Fourth International Conference on Electronics, Communication and Aerospace Technology (ICECA-2020)
IEEE Xplore Part Number: CFP20J88-ART; ISBN: 978-1-7281-6387-1

2x2 Matrix Multiplication with 4-Bit elements in 45nm CMOS Technology

Chiranjit R Patel*, Vivek Urankar*, Vivek B A*, Dr. Sampath Kumar R†
*Electronics and Communiction,
RNS Instituite of Technology, Bengaluru, 560098, Karnataka, India
†Mathematics,
RNS Institute of Technology, Bengaluru, 560098, Karnataka, India

Email: chiranjitpatel08@gmail.com (Chiranjit R Patel)
Email: vivekurankar@gmail.com (Vivek Urankar)
Email: vivek.adishesha@gmail.com (Vivek B A)
Email: r.sampathkumar1967@gmail.com (Dr. Sampath Kumar R)

***Abstract* - A simple and effective method for matrices multiplication is proposed. The determination of the resultant output matrix can either performed in parallel or sequentially, both resulting in the same output. The selection of the algorithm is application-specific and filters down to the frequency of operation or power consumption. Multipliers and adders are the main hardware blocks in a matrix multiplication system. Hence, choosing a multiplier which is not only fast but also consumes lesser power and area is vital. This study also proposes a modified 2-bit and 4-bit multiplier architectures based on Vedic mathematics, which is proven to be more efficient than the standard architectures. The simulations are performed using Cadence Virtuoso 45nm CMOS technology. The proposed 7T SRAM cell makes use separate write and reads ports which are controlled by Read Word Line (RWL) and Write Word Line (WWL). Spectre simulations are performed for voltage ranges from 0.8V to 1.5V. The simulations also show that the matrix multiplier of 2-bit and 4-bit elements can operate at 2GHz and 0.7GHz at 1.2V respectively, and consumes an average of 140µW and 350µW. The layouts designed in 45nm are found to be matching with the schematic (LVS clean) and follows all Foundry rules (DRC clean).**

***Index Terms*—CMOS, Matrix multiplier, SRAM, Vedic - multiplier, Urdhva Tiryagbhyam**

I. INTRODUCTION

Matrix Multiplication is a binary operation that operates on two matrices to produce the output matrix. The rate at which technology is progressing, the demand for complex circuits to be faster, area-efficient, and low power has exponentially increased [15]. The reduction of dynamic power dissipation is seen as the main challenge. Power can be saved by various techniques [1, 2]. Images are represented in the form of matrices whose elements are binary numbers. The elements represent each pixel. The transformation of 3D rendering geometry and frequently makes use of matrix multiplications. Operations are performed on these matrices for image scaling, controlling contrast, saturation, and hue. In cryptography, matrices are used for encrypting messages and coding. A popular example is the hill cipher which utilizes matrix multiplication of encryption and decryption of data. Various multiplication algorithms [5] and optimization techniques are shown in the study of [13, 14].

Many security systems are designed to work with matrices. Like this matrix multiplication extends its application in various domains. The efficiency of matrix multiplication depends on the multiplier and adder used.
Choosing a multiplier which is not only fast but also consumes lesser power and area is very important in the overall performance of the design. The frequency of multiplying the matrices proceeds down to the multiplier and adder used. This paper also proposes a modified binary multiplier based on Urdhva Tiryagbhyam Sutra. The Urdhva Tiryagbhyam is a Vedic Sutra which is proven to deliver accurate results and is comparatively faster when compared to the conventional multiplication algorithms.

II. LITERATURE SURVEY

In today's world of modernization, the demands for low power circuits have increased. The works of [1] and [2] deal with techniques by which power consumption can be reduced. In [3], the authors in this paper focus on circuit optimization and design automation techniques to bring leakage current under control. The constraints in determining the sizes of transistors in the bit cells are read stability and write ability [12] are of major concern in nanometer CMOS technologies, as studied by the authors in [7]. In the works of [8], the 7T bit cell offers a reduction in reading power. The SRAM bit cell must be robust [9] and have good read stability and write ability [10]. In [14] the authors show how to exploit the standard multiplication algorithms in an improved way. In [16], the authors propose a Vedic multiplier based on Urdhva Tiryagbhyam sutra, the modified Carry Save Adders were also displayed as part of their study. With further development to the multiplier based on Urdhva Tiryagbhaym sutra, the authors in [17] propose an inverted architecture which was proven more effective in terms of power consumption and frequency of operation.

The findings of the literature survey have been tabulated in Table I and Table II.

