International Journal of Modern Science and Technology, 2020;5(11):251-265.
ISSN: 2456-0235.
www.ijmst.co

Research Article

A novel design and analysis of low power multipliers using full swing gate diffusion input method

R. Indhumathi*, K. Gavaskar, S. R. Nandhini, J. Raja Nandhini

Department of Electronics and Communication Engineering, Kongu Engineering College, Perundurai, Erode. India.

*Corresponding author's e-mail: indhuma2228@gmail.com

Abstract

In the present paper mainly focused on accuracy and power of different types of multipliers using full swing gate diffusion input (FSGDI) method. The main objective of this paper is to reduce the power consumption and improve the accuracy of the multipliers. In normal gate diffusion input method (GDI) required less transistors compared with full swing gate diffusion method. But cannot get the full level of output there is some charge sharing problem is occurred in GDI method. In FSGDI method the transistor counts gets increased compared with GDI method but there is no charge sharing problem is occurred. There are three types of multiplier was implemented using FSGDI method like conventional array multiplier, Vedic multiplier and Wallace tree multiplier. Based on the power results and performance analysis Wallace tree multiplier is efficient to design using Tanner tool version 13.

Keywords: Full swing gate diffusion input; Gate diffusion input; Array multiplier; Vedic multiplier; Wallace tree multiplier; Low power.

Introduction

Today, more and more, high speed mobile computational devices and equipment are being introduced in the market. These computational devices strain and drain the battery very quickly in these computational devices. Researchers are making efforts to find ways and means to conserve the battery power for longer period. Multipliers is the key components in computational devices to support high speed computational intensive applications in real time[1]. Thus it becomes more important to reduce power dissipation and improve the accuracy in these multiplier modules as they affect the performance of the device.

Several VLSI design techniques have been attempted to optimize the power and improve accuracy by the multiplier module, but there are very few design techniques that gives the required extensibility both in terms of power and accuracy. In this paper a high speed, reliable and efficient multiplier VLSI module design is presented using FSGDI (Gate Diffusion Input) technique [2], addressing both power consumption and accuracy. Further, comparative study results of the proposed design over the traditional Gate Diffusion Input (GDI) design are also presented [3]. Detailed design steps and comparative study using Tanner simulation tool at 250 nm CMOS technology is discussed.

Proposed method

The present work is mainly focused on reduction of power and improve the accuracy of 4-bit conventional multiplier, vedic multiplier and Wallace tree multiplier using FSGDI.

Full swing gate diffusion input

In normal GDI technique there is a charge sharing problem is occurred but it has the area efficient compared with CMOS and FSGDI method. In CMOS technology the substrate terminal of PMOS transistor is always connected to VDD and NMOS transistor is connected to GND. But in GDI method it may change the VDD and VSS according to the required design.

Array multiplier

An array multiplier is a digital combinational circuit used for multiplying two unsigned binary numbers by using the full adders and half adders circuit [4]. It is used to perform simultaneous addition of the various product terms. An array of AND gates is used before the Adder array circuit to form the product terms.

Received: 28.09.2020; Received after Revision: 10.11.2020; Accepted: 10.11.2020; Published: 04.12.2020.
©2020 The Authors. Published by G. J. Publications under the CC BY license.

Vedic multiplier

To improve the speed and reducing the area Vedic multipliers is plays important role in multiplication [5, 6]. Vedic Mathematics also used for multiplication which can be increase the speed of multiplier by reducing the number of iterations that means it increase the speed of the multiplier and processor or system.

Wallace tree multiplier

By using the full adders and half adders Wallace tree multiplier is used to reduce the partial product tree into two rows, and then a final adder is used to add these two rows of partial products to generate the output [7, 8]. Wallace tree multiplier perform multiplication operation in four steps. (1) Generate the all partial products. (2) By using the half adder and full adder the partial product tree is reduced until it to be two terms. (3) Finally, a fast adder circuit is used to add these two terms [9, 10]. (4) Then the final output will be generated.

Design and simulation

AND, OR and EX-OR Gate design using GDI method

Fig. 1 shows the AND gate design using one PMOS and one NMOS transistors. Input a is given to the gate terminal and b is given to the source terminal of NMOS. PMOS is connected to GND.

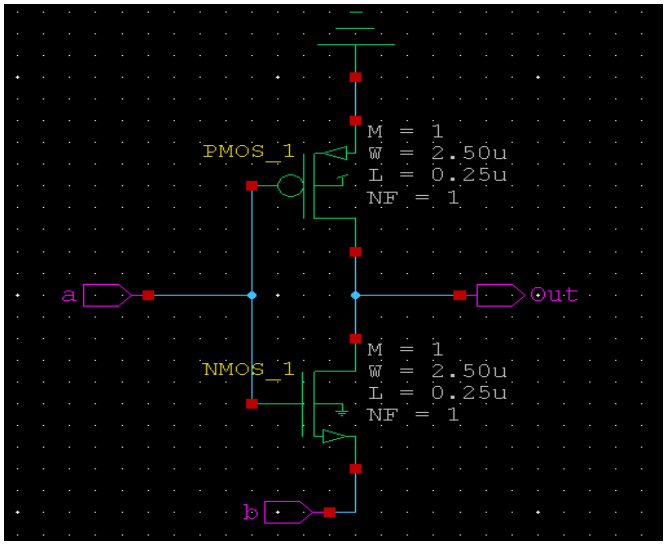

Fig. 1. AND gate using GDI

Fig. 2 shows the simulation output for AND gate. The output is 1 when both the input is 1. There is some distortion occurred in output.

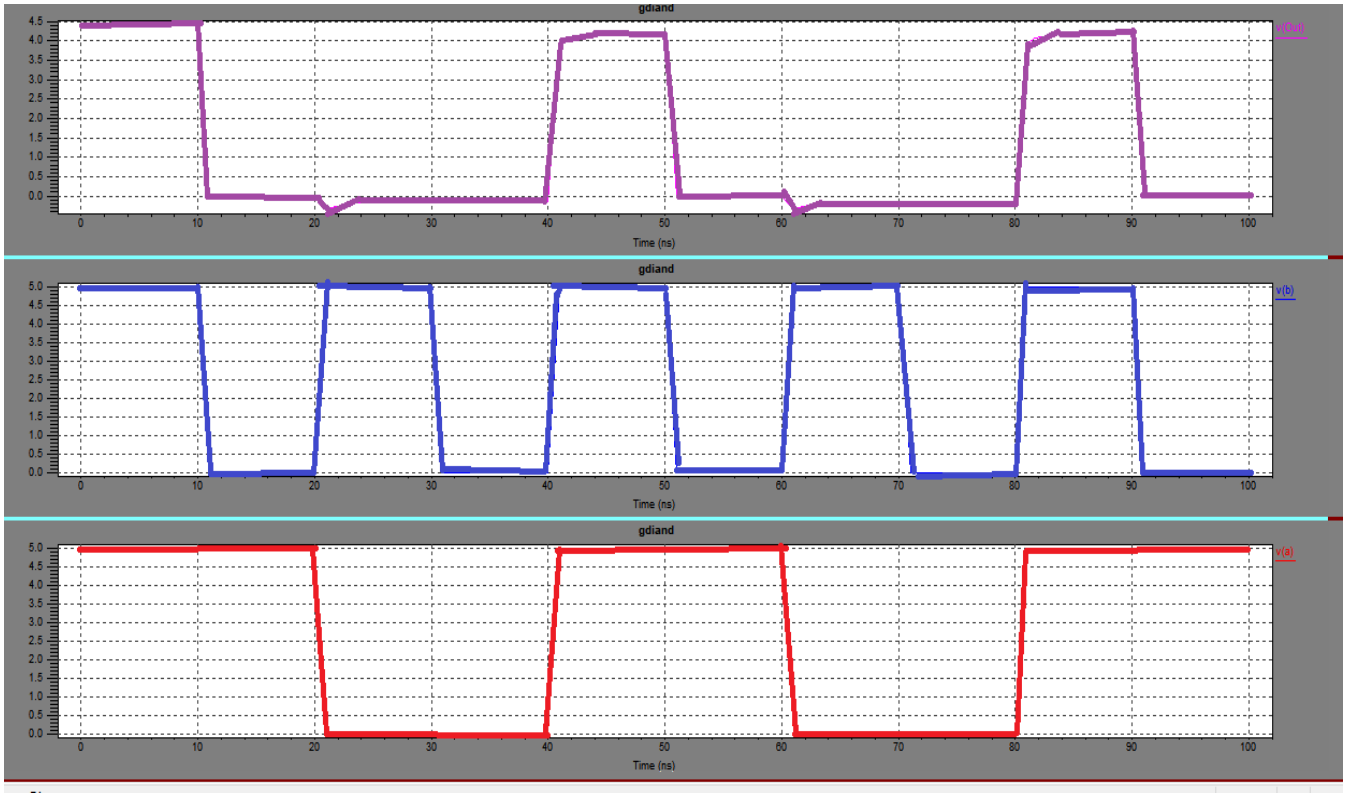

Fig. 2. Simulation output of AND gate using GDI

Fig. 3 shows the OR gate design using one PMOS and one NMOS transistors. Input a is given to the gate terminal and b is given to the source terminal of PMOS. NMOS is connected to GND.

©2020 The Authors. Published by G. J. Publications under the CC BY license.

