FOREX Publication
Open Access | Rapid and quality publishing

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 2 | Pages 435-442 | e-ISSN: 2347-470X

Exact Computing Multiplier Design using 5-to-3 Counters for Image Processing

Perumal B[1], Balamanikandan A[2*], Jayakumar S[3], Ashok Kumar N[4] and Saranya K[5]

[1]*Electrical and Electronics Engineering, Adhiyamaan College of Engineering, Hosur, India; perumalbalan7@gmail.com*
[2]*Electronics and Communication Engineering, Mohan Babu University (Erstwhile SreeVidyanikethan Engineering College), Tirupati, India; balamanieee83@gmail.com*
[3]*Electronics and Communication Engineering, Sri Sairam College of Engineering, Bengaluru, India; jayakmr1982@gmail.com*
[4]*Electronics and communication engineering, Mohan Babu University (Erstwhile SreeVidyanikethan Engineering College), Tirupati, India; ashoknoc@gmail.com*
[5]*Electrical and Electronics Engineering, Dr. Mahalingam College of Engineering and Technology, Pollachi, India; saranya@drmcet.ac.in*

***Correspondence:** Balamanikandan A; balamanieee83@gmail.com

ABSTRACT- This work presents a novel approach to improve the area and energy efficiency of 5:3 counter, a key element used in digital arithmetic. To provide an effective substitute for addition operations, mostly in the partial product reduction stage of larger multipliers, this study suggests a new 5:3 counter. The Input Shuffling Unit (ISU) is employed within the proposed 5:3 counter to minimize gate-level implementation and path delay during partial product reduction in 16-bit and larger multipliers, thereby enhancing area and energy efficiency. Consequently, there are 84% fewer choices of input-output combinations, thereby decreasing the circuit complexity with respect to area and energy usage. When compared to its existing counterparts, the suggested 5:3 compressor improves area utilization and energy usage by an average of 11%, 17%, and 17% in 8-, 16-, and 32-bit multipliers, respectively. The results of simulations demonstrate the superiority of our method over traditional designs, providing an increase in both area and energy efficiency. These results highlight the applicability and scalability of our method, which is appropriate for a variety of applications such as embedded systems and digital signal processing.

Keywords: 5:3 Counter, Multiplier design, Area and energy optimization, Input shuffling unit.

ARTICLE INFORMATION

Author(s): Perumal B, Balamanikandan A, Jayakumar S, Ashok Kumar N and Saranya K;

Received: 16/02/2024; **Accepted**: 23/04/2024; **Published**: 30/04/2024;
e-ISSN: 2347-470X;
Paper Id: IJEER 1602-19;
Citation: 10.37391/IJEER.120215
Webpage-link:
https://ijeer.forexjournal.co.in/archive/volume-12/ijeer-120215.html

Publisher's Note: FOREX Publication stays neutral with regard to Jurisdictional claims in Published maps and institutional affiliations.

1. INTRODUCTION

Adders and multipliers mostly contribute to the hardware circuits in signal and image processing applications due to the repeated usage of these blocks in arithmetic manipulations. Additionally, the blocks impose high impact based on the architecture level realization in digital circuits. The less area, high speed, low power design is a need of the hour. Numerous methods now in use that are documented in the literature focus on lowering these design parameters. 4:2 compressor and a 5:3 counter, employs the two distinct carry handling mechanisms that can be used to count the ones in any arithmetic circuit. In 4:2 compressor the Carry propagates to nth column and Cout propagates to $n+1^{th}$ column, where as in 5:3 counter Cout propagates to $n+1^{th}$column and Carry propagates to nth column. The paper proposes a new 5:3 counter that uses an Input Shuffling Unit (ISU) to rearrange the incoming inputs, resulting in less circuit complexity and energy consumption.

A thorough overview of related works on counter-based multipliers is provided in *section 2*. The details of the proposed 5:3 counter is provided in *section 3*. The description of improved 15:4 counter is provided in Section 4. The application of existing state of art and proposed counters in multiplier design is covered in *section 5*. The structural parameters are covered in the last *section 6*, precision analysis using image multiplication is covered in *section 7* and the conclusion is covered in *section 7*.

2. RELATED WORKS

Hemanth et al. [1] employed two 5:3 compressors, six full adders, and two half adders. Here, the 5:3 compressor uses an input re-ordering circuit at the input side to reduce the output combinations from 32 to 18 for a 16x16 multiplier design. It consists of fifteen inputs (from x0 to x14) and four outputs (o0, o1, o2, and o3) respectively. Despite having comparable functions, the counter and compressor handle carry and Cout differently [1]. A 5:3 counter used in hardware design efficiently compresses five input signals into three output signals. It operates by detecting common patterns in the input signals and generating compressed outputs based on these patterns.

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 2 | Pages 435-442 | e-ISSN: 2347-470X

Marimuthu et al. [2] employed seven AND gates, two OR gates, one XOR gate, one NOT gate, and three 4:1 MUX gates. The realization of the sum utilizes one XOR gate and one NOT gate, while carry employs six AND gates and two OR gates, and Cout employs four AND gates and one OR gate.