TABLE I

Reference [8]	6T bit cell	7T bit cell
Technology node	32nm	32nm
Area (nm^2)	9216	10240
Read power (1V)	95.99 nW	32.98 nW
Read power (0.9V)	79.07 nW	22.64 nW

TABLE II
COMPARATIVE STUDY OF STRASSEN'S MATRIX MULTIPLICATION ALGORITHM [18]

Matrix dimensions	Recursion level	Conventional matrix multiplication	Strassen algorithm
16	2	1 ms	6 ms
32	2	1 ms	61 ms
64	2	2 ms	196 ms
128	2	9 ms	983 ms
512	8	1593 ms	2490 ms
512	16	1569 ms	1241 ms
1024	32	15182 ms	6222 ms

III. DESIGN METHODOLOGY

Matrix multiplication operates on a pair of matrices and produces one output matrix. If matrix A has 'p' rows and 'q' columns and matrix B has 'r' rows and 's' columns, the resultant matrix will have 'p' rows and 's' columns. This is only possible if 'q' is equal to 'r'.

Fig.1 shows the general architecture of the matrix multiplier. Designing high-speed low power multipliers and adders was the main challenge because the overall efficiency of the system depends on these blocks.

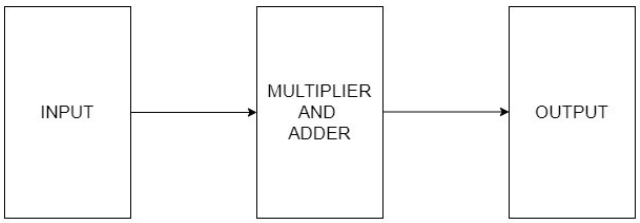

Fig. 1. General Matrix multiplication architecture

One of the most widely accepted procedure for multiplying matrices is to multiply the elements of a given row in the first matrix with the elements of a given column in the second matrix and to sum up all the partial products, which forms the resultant matrix. This procedure is repeated to find all the elements of the resultant matrix.

The main blocks comprise multipliers and adders to perform the binary matrix multiplication. The two input matrices that are to be multiplied are stored in registers sequentially. The resultant output is stored in an SRAM array [4], discussed in the further sections.

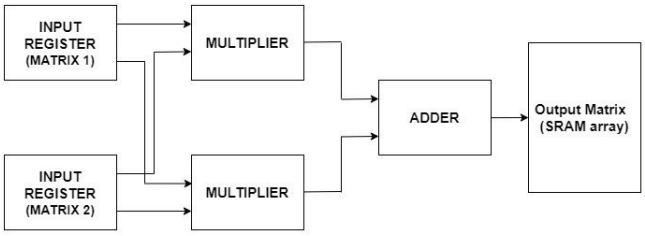

Fig. 2. Matrix multiplier Block diagram

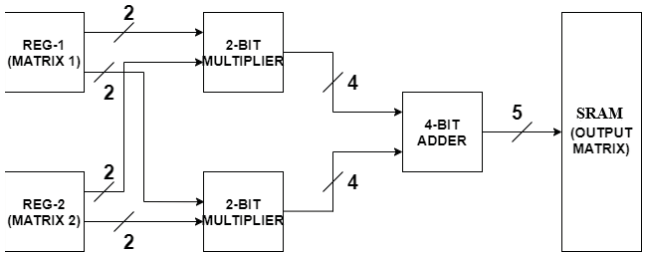

Fig. 3. 2x2 matrix multiplier (2-bit element)

Figure 3 shows the methodology for the multiplication of two matrices, each of which is two-bit wide. The product is obtained by a 2-Bit multiplier and summed with a 4-Bit adder. Each element of the output matrix is five bits wide. The output matrix is stored is a specific dimension SRAM array [11]. Data stored in the SRAM can be retrieved later for further computations.

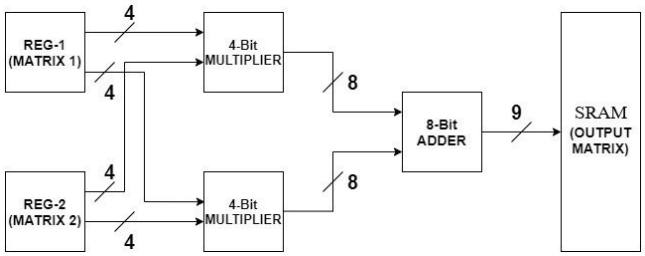

Fig. 4. 2x2 matrix multiplier (4-bit element)

Further, the resolution of the input matrix is increased to four bits. This requires a 4-Bit multiplier and an 8-Bit adder.

