IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-6, June 2019 E-ISSN: 2347-2693

# Design and Simulation of Two – bit Multiplier Circuit using MGDI Technique

**Subhashis Maitra**

Dept. of Electronics and Communication, Kalyani Government Engineering College, Kalyani, Nadia, West Bengal, India

*Corresponding Author: subha_shis06@yahoo.co.in*

DOI: https://doi.org/10.26438/ijcse/v7i6.5661 | Available online at: www.ijcseonline.org

Accepted: 12/Jun/2019, Published: 30/Jun/2019

***Abstract*** **-** Multiplier in Digital Signal Processing (DSP) and Elliptic Curve Cryptography (ECC) are crucial. Thus modern DSP and ECC systems require to develop low power multiplier circuits to reduce the power dissipation and at the same time to increase the speed. One of the efficient ways to reduce power dissipation is by the use of Modified Gate Diffusion Input (MGDI) which at the same time reduces the circuit parameters like transistor count, implementation cost, space required and propagation delay. This paper proposes a new design technique for two-bit binary multiplier and hence multi-bit binary multiplier using the proposed two-bit multiplier circuit. This paper also implements the proposed two-bit multiplier using DSCH 3.5. The proposed technique claims lower power consumption, lower cost, lower space required and also lesser number of transistor than other conventional techniques like CMOS, PTL, CPL etc. A comparative study of the proposed technique has been dealt here clearly which shows the novelty of the proposed technique.

***Keywords*** **-** CPL, DSCH 3.5, Karnaugh' map, Multiplier, PTL, Shannon's Expansion Theorem.

## I. INTRODUCTION

Multiplication is an important fundamental function of arithmetic operations. Digital signal processing and elliptic curve cryptography involve lot of multiplication processes. Digital Signal Processors are playing the crucial role to handle the complexities of Digital signal processing. Arithmetic operations like addition, multiplication are the important tasks that must be considered in case of designing high speed processors. Digital signal processor takes care of convolution, correlation, and filtering of digital signal which require multiplicative operations. Similarly elliptic curve cryptography requires point multiplications. Since multiplication operations dominate the execution time of most of the signal processing algorithms, so design of high speed multipliers is the major design criteria in modern design techniques. The demand of high speed processing has been increasing as a result of expanding computer, signal processing and cryptographic applications. High speed arithmetic operations are important to achieve the desire performance in many real time signal and image processing and cryptographic applications [1]. One of the important arithmetic operations in these kinds of applications is high speed multiplication. Time delay and power consumption are the major issues for these kinds applications. Many techniques were proposed to reduce time delay and power consumption [2], [3], [4], [5].

Multipliers have three stages [6], like generation of partial products, reduction of partial products and addition. Reduction of partial products take much time and power in the multiplier circuit. Different approaches have been made in different literature [6],[7], to reduce the critical path in the multiplier. Among them, the use of compressors in the partial product reduction stage is the most popular [6]. R. Marimuthu et. al explained in [6] a multiplier circuit using 15-4 compressor. They proposed a 16x16 multiplier using the said compressor and explained the advantages of the proposed method. G. Ganesh Kumar et. al explained in [8] a high speed Vedic multiplier using Vedic mathematic technique. They implemented a 32x32 Vedic multiplier. The design was based on Vedic method of multiplication [9]. They claimed that the propagation delay of their proposed multiplier was 31.526 ns. Neha Goyal et. al proposed Booth multiplier in [10]. They claimed that Booth multiplier was superior in respect of area and complexity. They claimed that area of Booth multiplier is less than combinational multiplier since numbers of gate used in Booth multiplier is less. In [11], B.S. Premananda et. al proposed 8-bit Vedic multiplier. They implemented an 8-bit multiplier using four 4-bit Vedic multiplier and modified ripple carry adder. They claimed that their proposed architecture gave a total delay of 15.050 ns.

This paper introduces a new design technique for 2-bit multiplier and implements the same using MGDI cell. The proposed 2-bit multiplier is then used to implement a 4-bit multiplier. Section II of this paper presents the basic functions of GDI cell. Section III presents a two-bit

 

multiplier circuit using four AND gates and two half adders. Section IV introduces the proposed design and section V introduces the implementation of the proposed design using MGDI cell. Section VI presents the implementation of 4-bit multiplier using the proposed 2-bit multiplier. The last section, section VII concludes the paper.