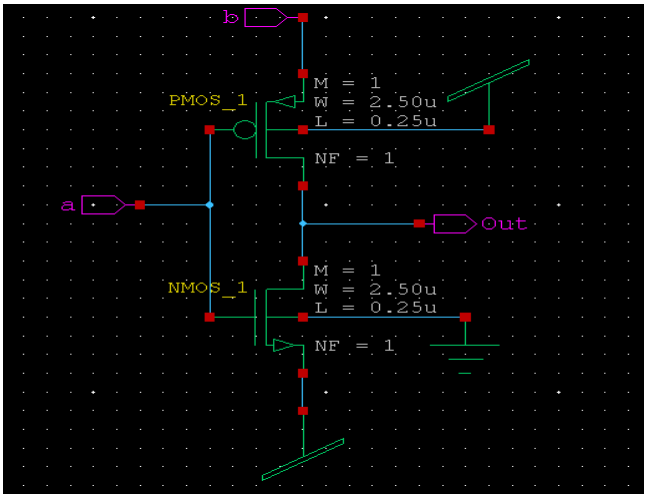

Fig. 3. OR gate using GDI

Fig. 4 shows the simulation output for OR gate. The output is 1 when any one of the input is 1. There is some distortion occurred in output. Fig. 5 shows the EX-OR gate design using two PMOS and two NMOS transistors. Input a and b is given to the gate terminal. Fig. 6 shows the simulation output for EX-OR gate. The output is 1 when the inputs are 01 and 10. There is some distortion occurred in output.

AND, OR and EX-OR Gate design using FSGDI method

Fig. 7 shows the AND gate using FSGDI method. There are 2 PMOS and 3 NMOS transistor are required.

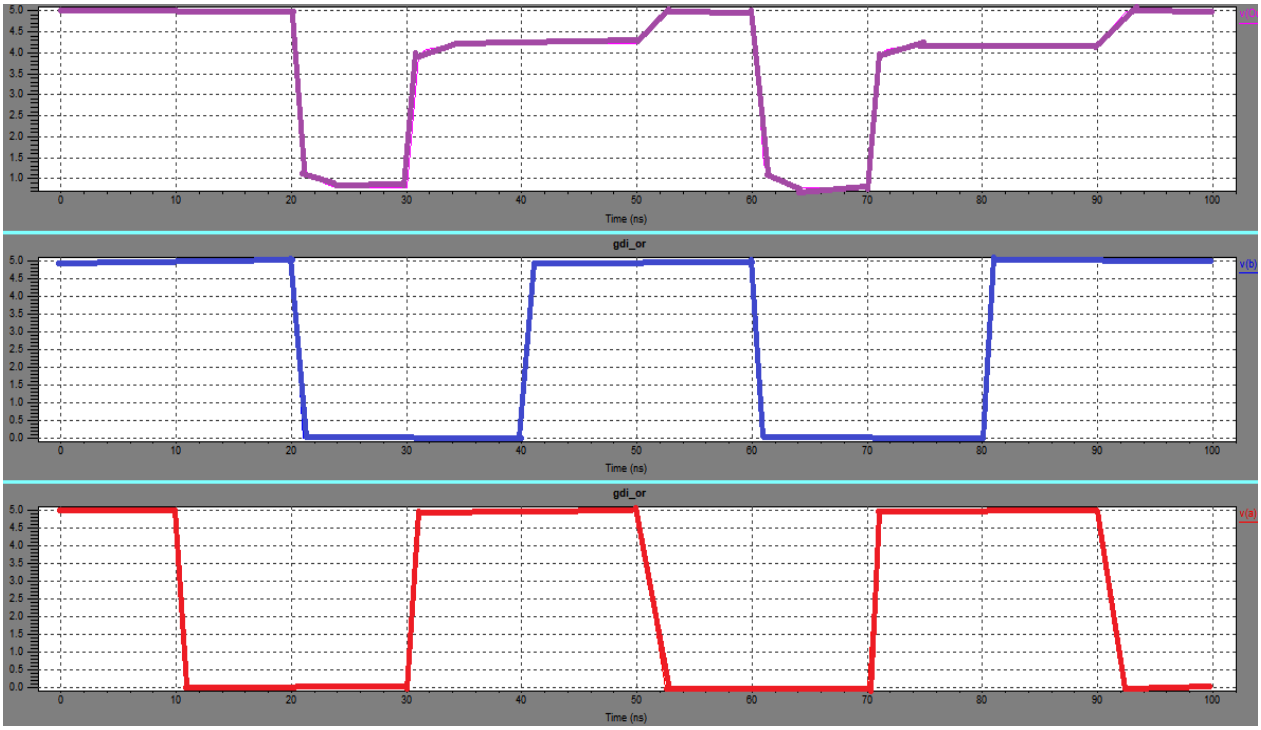

Fig. 4. Simulation output of OR gate using GDI

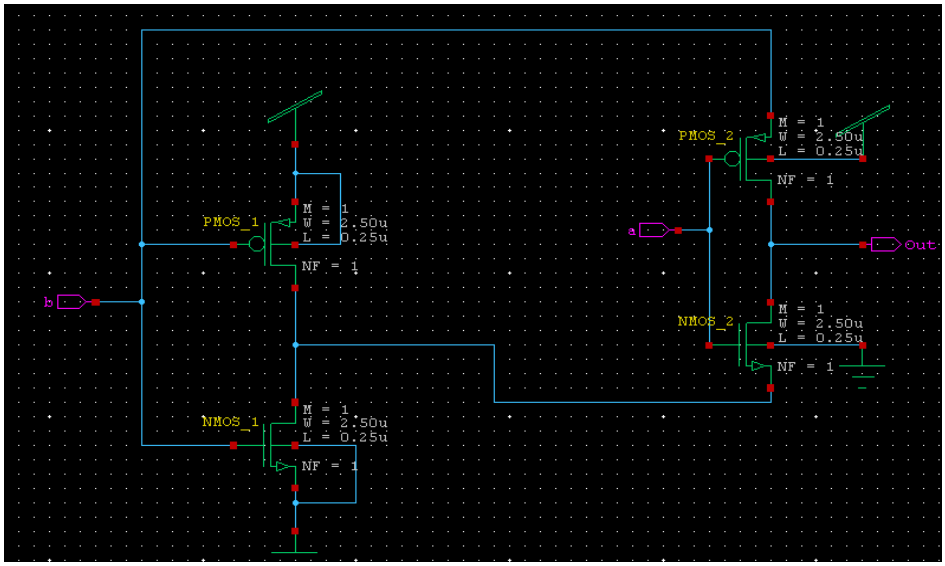

Fig. 5. EX-OR gate using GDI

©2020 The Authors. Published by G. J. Publications under the CC BY license.