IV. SRAM

Each cell in the SRAM array mainly consists of a pair of cross-coupled inverters that retains the data as long as power is applied. This positive feedback eliminates any disturbance caused either due to the noise of leakage currents. A pair of access transistors (M5 and M6) is connected to the feedback as shown in Figure 5. The cell is written by driving the desired data onto the bit line (BL) and bit line bar (BLB), then asserting the word line (WL) high, which forces the cell to be written with the desired data. Separating the read and write access transistors, eliminates the pre-charging of the bit-lines

and its complement and requirement of additional circuitry like sense amplifiers during a read operation, which simplifies the overall circuit. This greatly worked in our favor in reducing the overall power consumption by the SRAM block.

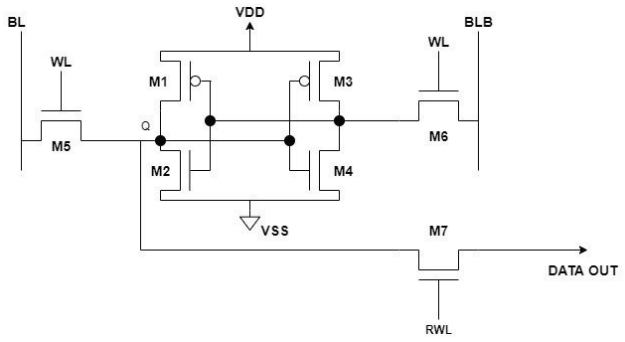

Fig. 5. Schematic of 7T SRAM bit cell

When the word lines (WL) are not active, the bit cell is in idle state and the two back-to-back inverters will reinforce each other via positive feedback long as there is a supply voltage.

SRAM requires a minimum supply voltage (VDD) to operate because the static noise margin depends upon VDD. The row decoder is implemented which drives four-word lines. In the SRAM array, each row has a separate word line. The row decoder ensures that at any instant of time, only one row is activated during write. The word lines are heavily loaded and in larger arrays this leads to high RC delay, hence the decoder is designed with high drive strength with the help of buffers.

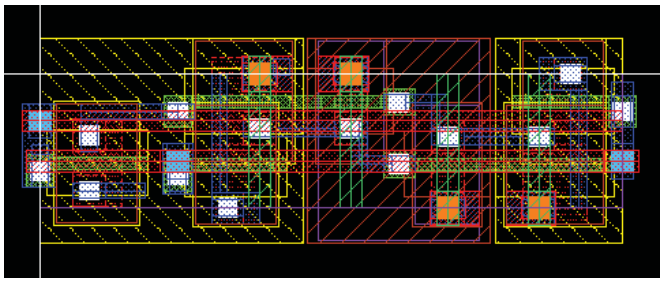

Fig. 6. Layout of 7T SRAM bit cell

The SRAM cells must be as compact as possible to achieve higher density [6]. The layout of the bit cell is designed such that it occupies the minimal area. The design of 7T is created by using 3T (half cell) in an anti-mirrored fashion to ensure symmetry, along with one read access transistor M7, as shown in Figure 6.

Figure 7 shows the schematic of the SRAM array, each individual block refers to a single bit cell. To minimize the power consumed by the SRAM array, techniques like power gating are used [3]. The SRAM array is constructed by mirroring and overlapping the cells such that the adjacent VDD and VSS rails are shared along the pr boundaries of the cell.

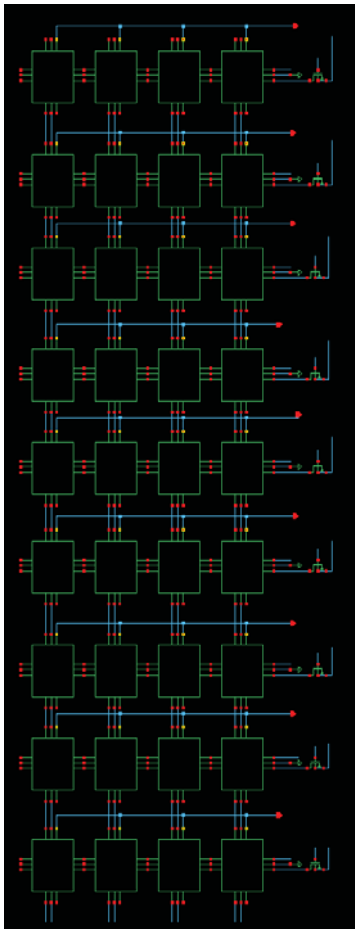

Fig. 7. Schematic of SRAM array

To reduce the resistance the power rails are made thicker than the minimum path sizing.