Asif and Kong [3] used the concept of the carry-lookahead adder, which utilizes propagate and generate signals to speed up addition. To obtain the value of the sum, one XOR gate and the output from the PG block (P0 and P1) are used. In the case of carry, three XOR gates, one AND gate, and the output from the PG block (P0, P1, H0, H1, H2, G0, and G1) are used, and in Cout, five AND gates, three OR gates, and the output from the PG block (H0, H1, H2, H3, G0, and G1) are used. Chowdhury et al. [4] used a 5:3 compressor with five XOR gates, two MUX gates, and one AND gate. The sum operation utilized three XOR gates, resulting in a more critical path delay and power consumption. Additionally, the usage of two MUX gates and one XOR gate in the case of carry, and two MUX gates in Cout, resulted in higher energy consumption. Among them, numerous literatures [5] – [18] use the compressor and counter based optimization strategies, which handle the carry and Cout differently. In[19] though the reordering is employed and compared with [1] the overall hardware cost is more.

3. PROPOSED 5:3 COUNTER

It can be noted from the existing state-of-the-art design [1] that carry generation has the largest path delay as it passes through multiple stages: stage 1 (reordering circuit), stage 2 (Invert AND logic), stage 3 (EXOR represented as R), stage 4 (MUX), and stage 5 (AND). This induces the largest delay during the overall implementation of the 5:3 counter. The suggested ISU architecture receives the incoming data combinations and shuffles them. The proposed design reduces the original 32 options to just 6 combinations. *Table 1* displays the 5:3 counter truth table with shuffling details proposed at the input end. With an ISU at the input side, the suggested 5:3 compressor reduces input data combinations by 81%, as shown in *table 2*. In comparison to the current state of designs, this lowers the gate count of the suggested design. The inputs are shuffled and processed based on *equation (2)* to *equation (8).*

Table 1. Truth table of Proposed 5:3 Counter

Input					Count Value	5:3 Counter			4:2 Compressor		
X0	X1	X2	X3	X4(Cin)		Cout	Carry	Sum	Cout	Carry	Sum
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	0	0	1	0	0	1
0	0	0	1	0	1	0	0	1	0	0	1
0	0	0	1	1	2	0	1	0	1	0	0
0	0	1	0	0	1	0	0	1	0	0	1
0	0	1	0	1	2	0	1	0	1	0	0
0	0	1	1	0	2	0	1	0	1	0	0
0	0	1	1	1	3	0	1	1	1	0	1
0	1	0	0	0	1	0	0	1	0	0	1
0	1	0	0	1	2	0	1	0	0	1	0
0	1	0	1	0	2	0	1	0	0	1	0
0	1	0	1	1	3	0	1	1	1	0	1
0	1	1	0	0	2	0	1	0	0	1	0
0	1	1	0	1	3	0	1	1	1	0	1
0	1	1	1	0	3	0	1	1	1	0	1
0	1	1	1	1	4	1	0	0	1	1	0
1	0	0	0	0	1	0	0	1	0	0	1
1	0	0	0	1	2	0	1	0	0	1	0
1	0	0	1	0	2	0	1	0	0	1	0
1	0	0	1	1	3	0	1	1	1	0	1
1	0	1	0	0	2	0	1	0	0	1	0
1	0	1	0	1	3	0	1	1	1	0	1
1	0	1	1	0	3	0	1	1	1	0	1
1	0	1	1	1	4	1	0	0	1	1	0
1	1	0	0	0	2	0	1	0	0	1	0
1	1	0	0	1	3	0	1	1	0	1	1
1	1	0	1	0	3	0	1	1	0	1	1
1	1	0	1	1	4	1	0	0	1	1	0
1	1	1	0	0	3	0	1	1	0	1	1
1	1	1	0	1	4	1	0	0	1	1	0
1	1	1	1	0	4	1	0	0	1	1	0
1	1	1	1	1	5	1	0	1	1	1	1

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 2 | Pages 435-442 | e-ISSN: 2347-470X

The 5:3 counter is specially designed to count the number of ones in incoming data. The inputs are received and shuffled into intermediate inputs P, Q, R, S, T, as shown in *equation 1*. They are obtained by optimizing the *equation 1*. The gate-level implementation in *figure 1* focuses on the minimum transistor count when implemented in standard cell realization. The novelty of the proposed method lies in shuffling and then performing Boolean operations to reduce the structural cost and overall path delay of the design. The shuffling and handling of inputs of the counter reduce the delay that occurs during carry handling, thus providing energy efficiency with the minimum path delay of only 2 AND ,3 OR and a NOT as shown in *figure 1*.