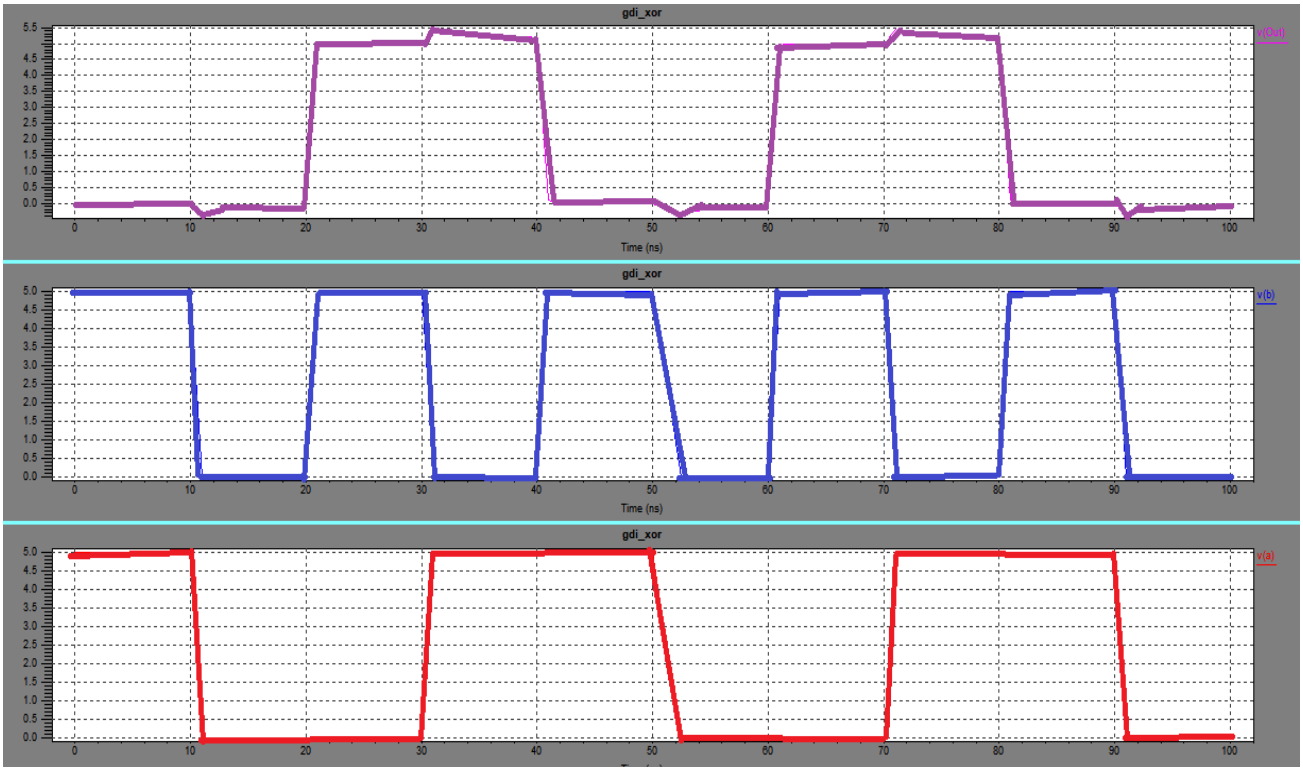

Fig. 6. Simulation output of EX-OR gate using GDI

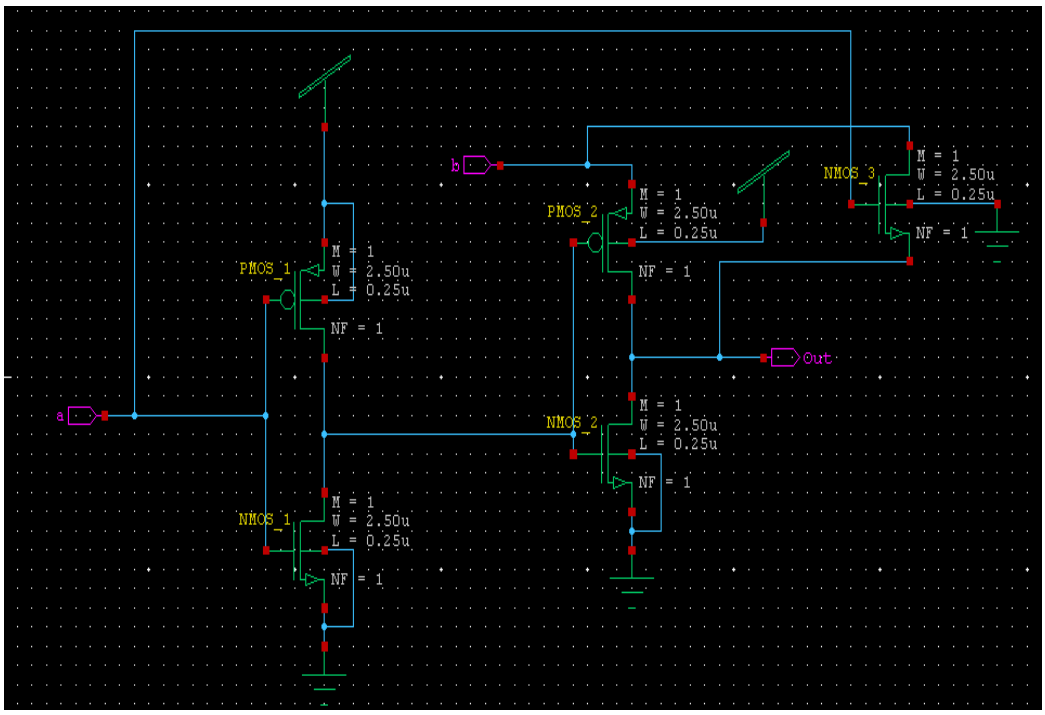

Fig. 7. AND gate using FSGDI

Fig. 8 shows the simulation output for AND gate. The output is 1 when both the input is 1. There is no distortion occurred in output. Fig. 9 shows the OR gate using FSGDI method. There are 3 PMOS and 3 NMOS transistor are required. Fig. 10 shows the simulation output for OR gate. The output is 1 when any one of the input is 1. There is no distortion occurred in output. Fig. 11 shows the EX-OR gate design using 3 PMOS and 3 NMOS transistors. Input a and b is given to the gate terminal. Fig. 12 shows the simulation output for EX-OR gate. The output is 1 when the inputs are 01 and 10. There is some distortion occurred in output.

Adders design using GDI and FSGDI method

Fig. 13 shows the half adder using GDI and FSGDI. By converting AND and EX-OR gate into a symbol to generate sum and carry.

©2020 The Authors. Published by G. J. Publications under the CC BY license.

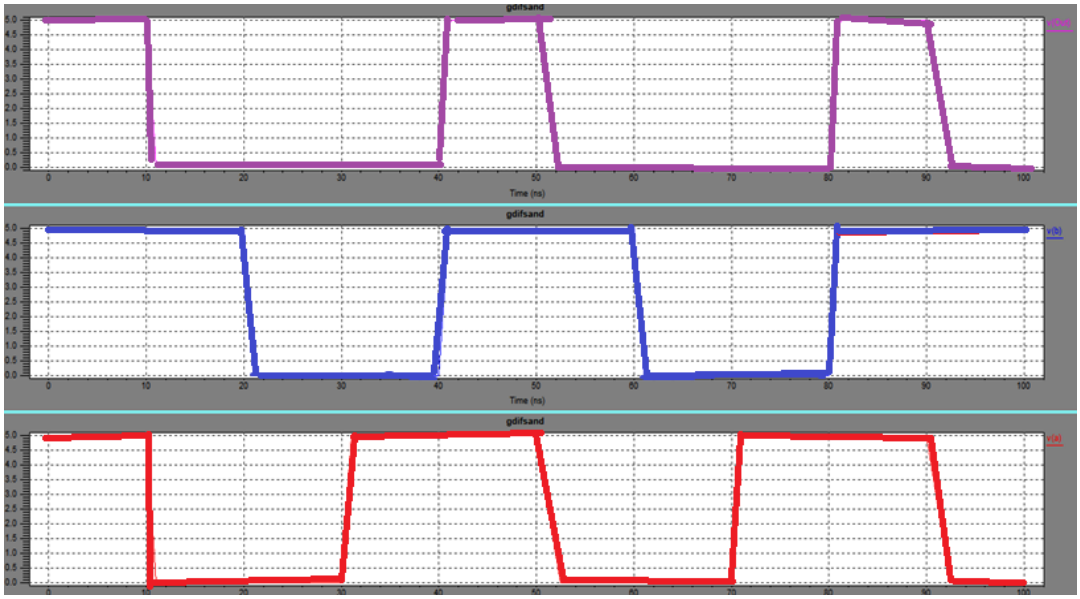

Fig. 8. Simulation output of AND gate using FSGDI

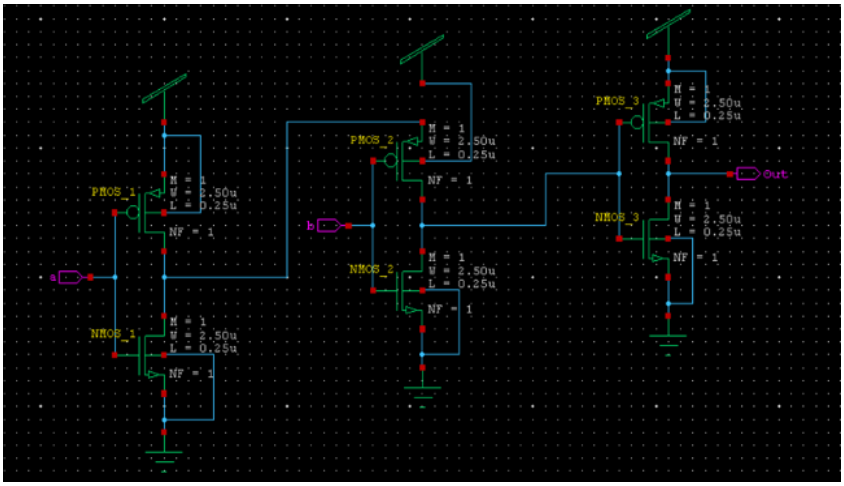

Fig. 9. OR gate using FSGDI

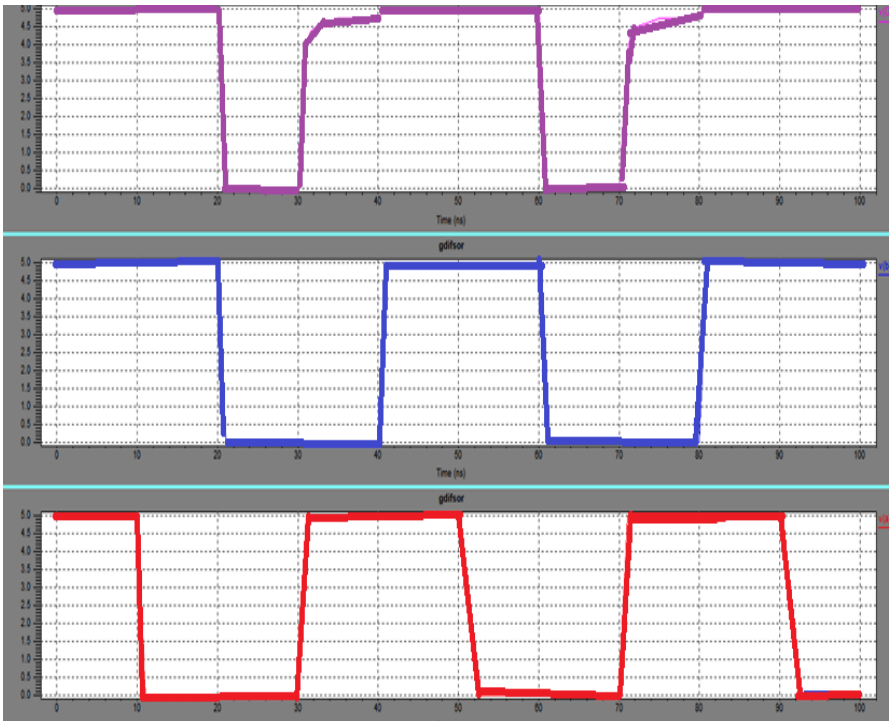

Fig. 10. Simulation output of OR gate using FSGDI

©2020 The Authors. Published by G. J. Publications under the CC BY license.