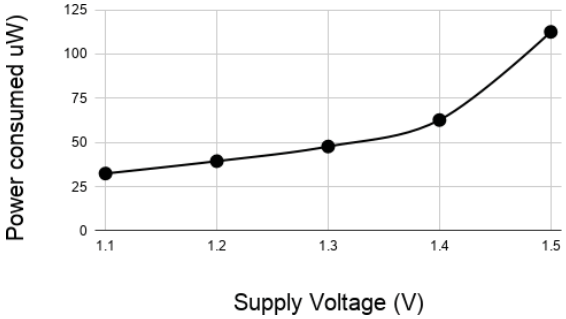

Fig. 8. Effect of different voltage ranges on power consumed by SRAM array

The graph depicted in Figure 8 shows the consumption of power by the proposed SRAM array for different operating voltages. There is an exponential rise in power consumption after 1.35V. Various techniques to reduced power are discussed in the works of [2, 3].

The proposed SRAM array utilizes power gating, which activates the array after the first computation of multiplication and addition.

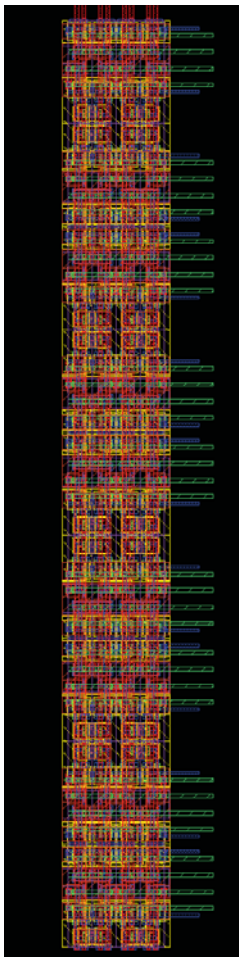

Fig. 9. Layout of SRAM array

TABLE III
AREA OCCUPIED BY BITCELL AND ARRAY

	Technology node	Area (μm^2)
7T Bitecll	45nm	1.085
SRAM array	45nm	39.06

TABLE IV
WRITE AND READ ACCESS TIME AT 1.2V

Write/read access	7T bitcell
Write 1	14.2 ps
Write 0	4.8 ps
Read 1	6.8 ps
Read 0	6.6 ps

Table V shows the calculated average power of an individual 7T bit cell and SRAM array, considering all the cases which include writing 1, writing 0, reading 1 and reading 0.

TABLE V
AVERAGE POWER CONSUMPTION FOR DIFFERENT SUPPLY VOLTAGES

Input voltage	7T bitcell	SRAM array
1.2 V	1.2 µW	39.42 µ W
1.3 V	1.48 µ W	47.7 µ W
1.4 V	2 µ W	62.7 µ W
1.5 V	3.57 µ W	112.6 µ W

V. DESIGN OF MULTIPLIER

The multipliers are designed not only to operate at a high frequency but at the same time to have power consumption at a minimum. The multipliers are based on Urdhva Tiyagbhyam Sutra which is proven an efficient way to multiplier binary numbers.

The 4-Bit multiplier instantiates the 2-Bit multiplier for a total of four times along with two specially modified Carry Save Adders. Figure 10 shows the modified 2x2 multiplier based on Urdhva Tiryagbhyam sutra.

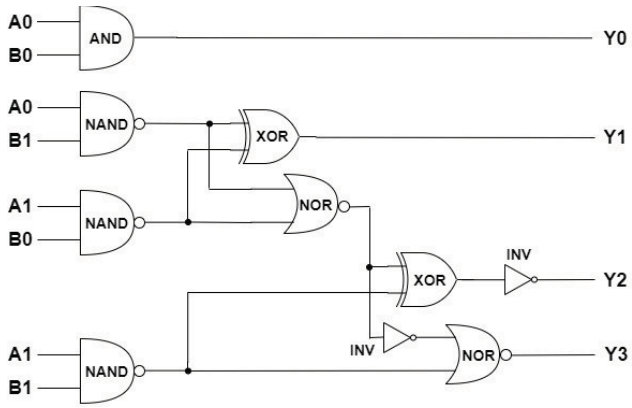

Fig. 10. Modified 2-bit Vedic multiplier

The standard 2-Bit multiplier is modified to obtain more superior speeds and lower power consumption. The use of universal gates is preferred, which also benefits the area consumed by the layout.

As shown in Figure 11 the blocks labeled as 1, 2, 3, and 4 are the four 2x2 multipliers. And the blocks 5 and 6 points refer to CSA-1 and CSA-2 respectively.

