Design and Synthesis of Novel Reversible Circuits for Next Generation Green Computing

Thesis submitted by

Mahamuda Sultana

Doctor of Philosophy (Engineering)

Department of Computer Science & Engineering
Faculty Council of Engineering and Technology
Jadavpur University
Kolkata, India

2019

JADAVPUR UNIVERSITY

KOLKATA – 700032, INDIA

INDEX NO. 136/17/E

Title of Thesis

Design and Synthesis of Novel Reversible Circuits for Next Generation Green Computing

Name, Designation & Institution of Supervisor:

Dr. Atal Chaudhuri
Professor,
Department of Computer Science and Engineering,
Jadavpur University,
Kolkata – 700032, India

List of Publications

International Journals:

1 Mahamuda Sultana, Ayan Chaudhuri, Diganta Sengupta and Atal Chaudhuri, “Toffoli Netlist and QCA Implementations for Existing Four Variable Reversible Gates – A Comparative Analysis”, Microsystem Technologies, Springer, (25), pp. 1987-2009, 2018. (SCI Indexed – Impact Factor 1.581)

2 Mahamuda Sultana, Diganta Sengupta, Atal Chaudhuri, “Taxonomy Proposal for Research on Reversible Logic”, International Journal of Engineering and Advanced Technology, Volume 8, Issue 6, pp-2608-2613, August 2019. (SCOPUS).

International Conferences:

3 Mahamuda Sultana, Ayan Chaudhuri, Diganta Sengupta and Atal Chaudhuri “Logic Design and Quantum Mapping of a Novel Four Variable Reversible s2c2 Gate”, 52nd Annual Convention of Computer Society of India (CSI 2017), January 19-21, 2018, Science City, Kolkata, Springer Nature CCIS Series, vol. 836, pp. 416-427. (SCOPUS)

4 Sinjini Banerjee, Priyabrata Sahoo, Mahamuda Sultana, Ayan Chaudhuri, Diganta Sengupta and Atal Chaudhuri “Toffoli Netlist Based Synthesis of Four Variable Reversible Functions”, in Proc. 2017 Third IEEE International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN 2017), November 03-05, 2017, RCC Institute of Information Technology, pp 315-320. (SCOPUS)

5 Mahamuda Sultana, Munna Prasad, Puja Roy, Soumya Sarkar, Suranjan Das and Atal Chaudhuri, "Comprehensive quantum analysis of existing four variable reversible gates," in 2017 Devices for Integrated Circuit (DevIC), IEEE, Kolkata, 2017, pp. 116-120. (This paper has been presented by me in the conference). (SCOPUS)

CERTIFICATE FROM THE SUPERVISOR

This is to certify that the thesis entitled **"Design and Synthesis of Novel Reversible Circuits for Next Generation Green Computing"** submitted by **Ms. Mahamuda Sultana**, who got her name registered on 27th March, 2017, for the award of PhD (Engg.) degree of Jadavpur University is absolutely based upon her own work under the supervision of **Prof. (Dr.) Atal Chaudhuri** and that neither her thesis nor any part of the thesis has been submitted for any degree or any other academic award anywhere before.

SIGNATURE OF THE SUPERVISOR WITH DATE AND OFFICIAL SEAL

_____

DR. ATAL CHAUDHURI

VICE-CHANCELLOR, VSSUT, BURLA, ODISHA

PROFESSOR-IN-LIEN, DEPARTMENT OF COMPUTER SCIENCE AND ENGG,

JADAVPUR UNIVERSITY,

KOLKATA – 700032, INDIA

DECLARATION

I hereby declare that the work described in this thesis is entirely my own. No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or institute. Any help or source information, which has been availed in the thesis, has been duly acknowledged.

Signature of the Candidate

_____

MAHAMUDA SULTANA

DATE:

Dedicated

to

My son, Ryne

& My Parents

ACKNOWLEDGEMENT

This thesis is not only a mark for my research but also a relationship with Jadavpur University for the past decade, more specifically, with Department of Computer Science and Engineering. I have been attached to Jadavpur University since my master's days. I admit that I feel proud to be a part of such a world class university. I have been constantly encouraged by the faculty members of Department of Computer Science and Engineering to explore different dimensions of research and know my capabilities as a researcher. In all these years, I have gained huge knowledge and support from a substantial amount of individuals who have left a mark in my life.

First and foremost, I would like to thank my research advisor, Prof. (Dr.) Atal Chaudhuri for the constant support and guidance for steering me out through the research. He has been instrumental in providing needed navigation in times of need, a huge source of inspiration in troubled times. He has been supportive in not only the selection of the research topic but also showing me the light at the end of the tunnel. He has taught me not only how to explore a domain but also to think different.

I would also like to express my sincere gratitude to my thesis committee, Prof. (Dr.) Mahantapas Kundu, Head, Dept. of CSE, Jadavpur University, Prof. (Dr.) Mita Nasipuri, member, Doctorate Committee, Jadavpur University, Prof. (Dr.) Susmita Ghosh, Prof. (Dr.) Sarmistha Neogy, Prof. (Dr.) Diganta Saha, and Prof. (Dr.) Sankhayan Choudhury, University of Calcutta for their valuable support in my endeavor. Apart from my thesis committee, I have gained huge help from Prof. (Dr.) Chitrita Chaudhuri in my days through my research. My growth as a researcher would not have been realized with their support and guidance.

I have been helped by dozens of people in the Department of Computer Science and engineering, Jadavpur University. I would like to thank Mr. Prabhat Chatterjee, Mr. Dilip Lodh, Mr. Kamal Kanti Ghoshal, Mr. Niloy Das, and Mr. Parimal Chakraborty for all the happy memories. I would take this opportunity to thank Department of Information Technology, Techno International New Town for providing me enough assistance to carry out my research unhindered.

I would like to conclude with due acknowledgement to my family. I would like to thank my parents Sk. Golam Mohobub and Mrs. Gulnahar Begum for educating me to see the light of this day. I would like to mention that my father being an academician always wanted me to achieve higher goals in education. I truly feel pride in realizing his dreams and gift him the pride of having a researcher as a daughter. I would like to thank my aunt Mrs. Jhumjhum Chaudhuri to constantly motivate me for research work. She would constantly encourage to pursue research work. I would like to thank my parent in laws, Mr. Dilip Kumar Sengupta and Mrs. Swapna Sengupta, my brother in law Munshi

Tanbir Ahammed, my sisters Dr. Mafruza Sultana, Mrs. Samadrita Banerjee, Mrs. Masuma Sultana, Ms. Marsina Sultana, and my brothers Dr. Pinaki Bahskar, Sk. Mustak Rased, Sk. Mamun Rased, and Ayan Chaudhuri for being there when I needed them the most. I would also like to thank all my family members to educate and grow me up to make this day happen. They always believed in me and were very supportive.

Finally, I would like to mention the name of my husband Dr. Diganta Sengupta, for his unfathomed support and guidance in times of crisis. Without his support, I would not have been the person that I am. Eventually, I would like to express my acknowledgement to my son, Ryne Sengupta, the very reason for my existence. Ryne has been naughty but has served me in learning to have more patience. I have learned that patience and perseverance is the main key to maturity, the one thing taught well by my son.

In my journey in the doctoral research, I have been helped by many notable individuals who have motivated me towards my goal. Some of them I have mentioned, and the rest remain in my mind for the rest of my life.

_____

MAHAMUDA SULTANA

DATE:

Title of the Thesis

Design and Synthesis of Novel Reversible Circuits for Next Generation Green Computing

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

1 INTRODUCTION

"Green Computing" is the "study and practice of Designing, Manufacturing, Using and Disposing of Computers, Servers and associated Sub-Systems". Modern CPUs are contributing at large to the heat dissipation with greenhouse effect booming big and atmospheric temperature soaring all-time high. On the other hand advancing technology demands faster clocking speed with CMOS reaching threshold capacities owing to increased chip densities. Taking all these issues into consideration, the target of Green Computing is to

- Save power
- Introduce reversibility or reversible brain in Nano-Technology
- Solve complex problems in polynomial time (range in THz).

Reversible Logic has the attribute to have near zero heat dissipation. Hence, reversible logic has the potential to promote Green Computing. Reversible logic forms that domain of computing which is both forward as well as backward deterministic. Therefore these systems form the basis for bidirectional computing as well as those domains where theoretical assessment of intermediate steps of multi-layer computing paradigm is concerned. Reversible computing can also be termed as information preserving system. This thesis discusses origins and the need for reversible computing and proposes a comparative analysis of existing reversible gates. The thesis also provides a novel four variable reversible gate, a synthesizing algorithm for reversible logic synthesis and designs for reversible decimal counters. The thesis concludes with taxonomy proposal for future researchers. The encircled part in Red in Figure 1.1 provides the area of study in this thesis.

1.1 THE NEED FOR REVERSIBILITY

Classical digital computers flourished in the mid-20^{th} Century with computer scientists concentrating both on the software as well as the hardware inventions. With passing time and need, hardware starting serving bottleneck for complex computations which needed speed, accuracy and reliability. Modern processors are based on CMOS. Recent research has evolved the notion of quantum computing but it is at an amateur stage. Digital circuits consume a huge amount of heat due to bit changes. More appropriately, every bit change from logic 1 to logic 0 and vice-versa account for $KTLog_e2$ Joules of dissipated heat; where *K=Boltzmann Constant* and *T=Absolute Temperature*. Although this had been pro-

posed by Landaeur [**1**] but has been recently proven in [**2**].

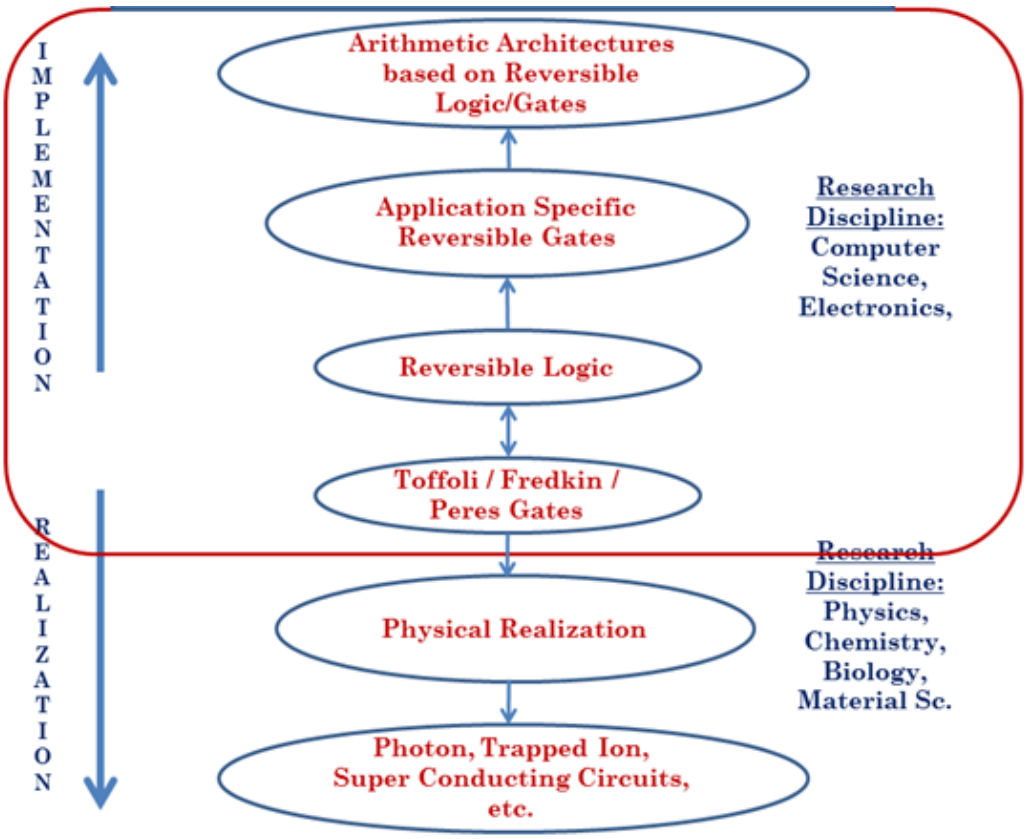


Figure 1.1. Area of research in this thesis

Large scale miniaturization and increase in chip densities aligned with Moore's Law (1965) have witnessed threshold limits in CMOS. Zhirnov et al. [**3**] has warned against rapid growth of future CMOS technology. His assessment is based on the power estimation using the "*2001 International Technology Roadmap for Semiconductors*" [**4**]. Bennett stated that heat generation can be arrested if the computation can be made reversible [**5**]. Thus, it can be believed that the concept of reversible logic was first conceived in 1973. The main motive for scientific interest in reversible logic can be assigned to multiple sources. The fundamental attribute for quantum computing is logical reversibility. Reversible Logic has gained prominence in quantum computing researchers. Another important constituent for motive is the power consumption attribute. As has already been stated that heat dissipation is associated with every bit change, hence reversibility gained importance in such conditions also. Due to its bidirectional nature, reversible computing forms an important constituent for program debugging, since the intermediate information transfer is lossless. For example, [**6**] reversible debugging provides watch-points in the reverse expressions. Reversible Logic finds usability in multiple other domains such as Quantum Dot

Cellular Automata [**7**] [**8**] [**9**] [**10**], SFL technology [**11**], Nano-Technology [**12**], Optical Technology [**13**], and Cryptography [**14**] [**15**]. Multiple proposals have been made for reversible gates, some are four variable, some three variable and some are two variable. The fundamental gate library of reversible logic comprises of Reversibility is implemented using Fredkin gate [**16**], Feynman Gate [**17**], Peres gates [**18**], and the Toffoli Gate [**19**],. A lot of definitions hold well in purview of reversible logic and allied domains. The author presents the Definitions in the next subsection and then explores further.

1.2 DEFINITIONS RELATED TO REVERSIBLE LOGIC

Definition 1: *If the input variable set is* $\{a_1, a_2, a_3 \dots \dots \dots a_n\}$*, then Multiple Control Toffoli (MCT) is defined as* ***TOFx(C; T)****;* $C \cap T = \emptyset$*; where C comprises of the set of Control Lines and T denoted the single target line which is being controlled by the control lines.*.

TOF2$\{x_1, x_2\}$ *is a* two variable Toffoli gate having x_1 as the one and only control line and x_2 as the target line. The target line gets inverted when x_1 is at Logic '1'. Hence, in *TOF2*, there exists only one control line and one target line. Two variable Toffoli gate is also known as the Feynman gate. Three input Toffoli gate is *TOF3*$\{x_1, x_2, x_3\}$. In generic terms, *TOF3* if referred to as the Toffoli Gate where x_1, x_2 and x_3 are the control and target lines respectively. The target line gets inverted when both the control lines x_1 and x_2 are at Logic '1'. In *TOF3*, there will be 2 control lines and one target line. Hence, *TOF1* has only one target line and zero control lines. In this case, since there is only one target line, it gets inverted without any control. Therefore *TOF1* is basically an inverter. When a signal passes through this gate, it always gets inverted.

Definition 2: If the input variable set is $\{a_1, a_2, a_3 \dots \dots \dots a_n\}$*, Fredkin gate is denoted as* ***FREx(C; T)****;* $C \cap T = \emptyset$*, where C =*$\{a_1, a_2 .. a_{j-1}, a_{j+1} .. a_{k-1}, a_{k+1} .. a_n\}$ *and T =* $\{a_j, a_k\}$*. Each input-output vector mapping is done for input sequence* $\{a_1, a_2, a_3 \dots \dots \dots a_n\}$ *and output sequence* $\{a_1, a_2 \dots a_{j-1}, a_k, a_{j+1} \dots a_{k-1}, a_j, a_{k+1} \dots a_n\}$ *if all the lines the Control Lines are at Logic '1'.*

In two input Fredkin gate, *FRE2*$\{x_1, x_2,\}$, in which the values of x_1 and x_2 are swapped at the output. Similarly in three input Fredkin gate, *FRE3*$\{x_1, x_2, x_3\}$, there are two target lines x_2 and x_3. The gate swaps the values of x_2 and x_3 when a high pulse is at x_1, otherwise the

values do not change at the output.

Definition 3: For a given variable set $\{a_1, a_2, a_3\}$, the Peres gate is denoted as a cascade of TOF3$\{x_1, x_2, x_3\}$ and TOF2$\{x_1, x_2\}$. Hence, Peres gate is a three inputs Toffoli Gate followed by two inputs Toffoli Gate.

Therefore Definition 1, Definition 2 and Definition 3 provide the operation principles of Toffoli Gate, Feynman Gate, Fredkin Gate and Peres Gate. Figure 1.2 presents the representations for the fundamental reversible gates.

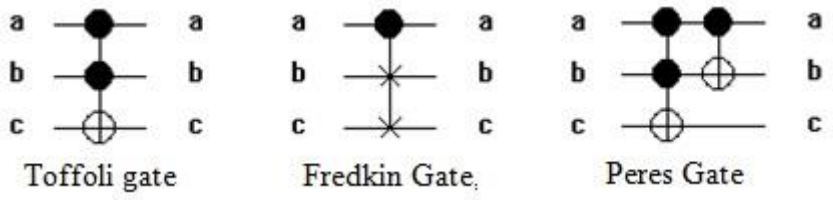


Figure 1.2. Fundamental reversible gates

Implementing Digital Logic gates, the outputs of the fundamental reversible gates described are as follows:

For an input variable set $\{A, B, C\}$, where A stands for the Most Significant Bit (MSB) and C stands for the Least Significant Bit (LSB), the output set $\{P, Q, R\}$ is given by the following expressions for the concerned gates:

Toffoli Gate: $\{P, Q, R\} = \{P = A, Q = B, R = (A.B) \oplus C\}$

Fredkin Gate: $\{P, Q, R\} = \{P = A, Q = \bar{A}B + A.C, R = A.B + \bar{A}C\}$

Peres Gate : $\{P, Q, R\} = \{P = A, Q = A \oplus B, R = (A.B) \oplus C\}$

Where '$\oplus$' stands for the XOR operation, '.' Stands for the AND operation, '$\bar{x}$'stands for the NOT operation on 'x', and '+' stands for the OR operation

Definition 4 provides the definition for reversible gates conforming to the concept of re-

versibility.

Definition 4: Given a Boolean specification $f: B^n \rightarrow B$ *is reversible if 'f' is bijective over a set of input variables* $X = \{x_1, x_2 \ldots\ldots x_n\}$. Given an Input-Output Vector $I_v = \{I_1, I_2 \ldots I_n\}$ and $O_v = \{O_1, O_2 \ldots O_n\}$ respectively, a reversible gate should possess the following attributes as per *Definition 4:*

The four major criteria for a function to be reversible is as follows.

- There exists unique values of O_v for each and every I_v.
- I_v-O_v relationship has a one-to-one Mapping and is Bijective.
- Branching of an output line is not allowed..
- No feedbacks connections are allowed in Reversible gates.

Definition 5: A reversible gate is termed as a Majority gate if it realizes a Majority Boolean Function (MBF). An MBF is a specification which has odd number of inputs and produces an output which has the majority votes among the inputs.

Illuatration:

Figure 1.13 presents the schematic representation of a Majority Gate. The Majority Gate is also known as Majority Voter. The final output $M = A.B + B.C + A.C = TRUE\ if\ any\ of\ the\ two\ inputs\ are\ TRUE$.

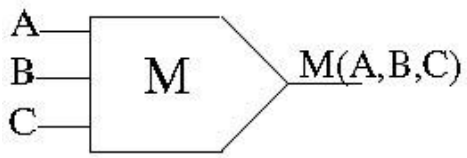


Figure 1.3. Majority Voter

Definition 6: The Quantum Cost (QC) is defined as the summation of individual quantum cost of each Toffoli gate in the Circuit (Toffoli Netlist). Therefore, $QC = \sum_{i=1}^{n} TOFx_i\ where\ TOFx_i\ is\ the\ cost\ of\ the\ i^{th} Toffoli\ Gate.$

The Quantum Costs of one input and two inputs Toffoli gate is 1 whereas that of three in-

put and four input Toffoli gates are 5 and 13 respectively.

Definition 7: The Hamming Distance between two bit streams is defined as the number of changes at subsequent bit positions in the bit streams and is denoted by δ(A, B). 'A' and "B" are the two bit streams.

Hence, the Hamming Distance is basically the number of bit dissimilarities between two bit streams.

Definition 8: In terms of reversible logic, the Complexity is defined as the summation of all the Hamming Distances and is given by $\sum_{i=0}^{m-1} \delta(A_i, B_i)$. *Here* A_i *and* B_i *are the input and output bit streams of the reversible function.*

Not all Boolean functions are reversible in nature. Hence, to make a non-reversible Boolean specification reversible, few input lines are required to be added and that also generates a few output lines which are redundant which are called Garbage outputs. The extra inputs are called as the Ancilla lines and the redundant outputs are known as Garbage outputs.

Definition 9: Conversion of non-reversible function into a reversible function can be done but at the cost of additional lines at the input, hence at the output. The additional lines at the input is termed as the Ancilla Lines and that at the output is termed as Garbage outputs. The values of the Ancilla lines is either of the two Boolean Logic, i.e. either '0' or '1'.

The quantum metrics that have been used to judge various reversible gates and proposals are provided in Definition 10.

Definition 10: Quantum Metric comprises of the following attributes.

- *Gate count in a Toffoli Netlist*
- *Quantum Cost of the circuit.*
- *Number of 2-qubit gates required to implement the design*
- *Number of QCA cells (4-Dot-2-Electron)*
- *Cell Area*

- *Majority Voter Count*
- *Inverter Count*
- *AND Gate Count*
- *OR Gate count, and*
- *Complexity of the Toffoli Netlist.*

Reversible models can be described in preliminary six ways – truth tables, Reed-Muller expansions, Decision diagrams, cycle expansions and matrix representations.

1.3 QUANTUM CIRCUIT COST

Multiple reversible logic synthesis and optimization algorithms have been proposed over date. All the models adhere to some or the other technology specific constraints and cost metrics. Definition 6 describes the Quantum Cost as a function of Toffoli Gates. This section describes the quantum circuit cost details. The quantum circuit cost majorly depends upon

1.3.1 SPEED

When the platform for a quantum computation is iron-trap based technology, then the approximation of computation runtime is termed as the speed of the quantum computation. Iron-trap technology uses laser pulses.

1.3.2 QUANTUM COST

It has already defined in Definition 6. It is the cumulative cost of all the Toffoli gates in a Toffoli cascaded architecture known as Toffoli Netlist. A number of publications have worked in quantum cost such as [**20**] [**21**].

1.3.3 NUMBER OF ANCILLA AND GARBAGE LINES

Definition 9 details the Ancilla and the Garbage outputs. Authors of [**22**] have proposed mechanisms to minimize the garbage count.

1.3.4 NUMBER OF 1-QUBIT GATES

TOF2 is also known as a CNOT gate. CNOT is a linear gate [**23**]. The 1-qubit gate count is the measure the non-linearity of the circuit becomes an important consideration for quan-

tum circuit cost. Inverter has been excluded from the non-linearity measure although it is a 1-qubit gate.

1.3.5 INTERACTION COST

Like Quantum Cost, Interaction cost is the summation of the individual interaction cost of a 2-qubit gate. Interaction cost is basically the distance between two gate qubits of 2-qubit gates.

1.3.6 DEPTH OF THE CIRCUIT

While designing a quantum circuit, the elementary gate count is calculated for any path. The largest count between the input and output accounts for the depth of the circuit. If any path can be invoked independently, then that becomes the depth if it has lower path.

1.4 REVERSIBLE LOGIC SYNTHESIS AND OPTIMIZATION

Reversible logic research has faced multiple dimensions, of which, synthesis algorithms have formed a major part.

Maslov et. Al. [**24**] have proposed synthesis algorithms to design circuits based on Fredkin-Toffoli networks. They have Unidirectional as well as Bidirectional algorithms for reversible logic synthesis. We have used the unidirectional algorithm to generate the Toffoli Netlist for four variable reversible gates, as will be discussed in the following chapters. We have also proposed a template based synthesis approach in [**25**]. Other template based synthesis algorithms are [**26**] [**27**] [**28**] [**29**] [**30**] [**31**] [**32**] [**33**] [**34**] [**35**]. Cycle based synthesis approach has been proposed in [**36**] where they design their algorithm on Reed-Muller spectrum using NCT library. Evolutionary algorithm based reversible circuit synthesis algorithm has been proposed in [**37**]. Reversible synthesis algorithm based on permutation cycle and group theory has been proposed in [**38**] [**39**] [**40**].

