The CSI Journal on
Computer Science and Engineering
Vol. 17, No. 1, 2019
Pages 9-20
Regular Paper

Novel Designs of Fast Parity-Preserving Reversible Vedic Multiplier

Ehsan PourAliAkbar[1] Keivan Navi[2] Majid Haghparast[3] Midia Reshadi[1]

[1] Department of Computer Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran

[2] Faculty of Computer Science and Engineering, Shahid Beheshti University G. C., Tehran, Iran

[3] Department of Computer Engineering, Yadegar-e-Imam Khomeini (RAH) Shahre Rey Branch, Islamic Azad University, Tehran, Iran

Abstract

Reversible logic is a new technology that is considered as an essential requirement for the design of quantum computers. In the calculation unit of computers, multiplication is one of the most frequently used operations. In this paper, we propose new optimized algorithms to design a parity-preserving reversible Vedic multiplier. Three approaches for designing optimized reversible Vedic multiplier circuits are proposed which are better than the existing circuits in terms of quantum cost, number of garbage outputs, number of constant inputs, and other criteria. The proposed reversible Vedic multipliers can be generalized to produce parity-preserving n*n reversible multiplier. We have also achieved some relations which can calculate the quantum cost, the number of constant inputs, and the number of required garbage outputs for the proposed Vedic circuit of any dimension. We have shown that our proposed reversible Vedic multiplier in n*n scale is the best compared to the existing Vedic multipliers.

Keywords: Reversible logic, Reversible array multiplier, Vedic multiplier, parity-preserving, quantum computing.

1. Introduction

One of the most important factors in circuit design is reducing power dissipation. So far, many works have been done in the design of VLSI circuits. The main purpose of all these works was to reduce energy waste. Reversible logic is used as one of the modern technologies in circuit design and performs more efficiently than irreversible circuits in designing different circuits without energy dissipation. Instead, in reversible logic, we have to deal with design difficulties and other kinds of complexities that make reversible circuit designing more difficult than irreversible circuit designing. That is because fan-out and feedback are not allowed in reversible circuits. Reversible logic circuits can be used in various designs such as DNA computations, nanotechnology, decoding devices [1], quantum computing [2], and low-power circuits [3]. In reversible logic, the numbers of inputs and outputs are equal and there is a one-to-one relationship between inputs and outputs. This means that output values can be achieved from input values and input values can be recovered from output values, where input vector is (Iv), and output vector is (Ov); they are defined as Eq. (1) [4-6].

$$Iv = (I_1, I_2, I_3, \ldots, I_n) \quad (1)$$
$$Ov = (O_1, O_2, O_3, \ldots, O_n)$$

To have a parity-preserving circuit, the approach is to make use of parity-preserving gates. To use this property, there must be a parity between input and output so that XOR input values are equal to XOR output values in every state [7].

One of the most important operations used in computing systems is multiplication. High-speed multipliers have always been required in computing systems. Improvement in multiplier circuits has a direct impact on the design of different electronic circuits. The design of multiplier circuits in reversible logic can be useful in quantum computers [8].

The Vedic multiplier is identified as an Indian multiplier circuit which performs multiplication by dividing the multiplier circuit into smaller parts and synthesizing their results. Vedic computing is an Indian computational system that has a single computational technique with 16 Sutra. The multiplication Sutra operation, among these 16 sutras, is Urdhva Tiryakbhyam Sutra, which means vertical and crosswise.

In this paper, all of our proposed circuits are parity-preserving. Reversible circuits that are Parity-Preserving are capable of detecting errors. So far, much research has been done on parity-preserving reversible circuits which can be referenced to the papers presented in [7, 9-17].

Many multiplier circuits have been presented in reversible logic till now [9, 10, 12, 18-23]. Some of these circuits have been also parity preserving. Moreover, there have been some articles on Vedic multipliers, such as [9, 13, 14, 24-33]. The papers [29, 30] provide reversible Vedic multiplier circuits that are not Parity-Preserving and are not comparable to our proposed work. In [31], the efficiency of reversible Vedic multiplier circuits is discussed. In the paper presented in [32], an irreversible Vedic multiplier circuit is presented. Also, in the article [33], the authors have designed a Vedic multiplier circuit by Routing Rearrangement which is irreversible. The authors of any of the above articles did not provide a parity-preserving reversible circuit. Therefore, these circuits cannot be compared with our proposed designs. But among which [13, 14] are parity preserving.

In our proposal, we have initially suggested an optimized algorithm for designing a reversible Vedic circuit. Thus, we have extended this algorithm to a general algorithm with any dimension. In the next step, we have designed a 2*2 reversible Vedic circuit and have introduced the required circuits to present our proposed 4*4 reversible Vedic multiplier circuit. Then, making use of our proposed algorithm, we suggest an n-bit reversible Vedic multiplier circuit that can perform multiplication in any dimension. Next, we present some relations and formulas for calculating the numbers of constant inputs, garbage outputs, and the quantum cost of the proposed circuit, which can help compute the values for these items of the proposed circuit in any dimension.

In Section 2, we have described some basic concepts of reversible logic and Vedic multiplier. In Section 3, a review of the previous studies will be presented. In Section 4, we present our proposed algorithm and circuits. The evaluation of proposed circuits and their comparisons with previous circuits will be carried out in Section 5. Finally, the conclusion will be presented in Section 6.

2. Basic Concepts

In this section, we define and analyze the concepts of reversible, reversible parity-preserving gates, and the structure of the Vedic multiplier circuit.

2.1. Reversible logic

A circuit is reversible if there is a one-to-one relationship between input and output values as well as the physical reversibility. In reversible circuits, output values can be achieved from input values and input values can be recovered from output values.

One of the features that can be added to a reversible gate or circuit is the parity-preserving property. A reversible gate is parity-preserving if the input and output XOR values are equal. This is defined as Eq. (2).

$$I_0 \oplus I_1 \oplus \ldots \oplus I_{n-1} = O_0 \oplus O_1 \oplus \ldots \oplus O_{n-1} \quad (2)$$

If a reversible circuit is made of parity preserving reversible gates, the whole circuit will also be parity-preserving. In reversible logic, there is also another criterion called constant input. In a reversible circuit, inputs having their values unchanged at either 0 or 1 are called constant input. These constant inputs are added to irreversible gates to convert them into reversible gates [34]. On the other hand, some outputs in a reversible circuit are left unused during the calculation; these outputs are called garbage outputs [35]. Quantum cost is another measure to evaluate the efficiency of reversible circuits. In reversible logic, the quantum cost of 2*2 circuits is always 1. The quantum cost of circuits larger than 2*2 is calculated as the sum of the quantum cost of each 2*2 and 1*1 circuit used in the circuit. There are also two quantum gates called V and V+ in reversible logic. The V gate is known as 'the square root'. Eq. (3) describes the functionality of the V and V+ gates:

$$V*V = V^{+}*V^{+} = NOT \quad (3)$$
$$V*V^{+} = I$$

2.2. Reversible gates

Up to now, some different gates in reversible logic have been presented. Generally, reversible gates can be divided into two different categories. The first category is non-parity-preserving gates; this kind of gates are designed based on some properties but lack the parity-preserving property. The second category is parity-preserving gates; if we only use parity-preserving gates to design reversible circuits, these kinds of gates add the parity-preserving property to the circuit. Toffoli [36], and HNG [37] gates are two examples of non-parity-preserving gates presented so far.

Many parity-preserving reversible gates have been presented in the literature, including FRG [38], F2G [15], and NFT [16]. Each one of these gates can be used in a specific place and for a particular purpose.