The outputs for the 2x2 multipliers are propagated to the CSA's. The two rows for the CSA's convert the outputs of the multipliers to a running sum. A couple of the least significant bits generated by CSA-1 contribute to the output of the 4x4 multiplier, remaining bits are fed to CSA-2. The outputs of CSA-2 are grouped to form the higher nibble of the output.

Figure 12 and Figure 13 show the layout design of the 2x2 and 4x4 bit multipliers. The area occupied by the 2-bit and 4-bit multipliers is 12.9 μm^2 and 135.4 μm^2 respectively. The markings 5 and 6 refer to CSA-1 and CSA-2. The authors have previously proposed the Carry Save Adder designs in the works of [16].

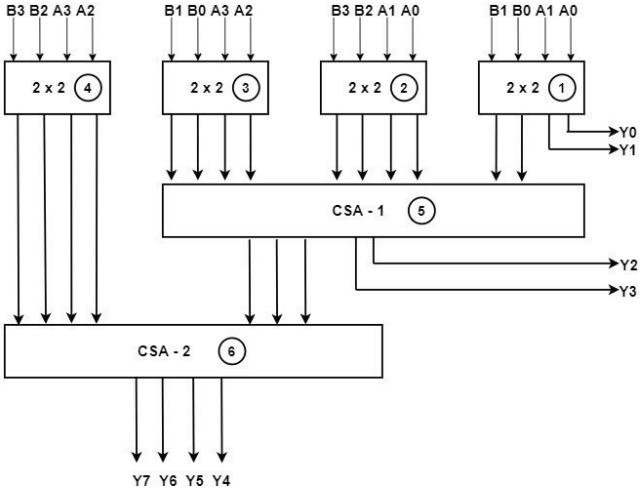

Fig. 11. Block diagram 4-bit Vedic multiplier

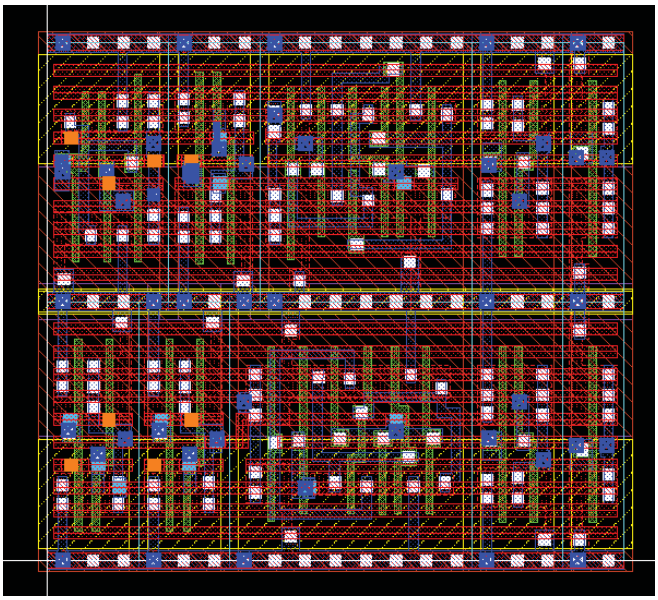

Fig. 12. Layout of 2-bit multiplier

The labels 1, 2, 3, and 4 refer to four 2-bit multipliers. And the labels 5 and 6 refer to CSA-1 and CSA-2 respectively, as depicted in Figure 11.

The performance of the standard Vedic 2x2 multiplier and the proposed inverted gate 2x2 multiplier are shown in Figure 14 and Figure 15. The color red refers to the standard design and green refers to the modified design. At 0.8V it is seen that the proposed design is 37% faster and saves 8% of the power. This proves the design can be used at low power applications and yet have an edge at speed.

Figure 16 shows a detailed analysis of the proposed 4x4 multiplier architecture. The simulations are performed for the voltage ranges of 0.8V to 1.5V using Spectre simulations. The propagation delay is obtained by considering the worst case.