As discussed earlier, we provide the steps for Unidirectional Algorithm proposed in [**30**] [**24**] for reversible logic synthesis. This algorithm has been used to generate the Toffoli implementation in subsequent chapters of this thesis. The process has been elaborated for the MTSG Gate [**41**].

The block diagram for the MTSG gate is provided in Figure 1.4. The truth-table for The

MTSG gate is presented in Figure 1.5.

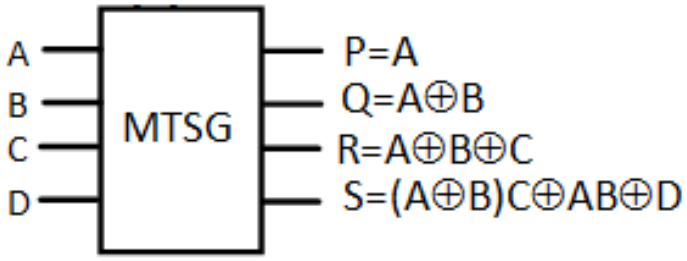


Figure 1.4. Block Diagram for MTSG gate

Input				Output			
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	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	1	0
1	0	0	1	1	1	1	1
1	0	1	0	1	1	0	1
1	0	1	1	1	1	0	0
1	1	0	0	1	0	0	1
1	1	0	1	1	0	0	0
1	1	1	0	1	0	1	0
1	1	1	1	1	0	1	1

Figure 1.5. Truth Table for MTSG Gate.

Figure 1.6 through Figure 1.11 present the subsequent steps for synthesis as per the Unidirectional Algorithm.

a0	b0	c0	d0
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	1	0
0	1	1	1
0	1	0	1
0	1	0	0
1	1	1	0
1	1	1	1
1	1	0	1
1	1	0	0
1	0	0	1
1	0	0	0
1	0	1	0
1	0	1	1
TOF2(b0,c0)			

Figure 1.6. Step 1 of Unidirectional Synthesis Algorithm

a1	b1	c1	d1
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	1
0	1	1	0
1	1	0	0
1	1	0	1
1	1	1	1
1	1	1	0
1	0	0	1
1	0	0	0
1	0	1	0
1	0	1	1
TOF3(b1,c1,d1)			

Figure 1.7. Step 2 of Unidirectional Synthesis Algorithm

a2	b2	c2	d2
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1
1	0	0	1
1	0	0	0
1	0	1	0
1	0	1	1
TOF2(a2,b2)			

Figure 1.8. Step 3 of Unidirectional Synthesis Algorithm

a3	b3	c3	d3
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	1
1	1	0	0
1	1	1	0
1	1	1	1
TOF3(a3,b3,d3)			

Figure 1.9. Step 4 of Unidirectional Synthesis Algorithm

a4	b4	c4	d4
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	1
1	1	1	0
TOF4(a4,b4,c4,d4)			

Figure 1.10. Step 5 of Unidirectional Synthesis Algorithm

a5	b5	c5	d5
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1
Completed			

Figure 1.11. Step 6 of Unidirectional Synthesis Algorithm

Algorithm elaboration:

- The PQRS output in Figure 1.5 has been termed as *a0b0c0d0* a0b0c0d0 in Figure 1.6.
- The blue columns in the Figure 1.6 through Figure 1.10 are the control lines, whereas and the columns in red are the target lines. For example

  *TOF2(**b0**, **c0**)* changes each bit of column **c0** in Figure 1.6 to column **c1** in Figure 1.7 depending upon **b0**; i.e. if value of b0 is high then the bits of c0 are inverted, else they remain unchanged. In this particular case, **a0** and **d0** are ignored.

  Similarly, *TOF3(**b1**, **c1**, **d1**)* changes column **d1** in Figure 1.7 to column **d2** in Figure 1.7 depending upon both **b1** and **c1** as this is a TOF3 gate. Hence, in TOF3 gate, there are two control lines denoted by b1 and c1, and one target line denoted by d1. The combined values of columns b1 and c1 affect the column d1. In this particular case, value for column **a1** is ignored.

- This is a recursive algorithm which continues till all the minterms are sequentially placed in the truth table starting from decimal 0 to decimal 15.
- The algorithm executes as per the given order.
  $f(0) to\ f(15), f(0) = \{0000\}, f(15) = \{1111\}, where\ f(i+1) = f(i) + 1; 0 \leq i < 15$

Hence, the final Toffoli Netlist for MTSG gate is provided in Figure 1.12. It has been generated using the Toffoli gates in reverse order from Figure 1.6 through Figure 1.10. It can be further observed that the Toffoli Netlist comprises of five Toffoli gates from the five steps in Figure 1.6 through Figure 1.10.

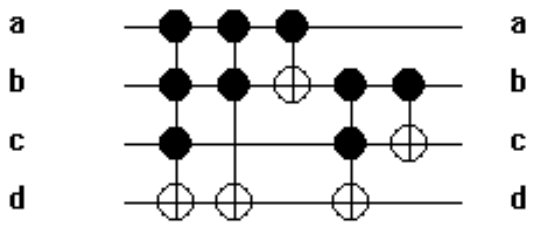


Figure 1.12. Toffoli Implementation for MTSG gate

Parallel to research on reversible synthesis algorithm, another domain of optimization is also important [**42**] [**43**] [**44**]. The Toffoli Netlist produced may or may not be optimized, where lack of optimization may lead to increased quantum metric cost seen earlier.

In this thesis, a state-of-the-art algorithm for optimization [**45**] [**46**] has been used according to the three rules provided below.

Rule 1:

Two adjacent Toffoli gates are considered. If the first gate and second gates have negative and positive control lines respectively at the same positions, then these two gates are replaced using a single gate. This replaced gate has no connection for that line. The process is shown in Figure 1.13.

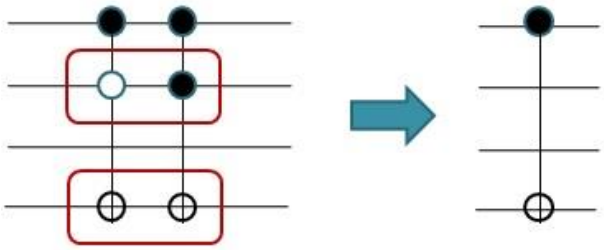


Figure 1.13. Explanation of Optimizing Rule 1

Rule 2:

Two adjacent Toffoli gates are considered. If the first gate and second gates have negative and Don't Care control lines respectively at the same positions, then these two gates are replaced using a single gate. This replaced gate has positive control for that line. The process is shown in Figure 1.14.

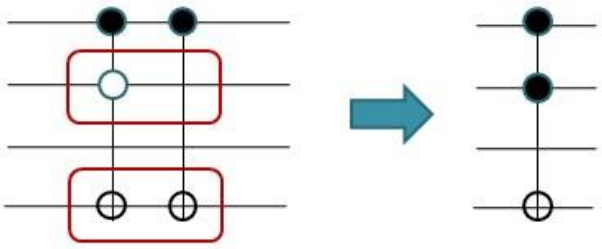


Figure 1.14. Explanation of Optimizing Rule 2

Rule 3:

Two adjacent Toffoli gates are considered. If the first gate and second gates have positive and Don't Care control lines respectively at the same positions, then these two gates are replaced using a single gate. This replaced gate has negative control for that line. The process is shown in Figure 1.15.

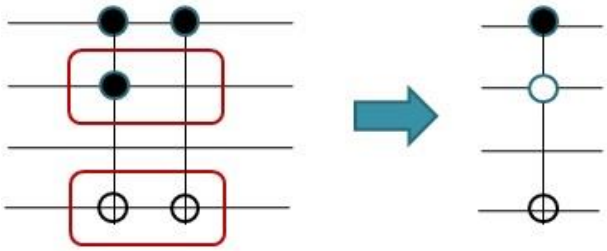


Figure 1.15. Explanation of Optimizing Rule 3

Illustrative example:

Applying the three rules of optimization to Figure 1.12, Figure 1.16 is obtained. It can be observed that Rule 3 is applicable as lines a, b, and d are exactly identical in Figure 1.12 for the first two Toffoli gates. Hence, after application of Rule 3, the first two Toffoli gates in Figure 1.12 are replaced by a single Tof4 gate in Figure 1.16. The single TOF4 (four input Toffoli Gate) contains a negative control line.

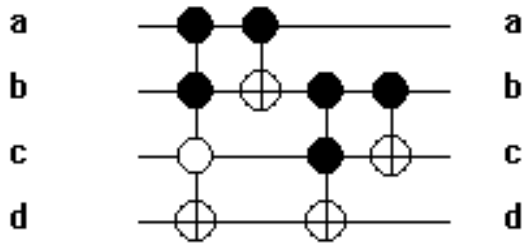


Figure 1.16. Optimized Toffoli Netlist for MTSG Gate

1.5 FOUR VARIABLE REVERSIBLE GATES

In quest for four variable reversible functions supporting one or other Boolean specifications have been proposed in literature. Of these, notable are RI [**47**], R2 [**47**], TSG [**48**],

HNG [**49**], SCG [**50**], RPS [**51**], HNFG [**52**] [**53**], MKG [**52**], IG [**54**], FAG [**55**], ALG [**56**], MTSG [**41**], DFG [**55**], DKG [**57**], MRG [**57**], BG-GB Gates [**58**], s2c2 [**59**], TCG [**60**] [**61**] and BVMF [**62**] gates. All of them have implemented some or the other Boolean specification as per need.

For example, most widely proposed Boolean specifications are Adders, Subtractors, Decoders, Parity Generators, Parity Checkers to name a few. Full adders have been implemented using the SCG gate, HNG gate, and s2c2 gate. Among these two, SCG gate and s2c2 gate also doubles as a subtractor. SCG and BVMF gates have been used to realize comparators. The BVMF gate also can realize multiplexers and decoders.

The TSG, FAG and MTSG gates can be used to realize 4:2 compressors. Preliminary Arithmetic and Logic Units have been proposed using the TSG, DKG, MRG and ALG gates. MKG and DFG gates have implemented reversible n-bit multipliers. Reversible counter proposals can be found in [**63**] [**64**] [**65**] [**66**] [**67**].

The rest of the thesis is organized as follows.

Chapter 2 details the comprehensive analysis of four variable reversible gates in literature. It does the analysis based on Quantum Metrics defined earlier in this chapter. The analysis has been published in [**68**].

Chapter 3 presents the four variable reversible s2c2 gate designed by the author. The gate is capable of implementing a full adder, full subtractor, AND, OR, NOT, and XOR operations. The findings have been published in [**59**].

Chapter 4 presents the template synthesis algorithm. The study used templates for reversible logic synthesis of four variable reversible gates. All the possible !16 four variable reversible functions can be synthesized using the algorithm. The study has been published in [**25**].

Chapter 5 provides the study on reversible synchronous decimal up/down counter. The chapter details three designs comprising of the reversible decimal up counter, reversible decimal down counter, and reversible decimal up/down counter. The work has been communicated in [**69**]

Chapter 6 provides a taxonomy on research on reversible logic. This study basically divides the multi-dimensional research on reversible logic into nodes sot that researchers do

not have to explore the massive literature on reversible and directly get a first-hand reference for their topic of interest. The taxonomy has been published in [**70**].

The thesis is concluded in Chapter 7 providing the conclusion and future scope of the work.

CHAPTER 2

Comparative Analysis of Four Variable Reversible Gates

Mahamuda Sultana, Ayan Chaudhuri, Diganta Sengupta and Atal Chaudhuri, "Toffoli Netlist and QCA Implementations for Existing Four Variable Reversible Gates – A Comparative Analysis", **Microsystem Technologies, Springer**, (25), pp. 1987-2009, 2018. (**SCI Indexed – Impact Factor 1.581**).

2 COMPARATIVE ANALYSIS OF FOUR VARIABLE REVERSIBLE GATES

Since CMOS has already started encountering threshold limits due to primarily two factors. The first factor is speed. With advancing technology and technological needs, time has already elapsed for the need for faster processors. CMOS has clocking speeds in GHz but recent complex algorithms demand speeds of range THz; something not achievable using CMOS. With large scale miniaturization, denser packaging of chips has contributed in enormous amount of heat generation, thereby constraining enhancement of clocking speeds. The second factor is heat generation. Heat generation can be limited using concepts of reversibility as has been already discussed in Chapter 1. Regarding clocking speed, Quantum Dot Cellular Automata provides a platform for implementation of reversible functions at THz speeds. But physical constraints restrict use of Quantum Dot Cellular Automata (QCA) for large scale implementation. None-the-less reversible functions are getting implemented using QCA at the research level. These implementations form the basis for part of Quantum Computers also as the inherent attribute of Quantum Computing requires the operations to be reversible.

Taking note of such technological advancements, several computer researchers have proposed four variable reversible gates till date. Each proposal has been motivated for implementation of certain Boolean specification, viz. Full Adder, Full Subtractor, Comparator, Decoder to name a few. Future proposals for four variable reversible gates are becoming complex as the researchers have to perform extensive literature study regarding existing proposals and also their value additions. This chapter provides a comprehensive survey of existing four variable reversible gates. Since, reversible gates must be implemented using fundamental reversible gates and/or QCA cells, all the existing gates have been realized using Toffoli gates as some of the proposals have lacked such considerations. This chapter provides the Toffoli implementations along with the details of the gates in terms of Quantum Metrics, viz. the quantum cost, number of gates, number of two qubit gates and the complexity of each gate. This chapter not only provides the Toffoli realizations, but also the QCA realizations for each four variable reversible gate. The chapter will serve as a benchmark for future proposals for both the domains – reversible Toffoli networks as well as design implementing Quantum Dot Cellular Automata. Regarding QCA implementations, the analysis has been done according the following attributes; QCA cell count, design area coverage, majority voter count, inverter count using majority voters, AND gates count using majority voter, OR gate count to implement the design using QCA.

While designing reversible architectures also, much ambiguity lies in selection of a certain gate. This chapter will serve as a ready reckoner as it provides all the basic metrics required to confirm a certain gate according to the requirement. The contents of this chapter have been published in [**59**] [**68**].

2.1 COMPARATIVE ANALYSIS OF FOUR VARIABLES REVERSIBLE GATES

This section provides the designs for the existing four variable reversible gates. The details have been presented in terms of block diagram of the gates, Toffoli Netlist implementation, Optimized Toffoli implementation, and QCA implementation. The Toffoli implementations [**26**] [**30**] and optimization [**45**] [**46**] have been done using the state-of-the-art algorithms explained in Chapter 1.

The design parameters used in the setup engine of QCA Designer are as follows:

QCA Cell Width	: 18nm
QCA Cell Height	: 18nm
Number of Samples	: 12800
Radius of effect	: 65nm
Relative permittivity	: 12.9
Clock (High, Low, and Shift)	: 9.800000e-022, 3.800000e-023, 0.00e+00
Layer Separation	: 11.500000
Maximum Iteration per sample	: 100
Clock Amplitude Factor	: 2
Convergence Tolerance	: 0.001

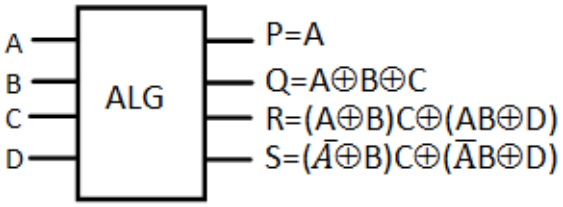
The simulation has been done using the software proposed in [**71**] known as the QCA Designer. The area of a cell is calculated by multiplying the QCA Cell width with the QCA Cell Height. This area has been used as a quantum metric attribute for comparative analy-

sis in Table 2.

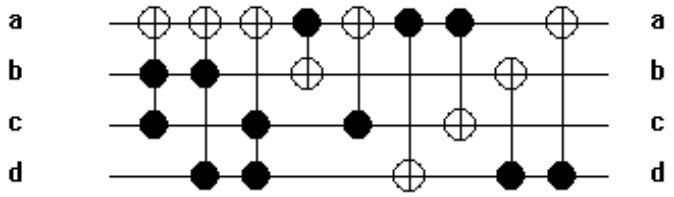
The simulation engine of QCA Designer has been set up using the parameters stated above. The high/low pulses have been used to accomplish the clock eliminating the shift between consecutive clocks. The initial sampling rate which has been used throughout the execution is 12800. When cell polarization is greater than the convergence tolerance, the sample completes. The maximum distance an electron affects another electron according to Coulomb's Law is called the radius of effect. The kink energy is calculated using the relative permittivity of a material.

The subsequent subsections provide the details.

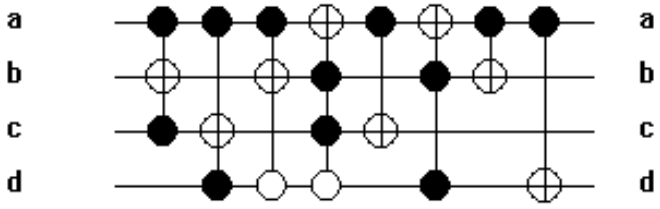
2.1.1 TOFFOLI NETLIST AND QCA IMPLEMENTATION OF ALG GATE



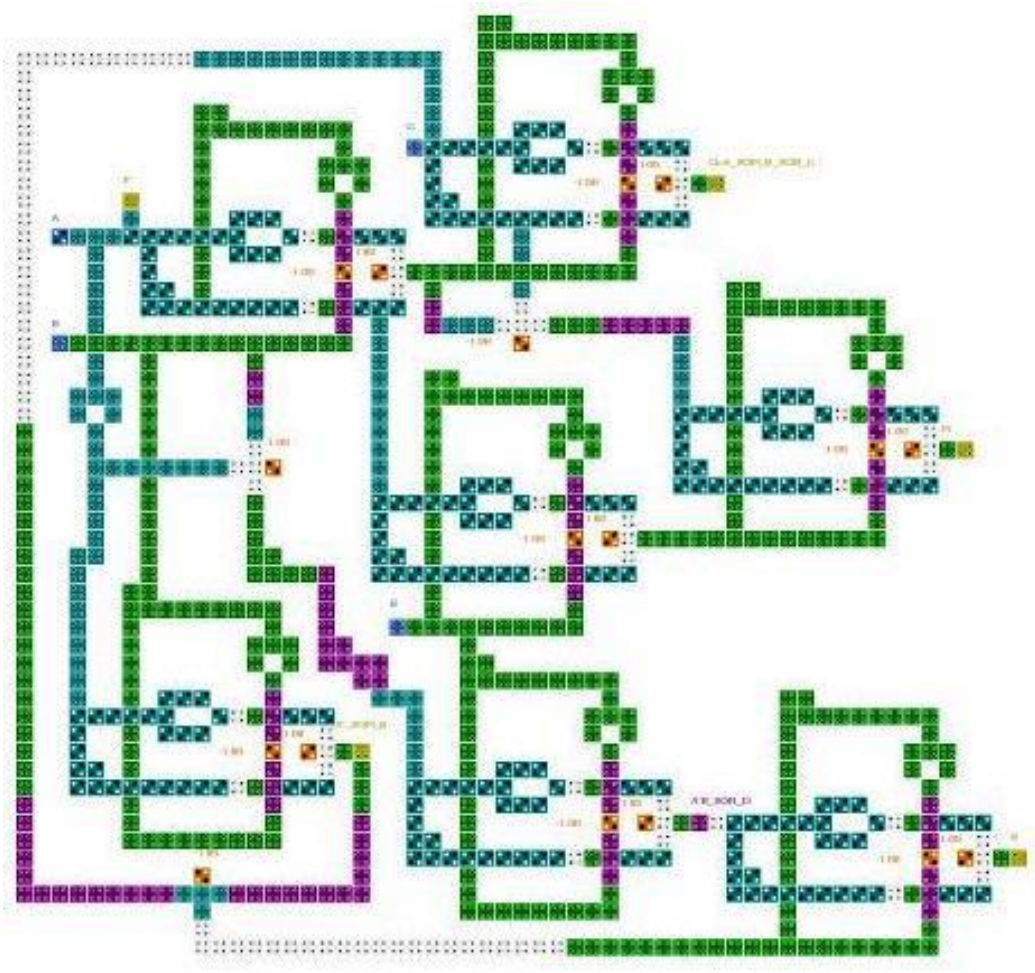
(i)



(ii)



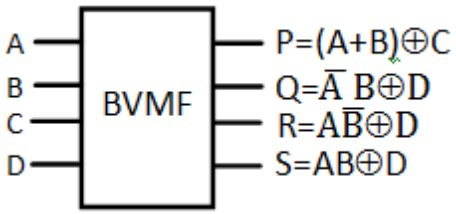
(iii)



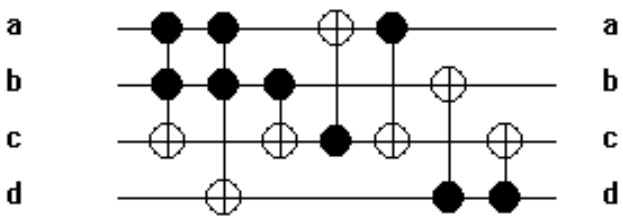
(iv)

Figure 2.1. (i) Block Diagram for ALG [**56**] Gate (ii) ALG-Toffoli Implementation (iii) Toffoli Netlist- Optimized (iv) ALG Gate - QCA realization

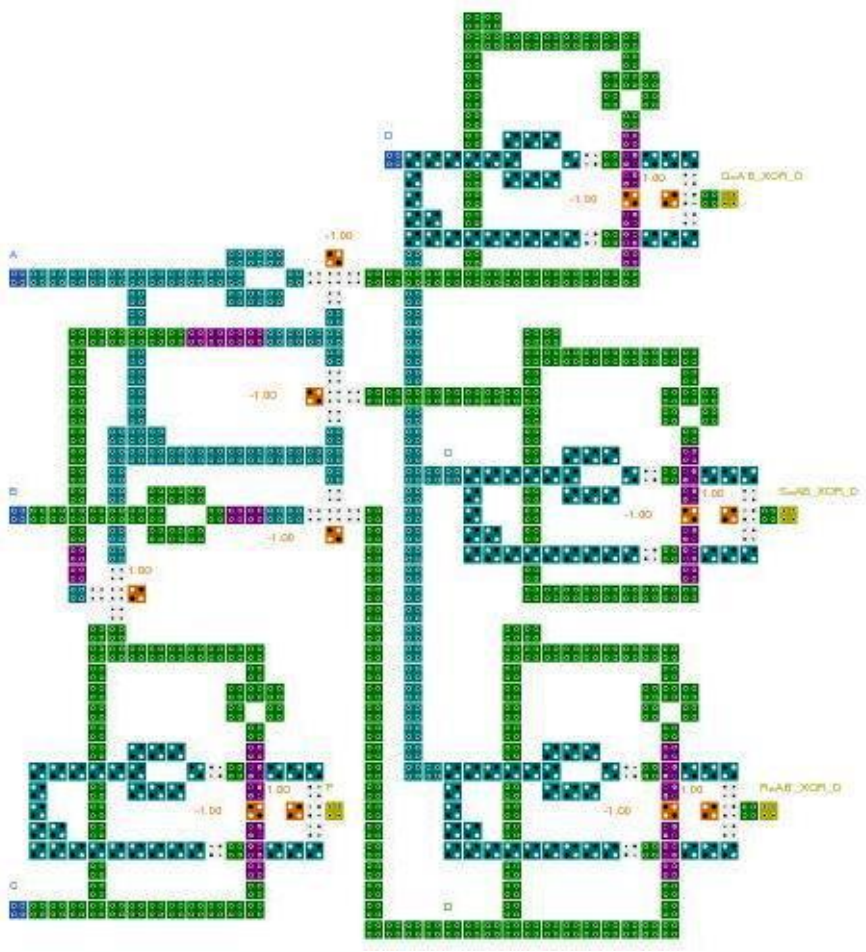
2.1.2 TOFFOLI NETLIST AND QCA IMPLEMENTATION OF BVMF GATE