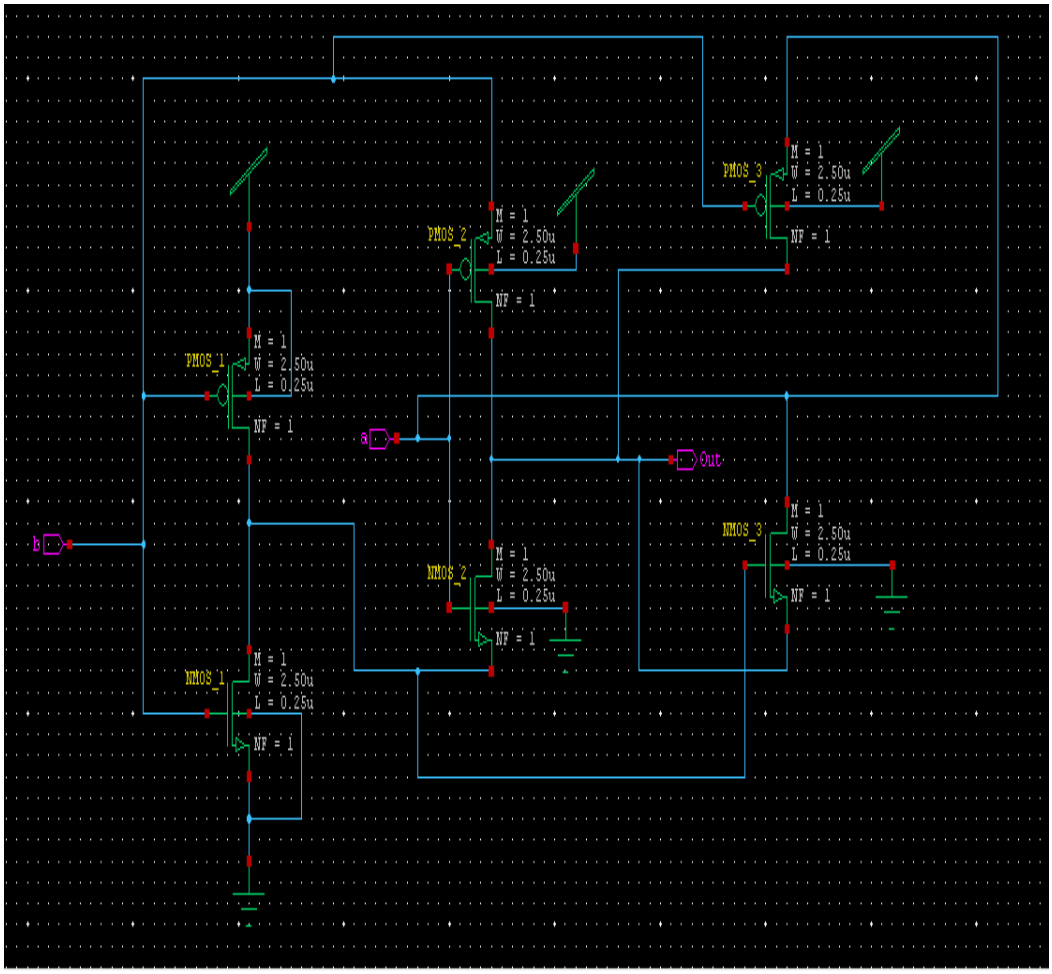

Fig. 11. EX-OR gate using FSGDI

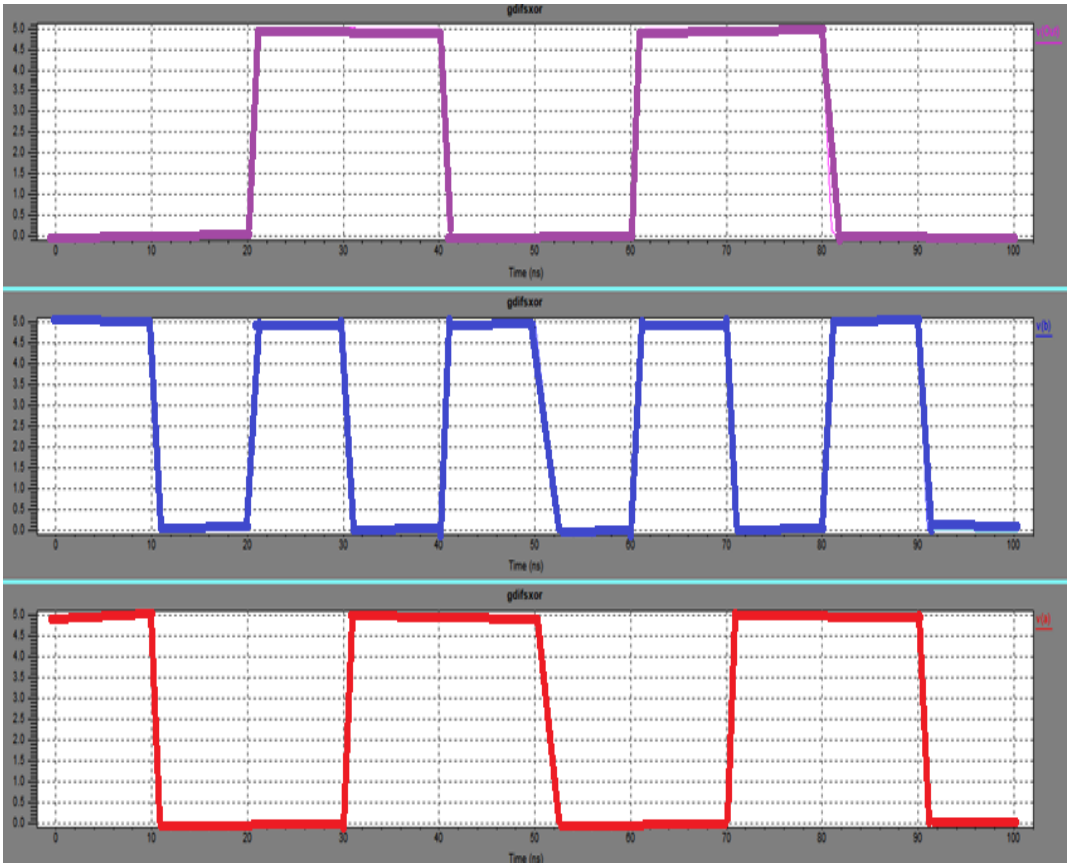

Fig. 12. Simulation output of EX-OR gate using FSGDI

©2020 The Authors. Published by G. J. Publications under the CC BY license.

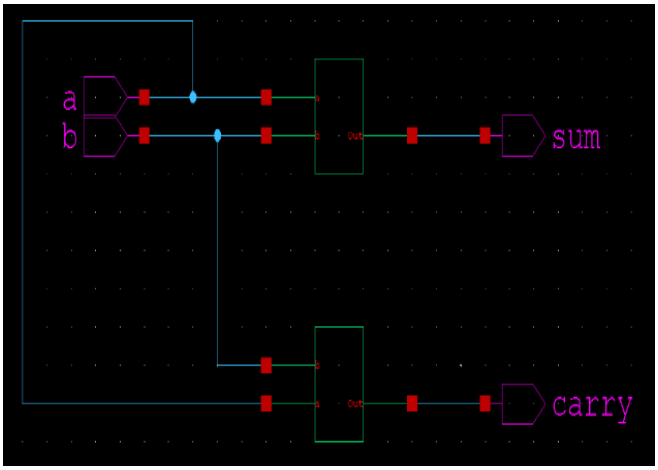

Fig. 13. Half adder design using GDI

Fig. 14 shows the full adder using GDI and FSGDI. By converting half adder and OR gate into a symbol to generate sum and carry. From Fig. 15, 16, 17 and 18 shows the simulation output for half adder and full adder using GDI and FSGDI. If any one of the input is 1 the sum output is 1 and carry is 0.

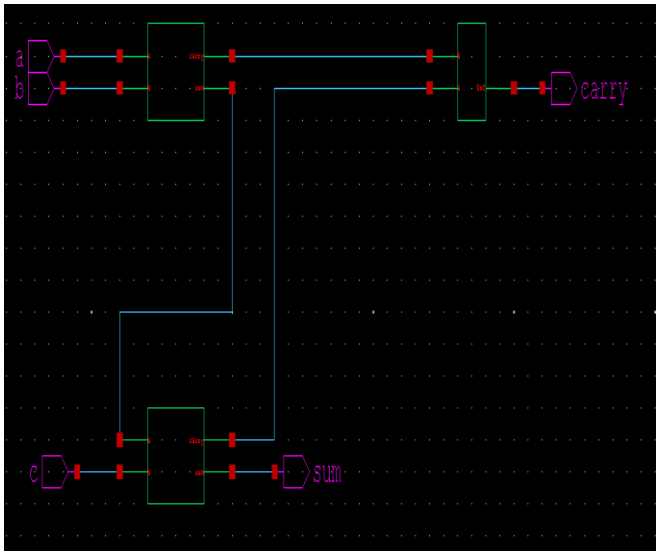

Fig. 14. Half adder design using GDI

If both the inputs are 1 the sum is 0 and carry will be 1. Fig. 19 and Fig. 20 shows the design of RCA using FSGDI method. The full adder is covert into a symbol for input a0,a1,a2,a3 and b0,b1,b2,b3 and s0,s1,s2,s3 are sum output cout is carry out.

