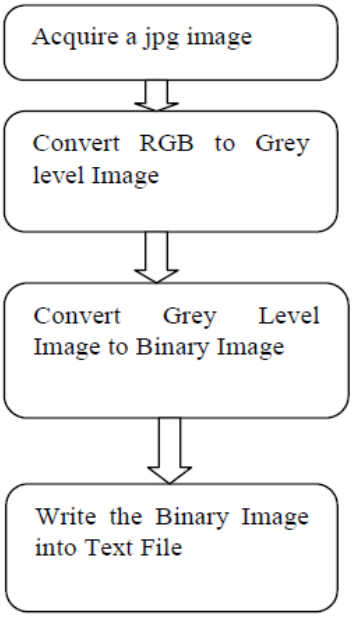


Figure 3 Process flow of RGB to binary image conversion (MATLAB Part)

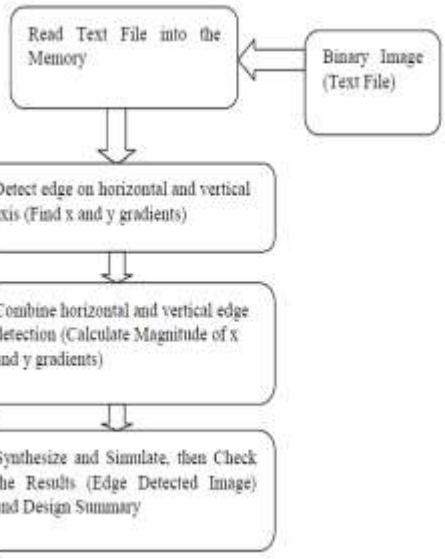
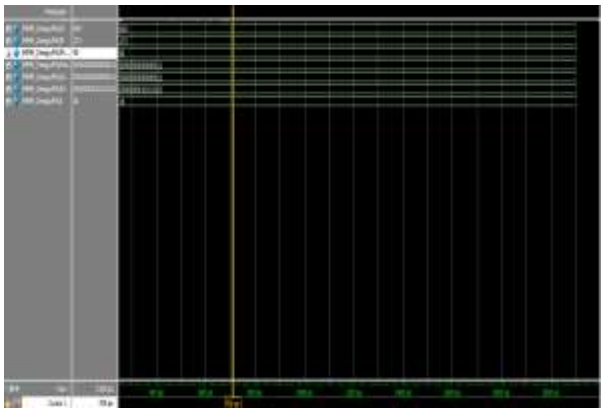


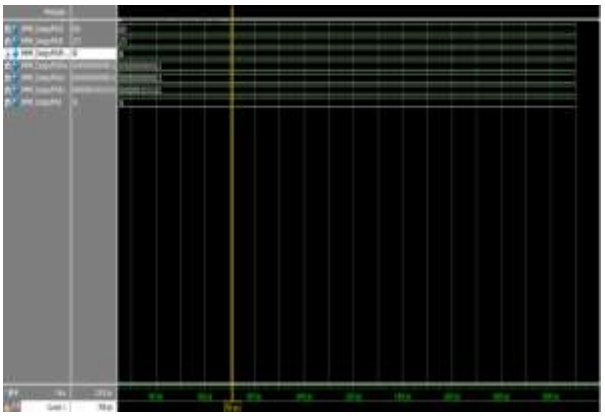
Figure .4 VLSI PART (USING Modelsim)