(i)



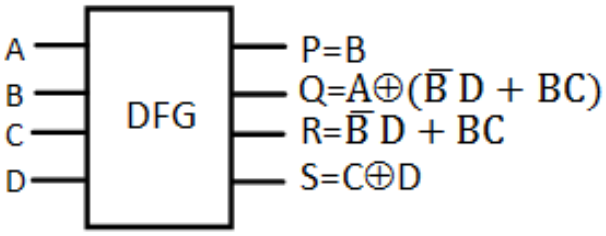
(ii)



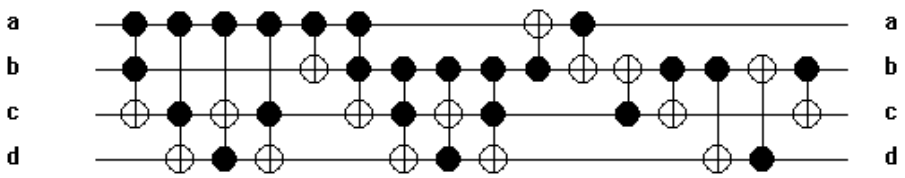
(iv)

Figure 2.2. (i) Block Diagram for BVMF [**62**] Gate (ii) BVMF-Toffoli Implementation (iii) BVMF Gate - QCA realization

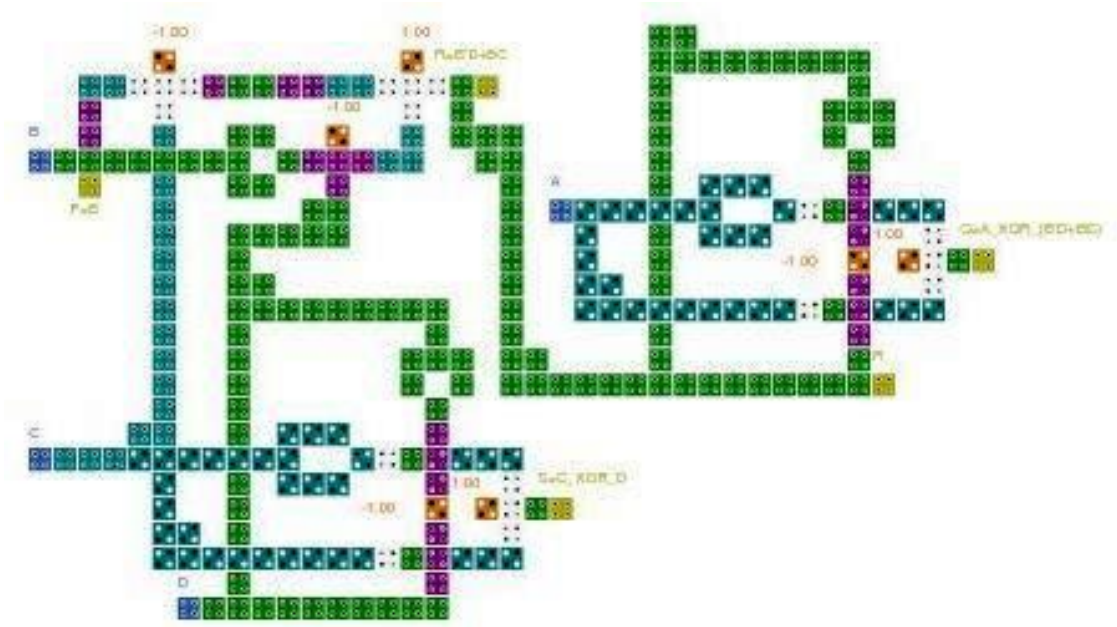
2.1.3 TOFFOLI NETLIST AND QCA IMPLEMENTATION OF DFG GATE



(i)



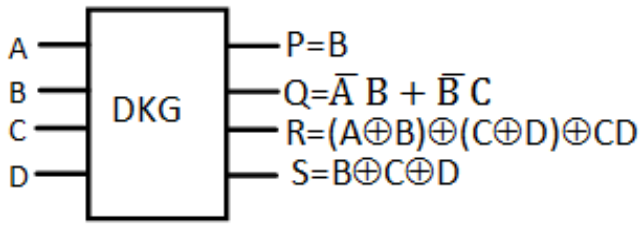
(ii)



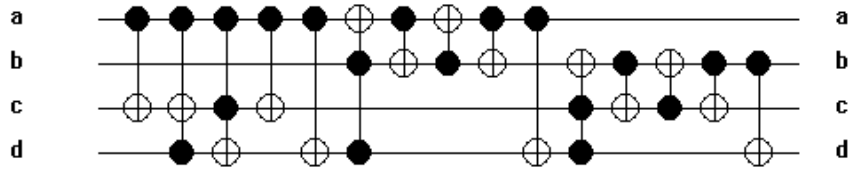
(iii)

Figure 2.3. (i) Block Diagram of DFG [**55**] Gate (ii) DFG - Toffoli Implementation (iii) DFG Gate - QCA realization

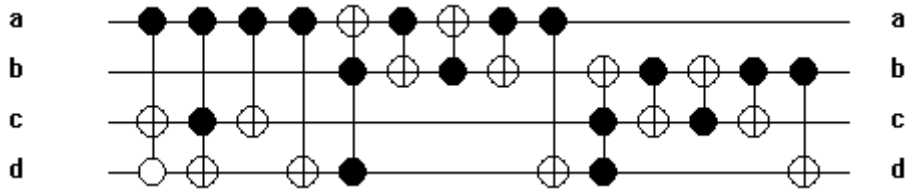
2.1.4 TOFFOLI NETLIST AND QCA IMPLEMENTATION OF DKG GATE



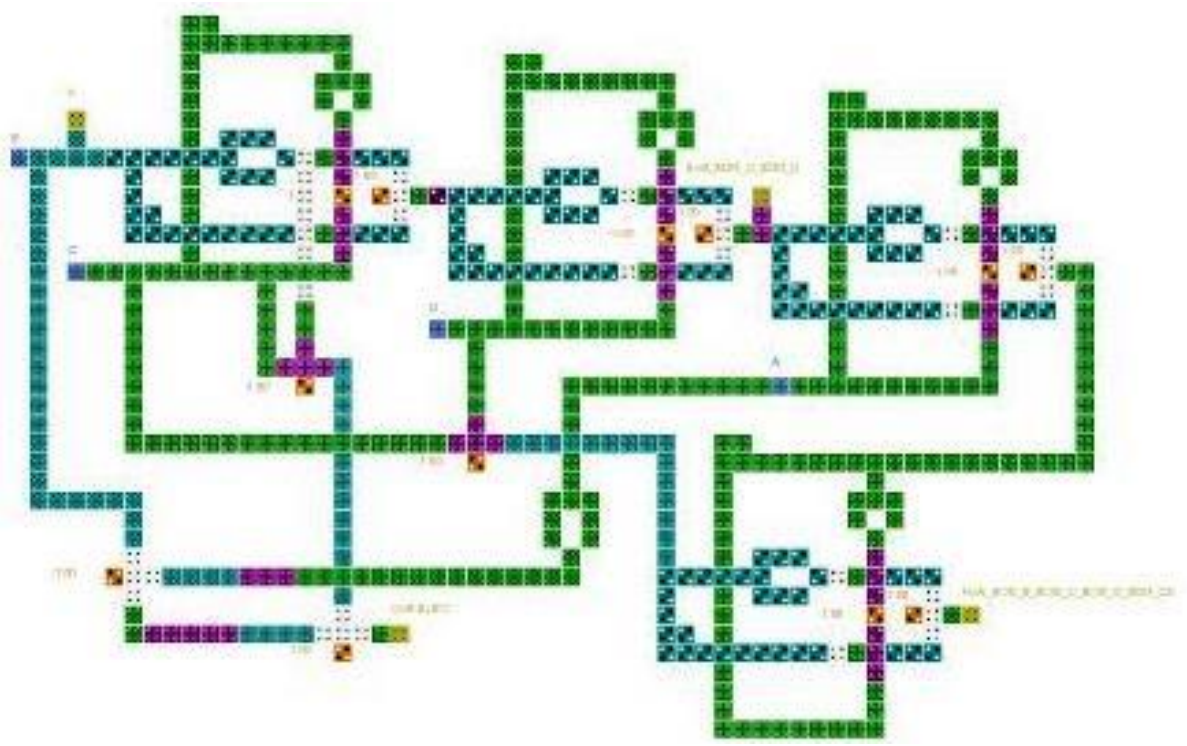
(i)



(ii)



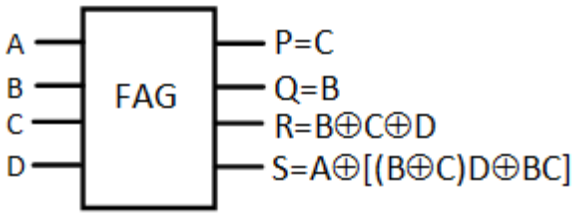
(iii)



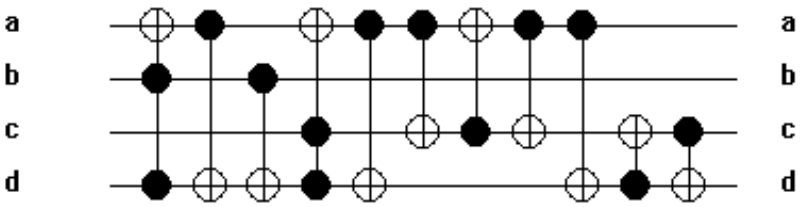
(iv)

Figure 2.4. (i) Block Diagram of DKG [**57**] Gate (ii) DKG-Toffoli Implementation (iii) Toffoli Netlist - Optimized (iv) DKG Gate - QCA realization

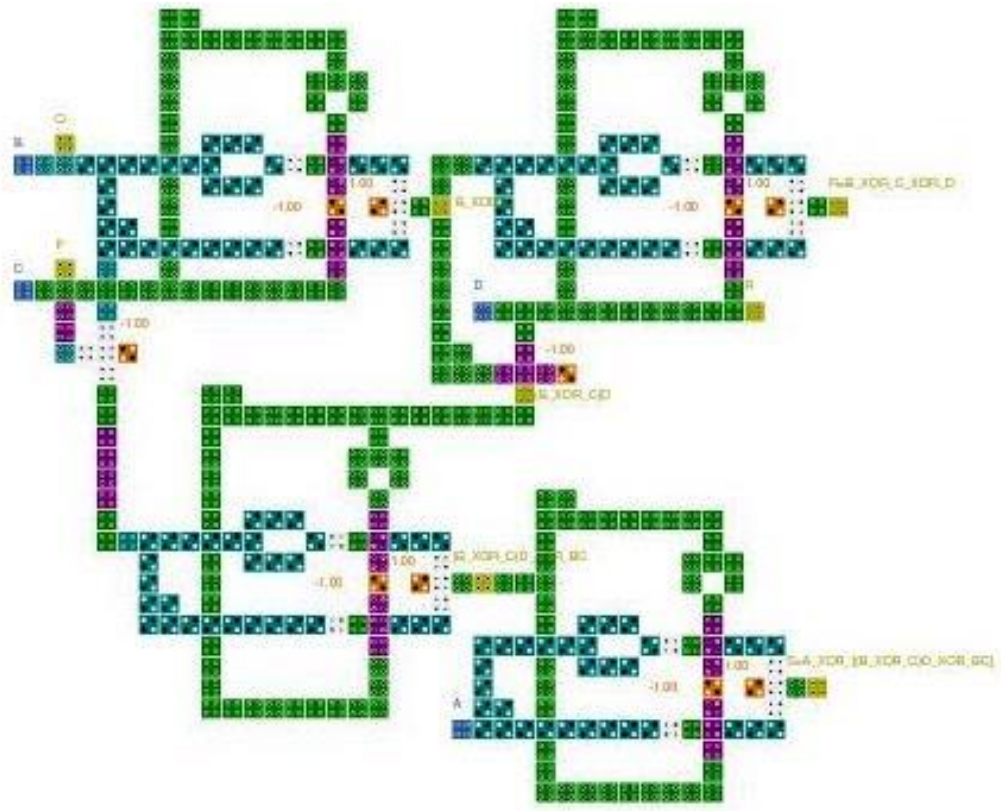
2.1.5 TOFFOLI NETLIST AND QCA IMPLEMENTATION OF FAG GATE



(i)



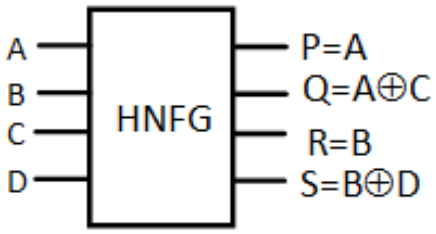
(ii)



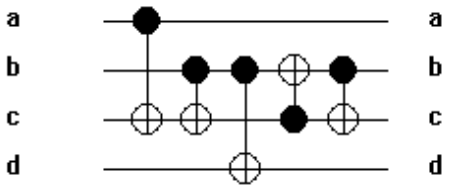
(iii)

Figure 2.5. (i) Block Diagram of FAG [**55**] Gate (ii) FAG - Toffoli Implementation (iii) FAG Gate - QCA realization

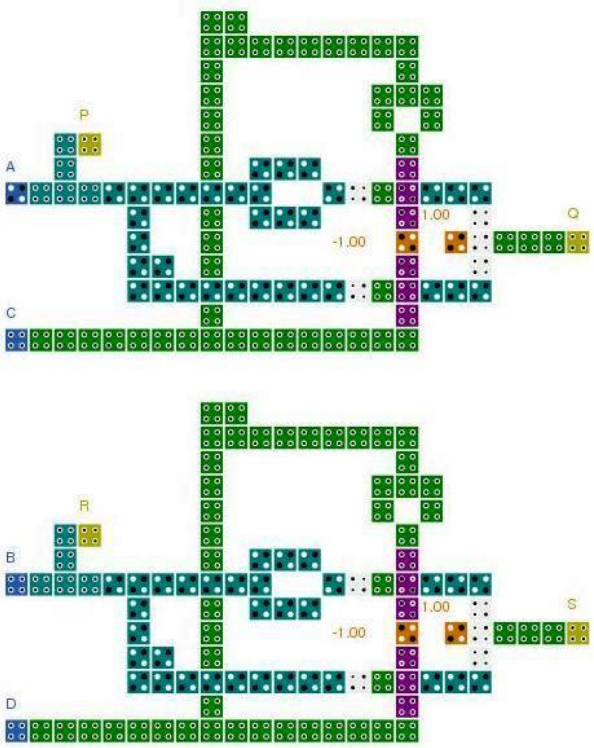
2.1.6 TOFFOLI NETLIST AND QCA IMPLEMENTATION OF HNFG GATE



(i)



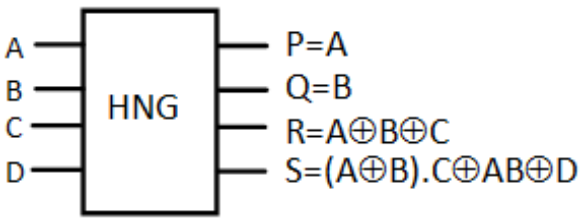
(ii)



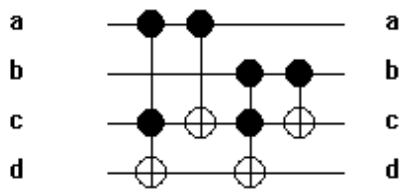
(iii)

Figure 2.6. (i) Block Diagram of HNFG [**52**] [**53**] Gate (ii) HNFG - Toffoli Implementation (iii) HNFG Gate - QCA realization

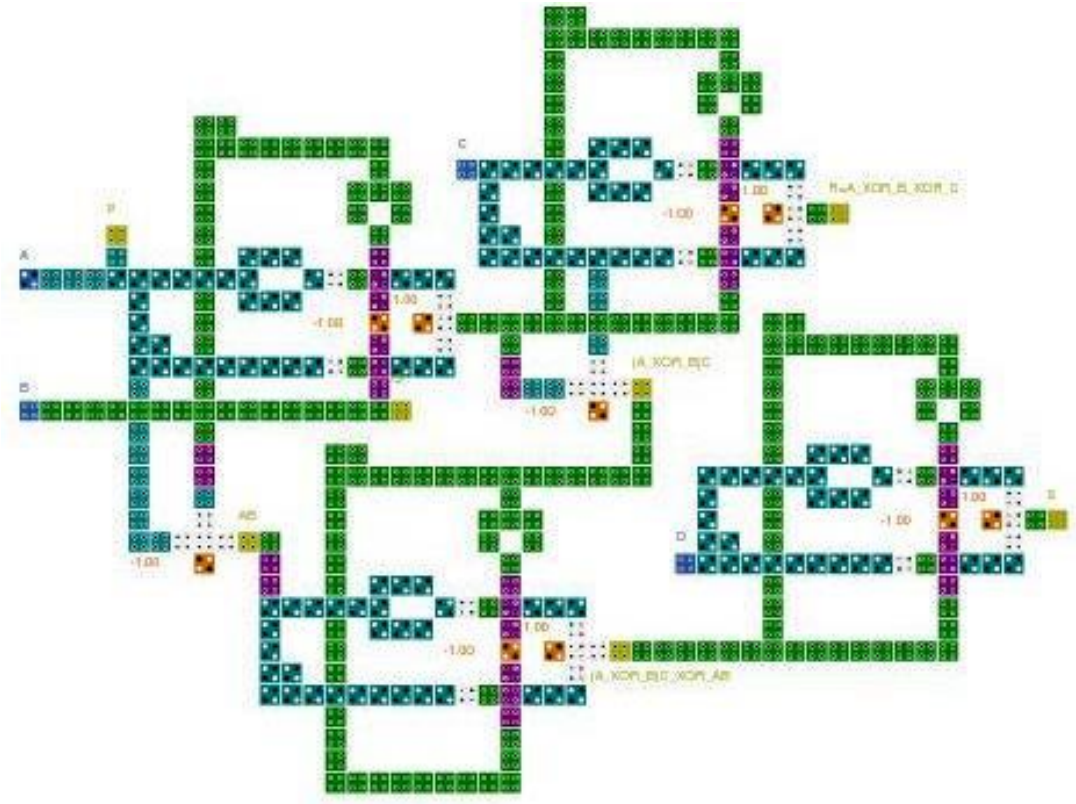
2.1.7 TOFFOLI NETLIST AND QCA IMPLEMENTATION OF HNG GATE



(i)



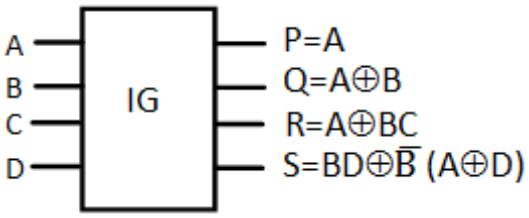
(ii)



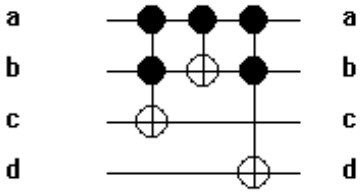
(iii)

Figure 2.7. (i) Block Diagram of HNG [**49**] Gate (ii) HNG - Toffoli Implementation (iii) HNG Gate - QCA realization

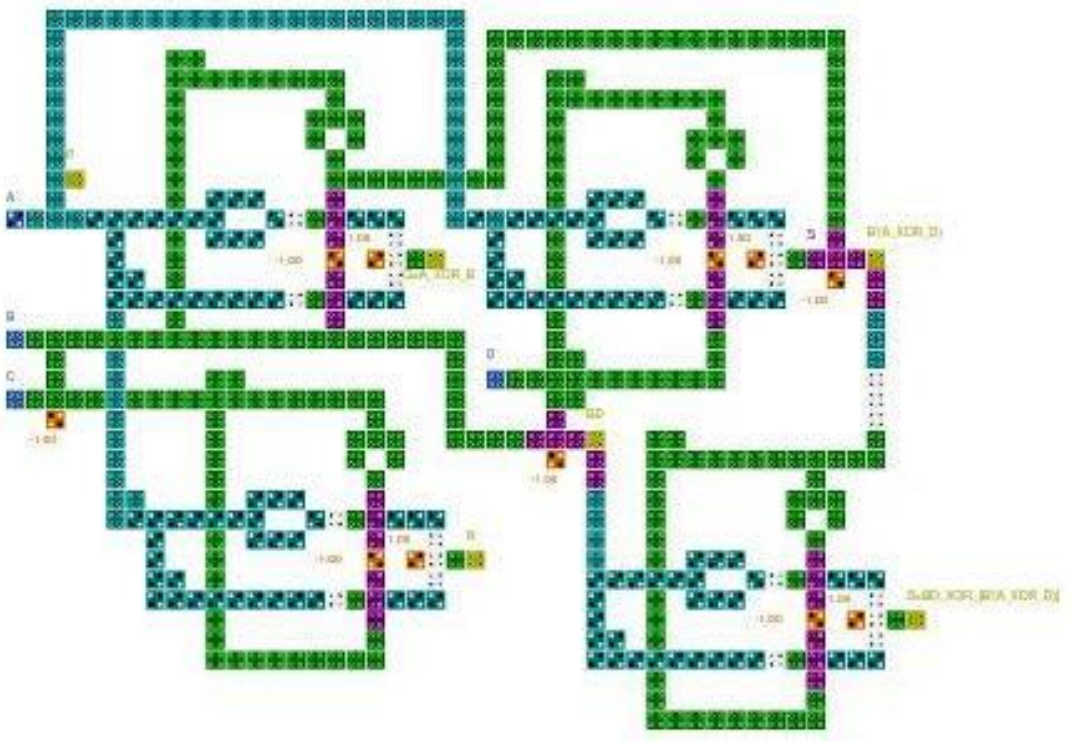
2.1.8 TOFFOLI NETLIST AND QCA IMPLEMENTATION OF IG GATE



(i)



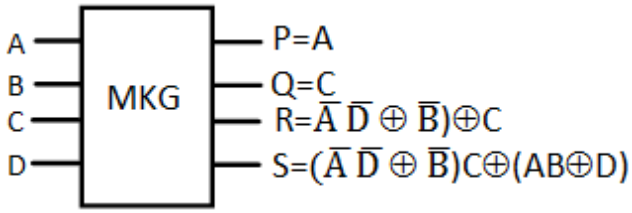
(ii)



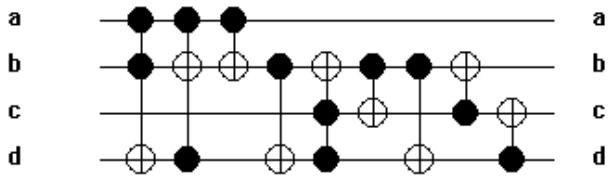
(iii)

Figure 2.8. (i) Block Diagram of IG [**54**] Gate (ii) IG - Toffoli Implementation (iii) IG Gate - QCA realization

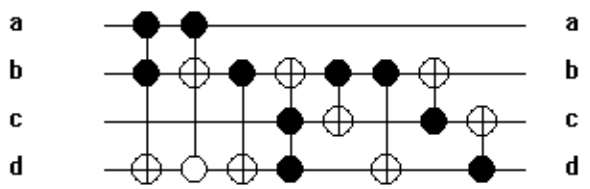
2.1.9 TOFFOLI NETLIST AND QCA IMPLEMENTATION OF MKG GATE