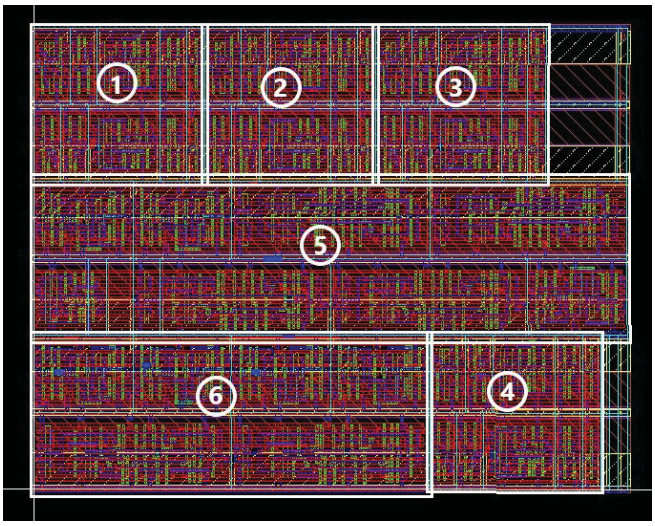

Fig. 13. Layout of 4-bit multiplier

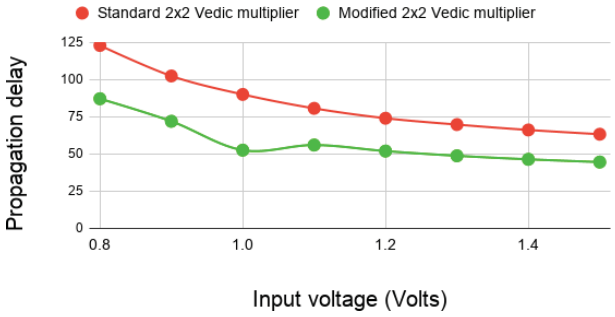

Fig. 14. Propagation delay comparison of 2x2 multiplier designs

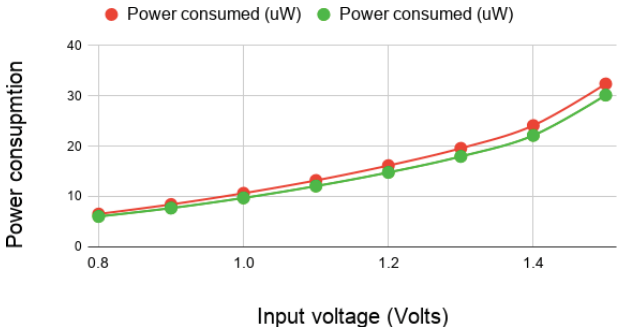

Fig. 15. Power consumption comparison of the 2x2 multiplier designs

TABLE VI

	2-bit multiplier [16]	Modified 2-bit multiplier [proposed]	4-bit multiplier [proposed]
Input voltage	1.1 V	1.1 V	1.1 V
Technology node	45nm	45nm	45nm
Propagation delay	60 ps	54 ps	399.4 ps
Area	15.08 μm^2	12.996 μm^2	135.432 μm^2

Depending upon the application of the design, the multiplier can be operated at any particular voltage.

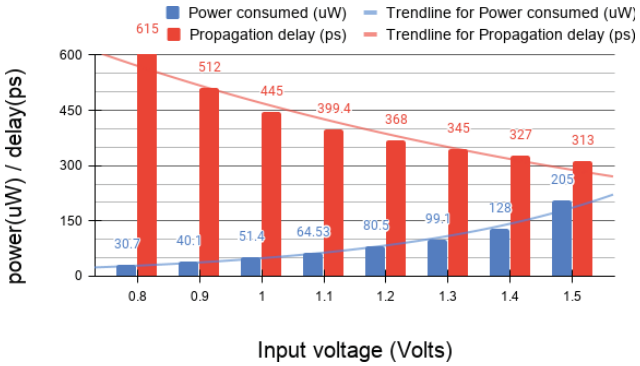

Fig. 16. Effect of various power supply levels on 4-bit multiplier

VI. DESIGN OF 2X2 MATRIX MULTIPLIER

The architecture of two variants of 2x2 matrix multiplier has been proposed. Each element of the first variant is 2-bit wide and each element of the second variant is 4-bit wide.

The verification of the matrix multiplier of both the 2- bit and 4-bit element variants are done using the Cadence Spectre simulator. The examples are considered for both the matrix multiplier variants. Example case for 2-bit elements is shown in Figure 17. The waveforms are depicted in Figure 18.

$$\begin{bmatrix} 11 & 00 \\ 11 & 00 \end{bmatrix} \times \begin{bmatrix} 11 & 00 \\ 11 & 00 \end{bmatrix} = \begin{bmatrix} 01001 & 00000 \\ 01001 & 00000 \end{bmatrix}$$

Fig. 17. Example of 2-bit element

Figure 19 shows example of 4-bit elements. The two matrices are given as input to the matrix multiplier block and the waveform is shown in Figure 20.

Figure 21 shows the layout design of a 2x2 matrix multiplier with 4-bit elements, the schematic shown in Figure 3.