Feynman Double gate (F2G) is a parity-preserving gate with 3 inputs and 3 outputs. The inputs and outputs of the F2G gate are as shown in Eq. (4).

$$Iv = (A, B, C) \quad (4)$$
$$Ov = (P = A, Q = A \oplus B, R = A \oplus C$$

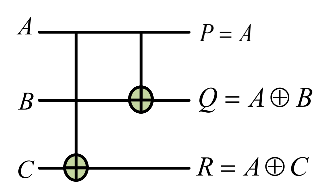


Fig. 1: parity-preserving reversible F2G gate.

Fig. 1 shows the reversible F2G gate. The F2G gate is used to produce Fan-out in the circuits. If the inputs B and C are set to zero, the value of the first input (A) is obtained from all three outputs. The quantum cost of the F2G gate is 2.

New fault tolerant gate (NFT) is a parity-preserving gate. It has 3 inputs and 3 outputs which are as Eq. (5).

$$Iv = (A, B, C) \quad (5)$$
$$Ov = (P = A \oplus B, Q = B'C \oplus AC', R = BC \oplus AC'$$

Fig. 2 shows the NFT gate. The NFT gate is used to generate AND operation in parity-preserving circuits. If input A of this gate is set to zero, then we have the AND operation for inputs B and C in output R. The quantum cost of the NFT gate is 5.

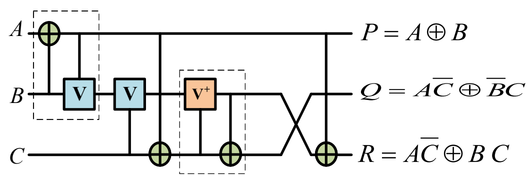


Fig. 2: parity-preserving reversible NFT gate.

2.3. Vedic Multiplier

There are different methods to perform multiplication operations. The Vedic multiplier is one of these methods. It is one type of Indian multiplier that uses a computational technique based on 16 Sutras. Among the 16 Sutras, Urdhva Tiryakbhyam Sutra is used to carry out the multiplication. Urdhva Tiryakbhyam Sutra can be used in all multiplication operations. In Vedic multiplication, the multiplication operation is performed step by step [39-41].

Fig. 3 shows a 2*2 Vedic multiplication operation using the Urdhva Tiryakbhyam Sutra technique. In the 2*2 multiplication operation, as shown in Fig. 3, the multiplication operation is converted into 4 different sections that are executed in parallel. At the same time, four multiplication operations A0*B0, A1*B0, A0*B1, and also A1*B1 are calculated, and then the results are placed in their position and add together. The double-ended blue arrows in Fig. 3 indicates the operation of multiplication between two values. Larger multipliers are also made using smaller Vedic multipliers.

Fig. 4 shows a 4*4 Vedic multiplication operation using the Urdhva Tiryakbhyam Sutra technique which is designed by using four 2*2 Vedic multipliers. As shown in Fig. 4, the multiplication operation of two 4*4 numbers in the Vedic multiplication method is divided into four 2*2 multiplication operations. Thus, at the same time, four multiplication operations A1A0*B1B0, A3A2*B1B0, A1A0*B3B2, and A3A2*B3B2 are calculated, which, based on the explanations given in Fig. 3, each of these multiplication operations is also divided into 4 parts.

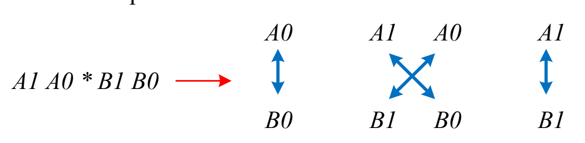


Fig. 3: Urdhva Tiryakbhyam 2*2 multiplication.

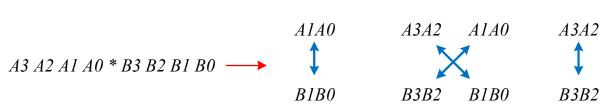


Fig. 4: Urdhva Tiryakbhyam 4*4 multiplication.

3. Related Work

In this section, we review work in the field of parity-preserving reversible Vedic multiplication and we will analyze the previous reversible parity-preserving multiplier circuits.

Several reversible Vedic circuits have been presented so far. These circuits fall into two categories: non-parity-preserving and parity-preserving. Non-parity-preserving reversible Vedic circuits are usually designed at a lower cost, but they cannot preserve parity. One can spend more to design and develop a more reliable circuit with higher capability. The circuits presented in [25, 26] are non-parity-preserving reversible Vedic multipliers.

On the other hand, two parity-preserving reversible Vedic circuits have been presented so far. In the paper on the 4*4 parity-preserving reversible Vedic multiplier presented in [13], the Carry look-ahead adder circuit is used to accelerate the operation. In this article, the carry look-ahead adder designed is a 2-bit adder, but the Vedic multiplier circuit needs a larger adder. The author has not even outlined a design with larger scales.

The other parity-preserving reversible circuit is presented in [14]. In this paper, a combination of two reversible Islam gates is used to create a full adder. The Islam Gate is a 4*4 gate that is parity-preserving and can be used as the producer of the AND function as well as the Half Adder function [17].

Since our suggested reversible Vedic multiplier circuits are parity-preserving, so we try to improve the two previous designs in [13, 14]. In this design, 4 reversible 2*2 circuits and 3 reversible 4-bit adder circuits are required.

4. The Proposed Parity-Preserving Reversible Vedic Multiplier

In this paper, we have designed our proposed parity-preserving reversible Vedic circuit. We need to use more inputs and outputs in the circuit because of the parity-preserving circuit. This can increase the activity and thus increase the dynamic power consumption. First, we have offered three different reversible blocks that we can use in our proposed designs. These blocks have parity-preserving capabilities, and each one is used in a specific position. We have presented three proposals from the 2*2 multiplication with the help of our proposed blocks. Then, we have offered our proposed reversible 4-bit circuit. In the next step, we have combined our proposed circuits and proposed a parity-preserving reversible 4*4 Vedic circuit; and finally, we have designed a parity-preserving reversible n*n Vedic circuit which can be used to perform multiplication in any scale. After presenting our designs, we will provide the formulas in which we can obtain the value of all performance measures in any dimension for all of our proposed methods.

4.1. First proposed block

In this subsection, we propose a parity-preserving reversible 4*4 block. The aim of our proposed block design is to generate the AND function in parity-preserving reversible logic. Fig. 5 represents this block, called B1. In this design, we have tried to design an AND function that is parity-preserving with an optimum quantum cost. A and B are inputs of AND function. Also, this block requires two constant inputs. One of the advantages of our proposed block is that we can simultaneously design two AND functions from A and B inputs in the third and fourth outputs. This block is used in places other than AND production. The quantum cost of this block is 6.

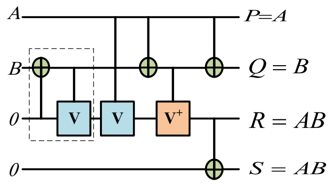


Fig. 5: Our first proposed parity-preserving reversible block (B1).

4.2. Second Proposed Block

In our proposal, we have proposed another Parity-Preserving block for the production of half adder, which can generate the sum and carry functions as parity-preserving reversible. Fig. 6 represents this block, which is named B2. B2 is a 4*4 reversible block. A and B are inputs of the half adder function, and Q and S are the outputs of this function. In this proposal, we are looking to design an optimal block for low quantum half adder design and parity-preserving. The quantum cost of the B2 block is 6.

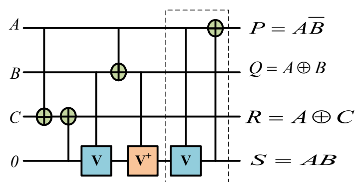


Fig. 6: Our second proposed parity-preserving reversible block (B2).