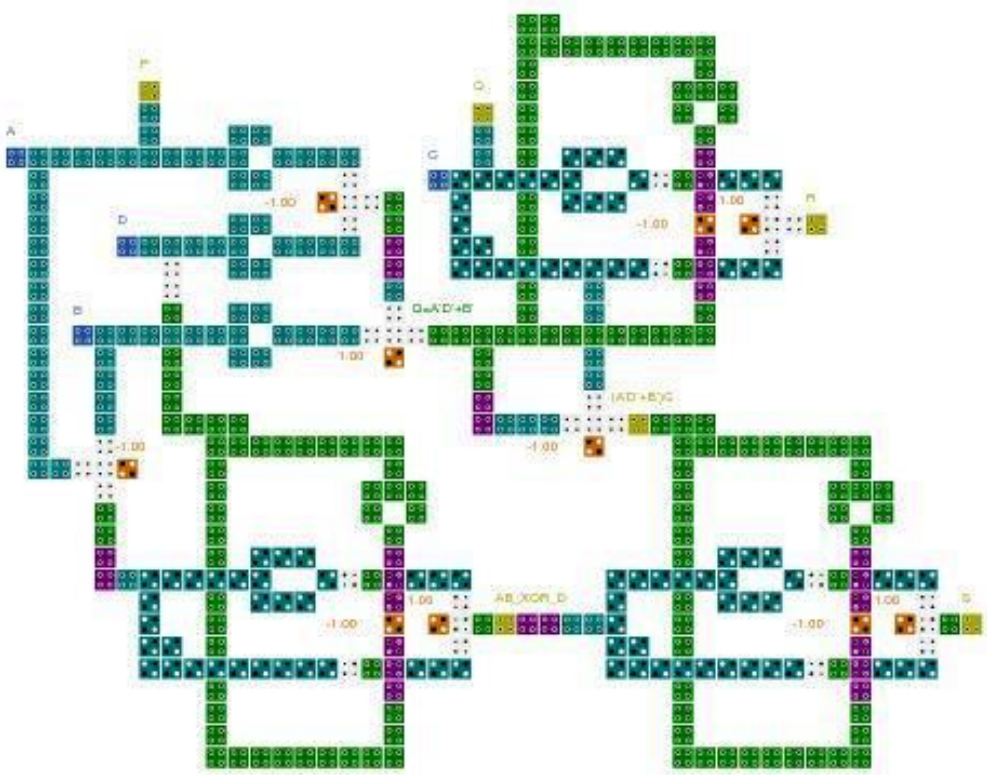
(i)



(ii)



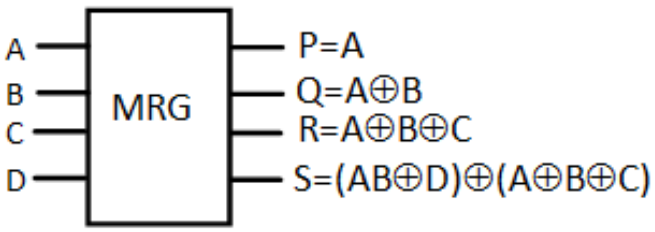
(iii)



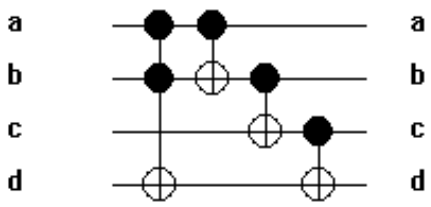
(iv)

Figure 2.9. (i) Block Diagram of MKG [**52**] Gate (ii) MKG - Toffoli Implementation (iii) Toffoli Netlist - Optimized (iv) MKG Gate - QCA realization

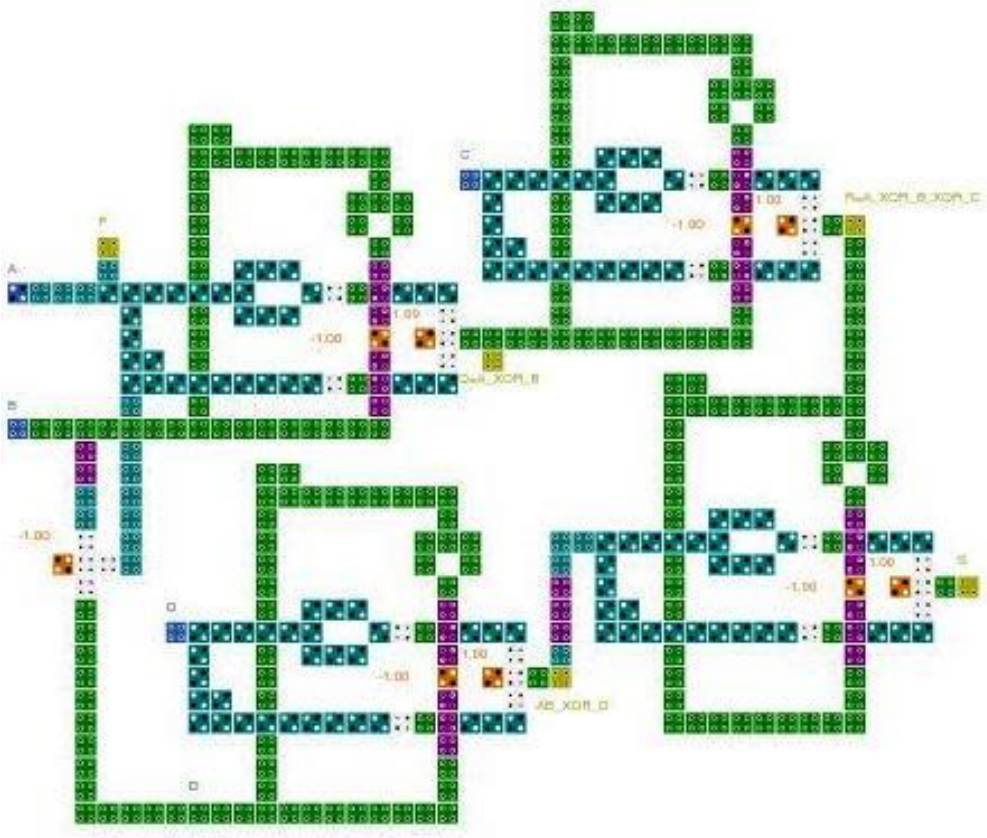
2.1.10 TOFFOLI NETLIST AND QCA IMPLEMENTATION OF MRG GATE



(i)



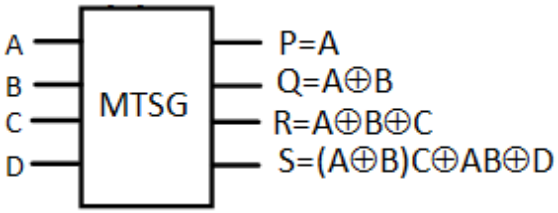
(ii)



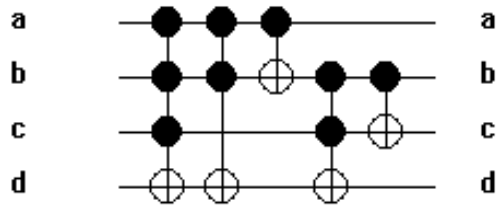
(iii)

Figure 2.10. (i) Block Diagram of MRG [**57**] Gate (ii) MRG - Toffoli Implementation (iii) MRG Gate - QCA realization

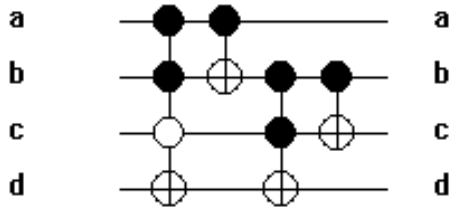
2.1.11 TOFFOLI NETLIST AND QCA IMPLEMENTATION OF MTSG GATE



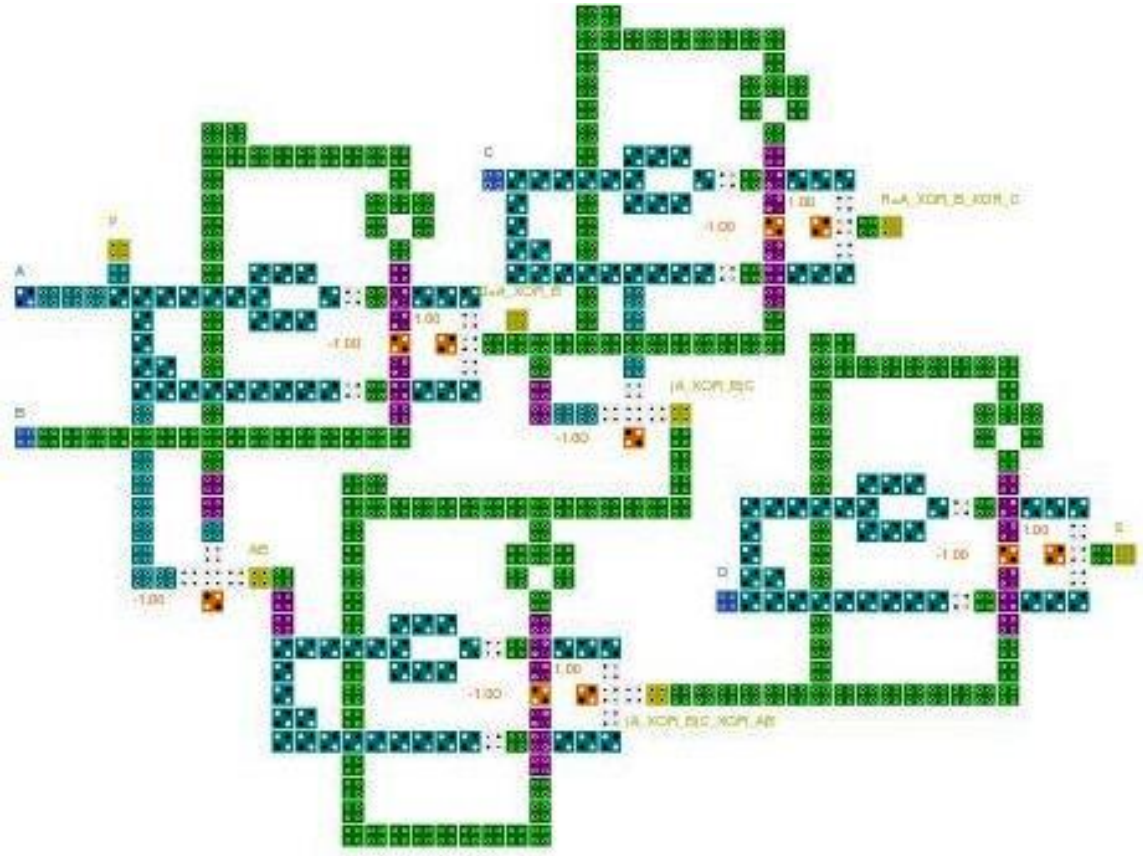
(i)



(ii)



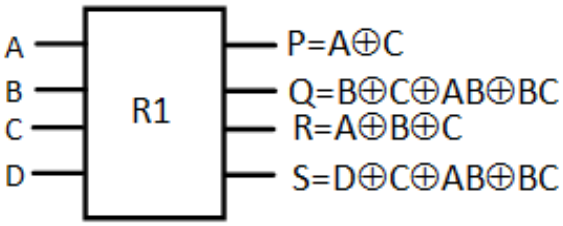
(iii)



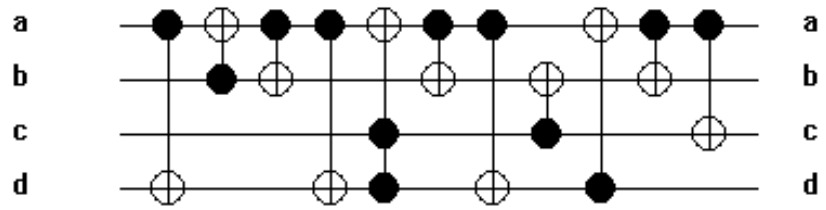
(iv)

Figure 2.11. (i) Block Diagram of MTSG [**41**] Gate (ii) MTSG - Toffoli Implementation (iii) Toffoli Netlist - Optimized (iv) MTSG Gate - QCA realization

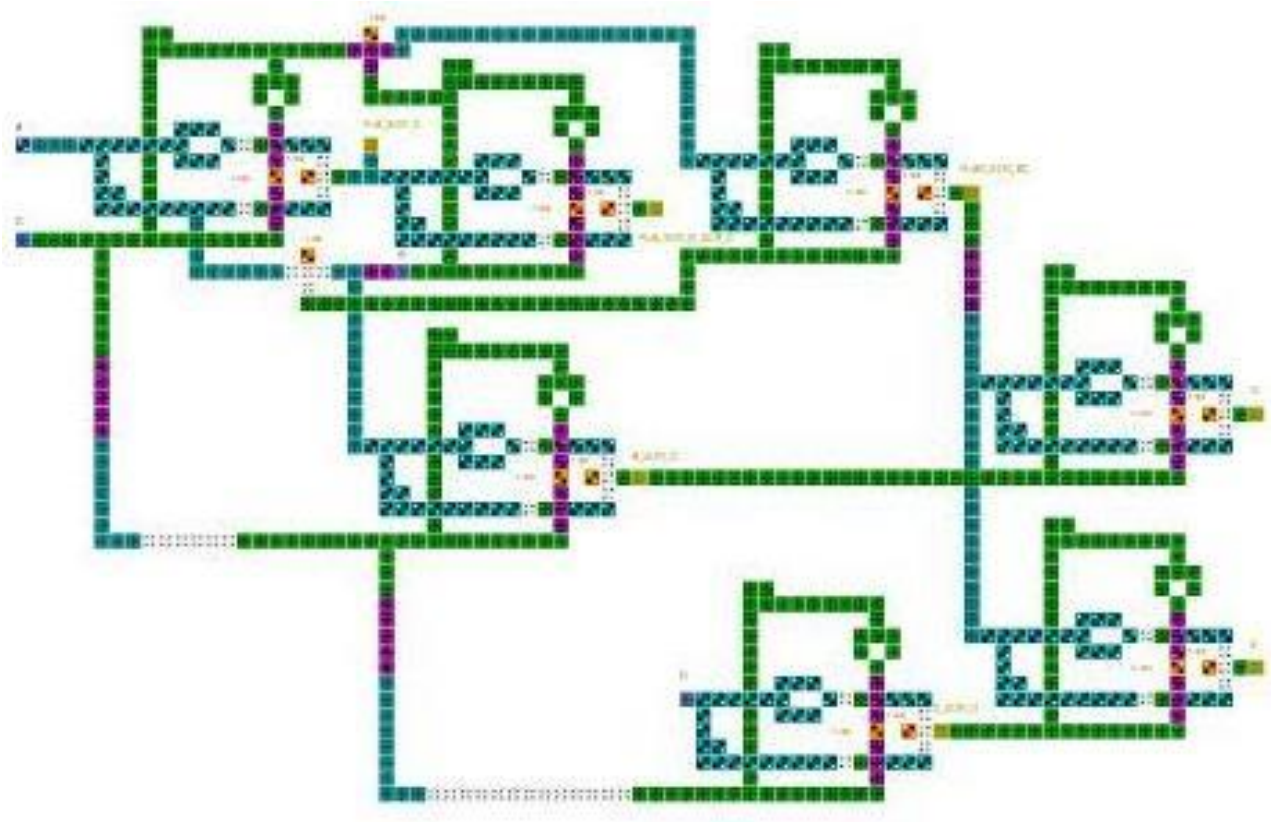
2.1.12 TOFFOLI NETLIST AND QCA IMPLEMENTATION OF R1 GATE



(i)



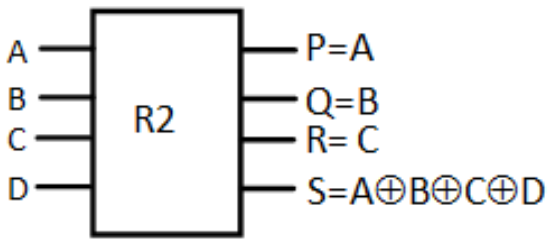
(ii)



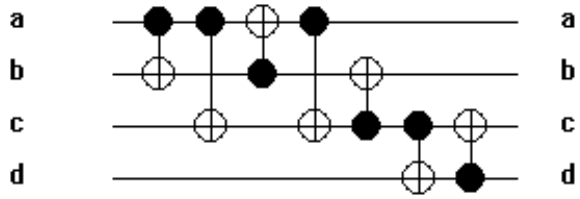
(iii)

Figure 2.12. (i) Block Diagram of R1 [**47**] Gate (ii) R1 - Toffoli Implementation (iii) R1 Gate - QCA realization

2.1.13 TOFFOLI NETLIST AND QCA IMPLEMENTATION OF R2 GATE



(i)

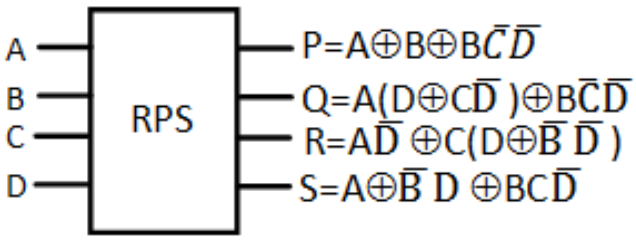


(ii)

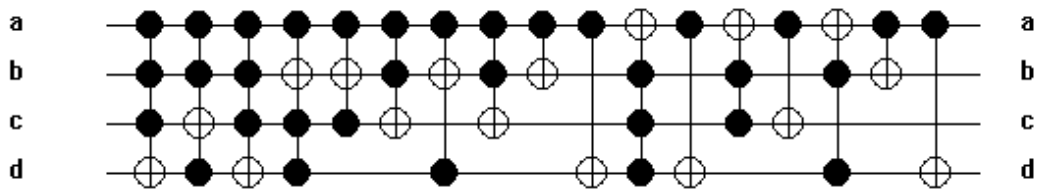
(iii)

Figure 2.13. (i) Block Diagram of R2 [**47**] Gate (ii) R2 - Toffoli Implementation (iii) R2 Gate - QCA realization

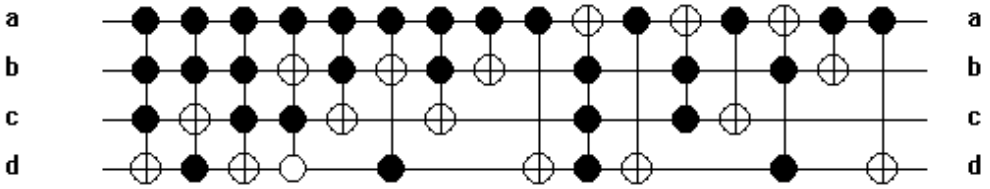
2.1.14 TOFFOLI NETLIST AND QCA IMPLEMENTATION OF RPS GATE



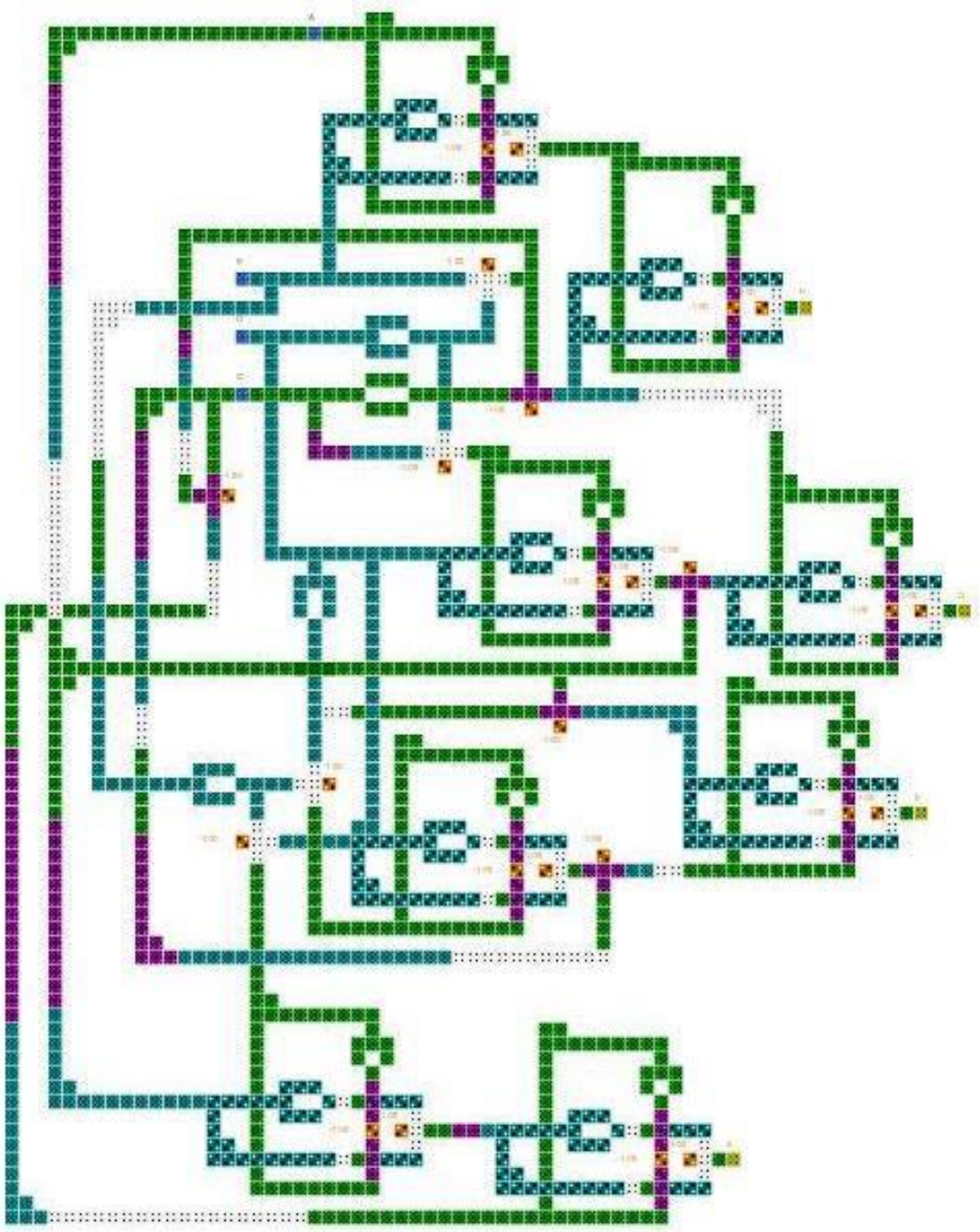
(i)



(ii)



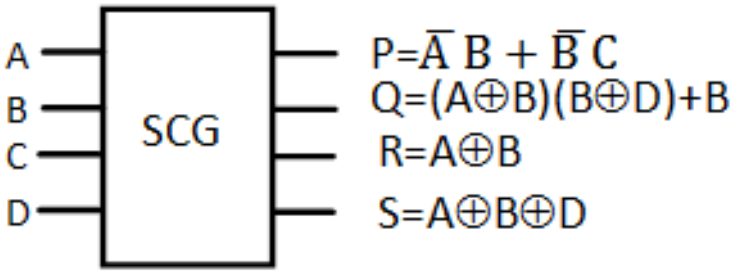
(iii)



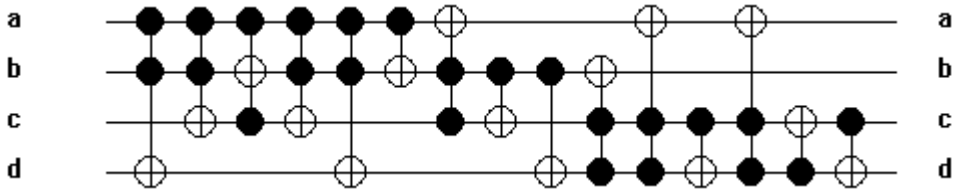
(iv)

Figure 2.14. (i) Block Diagram of RPS [**51**] Gate (ii) RPS - Toffoli Implementation(iii) Toffoli Netlist - Optimized (iv) RPS Gate - QCA realization