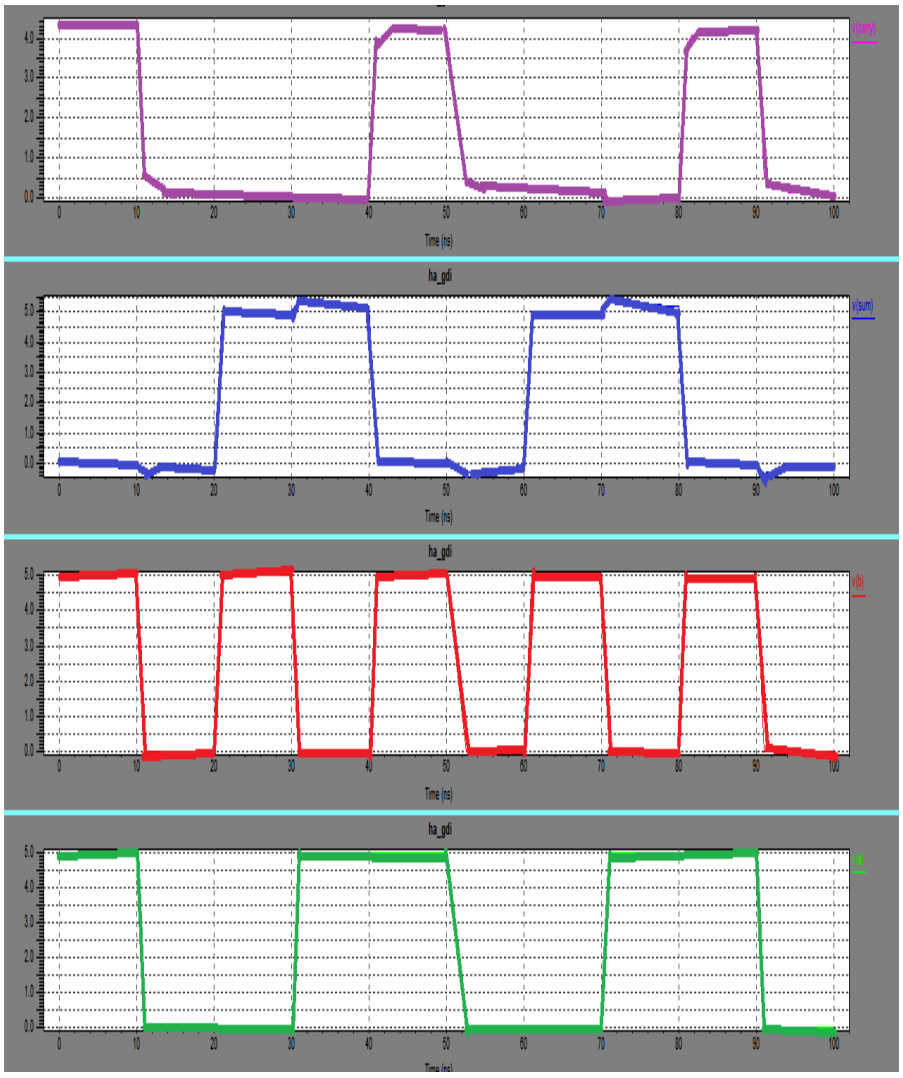

Fig. 15. Simulation output for half adder using GDI

©2020 The Authors. Published by G. J. Publications under the CC BY license.

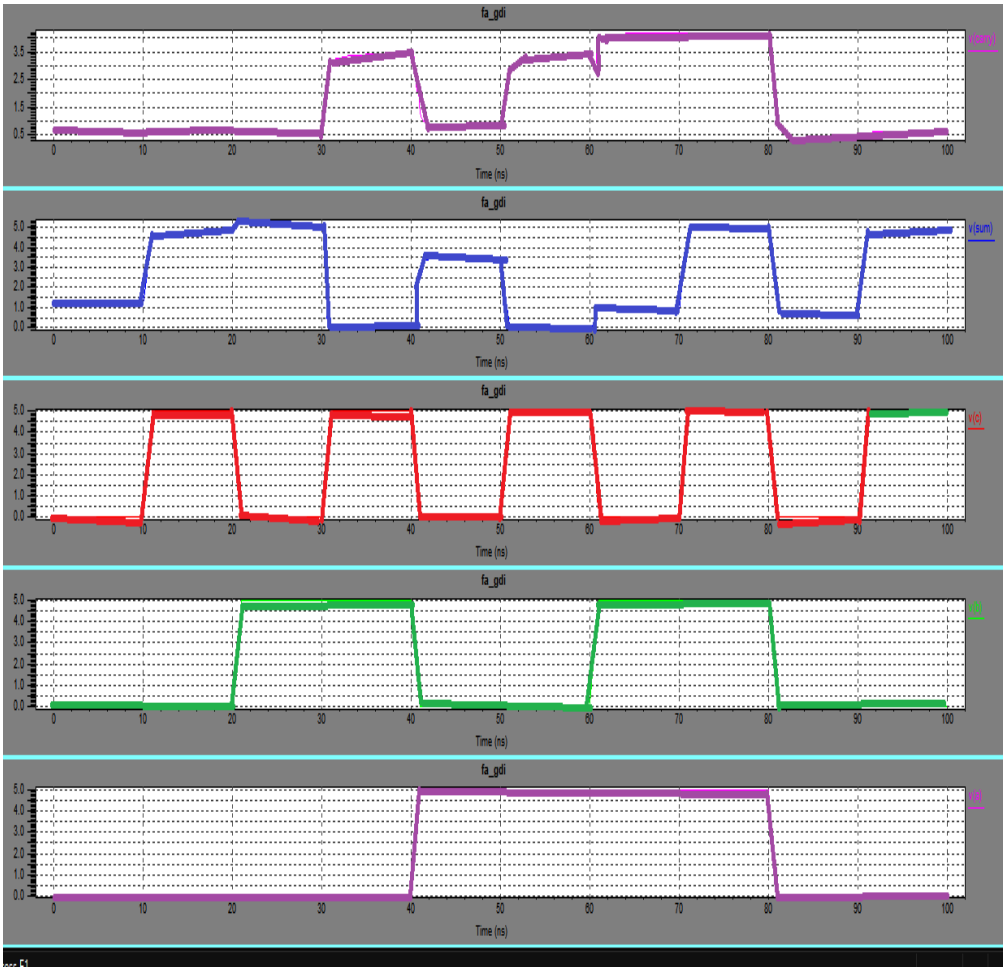

Fig. 16. Simulation output for full adder using GDI

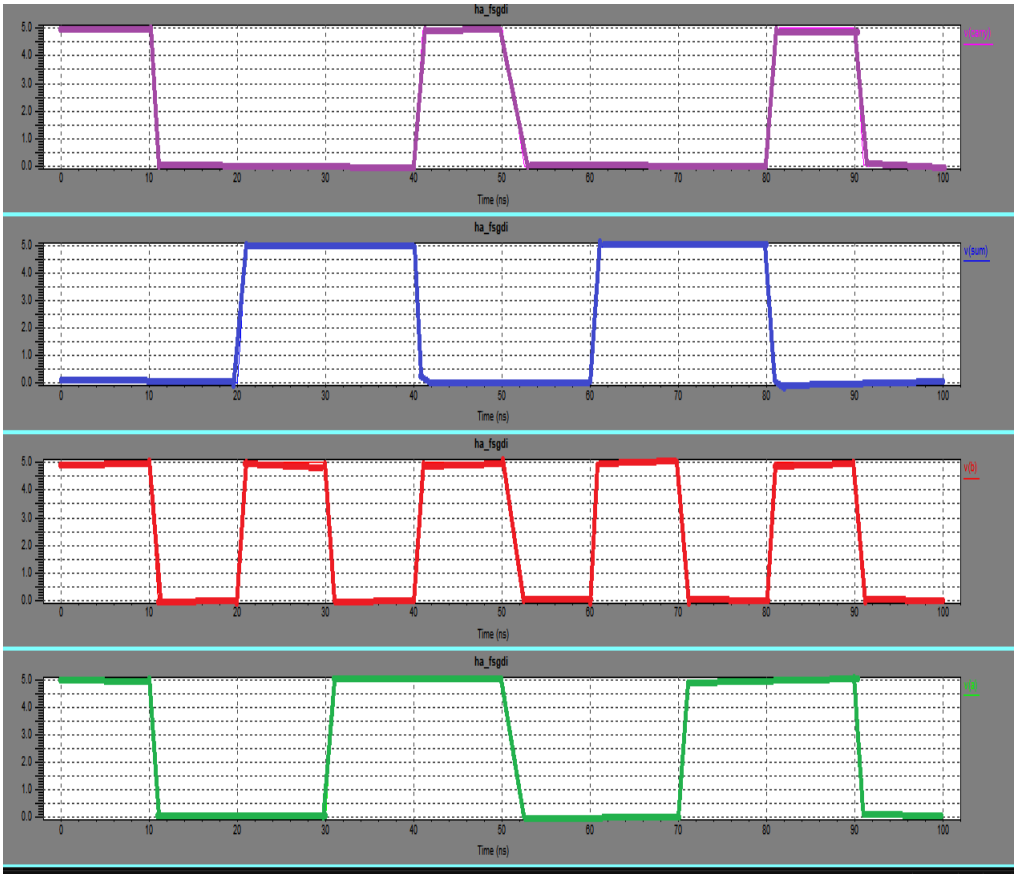

Fig. 17. Simulation output for half adder using FSGDI

©2020 The Authors. Published by G. J. Publications under the CC BY license.

