

DESIGN OF PARITY PRESERVING REVERSIBLE LOGIC CIRCUIT FOR LOW POWER APPLICATIONS

A DISSERTATION

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE
OF

MASTER OF TECHNOLOGY
IN
VLSI DESIGN & EMBEDDED SYSTEM

Submitted by:

Sumit

2K19/VLS/20

Under the supervision of

Dr. GURJIT KAUR



DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

AUGUST, 2021

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

CANDIDATE'S DECLARATION

I, Sumit, Roll No. 2K19/VLS/20 student of M. Tech (VLSI Design and Embedded System), hereby declare that the project titled “**Design of Parity Preserving Reversible Logic Circuit for Low Power Applications**” which is submitted by me to the Department of Electronics and Communication Engineering, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associate ship, Fellowship or other similar title or recognition.

Place: Delhi

Date: 10/08/2021



Sumit

**DEPARTMENT OF ELECTRONICS AND
COMMUNICATION ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042**

CERTIFICATE

I hereby certify that the Project Dissertation titled “**Design of Parity Preserving Reversible Logic Circuit for Low Power Applications**” which is submitted by Sumit, Roll No. 2K19/VLS/20, Electronics and Communication Engineering, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is a record of the project work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place: Delhi

Date: 10/08/2021



Dr. GURJIT KAUR

**Professor, Department of ECE
Delhi Technological University**

To My Parents,
Mrs. Krishna Rani & Late Satish Kumar Pahuja
And
All My Teachers

ACKNOWLEDGEMENT

I would like to articulate my deep gratitude to my supervisor **Dr. GURJIT KAUR** (Professor, ECE), for continuous support, patience, moral support, motivation and immense knowledge during my M. Tech research project. Furthermore, I would like to thank her for introducing me to the topic as well as for the support on the way for the completion of this project report under the guidance of professional like her. I could not have imagined having a better guide and advisor for my M. Tech study.

I am also grateful to **Prof. N.S. RAGHAVA**, HOD, Department of Electronics and Communication Engineering, DTU for providing the right academic resources & environment for this work to be carried out. I would also acknowledge **Prof. NEETA PANDEY**, for her immense support. Last but not the least, my whole-heartedly thanks to all my friends who have patiently extended all sorts of help for accomplishing this undertaking.

Place: Delhi

Date: 10/08/2021



Sumit

ABSTRACT

Arithmetic and Logical Unit (ALU) and Adders are key aspects of the digital world and requires colossal amount of power. This report outlines a new design of one-bit Parity Preserving Reversible ALU circuit, 8-bit Kogge Stone adder and 16-bit Brent Kung adder. To design these circuits with low power dissipation, we have used the concept of Reversible Logic computation. Conventional Circuits dissipate an enormous amount of power due to the loss of information bits in computation, but Reversible Circuits has no data loss as one on one mapping between outputs and inputs which leads to the minimization of power dissipation. In our design, we have used Parity Preserving Reversible Gates which are having Fault tolerance property as fault occurring at internal nodes results in an error at the output. So, Parity Preserving Reversible Gates is the one in which Output Parity remains same as of the Inputs.

The aimed circuits have been carried out using Xilinx ISE 14.7 version software in Verilog HDL. To demonstrate the efficiency of proposed circuits, each subpart is shown in terms of various parameters such as Quantum cost, Ancilla Inputs, Gate Count and Garbage Outputs and also a comparison with existing work is shown. The intended design is far better than the existing one because of its fault-tolerant capabilities. So yet, no Kogge Stone adder and Brent-Kung adder circuit employing Reversible Logic Gates with Fault-Tolerant capabilities has been suggested, to the best of our knowledge. The proposed circuits extend its application over DNA mapping, Optical computation, cryptography, nanotechnology, quantum computing and digital signal processing.

TABLE OF CONTENTS

| | |
|---|-----|
| Candidate's Declaration | ii |
| Certificate | iii |
| Acknowledgement | v |
| Abstract | vi |
| Table of Contents | vii |
| List of Figures | ix |
| List of Tables | xi |
| CHAPTER 1 INTRODUCTION | 1 |
| 1.1 Motivation | 3 |
| 1.2 Objectives | 3 |
| 1.3 Organization of Report | 4 |
| CHAPTER 2 LITERATURE REVIEW | 5 |
| CHAPTER 3 REVERSIBLE LOGIC GATES | 8 |
| 3.1 Irreversible Computation | 8 |
| 3.2 Reversible Computation | 9 |
| 3.3 Reversible Logic Gates | 10 |
| 3.4 Parity Preserving Reversible Circuits | 12 |

| | |
|---|----|
| 3.5 Various Reversible Gates | 13 |
| CHAPTER 4 PROPOSED PARITY PRESERVING REVERSIBLE LOGIC CIRCUITS | 20 |
| 4.1 Arithmetic and Logical Unit | 20 |
| 4.1.1. Parity Preserving Reversible Arithmetic Unit | 20 |
| 4.1.2 Parity Preserving Reversible Logic Unit | 22 |
| 4.1.3 Parity Preserving Control Unit | 24 |
| 4.1.4 Final ALU design | 25 |
| 4.2 Kogge Stone Adder | 26 |
| 4.2.1. Proposed Parity Preserving Reversible 8-bit Kogge Stone Adder | 29 |
| 4.3 Brent Kung Adder | 33 |
| 4.3.1. Proposed Parity Preserving Reversible 16-bit Brent Kung Adder | 35 |
| CHAPTER 5 SIMULATIONS & RESULTS | 40 |
| CHAPTER 6 CONCLUSION AND FUTURE SCOPE | 44 |
| LIST OF PUBLICATIONS | 45 |
| REFERENCES | 46 |

LIST OF FIGURES

| Fig. No. | Title | Page No. |
|-----------------|---|-----------------|
| Fig. 3.1 | Irreversible Machine | 9 |
| Fig. 3.2 | Reversible Machine | 10 |
| Fig. 3.3 | Reversible Logic Block | 11 |
| Fig. 3.4 | CNOT Gate | 13 |
| Fig. 3.5 | TOFFOLI Gate | 14 |
| Fig. 3.6 | PERES Gate | 14 |
| Fig. 3.7 | COG Gate | 15 |
| Fig. 3.8 | NFT Gate | 16 |
| Fig. 3.9 | F2G Gate | 16 |
| Fig. 3.10 | FREDKIN Gate | 17 |
| Fig. 3.11 | MIG Gate | 18 |
| Fig. 3.12 | E2 Gate | 19 |
| Fig. 4.1 | RTL Schematic of Parity Preserving Reversible Arithmetic Unit | 22 |
| Fig. 4.2 | RTL Schematic of Parity Preserving Reversible Logical Unit | 24 |
| Fig. 4.3 | Parity Preserving Reversible Control Unit | 25 |
| Fig. 4.4 | RTL Schematic of Complete ALU using preservative gate | 26 |

| | | |
|-----------|---|----|
| Fig. 4.5 | 8 Bit Full Adder block diagram | 27 |
| Fig. 4.6 | 8 Bit Kogge-Stone Adder | 29 |
| Fig. 4.7 | RTL Schematic of Proposed Parity Preserving Reversible Kogge-Stone Adder | 31 |
| Fig. 4.8 | RTL Schematic of GP rectangular block | 32 |
| Fig. 4.9 | RTL Schematic of Large circular block | 32 |
| Fig. 4.10 | RTL Schematic of Small circular block | 33 |
| Fig. 4.11 | RTL Schematic of Triangular Si block | 33 |
| Fig. 4.12 | 16 Bit Full Adder block diagram | 34 |
| Fig. 4.13 | 16 Bit BKA circuit | 35 |
| Fig. 4.14 | Proposed PPRLG Brent Kung adder's RTL Schematic | 36 |
| Fig. 4.15 | GP Rectangular block's RTL Schematic | 37 |
| Fig. 4.16 | Large circular block's RTL Schematic | 38 |
| Fig. 4.17 | Small circular block's RTL Schematic | 38 |
| Fig. 4.18 | Triangular Si block's RTL Schematic | 39 |
| Fig. 5.1 | Parity-Preserving Reversible Arithmetic Unit's Simulation Waveforms | 40 |
| Fig. 5.2 | Parity-Preserving Reversible Logical Unit's Simulation Waveforms | 40 |
| Fig. 5.3 | Parity-Preserving Reversible ALU's Simulation Waveforms | 41 |
| Fig. 5.4 | Verilog HDL Simulation waveforms of Proposed Parity Preserving Reversible Kogge-Stone Adder | 43 |
| Fig. 5.5 | Proposed Brent Kung adder simulation waveforms | 43 |

LIST OF TABLES

| Table No. | Title | Page No. |
|------------------|---|-----------------|
| Table 3.1 | CNOT Gate Truth Table | 13 |
| Table 3.2 | TOFFOLI Gate Truth Table | 14 |
| Table 3.3 | PERES Gate Truth Table | 15 |
| Table 3.4 | COG Gate Truth Table | 15 |
| Table 3.5 | NFT Gate Truth Table | 16 |
| Table 3.6 | F2G Gate Truth Table | 17 |
| Table 3.7 | FREDKIN Gate Truth Table | 17 |
| Table 3.8 | MIG Gate Truth Table | 18 |
| Table 3.9 | E2 Gate Truth Table | 19 |
| Table 4.1 | Arithmetic Unit's Functions | 21 |
| Table 4.2 | Logical Unit's Functions | 23 |
| Table 4.3 | Various Parameters and Reversible gates used insub – parts of BKA | 37 |
| Table 5.1 | Comparison of Arithmetic Unit | 42 |
| Table 5.2 | Comparison of Logical Unit | 42 |
| Table 5.3 | Comparison of complete ALU | 42 |

CHAPTER 1

INTRODUCTION

To design digital circuits having low power dissipation, Future technologies can use reversible logic computing on a wide scale. There remains a tremendous amount of heat dissipation in case of Conventional Circuits because of information bits loss in computation. But Reversible Circuits consume less amount of heat as compare to Conventional Circuits since reversible logic gates have no information loss. The reversible logic circuit's key benefit is the retrieval of the inputs from the outputs. Parity Preserving Reversible Gates (PPRG) are special kind of reversible logic gates (RLG) having additional property such that Inputs and Outputs' parity are equal. This property enables any device to perform properly even in case of any failure at some intermediate nodes. Continuously research on computation circuit like ALU, Adders design is significant as these are fundamental block of any processors and hardware accelerators which is useful in the data processing. Various use of Adders is in memory address calculations, integer operations, complex floating-point arithmetic operations [1]. Various kind of adder architectures is available in literature such as Ripple-Carry Adder (RCA), Carry-Look Ahead Adder (CLA), Carry-Save Adder (CSAA), Carry-Select Adder (CSEA) and Carry-Bypass Adder (CBA) etc. However, the performance of these most of traditional traditional adders is not upto the mark as compared to parallel prefix adders. There is various kind of parallel prefix adders (PPA) such as Kogge-stone adder (KSA) and Brent Kung adder (BKA). Because of the simultaneous production of carry signals, PPA are superior.

ALU acts a significant part in the CPU of any digital system, working as a data processing unit, therefore its performance causes an important effect on the functioning of any system. This allows the computer to execute various Arithmetic and Logical Operations. Arithmetic and Logical Unit is a combination of mainly three units named as Arithmetic Unit, Logical Unit and Control Unit. Power dissipation, size, area, etc., are the main challenges in the VLSI industries. Nowadays because of Various advancements in technology, on a single die, billions or millions of transistor devices can be fabricated, so

area is decreasing day by day, but power dissipation is a significant factor. There are various ways to decrease the power dissipation at various level of abstraction such as:

- 1) At the technology level, by lowering voltage swings, by using variable V_t (threshold voltages), etc. [2] [4].
- 2) At the circuit level, one can reduce power by using Transistor sizing, by Transistor restructuring etc. [3] [4].
- 3) At Logic level, one can reduce power by using various technique such as State Machine Encoding, Gate Reorganization, Signal Gating, Logic Encoding, etc. [3].
- 4) At architecture and System level, by using techniques such as Power and Performance management techniques, switching activity techniques etc., one can reduce power dissipation. [3].

In this report, various Reversible Logic gates-based circuits are discussed to reduce dissipation of power. Due to its lower power dissipation, different studies on reversible computation have been described in the chapter 2.

To create low power design in the digital domain, Reversible logics are more advanced, robust technologies. No doubt there is an excellent advancement in technologies because of which heat dissipation in a circuit reducing time to time, but there exists a physical limit below which power dissipation can't reduce. In 1961 a German physicist named Rolf Landaeur (with IBM) said that it is irreversible that places a lower limit on energy consumption [5]. According to his principal named Landaeur principle, every irreversible operation one do consume a minimum amount of energy as described by the below equation.

$$Q \geq K T \ln 2 \quad (1)$$

where, K is the Boltzmann's constant ($K=1.380 \times 10^{-23}$ J/K) and T is the absolute heat temperature in Kelvin. Equation 1 tells us that for each item of information lost, minimum energy dissipation is $K T \ln 2$ joule. In the case of irreversible circuit design, this is the case. This barrier can be removed in case of reversible computation. In an ideal situation, A

reversible circuit has zero power dissipation. [6]. By using reversible logic gates, circuit have low power dissipation [7]. But to incorporate fault capability, one can use parity preserving reversible logic gates.

1.1 Motivation

The motive behind choosing Reversible Logic is that ideally, one can remove the energy barrier created by Irreversible logic gates. Reversible logic gates, which un-computing bits rather of discarding them, are one of the greatest ways to improve energy efficiency. Reversible Logic gates are also helpful in making any device as portable. And by incorporating Parity Preserving Reversible Logica gates, one has fault tolerant capability also. In this report, we designed various circuits which are ALU, BKA and KSA by using PPRLG. Benefits of designed ALU is that it has low power dissipation and also have fault tolerant capability. One of the key benefits of the proposed novel design of high-performance Kogge-Stone adder is that it is reversible as well as having a fault-tolerant feature. And proposed Brent Kung adder is smaller in size, simple structure, having low wiring load as compared to Kogge Stone adder but also have reversibility feature and fault tolerant capability.

1.2 Objectives

- 1) To accomplish broad exploration, and research on low power consumption and high-speed processing circuits.
- 2) To design a low power consumption based ALU and adder with fault tolerant capability.
- 3) To do comparative analysis of proposed parity preserving reversible ALU with non-parity preserving reversible ALU.
- 4) To investigate the proposed strategy as far as following measurements – Ancilla Inputs, Quantum Cost, Garbage Outputs, Gate Count.

1.3 Organization of Report

This report is divided into five chapters. The first chapter introduces the topic, and second chapter deals with the literature review. In chapter 3, we go through various fundamental ideas of reversible logic and fault tolerance. The proposed design of ALU, 8-bit KSA circuit and 16-bit BKA implemented using PPRG have been explained in chapter 4 while the simulation results of all of these recommended circuits are shown in Chapter 5.. Finally, the future scope has been discussed in chapter 6.