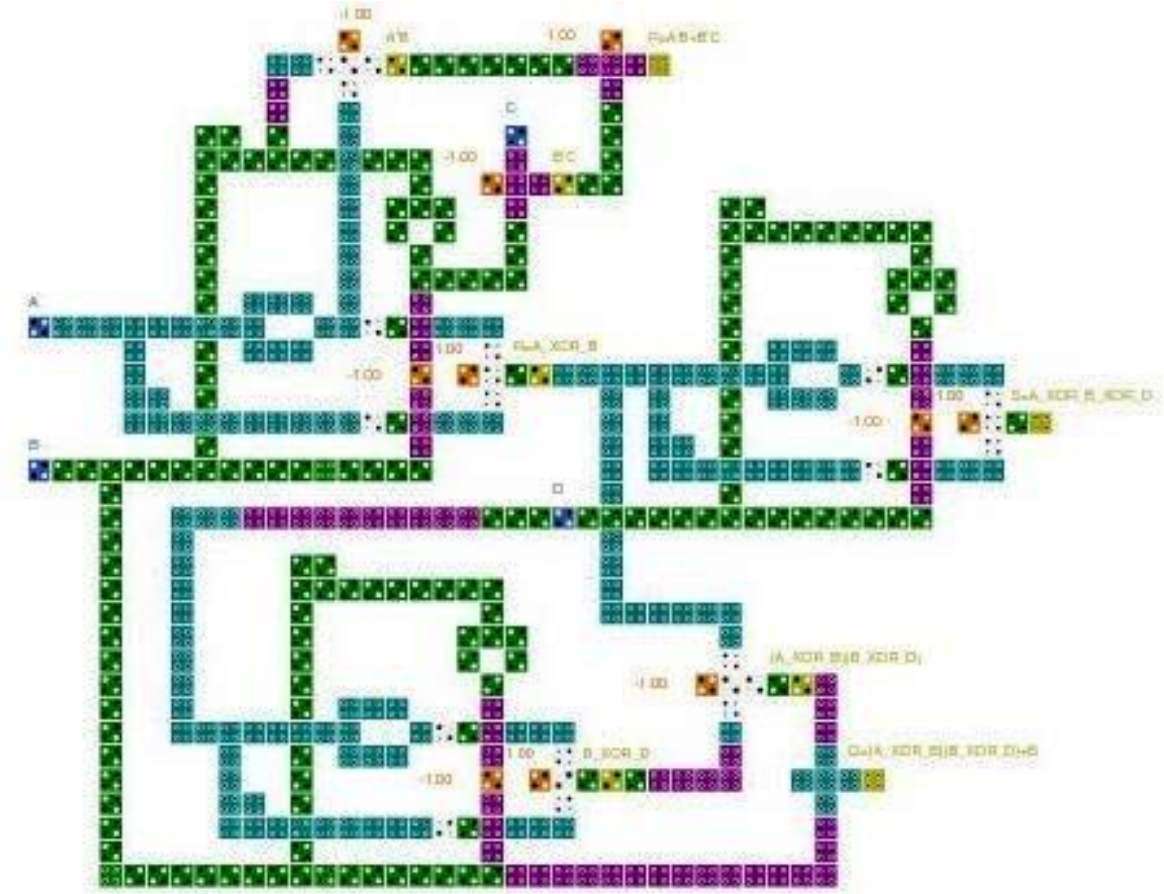
2.1.15 TOFFOLI NETLIST AND QCA IMPLEMENTATION OF SCG GATE



(i)



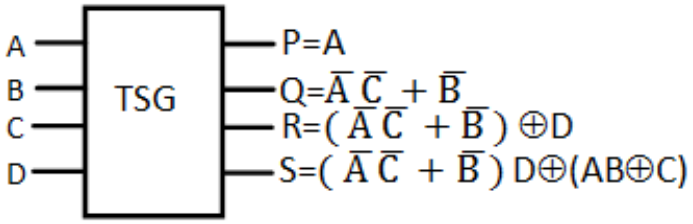
(ii)



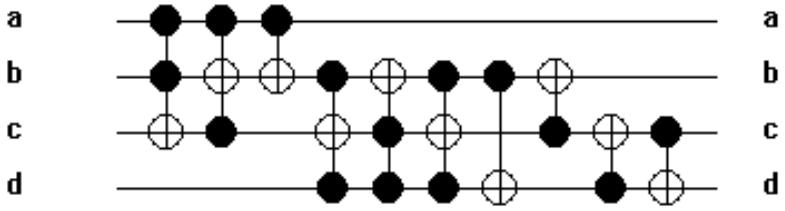
(iii)

Figure 2.15. (i) Block Diagram of SCG [**50**] Gate (ii) SCG – Toffoli Implementation (iii) SCG Gate - QCA realization

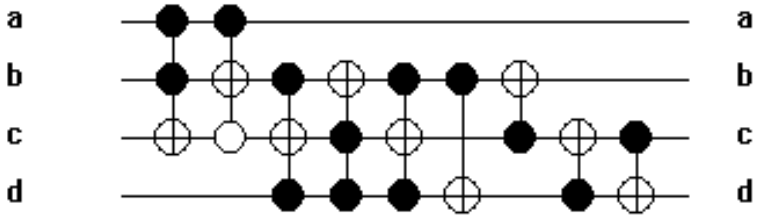
2.1.16 TOFFOLI NETLIST AND QCA IMPLEMENTATION OF TSG GATE



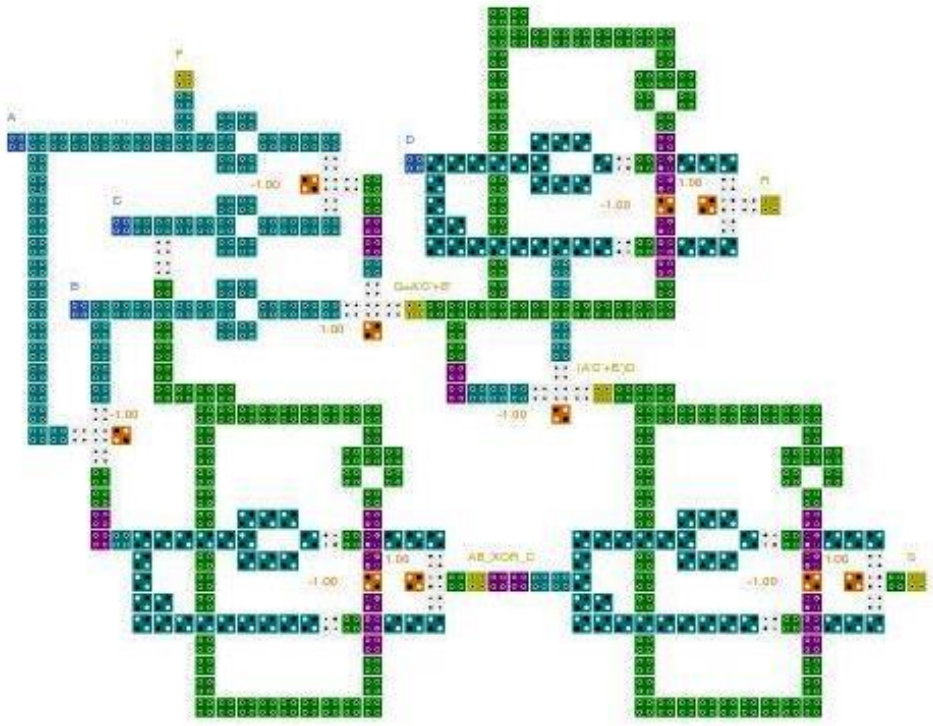
(i)



(ii)



(iii)



(iv)

Figure 2.16. (i) Block Diagram of TSG [**48**] Gate (ii) TSG - Toffoli Implementation (iii) Toffoli Netlist - Optimized (iv) TSG Gate - QCA realization

2.2 QUANTUM METRIC ANALYSIS

This section provides the analysis for the 4-variable reversible gates in terms of Toffoli implementation as well as QCA cell implementations. The analysis has been done using ten quantum metrics of which four metrics are for Toffoli realizations and six metrics are for QCA implementations.

Table 2.1. COMPARATIVE ANALYSIS OF REVERSIBLE GATES FOR TOFFOLI NETLIST DESIGNS

Name of the Gate	Unidirectional Algorithm			After Optimization			*Complexity*
	GC	*QC*	*TG*	*GC*	*QC*	*TG*	
SCG	15	43	51	Already optimized			32
HNG	4	8	12	Already optimized			16
DFG	16	46	48	Already optimized			28
HNFG	5	5	5	Already optimized			24
MTSG	5	21	17	4	18	12	22
FAG	11	19	19	Already optimized			24
BVMF	7	15	15	Already optimized			28
MRG	4	6	8	Already optimized			24
RPS	17	99	61	16	94	56	24
ALG	10	42	34	8	36	28	32
R1	10	18	18	Already optimized			36
R2	7	7	7	Already optimized			32
DKG	15	29	31	14	28	30	28
IG	3	9	11	Already optimized			16
MKG	9	21	21	8	20	20	24
TSG	10	28	30	9	27	29	28

Comparative analysis with respect to ten quantum metrics for existing four variable reversible gates is presented in Table 1. The initial three columns provide the details with respect to the fundamental Quantum Metrics followed by the optimized results in the next three columns. Complexity for each gate has been provided in the seventh column. Table 2 provides the comparative analysis with respect to the QCA implementation. The metrics used in analysis are the QCA Cell count, the design area coverage, majority voter count, inverter count using majority voter, AND gate count, and number of OR gates.

Depending upon the circuit to be realized, it depends upon the merit of the architect to choose a certain gate. The choice can be governed by the Boolean specification/function to be realized.

As discussed earlier, new proposals for four variable reversible gates can also be implemented using Quantum Dot Cellular Automata and the metrics analysed with Table 2 to prove worth of the proposal.

Table 2.2. COMPARATIVE ANALYSIS OF REVERSIBLE GATES FOR QCA DESIGNS

Name of the Gate	No. of Cells	Area	No. of Majority Voters	No. of Inverters	No. of AND Gates	No. of OR Gates
SCG	437	141588	10	6	6	4
HNG	420	136080	10	8	6	4
DFG	269	87156	7	5	4	3
HNFG	206	66744	4	4	2	2
MTSG	420	136080	10	8	6	4
FAG	419	135756	10	8	6	4
BVMF	532	172368	12	10	7	5
MRG	417	135108	9	8	5	4
RPS	1428	462672	24	20	12	8
ALG	890	288360	17	15	10	7
R1	856	277344	16	14	9	7
R2	286	92664	6	6	3	3
DKG	563	182412	12	9	7	5
IG	518	167832	11	8	7	4
MKG	399	129276	10	9	6	4
TSG	396	128304	10	9	6	4

2.3 DISCUSSION

Figure 2.17 presents the graphical representation for Table 1, i.e. the quantum metrics related to Toffoli implementations for the four variable reversible gates. It may be observed that the RPS Gate has the highest Quantum Cost whereas the MRG gate has the lowest Gate Count. All the other gates vary in certain metrics.

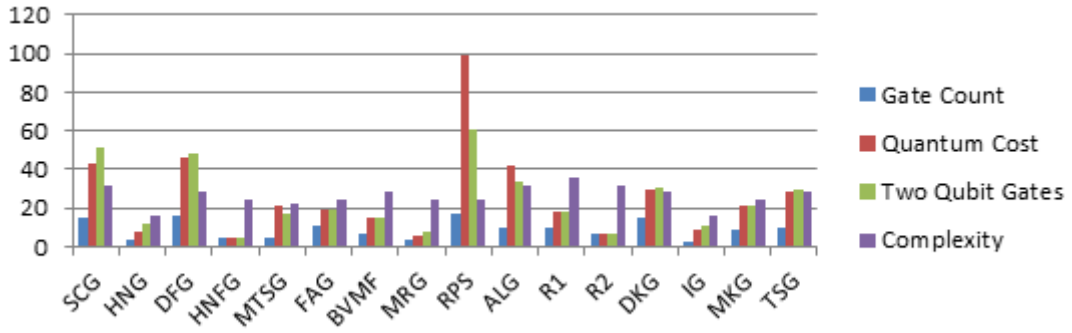


Figure 2.17. Quantum Metric analysis – Reversible Gate Domain

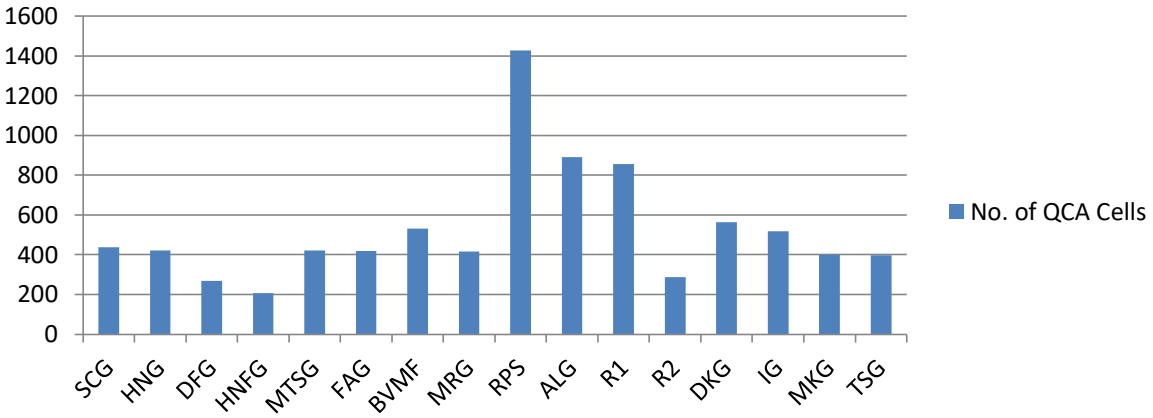


Figure 2.18. Number of QCA cells for each Reversible Gate Design

Figure 2.18 presents the QCA cell count for all the four variable reversible gates. Again as observed earlier, the RPS Gate accounts for the highest number of cells with HNFG consuming the least number of cells. Figure 2.19 presents the Majority Voter count, number of AND Gates, and number of OR Gates consumed to design the respective gates.

From Figure 2.17, Figure 2.18 and Figure 2.19, it can be claimed that the metric analysis for implementation of a design using reversible gates is proportional to the implementation using QCA cells. For example, it can be observed that the RPS Gate has the highest number of all the three gates.

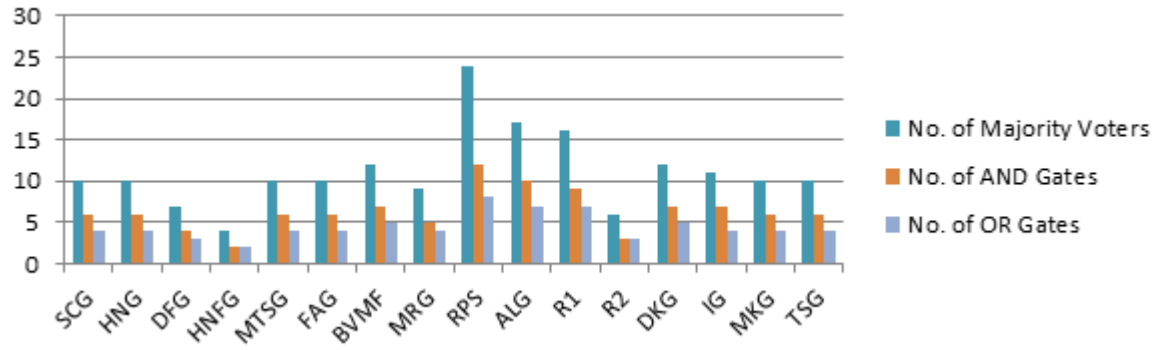


Figure 2.19. Majority Voter Gate count, AND Gate count, OR Gate count

Figure 2.20 presents the requirements for implementation a four variable reversible gate using QCA in terms of Design Area.

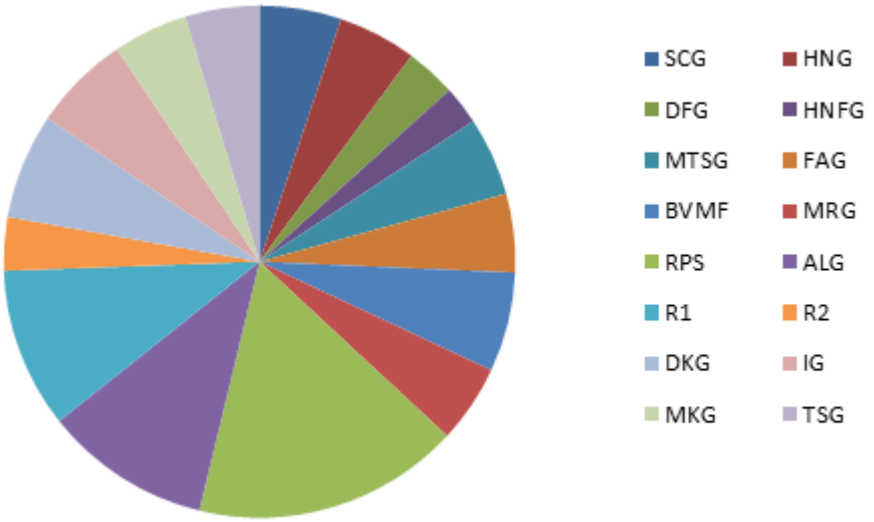


Figure 2.20. Area coverage using QCA platform of the 4-variable reversible gates.

It can be stated that the RPS gate has the largest area consumed followed by the ALG and the R1 gates. The Design Area consideration is very important while choosing certain for implementing a Boolean specification. Also it forms the basis whether the gate is to be realized using reversible Toffoli Netlist or QCA. For example, while choosing a gate for implementing a Full Adder, IG Gate fares better in terms of Toffoli Gate count but when implemented using QCA, the number of Majority Voters, the AND gate and OR gate counts are comparatively higher with respect to SCG Gate. Hence, it lies on the merit of the designer for the proper choice of gate.

2.4 CONCLUSION

Dramatic innovations in technology over the past three decades have raised the bar for expectations from digital hardware. To keep up with the pace of advancements, higher clocking speeds in CMOS with increased chip densities have been achieved. But as of date, CMOS has started experiencing threshold limitations as it provides clocking speeds in the range of GHz and the requirement has already exceeded THz. Also accumulating huge digital gates on limited area of chips has resulted in enormous amount of heat generation. To address these two issues, reversible logic concept gained acceptance among the global researchers. Two platforms have been used widely till date – Toffoli Gates and QCA cells. Also designs for application specific reversible functions have been implemented as reversible gates. This chapter has presented an analysis for all the existing four variable reversible gates in terms of primitive reversible gates and also QCA implementations to form a benchmark for future proposals in the domain. Future research prospect in analysis can be research on clocking speed for QCA implementation of the reversible gates. In terms of Toffoli implementations, the analysis with respect to Hadamard gates can be done to provide a more robust comparison.

CHAPTER 3

Design of Reversible s2c2 Gate

Mahamuda Sultana, Ayan Chaudhuri, Diganta Sengupta and Atal Chaudhuri "Logic Design and Quantum Mapping of a Novel Four Variable Reversible s2c2 Gate", 52nd Annual Convention of Computer Society of India (CSI 2017), January 19-21, 2018, Science City, Kolkata, Springer Nature CCIS Series, vol. 836, pp. 416-427. (SCOPUS)

3 DESIGN OF REVERSIBLE S2C2 GATE

Technological growth in emerging technologies has witnessed global proposals for n-variable reversible gates. More accurately most of the proposals in literature concentrate on value of n to be four. Most of the proposals have been governed by realization of some Boolean specification. The researchers first concentrated on the digital circuit that needs to be converted into the reversible domain and then designed the gates to achieve the reversible designs. An n variable reversible gate can have $!\,2^n$ combinations, i.e. $!\,2^n$ truth tables or gates, since the concept of reversibility suggests the number of input variables should be the number of output variables as discussed in Chapter 1. If the value for n is 4, then it resembles that there are four inputs and four outputs, in the reversible domain. Therefore, a single truth table will comprise of $2^4 = 16$ combinations and there will be $!\,2^4 = !\,16$ such truth tables. Hence, for value of n to be four, $!\,16$ reversible gates can be proposed. Not all such gates will serve the purpose of realizing Boolean functions. Hence, the literary proposals for attaining application oriented reversible gates.

The gate proposal in this chapter has been primarily designed for realizing a standalone full adder, full subtractor, and the fundamental Boolean gates. Standalone means that a single gate with changes in the input values will serve to realize the full adder or the full subtractor. Apart from the discussed realizations, the proposed gate can also be used to design a parity checker or parity generator unit, when provided with certain input combinations at the input. The target for achieving a full adders stems from the fact that full adders form the basis for most of the arithmetic units. Since the propose gate can implement the fundamental Boolean gates, hence it can also be said that any complex Boolean function can be done placing array of the proposed gate, as all the Boolean functions can be realized using the AND, OR, and NOT operations.

There are multiple proposals for realization of full adder and full subtractor using 4 variable reversible gates. Some proposals can realize full adder/full subtractor using a single gate with varying inputs some require two gates to realize full adder and full subtractor. The comparison provided in this chapter also reflects that the proposed gate in this chapter is better than the other literature proposals.

The proposed gate has also been designed using Toffoli gate cascades (Toffoli Netlist) and the details regarding the quantum metrics have been provided. It has been observed that in terms of quantum metrics also, the proposed gate reflects better results than

the literary counterparts.

3.1 PROPOSED S2C2 GATE

As discussed earlier, the initial motivation for designing this gate has been to achieve a full adder and a full subtractor using a single gate. It may be noted that certain input conditions are required to serve the gate as a full adder or a full subtractor. Hence, the gate either behaves as a full adder or a full subtractor at a time but not both simultaneously.

3.1.1 TRUTH TABLE AND PERMUTATION MATRIX

Table 3.1 presents the truth table for the proposed gate. The table also provides the Hamming Distance for each input-output combination and finally provides the Complexity of the gate. For details regarding the Hamming Distance and Complexity, readers are directed to Chapter 1 for the exact definitions.

Table 3.1. TRUTH TABLE OF PROPOSED S2C2 GATE

Inputs				Outputs				δ_i
A	B	C	D	P	Q	R	S	
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	1	0	2
0	0	1	0	0	0	0	1	2
0	0	1	1	0	1	1	1	1
0	1	0	0	0	0	1	1	3
0	1	0	1	0	1	0	1	0
0	1	1	0	0	1	1	0	0
0	1	1	1	0	1	0	0	2
1	0	0	0	1	0	1	1	2
1	0	0	1	1	1	0	1	1
1	0	1	0	1	0	1	0	0
1	0	1	1	1	0	0	0	2
1	1	0	0	1	1	0	0	0
1	1	0	1	1	1	1	0	2
1	1	1	0	1	0	0	1	3
1	1	1	1	1	1	1	1	0
Complexity								20

Table 3.2. PERMUTATION MATRIX (S2C2 GATE)

1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Table 3.2 provide the permutation matrix for the truth table given in Table 3.1. The permutaion matrix works in a fashion that the rows are the input values and the columns are the output values. A '1' in the permutation matrix denotes the cell where the input output converges.

Let R_nC_n denote the n^{th} row and the n^{th} column.

Therefore a '1' at R_0C_0 resembles that for input of 0000 at the 0^{th} row, the output is 0000 at the 0^{th} column.

Similarly, a '1' at R_1C_2 resembles that for input of 0001, the output is 0010, as can be seen from Table 3.1.

Similarly, a '1' at R_4C_3 resembles that for input of 0100, the output is 0011, as can be seen from Table 3.1.

Similarly, a '1' at R_7C_4 resembles that for input of 0111, the output is 0101, as can be seen from Table 3.1.

Similarly, a '1' at R_9C_{13} resembles that for input of 1001, the output is 1101, as can be seen from Table 3.1, and so on.

3.1.2 BLOCK DIAGRAM AND EQUATIONS

Figure 3.1 presents the generalized block diagram for s2c2 gate.

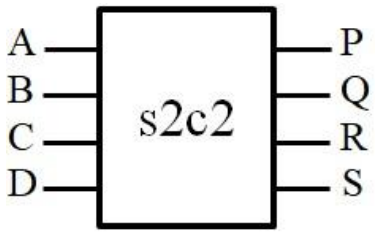


Figure 3.1. Block Diagram for proposed s2c2 gate

The output expressions can be fetched from the truth table provided in Table 3.1 using the K-Map approach.

The output expressions are as follows:

$P = A$

$Q = (A \oplus C)(B \oplus D) \oplus BD \equiv (A \oplus C)(B + D) + BD$

$R = A \oplus B \oplus D$

$S = A \oplus B \oplus C$

It can be observed that the final two output variables, R and S can be used to design a parity generator or a parity checker circuit as they provide XOR operation, the main operation in parity generator and parity checker circuits.

