International Journal on Applications in Electrical and Electronics Engineering
Volume 8 : Issue 2 : September 2022, pp 19 – 25 www.aetsjournal.com ISSN (Online) : 2395 - 3527

DESIGN OF POWER AND AREA EFFICIENT APPROXIMATE MULTIPLIERS FOR EDGE DETECTION ALGORITHM

SABITHASHINI . S , Dr. B. SYED MOINUDDIN BOKHARI

Abstract— In this paper, we present a novel approximate computing scheme suitable for realizing the energy-efficient multiply-accumulate (MAC) processing . First we design the approximate 4-2 compressors generating errors in the opposite direction while minimizing the computational costs. Based on the probabilistic analysis, positive and negative multipliers are then carefully developed to provide a similar error distance. Simulation results on various practical applications reveal that the proposed MAC processing offers the energy-efficient computing scenario by extending the range of approximate parts . This Design is implemented by Verilog HDL and simulated by Modelsim 6.4 c. The Performance is measured by Xilinx tool Synthesis Process. The proposed Sobel edge detection algorithm uses approximation methods to replace the complex operations; This design is done by Matlab and Modelsim using hdldameon, This proposed multipliers are replaced in the sobel operator based image Edge detection.

Keywords— Sobel edge detection , FPGA , MAC , Approximate design , Approximate 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.

1)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

Sabithashini . S , *Department of VLSI DESIGN* , M.A.M College of Engineering and Technology , *Trichy.*

(Email : sabitha17494shini@gmail.com)

Dr. B. Syed Moinuddin Bokhari , *Department of VLSI DESIGN* , M.A.M College of Engineering and Technology , *Trichy.*

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)Bio-Inspired Imprecise Computational Blocks For Efficient Vlsi Implementation Of Soft-Computing Applications, H. R. Mahdiani, A. Ahmadi, S. M. Fakhraie, and C. Lucas.

The conventional digital hardware computational blocks with different structures are designed to compute the precise results of the assigned calculations. The main contribution of our proposed Bio-inspired Imprecise Computational blocks (BICs) is that they are designed to provide an applicable estimation of the result instead of its precise value at a lower cost. These novel structures are more efficient in terms of area, speed, and power consumption with respect to their precise rivals. Complete descriptions of sample BIC adder and multiplier structures as well as their error behaviours and synthesis results are introduced in this paper**.**

2)Low voltage, low power (5:2) compressor cell forfast arithmetic circuits, Jiangmin gu and chip-hong chang.

This paper presents a new (5:2) compressor circuit capable of operating at ultra-low voltages. Its power efficacy is derived from the novel design of composite XOR-XNOR gate at transistor level. The new circuit eliminates the weak logic and threshold voltage drop problems, which are the main factors limiting the performance of pass transistor based circuits at low supply voltages. The proposed (5:2) compressor has been designed with special consideration on output drivability to ensure that it can function reliably at low voltages when these cells are employed in the tree structured multiplier and multiply accumulator. Simulation results show that the proposed (5:2) compressor is able to function at supply voltage as low as 0.7V, and outperforms other (5:2) compressors constructed with various combinations of recently reported superior low-power logic cells.

3)Ultra low-voltage low-power cmos 4-2 and 5-2 compressors for fast arithmetic circuits. Chip-hong chang, senior member, ieee, jiangmin gu, student member, ieee, and mingyan zhang, student member, ieee.

This paper presents several architectures and designs of low-power 4-2 and 5-2 compressors capable of operating at ultra low supply voltages. These compressor architectures are anatomized into their constituent modules and different static logic styles based on the same deep sub micrometer CMOS process model are used to realize them. Different configurations of each architecture, which include a number of novel 4-2 and 5-2 compressor designs, are prototyped and simulated to evaluate their performance in speed, power dissipation and power-delay product. The newly developed circuits are based on various configurations of the novel 5-2 compressor architecture with the new carry generator circuit, or existing architectures configured with the proposed circuit for the exclusive OR (XOR) and exclusive NOR (XNOR) [XOR–XNOR] module. The proposed new circuit for the XOR–XNOR module eliminates the weak logic on the internal nodes of pass transistors with a pair of feedback PMOS–NMOS transistors. Driving capability has been considered in the design as well as in the simulation setup so that these 4-2 and 5-2 compressor cells can operate reliably in any tree structured parallel multiplier at very low supply voltages. Two new simulation environments are created to ensure that the performances reflect the realistic circuit operation in the system to which

these cells are integrated. Simulation results show that the 4-2 compressor with the proposed XOR–XNOR module and the new fast 5-2 compressor architecture are able to function at supply voltage as low as 0.6 V, and outperform many other architectures including the classical CMOS logic compressors and variants of compressors constructed with various combinations of recently reported superior low-power logic cells.