CHAPTER 2

LITERATURE REVIEW

Landauer first introduced a reversible computation concept in 1961. According to him, there is heat loss because of loss of information in any irreversible computation, i.e., information loss is equal to energy loss [5]. Bennet, in the year 1973, proved thermodynamically that by using the reversible circuit, energy dissipation is less compared to irreversible computation [6]. The reversible concept practically validated in 2012 [7]. Time to time, many gates, devices designed by various Designers to make large scale reversible machine.

In paper [8], 8-bit ALU using reversible gates are proposed. An 8-bit ALU can be formed by cascading one-bit ALUs. Reversible gates such as Control Output Gate (COG) and Haghparast and Navi Gate (HNG) have been implemented. The most significant aspect of this paper is the reducing nature of gate count (GC) and transistor count (TC). In [10], the researcher proposed some reversible gates having property Reversibility along with Universality. Universality means one can implement universal gates like AND, OR and NOT gates by using these proposed reversible gates. Researcher proposed a 1-bit reversible ALU using these gates in [10]. Realization of the ALU having the same functionality is implemented in [9], but now ALU is overall more efficient with regards to Quantum Cost. In the Arithmetic Part, there is a significant improvement in terms of Garbage Outputs and Quantum Cost (QC) whereas Logical Unit & Overall ALU overshadow the previous design [10] at the expenses of increased Ancilla Inputs (or Constant Inputs) and Garbage Outputs. In general, all of the research papers, there is an issue that Fault-Tolerant capability has not been addressed. So, in this report, an ALU is proposed having similar functionality as discussed in paper [9][10] that shows not only reversibility property but also Parity Preserving Property. Having a Fault-Tolerant feature [11] in an Arithmetic and Logical Unit is our main motive in this report which was not the feature of papers [8]-[10]. This report presents a novel design of one-bit Parity Preserving Reversible ALU circuit.

Time to time, by using various technologies, many Adders invented by various researchers to make the adder better with respect to performance, power dissipation and area. Various adders in literature [12]-[14] are implemented by using various technologies. Adders are crucial to the efficient implementation of any Processor. Depending on requirements, the choice of a proper adder with various essential properties is more important for the efficient working of the device. Adders are speed limiting device, and it consumes a vast amount of power in the computation. So, it should be excellent in terms of both power and performance.

In paper [12], Based on two basic things, namely the number of slices, i.e., area and performance, various 4-bit adders like RCA, CLA, CSAA, CSEA, KSA etc. have been compared. According to this paper, the best adder among all this is KSA., since it takes less space and also needs less time than that of RCA. Using the KSA, one can achieve an efficient performance. This all adders in paper [12] are implemented using Irreversible concepts. Various adders such as RCA, CSEA, CLA and KSA have been analyzed and compared in paper [13]. This all adders implemented using reversible logic gates. Energy loss in case of reversible computation is less as compare to irreversible computation. Researchers in paper [14], proposed 8-bit KSA by using reversible logic gates. According to the paper [14], compared to traditional KSA, i.e., using irreversible logic gates., the power dissipation of Kogge-stone adder with the help of reversible logic gates is decreased by 36.88 percent. These all adders in paper [13][14] were reversible but not containing parity preservative property. Overall, in this both papers, there is an issue that Fault-Tolerant capability has not been addressed. So, some researchers proposed fault-tolerant adder circuit using parity preserving reversible logic gates. In [15], RCA, CSA, CLA has been proposed using PPRLG. In [16], reversible fault tolerable full-adder circuit, fault tolerable CSA, CLA and RCA have been proposed. Paper [17] provides useful approaches for constructing reversible Carry look-ahead and carry-skip adders using Parity Preserving Reversible logic gates, so have fault tolerant capability. In terms of garbage outputs, gate count, Ancilla inputs and hardware complexity, the suggested adders are optimized in [11]. But, till now no Kogge Stone adder circuit is proposed with the help of PPRLG. As KSA is better due to its excellent performance, & by using Reversible concepts, one can decrease power dissipation. Along with reversible ideas, we incorporated fault-tolerant capability also. So, in this report,

a Kogge Stone adder is proposed that shows not only reversibility property but also Parity Preserving Property. Having a Fault-Tolerant feature in a Kogge Stone adder is our one of main motive in this report which was not the feature of papers as mentioned above. This study presents a novel design of 8-bit Parity Preserving Reversible Kogge Stone adder circuit.

According to paper [18], Computation nodes are less in Brent Kung adder when one compare with KSA and also area is less in case of brent kung adder but have more delay as compared to KSA. Area occupied by KSA and Sklansky is larger than BKA. Structure of BKA is simpler with respect to KSA and BKA's fan-out with respect to Sklansky adder. However, no Brent Kung adder circuit by using PPRLG has been proposed thus far. Because of various features, Brent Kung adder is excellent. and one can reduce its power dissipation by incorporating RLG. We included ability to tolerate fault with reversible concepts. In this research report, a 16-bit Brent Kung adder through making use of PPRLG is introduced. Because of PPRLG, BKA design have characteristics that is fault tolerable, so more reliable.

CHAPTER 3

REVERSIBLE LOGIC GATES

There is always energy dissipation in the case of computation. Over the last many years, Researchers tried time to time to reduce this energy loss. As told earlier, according to a German physicist named Rolf Landauer in 1961, it is irreversible in classical computing that places a lower limit $KT\ln 2$ joule on the consumption of energy per bit operation. If the system is reversible, then ideally, one can make this lower limit as zero joules.

3.1 Irreversible Computation

The Traditional computation method is irreversible in nature. Let's take a look at an example. Simple OR gate is irreversible in nature as by seeing the output, no one can't say always about input i.e., for what input pattern this is the output. There is always one-bit information loss for 2 Input OR gate like if output is logic 1, then no one can't say this outputs logic 1 is because of 1,0 inputs or 0,1 inputs or 1,1 inputs. One can't recover inputs from output always. This is irreversible concept. Take another example. Let assume a machine. This machine will add two numbers A and B. Let $A=9$ and $B=7$, so this machine will give output $Y = 16$.

Now, if one wants to reverse the functionality of this machine (as shown in figure 3.1), i.e., if one run in opposite direction i.e., no one know that the machine's output is 16, and we'd like to know for which input values A and B we got this machine's output of 16. No one do this operation backwards since there are a variety of outcomes. One can say $A=10$ and $B=6$ is another acceptable reaction, as is $X = 18$ and $Y = -2$. As a result, this computer is unable to have answer since there are several methods for obtaining addition as 16.

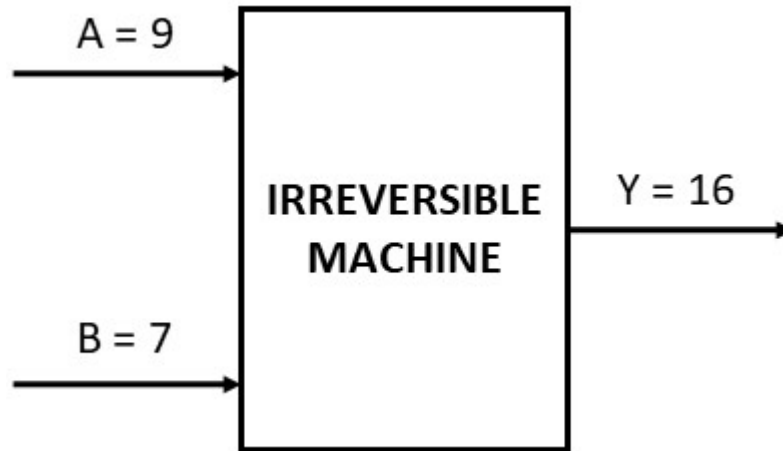


Fig. 3.1 Irreversible Machine

This is an example of irreversible machine. Here in both examples mentioned above, even if there are two inputs, there is always one output. So, one bit of information is lost here and so there is always some minimum power dissipation. In the case of irreversible computation, this barrier of the permissible limit value of energy computation per bit operation can be minimized through reversible computation. [5].

3.2 Reversible Computation

Reversible machine is different from the conventional irreversible computation in the manner that there is no information loss in case of reversible computation. One can recover any prior stage by computing in backward direction. Because of its ability to minimize power consumption in LPVD (Low Power VLSI Design), reversible computation has become popular. Let's look at the same example with extra output in figure 3.2 to better explain the principle of reversible computation.

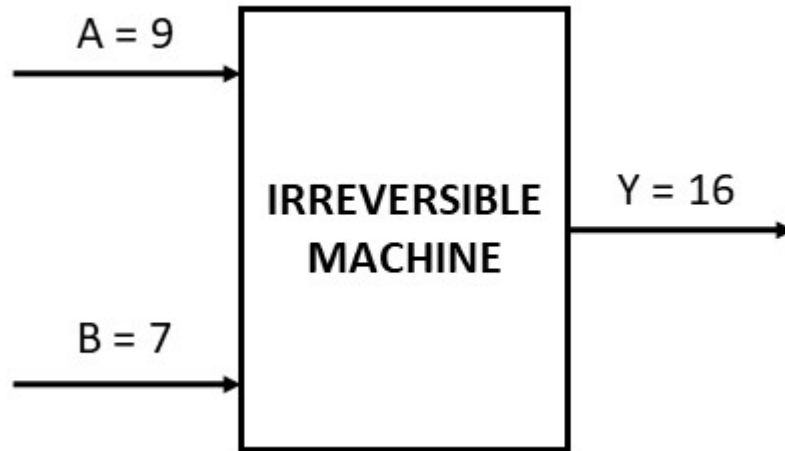


Fig. 3.2 Reversible Machine

If one give inputs $A=9$ and $B=7$, the outputs will be $A+B (=16)$ and $A-B (=2)$, respectively. If one tries to move in the opposite direction by using these outputs 16 and 8, by using simple mathematics, one can say this output is because of unique inputs which are 12 and 4. This concept is reversible and machine having this property is called reversible machine. Reversibility at all abstraction levels is needed for a machine to be reversible.

3.3 Reversible Logic Gates

Reversible logic gate (RLG) represented as $N \times N$ Circuits is an N equal number of input and output logic gate having one to one correspondence between N inputs vector (i.e., $In_1, In_2 \dots In_n$) and N outputs vector ($Out_1, Out_2 \dots Out_n$) [19]. This mapping of input and output allows to derive inputs uniquely from output. Following are the various requirements satisfied by the reversible logic gates:

- 1) The total number of inputs and outputs is equal.
- 2) A reversible logic gate maps each of combination of N Inputs to a unique of combination of N Outputs.
- 3) In the case of Reversible Gates, there are No fan-out (fan-out =1) and there are no feedback loops from output to input because one to several concepts is not reversible [19].

Through using networks with different types of RLG, a reversible circuit is built. Following characteristics can be used for logic synthesis using a reversible logic gate.

- 1) Unused outputs, i.e., garbage outputs should be minimum in number.
- 2) It should have the bare minimum of gates.
- 3) QC should be kept to a bare minimum.
- 4) It should have a minimum Ancilla Inputs
- 5) Circuit level should also be minimum.

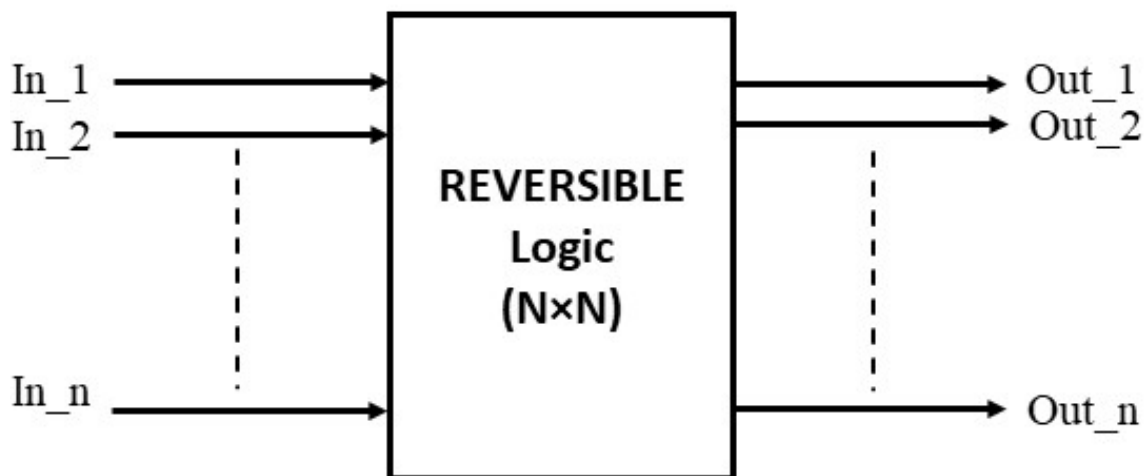


Fig. 3.3 Reversible Logic Block

If Input parity is the same as Output parity, a reversible circuit termed as parity preserving reversible circuit (or also called as conservative) [19]. Various parameters of Reversible Gates used by circuit designers to target for the purpose of optimization are as follows:

- 1) Ancilla Inputs:

Ancilla Inputs or Constant Inputs of Reversible Logic Gates are inputs which should be set at constant logic '0' or logic '1' to implement the required Digital Logic [19].

- 2) Garbage Outputs:

It is undesirable outputs which are needed only to fulfill the property of reversible

Logic gates i.e., to recover Inputs from Outputs [19].

3) Quantum Cost:

It means the cost of the Circuit or Gate in terms of a number of 1×1 and 2×2 primitive gates used to implement the Circuit or Gate. A circuit or gate is decomposed or converted in terms of the several primitive gate (NOT gate, CNOT, Controlled-V+ gate, Controlled-V gate) [19] [20].

4) Gate Count:

The total number of Reversible design entities necessary to accomplish required digital design using the Reversible approach is referred to as GC.

3.4 Parity Preserving Reversible Circuits

In various electronic systems, Fault-tolerant is possible through parity. Reversibility restores bit losses, but bit errors detection not possible in reversible circuits. But output error can be prevented by using PPRG. To have fault tolerable capability, a circuit requires PPRG to implement the system having fault tolerant capability. The system can also work in the condition of failure of any part by using the Parity Preserve functionality in a system as fault detection and diagnosis becomes easier and more efficient. The first step is to recognize the event of a failure in order to acquire fault tolerance. The parity preservation strategy is one approach. In case of PPRLG circuits, any fault that effects just one signal is observable at the primary outputs of the circuit. Thus, it is possible to integrate fault tolerance using parity preserving reversible logic circuits in computation.

