Journal of Engineering Science and Technology
Vol. 19, No. 4 (2024) 1235 - 1253
© School of Engineering, Taylor's University

DESIGN OF STORING AND RESTORING ARRAY DIVIDER CIRCUIT USING BINARY DECISION DIAGRAM-BASED ADDER/SUBTRACTOR CIRCUIT

C. SENTHILPARI[1,*], VISHNUPRIYA[1], CHU-LIANG LEE[1],
S.DEIVASIGAMANI[2], ROSALIND[3], G.NARMADHA[4]

[1]Faculty of Engineering, Multimedia University, Cyberjaya, Selangor DE, Malaysia
[2]Faculty of Engineering, Technology & Build Environment,
UCSI University, Kuala Lumpur, Malaysia
[3]School of Mathematics Actuarial and computer sciences, Herriot-Watt University, Putrajaya
[4]Department of Electrical and Electronics Engineering,
Sethu Institute of Technology Tamilnadu, India
*Corresponding Author: c.senthilpari@mmu.edu.my

Abstract

The Binary Decision Diagram (BDD) based circuits are tree-structured, equally sharing the current/power in the cell, which gives reduced power dissipation and increasing speed. The proposed BDD based adder/subtractor circuits are designed and verified in such a way, which trades off the traditional way of full adder/subtractor design, and achieves the required parameters of high speed, low latency, lesser occupying area and low power in the design. The schematic circuits are obtained by using Mentor graphics Silterra 0.13 µm. The proposed adder/subtractor circuit is implemented into a Restoring Array Divider (RAD) and Non-restoring Array Divider (NRAD) circuits for 5G base station application. The proposed full adder gives a power dissipation (32.11 nW), delay (140 ps) and occupying area (67.5 µm^2), which is lower than other reported circuits. The proposed subtractor circuit is compared with the existing circuits, which gives more than 95% improvement in Power dissipation and 17.39% improvement in propagation delay. The layout vs. circuit schematic comparison has been performed for the proposed adder-based RAD and NRAD and evaluated for chip area, propagation delay, and power dissipation. The proposed adder/subtractor-based RAD and NRAD circuits are compared with the results of existing works. The proposed adder/subtractor circuits give 36.02% power dissipation than A, Arya et al. DAXD 99.79% and 99.74% than A. Arya et al. Approximate Divider (ADIV) and Approximate Divider 6 (ADIV6) divider model circuit. The propagation delay and area are improved by 80% in terms of delay and more than 14% in terms of area than the recent report designs.

Keywords: Binary decision diagram (BDD), Delay, Latency, Non-restoring Array Divider (NRAD), Power dissipation, Storing Array Divider (RAD), Throughput.

1. Introduction

In real life, honest two-way communication happens when both parties send and receive information. However, in communication engineering, the actuality of binary signals, binary messages, and information is not readily apparent, and the intangibles must be fashioned through models. Since the dawn of the information age, the Shannon theory has been used widely. Claude Shannon describes the Sender, the Medium of transmission, and the Receiver as basic components of a human communication system [1].

His important facts were emphasised when he recognised the shared fundamental principles between telephone-switching circuits and Boolean algebra. A mathematical performance of the Boolean function could be written as a sum of two subfunctions.

This strategy is sometimes called the "Shannon expansion". Many notations and approaches have been developed to manipulate and notate Boolean functions [2]. An acyclic graph is a type of graph in which each endpoint represents a Boolean function, and the nodes that make up the graph are either 0 or 1. The Arithmetic Logic Unit and the Floating-Point Unit (FPU) are essential microprocessor components. The above-mentioned specialised circuits perform arithmetic operations like adding, subtracting, multiplying, dividing, and calculating parity.

A Standard adder circuit's equation, construct the full Adder's Sum and Carry circuits [3]. A logic '1' connection between the source inputs guarantees a constant "ON" state. The circuit operates by the well-known carry equation. A full adder has 3 binary inputs, such as A, B, and Cin, and 2 binary outputs, called a one-bit full adder (Sum, Cout). Creating and verifying a digital system can be described as a Boolean sequence [4].

A well-performing Boolean function is a data structure that uses time and space effectively during execution. Binary decision diagrams, often known as Boolean function graphs, depict a Boolean function.

The problems of the existing circuit are a common knowledge, and that equivalence checking ensures correct implementation [5, 6]. Due to the many nodes in the design, the existing adder/subtractor circuit gives higher power dissipation and propagation delay. So, this paper overcomes the existing authors circuit and trades off the inaccuracies using BDD based design. This paper aims to evaluate an existing design circuit, draw an appropriate BDD diagram to minimise the node and schematise the circuit using the minimised node.

This paper aims to determine the BDD structure and design using a pass transistor model. Evaluation of the structure is accomplished using mentor graphics. The proposed adder and subtractor circuits are implemented in RAD and NRAD circuits. Therefore, this paper proposes a full subtractor and adder design for RAD/NRAD circuits. These adder/subtractor circuits are designed based on the BDD diagram. The proposed adder/subtractor circuit has fewer transistors than the existing design due to the BDD reduction method, which reduced the power dissipation, reduced the area and increased the speed. According to SDG 9, 5G technology could use the proposed circuit.

This paper is arranged into the following sections. Section 1 involves the introduction, objective, and problem statement. Section 2 involves the related works. Section 3 has expressed the design idea methodology and simulation. Section 4 involves the simulation results and justification for the design.

2. Related Works

Binary Decision Diagrams (BDD) [7] are binary tree diagrams. These data structures were not commonly used for symbolic Boolean manipulation until a set of algorithms worked on them [8]. The data structure known as BDD represents various Boolean operations. The BDD diagram is a directed acyclic network where the vertices represent the two possible outcomes of a binary decision diagram, denoted by the description, logic 0 and logic 1. Both regular and compact versions of ordered binary decision diagrams exist.

One can use ordered binary decision diagrams (OBDDs) to represent and manipulate Boolean functions across a finite domain with relative ease and generality. This is made possible by the diagrams' tree-like structure. Each input variable appears once along each path in an OBDD, making it a binary decision diagram. Each terminal node in the rooted finite directed acyclic graph representing f takes on the value 0 or 1 depending on the values.

Each non-terminal node along the path to the root is labelled with an argument variable and has two successors, indicating the choice between setting the value to 0 or 1. Although the choice of variable ordering is completely arbitrary, it must be kept consistent across all functions being processed in parallel, and this has significant implications for the size of an OBDD representation.

Reduced Ordered Binary Decision Diagrams (ROBDDs) are used in logic synthesis, verification, and VLSI-CAD as a memory-efficient data format for manipulating Boolean functions. Other classes of BDDs have been proposed that can take advantage of both scenarios despite the trade-offs in complexity and memory requirements per node [9].