S=(YA'+WA)'

Q=XA+ZA'

P=AZ

T=A'W

R=X+A(BC+DE+(D+E) (B+C)) (1)

Table. 2 ISU and its occurrence in proposed 5:3 counter

Output of ISU						
P	**Q**	**R**	**S**	**T**	**Value**	**No of time occurred**
0	0	0	0	0	0	1
0	0	0	0	1	1	5
0	0	0	1	1	2	10
0	0	1	1	1	3	10
0	1	1	1	1	4	5
1	1	1	1	1	5	1

Table 2 shows the shuffled intermediate elements with their count value and time of occurrence in the 5 to 3 counters. The *equation (2)* to *equation (8)* is used to obtain the optimized design of 5:3 Counter is obtained as trade-off between logic and hardware minimization

Sum=A^B^C^D^E (2)

Cout=Z+AX (3)

Carry= (A'Y+W+ Cout')' (4)

W=B'C'D'E' (5)

Z=BCDE (6)

X=BC(D+E) +DE(B+C) (7)

Y= B'C'(D'+E') +D'E'(B'+C') (8)

The sum generation utilizes three XOR gates. The carry and Cout of the proposed 5:3 counter is shown in *figure 1*. The symbol "^" stands for XOR operation, "+" for OR, "." for AND, and "'" is used for the invert operation in *equations (1)* to *(8)*. *Equations (2)* to *(8)* are constructed with minimum utilization of gates, considering hardware efficiency in transistor-level implementation to optimize the area and path delay, as can be observed from *figure 1* by applying Boolean minimization techniques. Area, power, and delay reduction are crucial in IC design to optimize performance and cost. Smaller area leads to more compact chips; lower power consumption reduces energy usage and heat dissipation, while reduced delay improves speed. These factors collectively enhance efficiency, competitiveness, and reliability of integrated circuits. It can be noted from *table 3* the gate count and path delay of proposed design is minimum.

The sum generation utilizes three XOR gates. The carry and Cout of the proposed 5:3 counter are shown in *figure 1*. The symbol "^" stands for XOR operation, "+" for OR, "." for AND, and "'" is used for the invert operation in *equations (1)* to *(8)*. *Equations (2)* to *(8)* are constructed with minimum utilization of gates, considering hardware efficiency in transistor-level implementation to optimize the area and path delay, as can be observed from *figure 1* by applying Boolean minimization techniques. Area, power, and delay reduction are crucial in IC design to optimize performance and cost. Smaller area leads to more compact chips; lower power consumption reduces energy usage and heat dissipation, while reduced delay improves speed. These factors collectively enhance efficiency, competitiveness, and reliability of integrated circuits. It can be noted from *table 3* the gate count and path delay of proposed design is minimum.

Table 3. Gate Count and Path Delay of Existing and Proposed Design of 5 to3 Compressor

Design	Gate Count	Path Delay
Asif and Kong [2]	4-2input EXOR and 4 AND,6-2 input AND,4 -2input OR,4 -3 input OR	EXOR+AND+2OR
Hemanth et al [1]	6 -2 input AND,3 –2 input OR,1-2:1 MUX 3 -2 input EXOR,2 NOT	3AND+NOT+EXOR +MUX
Proposed Design	3-2 input EXOR,8 –2 input AND,4-2 input OR,2-NAND (Bubbled AND=NAND),1-input OR	3 AND +2 OR+1 NOT

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 2 | Pages 435-442 | e-ISSN: 2347-470X

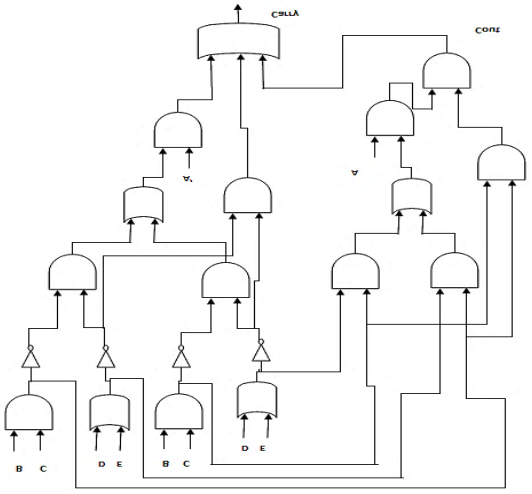


Figure 1. Carry and Cout of Proposed Design

4. PROPOSED 15:4 COUNTER

The proposed 15:4 counter, depicted in *figure2*, features inputs labelled X0 to X14 and outputs labelled O0 to O3. Despite appearing to have the same path delay as one 5:3 counter and two full adders, the delay incurred in the 5:3 counter is less, resulting in overall reduced path delay compared to the design by Hemanth et al.