Between its Inputs In₁, In₂, In₃..... In_n and Outputs Out₁, Out₂, Out₃..... Out_n, the following condition will always be met for a circuit to have parity Preserving feature, i.e.

$$\text{In}_1 \oplus \text{In}_2 \oplus \text{In}_3 \oplus \dots \oplus \text{In}_n = \text{Out}_1 \oplus \text{Out}_2 \oplus \text{Out}_3 \oplus \dots \oplus \text{Out}_n \quad (2)$$

i.e., The Input Vector Hamming Weight is equal to the Output Vector Hamming Weight. The Parity Preserving System is also known as the Fault Tolerant System, and the RLG possessing this property are known as Fault-tolerant reversible logic gates or Conservative

Reversible Logic gates or Parity preserving Reversible Logic gates (PPRLG) [21]-[24].

3.5 Various Reversible Gates

Various Reversible Gates are as follows:

1) Controlled NOT gate

- Controlled NOT gate (also termed as CNOT) is 2×2 Reversible Logic gate.
- A and B are two inputs whereas $P = A$ and $Q = A \oplus B$ are Outputs.
- If Input $A=1$ then Output Q is Complement of the Target input i.e., $Q = \bar{B}$
- Non parity preserving Reversible Logic gate
- Quantum Cost (QC) is 1 [24] [25].

Table 3.1 CNOT Gate Truth Table

| A | B | P | Q |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 |

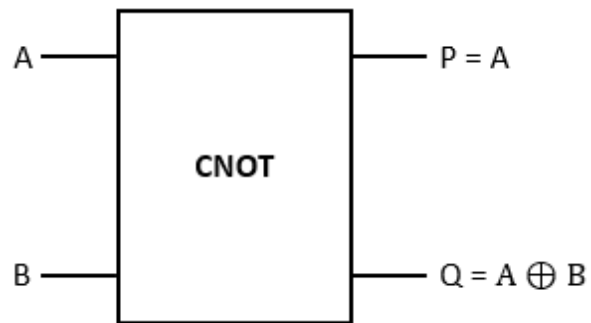


Fig. 3.4 CNOT Gate

2) TOFFOLI gate

- It is $N \times N$ reversible logic gate. It's a more advanced variant of the CNOT gate such that $O_1=I_1, O_2=I_2, O_3=I_3, \dots, O_{(N-1)} = I_{(N-1)}$ and $O(N)=I_1.I_2.I_3, \dots, I_{(N-1)} \oplus I(N)$ whereas $I_1, I_2, I_3, \dots, I_{(N-1)}, I(N)$ are N inputs and $O_1, O_2, O_3, \dots, O_{(N-1)}, O(N)$ are N outputs of the respective gate.
- For a 3×3 TOFFOLI gate, A, B and C are three inputs, and Outputs are $P = A, Q = B$, and $R = AB \oplus C$.
- Non parity preserving Reversible Logic gate
- Quantum Cost (QC) is 5 [26] [27] [29].

Table 3.2 TOFFOLI Gate Truth Table

| A | B | C | P | Q | R |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 |

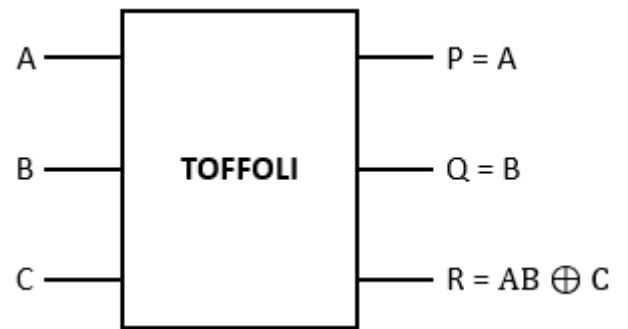


Fig. 3.5 TOFFOLI Gate

3) PERES gate

- This gate is a 3×3 reversible logic gate.
- A PERES reversible logic gate, A, B and C are three inputs, and Outputs are P = A, Q = A ⊕ B, and R = AB ⊕ C.
- Non parity preserving gate
- QC is 4 [29] [30].

Table 3.3 PERES Gate Truth Table

| A | B | C | P | Q | R |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 0 | 0 |

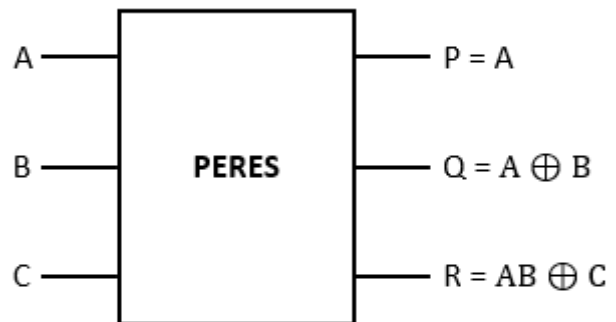


Fig. 3.6 PERES Gate

4) COG gate

- COG gate (also termed as Control Output Gate) is 3×3 reversible logic gate having; A, B, C are three Inputs and Outputs are $P = A$, $Q = \bar{A}B \oplus AC$, $R = (B \oplus C)'$.
- Non parity preserving gate.
- Quantum Cost (QC) is 4 [31].

Table 3.4 COG Gate Truth Table

| A | B | C | P | Q | R |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 |

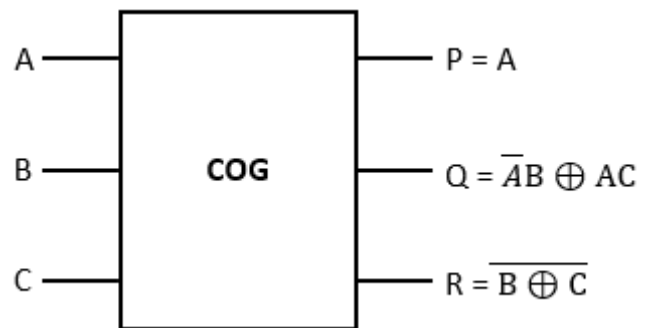


Fig. 3.7 COG Gate

Some of the Reversible gates have properties of Parity Preserving Gates which are mentioned as follows:

5) NFT gate

- It is 3×3 gate (also termed as New Fault-Tolerant) having Inputs A, B, C and Outputs are $P = A \oplus B$, $Q = \bar{B}C \oplus A\bar{C}$, $R = BC \oplus A\bar{C}$.
- It is parity preserving Reversible logic gate.
- QC is 5 [11][28].

Table 3.5 NFT Gate Truth Table

| A | B | C | P | Q | R |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 0 | 1 |

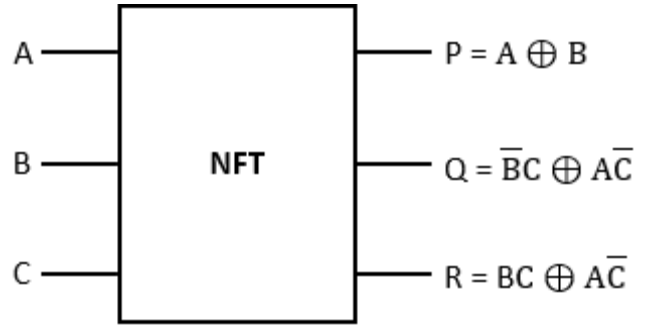


Fig. 3.8 NFT Gate

6) F2G gate

- It is 3×3 gate with Inputs A, B, C, and Outputs P = A, Q = A ⊕ B, and R = A ⊕ C.
- F2G gate is parity preserving Reversible logic gate
- QC of the F2G gate is 2 [11][28].

Table 3.6 F2G Gate Truth Table

| A | B | C | P | Q | R |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 0 | 0 |

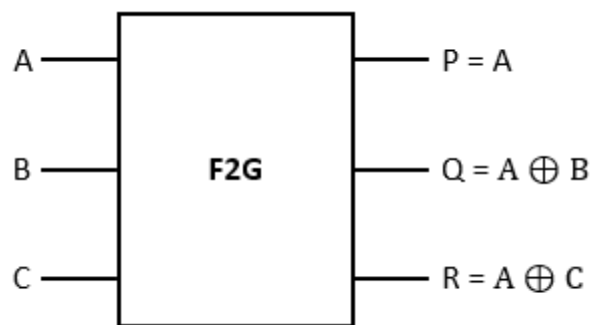


Fig. 3.9 F2G Gate

7) FREDKIN gate

- FREDKIN gate act as a cross switch. Here if A = 0 then Q = B and R = C. If A= 1 then Q = C and R = B. If the control bit is 1 then it performs then it can do swap operation.

- Three Inputs are A, B and C, whereas $P = A$, $Q = \bar{A}B \oplus AC$, $R = \bar{A}C \oplus AB$ are three outputs.
- This is Reversible logic gate having fault tolerant capability.
- The Quantum Cost here is 5 [11] [29].

Table 3.7 FREDKIN Gate Truth Table

| A | B | C | P | Q | R |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 |

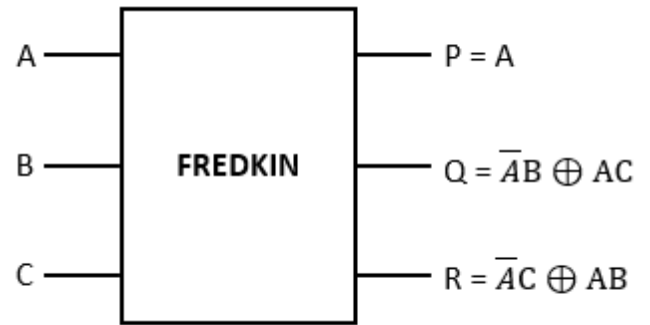


Fig. 3.10 FREDKIN Gate

8) MIG gate

- MIG gate is 4×4 PPRLG with A, B, C and D are inputs and $P = A$, $Q = A \oplus B$, $R = AB \oplus C$, $S = A\bar{B} \oplus D$ are Outputs.
- It is Reversible logic gate having fault tolerant capability.
- QC here is 7 [33] [34].

Table 3.8 MIG Gate Truth Table

| A | B | C | D | P | Q | R | S |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |

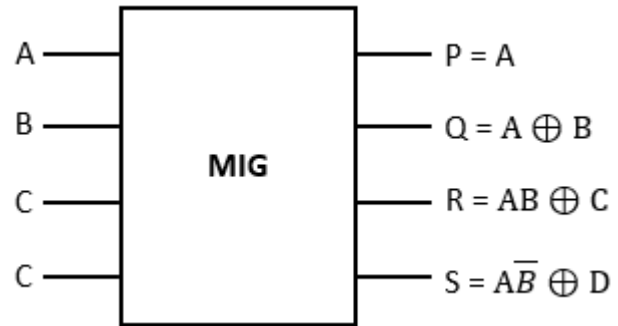


Fig. 3.11 MIG Gate

9) E2 gate

- E2 gate is 5×5 parity preserving gate having Inputs A, B, C, D, E and Outputs are $P = AB ⊕ D(A ⊕ B) ⊕ B ⊕ D$, $Q = A ⊕ B$, $R = A ⊕ B ⊕ C ⊕ D$, $S = AB ⊕ D(A ⊕ B)$, $T = A ⊕ D ⊕ E$.
- QC of the this gate is 8 [11].

Table 3.9 E2 Gate Truth Table

| A | B | C | D | E | P | Q | R | S | T |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |

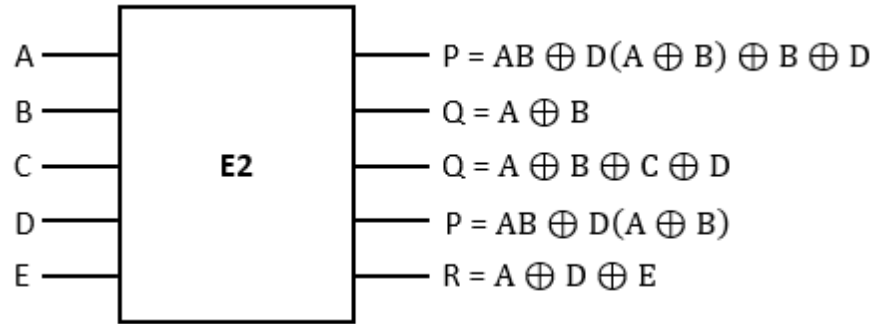


Fig. 3.12 E2 Gate

CHAPTER 4

PROPOSED PARITY PRESERVING REVERSIBLE LOGIC CIRCUITS

In this Chapter, three proposed circuits are discussed by using parity preserving reversible logic concepts which are one-bit ALU, 8-bit KSA and 16-bit BKA which are as follows.

4.1 Arithmetic and Logical Unit

An Arithmetic and Logical Unit is a Heart of a CPU of any system, working as a data processing unit. This allows the computer to execute number of Arithmetic and Logical Operations. ALU is a combination of mainly three units named as Arithmetic Unit, Logical Unit and the third one is Control Unit. In current section, ALU having the same functionality as in paper [9] but using Parity Preserving Reversible Gates are analyzed, so Proposed ALU has a fault-tolerant capability which was not in paper [9]. In this Chapter, three proposed circuits are discussed by using parity preserving reversible logic concepts which are one-bit ALU, 8-bit KSA and 16-bit BKA which are as follows.

4.1.1 Parity Preserving Reversible Arithmetic Unit

Here the Arithmetic Unit has five inputs (A, B, Cin, X, and Y) and two outputs (S and Cout). Based on two selection bits such as X and Y and Carry input, i.e. Cin from the previous stage, one can do various Arithmetic operation on inputs A and B and will get output S and Cout as mentioned in Table 4.1. Here output S is final Sum or Difference bit, and Cout is final Carry or Borrow bit [9][10].

Table 4.1 Arithmetic Unit's Functions

| X | Y | Cin | OUTPUT | FUNCTION |
|----------|----------|------------|---------------|---------------------------------------|
| 0 | 0 | 0 | B | Transfer B |
| 0 | 0 | 1 | B+1 | Increment B |
| 0 | 1 | 0 | A+B | Addition |
| 0 | 1 | 1 | A+B+1 | Add with Carry |
| 1 | 0 | 0 | B+A' | 1's complement subtraction(B-A) |
| 1 | 0 | 1 | B+A'+1 | 2's complement subtraction(B-A) |
| 1 | 1 | 0 | B-1 | Decrement B(2's complement decrement) |
| 1 | 1 | 1 | B | Transfer B |

S and Cout function are implemented based on the following Boolean Equation as:

$$S = (\bar{A}.X \oplus A.Y) \oplus B \oplus Cin; Cout = (\bar{A}.X \oplus A.Y).(B \oplus Cin).(B.Cin) \quad (3)$$

Figure 4.1 shows the RTL diagram of the Parity Preserving Reversible Arithmetic Unit, which has the same functionality as mentioned in Table 4.1. Various Features of the proposed Reversible Arithmetic Unit are as follows:

- 1) Arithmetic Unit uses NFT, E2 and F2G Parity Preserving Reversible Gates [11][25][26].
- 2) These all gates are Parity-Preserving Reversible Gates, so have the fault-tolerant capability.
- 3) The Quantum cost is 22 for the proposed Parity-Preserving Reversible Arithmetic Unit.
- 4) Garbage Output or Unused output is 11, and Ancilla Inputs are 6.