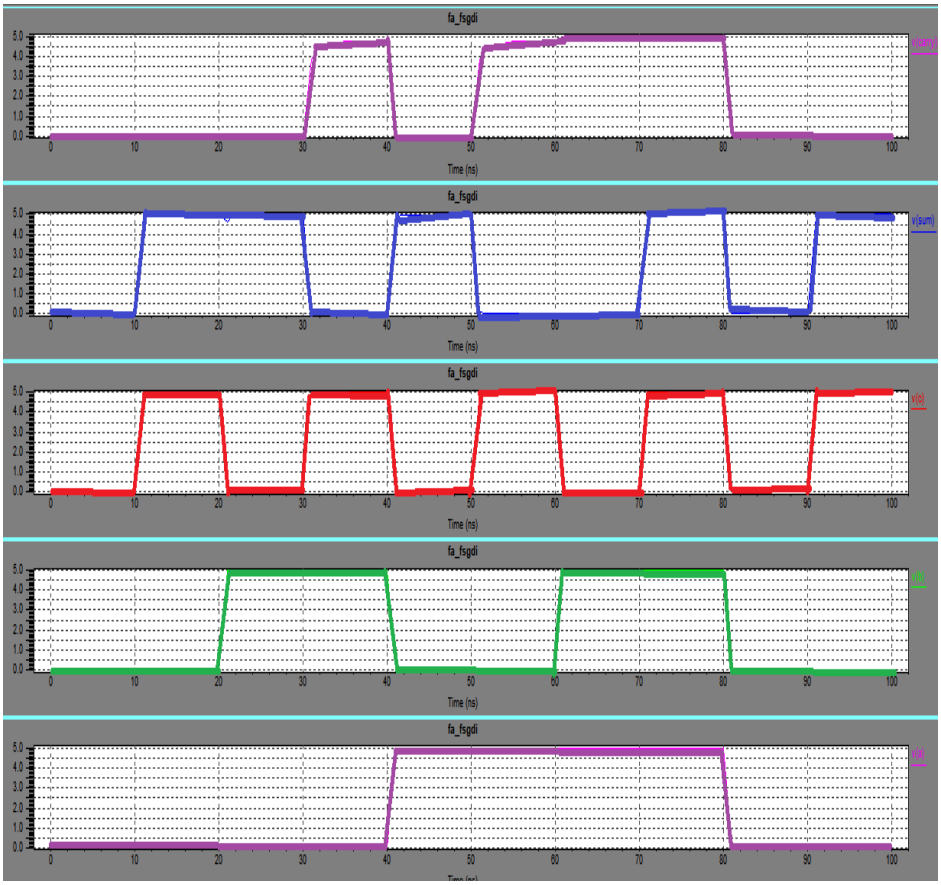

Fig. 18. Simulation output for full adder using FSGDI

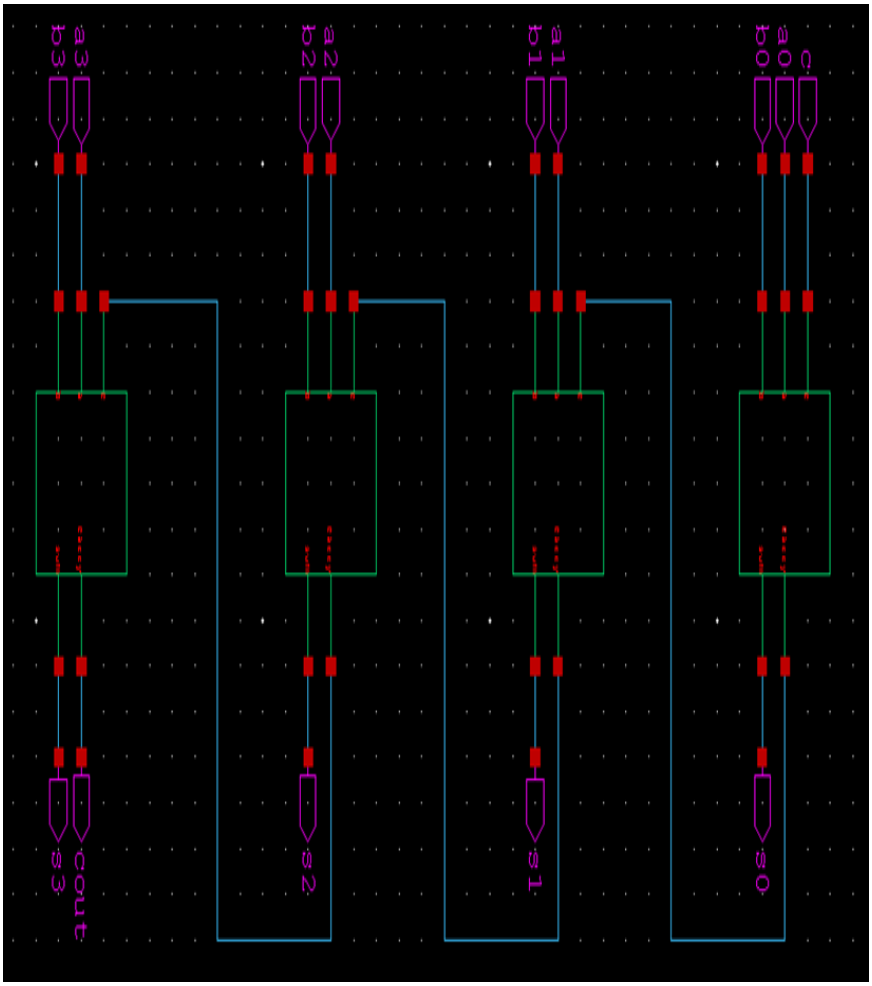

Fig. 19. 4-bit Ripple Carry Adder

©2020 The Authors. Published by G. J. Publications under the CC BY license.

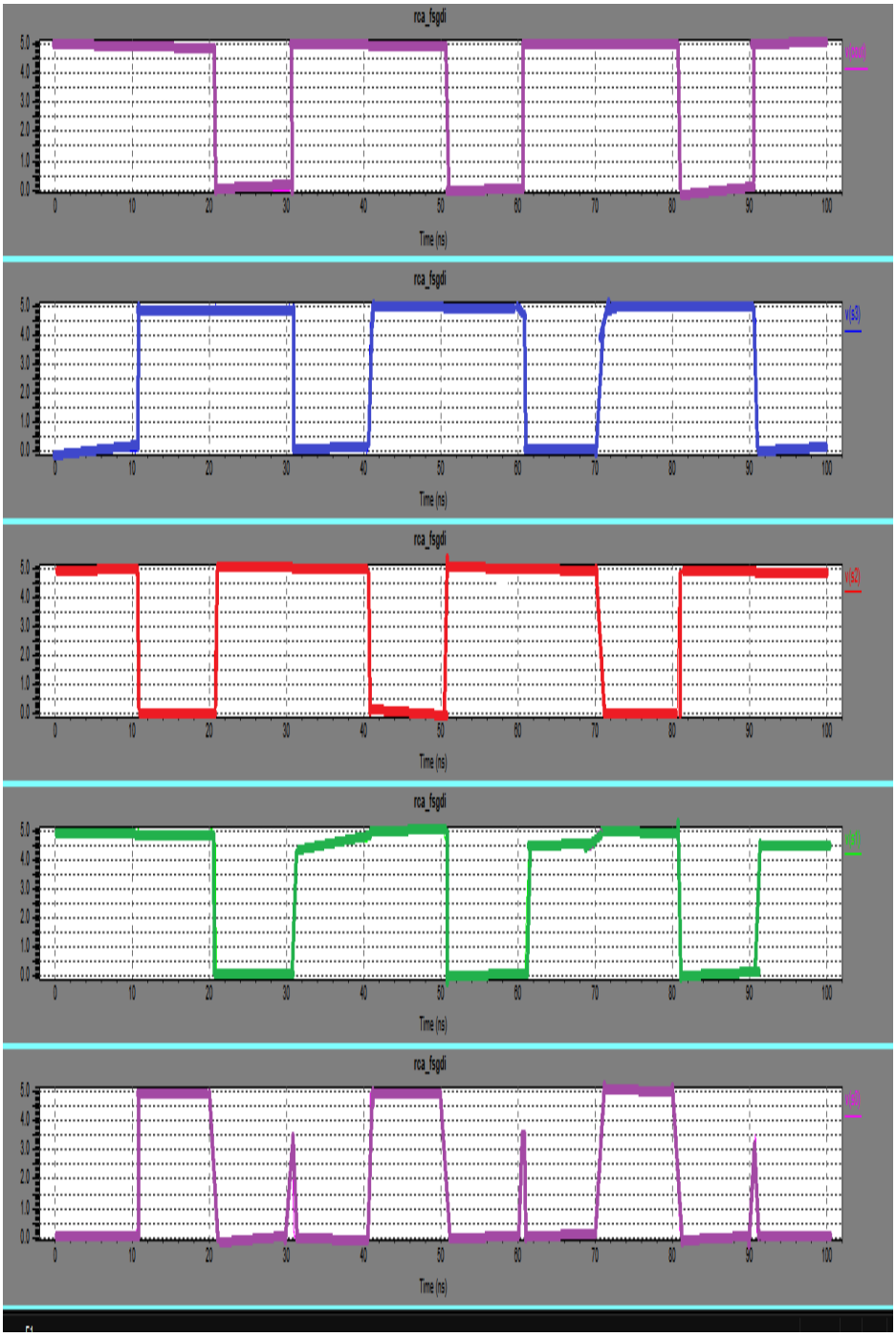

Fig. 20. Simulation output of 4-bit Ripple Carry Adder

Multipliers design and output using FSGDI method

Fig. 21 shows the 4-bit array multiplier design using FSGDI. There are 16 AND gates, 4 half adders and 8 full adders are required to design this multiplier. Fig. 22 shows the 4-bit Vedic multiplier using FSGDI. There are 4 2-bit Vedic multiplier and 3 4-bit RCA are required to design this multiplier. Fig. 23 shows the 4-bit Wallace tree multiplier design using FSGDI. There are 16 AND gates, 4 half adders and 8 full adders are required to design this multiplier.

Fig. 24, 25 and 26 shows the simulation outputs for 4-bit conventional array multiplier, Vedic multiplier and Wallace tree multiplier using FSGDI method.

Result and discussion

In the present work, authors compared the result and analysis of power, area and accuracy of the multipliers. By using the FSGDI method the accuracy was increased that means no charge sharing problem is occurred at the output side but the area gets increased that is the disadvantage compared with GDI method. There are 358 transistors are required to design

©2020 The Authors. Published by G. J. Publications under the CC BY license.

conventional array and Wallace tree multiplier and 390 transistors are required to design the Vedic multiplier. According to the power result conventional consumes 10.128mW power, Vedic multiplier consumes 13.125mW power and 6.977mW power consume for Wallace tree multiplier. Based on the area, power and accuracy results Wallace tree multiplier is efficient in order to get full output swing 20% of area has been reduced and 48% power gets reduced using FSGDI method.