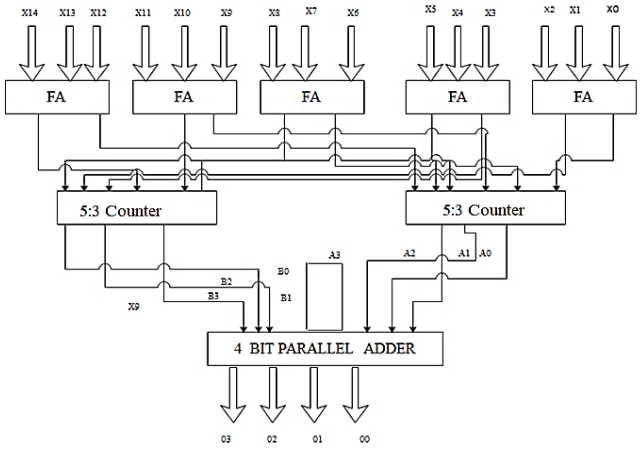


Figure 2. Proposed 15:4 Counter

5. PROPOSED 8-BIT BINARY MULTIPLIER

The handling of partial products is pivotal in digital circuit IC design and extensively researched. In the proposed approach 5:3 counter architecture employs minimum gate count tailored for efficient partial product reduction. In *figure 3*, depicting an 8x8 multiplier with three stages, seven 5:3 counters are strategically positioned in *stage 1*. Similarly, three in *stage 2*, optimizing for final output generation with minimum structural cost and delay. Cout and carry that propagate to nth and n=1 column incurs less delay. Technical analysis, as per *table 3*, demonstrates the proposed counter's superiority in gate count and path delay reduction, crucial metrics in digital circuit IC design for minimizing area and optimizing performance, particularly in the partial product reduction stage of the 8x8 design delineated in *figure 3*.

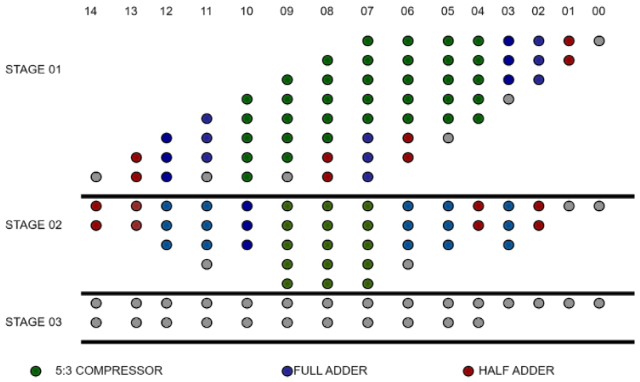


Figure 3. 8×8 multiplier design using proposed 5:3counters

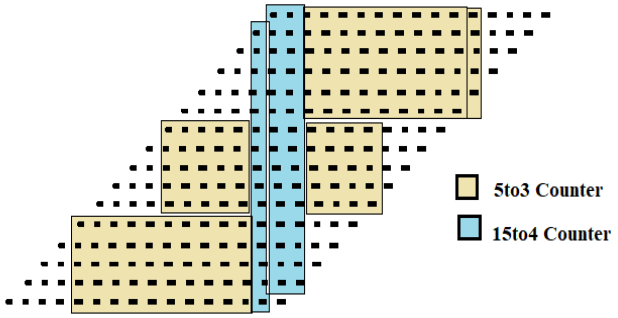


Figure 4. 16×16 multiplier design using proposed 5:3 and 15to4 counters

From *figure 4*, a direct implementation of a 15-to-4 counter is evident in the partial product implementation. Subsequent stages also underscore the pivotal role of 5-to-3 and 15-to-4 counters in the Partial Product Reduction Unit, significantly curbing overall energy consumption and area utilization. The existing techniques optimize area and speed in digital multiplication circuits by efficiently handling partial products through diverse algorithmic and architectural methodologies whereas these 5 to 3 and 15 to 4 counters provide an alternate solution for these applications.

Existing methods typically involve the direct accumulation of partial products using standard adder architectures, such as ripple carry adders or carry look-ahead adders. While these approaches are conceptually simple, they suffer from limitations in terms of speed and area efficiency, especially for large operand sizes. To address these challenges, the proposed designs have explored alternative architectures and optimization techniques of counter-based partial product reduction that employs 5 to 3 and 15 to 4 counters. These techniques aim to exploit parallelism, reduce critical path delays, and minimize resource utilization, thereby improving the overall performance of hardware multiplier designs. From *figure 4* it is evident that n-1 to $n+^{th}$ column employs 15 to 4 counter and nearly 75% of partial product reduction stage

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 2 | Pages 435-442 | e-ISSN: 2347-470X

except least n/4th and most n/4th column employ 5 to 3 counter. This hardware implementation enhances the parallel reduction of partial product when compared to the existing add and accumulate structures thereby reducing delay. As gate level optimization employed in proposed counter design the overall hardware implementation also simplifies and contributes to the area reduction also during implementation in partial product reduction stage.

6. PERFORMANCE AND RESULT

The proposed multipliers, along with state-of-the-art designs for comparison, are implemented using structural Verilog HDL code and synthesized utilizing Cadence Encounter with 180nm and 45nm technologies on an ASIC platform. This validation process aims to assess the scalability and efficacy of the proposed design at varying size and technology.