IV. SIMULATION RESULTS

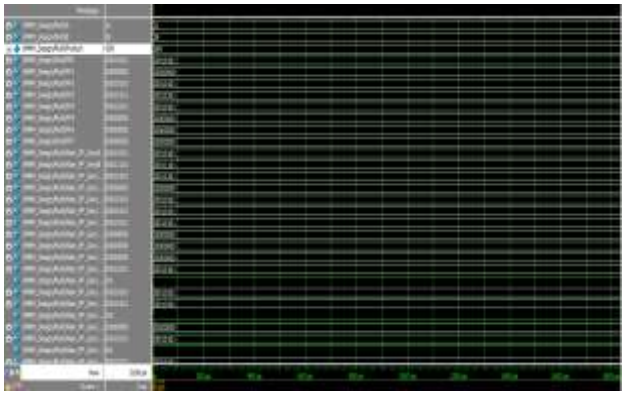
1)MMM DESIGN



2)MODULO OPERATION

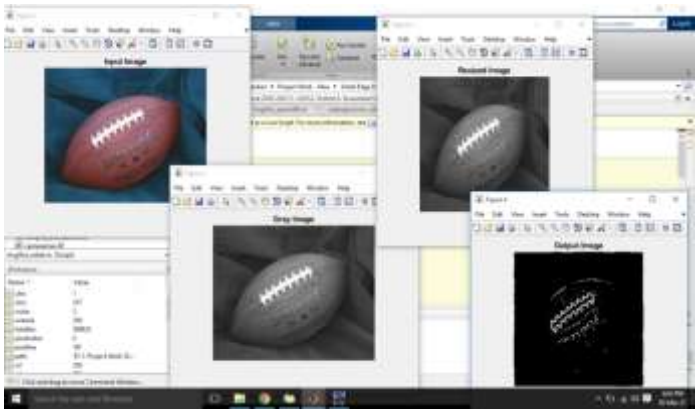


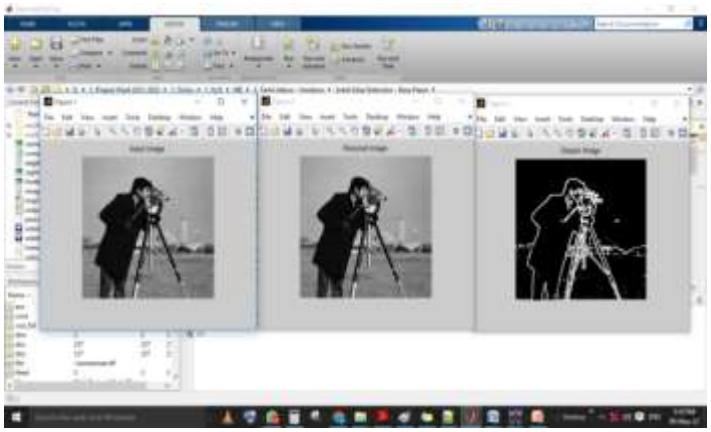
3)MULTIPLICATION UNIT



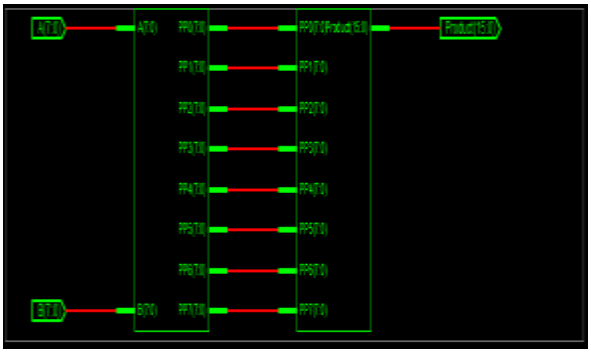
4) VLSI PART SOBEL PROCESS

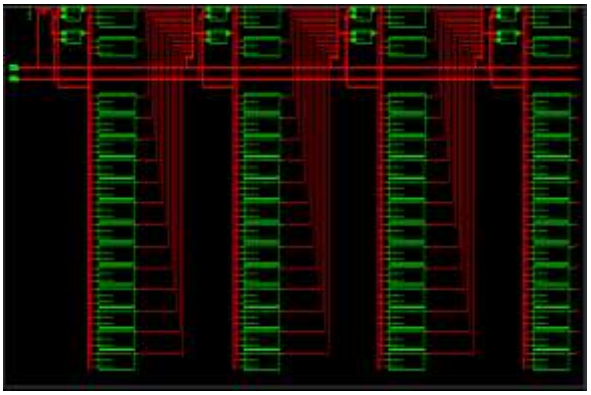
5) SOBEL OPERATOR BASED OUTPUT





6)RTL View of the proposed Design



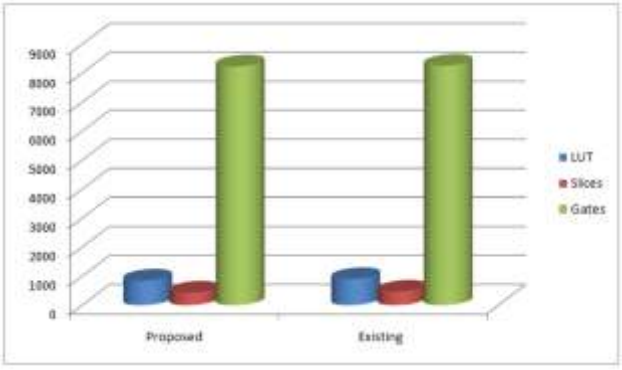


V. COMPARISON AND ANALYSES

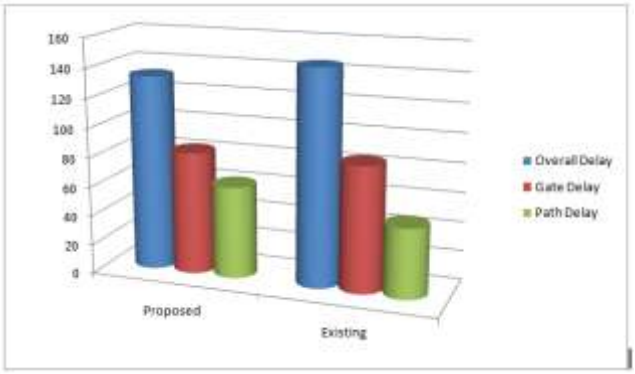
The suggested MMM design concepts are built in Verilog HDL, generated with Xilinx for various bit sizes, and the latency and area are compared. As seen in Fig. 4, MMM Design with Segmented and non segmented has the smallest area and has the shortest latency when compare to Conventional MMM as the number of bits increases. The results demonstrate that using proposed design for inclusion achieves the suggested Design's overall minimum area.