3.1.3 TOFFOLI REALIZATIONS

As discussed earlier, the Toffoli designs have implemented using the Unidirectional Algorithm and the Bidirectional Algorithm proposed in [**30**] [**24**] and also discussed in Chapter 1 of this thesis. The Toffoli designs have been further optimized using the Post-Synthesis Optimization Algorithm mentioned in [**72**] and also discussed in Chapter 1 of this thesis. The programs for Toffoli design have been coded in the language for RCViewer+ [**73**], a

tool used for designing reversible circuits.

The codes for RCViewer+ [**73**] for realizing the Toffoli Netlist using Unidirectional Algorithm of presented below.

$TOF3\ (A, B, D)$

$TOF3\ (A, B, C)$

$TOF3\ (A, C, B)$

$TOF3\ (A, B, D)$

$TOF3\ (A, B, D)$

$TOF2\ (A, B)$

$TOF2\ (B, D)$

$TOF2\ (B, C)$

$TOF3\ (C, D, B)$

$TOF2\ (C, D)$

$TOF2\ (D, C)$

$TOF2\ (C, D)$

Figure 3.2 presents the Toffoli design for the s2c2 gate (the design has been made using the Unidirectional Algorithm of [**30**] [**24**]).

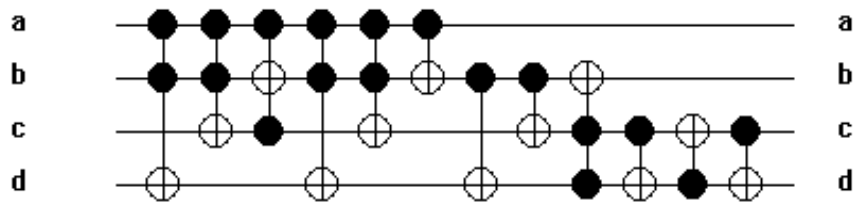


Figure 3.2. Reversible s2c2 gate Toffoli Implementation using Unidirectional Algorithm

Figure 3.2 reflects that twelve Toffoli gates have been used to design the proposed gate. Of these seven gates are three input Toffoli gates and five gates are two input Toffoli gates.

The codes for RCViewer+ [**73**] for realizing the Toffoli design implementing Bi-Directional Algorithm is presented below.

$TOF3\ (A, C, B)$

$TOF2\ (A, B)$

$TOF2\ (A, C)$

$TOF2\ (A, D)$

$TOF3\ (B, C, D)$

$TOF2\ (B, D)$

$TOF3\ (C, D, B)$

$TOF3\ (B, C, D)$

$TOF2\ (B, C)$

$TOF2\ (C, D)$

$TOF2\ (D, C)$

$TOF2\ (C, D)$

Figure 3.3 presents the Toffoli design for the s2c2 gate. (The design has used the Bi-Directional Algorithm of [**30**] [**24**]).

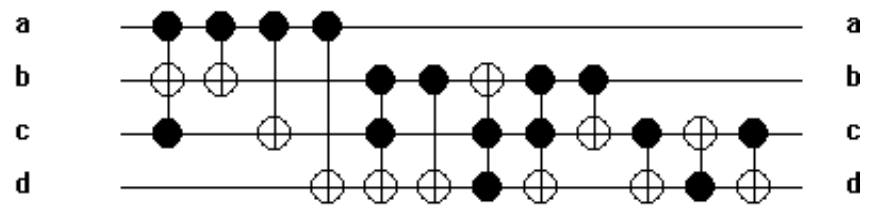


Figure 3.3. Reversible s2c2 gate Toffoli Implementation using Bi-Directional Algorithm

Figure 3.3 reflects that twelve Toffoli gates have been used to design the proposed gate. Of these four gates are three input Toffoli gates and seven gates are two input Toffoli gates.

Both the design using Toffoli Gates have been fed to the optimization algorithm. The design based on Unidirectional Algorithm could not be optimized further. Only the design based on Bi-Directional Algorithm could be optimized. Figure 3.4 provides the optimized design for the proposed gate.

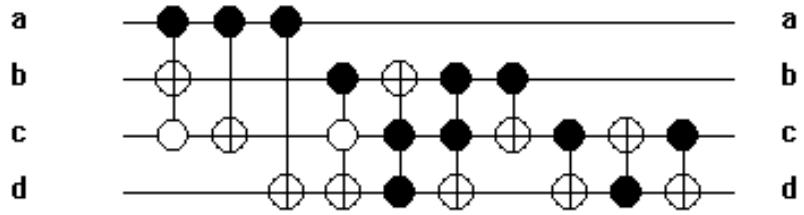


Figure 3.4. Optimized Toffoli design for proposed s2c2 gate

It can be observed the optimization has resulted in a reduced gate count of ten reversible Toffoli Gates containing six two input Toffoli gates, two three input Toffoli Gates having positive control lines, and two three input Toffoli gates having negative control lines.

3.1.4 QUANTUM METRICS FOR THE PROPOSED S2C2 GATE

It may be noted that the Quantum Cost (please refer Chapter 1 for the definition of Quantum Cost) for one input Toffoli gate is '1'. The Quantum Cost for two input Toffoli gate is '1'. The Quantum Cost for three input Toffoli Gate is 5, and the Quantum Cost for four input Toffoli Gate is '13'.

Using these metrics, Table 3.3 presents the quantum metric for the proposed designs. For definition of Quantum Metric, please refer Chapter 1.

Table 3.3. QUANTUM METRICS FOR S2C2 GATE

Toffoli Netlist using	Gate Count	Quantum Cost	# Two Qubit Gates
Unidirectional Algorithm	12	32	36
Bi-Directional Algorithm	12	26	28
Optimized Bi-Directional	10	24	26

3.1.5 BOOLEAN FUNCTION REALIZATIONS

As mentioned in the beginning of the chapter that the proposed gate realized full adder, full subtractor, and Boolean logic gates, Table 3.4 presents the complete set of Boolean realizations possible using the proposed gate.

Table 3.4. BOOLEAN FUNCTION REALIZATION USING S2C2 GATE

Inputs				Outputs				Logic Realizations
A	B	C	D	P	Q	R	S	
A	1	1	1	A	1	A	A	Signal Duplication (FO3)
1	B	1	1	1	B	B	B	Signal Duplication (FO3)
1	1	C	1	1	1	1	C	Through Signal
1	1	1	D	1	D	D	1	Signal Duplication (FO2)
0	B	1	1	0	1	B'	B'	Negation (NOT)
0	B	1	0	0	B	B	B'	Duplication (FO2) + Negation (NOT)
1	B	0	0	1	B	B'	B'	NOT Duplication
A	B	0	0	A	A'B	A⊕B	A⊕B	XOR Duplication
1	B	0	D	1	B+D	(B⊕D)'	B'	OR, XNOR
0	B	0	D	0	BD	(B⊕D)	B	AND, XOR
1	B	1	D	1	BD	(B⊕D)'	B	AND, XNOR
A	B	1	1	A	A'+B	(A⊕B)'	(A⊕B)'	XNOR
Cin	B	0	A	Cin	AB⊕Cin(A⊕B)	A⊕B⊕Cin	B⊕Cin	Full Adder + Control Through + XOR
Cin	B	1	A	Cin	AB⊕Cin'(A⊕B)	A⊕B⊕Cin	(B⊕Cin)'	Full Subtractor + Control Through + XNOR

From Table 3.4, it can be observed that the input combinations for realizing full adder and full subtractor is provided in the last two rows. Further, since reversible circuits cannot have branching at the output, hence certain input combinations provide copy operation also. For example, the first and the second rows provide copy operation for the input signal 'A'. Three values of signal 'A' have been generated.

Expressions for XOR and XNOR operations can also be generated, thus aiding in the realization of parity generator and parity checker circuit. Not only parity generator or parity checker, any XOR or XNOR intensive operation can be realized using proper input combinations of the proposed s2c2gate. Since, XOR forms an intrinsic part of cryptography; hence it may be assumed that the proposed gate will serve the purpose for cryptographic applications also.

Further observation reveals that the basic Boolean operations, viz. the AND, NOT, and OR operations can also be derived using the proposed gate. Since, these three operations are possible, hence it can be claimed that any Boolean specification, simple or complex can be designed using the proposed gate.

3.2 COMPARATIVE ANALYSIS

Table 3.5 presents the comparative data of s2c2 gate with respect to other peer reversible gates based on four inputs.

It has been seen earlier that the proposed gate exhibits better results with respect to other gates in most of the cases. In some cases, other proposals may fare better in some parts but it will be seen later that in other realizations, the proposed gate is better. Hence, taking cognizance of both the quantum metrics and the other parameters, it can be assumed that the proposed s2c2 gate is better than the other literary counterparts.

Figure 3.5 presents the graphical analysis for the comparison provided in Table 3.5.

Table 3.5. COMPARATIVE ANALYSIS OF THE PROPOSED SC2C GATE

Name of the Gate	GC	QC	TQG	C_f
FAG	11	19	19	24
SCG	15	43	51	32
DFG	16	46	48	28
MRG	4	6	8	24
TSG	10	28	30	28
HNG	4	8	12	16
R1	10	18	18	36
R2	7	7	7	32
HNFG	5	5	5	24
MKG	9	21	21	24
IG	3	9	11	16
RPS	17	99	61	24
ALG	10	42	34	32
MTSG	5	21	17	22
DKG	15	29	31	28
BVMF	7	15	15	28
TCG	10	34	26	33
Proposed s2c2 Gate using				
Basic/Unidirectional Algorithm	12	32	36	20
Bi-Directional Algorithm	12	26	28	
Optimized Bi-Directional Algorithm	10	24	26	

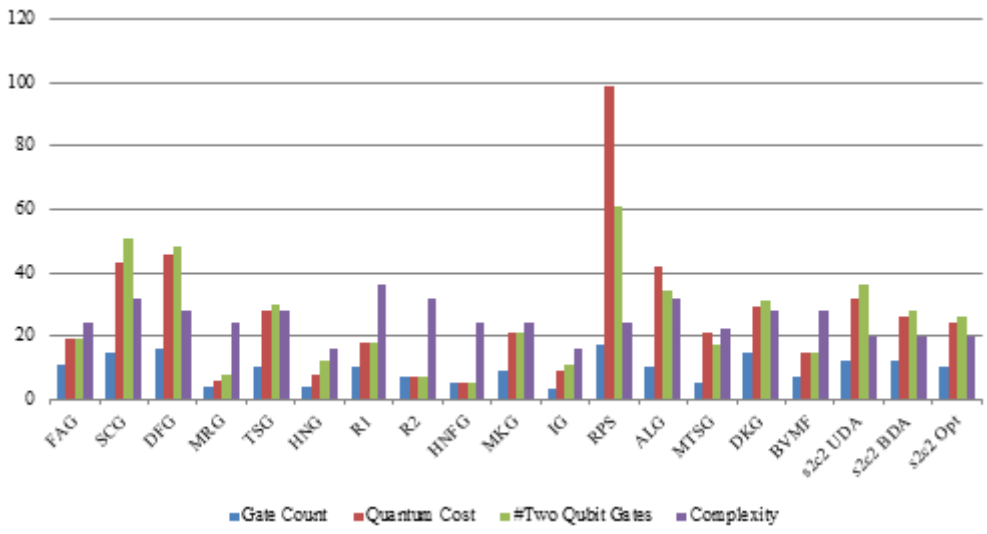


Figure 3.5. Graphical analysis for comparative analysis in Table 3.5

The last three characteristics in Figure 3.5 present the data for the Unidirectional Algorithm, Bi-Directional Algorithm, and the Optimized design. In all the designs, it may be deduced that the proposed gate fares better.

Table 3.6 presents the comparison with other reversible gate proposals in terms of basic Boolean logic gate operations.

Table 3.6. COMPARATIVE ANALYSIS WITH RESPECT TO BASIC BOOLEAN FUNCTIONS

Name of the Gate	AND	OR	NOT	XOR
FAG	1	0	0	4
SCG	3	2	2	3
DFG	4	2	1	2
MRG	1	0	0	4
TSG	3	1	3	3
HNG	1	0	0	4
R1	2	0	0	6
R2	0	0	0	3
HNFG	0	0	0	2
TCG	0	2	0	3
MKG	2	0	3	4
IG	2	0	1	4
RPS	9	0	3	8
ALG	4	0	1	7
MTSG	2	0	0	4
DKG	3	1	2	5
BVMF	3	1	2	4
Proposed s2c2	2	2	0	4

Figure 3.6 presents the graphical interpretation of Table 3.6. Figure 3.6 also reflects that the proposed gate is better in terms of basic boolean logic function realization.

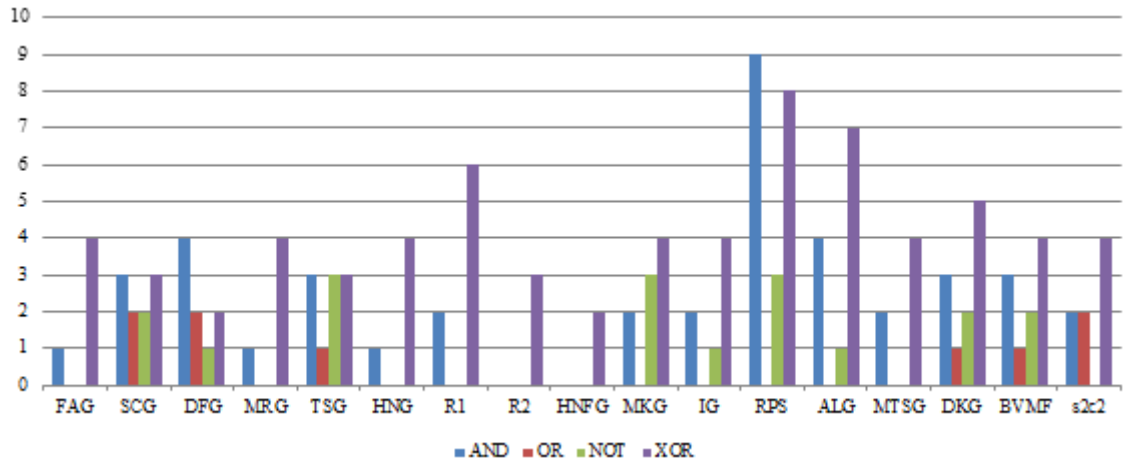


Figure 3.6. Graphical Representation for comparison based on basic Boolean logic operations

Apart from the quantum metric analysis and basic Boolean logic function analysis, another analysis has been done based on the primitive reason for proposal of s2c2 gate. The reason is realization of full adder and full subtractor using reversible gates having four inputs. Table 3.7 provides such a comparison.

Table 3.7. COMPARATIVE ANALYSIS OF REVERSIBLE GATES IN TERMS OF FULL ADDER/SUBTRACTOR REALIZATIONS

Name of Gate	# Gates	# Garbage outputs	Full Subtractor
TSG	1	2	False
IG	2	3	False
HNG	1	2	False
SCG	1	2	True
Proposed s2c2	1	2	True

It can be observed from Table 3.7, that although all the gates provided in Table 3.7 can realize full adders using a single gate, but they fail to realize a full subtractor using a single gate. The only gate that can realize a full adder and a full subtractor using a single gate other than s2c2 is SCG [**50**] Gate which has also been proposed by the authors of this the-

sis. Close observational of input ‘C’ in Table 3.4 reveals that the design can be changed from realizing a full adder to realizing a full subtractor.

3.3 CONCLUSION

This chapter has proposed a reversible gate having four inputs and four outputs. The gate has been realized using fundamental reversible gates, i.e. Toffoli Gates. It has been observed that the proposed gate not only realized a full adder or full subtractor, but also a lot of Boolean functions as well as parity generator and checkers circuits.

The work in this chapter can be further extended to realize 8/16/32/64 bit binary adder and subtractor. Since, multiplication and division is done using successive addition and subtraction respectively, hence cascaded designs of the proposed gate can be used for designing reversible multiplier and division circuits.

The proposed gate can also be used to design Binary to Gray code converters and vice versa since, XOR and XNOR operations can be used using the gate.

CHAPTER 4

Reversible Logic Synthesis Algorithm

Sinjini Banerjee, Priyabrata Sahoo, Mahamuda Sultana, Ayan Chaudhuri, Diganta Sengupta and Atal Chaudhuri "Toffoli Netlist Based Synthesis of Four Variable Reversible Functions", in Proc. 2017 Third IEEE International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN 2017), 2017, Kolkata, pp 315-320 (SCOPUS).

4 REVERSIBLE LOGIC SYNTHESIS ALGORITHM

Research on Reversible Logic has witnessed multi-dimensional aspects, one of which is the proposals for synthesis and optimization of reversible circuits. Multiple proposals exist which have used library / template based algorithms to synthesize reversible circuits. Other dimensions relate to Binary Decision Diagram (BDD) based synthesis algorithms. Further exploration reveals permutation and recurrence based algorithms to generate reversible circuits. A considerable amount of proposals have been made in optimization algorithms also. This chapter proposes a synthesis algorithm for synthesis of four input reversible functions. The algorithm takes into account a library which has been pre-defined. The library contains a already defined set of Control Lines. It basically contains Toffoli Netlist sets which can be invoked for certain transformations. The algorithm is optimized in nature and a certain Toffoli Netlist is selected using the Hamming Distance criteria (please refer Chapter 1 for complete definition of Hamming Distance). Since, the algorithm is itself optimized, hence need for post-synthesis optimization gets eliminated.

4.1 PROPOSED SYNTHESIS ALGORITHM

An n variable reversible gate can have $!\,2^n$ combinations, i.e. $!\,2^n$ truth tables or gates, since the concept of reversibility suggests the number of input variables should be equal to the number of output variables as discussed in Chapter 1. If the value for n is 4, then it resembles that there are four inputs and four outputs, in the reversible domain. Therefore, a single truth table will comprise of $2^4 = 16$ combinations and there will be $!\,2^4 = !\,16$ such truth tables. These sixteen bit combinations can interchange their position within a truth table for four inputs in $!\,16$ ways. We have discussed earlier that the synthesis of reversible gates containing four inputs have been done using Unidirectional Algorithm. The preliminary step in the proposed algorithm is identical to the Unidirectional Algorithm in [**29**] [**30**].

The target of any synthesis algorithm is to convert the output side of the truth table for reversible function in continuous values starting from 0000 till 1111, in case of four input variables. Mathematically,

$f(0)\ to\ f(15),\ f(0) = \{0000\},\ f(15) = \{1111\}, where\ f(i+1) = f(i) + 1; 0 \leq i \leq 15$

The final results provide the following:

$(i)_{decimal} \rightarrow f(i); 0 \leq i \leq 15; where\ f(i)\ is\ the\ output\ for\ (i)_{decimal}\ input.$

4.2 GENERATION OF CONTROL LINE SET LIBRARY

Step by step explanation of the algorithm for generation of Control Line Set Library.

Let

$f(i) = a_i^3 a_i^2 a_i^1 a_i^0\ and\ (i)_{decimal} = \sum_{j=0}^{3} 2^j a_i^j.$

- Initially the value for $f(0)$ is check if it is zero bit stream or not. i.e. To check if $f(0) = \{0000\}$
- If all the bits of $f(0)$ are not '0', i.e. $f(0)! = \{0000\}$, then the complete column containing the '1' at $f(0)$ is inverted and an inverter is placed for this operation. This operation is continued for all the '1's in $f(0)$. The inverter can be realized by a single input Toffoli Gate as has been discussed in chapter 1.
    - For example, let $f(0) = \{0101\}$, then inversion of the least significant two bits are required to make $f(0) = \{0000\}$. Hence, the values at the columns corresponding to the '1's are inverted and an inverter is accounted for the number of columns having '1's in $f(0)$.
    - Taking $f(i) = a_i^3 a_i^2 a_i^1 a_i^0$, if
    - $f(0) = \{0110\} \rightarrow invert\ all\ the\ values\ at\ (a_i^2, a_i^1)$
    - $f(0) = \{1001\} \rightarrow invert\ all\ the\ values\ at\ (a_i^3, a_i^0)$
    - $f(0) = \{0111\} \rightarrow invert\ all\ the\ values\ at\ (a_i^2, a_i^1, a_i^0)$
    - $f(0) = \{1100\} \rightarrow invert\ all\ the\ values\ at\ (a_i^3, a_i^2)$, and so on.
- After $f(0)$ has been converted to $\{0000\}$, next $f(1)$ is taken in to account. $f(1)$ is checked whether the value for it is $\{0001\}$. If the value of $f(1)! = \{0001\}$, there can be fourteen possibilities that can exist for the output of $f(1)$.
    - Since, $\{0000\}$ has already been used for $f(0)$, therefore $\{0000\}$ cannot be considered again because the concept of reversibility states that the input and output should be mapped one-to-one.
    - Since $\{0001\}$ is not present at $f(1)$, therefore all the 4-bit combinations from $\{0010\}$ till $\{1111\}$ can be at the output for $f(1)$. These account for

the fourteen combinations.

- Based on the value that is present in the output of $f(1)$, the number of Toffoli gates are required. Also the value should be converted to {0001}.

- The use of Toffoli gate/s for changing the output value for $f(1)$ into {0001} is governed by Table 4.1.

TABLE 4.1. CONTROL LINE SET LIBRARY

a^3	a^2	a^1	a^0	Control Line Combinations
0	0	0	0	
0	0	0	1	$[a^0, a^1, a^2, a^3, a^0a^1, a^0a^2, a^0a^3, a^1a^2, a^1a^3, a^2a^3, a^0a^1a^2, a^0a^1a^3, a^0a^2a^3, a^1a^2a^3]$
0	0	1	0	$[a^1, a^2, a^3, a^0a^1, a^0a^2, a^0a^3, a^1a^2, a^1a^3, a^2a^3, a^0a^1a^2, a^0a^1a^3, a^0a^2a^3, a^1a^2a^3]$
0	0	1	1	$[a^2, a^3, a^0a^1, a^0a^2, a^0a^3, a^1a^2, a^1a^3, a^2a^3, a^0a^1a^2, a^0a^1a^3, a^0a^2a^3, a^1a^2a^3]$
0	1	0	0	$[a^2, a^3, a^0a^2, a^0a^3, a^1a^2, a^1a^3, a^2a^3, a^0a^1a^2, a^0a^1a^3, a^0a^2a^3, a^1a^2a^3]$
0	1	0	1	$[a^3, a^0a^2, a^0a^3, a^1a^2, a^1a^3, a^2a^3, a^0a^1a^2, a^0a^1a^3, a^0a^2a^3, a^1a^2a^3]$
0	1	1	0	$[a^3, a^0a^3, a^1a^2, a^1a^3, a^2a^3, a^0a^1a^2, a^0a^1a^3, a^0a^2a^3, a^1a^2a^3]$
0	1	1	1	$[a^3, a^0a^3, a^1a^3, a^2a^3, a^0a^1a^2, a^0a^1a^3, a^0a^2a^3, a^1a^2a^3]$
1	0	0	0	$[a^3, a^0a^3, a^1a^3, a^2a^3, a^0a^1a^3, a^0a^2a^3, a^1a^2a^3]$
1	0	0	1	$[a^0a^3, a^1a^3, a^2a^3, a^0a^1a^3, a^0a^2a^3, a^1a^2a^3]$
1	0	1	0	$[a^1a^3, a^2a^3, a^0a^1a^3, a^0a^2a^3, a^1a^2a^3]$
1	0	1	1	$[a^2a^3, a^0a^1a^3, a^0a^2a^3, a^1a^2a^3]$
1	1	0	0	$[a^2a^3, a^0a^2a^3, a^1a^2a^3]$
1	1	0	1	$[a^0a^2a^3, a^1a^2a^3]$
1	1	1	0	$[a^1a^2a^3]$
1	1	1	1	

- It is important to note that the use of control lines in the manner of xy and yx is equivalent
- This process is continued from $f(2)\ to\ f(14)$. $f(15)$ is automatically generated since only one value will remain unutilized by the time the algorithm reaches $f(15)$.