Table 4. Comparison results of proposed and existing 5:3 counters

5:3 Compressor	Area (μm²)	Power (nW)	Delay (ps)	Area*delay (am²s)	Power*delay (zJ)
[4]	15.6	2211.3	161.7	2522.52	357567.21
[2]	31.2	2548	179.3	5594.16	456856.4
[3]	39.6	3728.4	150.7	5967.72	561869.88
[1]	16.8	2219.1	136.4	2291.52	302685.24
Proposed Design	14.4	2152.8	135.7	1954.08	292134.9

Table 5. Comparison results of proposed and existing 15:4 counters

15:4 Compressor	Area (μm²)	Power (nW)	Delay (ps)	Area*delay (Am²s)	Power*delay (zJ)
[4]	75.6	13941.2	320.1	24199.56	4462578.12
[2]	106.8	14696.5	334.4	35713.92	4914509.6
[3]	123.6	17538.3	319	39428.4	5594717.7
[1]	76.8	13269.1	286	21964.8	3794962.6
Proposed Design	73.2	12868.7	280	20496	3603040

Note from the *table 6* the proposed 8-bit ISU counter-based multiplier architecture exhibits 2% area reduction;6% power reduction;6% delay reduction;9% ADP reduction;13% energy reductions compared to the best of the existing designs.

Table 6. Comparison results of proposed and existing 8-bit multiplier

Designs	Area (μm²)	Power (nW)	Delay (ps)	Area*delay (Am²s)	Power*delay (zJ)
[4]	3794.56	471033.6	2299.95	8727298.272	1083353728
[2]	4031.72	456094.08	2366.64	9541629.821	1079410493
[3]	5587.78	503673.6	2667.6	14905961.93	1343599695
[1]	6519.48	573916.8	2857.41	18628827.35	1639915603
Proposed Design	6354.92	533928.96	2659.24	16899257.46	1419845248

The implementation of 16 bit and 32-bit multiplier employing the 15:4 counter is performed to check the validness of proposed design. *Table 7* and *8* compares the power, area, delay, Power Delay Product (PDP) and Area Delay Product (ADP) for 16- and 32-bit multipliers.

Table 7. Comparison results of proposed and existing 16-bit multiplier

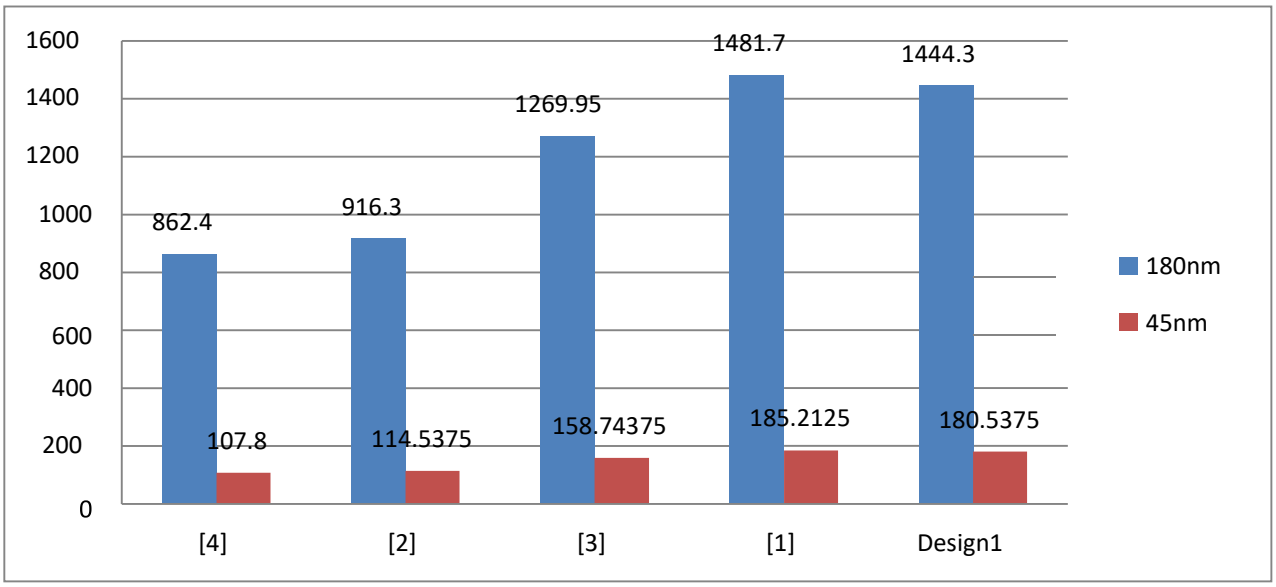
Designs	Area (μm²)	Power (nW)	Delay(ps)	Area*delay(am²s)	Power*dela y(zJ)
[4]	1897.28	235516.8	1149.975	2181824.568	270838432.1
[2]	2015.86	228047.04	1183.32	2385407.455	269852623.4
[3]	2793.89	251836.8	1333.8	3726490.482	335899923.8
[1]	3259.74	286958.4	1428.705	4657206.837	409978900.9
Proposed Design	3177.46	266964.48	1329.62	4224814.365	354961311.9