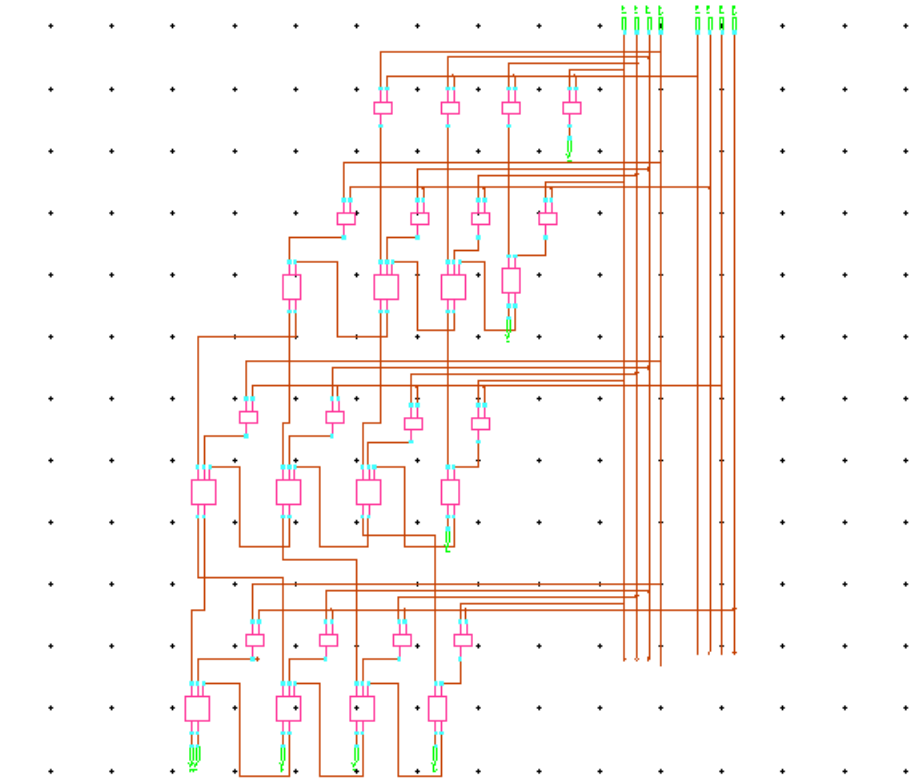

Fig. 21. 4-bit Conventional Array Multiplier

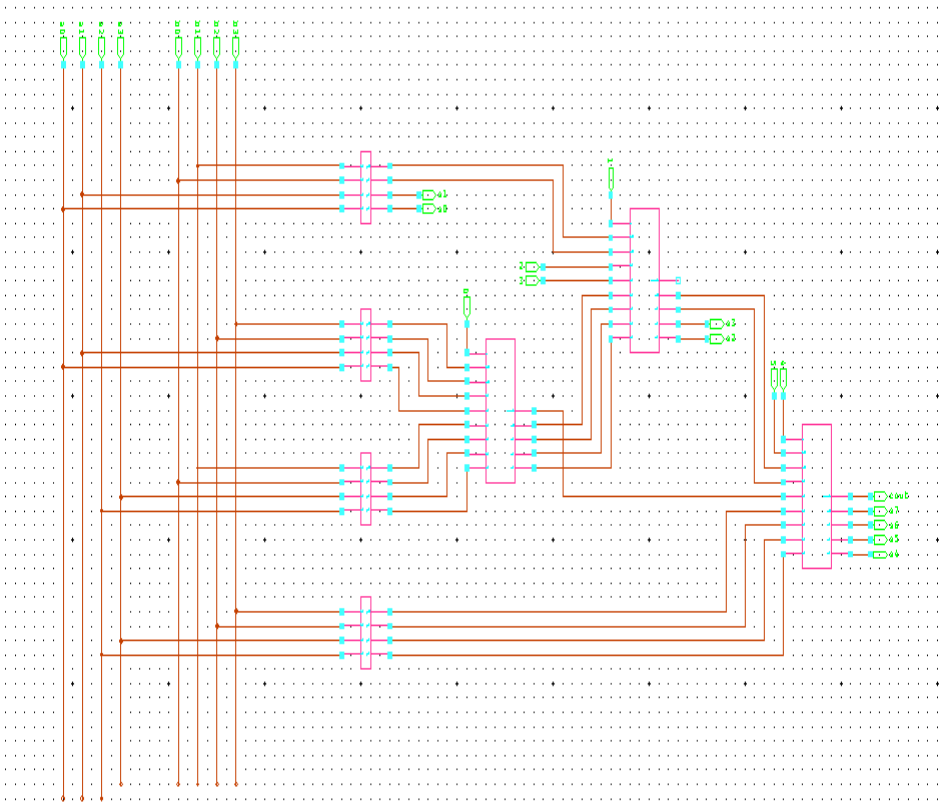

Fig. 22. 4-bit Vedic Multiplier

©2020 The Authors. Published by G. J. Publications under the CC BY license.

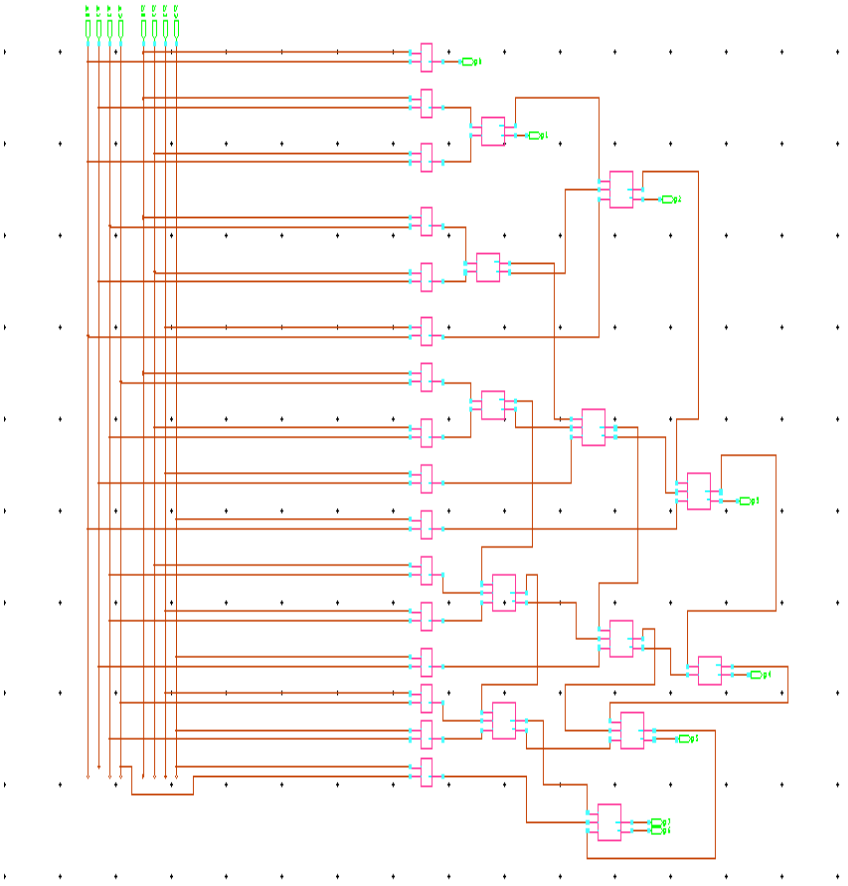

Fig. 23. 4-bit Wallace tree Multiplier

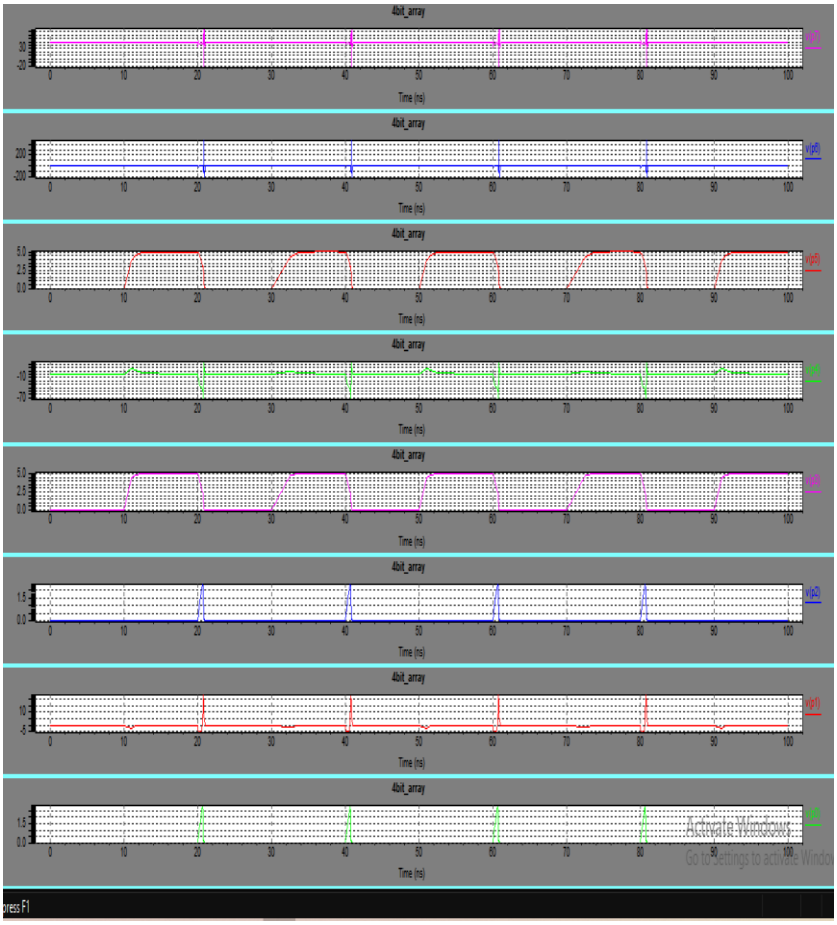

Fig. 24. Simulation output of conventional array multiplier

©2020 The Authors. Published by G. J. Publications under the CC BY license.