GF arithmetic circuits can be verified and debugged using cutting-edge forward rewriting-based techniques. This technique permits automatic detection and correction in GF circuits, and it is fault-tolerant, avoiding the polynomial-size explosion that plagues other systems. Boolean functions OBDDs are amenable to many transformations, and each function can be reduced to its form with little effort [10].

The design of adder/subtractor circuits uses multiplexer-based pass transistors. As per the literature review, the proposed adder/subtractor circuits are implemented into RAD and NRAD divider cells. The BDD-based circuits are given lower power dissipation, high speed and less occupying area than existing reporting circuits.

3. Design of RAD and NRAD using BDD

Multimedia 5G application circuits include a storing array divider and a Restoring circuit. The storing array division uses the radix-2 method, which needs to satisfy $2R_{j-1}$-Y≥ 0. The storing circuit has been designed based on a cell structure containing a full adder and multiplexer circuit. The divider inputs are connected to their respective cell inputs. The cell structure served as the inspiration for the creation of the Restoring array divider circuit. The subtractor and multiplexer circuitry are integrated into the Restoring array divider cell. The design of the subtractor and adder circuit description are described below.

3.1 Full subtractor

This subtractor circuit is designed based on BDD nodes and converted to a complementary pass transistor logic (CPL) ladder circuit, shown in Figs. 1 and 2. The inputs are A (the minuend), B (the subtrahend), and C (the subtrahend), and it gives out D (the difference) and B (the subtrahend) (borrow). According to the truth table of subtractors, the borrow logic '1' has to be considered for the design of the subtractor cell. The truth table's most significant bit (MSB) number must only be initiated with the logic 1 values.

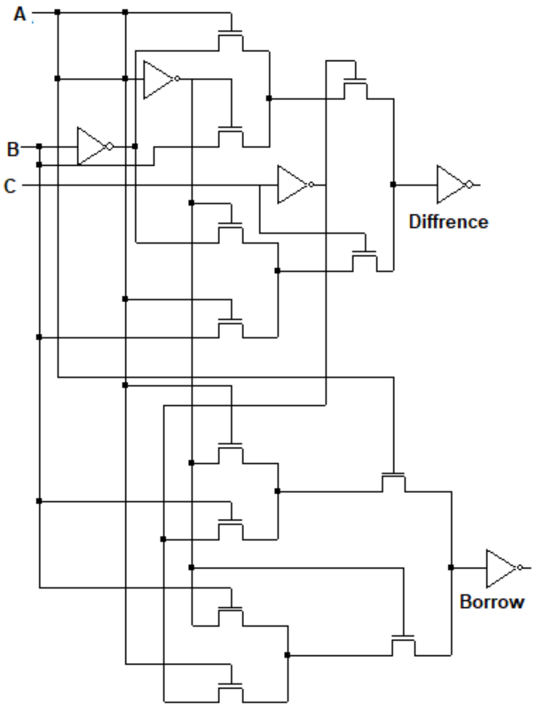

Fig. 1. Circuit diagram for full subtractor.

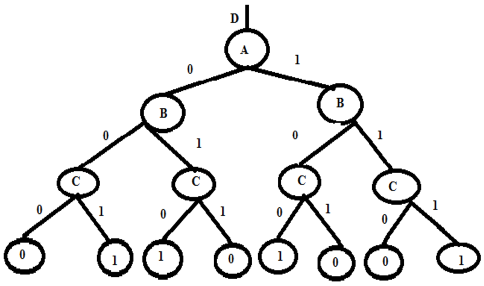

Fig. 2. BDD diagram for subtractor circuit.

The 2:1 multiplexer (MUX) performs equivalent to a Binary Decision Diagram (BDD) node. A Binary Decision Diagram (BDD) variable symbolizes the Multiplexer (MUX) control signal, with the BDD node's out and in branches representing the MUX's inputs and outputs, respectively. A BDD node can be related to a PTL circuit that utilizes two metal oxide semiconductor transistors, as depicted in Fig. 3.

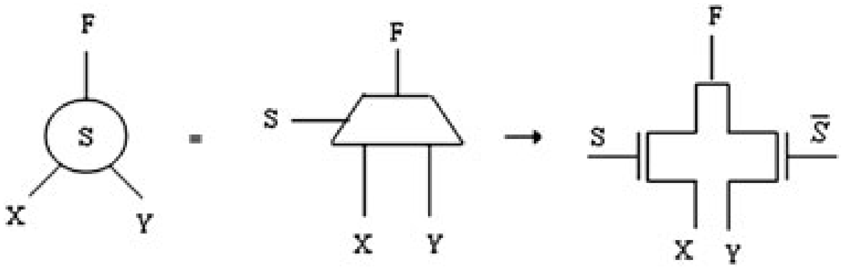

Fig. 3. Conversion BDD into PTL.

The BDD representing the function can exponentially increase in size with inputs. Selecting the appropriate variable ordering mechanism is crucial to prevent this issue. We discovered the optimal Binary Decision Diagram (BDD) by employing symmetric filtering to achieve variable ordering convergence. The Boolean function's BDD representation can be efficiently converted to a MUX network, which can be implemented using pass transistors to save space [11]. The direct correspondence between Binary Decision Diagrams (BDD) and Path Target Library (PTL) streamlines technology mapping.

The PTL implementation based on BDD is compact. Each BDD node can be mapped using a basic 2:1 MUX cell, typically implemented with n-channel metal oxide semiconductor (NMOS) transistors. An NMOS-only design necessitates more transistors compared to a design using both NMOS and PMOS transistors due to the requirement of a signal in both input phases for each binary node. Benchmark circuits utilise complementary metal-oxide-semiconductor (CMOS) inverters for generating control signal complements.

The gate area is competitive. The limited quantity of NMOS transistors in the circuit results in reduced gate capacitance, switching capacitance, power dissipation, and speed, as well as saving active area. The biggest drawback is the voltage drop of one threshold in an NMOS pass transistor when a high logic signal is passed (V_{DD}-V_T for input "1"). Consequently, a lengthy series of pass transistors is required to replenish voltage over time. Buffer-based standard cell libraries address this problem.

Utilising a buffer-based PTL cell library has three advantages. MUX nodes need to address level restoration logic initially. Secondly, buffers reduce signal distortion and voltage fluctuations, and thirdly, the replacement circuit functions similarly to cell-based static CMOS circuits. Binary Decision Diagrams (BDDs) are converted into Pass Transistor Logic (PTL) circuits during technology mapping in order to decrease the number of cells. Greedy covering, dynamic programming, or genetic algorithms can do this. We utilised inverted BDD to decrease both the area and delay.