4.3. Third Proposed Block

In the third subsection, we have proposed a 5*5 reversible block. This design is parity-preserving and can be used for the production of a full adder to be used in adder circuits. The aim of this proposed block is to design a reversible block that can generate a full adder function by low quantum cost in the form of parity-preserving. Fig. 7 illustrates this block which is named B3. A, B, and C are the inputs of the full adder function, and S and R are the sum and carry outputs of this function. In our third proposal, we have been looking at designing a reversible block that could produce a parity-preserving full adder circuit by low quantum cost. The quantum cost of the B3 block is 8.

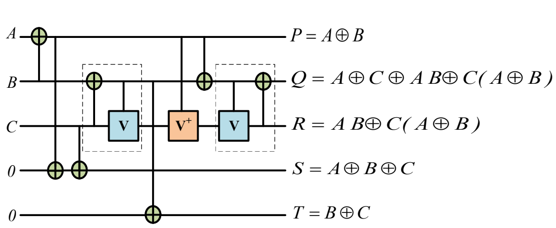


Fig. 7: Our third proposed parity-preserving reversible block (B3).

4.4. Proposed reversible 2*2 Vedic multiplier

In this section, we design a parity-preserving 2*2 Vedic multiplication circuit. In order to perform Vedic multiplication, two steps are required. For designing a reversible 2*2 multiplier based on the Vedic multiplication technique, first, we need to have the results of AND operations for circuits' inputs in order to add the results together.

In this section, we have suggested an algorithm for reversible 2*2 Vedic multiplication operation. The proposed algorithm 1 shows how the 2*2 Vedic multiplication is performed. We have used this algorithm to design our three proposed parity-preserving reversible 2*2 approach.

Algorithm 1: Proposed algorithm for designing proposed parity-preserving reversible 2*2 Vedic multiplier

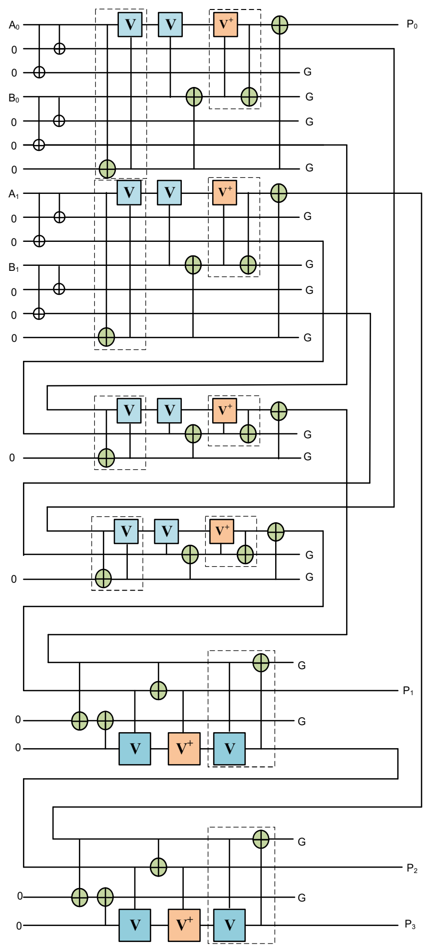
```
Mul 2*2 (|P0>, |P1>, |P2>, |P3> & |A0>, |A1>, |B0>, |B1>):
  Input: (|A0>, |A1>), (|B0>, |B1>)
  Output: (|P0>, |P1>, |P2>, |P3>)
    Begin
         Product fan-out of needed inputs (|A0>, |A1>, |B0>, |B1>) by F2G
              Set P0= A0 And B0
              Set P1= A0 And B1 Xor A1 and B0
              Set P2= (A0 And B1 And A1 and B0) Xor (A1 And B1)
              Set P3= A0 And B1 And A1 and B0
    End
```

4.5. First proposed parity-preserving 2*2 Vedic approach

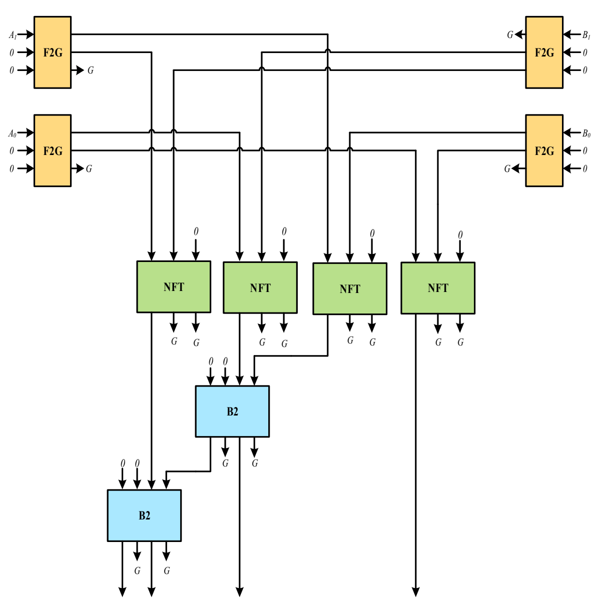
In our first proposal, we used F2G gates to produce Fan-out. In the proposed approach, four F2G gates are used for production of fan-out, four NFT gates are used for production of AND, and two proposed blocks B2 are used to produce their addition. These gates are parity-preserving so the first proposed circuit is parity-preserving. Fig. 8 shows our first proposed parity-preserving reversible 2*2 Vedic multiplier circuit.

4.6. Second proposed parity-preserving 2*2 Vedic approach

In our second proposal, we are trying to reduce the cost of the circuit to produce a 2*2 reversible parity-preserving circuit. The proposed circuit is designed to have fewer fan-outs. This will reduce the number of required gates and thus reduce the quantum cost, the number of constant inputs, and garbage outputs. We have tried to reduce the numbers of garbage outputs and constant inputs in our second proposal, with the help of the outputs of each floor as the inputs of the next level. In this design, an F2G gate to produce Fan-out, NFT gates to produce AND, and also our proposed B2 block to produce the Half Adder. In the proposed approach, the circuit speed has declined due to the classification of the AND stages of production, but this has improved another cost of the circuit. Fig. 9 represents our second proposed parity-preserving Vedic 2*2 reversible circuit.

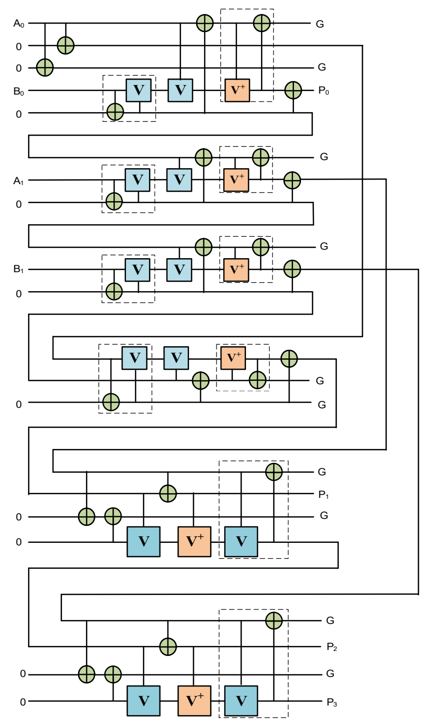


A: quantum design

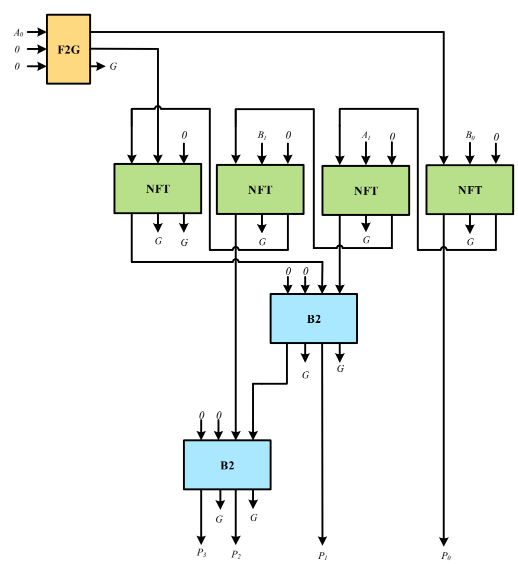


B: Block diagram

Fig.8: The first proposed parity-preserving reversible 2*2 Vedic multiplier circuit. A: quantum design. B: block diagram.