4)Design Of A Approximate Compressor For Array Multiplier, Mrs. J.Sunitha kumari, B. Jhansi reddy.

Multiplication is a fundamental operation in most of the signal processing algorithms. Multipliers have large area, long latency and consume considerable power and the design of good multipliers is always a challenge for VLSI system designers. For this inconvenience compressor for low latency, low power consumption and reduced stages of product is designed. In this paper approximate compressor design for reduction of multiplier stages in the Array multiplier is proposed. These results are carried out using Tanner EDA tool.

5)A design technique for faster array multiplier, B. Ramkumar, V. Sreedeep and Harish M Kittur.

In this work faster column compression multiplication has been achieved by using a combination of two design techniques: partition of the partial products into two parts for independent parallel column compression and acceleration of the final addition using a hybrid adder proposed in this work. Based on the proposed techniques 8, 16, 32 and 64-bit Array Multiplierare developed and compared with the regular Array multiplier. The performance of the proposed multiplier is analyzed by evaluating the delay, area and power, with 180 nm process technologies on interconnect and layout using industry standard design and layout tools. The result analysis shows that the 64-bit regular Array multiplier is as much as slower than the proposed multiplier and requires only 1.4% and less area and power respectively. Also the power-delay product of the proposed design is significantly lower than that of the regular Array multiplier.

III. PROPOSED SYSTEM

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 its based on proposed multipliers. 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 Modeslim read the text file generated by the MATLAB into the memory and store it into the RAM, then extract the Image window.

1)PROPOSED COMPRESSORS DESIGN:

Three approximate 4-2 compressors (UCAC1, UCAC2, and UCAC3) are proposed in this section. Then, the ECM is presented to detect an input pattern with a large probability and correct the erroneous compensation in this case. Furthermore, the proposed designs are embedded in 8-bit multipliers based on the partial product tree. And all the analyses are performed with the uniform distribution. The proposed approximate compressors and ECM are designed to simplify and accelerate the compression process, four 8-bit multipliers (N = 8) are designed to evaluate these blocks, accordingly.

1) MUL1: Multiplier with UCAC1 and Constant Correcting Bit;
2) MUL2: multiplier with UCAC1 and ECM;
3) MUL3: multiplier with UCAC2 and ECM;
4) MUL4: multiplier with UCAC3 and ECM;

2)PROPOSED MULTIPLIER BLOCK DIAGRAM:

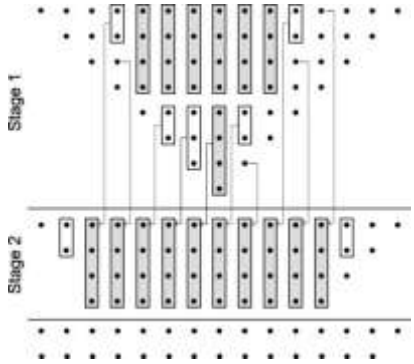


Figure 1 : Dadda Multiplier Using Proposed Designs

3)SOBEL EDGE DETECTION BLOCK DIAGRAM:

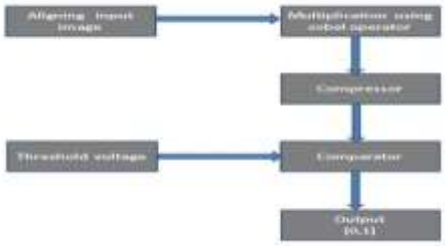


Figure 2

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.