The subtractor design is based on the formula mentioned in Eq. (1). The CPL circuit has used many inverters differences so that swing restoration would be formed in the output terminal.

$$D_{iff} = A \oplus B \oplus C_{in} \tag{1}$$

$$B = BC + \overline{A}\,(B + C) \tag{2}$$

The proposed circuit has been designed based on the BDD diagram shown in Fig. 2. This BDD diagram is associated with each node. The subtractor circuit has been designed based on 2's complement method of the adder cell, which is indicated in Fig. 2. The BDD variable ordering reduces the node structure and node as per Eqs. (1) and (2). The BDD-based proposed subtractor circuit designed based on subtractor truth table. The BDD diagram changed as logic Boolean identities were simplified and schematised using the mentor graphics tool.

Figure 4 illustrates all logic 1 values used in the design of the CPL-based subtractor in Fig. 1. This subtractor circuit has been schematised employing CPL. It employs 2's complement method for the equation (A+(-B)). The pass transistor logic (PTL) is foremost in IC design logic families. Eliminating unnecessary transistors reduces the number of transistors needed to build logic gates. Instead of switching supply voltages, transistors, pass logic levels between circuit nodes.

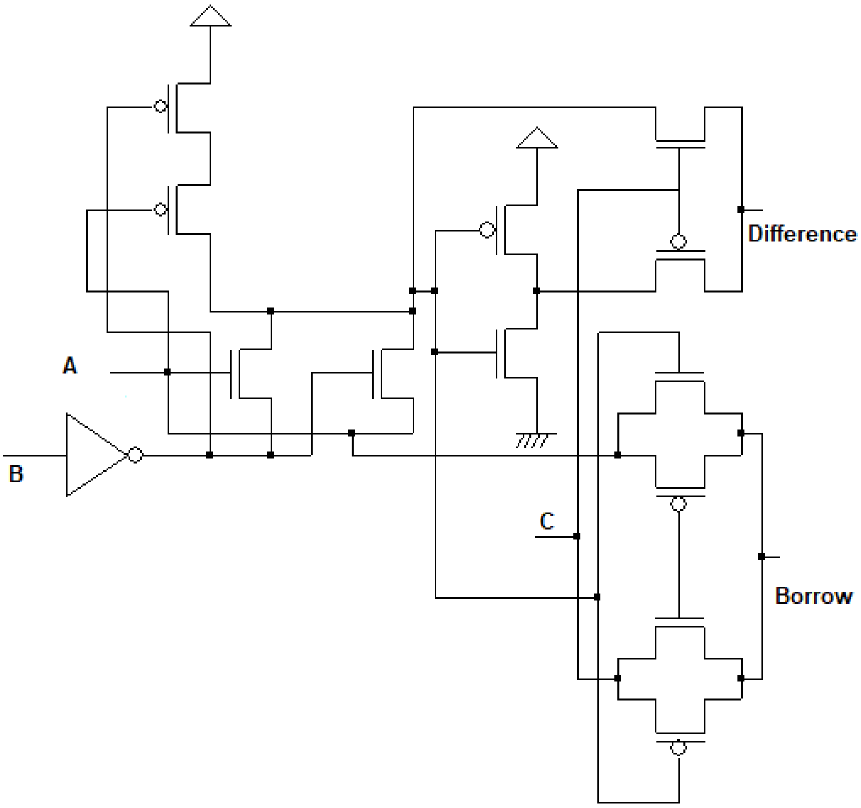

Fig. 4. Proposed subtractor based on binary decision diagram.

3.2 Full adder using BDD

The full-adder circuit for the CPL was built using Eqs. (3) and (4), which used two EXOR gates. Eq. (3) shows how we produced the multiplexing control input (MCIT) method. The modified circuit is shown in Fig. 5, which reduced 2 transistors/gate by the existing method. The BDD based full adder circuit is shown in Fig. 6. The carry output of the full adder is connected to the carry input of the subsequent full adder in the sequence.

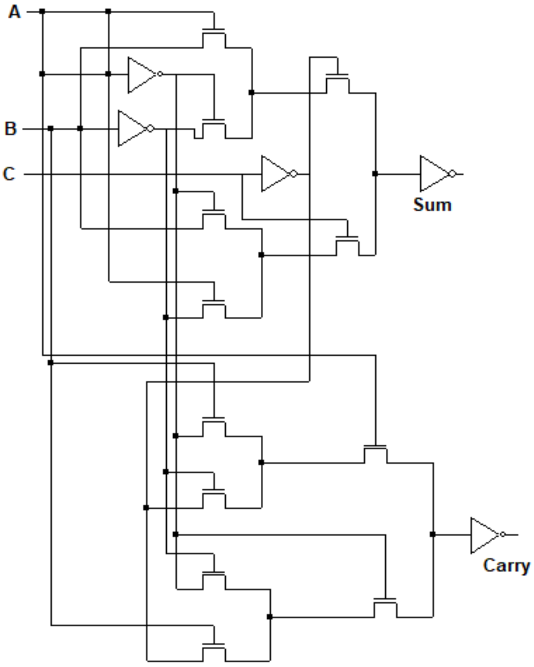

Fig. 5. Full adder circuit diagram.

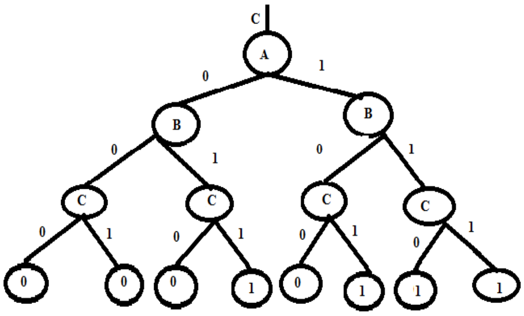

Fig. 6. Binary decision diagram for full adder.

The Boolean expression for Full Adder:

$$sum = A \oplus B \ \oplus C \tag{3}$$

$$Cout = C(A \ \oplus B) + \ A \cdot B \tag{4}$$

3.3 BDD-based proposed adder

Figure 7 depicts the proposed adder circuit designed using a reduced node of BDD, which is schematised bypass transistor logic. The proposed full adder and subtractor circuits are further implemented into NRAD and RAD divider circuits.