Table 8. Comparison results of proposed and existing 32-bit multiplier

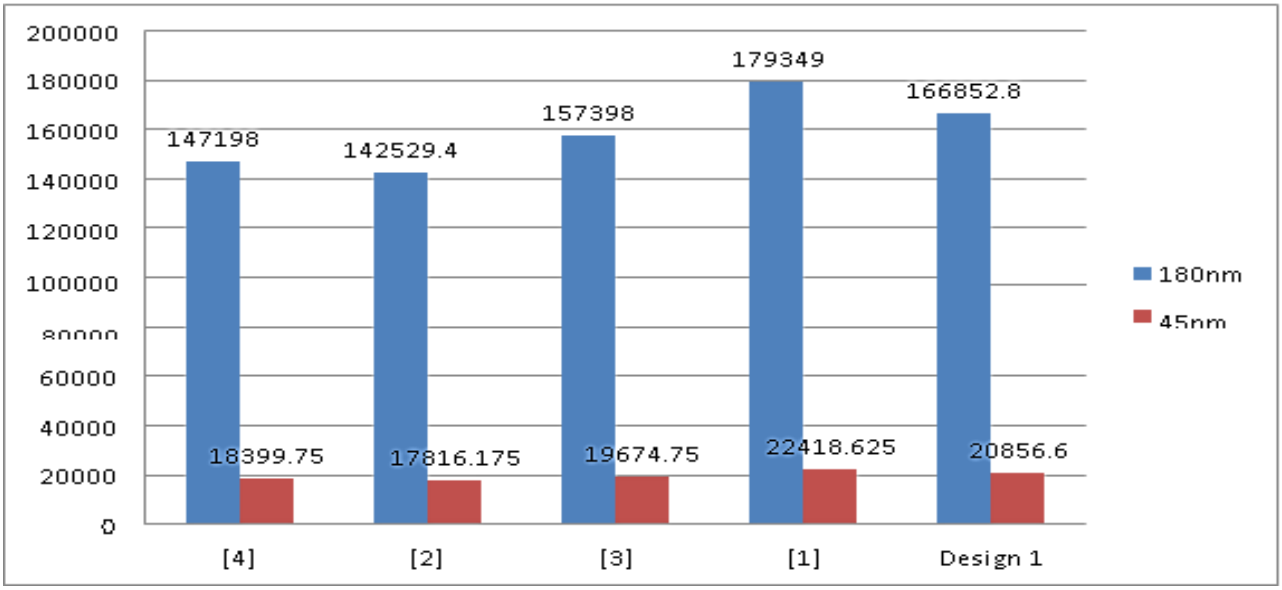
Designs	Area (μm²)	Power (nW)	Delay(ps)	Area*delay (Am²s)	Power*dela y(zJ)
[4]	3794.56	471033.6	2299.95	8727298.272	1083353728
[2]	4031.72	456094.08	2366.64	9541629.821	1079410493
[3]	5587.78	503673.6	2667.6	14905961.93	1343599695
[1]	6519.48	573916.8	2857.41	18628827.35	1639915603
Proposed Design	6354.92	533928.96	2659.24	16899257.46	1419845248

The area, power, and average latency in two distinct PDK technologies, 180nm, and 45nm, are measured and the results are shown in *figure 5* to assess the appropriateness of the ISU counter-based multiplier architectural design for ASIC design.

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 2 | Pages 435-442 | e-ISSN: 2347-470X



(a)



(b)

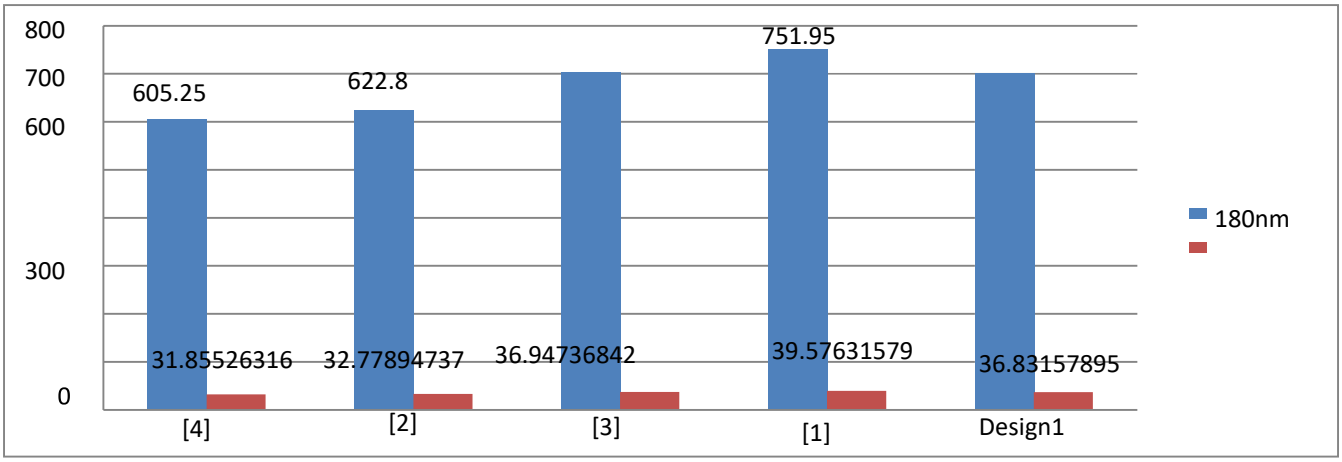


Figure 5- Comparison of 8 X 8 -ISU Counter Based Multiplier against prior designs in 180nm and 45nm ASIC technology (a)Area (b)Processing delay (c) Power-delay product.