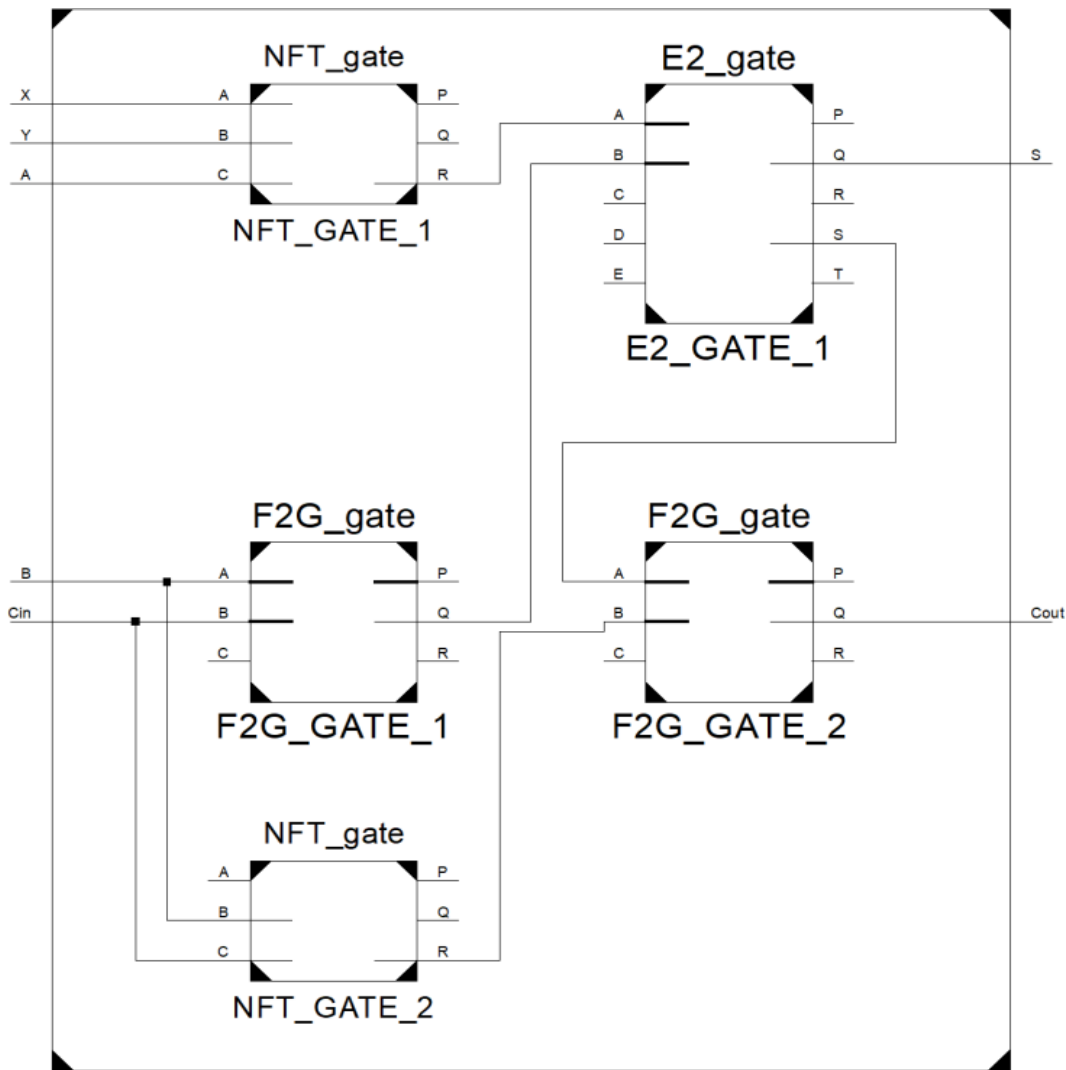


Fig. 4.1 RTL Schematic of Parity Preserving Reversible Arithmetic Unit

So proposed ALU has a fault-tolerant capability which was not in paper [9].

4.1.2 Parity Preserving Reversible Logic Unit

It is another Important block of CPU and responsible for various logical operations such as logical AND, logical OR, logical NOT, logical NOR, logical NAND, logical XOR, etc. and by using selection bits X, Y, N, P, one can implement these logical operations on Inputs A and B. In Logical Block shown in Fig. 14, we have total six inputs named as X, Y, N, P, A and B such that X, Y, N and P are selection bits and one Output as Lout [9][10]. Logical Unit functionality, as shown in Table 4.2 can be designed by following Boolean

Equation:

$$L_{out} = \bar{A}\bar{B}X \oplus \bar{A}\bar{B}Y \oplus \bar{A}BN \oplus ABP \quad (4)$$

Table 4.2 Logical Unit's Functions

| X | Y | N | P | OUTPUT |
|---|---|---|---|-----------------|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | A.B |
| 0 | 0 | 1 | 0 | $\bar{A}.B$ |
| 0 | 0 | 1 | 1 | B |
| 0 | 1 | 0 | 0 | $A.\bar{B}$ |
| 0 | 1 | 0 | 1 | A |
| 0 | 1 | 1 | 0 | $A \oplus B$ |
| 0 | 1 | 1 | 1 | A+B |
| 1 | 0 | 0 | 0 | $(A+B)'$ |
| 1 | 0 | 0 | 1 | $(A \oplus B)'$ |
| 1 | 0 | 1 | 0 | \bar{A} |
| 1 | 0 | 1 | 1 | $\bar{A}+B$ |
| 1 | 1 | 0 | 0 | \bar{B} |
| 1 | 1 | 0 | 1 | $A+\bar{B}$ |
| 1 | 1 | 1 | 0 | $(A.B)'$ |
| 1 | 1 | 1 | 1 | 1 |

Parity Preserving Reversible Logical Unit's RTL schematic is shown in figure 4.2 has the same functionality as described in Table 4.2. Following are some of features of the proposed Reversible Logical Unit:

- 1) Logical Unit uses NFT and F2G Parity Preserving Reversible Gates [11][25][26].
- 2) These all gates are Parity Preserving Reversible Gates, so have the fault tolerant capability.
- 3) Quantum cost is 34 for the proposed Parity-Preserving Reversible Logical Unit.
- 4) Garbage Output or Unused output is 18, and Ancilla Inputs are 9.

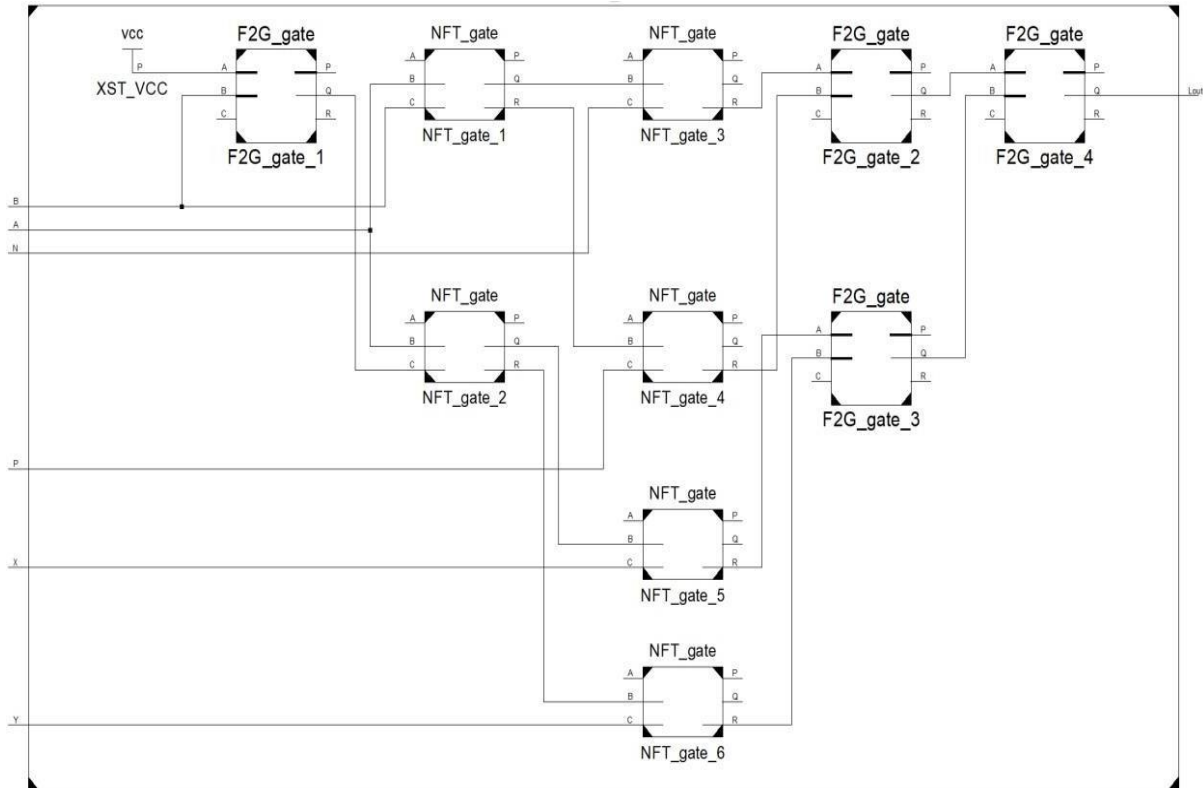


Fig. 4.2 RTL Schematic of Parity Preserving Reversible Logical Unit

4.1.3 Parity Preserving Reversible Control Unit

Here, Arithmetic operation will perform if 'Select' input is 1 and Logical operation will perform if 'Select' input is 0. Parity Preserving Reversible Control Unit RTL's schematic is shown in figure 4.3. Following are some of Features of the proposed Reversible Control Unit are as follows:

- 1) Control Unit uses NFT Parity Preserving Reversible Gates [11][25].
- 2) These all gates are Parity Preserving Reversible Gates, so have the fault tolerant capability.
- 3) Quantum cost is 10 for the proposed Parity-Preserving Reversible Control Unit.
- 4) Garbage Output or Unused output is 4, and Ancilla Inputs are 1.

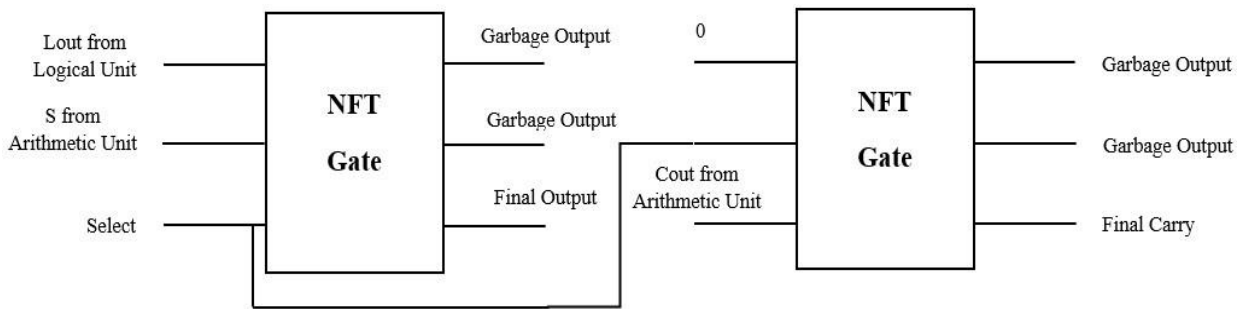


Fig. 4.3 Parity Preserving Reversible Control Unit

4.1.4 Final ALU design

The Integration of all these three units that are Parity Preserving Reversible Arithmetic Unit, Logical Unit and Control Unit together is used to form Parity Preserving Reversible Arithmetic and Logical Unit. RTL Schematic of Complete ALU using the parity-preservative gate is shown in figure 4.4.

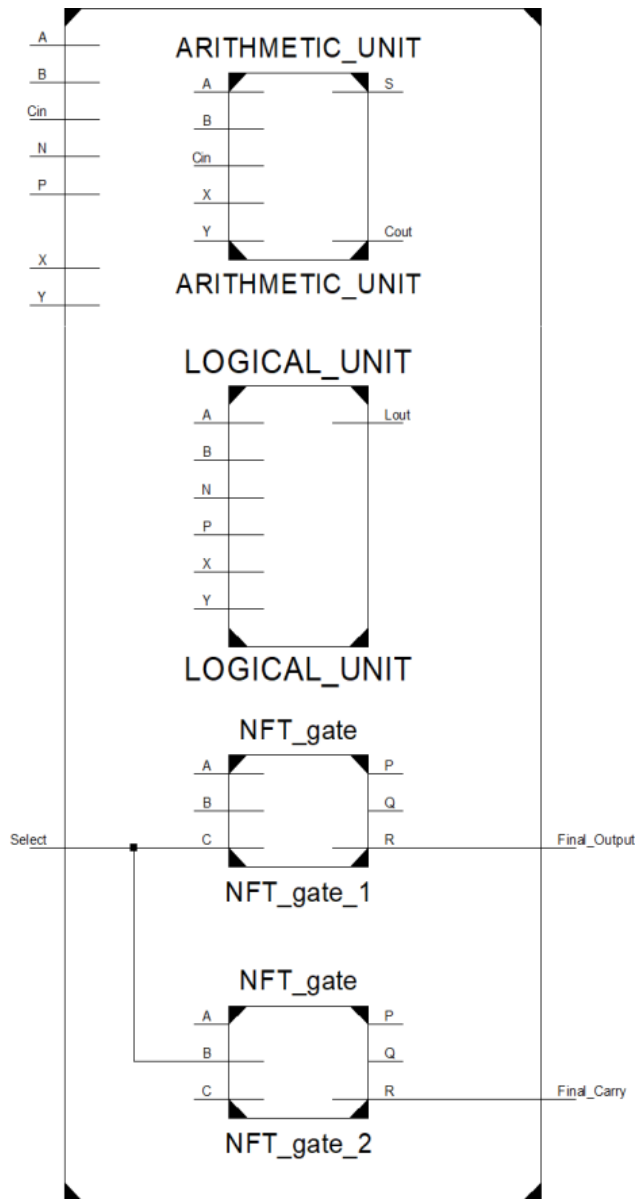


Fig. 4.4 RTL Schematic of Complete ALU using preservative gate

4.2 Kogge Stone Adder

Since adder speed is determined by carry availability before the next bit addition is done, i.e., before do the next bit addition, one don't have to wait until the last carry output. If one uses an algorithm like that so the adder would be faster. On this algorithm, Carry Look Ahead Adder is focused. Because of parallel carry generations, speed is very high here, i.e.,

no rippling effect and thus addition time is independent of the number of bits. To be added each bit in a binary sequence, the carries in the intermediate stages will be computed before using Carry propagate and carry generate independently of input carry. PPA is a subset of adders originating from the Carry look ahead adder. These adders are said to have logarithmic complexity [36].