A: Quantum design

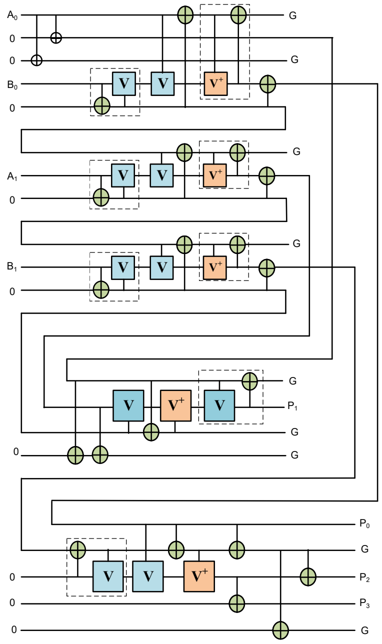


B: Block diagram

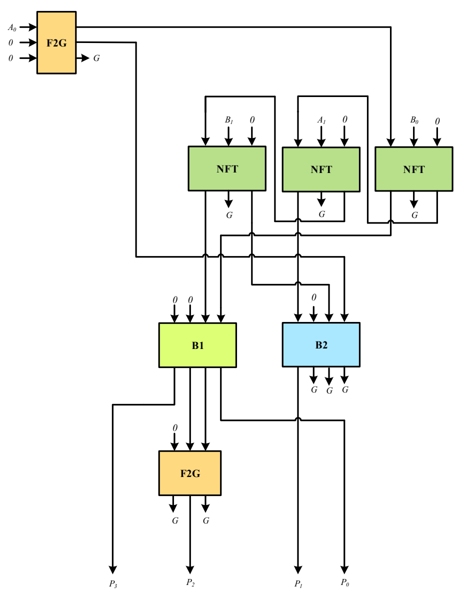
Fig. 9: The second proposed parity-preserving reversible 2*2 Vedic multiplier circuit. A: quantum design. B: block diagram.

4.7. Third proposed parity-preserving 2*2 Vedic approach

In this method, we have tried to perform 2*2 multiplication operations by reducing the number of proposed circuit costs compared to previous methods again. First, we have tried to reduce the production of Fan-out, thus we have used less F2G. In our proposal, we have tried to use possibly as many gate outputs as gate inputs. Also, the best possible use of each gate is made. The circuit in Fig. 10 represents the third design of our 2*2 reversible circuit.



A: Quantum design

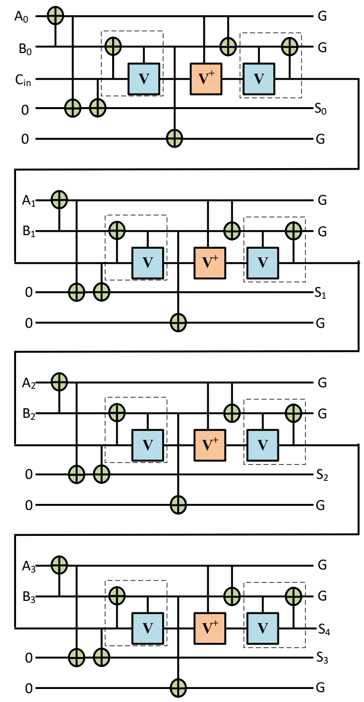


B: Block diagram

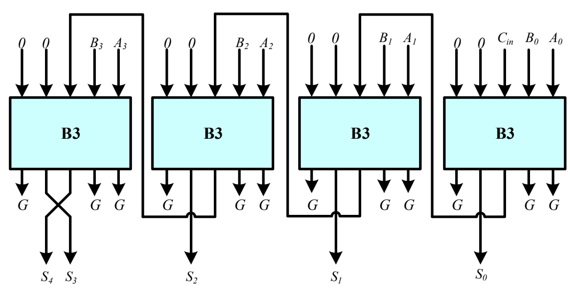
Fig. 10: The third proposed parity-preserving reversible 2*2 Vedic multiplier circuit. A: quantum design. B: block diagram.

4.8. Proposed parity-preserving reversible 4-bit Adder

We also need a 4-bit parity-preserving circuit in our proposed Vedic circuit. We have used our proposed B3 block in this plan to reduce the quantum cost of the 4-bit adder circuit. Fig. 11 illustrates our 4-bit reversible parity-preserving circuit.



A: quantum design



B: block diagram

Fig. 11: The proposed parity-preserving reversible 4-bit adder circuit. A: quantum design. B: block diagram.

4.9. Proposed parity-preserving reversible 4*4 Vedic multiplier

The next step in our proposal is to design a 4*4 parity-preserving Vedic circuit. The design of a Vedic 4x4 circuit requires the use of 2*2 multiplication circuits and 4-bit adder circuit. We will use our proposed design for a 2*2 multiplier and a 4-bit adder to design our proposed reversible Vedic 4 circuit. Using the gates and reversible parity-preserving circuits in our circuit will cause our final proposal to be parity-preserved. We first come up with the design of our proposed algorithm to produce a parity-preserving Vedic 4*4 reversible circuit. Algorithm 2 represents the structure of our proposed approach.

Algorithm 2: Proposed algorithm for designing proposed parity-preserving reversible 4*4 Vedic multiplier

Mul 4*4 (|P0>, |P1>, |P2>, |P3>, |P4>, |P5>, |P6>, |P7> & |A0>, |A1>, |A2>, |A3>, |B0>, |B1>, |B2>, |B3>):
Input: (|A0>, |A1>, |A2>, |A3>), (|B0>, |B1>, |B2>, |B3>)
Output: (|P0>, |P1>, |P2>, |P3>, |P4>, |P5>, |P6>, |P7>)
Begin

Product fan-out of needed inputs (|A0>, |A1>, |A2>, |A3>, |B0>, |B1>, |B2>, |B3>) by F2G
Mul 2*2 (|P0>, |P1>, |T0>, |T1> & |A0>, |A1>, |B0>, |B1>)
Mul 2*2 (|T2>, |T3>, |T4>, |T5> & |A0>, |A1>, |B2>, |B3>)
Mul 2*2 (|T6>, |T7>, |T8>, |T9> & |A2>, |A3>, |B0>, |B1>)
Mul 2*2 (|T10>, |T11>, |T12>, |T13> & |A2>, |A3>, |B2>, |B3>)
Add 4bit (|S0>, |S1>, |S2>, |S3>, |C4> & |T0>, |T1>, |0>, |0>, |T2>, |T3>, |T4>, |T5>)
Add 4bit (|P2>, |P3>, |S4>, |S5>, |C6> & |S0>, |S1>, |S2>, |S3>, |T6>, |T7>, |T8>, |T9>)
Add 4bit (|P4>, |P5>, |P6>, |P7>, |C8> & |S4>, |S5>, |C6>, |0>, |T10>, |T11>, |T12>, |T13>)
End

As can be seen from proposed algorithm 2, we need 4 circuits of 2*2 multipliers to generate the Vedic 4*4 reversible parity-preserving circuit, which we use in the 2*2 multiplication circuit. On the other hand, we need 4-bit adder circuits to generate the product of the addition, with the proposed 4-bit reversible parity-preserving circuit usage. Fig. 12 represents the multiplication procedure of the parity-preserving 4*4 Vedic reversible multiplier circuit.