4) PROPOSED DESIGN FLOW CHART

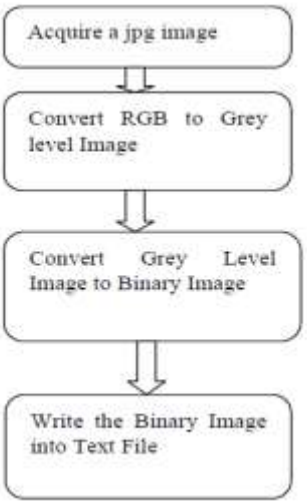


Figure 3 : Process flow of RGB to binary image conversion (MATLAB Part)

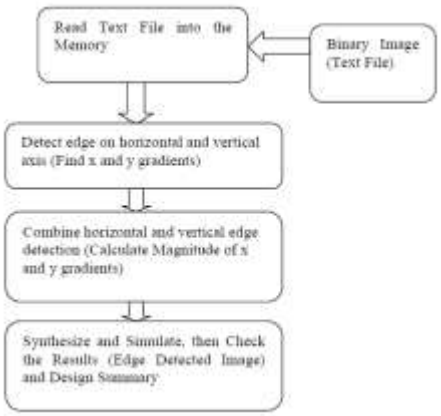
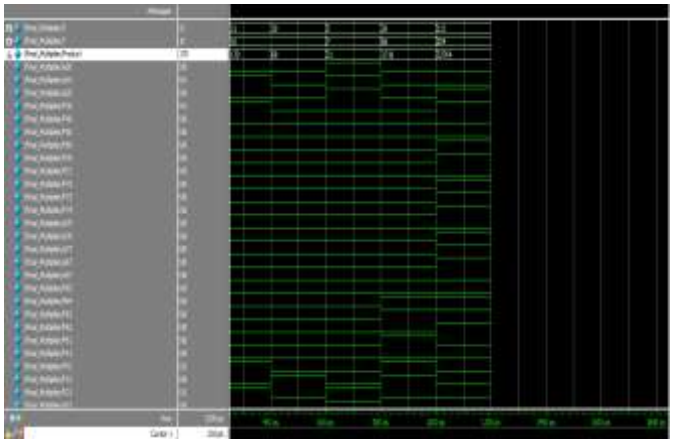


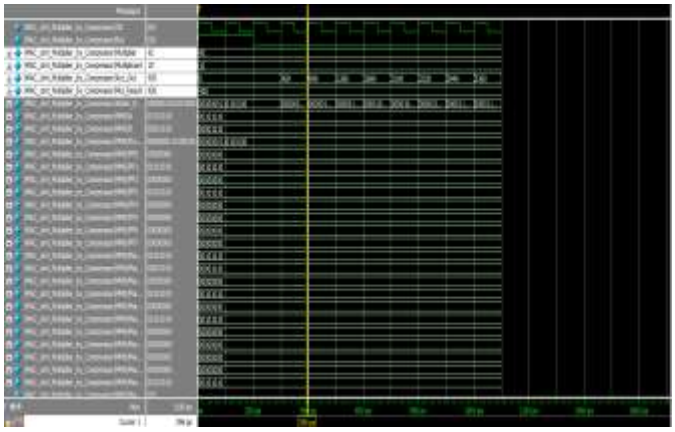
Figure 4 : VLSI PART (USING Modelsim)

IV. SIMULATION RESULTS

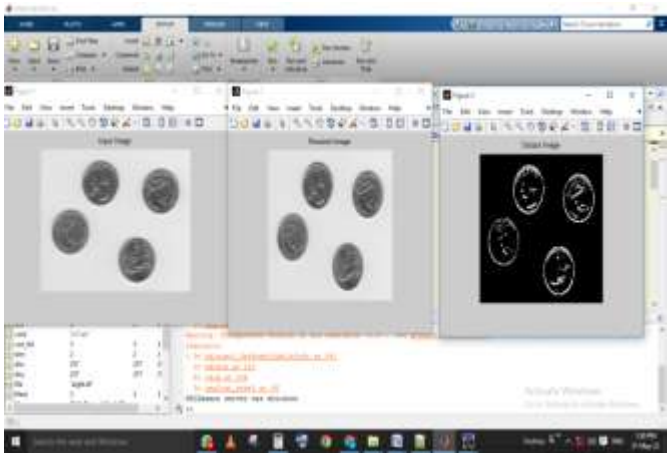
1) FINAL MULTIPLIER:



2)MAC DESIGN USING PROPOSED COMPRESSOR

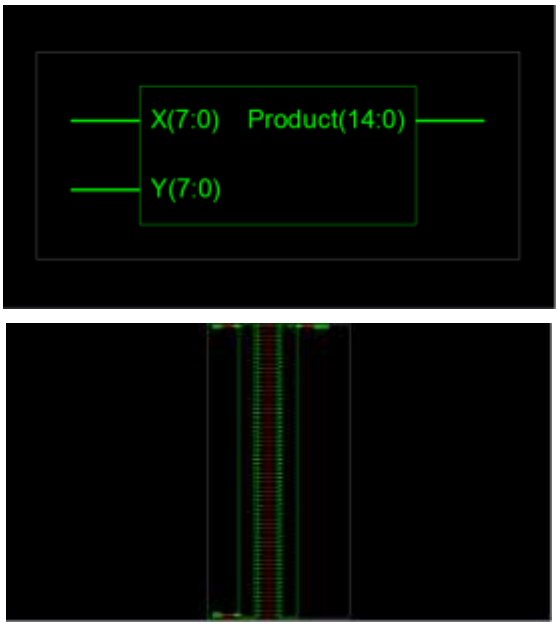


3)MATLAB CODE RESULTS



4)MODESLIM OUTPUT

5)RTL View of the proposed Design

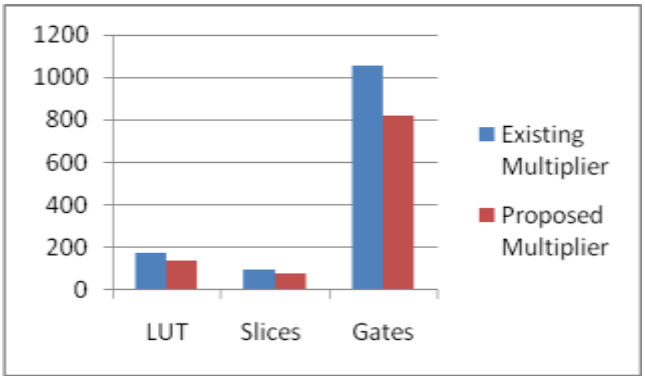


V. COMPARISON AND ANALYSES

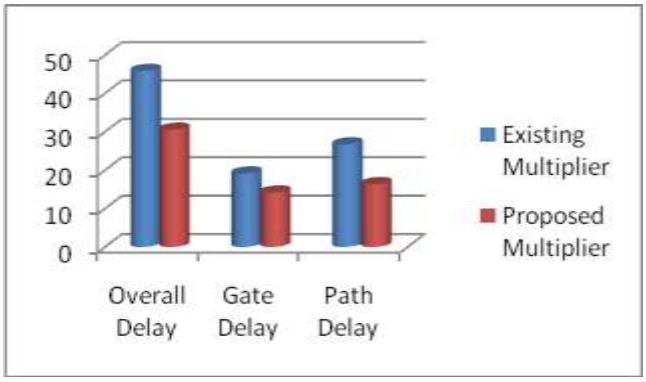
The suggested Approximate multiplier design concepts are built in Verilog HDL, generated with Xilinx for various bit sizes, and the latency and area are compared. As seen in Fig. 4, Multiplier Design with Segmented and non segmented has the smallest area and has the shortest latency when compare to Conventional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

S. No	Method Name	Area			Delay		
		LUT	Slices	Gates	Max Delay	Gate Delay	Path Delay
1	Existing Normal Multiplier	176	97	1056	45.844ns	19.138ns	26.706ns
2	Proposed Compressor based Multiplier	137	77	822	30.528ns	14.179ns	16.349ns

1)AREA COMPARISON



2)DELAY COMPARISON



VI. CONCLUSION

All approximate multipliers are designed for n = 8. The multipliers are implemented in Verilog and synthesized using This paper deals with the analysis and design of two new approximate 4-2 compressors for utilization in a multiplier. we will design a Efficient Array Multiplier using our proposed Multiplier. The proposed approximate compressors are proposed and analyzed for a Array multiplier. This Proposed Multiplier is used in sobel operator design. Sobel operator executed in matlab and modals software. The Proposed method is implemented using Verilog HDL and Simulated and Synthesised by modelsim and Xilinx tools. The Design was analysed by Xilinx Tool.