The 2x2 matrix multiplier layout design is shown in Figure 22, the schematic shown in Figure 4.

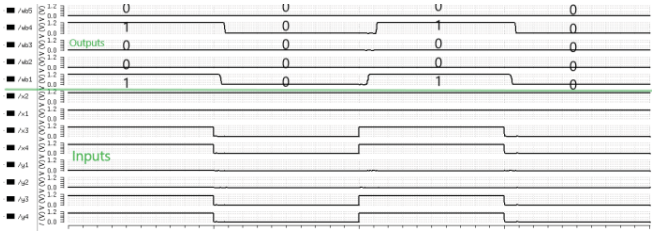

Fig. 18. 2-bit element simulated waveform

$$\begin{bmatrix} 1111 & 0000 \\ 1111 & 0000 \end{bmatrix} \times \begin{bmatrix} 1111 & 0000 \\ 1111 & 0000 \end{bmatrix} = \begin{bmatrix} 011100001 & 000000000 \\ 011100001 & 000000000 \end{bmatrix}$$

Fig. 19. Example of 4-bit element

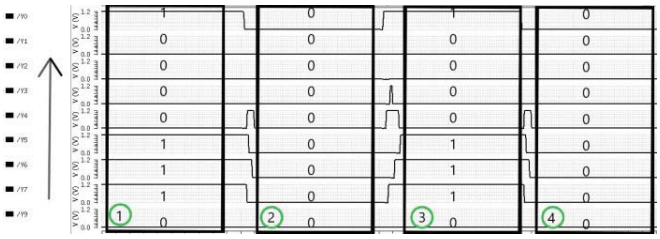

Fig. 20. 4-bit element simulated waveform

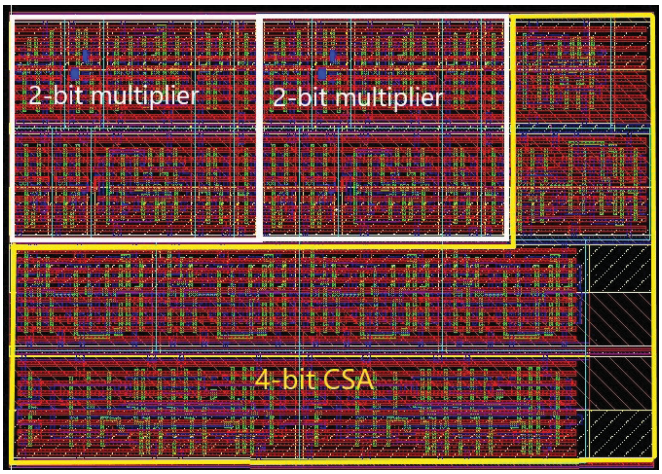

Fig. 21. Layout of 2x2 matrix multiplier (2-bit element)

TABLE VII
AREA OCCUPIED BY THE DIFFERENT BLOCKS DESIGNED IN 45NM

Designs	Area (μm^2)
2-bit multiplier	12.996
CSA-1	45.144
CSA-1	30.096
4-bit multiplier	135.432
2x2 matrix multiplier(2-bit element)	67.032
2x2 matrix multiplier(4-bit element)	406.296

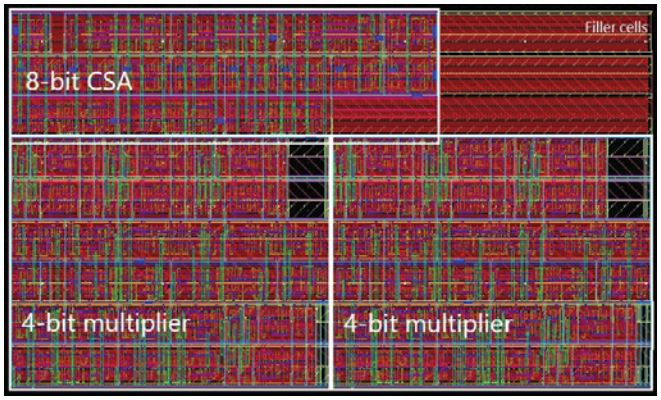

Fig. 22. Layout of 2x2 matrix multiplier (4-bit element)

Table VII summarizes the area occupied by the different blocks.

It is found that the matrix multiplication of 2-bit elements consumes an average of 140µ W at 1.2V and can operate at 2GHz. The same 2x2 matrix multiplication was performed with 4-bit resolution of the elements. Results show there is a 350µ W of power consumed on an average and can be operated at 0.7GHz at 1.2V.