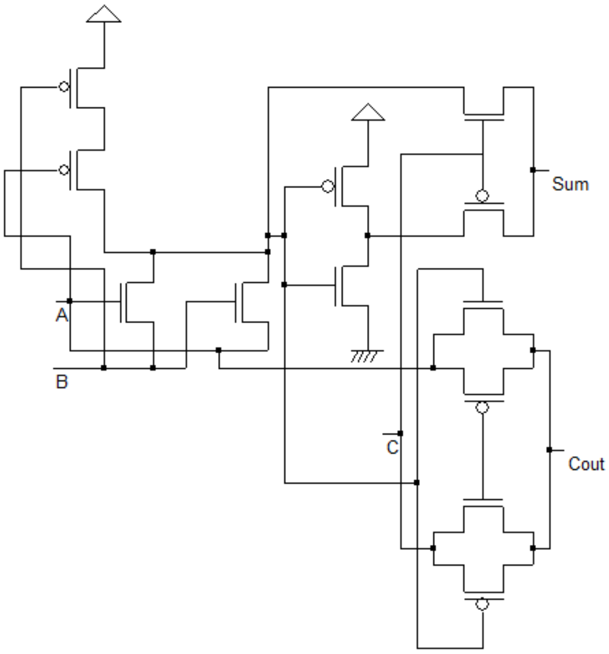

Fig. 7. Proposed adder.

3.4. Restoring Array Divider (RAD) architecture

The RAD circuits are arranged in the cell pattern containing a full subtractor and multiplexer circuit. The quotient is connected with bubbled OR gate architecture and gives a quotient output. The jth bit of the quotient of the binary representation of the number Z (zj) is equal to qj when using the radix technique. When Rj′ is less than zero, the retention technique (Y) is implemented. To increase efficiency, we can use 2Rj - 1 as Rj by maintaining Rj - 1 instead of adding Y to Rj′ when Rj′ is less than 0.

CPL is utilised during the manufacturing process of both the Restoring and the storing array divider circuits. The CPL design for the Storing array divider is built on top of subtractor cells as its foundation. A full subtractor based on CPL is included in the RAD cell, and a multiplier with 2 inputs.

3.4.1. Non-restoring Array Divider (NRAD)

A remainder correction circuit is required to get the right remainder while using a Restoring division. If the final piece of the quotient is 1, the remainder is correct; otherwise, we need to add the divisor back to the partial remainder to get the right answer. Besides its application as an array divider, this circuit is a controlled adder.

First, the registers must be shifted, then the adder's signals must be allowed to travel, the next quotient digit must be calculated and stored, and finally, the trial difference, if necessary, must be recorded. The lengthening of the clock cycle is due to later events in the same cycle are dependent on earlier ones.

3.4.2. 7×4-bit Restoring Array Divider

In binary Restoring division, the dividend and the ensuing partial product are calculated by subtracting or adding right-shifted iterations of the divisor. The carry-out from the partial remainder decides the next iteration's quotient and whether the shifted divisor is added or subtracted. A RAD is often implemented as a two-dimensional iterative array within a cell to shorten the number of division cycles.

The array is built from controlled Adders, a type of basic cell. The restoring array divider basic cell is based on one EX-OR (XOR) followed by a full adder circuit. The divisor and actual input are fed directly to the full adder circuit and exclusive circuits. The XOR circuit has checked the divider and actual input, clearly showing the quotient values.

For a complete, 7×4-bit Restoring Array Divider shown in Fig. 8, it has 13 inputs and 8 outputs, namely, d_0, d_1, d_2, d_3, d_4, z_0, z_1, z_2, z_3, z_4, z_5, z_6, z_7, can for NRAD inputs while for outputs q_0, q_1, q_2, q_3, r_0, r_1, r_2, r_3. There are 4 rows consisting of 4 NRAD symbols circuits connected. Each cell of NRAD has a full adder connected to 2 input XOR circuits.

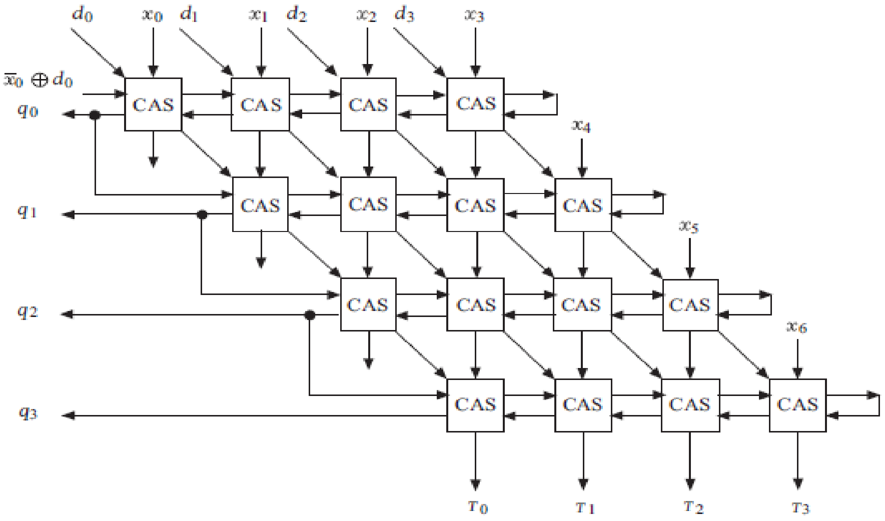

Fig. 8. Restoring divider architecture.

When D is a divisor, and A is the dividend, the quotient Q can be calculated using the binary division method [11, 12]. At each level of the procedure, D is either used to partition A into groups of bits or not. Bits are divided by the divisor if and only if the divisor's value must be less than input bit values. Therefore, the divider quotient can only be either logical 0 or logical 1. A full adder and an XOR gate with 2 inputs comprise a Restoring divider cell.

The upper half of Fig. 9 comprises four stages of cascaded full adders. The stages of dividend bits dictate the placement of the number of levels, while the stages are connected by full adder seriously, which determines stages of divisor bits. Using four serially connected full adders, the seven dividend bits are divided into four stages, and the four divisor bits are divided into four stages.

The first stage of the flowed full adders uses the four bits of the dividend considered the most significant. Subsequent stages of the cascaded full adders use one more bit of the dividend, and the three bits are carried over from the stage before it (thus the number four).