## II. BASIC FUNCTION OF GDI CELL

The basic GDI (Gate Diffusion Input) cell as shown in Figure 1[12],[13], is like a CMOS inverter. Here the drain of a nMOS is connected to the drain of a pMOS. This common terminal is known as output terminal. The gate inputs for both the nMOS and pMOS are common and is termed as G. The source of pMOS is termed as P and the source of nMOS is termed as N. Source of pMOS is not connected to $V_{DD}$ and source of nMOS is not connected to ground like CMOS inverter. Bulks of both pMOS and nMOS are connected to P and N respectively so that it can be arbitrarily biased at contrast with a CMOS inverter.

Table 1 shows different logic functions for different inputs at P and N. For example, when P = B and N = 0, then if G = A, output will be $\bar{A}B$ which is denoted as function $F_1$. Similarly there is another function $F_2 = \bar{A} + B$, when N = B, P = 1 and G = A.

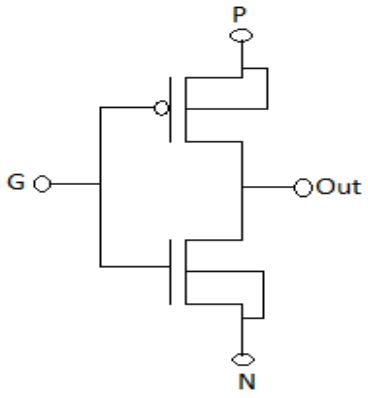


Figure 1. Diagram of basic GDI Cell.

Table 1. Various logic functions of GDI cell for different input configurations.

| N | P | G | Out | Function |
|---|---|---|---|---|
| 0 | B | A | $\bar{A}B$ | $F_1$ |
| B | 1 | A | $\bar{A} + B$ | $F_2$ |
| 1 | B | A | A + B | OR |
| B | 0 | A | AB | AND |
| C | B | A | $\bar{A}B + AC$ | MUX |
| 0 | 1 | A | $\bar{A}$ | NOT |

Figure 2 shows the diagram of modified GDI (MGDI) cell [12]. This is a modified version of GDI Cell. Its performance is better than Transmission Gate (TG), PTL (Pass Transistor Logic) and CMOS. In MGDI cell, the bulk of nMOS and pMOS are connected to ground and $V_{DD}$ respectively as shown in the Fig.

The performance analysis made with respect to power dissipation, switching delay and transistor count of MGDI, GDI, CMOS and TG and is presented in Table 2. From the table, it is clear that GDI Cell requires only two transistor for implementation of NOT, OR, AND and MUX gates. The worst case condition with respect to transistor count is CMOS MUX. It is also clear from the table that power dissipation of MUX is larger than any other gate, since the implementation of MUX is complicated. On the other hand NOT gate has the lowest power dissipation. With respect to switching delay, it is observed that GDI and TG provide the smallest switching delay.

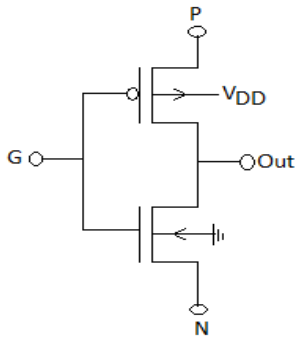


Figure 2. Diagram of MGDI Cell.