Kogge-Stone Adder belongs to a group of adder families known as the parallel family of adder prefixes. It is one of the speediest adders. It is the better option for high-speed adders in the market but at the expense of wide areas and complexities. Figure 4.5 shows a block schematic of a general 8-bit full adder. [36].

Here $A(7:0)$ and $B(7:0)$ are the 8 bits inputs, C_{in} is the carry input in this figure whereas $Sum(7:0)$ and C_{out} are the 8-bit addition of the inputs with Carry input is the C_{in} and the C_{out} is the Carry output

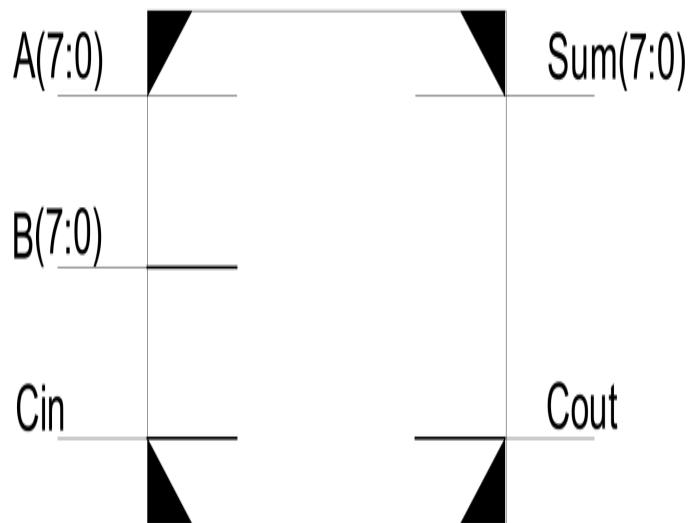


Fig. 4.5 8 Bit Full Adder block diagram

Figure 18 is an 8-bit KSA diagram.. There are three types of adder stages in this family described as follows:

1) Initialization stage:

Initialization stage also called a Pre-processing stage. Initialization stage produces two signals which are Propagate signals P_i and Generate signals G_i for every pair of 8 bits A and B and will go to the next stage, i.e. Prefix-tree stage.

$$P_i = A_i \oplus B_i \quad (5)$$

$$G_i = A_i \cdot B_i \quad (6)$$

Here GP rectangular block shown in figure 4.6 is part of Initialization stage where $G_i = A_i \cdot B_i$ and $P_i = A_i \oplus B_i$ are outputs for each input combination $A_i \cdot B_i$ where i is less than equal to 7 for 8-bit adder [37].

2) Prefix-tree stage:

For every bit, this stage combines the prefix signals to generate carry signals. The prefix-tree stage is made up of Carry Merge blocks organized to merge the prefix signals in a logarithmic manner. Carry Merge blocks in this report are termed as Large circular block (inside it written GP) and small circular block (inside it is written C_i). Large circular block generates both signals, i.e., merged propagate signals and generate signals which are as shown below:

$$G = G_i + P_i \cdot G_i' \quad (7)$$

$$P = P_i \cdot P_i' \quad (8)$$

where G_i and G_i' is the Generate signal from two different GP block and produce new Generate block. P_i and P_i' is the Propagate signal from two different GP block and produce new Propagate block according to the above equations. This operation is termed as dot operation, which are described as follows:

$$(G,P) = (G_i, P_i) \cdot (G_i', P_i') = (G_i + P_i \cdot G_i', P_i \cdot P_i') \quad (9)$$

Small circular block generates only a merged generate signal (here no propagate signal generates) i.e.

$$G = G_i + P_i \cdot G_i' \quad (10)$$

In small circular block, for each combination of A_i and B_i , final C_i is always G_i . A small circular block is a large circular block without the group propagate. Large circular block and small circular block are shown in figure 4.6 [37].

3) Summation stage:

This stage provides the final sum bit result based on the Propagate term from the first stage. and carries generates in the second stage for each A_i and B_i combination.

$$S_i = P_i \oplus C_{i-1} \quad (11)$$

Triangular block in figure 4.6 is Summation blocks [37]. A swift adder style is the Kogge-Stone Adder, but due to a large number of Carry Merge blocks, there is heavy wire routing between stages, wide circuit area and increased power consumption.

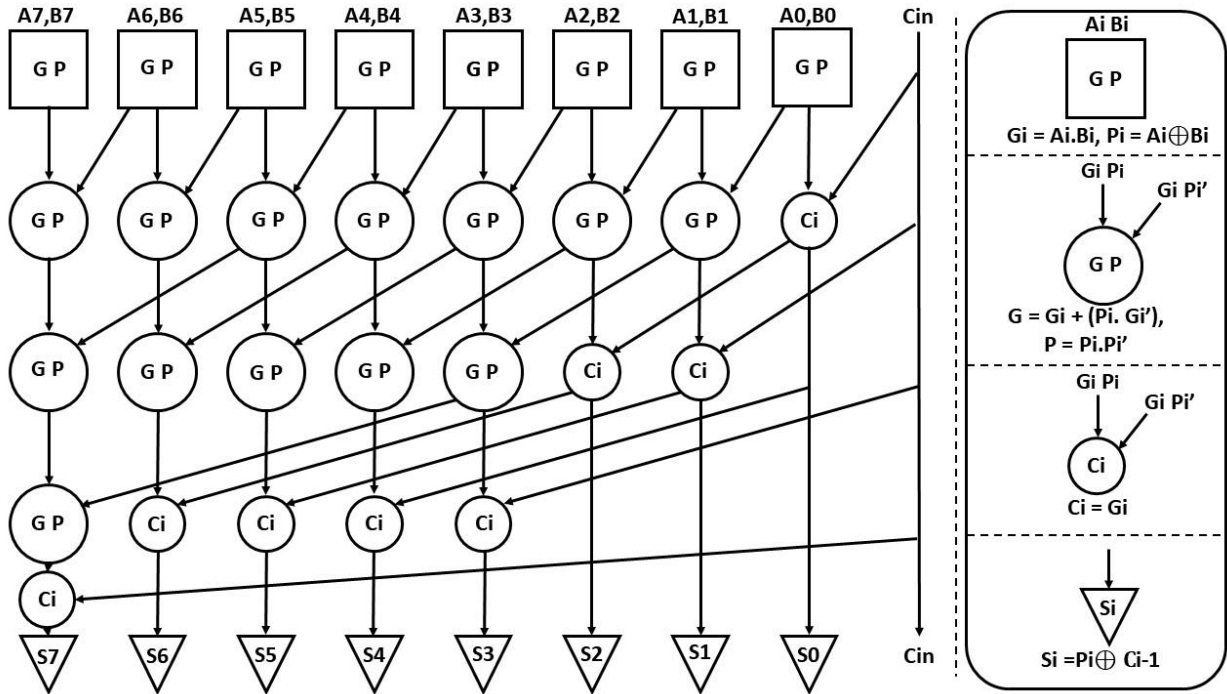


Fig. 4.6 8 Bit Kogge-Stone Adder

A member of the radix-2 tree class is this complicated but speedier adder. Class Radix-2 implies that the tree is binary, i.e., that at each hierarchy level, it combines two carries at a time [36].

4.2.1 Proposed Parity-Preserving Reversible 8-bit Kogge Stone Adder:

In this part, KSA having the same functionality as mentioned above using Parity Preserving Reversible Gates are proposed, so Proposed circuit has a fault-tolerant capability along with high performance which was not discussed till now in literature.

RTL (Register-Transfer Level) schematic of Proposed KSA using PPRLG is shown in figure 4.7 has same functionality as described in the above section. Large GP rectangular block is implemented with the help of MIG reversible logic gate having the fault tolerant capability and its RTL (Register-Transfer Level) schematic is shown in figure 4.8. Large circular GP block is implemented with the help of two NFT and one E2 gate MIG reversible

logic gate having the fault tolerant capability and its RTL (Register-Transfer Level) schematic is shown in figure 4.9. Reversible logic gates having fault tolerant capability is used to implement Small circular GP block is one NFT and one E2 gate and its RTL (Register-Transfer Level) schematic is mentioned in figure 4.10. RTL (Register-Transfer Level) schematic of Parity preserving reversible logic gate F2G is to produce final sum output is shown in figure 4.11. The following are different features of the proposed design KSA using Parity Preserving Reversible Gates.:

- 1) Kogge-Stone Adder uses NFT, E2 and F2G Parity Preserving Reversible Gates.
- 2) All these gates have fault-tolerant capabilities, so Kogge-Stone Adder has the fault-tolerant ability.
- 3) The Quantum cost is 7, 18, 2 and 13 for every GP rectangular block, Large circular block, Small circular block and Triangular block and respectively.
- 4) Garbage Output is 2, 8, 6 and 2 for every GP rectangular block, Large circular block, Small circular block and Triangular block and respectively.
- 5) Ancilla Inputs is 2, 5, 4 and 1 for every GP rectangular block, Large circular block, Small circular block and Triangular block and respectively.
- 6) Gate count is 1, 3, 2 and 1 for every GP rectangular block, Large circular block, Small circular block and Triangular block and respectively.

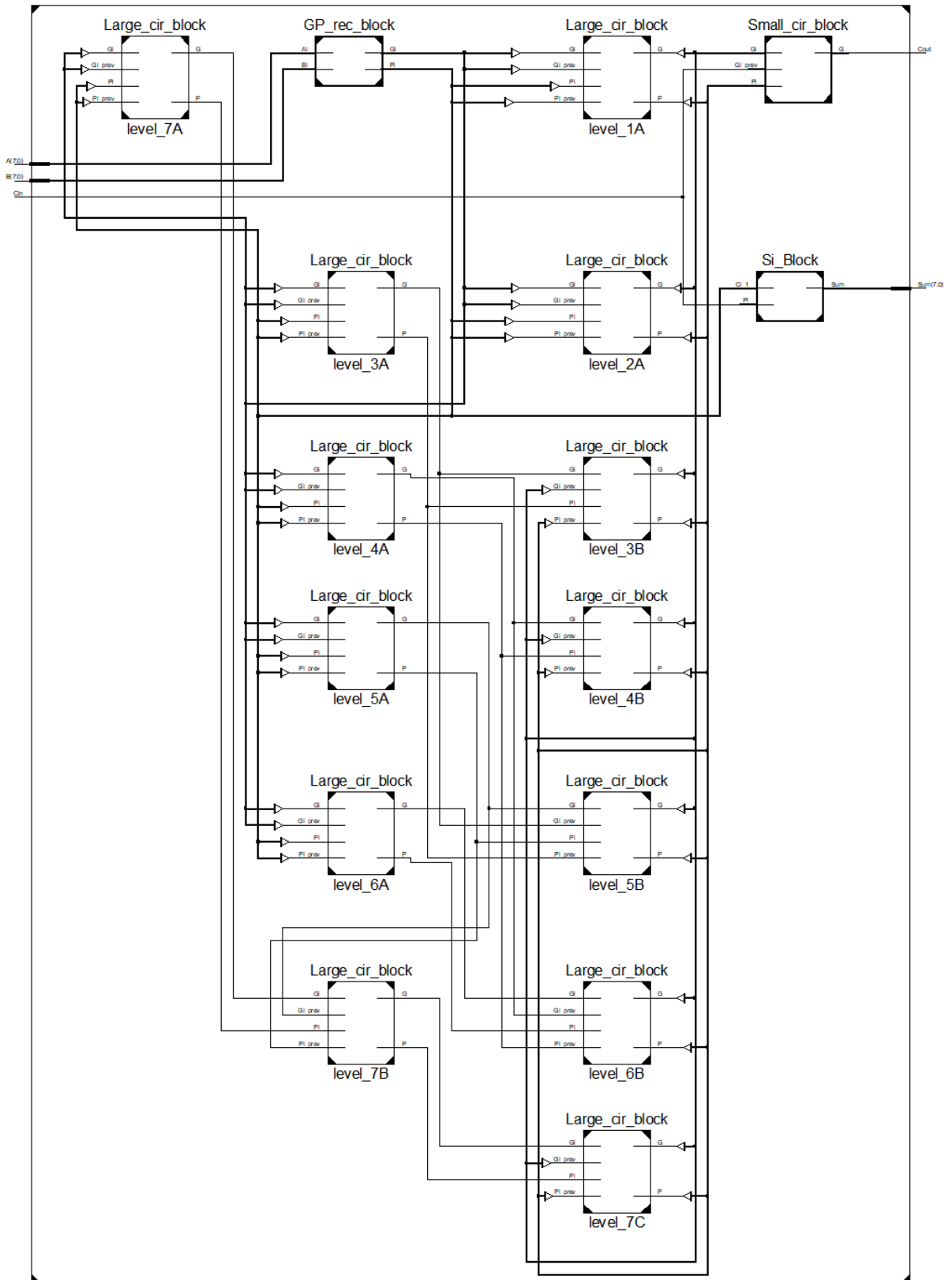


Fig. 4.7 RTL Schematic of Proposed Parity Preserving Reversible Kogge-Stone Adder