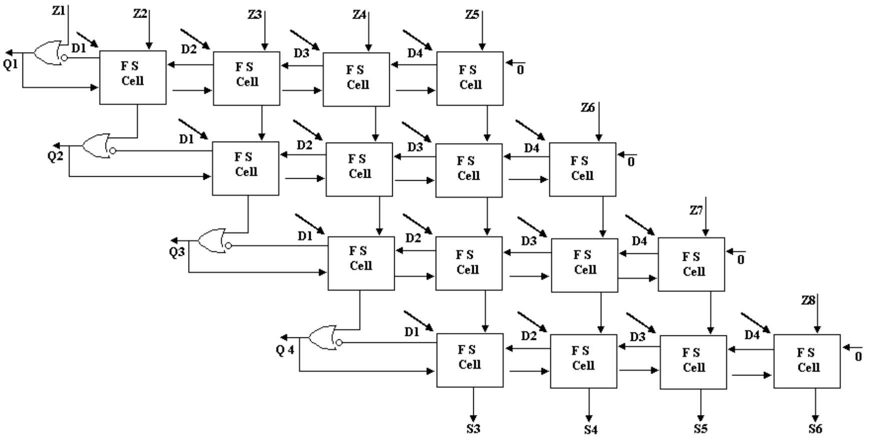

Fig. 9. Restoring array divider.

3.5. Performance analysis

Mentor Graphics is used to simulate the proposed design, and the results are evaluated regarding power dissipation, area, and propagation delay. Table 1 contains the Area and Power Dissipation (ADP) product and Power and Delay (PDP) product values for adder, subtractor, storing and Restoring array dividers.

The proposed adder and subtractor circuits use 12 transistors. The proposed BDD-based adder circuit has a regular tree structure and the electron flow in between transistors is in a regular manner. So, the power dissipation in the proposed adder is 32.11 nW, and the propagation delay is 140 pS. Due to fewer transistors in the adder circuit, the occupying area is 67.5 μm^2. The inputs and processing time are low because the transistors are arranged regularly. So, the latency of the full pattern input is 2.65 ns, and the throughput of the full adder is 377.35 Mbps.

The proposed subtractor circuit has reduced the nodes using BDD. So, the subtractor circuit's power consumption is reduced, which is 29.1 nW, with delay of 190 ps. The connection wire in the inverter is increased, so the occupying area is slightly increased at 69.75 μm^2.

The proposed adder and subtractor-based RAD and NRAD cells are arranged in the 7×4-bit divider circuits. The RAD circuit with an adder requires 312 transistors, whereas the NRAD circuit implemented with a subtractor requires 512 transistors. The adder and subtractor cells utilised in the design each have a voltage of 1.5 V, while the RAD and NRAD cells each have a voltage of 2 V. The cells that have been suggested and designed will operate at a frequency of 2.3 GHz.

Table 1. Simulation results of the adder, subtractor, RAD and NRAD circuits.

Circuit	Power consumption (nW)	Delay (ns)	ADP (µm^2-ns)	PDP (fJ)	Area (µm^2)	Latency ns	Throughput (Mbps)
Full adder	32.11	0.14	9.45	4.49	67.5	2.65	377.35
subtractor	29.1	0.19	13.25	5.52	69.75	2.69	371.74
RAD	57.8	0.42	528.7	24.27	1258.9	2.92	342.46
NRAD	96.6	0.48	967.1	46.36	2014.8	2.98	333.57

The BDD-based Adder subtractor circuits are compared with Arya et al. [16] ESC and CFSC subtractor circuits [13, 14]. The proposed subtractor circuit performs better than the Arya et al. (2021) ESC and CFSC subtractor circuits, due to the reduced design tree and perfect balance tree design, reducing the critical path and equal power distribution [15]. The performance of the suggested subtractor circuit is superior to that of the ESC and SFSC subtractor circuits by more than 90 percentage points in terms of power dissipation and by 33.33 percentage points in terms of propagation latency.

However, the available space has had to be reduced to accommodate the large number of transistors required for the 1-bit subtractor cell [16]. The proposed adder and subtractor circuits are implemented in NRAD and RAD divider circuit. Reddy et al. [17] divider circuit compared with our proposed model in terms of power dissipation and propagation delay, which our proposed circuit gives 83.69% and 99.04% improvement in terms of power dissipation and propagation delay than Reddy et al. [17] divider circuit.

The comparison, as shown in Table 3, is made between the proposed adder and subtractor circuit and the Arya et al. [16], DAXD, ADIV, and ADIV6 divider circuits. Our proposed adder and subtractor circuit we described, performs significantly better than the one developed by Arya et al. [16] . BDD has been used as the foundation for the design of the suggested adder and subtractor circuit. Therefore, the circuit tree structure is balanced and normally distributes electrons.

Table 2. Comparison with existing authors.

Reference	Power nW	% of Improvement	Delay	% of Improvement	Area µm^2	% of Improvement
Proposed Adder	32.11	----	0.14 ns	----	67.5	----
Proposed subtractor	29.1	----	0.19 ns	----	69.75	----
Proposed NRAD	96.6	----	0.48 ns	----	2014.8	----
Proposed RAD	57.8	----	0.42 ns	----	1258.9	----
Reddy et al. [17]	592.3	83.69	50.14 ns	99.04		
ESC Subtractor [16]	1.5 µW	97.85	0.21	33.33	30 µm^2	-56.98
CFSC Subtractor [16]	1.84 µW	98.41	0.23	17.39	34 µm^2	-51.25
DAXD divider 14 bit [16]	90.35	36.02	3.77	88.85	2354	14.40
ADIV divider 6 [16]	46.06	99.79	5.92	91.89	2446	17.62
ADIV-6 [16]	37.52	99.74	5.92	91.89	1,262,817	99.99
Jiang et al. [19]	38.06 µW	99.84	2.36	82.20	844.1	-49.14

Therefore, there will be no dissipation during charge sharing, and the RAD and NRAD circuits will have a reduced critical path. The adder and subtractor circuit we described performs significantly better than the one developed by N. Arya. BDD has been used as the foundation for the design of the suggested adder and subtractor circuit. Therefore, the circuit tree structure is balanced and there is a normal electron flow in the circuit.

Therefore, there will be no dissipation during charge sharing, and the RAD and NRAD circuits will have a reduced critical path. The Arya et al. [16] circuits are based on CMOS design, which contains P-MOSFET model, and NMOSFET model. So, the electron flow would be smooth way, but it occupies more transistor, which increased the logical effort. The Arya et al. [16] circuit gives more power dissipation and lower speed than our proposed circuit.

The proposed subtractor based divider circuit further compared with Jiang et al. [19] adaptive approximation approach circuit, which designed reduced width divider shift the input method. This method used for reduced the input pattern and finally corrected the error by truncated and compute the 2 k/k division method. Our proposed method has designed node reduction, straight forward design. Our proposed circuit used the exor gate as parity check, which reduced the error in the bit. So, our proposed circuit gives 99.84% power reduction, and improved speed 82.20% due to a smaller number of transistor used for the design cell.