Table 2. Performance analysis of different parameters of different gates.

| Gates | Transistor counts | | | | Power dissipation in µW | | | | Switching delay in ns | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *GDI* | *MGDI* | *CMOS* | *TG* | *GDI* | *MGDI* | *CMOS* | *TG* | *GDI* | *MGDI* | *CMOS* | *TG* |
| AND | 2 | 2 | 6 | 6 | 1.284 | 0.982 | 1.71 | 1.52 | 0.8 | 0.8 | 1.4 | 0.8 |
| OR | 2 | 2 | 6 | 6 | 1.32 | 1.23 | 1.55 | 1.55 | 0.8 | 0.8 | 1.7 | 1.1 |
| NOT | 2 | 2 | 2 | 2 | 0.61 | 0.52 | 0.66 | 0.9 | 0.8 | 0.8 | 1.4 | 1.1 |
| XOR | 4 | 3 | 12 | 8 | 1.47 | 1.24 | 1.5 | 1.1 | 1.25 | 1.2 | 2.2 | 1.4 |
| XNOR | 4 | 3 | 12 | 8 | 1.46 | 1.25 | 1.6 | 1.2 | 1.25 | 1.2 | 2.2 | 1.4 |
| MUX | 2 | 2 | 8 | 6 | 1.9 | 1.9 | 2.3 | 2.1 | 0.8 | 0.8 | 2.1 | 1.0 |

## III. CONVENTIONAL 2-BIT MULTIPLIER

This section presents the logic diagram of conventional 2-bit multiplier. Figure 3 shows the diagram which requires four AND gates and two half adders. Suppose the number $A(A_1A_0)$ is to be multiplied with $B(B_1B_0)$, i.e.

$A_1\ A_0$
$B_1\ B_0$ X

 

---

$A_1B_0$ $A_0B_0$

$A_1B_1$ $A_0B_1$

---

$A_1B_1$ $A_1B_0$ $A_0B_0$

Carry $A_0B_1$

Therefore four AND gates perform $A_1B_1$, $A_1B_0$, $A_0B_1$ and $A_0B_0$ while two half adders perform ($A_1B_0 + A_0B_1$) and

($A_1B_1$ + previous carry). Here $P_0$, $P_1$, $P_2$ and $P_3$ are the partial products.

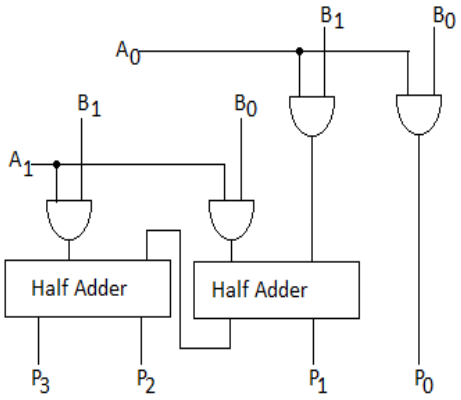


Figure 3. Conventional 2-bit multiplier circuit.

## IV. PROPOSED MULTIPLIER

This section introduces a new design technique of 2-bit multiplier. The truth table for the proposed technique is shown in Table 3.

From Table 3, using the Karnaugh's map minimization technique (shown in Figure 4), the equations for the product terms $P_1$, $P_2$, $P_3$ and $P_4$ can be found out as

$P_0 = A_0B_0$, (1)

$P_1 = A_1\overline{B_1}B_0 + A_1\overline{A_0}B_0 + \overline{A_1}A_0B_1 + A_0B_1\overline{B_0}$ (2)

$P_2 = A_1\overline{A_0}B_1 + A_1B_1\overline{B_0}$ (3)

$P_3 = A_1A_0B_1B_0$ (4)

According to Shannon's expansion theorem, equation (2) can be written as

$P_1 = \overline{B_1}B_0A_1 + B_1\overline{A_1}A_0 + \overline{B_0}A_0B_1 + B_0A_1\overline{A_0}$ (5)

and equation (3) can be written as

$P_2 = A_1B_1(\overline{A_0} + \overline{B_0})$ (6)

Equations (1), (5), (6) and (4) can be implemented using MGDI Cell which will be represented in the next section.

The implementation follows Table 1 for AND, OR and inverter.