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 2 | Pages 435-442 | e-ISSN: 2347-470X

7. PRECISION ANALYSIS: IMAGE MULTIPLICATION

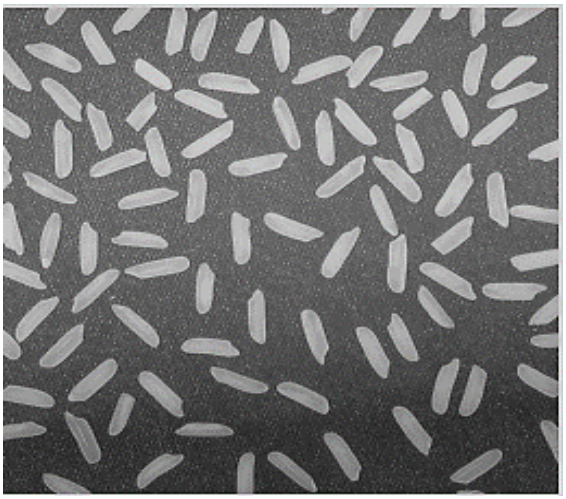
In this section, the application of the proposed exact multipliers to image processing is implemented to demonstrate the effectiveness of the design. The fusion of two images into a single output image is achieved using an image multiplier, which operates on a pixel-by-pixel basis by multiplying corresponding pixels from each input image. *Figure 6* provides visual representations of the input images and the resulting multiplied output image. To facilitate this process; an 8x8 exact multiplier Verilog program was developed and simulated using MATLAB System Generator. The simulation results in the multiplied output image, showcase the application and functionality of the proposed design in image processing tasks.

$$Q(i,j) = P_1(i,j) * P_2(i,j) \quad (9)$$

where $\mathbf{P_1(i,j)}$, $\mathbf{P_2(i,j)}$ in equation represents the pixel values of the input image 1 and input image 2.

Input Image 1. cameraman

Input Image 2. rice.png



Output Image

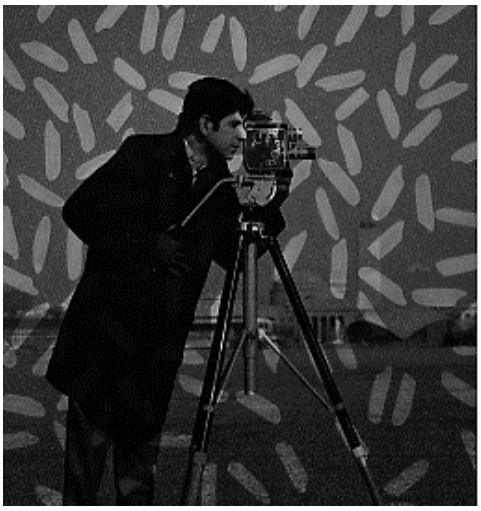


Figure 6. Implementation of f 8 X 8 -ISU Counter Based Multiplier in Image Multiplication

For a Virtex 7 FPGA, the resource utilization and frequency performance of proposed 8-bit multiplier is shown in *table 8.*

Table 8. Resource utilization and frequency performance of proposed 8-bit multiplier

Target Family	Device	SG and P	LUT	Flip flop	Slices	Frequency (MHz)
Virtex 7	XC7VX 690T	-3 and FFG1157	216	227	51	115.96

8. CONCLUSION

The new 5:3 counter with only 6 combinations of input after shuffling exhibits low hardware utilization. The implementation of new 5:3 counter in 15:4 counter and multiplier designs showcase the efficiency of the new design in area and energy optimization. To prove the effectiveness of the design the 16- and 32-bit multiplier implementations are performed and found to be effective in hardware metrics. It can be observed that proposed design exhibits 9% and 13 % reduction in ADP and PDP in 32-bit multiplier thus proving its effectiveness compared to the existing state of art designs. The proposed 5:3 counter, which utilizes an Input Shuffling Unit (ISU), significantly reduces input and output combinations by 84%. This reduction leads to less circuit complexity and lower energy consumption. The implementation of the proposed 5:3 counter in 8-, 16-, and 32-bit multiplier designs demonstrates its superiority over existing counterparts. It achieves better area and energy efficiency metrics. The scalability of the proposed method across different Process Design Kits (PDK) technologies is also evaluated, and observe that in 45nm PDK technology, as compared to 180nm PDK, the area, processing delay, and energy dissipation scale down by 8x, 8x, and 16x, respectively.

FOREX Publication
Open Access | Rapid and quality publishing

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 2 | Pages 435-442 | e-ISSN: 2347-470X

REFERENCES

[1] H. K. L., N. M., V. Janjirala, S. Veeramachaneni, and N. Mahammad S, "Efficient design of 15:4 counter using a novel 5:3 counter for high-speed multiplication," IET Computers & Digital Techniques, vol. 15, no. 1, pp. 12–19, Dec. 2020, doi: 10.1049/cdt2.12002.