REFERENCES

[1] J. Liang, J. Han, and F. Lombardi, "New metrics for the reliability of approximate and Probabilistic Adders," IEEE Trans. Computers, vol. 63, no. 9, pp. 1760–1771, Sep. 2013.

[2] V. Gupta, D. Mohapatra, S. P. Park, A. Raghunathan, and K. Roy, "IMPACT: IMPrecise adders for low-power approximate computing," in Proc. Int. Symp. Low Power Electron. Design, Aug. 2011, pp. 409–414.

[3] S. Cheemalavagu, P. Korkmaz, K. V. Palem, B. E. S. Akgul, and L. N. Chakrapani, "A probabilistic CMOS switch and its realization by exploiting noise," presented at the IFIP Int. Conf. Very Large Scale Integ., Perth, Australia, Oct. 2005.

[4] H. R. Mahdiani, A. Ahmadi, S. M. Fakhraie, and C. Lucas, "Bioinspired imprecise computational blocks for efficient VLSI implementation of soft-computing applications," IEEE Trans. Circuits Syst. I: Reg. Papers, vol. 57, no. 4, pp. 850–862, Apr. 2010.

[5] M. J. Schulte and E. E. Swartzlander Jr., "Truncated multiplication with correction constant," in Proc. Workshop VLSI Signal Process. VI, 1993, pp. 388–396.

[6] E. J. King and E. E. Swartzlander Jr., "Data dependent truncated scheme for parallel multiplication," in Proc. 31st Asilomar Conf. Signals, Circuits Syst., 1998, pp. 1178–1182.

[7] P. Kulkarni, P. Gupta, and M. D. Ercegovac, "Trading accuracy for power in a multiplier architecture," J. Low Power Electron., vol. 7, no. 4, pp. 490–501, 2011.

[8] C. Chang, J. Gu, and M. Zhang, "Ultra low-voltage low-power CMOS 4-2 and 5-2 compressors for fast arithmetic circuits," IEEE Trans. Circuits Syst., vol. 51, no. 10, pp. 1985–1997, Oct. 2004.

[9] D. Radhakrishnan and A. P. Preethy, "Low-Power CMOS pass logic 4-2 compressor for high-speed multiplication," in Proc. IEEE 43rd Midwest Symp. Circuits Syst., 2000, vol. 3, pp. 1296–1298.

[10] Z. Wang, G. A. Jullien, and W. C. Miller, "A new design technique for column compression multipliers," IEEE Trans. Comput., vol. 44, no. 8, pp. 962–970, Aug. 1995.

[11] J. Gu and C. H. Chang, "Ultra low-voltage, low-power 4-2 compressor for high speed multiplications," in Proc. 36th IEEE Int. Symp. Circuits Syst., Bangkok, Thailand, May 2003,pp. v-321–v-324.

[12] M. Margala and N. G. Durdle, "Low-power low-voltage 4-2 compressors for VLSI Applications," in Proc. IEEE Alessandro Volta Memorial Workshop Low-Power Design, 1999, pp. 84–90.

[13] B. Parhami, Computer Arithmetic; Algorithms and Hardware Designs, 2nd ed. London, U.K.: Oxford Univ. Press, 2010.

[14] K. Prasad and K. K. Parhi, "Low-power 4-2 and 5-2 compressors," in Proc. 35th Asilomar Conf. Signals, Syst. Comput., 2001, vol. 1, pp. 129–133.

[15] M. D. Ercegovac and T. Lang, Digital Arithmetic. Amsterdam, The Netherlands: Elsevier, 2003.
[16] D. Baran, M. Aktan, and V. G. Oklobdzija, "Energy efficient implementation of parallel CMOS multipliers with improved compressors," in Proc. ACM/IEEE 16th Int. Symp. Low Power Electron. Design, 2010, pp. 147–152.
[17] D. Kelly, B. Phillips, and S. Al-Sarawi, "Approximate signed binary integer multipliers for arithmetic data value speculation," in Proc. Conf. Design Architect. Signal Image Process., 2009, pp. 97–104.
[18] J. Ma, K. Man, T. Krilavicius, S. Guan, and T. Jeong, "Implementation of high performance multipliers based on approximate compressor design," presented at the Int. Conf. Electrical and Control Technologies, Kaunas, Lithuania, 2011.