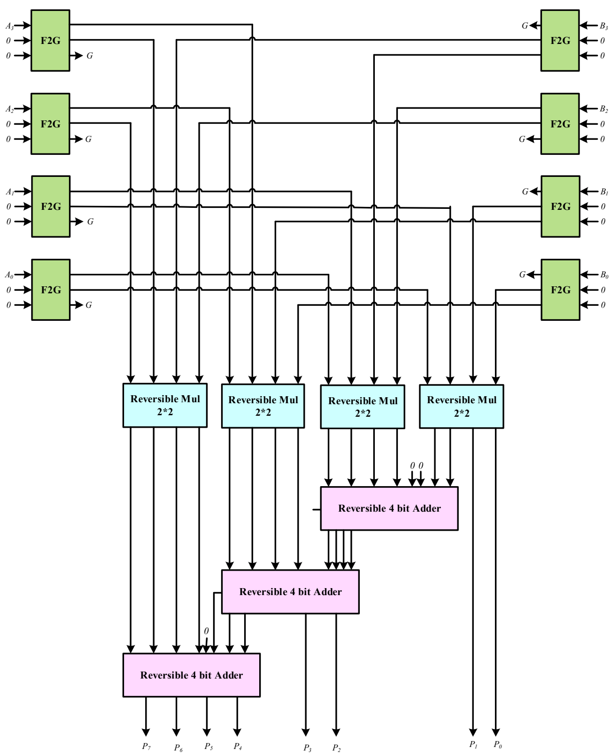


Fig. 12: Our proposed parity-preserving reversible 4*4 Vedic multiplier circuit.

Since operation 4*4 Vedic requires several inputs for multiple inputs, F2G gates are used to generate Fan-outs. The inputs obtained by the F2G gates are parallel to the 2*2 multipliers, so that the production of AND and the total product of the multiplication is done. Then, to combine the results of the 4 reversible circuits, we can obtain the final results with the aid of a 4-bit reversible parity-preserving circuit.

4.10. Proposed parity-preserving reversible n*n Vedic multiplier

We have also sought to design a reversible parity-preserving Vedic n*n multiplication circuit so that we can extend our proposed reversible Vedic circuit to any dimensions. The design of this circuit makes it possible for us to multiply both numbers of circuits needed in reversible logic and also to perform multiplication operations. Initially, we will design the proposed algorithm of this Vedic circuit of reversible parity-preserving n-bits. Algorithm 3 illustrates how our proposed design works.

Algorithm 3: Proposed algorithm for designing proposed parity-preserving reversible n*n Vedic multiplier

Mul n*n (|P0>, |P1>, ..., |P2n-1> & |A0>, |A1>, …,|An-1>, |B0>, |B1>, …, |Bn-1>):
Input: (|A0>, |A1>, …, |An-1>), (|B0>, |B1>, …, |Bn-1>)
Output: (|P0>, |P1>, …, |P2n-1>)
Begin
If (n==2)
Begin
Product fan-out of needed inputs (|A0>, |A1>, |B0>, |B1>) by F2G
Set P0= A0 And B0
Set P1= A0 And B1 Xor A1 and B0
Set P2= (A0 And B1 And A1 and B0) Xor (A1 And B1)
Set P3= A0 And B1 And A1 and B0
End
Else
Begin
Product fan-out of needed inputs (|A0>, …, |An-1>, |B0>, …, |Bn-1>) by F2G
Mul (n/2)*(n/2) (|P0>, …, |$P_{n/2-1}$>, |T0>, …,|$T_{n/2-1}$> & |A0>, …, |$A_{n/2-1}$>, |B0>, …,|$B_{n/2-1}$>)
Mul (n/2)*(n/2) (|Q0>, …, |Qn-1> & |A0>, … |An/2-1>, |Bn/2>, …, |Bn-1>)
Mul (n/2)*(n/2) (|R0>, …, |Rn-1> & |An/2>, …, |An-1>, |B0>, …, |Bn/2-1>)
Mul (n/2)*(n/2) (|S0>, …, |Sn-1> & |An/2>, …, |An-1>, |Bn/2>, …, |Bn-1>)
Add n bit (|U0>, …, |Un-1>, & |T0>, …,|$T_{n/2-1}$>, |0>, …, |0>, |Q0>, …, |Qn-1>)
Add n bit (|$P_{n/2}$>, …, |P_{n-1}>, |V_0>, …, |$V_{n/2-1}$>, |C> & |U_0>, …, |U_{n-1}>, |R_0>, …, |R_{n-1}>)
Add n bit (|Pn>, …, P2n-1>, & |V0>, …, |Vn/2-1>, |C>, |0>, |S0>, …, |Sn-1>)
End
End

As shown in the proposed algorithm 3, in our proposal, we need to produce a Fan-out of all inputs to reverse the Vedic n*n multiplication. Then, 4 multiplication actions will be performed with the help of multiplication functions. That is why four circuit multipliers with a half scale value will be needed. In the next step, we need 3 n-bit adder circuits that will work in three stages of aggregation to eventually reach a 2n-bit output, which is the result of multiplying two n-bit numbers. Our proposed n-bit circuit is shown in Fig. 13.
As shown in Fig. 13, in our proposal, the number of F2G gates in the first stage is 2n, which is also required in the next steps. The F2G gate is also required in 4 multiplication circuits (n / 2) * (n / 2). In our proposal, we are looking for relationships to calculate the number of required gates for a reversible Vedic multiplication circuit in any given dimension. Based on our proposed circuit in Fig. 13, we need 2n F2G gates to produce Fan-out. The F2G gate is required at a smaller multiplication step to get a 2*2 multiplication. Based on the fact that in this paper we present three proposed designs to produce a reversible 2*2 circuit, so that our proposed n*n multiplication circuit can use all of the three proposed methods of

multiplication of 2*2 to three results different from the proposed circuit n*n. If in our proposed parity-preserving Vedic n*n multiplication circuit, we use our first proposed 2*2 reversible multiplier circuit, we will have the proposed Eq. (6) to obtain the F2G gate numbers.

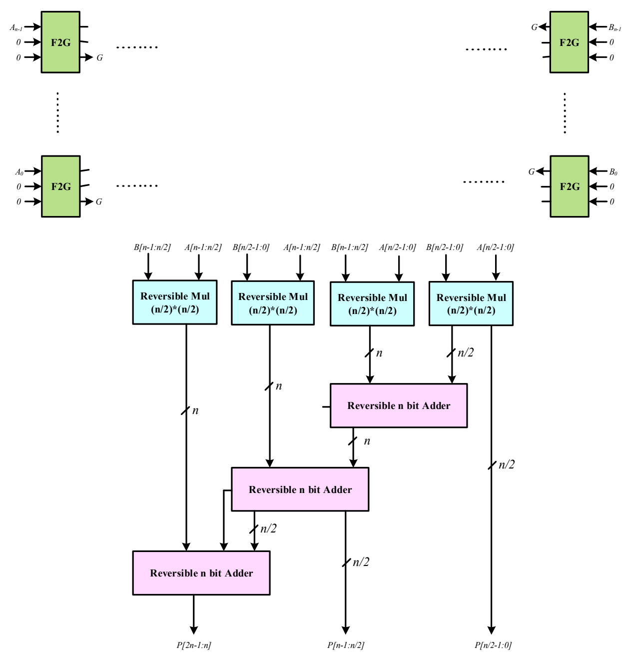


Fig. 13: Our proposed parity-preserving reversible n*n Vedic multiplier circuit.

$$\begin{cases} F(n) = 4F\left(\frac{n}{2}\right) + 2n \\ F(2) = 4 \end{cases} \tag{6}$$

Since each circuit of the multiplier is converted to 4 circuits with half-number inputs, and because it requires 2n F2G gates to produce a Fan-out. Thus, the above relation is established, which is a recurrence relation. For this reason, we compute it by using Eq. (7).