3.6. Monte-Carlo method analysis for NRAD/RAD

A series of numbers can be broken down into component frequencies using the digital Fourier Transform (DFT) [20, 21]. The sequence of infinite instructions for the probability distribution format can be roughly categorised as either time decimation or frequency decimation. The Monto Carlo method of the Mentor graphics software provided the Fast Fourier Transform (FFT), which will provide an analogue simulation of the RAD and NRAD circuit's inputs and outputs.

The Monte Carlo for RAD and NRAD are shown in Figs. 10 and 11, respectively. Figure 10 shows the relationship between frequency (GHz) and voltage (mV) for node 0. The highest value of frequency is 39 MHz at 188.25 mV. A Binary Decision Diagram can also be implemented in approximate arithmetic, which can be used for image processing applications.

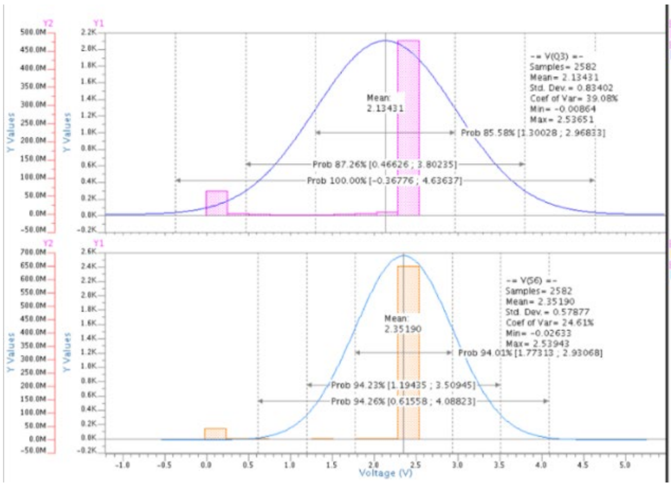

Fig. 10. Monte Carlo simulation for RAD.

A binary decision diagram based adder/subtractor implemented RAD and NRAD circuits has analysed the Monte Carlo simulation. The Monte Carlo simulation is based on histogram and Gaussian methods. The histogram method analyses a rough sense of density, which is the underlying distribution of data. The value of the distribution measures the probability of density (Fig. 11). The means V(Q3) sample quality is 2582, and the standard deviation is 0.83402 V. The pdf coefficient of variation is 39.08%.

The Eq. (5) has measured the probability distribution format.

$$P_{r[a\leq x\leq b]} = \int_a^b f_X\,(x)dx \tag{5}$$

The FX is the cumulative distribution function of X.

The Gaussian distribution of the function has been formulated using Eq. (6).

$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\left(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right)} \tag{6}$$

The Gaussian function was improved by composing the expositional function with a concave quadric function.

The pdf median is 87.25%, which shows the perfect design and gives better results than other existing circuits. If a circuit needs to generate a random sample from a distribution quickly, the inverse transformation method is the way to go [22]. The probability density function is maximised since our proposed circuit uses a normal sampling rate and signal distribution. The same design was analysed regarding V(S6) adder-based cells and used 2582 samples. The mean value of this design is 2.35190, and the probability of the design has improved by 94.01% Adder-based design. The variation voltages are changed between 1.5 V to 3 V.

The recommended adder and subtractor system Divider circuit The Gaussian generator has an input that is connected to either the first output of the first digital corrector or the first output of the second digital corrector. The sequence bit generator includes an adder configuration that produces a sum of logical '1' sequential bits. The signal received from the corrector is utilized to calculate the difference between the difference and borrow values of the Sum. An arrangement that divides the difference by a standard deviation to produce a Gaussian deviate at the output of the Gaussian generator.

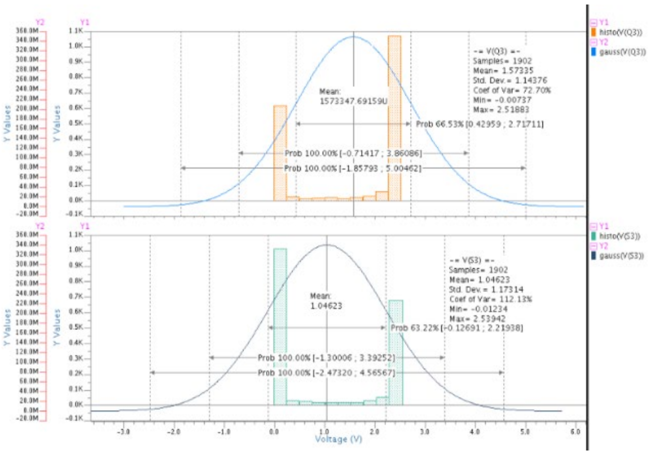

Fig. 11. Monte Carlo simulation for NRAD.

The layout of RAD and NRAD is shown in Figs. 12 and 13. The layout has been drawn using mentor graphics using a schematic-driven layout (SDL). The layouts are analysed using design rule check (DRC), which verifies all the polygons and layers [20]. The adder and subtractor circuits are implemented into RAD and NRAD cell structures. According to transistor metals are arranged as per scaling theory. The transistor metals include the polysilicon, N + metal, P + metal wires and via.

The layouts of the proposed adder and subtractor based divider circuit is shown in Figs. 12 and 13. The design must be checked to align with the metals. The proposed adder and subtractors are reductant to the nodes of logical efforts. Then, the pass transistors are arranged in a regular pattern. Hence, the metal layer and input-out layers are placed regularly. Therefore, the inputs are equally distributed in the electron flow. The outcome is, the power consumption is lower, and an increase in speed in the proposed adder/subtractor-based NRAD and RAD divider circuit.

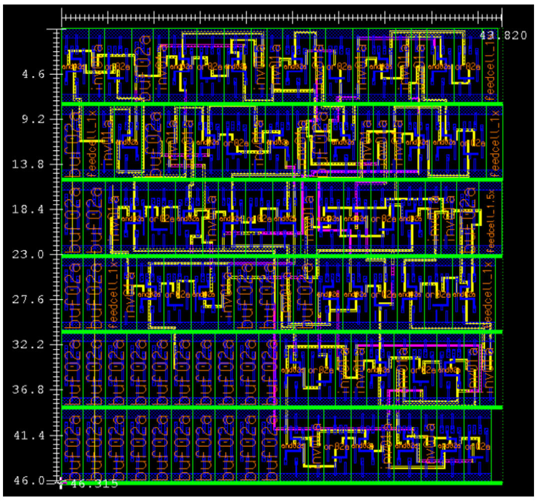

Fig. 12. The layout of RAD.