Table 3. Truth table for the proposed multiplier.

| **Input** | | | | **Output** | | | |
|---|---|---|---|---|---|---|---|
| ***Number A*** | | ***Number B*** | | ***$P_3$*** | ***$P_2$*** | ***$P_1$*** | ***$P_0$*** |
| $A_1$ | $A_0$ | $B_1$ | $B_0$ | | | | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |

 

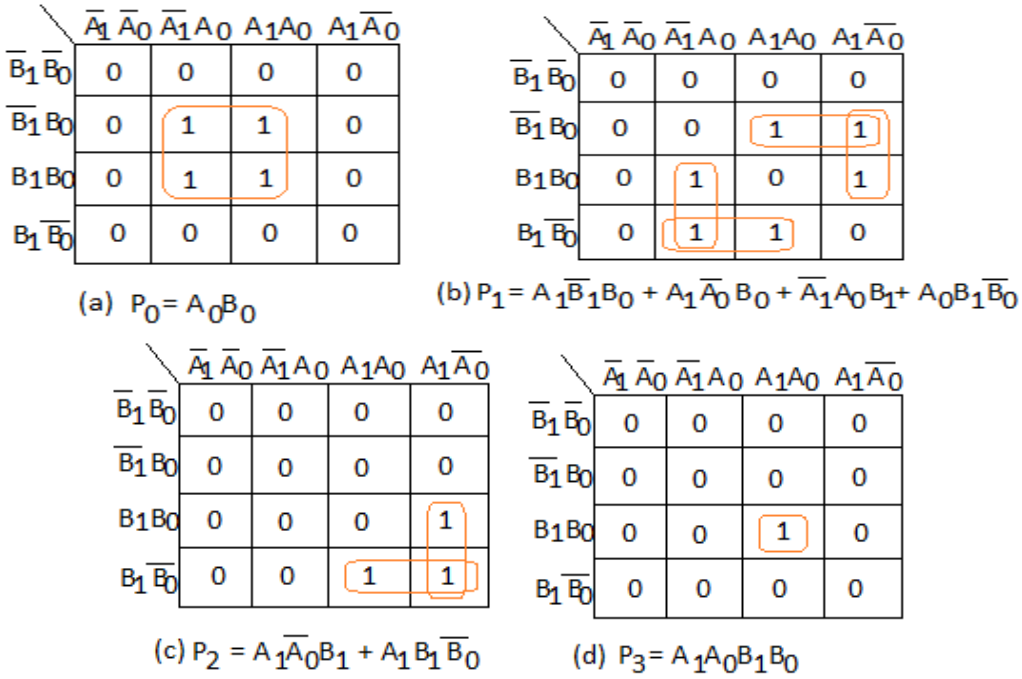


Figure 4. Karnaugh's map for the proposed 2-bit multiplier

## V. IMPLENTATION OF THE PROPOSED 2-BIT MULTIPLIER USING MGDI CELL

Figure 5 shows the implementation of the proposed 2-bit multiplier using MGDI cell. For this purpose, Dsch38 lite software has been used. Figure 6 shows the timing diagram. Table 4 compares the proposed method with conventional method using CMOS, PTL, Transmission gate and MGDI cell in terms of transistor counts, power dissipation and circuit delay.

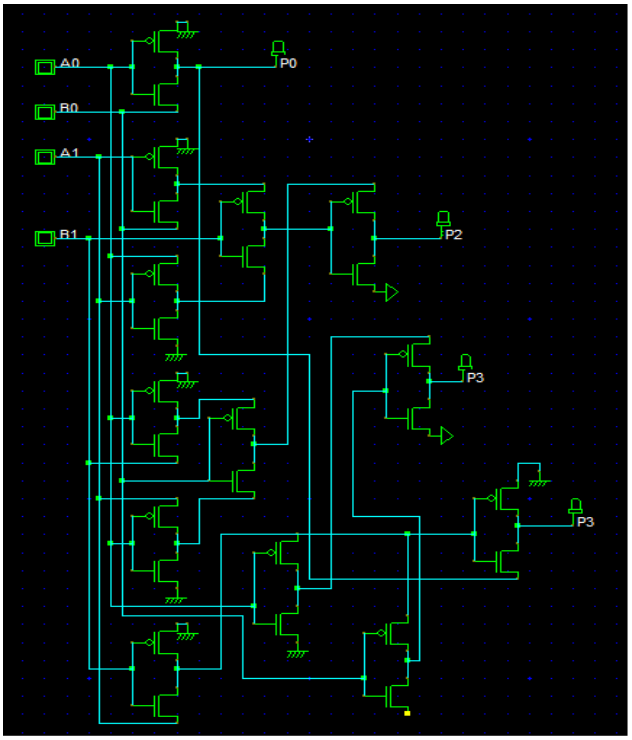


Figure 5. Implementation of the proposed method.