Method	Area			Delay		
Spartan 3	LUT	Slices	Gates	Overall Delay	Gate Delay	Path Delay
Proposed	899	467	8232	146.021ns	83.392ns	62.629ns
Modification	845	432	8256	132.941ns	85.353ns	47.588ns

1) AREA COMPARISON



2)DELAY COMPARISON



VI. CONCLUSION

This paper introduced an optimized architecture of the modular multiplier, designed to reduce the area and enhance the frequency of MMM algorithm. The proposed design offers the lowest area and greatest implementation efficiency on FPGA than the relevant designs of the modular multiplier. This paper mainly focuses on the design and simulation System of the Sobel edge detection method based on proposed MMM Design. This method uses two 3×3 convolution masks to estimate gradient in X and Y-direction and which is easy to implement than other operators. The Sobel edge detection method calculates 2-D spatial gradient of image intensity at each point of an image by convolution with small and integer valued filters therefore relatively less expensive in terms of computations. Application of Sobel edge detection algorithm to an image may considerably decrease the amount of details to be processed and with pulse width modulation, time-encoded signals corresponding to specific values are generated by adjusting the frequency and duty cycles of signals. With this approach, the latency, area consumption is all greatly reduced.

REFERENCES

[1] A. Alaghi and J. P. Hayes, "Survey of stochastic computing," ACM Trans. Embedded Comput. Syst., vol. 12, no. 2s, pp. 92:1–92:19, May 2013.

[2] J. P. Hayes, "Introduction to stochastic computing and its challenges," in Proc. 52nd ACM/EDAC/IEEE Design Autom. Conf. (DAC), Jun. 2015, pp. 1–3.

[3] B. R. Gaines, "Stochastic computing systems," in Advances in Information Systems Science, J. T. Tou, Ed. New York, NY, USA: Springer, 1969, pp. 37–172.

[4] P. Li, D. J. Lilja, W. Qian, K. Bazargan, and M. D. Riedel, "Computation on stochastic bit streams digital image processing case studies," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 22, no. 3, pp. 449–462, Mar. 2014.

[5] M. H. Najafi and M. E. Salehi, "A fast fault-tolerant architecture for sauvola local image thresholding algorithm using stochastic computing," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 24, no. 2, pp. 808–812, Feb. 2016.

[6] A. Alaghi, C. Li, and J. P. Hayes, "Stochastic circuits for real-time image-processing applications," in Proc. 50th ACM/EDAC/IEEE Design Autom. Conf. (DAC), May 2013, pp. 1–6.

[7] D. Fick, G. Kim, A. Wang, D. Blaauw, and D. Sylvester, "Mixed-signal stochastic computation demonstrated in an image sensor with integrated 2D edge detection and noise filtering," in Proc. IEEE Custom Integr. Circuits Conf. (CICC), Sep. 2014, pp. 1–4.

[8] S. S. Tehrani, W. J. Gross, and S. Mannor, "Stochastic decoding of LDPC codes," IEEE Commun. Lett., vol. 10, no. 10, pp. 716–718, Oct. 2006.

[9] N. Onizawa, W. J. Gross, T. Hanyu, and V. C. Gaudet, "Asynchronous stochastic decoding of LDPC codes: Algorithm and simulation model," IEICE Trans. Inf. Syst., vol. 97, no. 9, pp. 2286–2295, 2014.

[10] B. D. Brown and H. C. Card, "Stochastic neural computation. I. Computational elements," IEEE Trans. Comput., vol. 50, no. 9, pp. 891–905, Sep. 2001.

[11] K. Kim, J. Kim, J. Yu, J. Seo, J. Lee, and K. Choi, "Dynamic energyaccuracy trade-off using stochastic computing in deep neural networks," in Proc. 53rd Annu. Design Autom. Conf. (DAC), New York, NY, USA,2016, pp. 124:1–124:6.

[12] B. Li, M. H. Najafi, and D. J. Lilja, "Using stochastic computingto reduce the hardware requirements for a restricted Boltzmannmachine classifier," in Proc. ACM/SIGDA Int. Symp. Field-Program.Gate Arrays (FPGA), New York, NY, USA, 2016, pp. 36–41.

[13] Y. Liu, H. Venkataraman, Z. Zhang, and K. K. Parhi, "Machine learningclassifiers using stochastic logic," in Proc. IEEE 34th Int. Conf. Comput.Design (ICCD), Oct. 2016, pp.408–411.

[14] W. Qian, X. Li, M. D. Riedel, K. Bazargan, and D. J. Lilja, "An architecturefor fault-tolerant computation with stochastic logic," IEEE Trans.Comput., vol. 60, no. 1, pp. 93–105, Jan. 2011.

[15] Q. Tang, B. Kim, Y. Lao, K. K. Parhi, and C. H. Kim, "True randomnumber generator circuits based on single- and multi-phase beat frequencydetection," in Proc. IEEE Custom Integr. Circuits Conf. (CICC),Sep. 2014, pp. 1–4.

[16] W. H. Choi et al., "A magnetic tunnel junction based true randomnumber generator with conditional perturb and real-time output probabilitytracking," in Proc. IEEE Int. Electron Devices Meeting (IEDM),Dec. 2014, pp. 12.5.1–12.5.4.