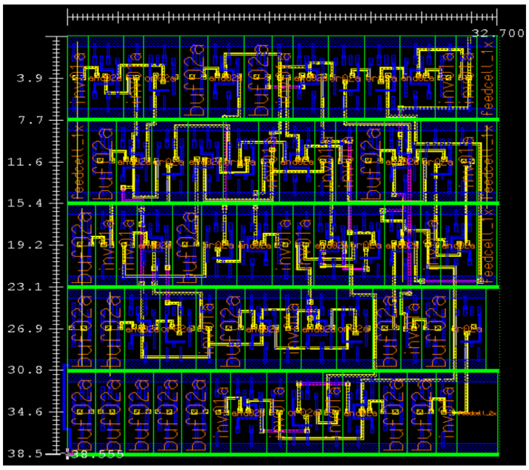

Fig. 13. The layout of NRAD.

4. Results and Discussion

The proposed subtractor adder circuits are simulated using the Mentor Graphics design tool. The signal to perform rapid shifting is transmitted in the proposed RAD/NRAD circuit based on an adder/subtractor. Using the Silterra CAD application, the Non-Restoring and Restoring array 7×4-bit divider circuits are simulated. The outputs between 1111111 1000 and 111111 11 11 are examined.

The simulation timing diagram is visually represented in Figs. 15 and 16, providing unambiguous evidence of an improved adder/subtractor circuit that has been proposed. The structure of the suggested subtractor is that of a balanced tree [23]. As a result, the inputs are in equilibrium, and the timing diagram (Fig. 13) shows no losses in delivering the outputs.

As a result of the `well-designed -tree structure, the output of the subtractor reveals that both the difference and the borrow produce flawless outcomes. The logic function is moved up to the transistor level, and the logic 1 input is distributed evenly between the electron and themselves. The truth tables, therefore, provide the outputs [24].

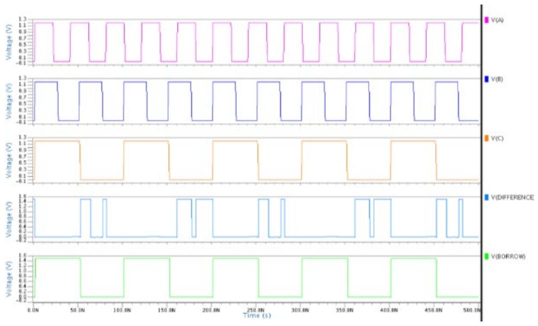

Fig. 13. Full subtractor timing diagram.

The full adder circuit is designed using Eqs. (2) and (3). The full adder logic transition of the electron charges, sharing with the transistor, is good. When A=1, B=0 and Cin =0, The logic transition has a skew problem, which can be corrected by adding the capacitor in the output node because the charge carrier has charged (stored) and discharged the carrier instantly, as indicated in the timing diagram. (Fig. 14).

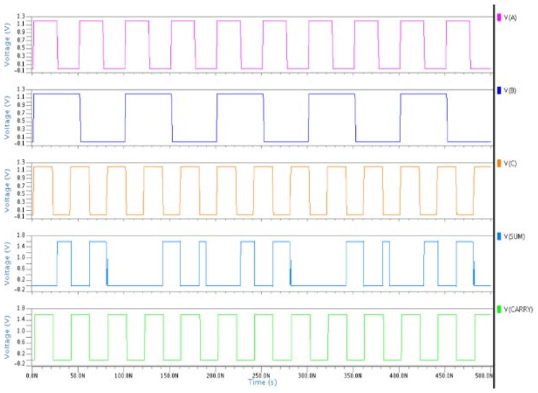

Fig. 14. Full adder – timing diagram.

The timing diagram of the RAD circuit is shown in Fig. 15. The restoring array divider cell is based on the full subtractor followed by a 2:1 multiplexer circuit. Here the remainder input is used as multiplexer selection input. The dividend and divisor input are fed into the restoring array divider cell. The proposed subtractor circuit is well suited in the RAD block, clearly shown as a perfect output in the timing diagram. The output of the quotient and borrow out are shown as perfect voltage out, which means it gives better voltage out, lower power dissipation and high speed.

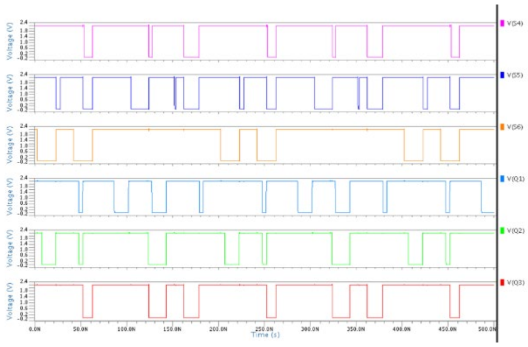

Fig. 15. RAD – timing diagram.

The stated adder-based NRAD circuit provides more effective results as a result of the BDD-based model design. The adder-based NRAD cell provides a divisor, dividend, and quotient carry-in to a full adder circuit. The full adder circuit based on BDD decreases power dissipation and enhances speed because of the tree-like arrangement of the PTL and its typical electron flow.

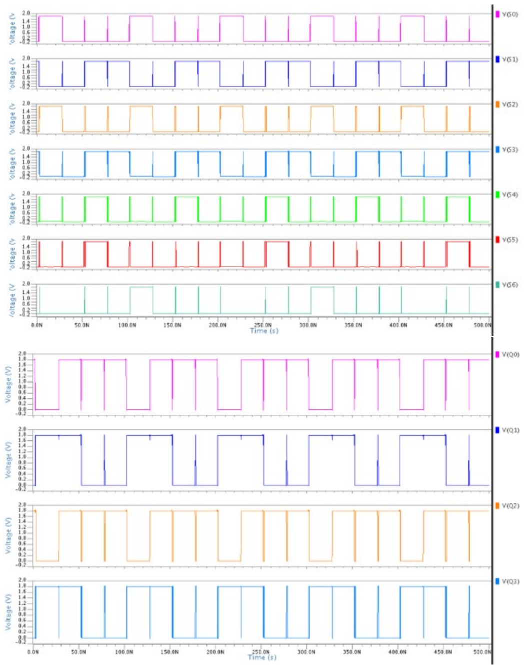

Fig. 16. NRAD – timing diagram.

5. Conclusions

The adder and subtractor circuit have been proposed and implemented in the design of the array divider circuit that is employed for storing and restoring arrays. The proposed adder and subtractor circuit were schematised using the mentor graphics CAD tool. The performance of the proposed adder and subtractor circuit, which was implemented in both the NRAD and RAD circuits, is superior to that of the work done by others published with their respective circuits. The designed circuit can view the design trade-offs, such as power dissipation, chip size, and delay due to the regular arrangement of transistor and BDD models.