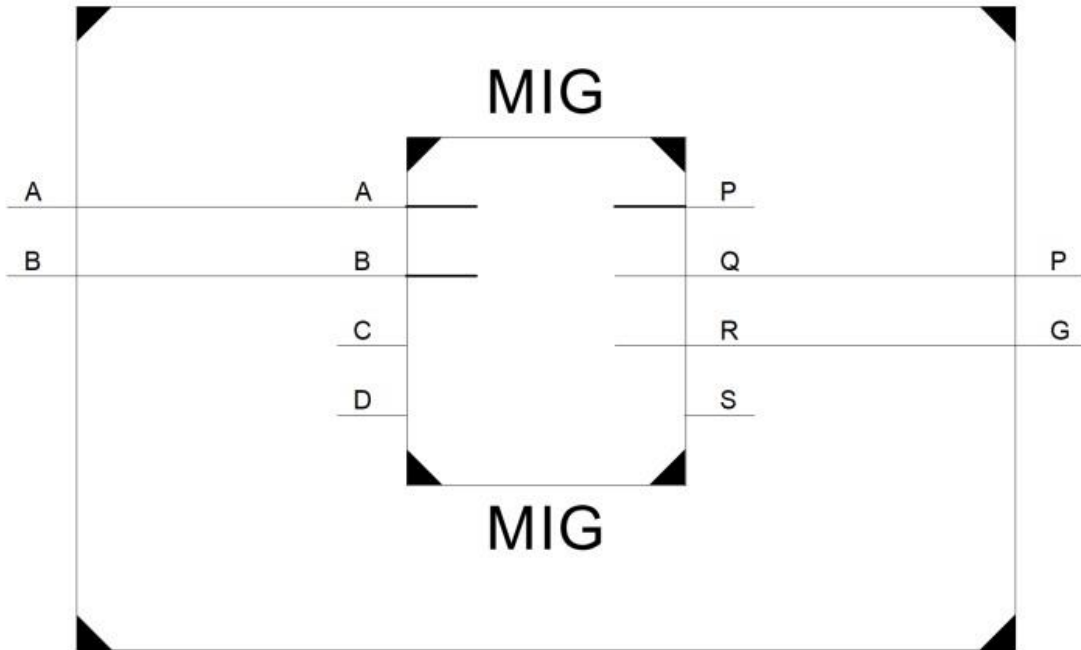


Fig. 4.8 RTL Schematic of GP rectangular block

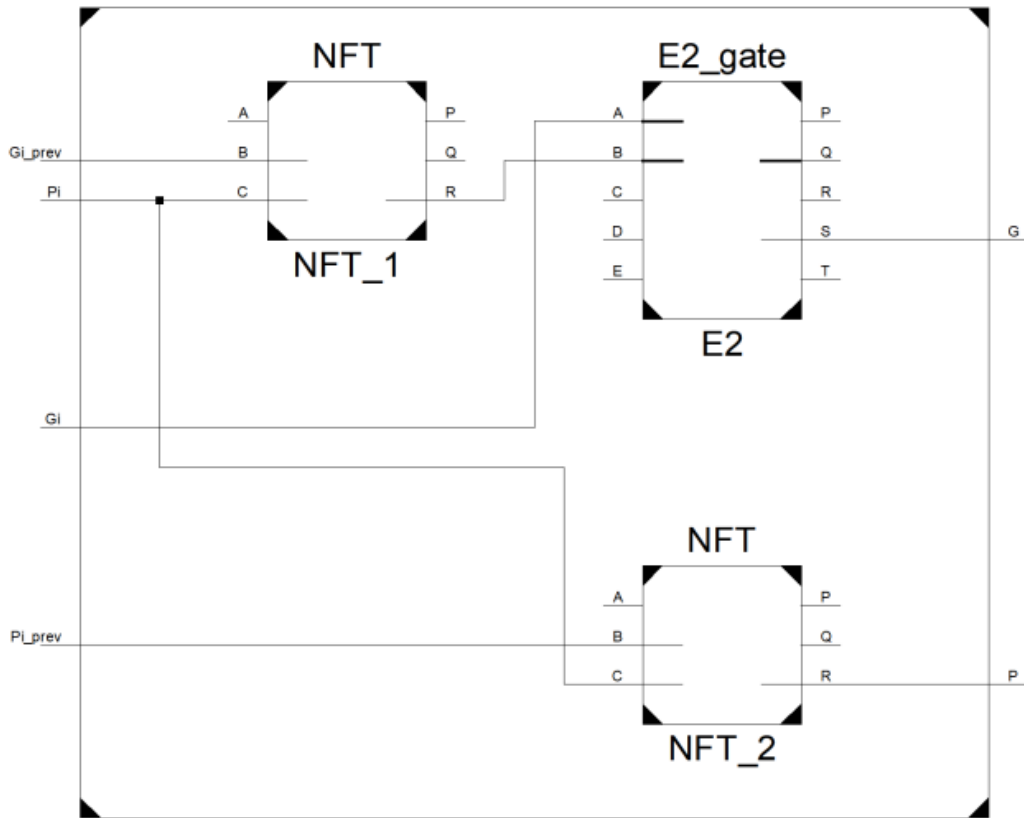


Fig. 4.9 RTL Schematic of Large circular block

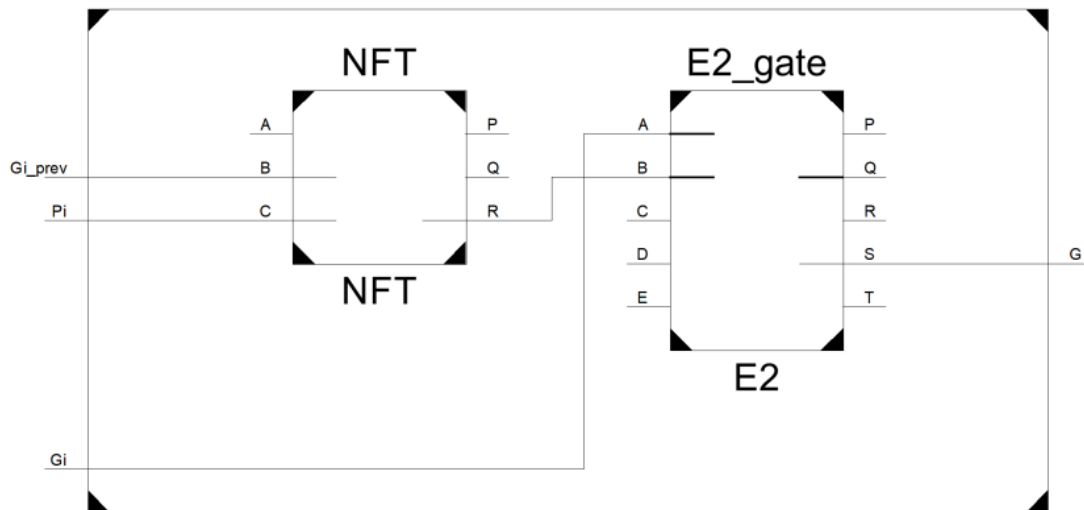


Fig. 4.10 RTL Schematic of Small circular block

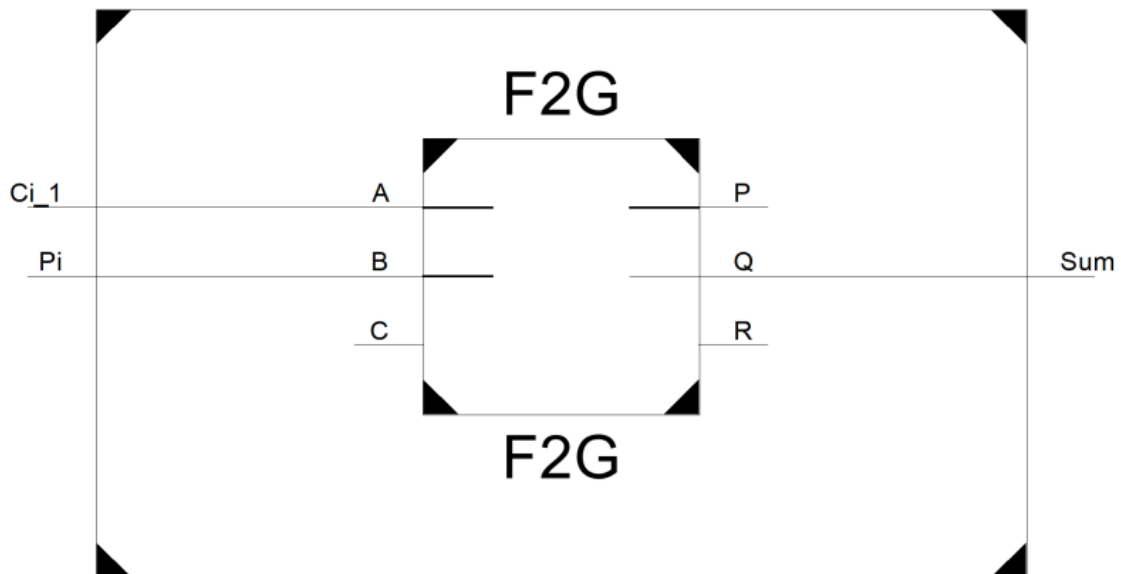


Fig. 4.11 RTL Schematic of Triangular Si block

4.3 Brent Stone Adder

In case of Ripple carry adder, first sum bit is not possible until carry input is not present and 2nd sum bit is not available till the time preceding carry is available and so on. One can say, carry have to ripple through each level, so it is time consuming adder as not all sum bits are available at the same time. To overcome this issue, carry look ahead adder comes into picture in which parallel generation carry is possible at

each level and so no rippling effect. Larger the size of input binary sequence to be added, more the complexity of carry increases. As a result, higher order CLA is harder to implement. To overcome this issue, there come Parallel Prefix adder into picture [38]. In-between phases, Carriers would be available prior with the help of CP (Carry propagate) and CG (Carry generate) no matter what C_{in} is, to add every bit of input binary sequence. The complexity of Parallel Prefix adder is considered to be logarithmic. Brent kung adder falls in category of PPA. A schematic of General 16-bit adder where 16 bits inputs and Carry Inputs is shown in left side of figure 4.12 [39]. Here Sum and Cout are 16-bit output sum and output carry of inputs respectively.

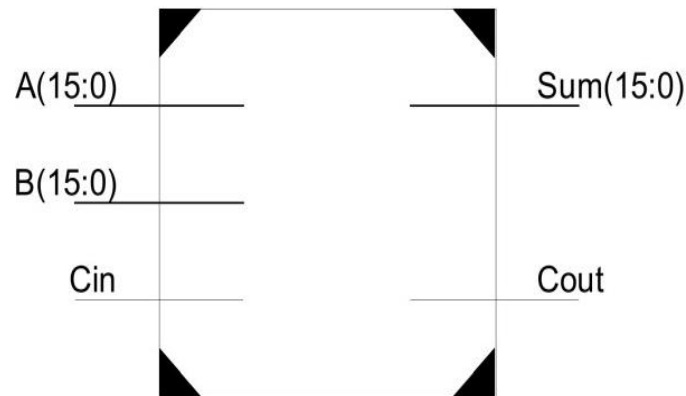


Fig. 4.12 16 Bit Full Adder block diagram

Figure 4.13 depicts a 16-bit BKA diagram.

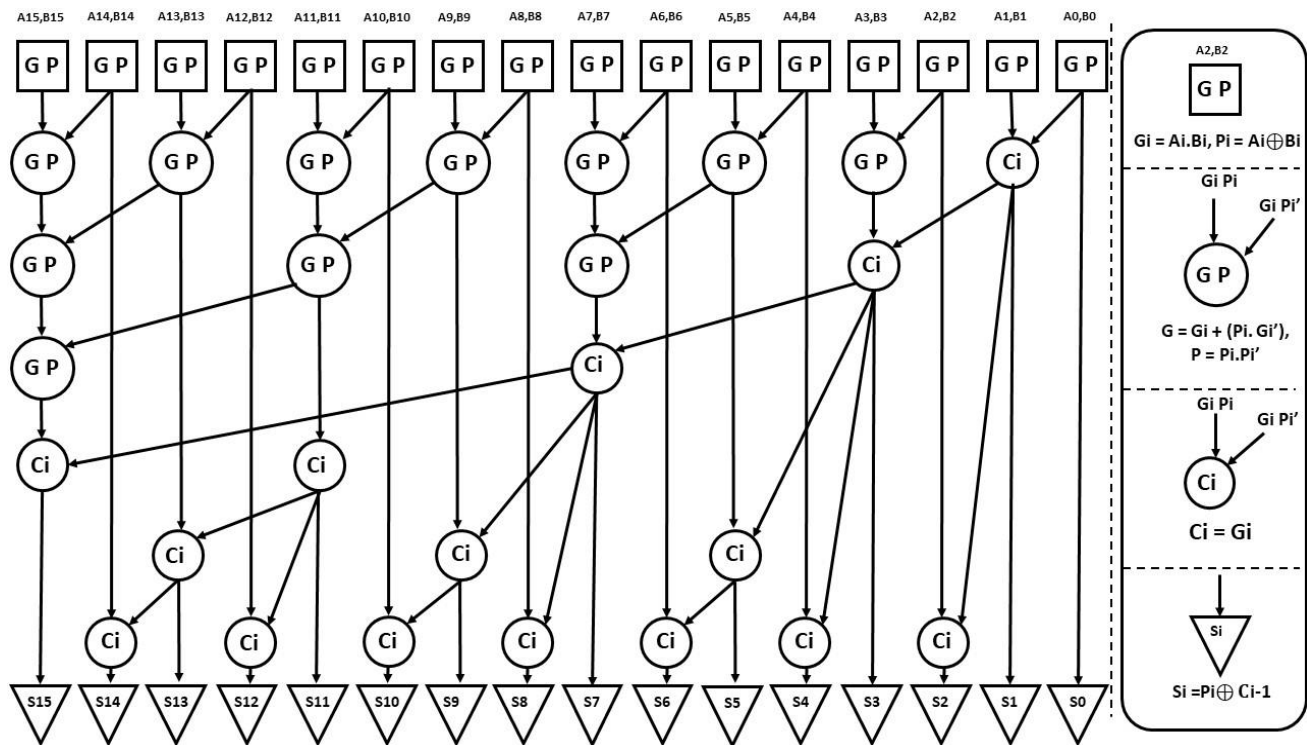


Fig. 4.13 16 Bit BKA circuit

4.3.1 Proposed Parity-Preserving Reversible 16-bit Brent Kung Adder:

BKA with the same behavior as indicated above by employing PPRLG is presented in this part. As a consequence, the suggested circuit is fault-tolerable, high-performable and low power loss which has not been addressed in the literature too far. Figure 4.14 shows a Brent Kung adder RTL schematic that has been implemented with PPRLG.

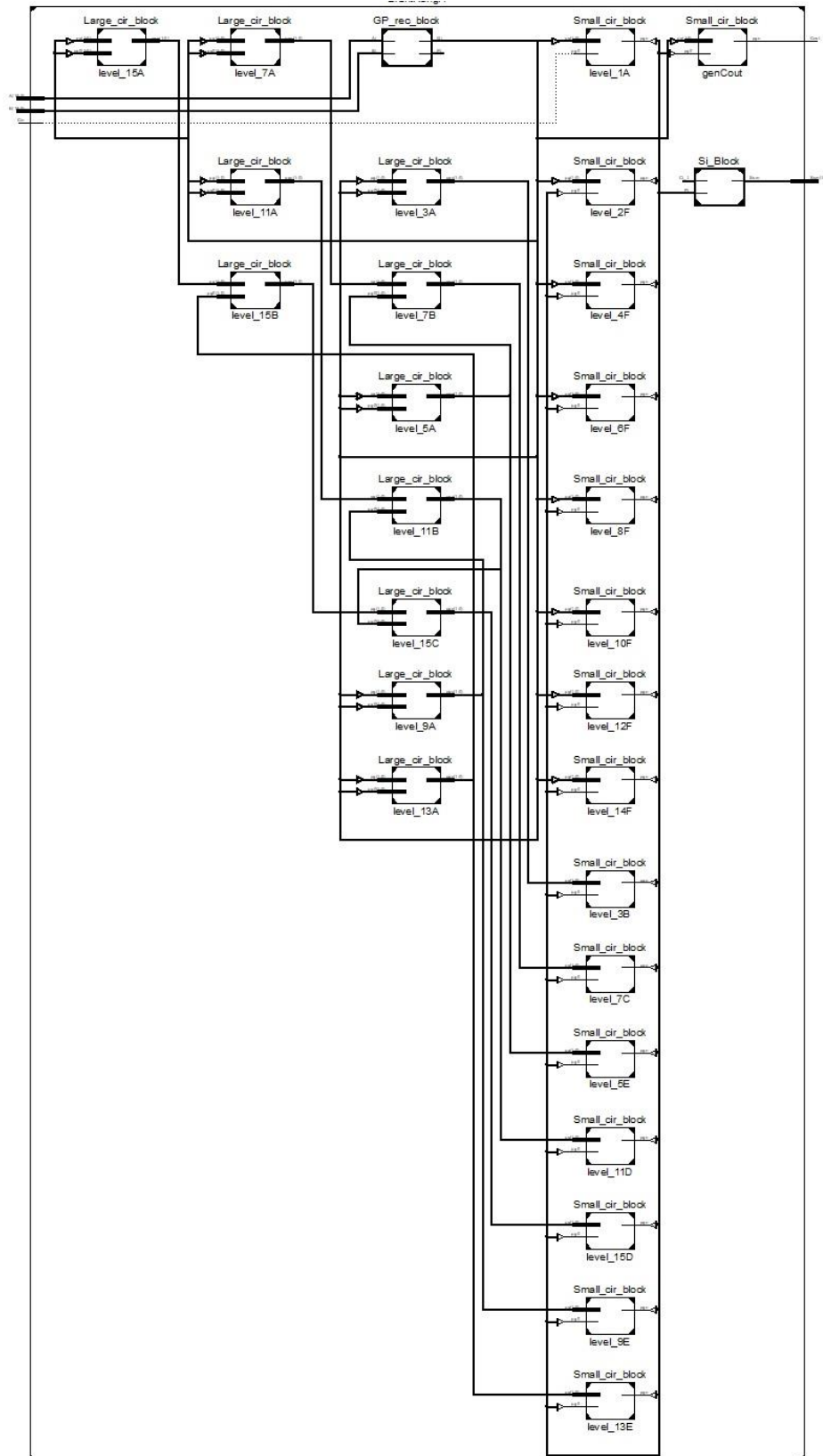


Fig. 4.14 Proposed PPRLG Brent Kung adder's RTL Schematic

Various blocks of figure 4.14 is implemented by various PPRLG is mentioned in table 4.3 along with various parameters mentioned and RTL schematic of each block is shown in figure 4.15, 4.16, 4.17 and 4.18.