$$\begin{aligned}
&F(n) = 4F\left(\frac{n}{2}\right) + 4\left(\frac{n}{2}\right) \\
&n = 2^K \quad , \quad K = \log_2 n \quad , \quad f_k = F(2^K) \\
&F(2^K) = 4F(2^{K-1}) + 4(2^{K-1}) \\
&f_K = 4f_{K-1} + 4(2^{K-1}) \\
&f_K = C_1 4^K + C_2 2^K \\
&F(2^K) = C_1 4^K + C_2 2^K \\
&F(n) = C_1 4^{\log_2 n} + C_2 2^{\log_2 n} \\
&F(n) = C_1 n^2 + C_2 n \\
&F(n) = 2n^2 - 2n
\end{aligned} \tag{7}$$

Therefore, the number of F2G gates of any scale in our first circuit is equal to $2n^2 - 2n$.
Also, the number of other required gates and blocks in the first proposed method is calculated in Eq. (8).

$$No.G = (2n^2 - 2n)F2G + (n^2)NFT + (\frac{n^2}{2})B2 + (\frac{n^2}{4} * 3)B3 \tag{8}$$

The above formula will calculate the number of required gates and blocks for the first proposed method for the calculation of larger than 4x4 multipliers.
We can also use the proposed method to find the number of constant inputs by using Eq. (9).

$$No.CI = (2n^2 - 2n) * 2 + (n^2) * 1 + (\frac{n^2}{2}) * 2 + (\frac{n^2}{4} * 3) * 2 = 7.5n^2 - 4n \tag{9}$$

The number of garbage outputs in our first proposed method will be calculated by Eq. (10).

$$No.GO = (2n^2 - 2n) * 1 + (n^2) * 2 + (\frac{n^2}{2}) * 2 + (\frac{n^2}{4} * 3) * 3 = 7.25n^2 - 2n \tag{10}$$

Also, the quantum cost of our proposed circuit in the first method will be calculated by Eq. (11).

$$QC = (2n^2 - 2n) * 2 + (n^2) * 5 + (\frac{n^2}{2}) * 6 + (\frac{n^2}{4} * 3) * 8 = 18n^2 - 4n \tag{11}$$

If we use a reversible Vedic n*n multiplication circuit, we will use the following equation for each dimension in order to generate a 2*2 multiplication from the second proposed circuit of the circuit, in Fig. 9. The total number of required gates and blocks by our second proposed method is shown in Eq. (12).

$$No.G = (1.25n^2 - 2n)F2G + (n^2)NFT + (\frac{n^2}{2})B2 + (\frac{n^2}{4} * 3)B3 \tag{12}$$

The above formula shows the number of required gates by our second proposed Vedic multiplication of 4x4 and larger.
Also, the number of constant inputs in the second proposed Vedic n*n multiplication circuit will be calculated by using Eq. (13).

$$No.CI = (1.25n^2 - 2n) * 2 + (n^2) * 1 + (\frac{n^2}{2}) * 2 + (\frac{n^2}{4} * 3) * 2 = 6n^2 - 4n \tag{13}$$

The number of garbage outputs for our second proposed method will be based on Eq. (14).

$$No.GO = (1.25n^2 - 2n) * 1 + (n^2 + \frac{n^2}{4}) * 1 + (\frac{n^2}{2}) * 2 + (\frac{n^2}{4} * 3) * 3 = 5.75n^2 - 4n \tag{14}$$

Besides, the quantum cost of our second proposed plan is based on Eq. (15).

$$QC = (1.25n^2 - 2n) * 2 + (n^2) * 5 + (\frac{n^2}{2}) * 6 + (\frac{n^2}{4} * 3) * 8 = 16.5n^2 - 4n \tag{15}$$

In our third proposed design on n*n dimensions, using the 2*2 multiplication circuit of our third suggestion, in Fig. 10, the number of required gates and blocks in each dimension can be calculated in Eq. (16).

$$No.G = (1.5n^2 - 2n)F2G + (\frac{n^2}{4} * 3)NFT + (\frac{n^2}{4})B2 + (\frac{n^2}{4})B1 + (\frac{n^2}{4} * 3)B3 \tag{16}$$

This number of gates is valid for our third proposed Vedic multiplication circuit in 4*4 dimensions, as well as in every n*n dimension.
Also, the number of constant inputs of our third proposal in the dimension n*n is as Eq. (17).

$$No.CI = \left(3n^2 - 4n - \frac{n^2}{4}\right) * 1 + \left(\frac{n^2}{4} * 3\right) * 1 + \left(\frac{n^2}{4}\right) * 1 + \left(\frac{n^2}{4}\right) * 2 + \left(\frac{n^2}{4} * 3\right) * 2 = 5.75n^2 - 4n \quad (17)$$

The number of garbage outputs in our third proposal is as Eq. (18).

$$No.GO = \left(1.5n^2 - 2n + \frac{n^2}{4}\right) * 1 + \left(\frac{n^2}{4} * 3\right) * 1 + \left(\frac{n^2}{4}\right) * 3 + \left(\frac{n^2}{4}\right) * 0 + \left(\frac{n^2}{4} * 3\right) * 3 = 5.5n^2 - 2n \quad (18)$$

Similarly, the quantum cost of our third proposal is as Eq. (19).

$$QC = (1.5n^2 - 2n) * 2 + \left(\frac{n^2}{4} * 3\right) * 5 + \left(\frac{n^2}{4}\right) * 6 + \left(\frac{n^2}{4}\right) * 6 + \left(\frac{n^2}{4} * 3\right) * 8 = 15.75n^2 - 4n \quad (19)$$

5. Evaluation of The Proposed Reversible Vedic Multiplier Circuits

The results of our three proposed designs in this paper indicate that all of our three proposed parity-preserving Vedic circuits have a better result than the previous ones. By evaluating our proposed work with previous work, we will prove that the proposed circuits have the best results in terms of the number of constant inputs, the number of garbage outputs, and quantum cost. On the other hand, our three circuits have advantages over each other. Since our proposed multiplication circuits are Vedic reversible multipliers, we have compared our proposed circuits to the previous parity-preserving reversible Vedic multiplication circuits. We initially compare the Vedic 2*2 multiplication circuits with the previous circuits. Table 1 shows these comparisons.
As outlined in Table 1, the number of constant inputs of our proposed first method is equal to 16, which is equal to the amount of previous methods. The number of our garbage outputs is 16, which is in terms of the number [14], but it works better than [13]. The quantum cost of our first proposed circuit is 40, which is better than the previous one.

Table 1: Comparison of different parity-preserving reversible 2*2 Vedic multiplier

	N. of Constant Input	N. of Garbage Output	Quantum Cost
The First Proposed Method	16	16	40
The Second Proposed Method	10	10	34
The Third Proposed Method	9	9	31
Haghparast & et.al[14]	16	16	42
Sahu & et.al [13]	16	18	42

In our second proposal, the number of constant inputs, as well as the number of garbage outputs, is 10, which is better than the previous circuits and the first plan of our predecessors. The quantum cost of this circuit is equal to 34, which is still the best compared to the previous work and the proposed design.
In the third proposal, by having tried to reduce all costs, it is clear that the number of constant inputs as well as the number of garbage outputs of the third plan is equal to 9, which is the best result compare with the results of the previous work and the results of the two proposed first and second projects. The quantum cost of our proposed plan also has the best results from the previous work and our two other proposals.
As the Vedic circuit design is important, the circuit speed is also important, so the first proposal can be used, which is better than the previous quantum cost quotes, the number of constant inputs, and the number of garbage outputs. But if other criteria are critical, our second and third proposed circuits have the best results. In our third proposed circuit, in terms of the number of constant inputs, the number of garbage outputs as well as the quantum cost, we had the best results compared with the previous designs and the two proposed first and second ones.
We now want to consider the proposed 4-bit adder circuit and compare it with previous designs. Table 2 shows these comparisons. In [13], authors have been looking at designing a Carry Look-ahead Adder circuit. The design in this article is a 2-bit circuit plan, while a Vedic multiplication circuit requires a 4-bit adder circuit. The authors did not extend their proposed proposal to larger dimensions. On the other hand, in the circuit presented in [13], fan-outs have been used repeatedly which is unauthorized in reversible circuits. Although the proposed circuit has a high cost, it's not comparable in terms of the error and also because its dimensions are not 4-bit.