As per, Layout Vs Schematic (LVS), the divider circuits gave low power consumption (57.8 nW and 96.6 nW) and lower propagation delay (0.42 ns and 0.48 ns). The divider circuits are further analysed in terms of bit pattern/cycle, the circuits give 2.92 ns and 2.98 ns latency. The proposed adder and subtractor circuits RAD and NRAD can ascertain the relationship between several metrics, including capacitance, power dissipation, V_{DD}, and I_{DD}. According to the findings of the performance study, it has been discovered that utilising different data supplies can lead to superior outcomes.

The PTL based circuit gives better performance only if few numbers of transistors used lower input bit pattern circuit (2^8). The higher bit pattern gets logical skew problem. The lower power consumption and speed divider circuits can be used in the 5G router circuit.

Abbreviations

DRC	Design rule Checker
LVS	Layout versus Simulation
NRAD	Non-restoring Array Divider
RAD	Restoring array Divider
SDL	Schematic-Driven Layout

References

1. Biswal, P.K.; and Biswas, S. (2015). A Binary Decision Diagram based on-line testing of digital VLSI circuits for feedback bridging faults. *Microelectronics Journal*, 46(7), 598-616.
2. Krauss, R.; Goli, M.; and Drechsler, R. (2023). Efficient binary decision diagram manipulation by reducing the number of intermediate nodes. *Proceedings of the* 2023 26th *International Symposium on Design and Diagnostics of Electronic Circuits and Systems (DDECS)*, Tallinn, Estonia.
3. Hsiao, S.-F.; Yeh, J.-S.; and Chen, D.-Y. (2002). High-performance multiplexer-based logic synthesis using pass-transistor logic. *VLSI Design*, 15(1), 417-426.
4. AnanthaLakshmi, A.V.; and Sudha, G.F. (2017). Design of a reversible floating-point square root using modified non-restoring algorithm. *Microprocessors and Microsystems*, 50, 39-53.
5. Hossain, M.S.; Hasan, M.M.; Rahman, M.M.; and Hossain, A.S.M.D. (2010). A new design technique of reversible gates using pass transistor logic. *Journal of Engineering Science*, 01(02), 25-32.

6. Lahiri, S.K.; and Wang C. (2020). *Automated and scalable verification of integer multipliers.* In Temel, M.; Slobodova, A.; and Hunt, W.A. (Eds), *Computer Aided Verification. Springer Nature Switzerland AG, Inc.*

7. Drechsler, R. (2021). Polynomial circuit verification using BDDs. *Proceedings of the* 2021 5th *International Conference on Electrical*, *Electronics*, *Communication*, *Computer Technologies and Optimisation Techniques (ICEECCOT)*, Mysuru, India, 49-52.

8. Wilhelm, D.; and Bruck, J. (2008). Stochastic switching circuit synthesis. *Proceedings of the* 2008 *IEEE International Symposium on Information Theory*, Toronto, ON, Canada.

9. Kumar, D.; Saha, P.; and Dandapat, A. (2017). Hardware implementation of methodologies of fixed point division algorithms. *International Journal on Smart Sensing and Intelligent Systems*, 10(3), 630-645.

10. Hsia, S.-C.; and Chen, K.-C. (2023). A 12T low-power full adder cell with a novel dynamic circuit. *International Journal of Electronics Letters*, 168-177.

11. Babar, J.; Jiang, C.; Ciardo, G.; and Miner, A.S. (2019). Binary decision diagrams with edge-specified reductions. *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, 11428, 303-318.

12. Clarke, E.M.; Henzinger, T.A.; Veith, H.; and Bloem, R. (2018). *Binary decision diagrams*. In Bryant, R.E. (Ed.), *Handbook of model checking*. Springer International Publishing AG, 191-217.

13. Bhuvaneswari, T.; Soong, L.W.; Singh, A.K.; and Prasad, V.C. (2014). Design of binary decision diagram (BDD) optical adder. *Proceedings of the* 2014 *International Conference on Advances in Engineering & Technology Research* (*ICAETR*-2014), Unnao, India.

14. Bhagyalakshmi V.; and Mantri, D.B. (2013). A BDD-based design of an area-power efficient asynchronous adder. *International Journal of Electronics and Communication Engineering*, 6(1), 93-103.

15. Gassoumi, I.; Touil, L.; and Mtibaa, A. (2021). An efficient design of QCA full-adder-subtractor with low power dissipation. *Journal of Electrical and Computer Engineering*, 2021, 8856399.

16. Arya, N.; Soni, T.; Pattanaik, M.; and Sharma, G.K. (2020). Bit significance based reconfigurable approximate restoring dividers and square rooters. *Microelectronics Journal*, 104(3), 104861.

17. Reddy, J.N.; Tiwari, M.; and Reddy, V.P. (2021). Power efficient two transistor exclusive or gate for full adder usinggdi in 45NM. *Turkish Journal of Computer and Mathematics Education*, 12(2) 1342-1347.

18. Senthilpari , C. (2014). A low-power and high-performance radix-4 multiplier design using a modified pass-transistor logic technique. *IETE Journal of Research*, 57(2), 149-155.

19. Jiang, H.; Liu, L.; Lombardi, F.; and Han, J. (2019). Low-power unsigned divider and square root circuit designs using adaptive approximation. *IEEE Transaction on Computers*, 68(11), 1635-1646.

20. Senthilpari, C.; Diwakar, K.; Deivasiganmani; Velrajkumar, P.; and Ayavoo, R. (2012). Power efficiency top-down ALU for error correction and detection

circuit. *Proceedings of the* 2022 3rd *International Conference on Communication, Computing and Industry* 4.0 (*C2I*4)*,* Bangalore, India.

21. Shivakumar, V.; Senthilpari, C.; and Yusoff, Z. (2022). Test power and area optimised logic built-in self-test with higher fault coverage for automobile SoCs. *Microelectronics Journal*, 124, 105430.
22. Shivakumar, V.S.; Senthilpari, C.; and Yusoff, Z. (2021). Design a 1.9 GHz low-power LFSR circuit using the reed-solomon algorithm for pseudo-random test pattern generation. *International Journal of Integrated Engineering*, 13(6), 220-232.
23. Tiankai, S. (2020). Analysis *and verification of arithmetic circuits using computer algebra approach*. PhD Thesis, Department of Electrical and Computer Engineering, University of Massachusetts Amherst.
24. Narmadha, G.; Deivasigamani, S.; Balasubadra, K.; and Selvaraj, M. (2022). A low power and high speed approximate adder for image processing applications. *Journal of Engineering Research*, 10(1A), 150-160.