TABLE VIII
PERFORMANCE ANALYSIS OF TWO VARIANTS OF 2X2 MATRIX MULTIPLIER

Analysis	supply voltage	Power	Operating frequency
2x2 (2-bit element	1.2 V	140 µ W	2 GHz
2x2 (4-bit element)	1.2 V	350µ W	0.7 GHz

Table VIII summarizes the results obtained by simulating the designs at 1.2V along with the average power consumed and a maximum frequency of operation.

VII. CONCLUSION

In this paper, a method to multiply binary 2x2 matrices is proposed. Modified 2-bit and 4-bit multipliers based on Urdhva Tiryagbhyam Sutra are also proposed. Simulations and layout designs of the various blocks are performed using Cadence 45nm CMOS technology. The designs are simulated for the voltage range of 0.8V to 1.5V. Simulation results also conclude that the proposed 2-bit element and 4-bit element matrix multiplier can operate at 2GHz and 0.7GHz respectively. The drawback of the SRAM design is, one V_{th} drop obtained during reading data of 1 stored in the SRAM.

The design and development of high-speed, effective matrix multiplier modules can be implemented in optical signal processors and various filters. The algorithm-based multipliers have high switching speeds and can be worked at high frequency. The resolution of the matrix elements can be increased by designing larger bit multipliers. Therefore, high-speed ASICs and DSPs are feasible to implement. Further, the size of the matrix can be increased, and various power reduction techniques can be incorporated into the design.

REFERENCES

[1] R. Sivakumar and D. Jothi, "Recent Trends in Low Power VLSI Design," *International Journal of Computer and Electrical Engineering*, 2014.
[2] A. P. Chandrakasan and R. W. Brodersen, "Minimizing Power Consumption in Digital CMOS Circuits," *Proceedings of the IEEE*, vol. 83, pp. 498–523, 1995.
[3] S. Patil and R. Dudhe, "Circuit Optimization and Design Automation Techniques for Low Power CMOS VLSI Design: A Review," 2015.
[4] J. E. R *et al.*, "A STUDY ON 6T SRAM AND 7T SRAM CELLS," *International Journal of Research in Engineering and Technology*, vol. 07, no. 02, 2018.
[5] K. Thabet and S. A. Ghuribi, "Matrix Multiplication Algorithms," *International Journal of Computer Science and Network Security*, vol. 12, 2012.
[6] G. Apostolidis *et al.*, "Design and Simulation of 6T SRAM Cell Architectures in 32nm Technology," *Journal of Engineering Science and Technology Review*, 2016.
[7] E. Grossar *et al.*, "Read Stability and Write-Ability Analysis of SRAM Cells for Nanometer Technologies," *IEEE Journal of Solid State Circuits*, vol. 41, 2006.
[8] A. Islam and M. Hassan, "Variability analysis of 6T and 7T SRAM cells in sub 45nm technology," *IIUM Engineering Journal*, vol. 12, 2011.
[9] K. Mehrabi *et al.*, "A Robust and Low Power 7T SRAM Cell Design," *CSI International Symposium on Computer Architecture and Digital Systems*, 2014.
[10] S. Yadav *et al.*, "Low Power SRAM Design with Reduced Read/Write Time," *International Journal of Information and Computation Technology*, vol. 3, pp. 195–200, 2013.
[11] S. Saun and H. Kumar, "Design and performance analysis of 6T SRAM cell on different CMOS technologies with stability characterization," in *IOP Conference Series: Materials Science and Engineering (ICMSMT)*, 2019.
[12] "Design of 64 bit SRAM using Lector Technique for Low Leakage Power with Read and Write Enable," *IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)*, vol. 7, no. 2, 2017.
[13] K. Matrouk *et al.*, "Analysis of Matrix Multiplication Computational Methods," *European Journal of Scientific Research*, vol. 121, 2014.
[14] C. E. Drevet *et al.*, "Optimization techniques for small matrix multiplication," *Theoretical Computer Science, Elsevier*, 2011.
[15] D. Harris and N. Weste, "CMOS VLSI Design : A circuits and system perspective."
[16] C. R. Patel *et al.*, "Vedic Multiplier in 45nm Technology," *2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 21–26, 2020.
[17] C. R. Patel *et al.*, "Inverted Gate Vedic Multiplier in 90nm CMOS Technology," *American Journal of Electrical and Computer Engineering*, pp. 10–15, 2020.
[18] J. Mathew and D. R. V. Kumar, "Comparative Study of Strassen's Matrix Multiplication Algorithm," *International Journal of Computer Science And Technology (IJCST)*, vol. 3, no. 1, pp. 749–754.