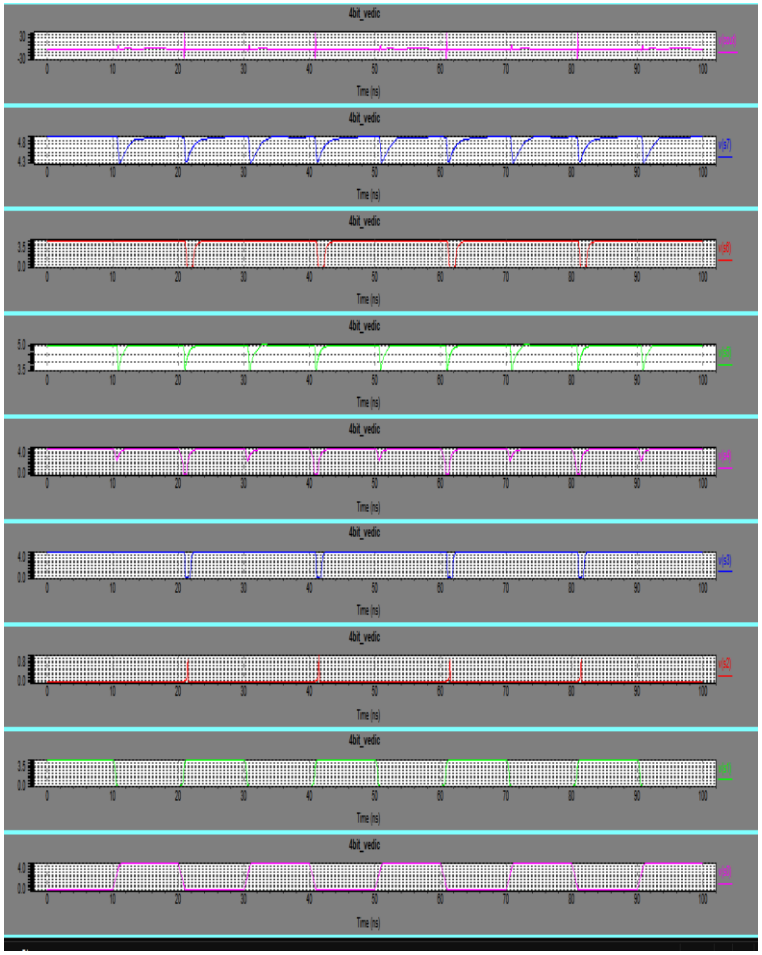

Fig. 25. Simulation output of Vedic multiplier

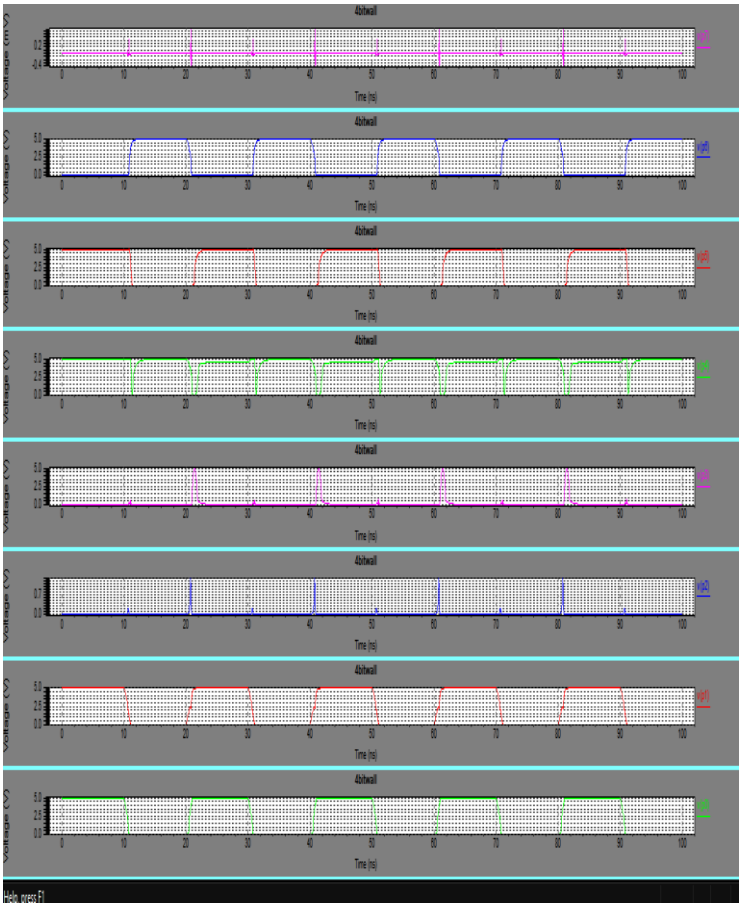

Fig. 26. Simulation output Wallace tree multiplier

©2020 The Authors. Published by G. J. Publications under the CC BY license.

Table 1. Comparison of area and power

Multipliers	Area (number of transistors)	Power (mW)
Conventional Array Multiplier	358	8.138
Vedic Multiplier	390	13.125
Wallace tree Multiplier	358	6.977

Table 2 shows the voltage and power relation of three multipliers. If the voltage increases the power gets increases.

Table 2. Voltages versus power relation

Voltage (V)	Array (mW)	Vedic (mW)	Wallace Tree (mW)
3	0.057	1.088	1.529
3.2	0.803	2.091	2.026
3.4	1.657	2.933	2.520
3.6	2.599	3.893	3.035
4.0	3.560	4.938	3.749
4.2	4.440	6.016	4.344
4.4	5.412	8.016	5.021
4.6	6.257	9.177	5.803
4.8	7.206	10.509	6.520

Conclusions

Based on the comparison values Wallace tree multiplier is efficient to design by using FSGDI method. The 20% of area has been reduced and 48% of power gets reduced. In future we compute delay and reduce the area of multiplier using FSGDI method.

Conflict of interest

Authors declared no conflict of interests.

References

[1] Hussain I, Chaudhury S. A comparative study on the effects of technology nodes and logic styles for low power high-speed VLSI applications. International Nanoparticles 2020;12(2):122-30.

[2] Morgenshtein A. Full-Swing Gate Diffusion Input logic—Case-study of low-power CLA adder design. Journal of Integration 2014;4(1):62-70.

[3] Hussain I, Pandey SK, Chaudhury S. Design and analysis of high performance multiplier circuit. Devices for Integrated Circuit Devices for Integrated Circuit (DevIC), Kalyani, India, 2019, pp. 245-247, doi: 10.1109/DEVIC.2019.8783322.

[4] Bhardwaj K, Mane PS, Henkel J. Power- and area-efficient approximate wallace tree multiplier for error-resilient systems. Fifteenth IEEE International Symposium on Quality Electronic Design. 2014.

[5] Jaiswal KB, Nithish Kumar V, Seshadri P, Lakshminarayanan G. Low power wallace tree multiplier using modified full adder, 3rd International Conference on Signal Processing, Communication and Networking (ICSCN), Chennai, 2015, pp. 1-4, doi: 10.1109/ICSCN.2015.7219880.

[6] Sureka N, Porselvi R, Kumuthapriya K. An efficient high speed Wallace tree multiplier. IEEE International Conference on Information Communication and Embedded Systems (ICICES). 2020;7: 962-975.

[7] Gavaskar K, Ragupathy US, Malini V. Design of Novel SRAM cell using Hybrid VLSI Techniques for Low Leakage and High Speed in Embedded Memories. Wireless Personal Communications 2019;4:2311-39.

[8] Gavaskar K, Ragupathy US, Malini V, Proposed Design of 1KB Memory Array Structure for Cache Memories. Wireless Personal Communications, Springer. 2019;4:1-29.

[9] Abed S, Khalil Y, Modhaffar M. High-performance low-power approximate Wallace tree multiplier. International Journal of Circuits and Theory Applications 2018;46:2334-48.

[10] Shanmuganathan R, Brindhadevi K. Comparative analysis of various types of multipliers for effective low power. Microelectronics Engineering 2019;214:28-37.

[11] Prashanthi M, Kyung KK, Minsu C. Novel Area-efficient Null Convention Logic based on CMOS and Gate Diffusion Input (GDI) Hybrid. Journal of Semiconductor Technology and Science 2020;20(1):127-34.

©2020 The Authors. Published by G. J. Publications under the CC BY license.

[12] Subba Rao K, Vassoudevan R. Low Power High Performance Full Adder Design Using Gate Diffusion Input Techniques. Journal of Computational and Theoretical Nanoscience 2020;4:1595-9.

[13] Sukhanya M, Gavaskar K. Functional verification environment for I2C master controller using system verilog, 4th International Conference on Signal Processing, Communication and Networking (ICSCN), Chennai, 2017, pp. 1-6, doi: 10.1109/ICSCN.2017.8085732.

[14] Sukhanya M, Gavaskar K. Design and Verification of LTSSM in USB 3.0 Link Layer using System verilog. 4th International Conference on Signal Processing, Communication and Networking (ICSCN), Chennai, 2017, pp. 1-6, doi: 10.1109/ICSCN.2017.8085689..

[15] Sarkar S, Sarkar S, Atta A, Pahari T, Majumdar N, Mondal S. 9T and 8T FullSubtractor Design using Modified GDI and 3T XOR Technique, Advances in Computer, Communication and Control. In: Biswas U., Banerjee A., Pal S., Biswas A., Sarkar D., Haldar S. (eds) Advances in Computer, Communication and Control. Lecture Notes in Networks and Systems, vol 41. Springer, Singapore. https://doi.org/10.1007/978-981-13-3122-0_49

[16] Tari HT, Zarandi AD, Reshadinezhad MR. Design of a high performance CNTFET-based full adder cell applicable in: Carry ripple, carry select and carry skip adders. Microelectronic Engineering. 2019;215:110980.

[17] Sanjana B, Ragini K. Design of a Novel High-Speed-and Energy-Efficient 32-Bit Carry-Skip Adder. In: Saini H., Singh R., Kumar G., Rather G., Santhi K. (eds) Innovations in Electronics and Communication Engineering. Lecture Notes in Networks and Systems, Vol 65. Springer, Singapore. https://doi.org/10.1007/978-981-13-3765-9_35

****

©2020 The Authors. Published by G. J. Publications under the CC BY license.