Table 2: Comparison of different parity-preserving reversible 4-bit adder

	N. of Constant Input	N. of Garbage Output	Quantum Cost
Our Proposed Method	8	12	24
Haghparast & et.al [14]	8	12	56

As shown in Table 2, the number of constant inputs, as well as the number of garbage outputs of our proposed circuit, is equal to the plan presented in [14]. Our quantum cost plan has a result of 24, which is the best result.
As mentioned earlier, with a combination of Vedic 2*2 multiplication circuits and a 4-bit adder circuit, a Vedic 4*4 multiplication circuit can be reversed. By placing the three reversible Vedic 2*2 circuits we are proposing a Vedic 4*4 reversible multiplication design. We can offer three proposed designs of 4*4 parity-preserving Vedic multiplication. Table 3 shows the results of the three proposed designs and compares them with previous designs.

Table 3: Comparison of different parity-preserving reversible 4*4 Vedic multiplier

	N. of Constant Input	**N. of Garbage Output**	**Quantum Cost**
The First Proposed Method	104	108	272
The Second Proposed Method	80	84	248
The Third Proposed Method	76	80	236
Haghparast & et.al [14]	102	110	352

Because of the lack of required information in [13] on both designs of the 4-bit adder circuit for the Vedic multiplication and its cost, Table 3 does not address this issue. As shown in Table 3, our third proposed method has the best result in terms of the number of constant inputs, the number of garbage outputs, and quantum cost.

In order to compare our three proposals in this paper, we will examine these three reversible Vedic multiplication circuits in the n*n dimension. Equations 7 to 19 shows the comparison of three proposed designs in n*n dimensions. These equations can be used for any circuit of magnitude larger than the 4*4 multiplication circuit in all three designs, and we have evaluated the efficiency of the Vedic reversible circuit.

6. Conclusion

In this paper, we have proposed a parity-preserving reversible Vedic multiplier. First, we have introduced three reversible parity-preserving Vedic 2*2 circuits. Then we presented a 4-bit reversible parity-preserving design. Subsequently, with the combination of three proposed reversible Vedic 2*2 multiplication designs, we proposed a 4*4 multiplication circuit that offered three proposed reversible 4*4 Vedic circuits and showed that our proposed third plan of work compared with the former, had improvement in quantum cost. After that, we also introduced a circuit of the Vedic n*n reversible parity-preserving circuit that can use our two proposed 2*2 designs. In order to design all the required circuits, the proposed algorithm was first presented. Finally, relations and formulas were also used to calculate quantum costs, the number of constant inputs, and the number of garbage outputs.

We have proved that all of our three proposals are better than the previous ones. Our first proposed design is the shallowest ever. Our third proposed circuit in every dimension has the best results, compared to the previous work and the first and second proposed designs in terms of the number of constant inputs, the number of garbage outputs as well as quantum costs. The presentation of a proposed algorithm and the proposed multiplication circuit in the n*n dimension have created the conditions that can be used in any aspect of our proposed design for reversible Vedic multiplication. The results show that our proposed design has the best results in terms of constant inputs, garbage output, and quantum cost.

References

[1] R. Wille, R. Drechsler, C. Osewold, and A. G. Ortiz, Automatic design of low-power encoders using reversible circuit synthesis, Design, Automation and Test in Europe, pp. 1036–1041, 2012.

[2] M. Nielsen and I. Chuang, Quantum Computation and Quantum Information, Cambridge Univ. Press, 2000.

[3] A. Berut, A. Arakelyan, A. Petrosyan, S. Ciliberto, R. Dillenschneider, and E. Lutz, Experimental verification of Landauer's principle linking information and thermodynamics, Nature, vol. 483, pp. 187–189, 2012.

[4] M. Perkowski, and P. Kerntopf, Reversible Logic, Invited tutorial, *Proc. EURO-MICRO,* Warsaw, Poland, 2001.

[5] H. Thapliyal, and M. B. Srinivas, Novel reversible TSG gate and its application for designing reversible carry look ahead adder and other adder architectures, *Proceedings of the 10th Asia-Pacific Computer Systems Architecture Conference (ACSAC 05)*, Lecture Notes of Computer Science, Springer-Verlag, 3740, pp. 775-786, 2005.

[6] M. Shams, M. Haghparast, and K. Navi, Novel reversible multiplier circuit in nanotechnology, *World Applied Science Journal 3*, pp. 806–810, 2008.

[7] B. Parhami, Fault Tolerant Reversible Circuits, *Proc. 40th Asilomar Conf. Signals, Systems, and Computers*, Pacific Grove, CA, 2006.

[8] X. D. Cai, D. Wu, Z. E. Su, M. C. Chen, X. L. Wang, L. Li, N. L. Liu, C. Y. Lu, and J. W. Pan, Entanglement-based machine learning on a quantum computer. Phys. Rev. Lett. 114(11), 110504, 2015.

[9] S. Babazadeh, and M. Haghparast, Design of a nanometric fault tolerant reversible multiplier circuit. *J Basic Appl Sci Res 2(2)*, pp. 1355–1361, 2012.

[10] X. Qi, and F. Chen, Design of fast fault tolerant reversible signed multiplier, *Int J Phys Sci 7*(17), pp.2506–2514, 2012.

[11] P. Nils, D. Gerhard, W. Robert, and D. Rolf, Fault detection in parity preserving reversible circuit, *2016 IEEE 46th International Symposium on Multiple-Valued Logic*, 2016.

[12] K. Bhardwaj, and M. Deshpande, K-Algorithm: an improved Booth's recoding for optimal fault tolerant reversible multiplier. *26th International Conference on VLSI Design*, pp. 362–367, 2013.

[13] A. Sahu, and A. K. Sahu, High Speed Fault Tolerant Reversible Vedic Multiplier, *International Journal of Innovative Research in Advanced Engineering (IJIRAE),* ISSN: 2349-2163 Issue 6, Volume 2, 2015.

[14] M. Haghparast and M. Shams, A Novel Nanometric Parity Preserving Reversible Vedic Multiplier, *Journal of Basic and Applied Scientific Research*, 3(8)771-776, 2013.

[15] M. Hagparast, and K. Navi, A novel fault tolerant reversible gate for nanotechnology based system, *Am J Appl Sci 5(5)*, pp. 519–523, 2008.

[16] M. Islam, M. M. Rahman, Z. Begum, and M. Z. Hafiz, Fault tolerant reversible logic synthesis: carry look-ahead and carry skip adders. *International Conference on Advances in Computational Tools for Engineering Applications (ACTEA)*, pp. 396–401, 2009.

[17] S. Islam, M. M. Rahman, Z. Begum, and M. Z. Hafiz, Realization of a Novel Fault Tolerant Reversible Full Adder Circuit in Nanotechnology, *The International Arab Journal of Information Technology*, Vol. 7, No. 3, 2010.