## VI. PROPOSED 4-BIT MULTIPLIER USING TWO 2-BIT MULTIPLIER

Figure 7 shows the block diagram of 4-bit multiplier using the 2-bit multipliers. In the Fig., register1 stores one 4-bit

number (say $B_3B_2B_1B_0$). SR1 (shift register) stores $A_2A_0$ and SR2 stores $A_3A_1$. For the 1st pulse, first 2-bit multiplier multiplies $A_1A_0$ with $B_1B_0$ and the product will go SR5. For the next pulse, second multiplier multiplies $A_1A_0$ with $B_3B_2$ and first multiplier multiplies $A_3A_2$ with $B_1B_0$ and the products will go to SR5 and SR4 respectively. These two product terms will be added in RCA1(Ripple carry adder) and the sum will be 2-times left shifted in SR3. This shifted sum is then added with the product of $A_1A_0$ and $B_1B_0$ in RCA2. The result is then feed back to RCA2 via register2. RCA2 will add the 4-times left shifted product term $A_3A_2B_3B_1$ with the previous result and stored in register2.

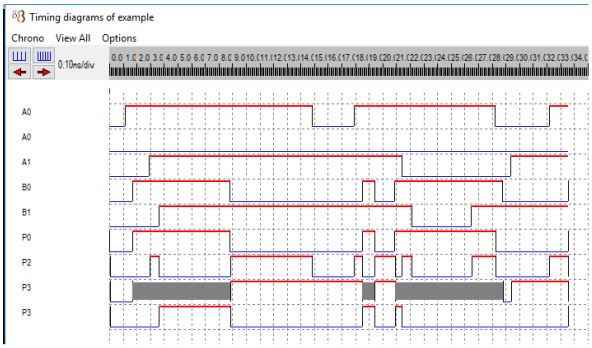


Figure 6. Timing diagram for the proposed 2-bit multiplier.

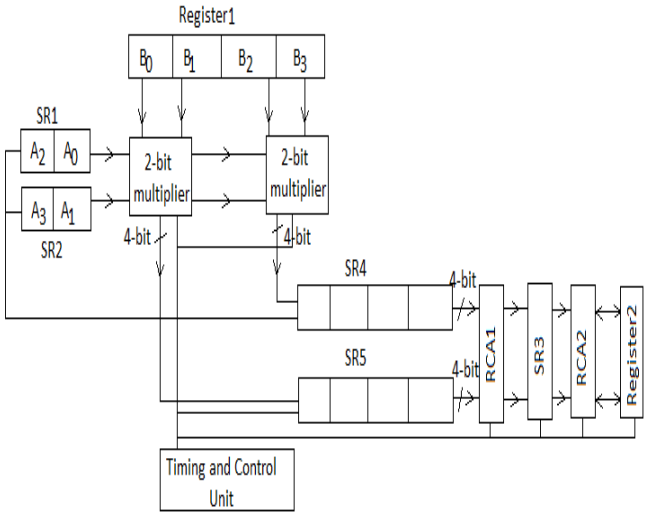


Figure 7. Block diagram of 4-bit multiplier.

The above architecture can also be used for multibit amplifier, only exception is the numbers of 2-bit multiplier and size of the registers and adders. The numbers of 2-bit multiplier depends on the numbers of bit constituted the numbers to be multiplied.

Table 4. Comparative study of the proposed method with conventional method using CMOS, PTL, Transmission gate.

| **Parameters** | **Conventional method using two half adder and four AND gates** | | | | **Proposed method** |
|---|---|---|---|---|---|
| | ***CMOS*** | ***PTL*** | ***Transmission gate*** | ***MGDI*** | |
| Transistor count | 48 | 36 | 40 | 18 | 26 |
| Propagation delay in ns | 10.61 | 11.75 | 11.54 | 8.76 | 5.24 |
| Power dissipation in $\mu W$ | 7.12 | 6.76 | 12.43 | 4.95 | 5.56 |

## VII. CONCLUSIONS

Though the transistor count for MGDI is lesser than the proposed method as shown in Table 4, the proposed method can be used for multi-bit multiplication whereas conventional method using MGDI is only for 2-bit multiplication. Since the proposed method for 2-bit multiplication is efficient with respect to propagation delay and power dissipation over CMOS, PTL and Transmission gate, it is the novelty of the proposed multi-bit multiplier.