TABLE 4.3 Various Parameters and Reversible gates used in sub – parts of BKA

| Various Block of Brent Kung adder mentioned in figure 5 | Corresponding Parity Preserving reversible logic Gate | Quantum cost | Garbage Output | Constant Inputs | Gate count |
|---|---|--------------|----------------|-----------------|------------|
| Large GP rectangular block | 1 MIG gate | 7 | 2 | 2 | 1 |
| Large circular GP block | 2 NFT & 1 E2 gate | 18 | 8 | 5 | 3 |
| Small circular GP block | 1 NFT & 1 E2 gate | 2 | 6 | 4 | 2 |
| final sum output | 1 F2G gate | 13 | 2 | 1 | 1 |

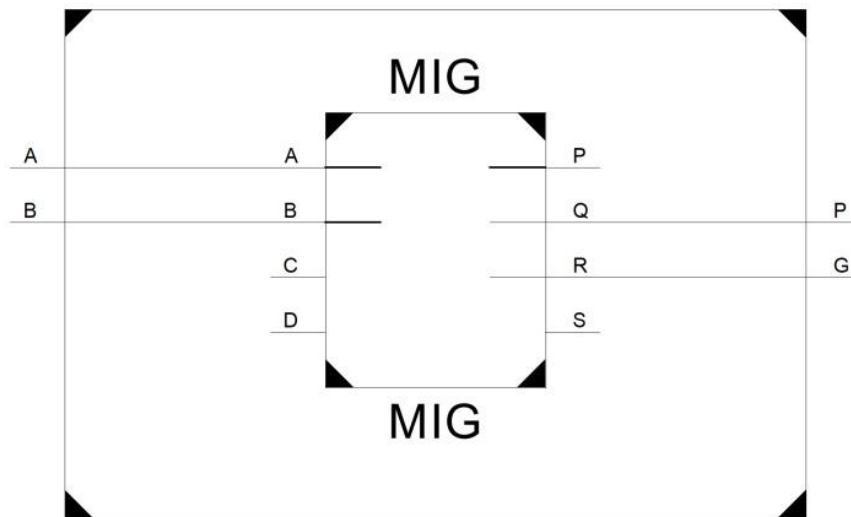


Fig. 4.15 GP Rectangular block's RTL Schematic

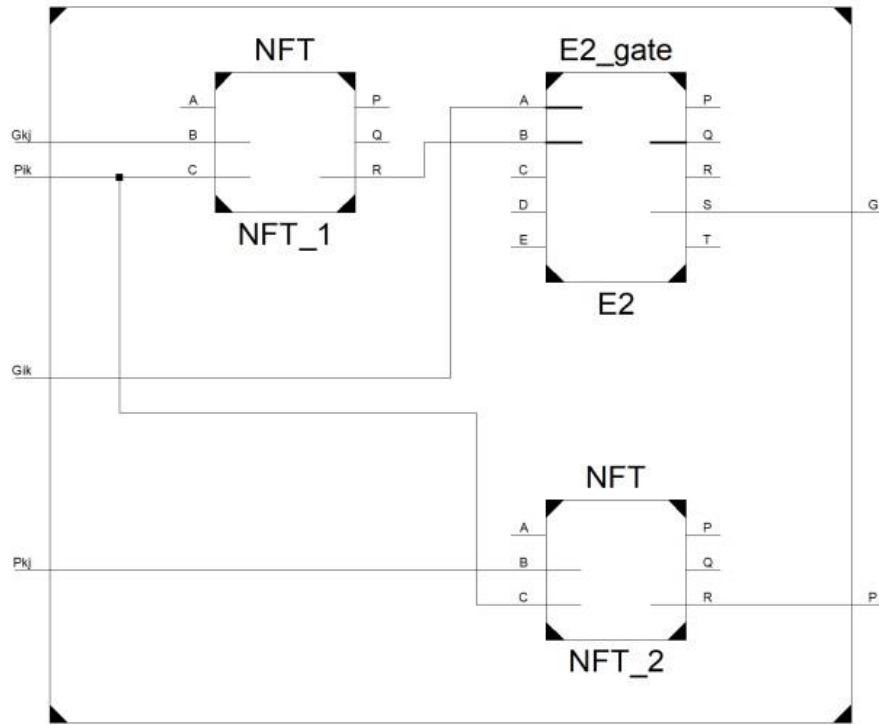


Fig. 4.16 Large circular block's RTL Schematic

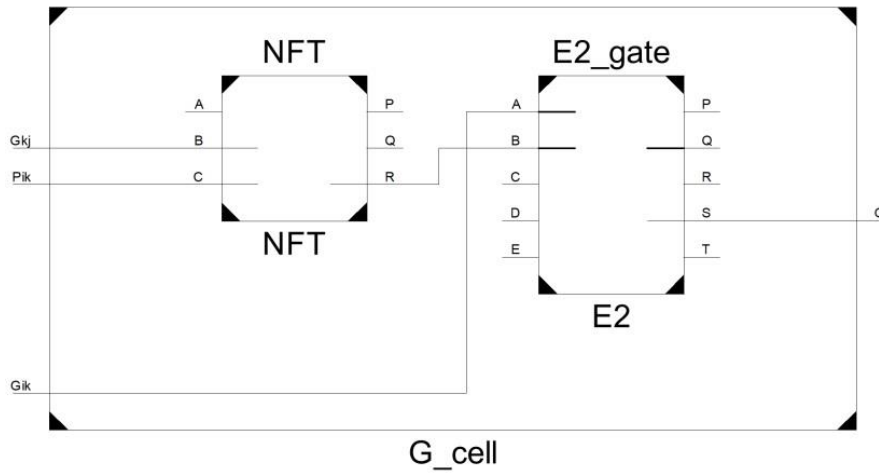


Fig. 4.17 Small circular block's RTL Schematic

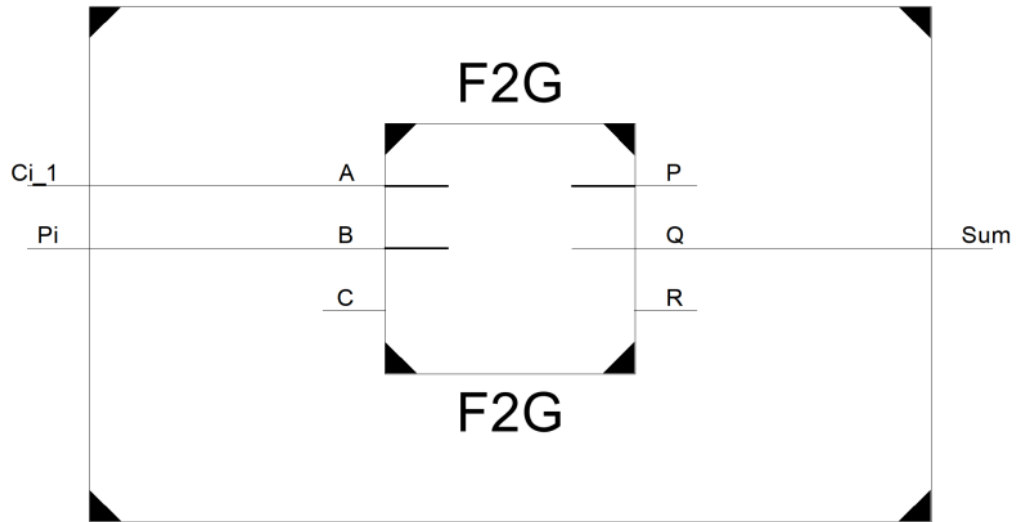


Fig. 4.18 Triangular Si block's RTL Schematic

CHAPTER 5

SIMULATIONS & RESULTS

The simulations of the all three proposed circuits mentioned earlier are done using Xilinx ISE 14.7 version environment on a computer, and the coding is done using Verilog HDL.

The functionality of the proposed Parity Preserving Reversible Arithmetic Unit, Logical Unit and complete Parity-Preserving Reversible ALU by integrating both Arithmetic Unit and Logical Unit is verified, and Figures 5.1, 5.2, and 5.3 represent the results of the simulations.

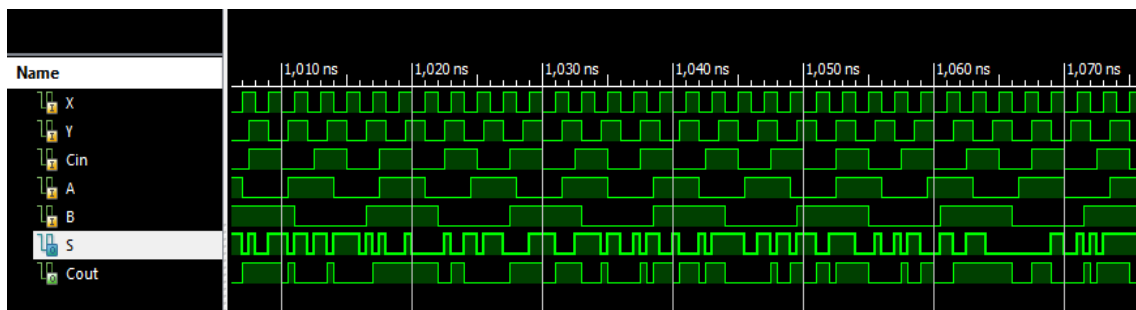


Fig. 5.1 Parity-Preserving Reversible Arithmetic Unit's Simulation Waveforms

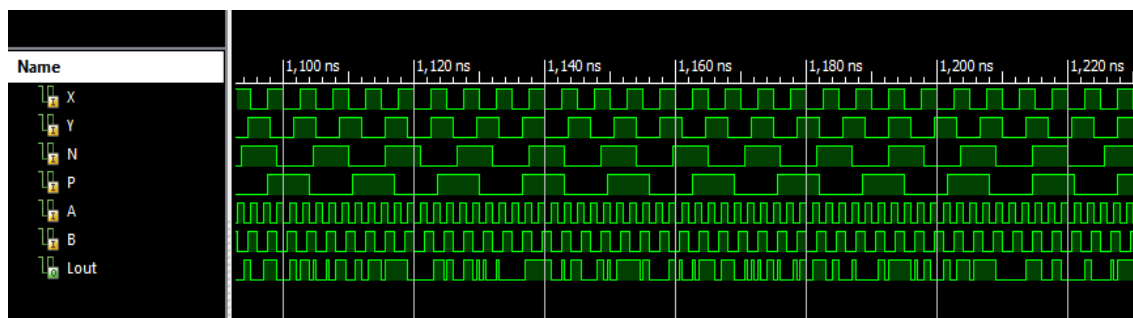


Fig. 5.2 Parity-Preserving Reversible Logical Unit's Simulation Waveforms

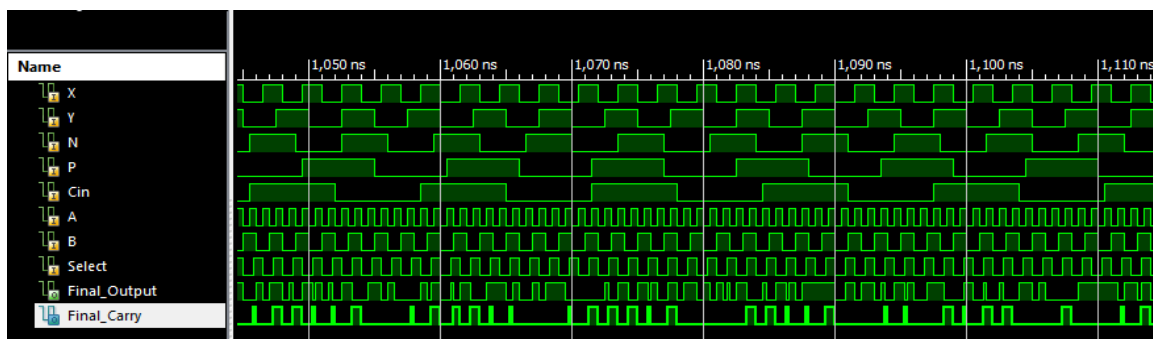


Fig. 5.3 Parity-Preserving Reversible ALU's Simulation Waveforms

The simulation waveforms reveal that the proposed ALU (Arithmetic and Logical Unit) gives accurate results for various values of inputs. Comparison Table 5.1 and 5.2 explain the comparison of our Proposed Parity Preserving Reversible Arithmetic Unit and Logical unit with Non-Parity Preserving Reversible Arithmetic Unit and Logical proposed in paper [9] and [10] respectively. This report is focused more on the Fault-Tolerant capability of ALU instead of reducing Quantum Cost, Ancilla Inputs (also known as Constant Inputs), Garbage Outputs and Gate Count. Results after comparison amongst the aimed Parity Preserving Reversible Arithmetic Logical unit and existing Non-Parity Preserving Reversible ALU [9][10] are as per the table 5.3. A comparative study carried out pertaining to the QC (Quantum Cost), Ancilla Inputs (Constant Inputs), Garbage Outputs, Gate Count and Fault-Tolerant capability. Simulation and Synthesis of Proposed 8-bit KGA and 16-bit BKA are also done using Xilinx ISE design suite version 14.7 by programming language Verilog and Figures 5.4 and figure 5.5 illustrate the results of simulations to confirm the functionality of the suggested adder respectively.