- It must be noted that for $f(2)$, the number of control line combinations are thirteen in Table 4.1. For $f(3)$, the number of combinations are twelve in Table 4.1. therefore with each iteration, the control line combinations are reduced.
- The above point can be utilized to generate the number of Control Line Sets required for Toffoli gate implementation. Hence the Control Line Set count is

$$14 + 13 + 12 + \cdots 1 = 105; \frac{m(m+1)}{2}; where\ m = 2^n - 2;$$

Where $n = Number\ of\ Bits, in\ this\ case\ 4.$

- Hence, the generalized equation for generating the Control Line Set for Toffoli Gate synthesis can be given by

$$CLS = \frac{(2^n - 2)(2^n - 1)}{2}$$

All the 105 Control Line combinations are present in Table 4.1

It is to be noted that $\{0000\}$ is the initial step of generation and $\{1111\}$ is generated automatically after $f(0)\ to\ f(14)$ have been synthesized, hence the number of ways in which $f(1)$ can be generated is $2^4 - 2 = 14$. These fourteen combinations are placed alongside the row for $\{0001\}$ in Table 4.1.

Similarly

$\{0010\}$ has thirteen possibilities

$\{0011\}$ has twelve possibilities

$\{0100\}$ has eleven possibilities

$\{0101\}$ has ten possibilities

$\{0110\}$ has nine possibilities

$\{0111\}$ has eight possibilities

{1000} has seven possibilities

{1001} has six possibilities

{1010} has five possibilities

{1011} has four possibilities

{1100} has three possibilities

{1101} has two possibilities

{1110} has one possibilities

Hence, generation of control combination set for $f(i)$ is done from the set for $f(i-1)$ post elimination of the combination which possesses logic '1' at the bit positions. For, example, in Table 4.1, the set for {0101} contains all the combinations that is applicable for {0100} except a^2 as a^2 is at logic '1' in {0100}.

4.3 REVERSIBLE DESIGN SYNTHESIS USING TOFFOLI NETLIST

This section provides the complete library of Control Line Sets and their respective Toffoli Gates for template mapping of the gate while synthesizing the reversible circuit. It may be noted again that $f(0)$ need no template matching as it is generated using inverters only as discussed earlier. Similarly, $f(15)$ is automatically generated.

For the rest of the fourteen combinations, the forth coming tables provide the library.

4.3.1 GENERATION OF $f(1)$

If $f(1)! = \{0001\}$, Then library provided in Table 4.2 is consulted.

It is to be noted that in Table 4.2, $abcd \equiv a^3a^2a^1a^0$. '*abcd'* notation has been used for its compatibility with *RCViewer+* [**74**] as suffixes are not supported by the tool.

It is worth noting that 't2' stands for two-input Toffoli Gate. Similarly 't1' and 't3' stand for one-input and three-input Toffoli gates respectively.

TABLE 4.2. LIBRARY FOR GENERATING {0001} – TOFFOLI NETLIST

To generate {0001} from				
a	**b**	**c**	**d**	**Toffoli Netlist**
0	0	1	0	t2 c d → t2 d c
0	0	1	1	t2 d c
0	1	0	0	t2 b d → t2 d b
0	1	0	1	t2 d b
0	1	1	0	t2 c d → t2 d c → t2 d b
0	1	1	1	t2 d c → t2 d b
1	0	0	0	t2 a d → t2 d a
1	0	0	1	t2 d a
1	0	1	0	t2 c d → t2 d c → t2 d a
1	0	1	1	t2 d c → t2 d a
1	1	0	0	t2 b d → t2 d b → t2 d a
1	1	0	1	t2 d b → t2 d a
1	1	1	0	t2 c d → t2 d c → t2 d b → t2 d a
1	1	1	1	t2 d c → t2 d b → t2 d a

Generation of $f(2)\ to\ f(14); i.e. \{0010\}\ to\ \{1110\}$ has been done using the libraries shown in Table 4.2 through Table 4.15.

A notable point in the conversion is that the number of Toffoli gates required for synthesis is equal to the Hamming distance between the $f(x)$ and the present relation of $f(x)$ on the right hand side of the truth table for the reversible gate. Hence, if we are synthesizing$f(2) = \{0010\}$, and $f(2)$ presently has $\{0101\}$ as its output relation, then the Hamming Distance between $\{0010\}$ and $\{0101\}$ is 3, therefore three Toffoli Gates will be used for its substitution.

4.3.2 GENERATION OF $f(2)$

If $f(2)! = \{0010\}$, Then library provided in Table 4.3 is consulted.

4.3.3 GENERATION OF $f(3)$

If $f(3)! = \{0011\}$, Then library provided in Table 4.4 is consulted.

TABLE 4.3. LIBRARY FOR GENERATING {0010} – TOFFOLI NETLIST

To generate {0010} from				
a	**b**	**c**	**d**	**Toffoli Netlist**
0	0	1	1	t2 c d
0	1	0	0	t2 b c → t2 c b
0	1	0	1	t2 b d → t2 b c → t2 c b
0	1	1	0	t2 c b
0	1	1	1	t2 c d → t2 c b
1	0	0	0	t2 a c → t2 c a
1	0	0	1	t2 a d → t2 a c → t2 c a
1	0	1	0	t2 c a
1	0	1	1	t2 c d → t2 c a
1	1	0	0	t2 b c → t2 c b → t2 c a
1	1	0	1	t2 b d → t2 b c → t2 c b → t2 c a
1	1	1	0	t2 c b → t2 c a
1	1	1	1	t2 c d → t2 c b → t2 c a

TABLE 4.4. LIBRARY FOR GENERATING {0011} – TOFFOLI NETLIST

To generate {0011} from				
a	**b**	**c**	**d**	**Toffoli Netlist**
0	1	0	0	t2 b d → t2 b c → t3 c d b
0	1	0	1	t2 b c → t3 c d b
0	1	1	0	t2 b d → t3 c d b
0	1	1	1	t3 c d b
1	0	0	0	t2 a d → t2 a c → t3 c d a
1	0	0	1	t2 a c → t3 c d a
1	0	1	0	t2 a d → t3 c d a
1	0	1	1	t3 c d a
1	1	0	0	t2 b d → t2 b c → t3 c d b → t3 c d a
1	1	0	1	t2 b c → t3 c d b → t3 c d a
1	1	1	0	t2 b d → t3 c d b → t3 c d a
1	1	1	1	t3 c d b → t3 c d a

4.3.4 GENERATION OF $f(4)$

If $f(4)! = \{0100\}$, Then library provided in Table 4.5 is consulted.

4.3.5 GENERATION OF $f(5)$

If $f(5)! = \{0101\}$, Then library provided in Table 4.6 is consulted.

TABLE 4.5. LIBRARY FOR GENERATING {0100} – TOFFOLI NETLIST

To generate {0100} from				
a	**b**	**c**	**d**	**Toffoli Netlist**
0	1	0	1	t2 b d
0	1	1	0	t2 b c
0	1	1	1	t2 b d → t2 b c
1	0	0	0	t2 a b → t2 b a
1	0	0	1	t2 a d → t2 a b → t2 b a
1	0	1	0	t2 a c → t2 a b → t2 b a
1	0	1	1	t2 a d → t2 a c → t2 a b → t2 b a
1	1	0	0	t2 b a
1	1	0	1	t2 b d → t2 b a
1	1	1	0	t2 b c → t2 b a
1	1	1	1	t2 b d → t2 b c → t2 b a

TABLE 4.6. LIBRARY FOR GENERATING {0101} – TOFFOLI NETLIST

To generate {0101} from				
a	**b**	**c**	**d**	**Toffoli Netlist**
0	1	1	0	t3 b c d → t3 b d c
0	1	1	1	t3 b d c
1	0	0	0	t2 a d → t2 a b → t3 b d a
1	0	0	1	t2 a b → t3 b d a
1	0	1	0	t2 a d → t2 a c → t2 a b → t3 b d a
1	0	1	1	t2 a c → t2 a b → t3 b d a
1	1	0	0	t2 a d → t3 b d a
1	1	0	1	t3 b d a
1	1	1	0	t2 a d → t2 a c → t3 b d a
1	1	1	1	t2 a c → t3 b d a

4.3.6 GENERATION OF $f(6)$

If $f(6)! = \{0110\}$, Then library provided in Table 4.7 is consulted.

4.3.7 GENERATION OF $f(7)$

If $f(7)! = \{0111\}$, Then library provided in Table 4.8 is consulted.

TABLE 4.7. LIBRARY FOR GENERATING {0110} – TOFFOLI NETLIST

To generate {0110} from				
a	**b**	**c**	**d**	**Toffoli Netlist**
0	1	1	1	t3 b c d
1	0	0	0	t2 a c → t2 a b → t3 b c a
1	0	0	1	t2 a d → t2 a c → t2 a b → t3 b c a
1	0	1	0	t2 a b → t3 b c a
1	0	1	1	t2 a d → t2 a b → t3 b c a
1	1	0	0	t2 a c → t3 b c a
1	1	0	1	t2 a d → t2 a c → t3 b c a
1	1	1	0	t3 b c a
1	1	1	1	t2 a d → t3 b c a

TABLE 4.8. LIBRARY FOR GENERATING {0111} – TOFFOLI NETLIST

To generate {0111} from				
a	**b**	**c**	**d**	**Toffoli Netlist**
1	0	0	0	t2 a d → t2 a c → t2 a b → t4 b c d a
1	0	0	1	t2 a c → t2 a b → t4 b c d a
1	0	1	0	t2 a d → t2 a b → t4 b c d a
1	0	1	1	t2 a b → t4 b c d a
1	1	0	0	t2 a d → t2 a c → t4 b c d a
1	1	0	1	t2 a c → t4 b c d a
1	1	1	0	t2 a d → t4 b c d a
1	1	1	1	t4 b c d a

4.3.8 GENERATION OF $f(8)$

If $f(7)! = \{1000\}$, Then library provided in Table 4.9 is consulted.

TABLE 4.9. LIBRARY FOR GENERATING $\{1000\}$ – TOFFOLI NETLIST

To generate {1000} from				
a	**b**	**c**	**d**	**Toffoli Netlist**
1	0	0	1	t2 a d
1	0	1	0	t2 a c
1	0	1	1	t2 a d → t2 a c
1	1	0	0	t2 a b
1	1	0	1	t2 a d → t2 a b
1	1	1	0	t2 a c → t2 a b
1	1	1	1	t2 a d → t2 a c → t2 a b

4.3.9 GENERATION OF $f(9)$

If $f(9)! = \{1001\}$, Then library provided in Table 4.10 is consulted.

TABLE 4.10. LIBRARY FOR GENERATING $\{1001\}$ – TOFFOLI NETLIST

To generate {1001} from				
a	**b**	**c**	**d**	**Toffoli Netlist**
1	0	1	0	t3 a c d → t3 a d c
1	0	1	1	t3 a d c
1	1	0	0	t3 a b d → t3 a d b
1	1	0	1	t3 a d b
1	1	1	0	t3 a b d → t3 a d c → t3 a d b
1	1	1	1	t3 a d c → t3 a d b

4.3.10 GENERATION OF $f(10)$

If $f(10)! = \{1010\}$, Then library provided in Table 4.11 is consulted.

TABLE 4.11. LIBRARY FOR GENERATING {1010} – TOFFOLI NETLIST

To generate {1010} from				
a	**b**	**c**	**d**	**Toffoli Netlist**
1	0	1	1	t3 a c d
1	1	0	0	t3 a b c → t3 a c b
1	1	0	1	t3 a b d → t3 a b c → t3 a c b
1	1	1	0	t3 a c b
1	1	1	1	t3 a b d → t3 a c b

4.3.11 GENERATION OF $f(11)$

If $f(11)! = \{1011\}$, Then library provided in Table 4.12 is consulted.

TABLE 4.12. LIBRARY FOR GENERATING {1011} – TOFFOLI NETLIST

To generate {1011} from				
a	**b**	**c**	**d**	**Toffoli Netlist**
1	1	0	0	t3 a b d → t3 a b c → t4 a c d b
1	1	0	1	t3 a b c → t4 a c d b
1	1	1	0	t3 a b d → t4 a c d b
1	1	1	1	t4 a c d b

4.3.12 GENERATION OF $f(12)$

If $f(12)! = \{1100\}$, Then library provided in Table 4.13 is consulted.

TABLE 4.13. LIBRARY FOR GENERATING {1100} – TOFFOLI NETLIST

To generate {1100} from				
a	**b**	**c**	**d**	**Toffoli Netlist**
1	1	0	1	t3 a b d
1	1	1	0	t3 a b c
1	1	1	1	t3 a b d → t3 a b c

4.3.13 GENERATION OF $f(13)$

If $f(13)! = \{1101\}$, Then library provided in Table 4.14 is consulted.

TABLE 4.14. LIBRARY FOR GENERATING $\{1101\}$ – TOFFOLI NETLIST

To generate {1101} from				
a	**b**	**c**	**d**	**Toffoli Netlist**
1	1	1	0	t4 a b c d → t4 a b d c
1	1	1	1	t4 a b d c

4.3.14 GENERATION OF $f(14)$

If $f(14)! = \{1110\}$, Then library provided in Table 4.15 is consulted.

TABLE 4.15. LIBRARY FOR GENERATING $\{1110\}$ – TOFFOLI NETLIST

To generate {1110} from				
a	**b**	**c**	**d**	**Toffoli Netlist**
1	1	1	1	t4 a b c d

As discussed earlier, the synthesis algorithm provides optimized results due to the fact that with each progressing step, few combinations are utilized leaving behind lesser number of combinations. Since, one combination has been used once, it cannot be reused.

4.4 FLOW CHART

Figure 4.1 presents the flowchart for the library based synthesis algorithm.

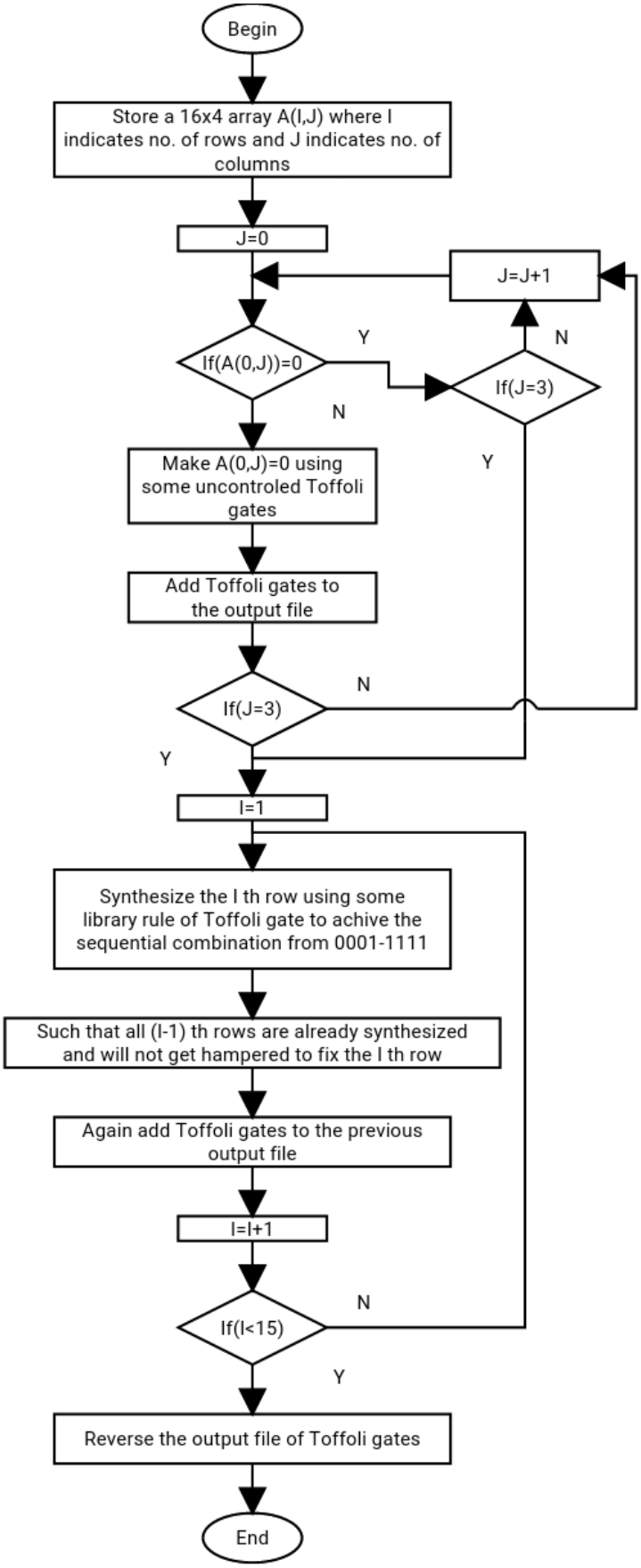


Figure 4.1. Flow Chart Library based Synthesis Algorithm

4.5 COMPARATIVE ANALYSIS

Synthesis of literary four variable reversible gate proposals has been done using the proposed library based synthesis algorithm. Initially the synthesis for the gates has been done using the Unidirectional Algorithm of [**29**] [**30**]. The results are already available at [**68**]. These results have been compared with the results of the proposed algorithm. The comparisons have been done on two parameters – number of Toffoli gates used and the quantum cost of the model. Table 4.16 presents the comparison results.

TABLE 4.16. COMPARATIVE ANALYSIS

Reversible Gate	Unidirectional Algorithm [29] [30]		Proposed Algorithm	
	#	QC	#	QC
ALG [56]	10	42	10	26
DFG [55]	16	46	13	37
FAG [55]	11	19	10	16
HNFG [52] [53]	5	5	5	5
HNG [49]	4	8	4	8
IG [44]	3	9	4	10
MKG [52]	9	21	9	21
RPS [51]	17	99	11	55
SCG [50]	15	43	14	38

[a] #: Gate Count

[b] QC: Quantum Cost

Figure 4.2 and Figure 4.3 present the results of the comparative analysis graphically. It can be deduced that except for a few, in most of the cases, the proposed synthesis algorithm provides better results. Also in one particular case, the proposed algorithm has higher values for the performance parameters. In all the rest cases, the proposed algorithm is better.

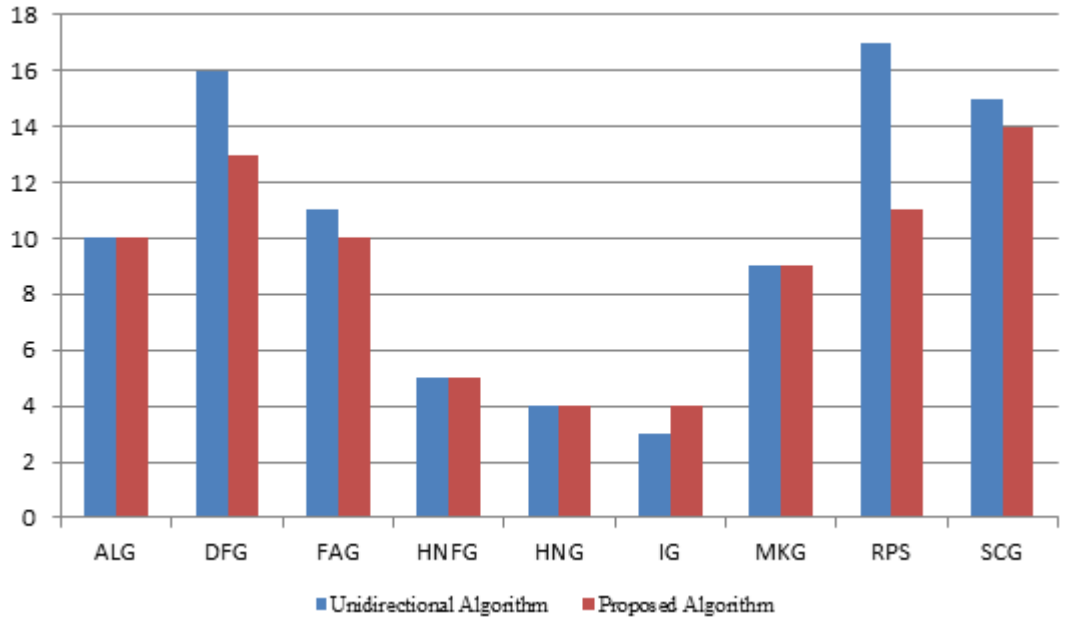


Figure 4.2. Graphical representation for comparative analysis – Gate Count

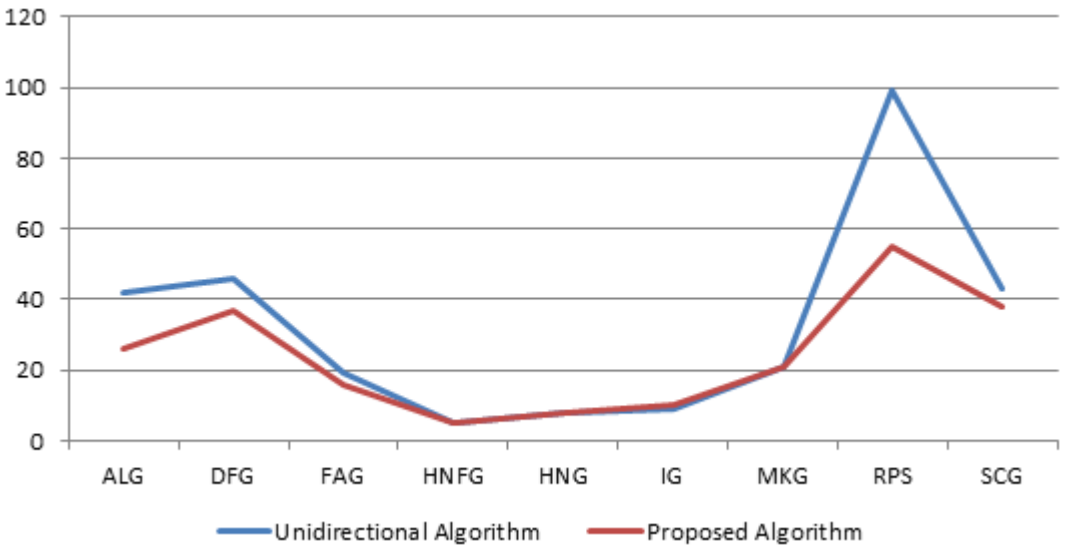


Figure 4.1. Graphical representation for comparative analysis – Quantum Cost

4.6 CONCLUSION

A novel synthesis algorithm has been proposed which is based on library based models for Control Line Set identification. Pre-determined values for control lines have been placed in the library and fetched at iteration. Comparison has been done with respect to Unidirectional Algorithm.

Future expansion of the work can be done to compare with respect to other synthesis algorithms in literature. Comparisons with other library based algorithms can also be done.

CHAPTER 5

Design of Reversible Decimal Counter

Mahamuda Sultana, Ayan Chaudhuri, Diganta Sengupta, Debashis De, Atal Chaudhuri, "Design of Synchronous Decimal Counter using Reversible Toffoli-Fredkin Netlist", Journal of Circuits, Systems, and Computers, World Scientific Publishing Company (SCI Indexed- Impact Factor 0.595) (Communicated).

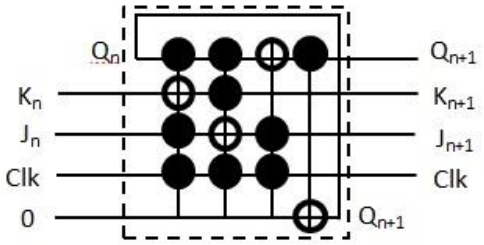
5 DESIGN OF REVERSIBLE DECIMAL COUNTER

Rise of emerging technologies in the last two decades have witnessed multiple new dimensions for designing computer architecture. One of the most interesting topic for global researchers have been the Reversible Logic. Reversibility is a condition which allows near zero heat dissipation for designed circuits. Reversibility can be obtained through fundamental reversible gates, viz. Toffoli Gate, Fredkin Gate, and Peres gate. Most of the proposals in literature for architecture designs have been done using the Toffoli Gate. A research dimension has been active which converts the existing Boolean functions in to reversible circuits using the Toffoli Gates. Of all the Boolean specification conversions into reversible domain, code converters [**74**], comparators [**75**], adders [**76**] [**77**] form the most widely proposed designs in the combinational part of digital electronics with proposals like finite state machines [**65**] contributing to the sequential domain. Decimal counters form an integral part of the digital domain as it finds huge acceptance in designing frequency dividers, integrated oscillations, and clock generators to name a few. Research has been conducted in the reversible counter domain but mostly for designing mod-2/4/8/16 counters. Designs for decimal counters have not been observed in literature. This chapter proposes a decimal counter in the synchronous domain for reversible implementation. The design has been in three stages. The first stage termed as 'Design D1' proposes a decimal up counter using Toffoli Netlist. The second design (Design D2) presents a decimal down counter design in the reversible domain. The final design called as the 'Design D3' is the combination of the two aforementioned designs and exhibits decimal UP/Down counter in the reversible domain. The proposed counter designs have been tried for optimization but they reflect the best optimized designs. The synchronous counter choice has been made because they work on a single clock and are easy to debug.