[18] V. G. Moshnyaga, Design of minimum complexity reversible multiplier, *Proceedings of IEEE Region 10 Conference (TENCON)*, pp. 1–4, 2015.
[19] M. Z. Moghadam, and K. Navi, Ultra-area-efficient reversible multiplier. *Microelectron J*, 43, pp. 377–385, 2012.
[20] S. Kotiyal, H. Thapliyal, N. Ranganathan, Reversible logic based multiplication computing unit using binary tree data structure. *J Supercomput*, 71, pp. 2668–2693, 2015.
[21] S. Kotiyal, H. Thapliyal, and N. Ranganathan, Circuit for reversible quantum multiplier based on binary tree optimizing Ancilla and Garbage bits, *Proceedings of 27th International Conference on VLSI Design (VLSID)*, pp. 545–550, 2014.
[22] E. PourAliAkbar, M. Haghparast, K. Navi, Novel design of a fast reversible Wallace sign multiplier circuit in nanotechnology. *Microelectron J*, 42, pp. 973–981, 2011.
[23] A. P. Hatkar, A. A. Hatkar, and N. P. Narkhede, ASIC design of reversible multiplier circuit, *Proceedings of International Conference on Electronic Systems, Signal Processing and Computing Technologies*, pp. 47–52, 2014.
[24] T. R. Rakshith, and R. Saligram, Optimized Reversible Vedic Multipliers for High Speed Low Power Operations, *Proceedings of 2013 IEEE Conference on Information and Communication Technologies (ICT 2013)*, 2013.
[25] T. R. Rakshith, and R. Saligram, Design of High Speed Low Power Multiplier using Reversible logic: a Vedic Mathematical Approach, *Intl. Conf. on Circuit, Power and Computational Technologies,* 2013.
[26] H. Thapliyal and M. B. Srinivas, Novel Reversible multiplier Architecture Using Reversible TSG Gate, *Proc. IEEE International Conference on Computer Systems and Applications*, pp. 100-103, 2006.
[27] M. K. Singh, N. Shivakumar, and D. S. Rao, High Speed and Low Power Multiplier Using Reversible Logic for Wireless Communications, *International Journal of Emerging Engineering Research and Technology*, PP. 62-69, 2015.
[28] T. N. Chudasama, and J. Y. Sasamal, An efficient design of Vedic multiplier using ripple carry adder in Quantum-dot Cellular Automata, *Computers and Electrical Engineering,* 2017.
[29] Z Ariafar and M Mosleh, Effective Designs of Reversible Vedic Multiplier, International Journal of Theoretical Physics, Volume 58, Issue 8, pp 2556–2574, 2019.
[30] M. Dasharatha, B. Rajendra Naik, N. S. S. Reddy and S Mude, VLSI Design and Synthesis of Reduced Power and High Speed ALU Using Reversible Gates and Vedic Multiplier, Advances in Decision Sciences, Image Processing, Security and Computer Vision, pp 272-280, 2019.
[31] P. Gowthami and R. V. S. Satyanarayana, Performance evaluation of reversible Vedic multiplier, ARPN Journal of Engineering and Applied Sciences, VOL. 13, NO. 3, 2018.
[32] A. Aggarwal, B. Pandey, S. Dabbas, A. Agarwal and S. Saurabh, LVCMOS-Based Low-Power Thermal-Aware Energy-Proficient Vedic Multiplier Design on Different FPGAs, System and Architecture, pp 115-122, 2018.
[33] G S C Teja, K B Sindhuri, N U Kumar and A K Vamsi, Implementation of Vedic Multiplier Using Modified Architecture by Routing Rearrangement for High-Optimization, 2018 3rd International Conference on Communication and Electronics Systems (ICCES), 2018.
[34] D. Maslov, and G. W. Dueck, Reversible cascades with minimal garbage, *IEEE Trans CAD Integr Circuits Syst 23(11)*, pp. 1497–1509, 2004.
[35] A. K. Biswas, M. M. Hasan, A. R. Chowdhury, and H. M. H. Babu, Efficient approaches for designing reversible binary coded decimal adders, *Microelectron J 39*, pp. 1693–1703, 2008.
[36] T. Toffoli, Reversible computing, *Tech. memo MIT/LCS/TM-151, MIT Lab. for Computer Science,* 1980.
[37] M. Haghparast, and K. Navi, A novel reversible BCD adder for nanotechnology based systems, *American Journal of Applied Sciences 5*, pp. 282–288, 2008.
[38] E. Fredkin, T. Toffoli, Conservative logic. *Int J Theor Phys 21*, pp.219–253, 1982.
[39] P. Saravanan, P. Chandrasekar, L. Chandran, N. Sriram, and P. Kalpana, Design and Implementation of Efficient Vedic Multiplier Using Reversible Logic, *Lecture Notes in Computer Science*, Volume 7373, pp. 364-366, 2012.
[40] H. D. Tiwari, G. Gankhuyag, C. M. Kim,and Y. B. Cho, Multiplier design based on ancient Indian Vedic Mathematics, *IEEE International Conference on SoC Design*, 2008.
[41] V. Kunchigi, L. Kulkarni, and S. Kulkarni, 32-Bit MAC unit design using Vedic multiplier, *International Journal of Scientific and Research Publications*, 3(2), pp. 1-7, 2013.

Ehsan PourAliAkbar received his BSc and MSc degree in software engineering from Dezfoul branch of Islamic Azad University, Dezfoul, Iran, in 2008 and 2011, respectively. He is currently a Ph.D candidate in software engineering at Science and Research Branch of Islamic Azad University. His research interests include reversible logic, computer Arithmetic, and wireless sensor network**.**

Email: pour.ali@srbiau.ac.ir

Keivan Navi received the B.Sc. degrees in Hardware Engineering from Beheshti University, Tehran, Iran, in 1987 and M.Sc. in Electrical Engineering (Hardware Engineering) Sharif University of Technology, Tehran, Iran, in 1990. He also received the Ph.D. degree in computer architecture from Paris XI University, Paris, France, in 1995. He is currently Professor in the faculty of computer science and engineering at Beheshti University. His research interests include Fuzzy Logic, Quantum Computing, Emerging Technologies design (CNT, QCA, SET), and their application in diseases (Kennedy, COVID19, ….) as well as cognitive sciences.

Email: navi@sbu.ac.ir

Majid Haghparast received his B.Sc. in computer hardware engineering in 2003. He received his M.Sc. and Ph.D. degrees in computer architecture in 2006 and 2009, respectively. Since 2007, he has been affiliated with the Computer Engineering Faculty, Yadegar-e-Imam Khomeini (RAH) Shahre Rey Branch, IAU University, Tehran, Iran. He is

currently an associate professor and the head of the computer engineering department at Electrical and Computer Engineering Faculty, Yadegar-e-Imam Khomeini (RAH) Shahre Rey Branch, IAU University, Tehran, Iran. He has also been selected as his university's outstanding researcher in 2017 and 2019. Majid has published more than 100 research papers in various international journals and conferences. His research interests include cloud computing, WSNs, reversible logic, IoT, fault tolerance, and computer arithmetic. Since April 2017 he is conducting his sabbatical at the Johannes Kepler University Linz, Austria, where he also is a Research Fellow. He served in more than 50 international conference advisory and technical program committees. Dr. Haghparast is on the panel of reviewers for various international journals.
Email: haghparast@srbiau.ac.ir

Midia Reshadi is currently an Assistant Professor in the Computer Engineering Department at the Science and Research Branch of Azad University since 2010. His research interest is Network-on-chip including performance and cost improvement in topology, routing, and application-mapping design levels of various types of NoCs such as 3D, photonic, and wireless. Recently, he has started carrying out research in NoC based deep neural network accelerators and Silicon interposer based NoC with his team consists of MSc and Ph.D. students.
Email: reshadi@srbiau.ac.ir

Paper Handling Data:

Submitted: 29.03.2019
Received in revised form: 07.09.2019
Accepted: 05.23.2020
Corresponding author: Dr. Keivan Navi
Affiliation of the corresponding author: faculty of computer science and engineering at Shahid Beheshti University