## REFERENCES

[1] C.S. Wallace, "Asuggestion for a fast multiplier," IEEE Trans. Elec. Comput, vol. EC – 13, no. 1, pp. 14 – 17, Feb. 1964.

[2] Jagadguru Swami Sri Bharath, Krsna Tirathji, "Vedic Mathematics or Sixteen Simple Sutras From The Vedas," Motilal Banarsidas, Varanasi (India), pp(s). 23 – 33, 1986.

[3] C. Liu, J. Han, and F. Lombardi, "A Low – power, high – performance approximate multiplier with configurable partial error recovery," in Proc. Int. Conf. Design Automation, Test in Europe (DATE), Mar. 2014, Art. No. 95.

[4] 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, Dec. 2011.

[5] S. Balamurugan and P. S. Mallick, ``Fixed-width multiplier circuits using column bypassing and decompositon logic techniques," Int. J. Elect. Eng. Inform., vol. 7, no. 4, pp. 655 – 664, Dec. 2015.

[6] R. Marimuthu, Y. Elsie Rezinold and P.S. Mallick, "Design and Analysis of Multiplier using Approximate 15-4 Compressor", IEEE Access, vol. 5, pp(s). 1027 – 1036, 2017.

[7] K. Y. Kyaw,W. L. Goh, and K. S. Yeo, ``Low-power high-speed multiplier for error-tolerant application," in Proc. IEEE Int. Conf. Electron Devices Solid-State Circuits (EDSSC), Dec. 2010, pp. 1-4.

[8] G. Ganesh Kumar and V. Charishma, "Design of High Speed Vedic Multiplier using Vedic Mathematics Techniques", International Journal of Scientific and Research Publications, Volume 2, Issue 3, pp(s). 1-5, March 2012.

[9] H. Thapliyal and M.B. Srinivas, "High Speed Efficient N x N Bit Parallel Hierarchical Overlay Multiplier Architecture Based on Ancient Indian Vedic Mathematics", Transactions on Engineering, Computing and Technology, vol. 2, pp(s). 225 – 228,2004.

 

[10] Neha Goyal, Khushboo Gupta and Renu Singla, "Study of Combinational and Booth Multiplier", International Journal of Scientific and Research Publications, Volume 4, Issue 5, pp(s). 1-4, May 2014.

[11] B.S. Premananda, Samarth S. Pai, B. Shashank and Shashank S. Bhat, "Design and Implementation of 8-bit Vedic Multiplier", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol 2, issue 12, pp(s). 5877 – 5882, December, 2013.

[12] S. Maitra, "Design and Simulation of 4-bit Parallel Adder using Minimum Number of Transistor", International Journal of Modern Communication Engineering, vol. no. – 7, issue no. – 3, pp(s). 13 – 18, May, 2019.

[13] R. Uma and P. Dhavachelvam, "Modified Gate Diffusion Input Technique: A New Technique for Enhancing Performance in Full Adder Circuits," Proceedings of 2nd International Conference on Communication, Computing and Security (ICCS2012), pp(s). 74 – 81.

**Author profile**

**Subhashis Maitra,** received the B.Tech. Degree in Instrumentation and Electronics Engineering and M.E. in Electronics and Tele-communication Engineering from Jadavpur University, Jadavpur, Kolkata – 700032, West Bengal, India and Ph. D. Degree from Maulana Abul Kalam Azad University of Technology, Salt Lake, West Bengal, India, in Computer Science. He works on Digital circuits and systems, Digital Signal Processing, Control System. He is an Assistant Professor of Kalyani Government Engineering College, Kalyani, Nadia, West Bengal, India. His publications is more than twelve in different International Journal.

 