[2] Marimuthu, R., et al.: Design of high speed and low power 15-4 compressor. In: International Conference on Communication and Signal Processing, pp. 533–536 (2013).

[3] Asif, S., Kong, Y.: Design of an algorithmic Wallace multiplier using high speed counters. In: Tenth International Conference on Computer Engineering Systems (ICCES), pp. 133–138 (2015).

[4] Chowdhury, S.R., et al.: Design, simulation, and testing of a high-speed low power 15-4 compressor for high-speed multiplication applications. In: First International Conference on Emerging Trends in Engineering and Technology, pp. 434–438 (2008).

[5] A. Balamanikandan and K. Krishnamoorthi, "Low area ASIC implementation of LUT–CLA–QTL architecture for cryptography applications," Wireless Networks, vol. 26, no. 4, pp. 2681–2693, May 2019, doi: 10.1007/s11276-019-02017-3.

[6] Vishnupriya. A and Sudarmani. R, "Efficient Serial Multiplier Design using Ripple Counters, Kogge-Stone Adder and Full Adder," International Journal of Computer Applications, vol. 67, no. 6, pp. 33–38, Apr. 2013, doi: 10.5120/11401-6717.

[7] S. S Bhairannawar, R. R, R. K.B, V. K.R, and P. L.M, "FPGA Based Efficient Multiplier for Image Processing Applications Using Recursive Error Free Mitchell Log Multiplier and KOM Architecture," International Journal of VLSI Design & Communication Systems, vol. 5, no. 3, pp. 93–114, Jun. 2014, doi: 10.5121/vlsic.2014.5309.

[8] P. Sharma, A. K. Dubey, and A. Goyal, "Efficient Computing in Image Processing and DSPs with ASIP Based Multiplier," Recent Patents on Engineering, vol. 13, no. 2, pp. 174–180, May 2019, doi: 10.2174/1872212112666180810150357.

[9] Veeramachaneni, S., Lingamneni, A., Krishna, M.K., Srinivas, M.B.: Novel Archi- textures for efficient (m,n) parallel counters'. In: Proceedings of the 17th ACM Great Lakes Symposium on VLSI. GLSVLSI '07, New York, NY: ACM, pp. 188–191 (2007)

[10] Gu, J., Chang, C.H.: Ultra-low voltage, low power 4-2 compressor for high-speed multiplications. In: Proceedings of the 2003 International Symposium on Circuits and Systems. ISCAS '03, vol. 5, pp. V–V (2003)

[11] Hsiao, S.F., Jiang, M.R., Yeh, J.S.: Design of high-speed low-power 3-2 counter and 4-2 compressor for fast multipliers, Electron. Lett. 34(4), 341–343 (1998)

[12] S. Umadevi and T. Vigneswaran, "Reliability improved, high performance FIR filter design using new computation sharing multiplier: suitable for signal processing applications," Cluster Computing, vol. 22, no. S6, pp. 13669–13681, Feb. 2018, doi: 10.1007/s10586-018-2067-5.

[13] Kwon, O., Nowka, K., Swartzlander, E.E.: A 16-bit by 16-bit mac design using fast 5: 3 compressor cells, J. VLSI Sign. Proc. Syst. Sign. Image Video Technol. 31(2), 77–89 (2002)

[14] P. Anguraj and T. Krishnan, "Design and realization of area-efficient approximate multiplier structures for image processing applications," Microprocessors and Microsystems, vol. 102, p. 104925, Oct. 2023, doi: 10.1016/j.micpro.2023.104925.

[15] M. Maria Dominic Savio, T. Deepa, P. Divya Dharshini, K. Sonali, and O. Singh, "Design of High-speed Multiplier using Input Scrambled 5-3 compressor for Error Tolerant image processing," Journal of Physics: Conference Series, vol. 2335, no. 1, p. 012044, Sep. 2022, doi: 10.1088/1742-6596/2335/1/012044.

[16] T. Thota, N. Chaganti, P. R. Nedhunuri, and J. V. R. Ravindra, "LI-PAMB: Low Complexity Implementation of Power and Area-efficient Modified Baugh Wooley Multiplier Using Exact Computing," International journal of simulation: systems, science & technology, Jan. 2019, Published, doi: 10.5013/ijssst.a.20.06.09.

[17] H. Song and P. Qiu, "Three-dimensional image registration using distributed parallel computing," IET Image Processing, vol. 12, no. 10, pp. 1713–1720, Oct. 2018, doi: 10.1049/iet-ipr.2017.1021.

[18] C.-C. Wang and G.-N. Sung, "Low-Power Multiplier Design Using a Bypassing Technique," Journal of Signal Processing Systems, vol. 57, no. 3, pp. 331–338, Dec. 2008, doi: 10.1007/s11265-008-0319-y.

[19] Snekha. S ,Naveenaasre. S ,Guru Prasath. K ,K. Saranya , " Design of 5 to 3 Compressor for High Speed Energy Efficient Arithmetic Circuits ", International Journal of Electrical, Electronics and Data Communication (IJEEDC) , pp. 15-18, Volume-10,Issue-6, (2022).