TABLE 5.1 Comparison of Arithmetic Unit

| Features Compared | [5] | [6] (Arithmetic Unit B) | Proposed Arithmetic Unit |
|--|-----------------------|-----------------------------------|---------------------------------|
| Parity Preserving or Non-Parity Preserving | Non-Parity Preserving | Non-Parity Preserving | Parity Preserving |
| Quantum Cost | 32 | 13 | 22 |
| Constant Inputs | 1 | 2 | 6 |
| Garbage Outputs | 6 | 5 | 11 |
| Gate Count | 5 | 4 | 5 |

TABLE 5.2 Comparison of Logical Unit

| Features Compared | [5] | [6] (Logical Unit B) | Proposed Logical Unit |
|--|-----------------------|--------------------------------|------------------------------|
| Parity Preserving or Non-Parity Preserving | Non-Parity Preserving | Non-Parity Preserving | Parity Preserving |
| Quantum Cost | 62 | 38 | 34 |
| Constant Inputs | 2 | 11 | 9 |
| Garbage Outputs | 14 | 16 | 18 |
| Gate Count | 9 | 14 | 10 |

TABLE 5.3 Comparison of complete ALU

| Features Compared | [5] | [6] | Proposed Logical Unit |
|--|-----------------------|-----------------------|------------------------------|
| Parity Preserving or Non-Parity Preserving | Non-Parity Preserving | Non-Parity Preserving | Parity Preserving |
| Quantum Cost | 94 | 59 | 76 |
| Constant Inputs | 3 | 14 | 16 |
| Garbage Outputs | 20 | 24 | 33 |
| Gate Count | 14 | 20 | 17 |



Fig. 5.4 Verilog HDL Simulation waveforms of Proposed Parity Preserving Reversible Kogge-Stone Adder

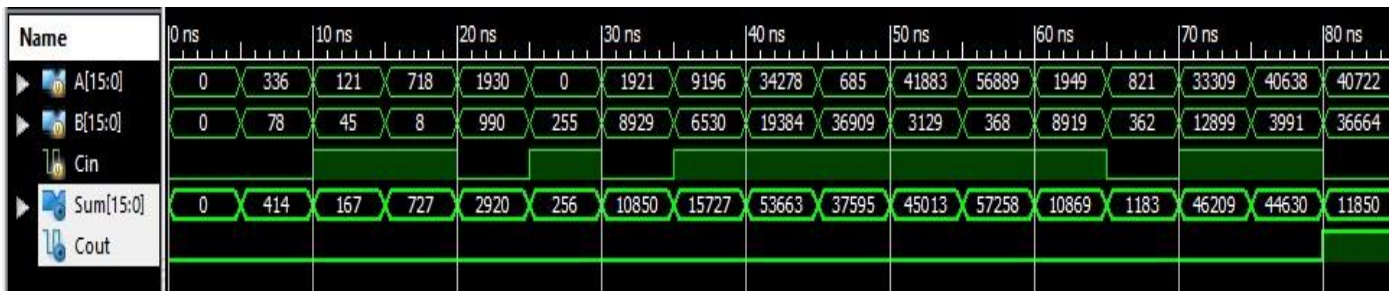


Fig. 5.5 Proposed Brent Kung adder simulation waveforms

The results of the simulation illustrate that the proposed Kogge Stone adder provides accurate results for different input values.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

As outlined herein, we have presented various Parity-Preserving Reversible circuits in this report. We suggested a Parity-Preserving Reversible Arithmetic and a Logical Unit. We have carried out a comparative study of the recommended design of ALU with the one already existing in paper [10][9]. The suggested Parity Preserving Reversible ALU has the capability of Fault- Tolerant, but at the cost of QC, Constant Inputs and Garbage Outputs. In our upcoming work, we intend to find an excellent proposed Parity Preserving Reversible ALU, with more efficient with regards to Constant Inputs, QC (Quantum Cost), Gate Count, Garbage Outputs and also with more functionalities. Further, this proposed Arithmetic and Logical Unit can be put to use in designing the CPU processor and also in various devices [17][32] and one can design the whole architecture employing the Parity Preserving Reversible Logic Gates in future. An optimized design in relations to performance, power dissipation and having fault-tolerant capability features, a system one can be designed.

We've proposed in this report Parity Preserving Reversible KSA. Proposed Parity Preserving Reversible KSA has the capability of Fault-Tolerant but at the cost of large complex circuitry. We intend to find an outstanding proposed Parity Preserving Reversible Kogge Stone adder for future work, with more improvements in terms of Constant Inputs, QC, Garbage Outputs, Gate Count and also with better performance. Further, this proposed Kogge Stone adder can be used to build the processor of a CPU, and in the future, the PPRLG can be used to create the complete architecture. A better in terms of speed, size area, power and having fault-tolerant capability features, a system one can be designed.

In this research, another design Fault-tolerant Brent Kung Adder is presented which is novel one. The Proposed Adder design described has the capability for fault tolerance, faster in speed, area is less, and wiring load is also less. This adder is faster, less computation nodes, area is less and wiring load is also less. All qualities of the proposed adder design can all be beneficial to design any system which require Performance, size, power, and fault-tolerant capabilities.

LIST OF PUBLICATIONS

- 1) Design of Parity Preserving Arithmetic and Logic using Reversible Logic Gates – Published in CONIT 2021, Scopus Indexed - Co Sponsored by IEEE Bangalore Section (DOI: 10.1109/CONIT51480.2021.9498539)
- 2) A Novel Low Power Design of Brent Kung Adder Having Fault Tolerant Capability – Accepted in SPIN 2021, Scopus Indexed and IEEE Xplore



International Conference on Intelligent Technologies (CONIT 2021)

25th – 27th June 2021



Certificate

This is to certify that Dr./Prof./Mr./Ms. Sumit Pahuja Affiliation Delhi Technological University, Delhi has presented Design of Parity Preserving Arithmetic and Logic Unit using Reversible Logic Gates in 2021 International Conference on Intelligent Technologies (CONIT) during 25th to 27th June 2021.

Dr. Basavaraj Anami
General Chair

Dr. Yerriswamy T
Convener

REFERENCES

- [1] C. L. Ayala, N. Takeuchi, Y. Yamanashi, T. Ortlepp and N. Yoshikawa, "Majority-Logic-Optimized Parallel Prefix Carry Look-Ahead Adder Families Using Adiabatic Quantum-Flux-Parametron Logic," in IEEE Transactions on Applied Superconductivity, vol. 27, no. 4, pp. 1-7, June 2017, Art no. 1300407,
- [2] G. V. Nikhil, B. P. Vaibhav, V. G. Naik and B. S. Premananda, "Design of low power barrel shifter and vedic multiplier with kogge-stone adder using reversible logic gates," 2017 International Conference on Communication and Signal Processing (ICCSP), 2017, pp. 1690-1694
- [3] Yeap, Gary K " Practical Low Power Digital VLSI Design", 1998
- [4] Gassoumi, L. Touil and B. Ouni, "Division circuit using reversible logic gates," 2018 International Conference on Advanced Systems and Electric Technologies (IC_ASET), 2018, pp. 60-65
- [5] R. Landauer, "Irreversibility and heat generation in the computing process," in IBM Journal of Research and Development, vol. 44, no. 1.2, pp. 261-269, Jan. 2000
- [6] C. H. Bennett, "Logical Reversibility of Computation," in IBM Journal of Research and Development, vol. 17, no. 6, pp. 525-532, Nov. 1973
- [7] A. Be'rut, A. Arakelyan, A. Petrosyan, S. Ciliberto, R. Dillenschneider, E. Lutz, "Experimental verification of Landauer's principle linking information and thermodynamics", 2012
- [8] A. Deeptha, D. Muthanna, M. Dhriti, M. Pratiksha and B. S. Kariyappa, "Design and optimization of 8-bit ALU using reversible logic," 2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2016, pp. 1632-1636
- [9] P. Khatter, N. Pandey and K. Gupta, "An Arithmetic and Logical Unit using Reversible Gates," 2018 International Conference on Computing, Power and Communication Technologies (GUCON), 2018, pp. 476-480
- [10] A. Kamaraj and P. Marichamy, "Design and implementation of arithmetic and logic

unit (ALU) using novel reversible gates in quantum cellular automata," 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), 2017, pp. 1-8

- [11] E. Pour, A. Akbar, K. Navi, M. Haghparast, M. Reshadi "Novel Optimum Parity-Preserving Reversible Multiplier Circuits" Circuits, Systems, and Signal Processing, March 2020
- [12] B. Koyada, N. Meghana, M. O. Jaleel and P. R. Jeripotula, "A comparative study on adders," 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), 2017, pp. 2226-2230
- [13] M. Morrison, M. Lewandowski, R. Meana and N. Ranganathan, "Design of a novel reversible ALU using an enhanced carry look-ahead adder," 2011 11th IEEE International Conference on Nanotechnology, 2011, pp. 1436-1440
- [14] G. V. Nikhil, B. P. Vaibhav, Vishnu G. Naik and B. S. Premananda, "Design of Low Power Barrel Shifter and Vedic Multiplier with Kogge-Stone Adder Using Reversible Logic Gates", 2017 International Conference on Communication and Signal Processing (ICCSP), 2017, pp. 1690-1694
- [15] Bharani Surya S., Gokul Prasad C., Raghul S., Mohankumar N. "Evolving Reversible Fault-Tolerant Adder Architectures and Their Power Estimation" in: Bindhu V., Chen J., Tavares J. (eds) International Conference on Communication, Computing and Electronics Systems. Lecture Notes in Electrical Engineering, vol 637. Springer, Singapore, 2020
- [16] Babu, H.M.H, "Reversible Fault-Tolerant Adder Circuits in Reversible and DNA Computing", H.M.H. Babu (Ed.): 2020, pp. 177-185
- [17] M. S. Islam, M. M. Rahman, Z. Begum and M. Z. Hafiz, "Fault tolerant reversible logic synthesis: Carry look-ahead and carry-skip adders," 2009 International Conference on Advances in Computational Tools for Engineering Applications, 2009, pp. 396-401
- [18] Y. d. Ykuntam, K. Pavani and K. Saladi, "Design and analysis of High speed wallace tree multiplier using parallel prefix adders for VLSI circuit designs," 2020 11th

International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2020, pp. 1-6

[19]Perkowski, M., Jozwiak, L., Kerntopf, P., A. Mishchenko, Anas Al-Rabadi, Alan Coppola, A.Buller, Xian Song, Mozammel Khan, Svetlana Yanushkevich, V. Smerko, M. Chrzanowska-jeske "A general decomposition for reversible logic", 2001

[20]Perkowski, M., Jozwiak, L., Kerntopf, P., A. Mishchenko, Anas Al-Rabadi, Alan Coppola, A.Buller, Xian Song, Mozammel Khan, Svetlana Yanushkevich, V. Smerko, M. Chrzanowska-jeske "A general decomposition for reversible logic", 2001

[21]B. Parhami, "Fault tolerant reversible circuits", in Proceedings of 40thAsimolar Conf. Signals, Systems,and Computers, Pacific Grove, CA, pp. 1726-1729, October 2006

[22]N. Przigoda, G. Dueck, R. Wille and R. Drechsler, "Fault Detection in Parity Preserving Reversible Circuits," 2016 IEEE 46th International Symposium on Multiple-Valued Logic (ISMVL), 2016, pp. 44-49

[23]M. Altun, S. Parvin and M. H. Cilasun, "Exploiting Reversible Computing for Latent-Fault-Free Error Detecting/Correcting CMOS Circuits," in IEEE Access, vol. 6, pp. 74475-74484, 2018

[24]R. Feynman, "Quantum mechanical computers", Optic News, Vol.11, Issue 2, (1985), pp. 11-20.

[25]Nielsen, M., Chuang, I, "Quantum computation and quantum information" Cambridge Univ. Press, 2000

[26]T. Toffoli, "Reversible computing" Lecture Notes in Computer Science, LNCS85 , pp. 632-644, 1980

[27]D. Maslov, G.W. Dueck and D.M. Miller, "Synthesis of Fredkin - Toffoli reversible networks", IEEE Trans. VLSI Systems, vol. 13, no. 6, pp. 765-769, 2005

[28]E. Fredkin and T. Toffoli, "Conservative logic", Intl. Journal of Theoretical Physics, pp. 219-253, 1982

[29]A. Peres, "Reversible logic and quantum computers", Physical Review A, Vol. 32, No. 6, (1985), pp.3266-3276.

[30]T. Toffoli, "Reversible computing" in Automata Languages and Programming,

Springer-Verlag, pp. 632-644, 1980.

- [31]S. Mamataj, B. Das, A. Rahaman, "An Ease implementation of 4-bit Arithmetic Circuit for 8 Operation by using a new reversible COG gate" International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol. 3, Issue 1, pp. 2278 – 8875, 2014
- [32]N. K. Misra, S. Wairya, B. Sen, "Design of conservative, reversible sequential logic for cost efficient emerging nano circuits with enhanced testability", pp. 2027-2037, 2018
- [33]J. B. Chacko and P. Whig, "Low Delay Based Full Adder/Subtractor by MIG and COG Reversible Logic Gate," 2016 8th International Conference on Computational Intelligence and Communication Networks (CICN), Tehri, 2016, pp. 585-589
- [34]S. S. Sudan and M. Singhal, "MIG and COG reversible logic gate based QSD addition/subtraction," 2017 International Conference on Computing, Communication and Automation (ICCCA), Greater Noida, 2017, pp. 1526-1531
- [35]G. Chen, X. Song, G. Yang, T. Wang, X. Mu and Y. Fan, "A Formal Proof of PG Recurrence Equations of Parallel Adders," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, August 2020
- [36]J. M. Rabaey, A. Chandrakasan, and B. Nikolic, Digital integrated circuits- A design perspective, 2ed ed. Prentice Hall, 2004.
- [37]C. L. Ayala, N. Takeuchi, Y. Yamanashi, T. Ortlepp and N. Yoshikawa, "Majority-Logic-Optimized Parallel Prefix Carry Look-Ahead Adder Families Using Adiabatic Quantum-Flux-Parametron Logic," in IEEE Transactions on Applied Superconductivity, vol. 27, no. 4, pp. 1-7, June 2017
- [38]David Harris, "A Taxonomy of Parallel Prefix Networks", pp. 0-7803-8104, 2003
- [39]A. Raju, R. Patnaik, R. K. Babu and P. Mahato, "Parallel prefix adders -A comparative study for fastest response," 2016 International Conference on Communication and Electronics Systems (ICCES), 2016, pp. 1-6.