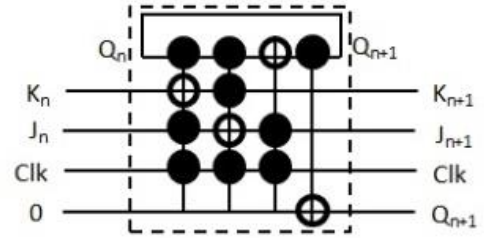
5.1 PROPOSED REVERSIBLE DECIMAL COUNTER

This section describes three designs – Design DI, Design D2, and Design D3. These three designs have been described in the following sections. The designs have been done using JK Flip Flop recursively. For sake of reversibility, reversible JK Flip Flop has been used. JK Flip Flop provides less complex architectures with minimal feedbacks compared to other flip flop versions. Hence, the choice for JK Flip-Flop. The reversible JK Flip Flop presented by Chuang and Wang [**78**] has been slightly modified and implemented for design-

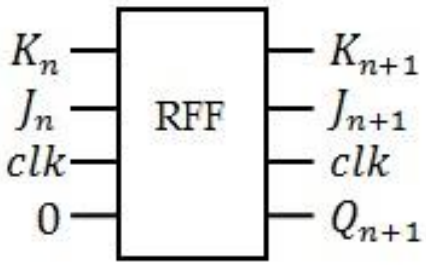
ing D1, D2 and D3. The design in [**78**] is presented in Figure 5.1a. The modification has been presented in Figure 5.1b. In [**78**], the feedback was done from the bottom Q_{n+1}, but in the modified version, the feedback is done from the topmost Q_{n+1}. It has been done to provide a seamless transformation from the input to the output. Readers are requested to follow the complete illustration provided in [**78**] to grasp the working principle of the JK Flip Flop presented in Figure 5.1a.



(a)



(b)

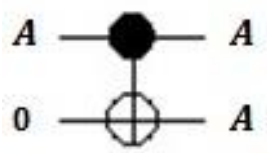


(c)

Figure 5.1. a. Reversible JK Latch in [**78**] b. Changed architecture c. Block Diagram for Figure. 5.2b

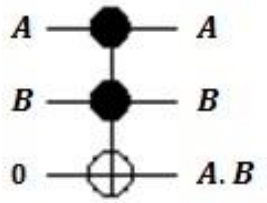
Figure 5.1c presents the schematic for the modified reversible JK Flip-Flop. This Flip Flop has been used in the following section to design the three designs D1 through Design D3.

Two-input Toffoli Gate can be used to generate copy signals as well as inversion as shown in shown in Figure 5.2a and Figure 5.2b. 3-input Toffoli Gate can be used to implement the Boolean AND operation with certain combination of inputs.



(a)

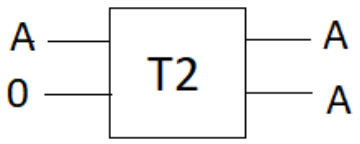
(b)



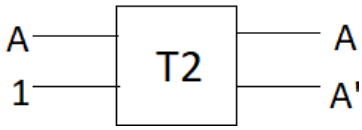
(c)

Figure 5.2. a. Copy signal done using Two-input Toffoli Gate b. Signal inversion done through Two-input Toffoli Gate c. AND operation done using Three-input Toffoli Gate

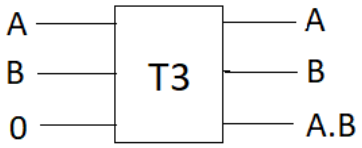
The three reversible implementations presented in Figure 5.2a, Figure 5.2b, and Figure 5.2c have been represented in terms of block diagrams in Figure 5.3a, Figure 5.3b, and Figure 5.3c.



(a)



(b)



(c)

Figure 5.3. Block Diagram representation of Figure 5.2a through Figure 5.2c

5.2 DECIMAL UP COUNTER (DESIGN D1)

Table 5.1 presents the design of the decimal up counter excitation table. The don't care conditions have been reflected using 'x'. K-Map method has been used to derive the characteristic equations. The counter implementation in Table 5.1 is of UP counter.

TABLE 5.1. SYNCHRONOUS UP COUNTER (EXCITATION TABLE)

Q_3	Q_2	Q_1	Q_0	Q_3'	Q_2'	Q_1'	Q_0'	J_3	K_3	J_2	K_2	J_1	K_1	J_0	K_0
0	0	0	0	0	0	0	1	0	x	0	x	0	x	1	x
0	0	0	1	0	0	1	0	0	x	0	x	1	x	x	1
0	0	1	0	0	0	1	1	0	x	0	x	x	0	1	x
0	0	1	1	0	1	0	0	0	x	1	x	x	1	x	1
0	1	0	0	0	1	0	1	0	x	x	0	0	x	1	x
0	1	0	1	0	1	1	0	0	x	x	0	1	x	x	1
0	1	1	0	0	1	1	1	0	x	x	0	x	0	1	x
0	1	1	1	1	0	0	0	1	x	x	1	x	1	x	1
1	0	0	0	1	0	0	1	1	x	0	x	0	x	1	x
1	0	0	1	0	0	0	0	x	1	0	x	0	x	x	1
1	0	1	0	x	x	x	x	x	x	x	x	x	x	x	x
1	0	1	1	x	x	x	x	x	x	x	x	x	x	x	x
1	1	0	0	x	x	x	x	x	x	x	x	x	x	x	x
1	1	0	1	x	x	x	x	x	x	x	x	x	x	x	x
1	1	1	0	x	x	x	x	x	x	x	x	x	x	x	x
1	1	1	1	x	x	x	x	x	x	x	x	x	x	x	x

$$J_3 = Q_0 Q_1 Q_2 \quad (4)$$

$$K_3 = Q_0 \quad (5)$$

$$J_2 = Q_0 Q_1 \quad (6)$$

$$K_2 = Q_0 Q_1 \quad (7)$$

$$J_1 = Q_0 \overline{Q_3} \quad (8)$$

$$K_1 = Q_0 \quad (9)$$

$$J_0 = 1 \quad (10)$$

$$K_0 = 1 \quad (11)$$

It can be observed from Equation (4) till Equation (9) that most of the outputs need fan out more than one. For fan out more than one, signal duplication is required, which is achieved by the concept provided earlier using two-input Toffoli Gates. Moreover, Equation (8) reflects signal inversion. Hence, two-input Toffoli gate is again used to generate the signal inversion of $\overline{Q_3}$. Figure 5.4 presents the synchronous decimal up counter. Equation (4) through Equation (11)have been used to design the schematic of Figure 5.4.

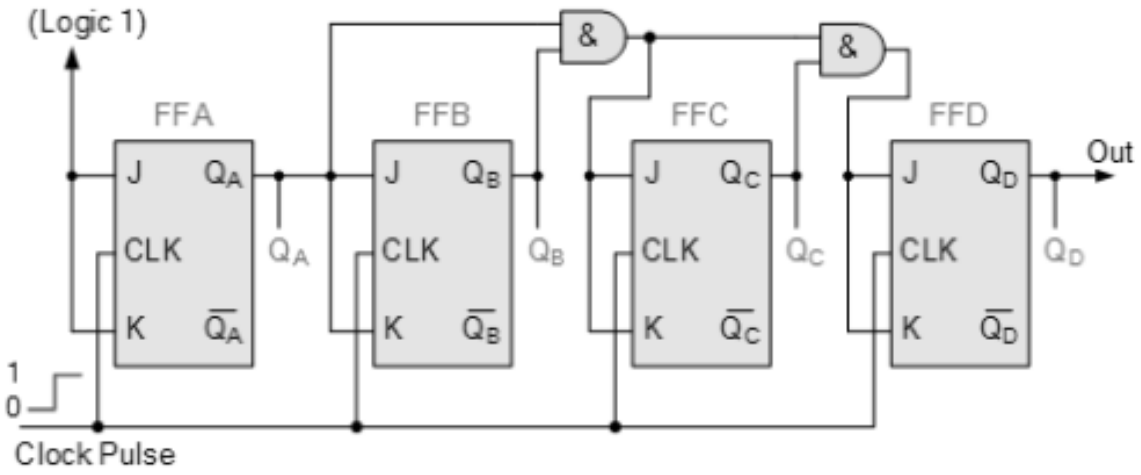


Figure 5.4. Decimal Synchronous Up Counter

Converting the counter design for Decimal up counter into the reversible domain, reversible JK Flip Flop presented in Figure 5.2c has been implemented as a replacement for the JK Flip Flop of Figure 5.4. The complete design using the reversible flip flop is presented in Figure 5.5. The fan outs have been taken care of using the designs presented in Figure 5.2.

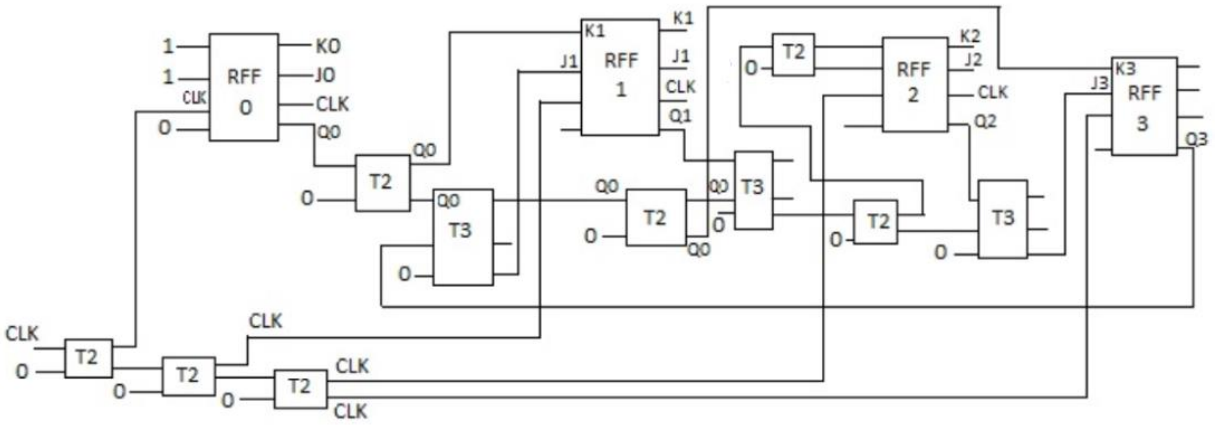


Figure 5.5. Decimal Up Counter (Synchronous) – Reversible version

Figure 5.6 presents the reversible Toffoli gate design for Figure 5.5. All the individual blocks of Figure 5.5 have been replaced by the Toffoli counterparts presented in Figure 5.1 and Figure 5.2. The design in Figure 5.6 is that of Design D1.

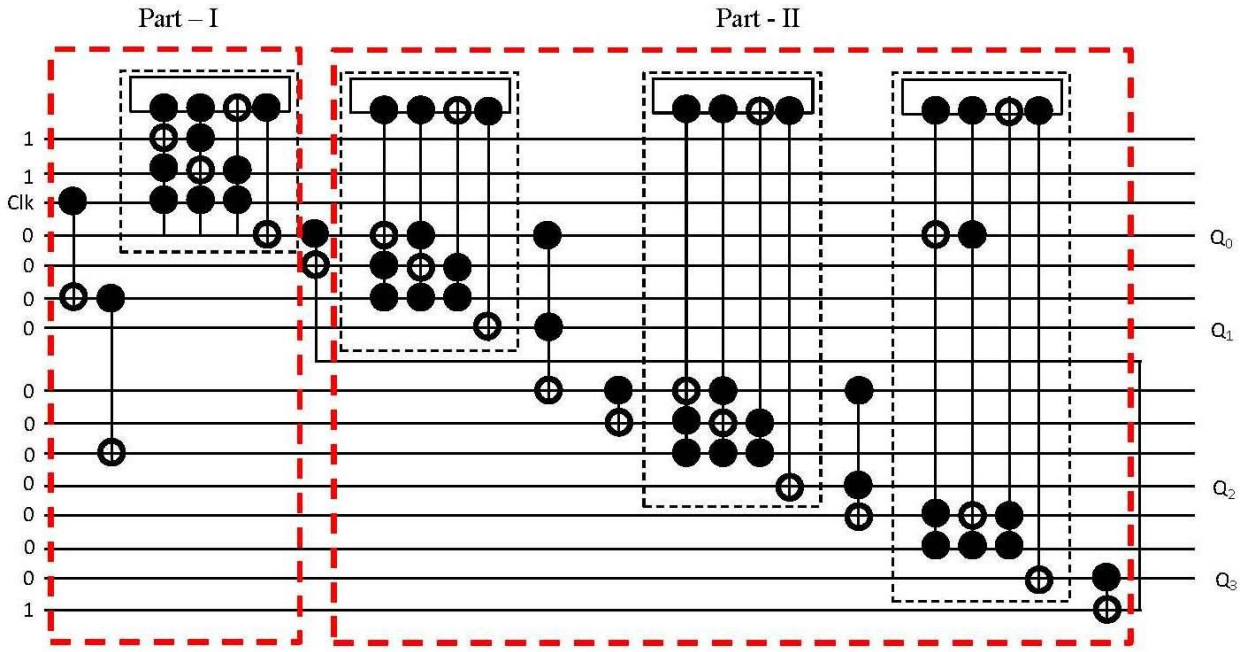


Figure 5.6. Toffoli realization of design presented in Figure 5.5

Figure 5.6 reflects the complete design has two parts – Part I resembling forward flow and Part II, also resembling forward flow. The two parts are connected via a feedback having a two-input Toffoli gate. Part I has been provided in Figure 5.7 whereas Part II has been presented in Figure 5.8.

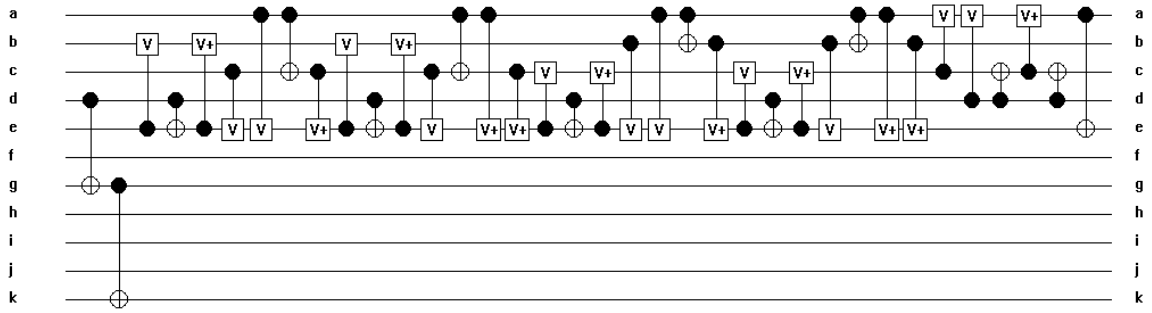


Figure 5.7. Part I Decomposed $V - V^{+} - TOFF2$ architectures

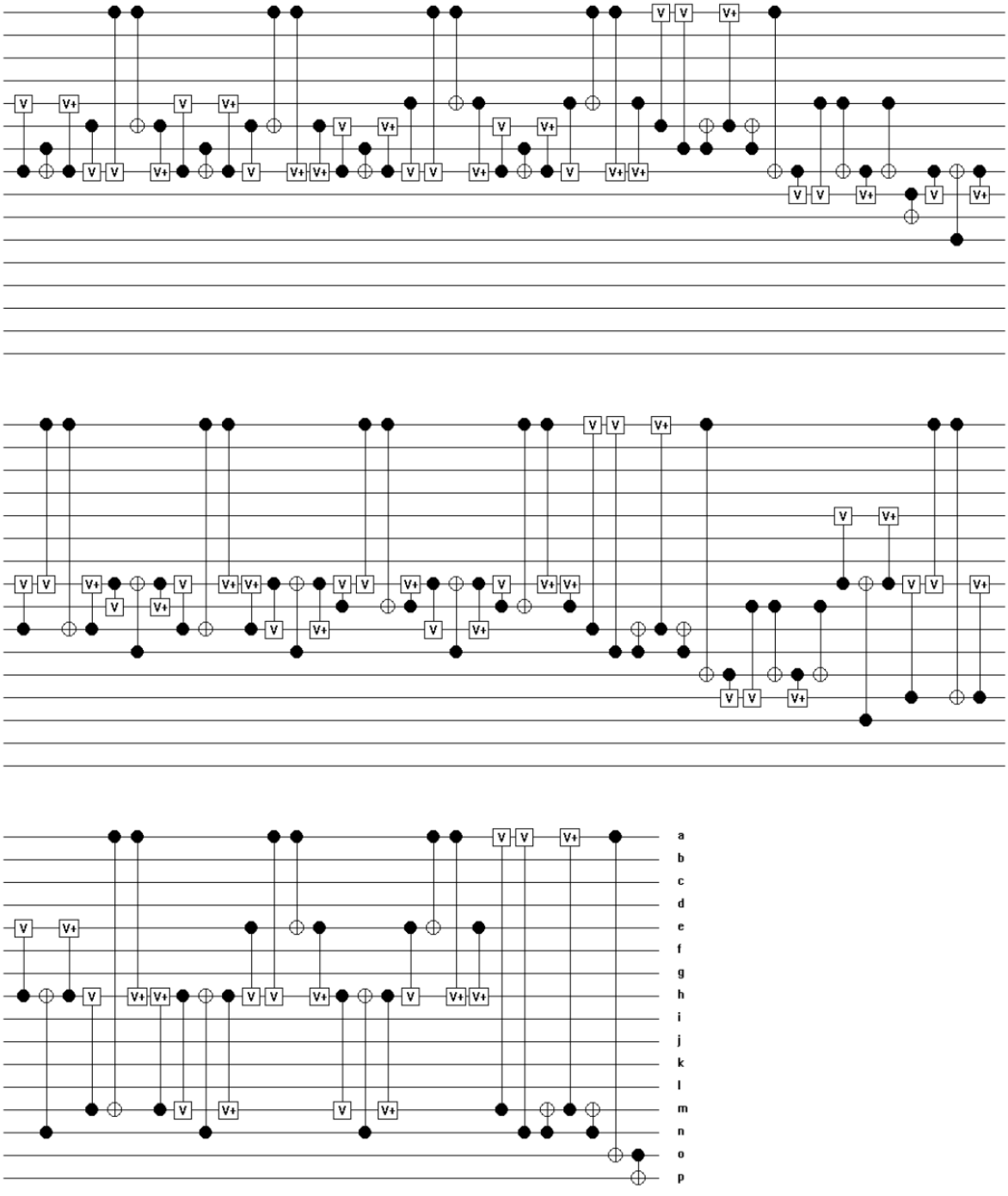


Figure 5.8. Part II Decomposed $V - V^{+} - TOFF2$ architectures

5.3 DECIMAL DOWN COUNTER (DESIGN D2)

Like the Up Counter, the Down Counter has been designed using excitation table provided in Table 5.2. The K-Map method has been utilized to form the characteristic equations.

TABLE 5.2. SYNCHRONOUS DOWN COUNTER (EXCITATION TABLE)

Q_3	Q_2	Q_1	Q_0	Q_3'	Q_2'	Q_1'	Q_0'	J_3	K_3	J_2	K_2	J_1	K_1	J_0	K_0
0	0	0	0	1	0	0	1	1	x	0	x	0	x	1	x
0	0	0	1	0	0	0	0	0	x	0	x	0	x	x	1
0	0	1	0	0	0	0	1	0	x	0	x	x	1	1	x
0	0	1	1	0	0	1	0	0	x	0	x	x	0	x	1
0	1	0	0	0	0	1	1	0	x	x	1	1	x	1	x
0	1	0	1	0	1	0	0	0	x	x	0	0	x	x	1
0	1	1	0	0	1	0	1	0	x	x	0	x	1	1	x
0	1	1	1	0	1	1	0	0	x	x	0	x	0	x	1
1	0	0	0	0	1	1	1	x	1	1	x	1	x	1	x
1	0	0	1	1	0	0	0	x	0	0	x	0	x	x	1
1	0	1	0	x	x	x	x	x	x	x	x	x	x	x	x
1	0	1	1	x	x	x	x	x	x	x	x	x	x	x	x
1	1	0	0	x	x	x	x	x	x	x	x	x	x	x	x
1	1	0	1	x	x	x	x	x	x	x	x	x	x	x	x
1	1	1	0	x	x	x	x	x	x	x	x	x	x	x	x
1	1	1	1	x	x	x	x	x	x	x	x	x	x	x	x

Figure 5.10 presents the reversible decimal down counter (synchronous). The Toffoli Netlist for the design in Figure 5.10 is shown in Figure 5.11. The RDFF shown in Figure 5.9 is the RFF shown in Figure 5.2c with an inverter at the output of RFF. The inverter has been designed using two-input Toffoli Gate as discussed earlier. The characteristic equations have been derived from Table 5.2 using K-Map method and are provided in Equation (12) through Equation (19).

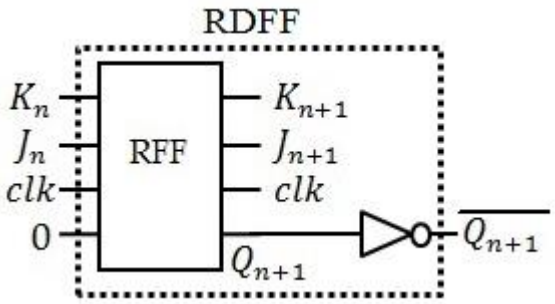


Figure 5.9. Schematic for RDFF

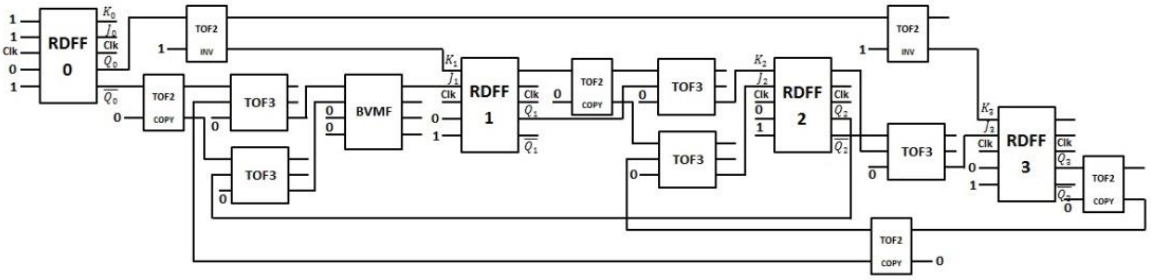


Figure 5.10. Synchronous Decimal Down Counter (Reversible)

$$J_3 = \overline{Q_0}.\overline{Q_1}.\overline{Q_2} \quad (12)$$

$$K_3 = \overline{Q_0} \quad (13)$$

$$J_2 = \overline{Q_0}.Q_3 \quad (14)$$

$$K_2 = \overline{Q_0}.\overline{Q_1} \quad (15)$$

$$J_1 = \overline{Q_0}.Q_3 + \overline{Q_0}.Q_2 \quad (16)$$

$$K_1 = \overline{Q_0} \quad (17)$$

$$J_0 = 1 \quad (18)$$

$$K_0 = 1 \quad (19)$$

For implementation of Equation (16) BVMF gate has been used. The readers are requested to refer [**62**] to get the complete details for the BVMF Gate. Choice for BVMF Gate has been made after acute analysis of the quantum metric for all the reversible gates. Equation (16) is being generated using BVMF Gate in the most efficient manner. Decomposition of Figure 5.11 could not be done as the design has multiple feedback paths.

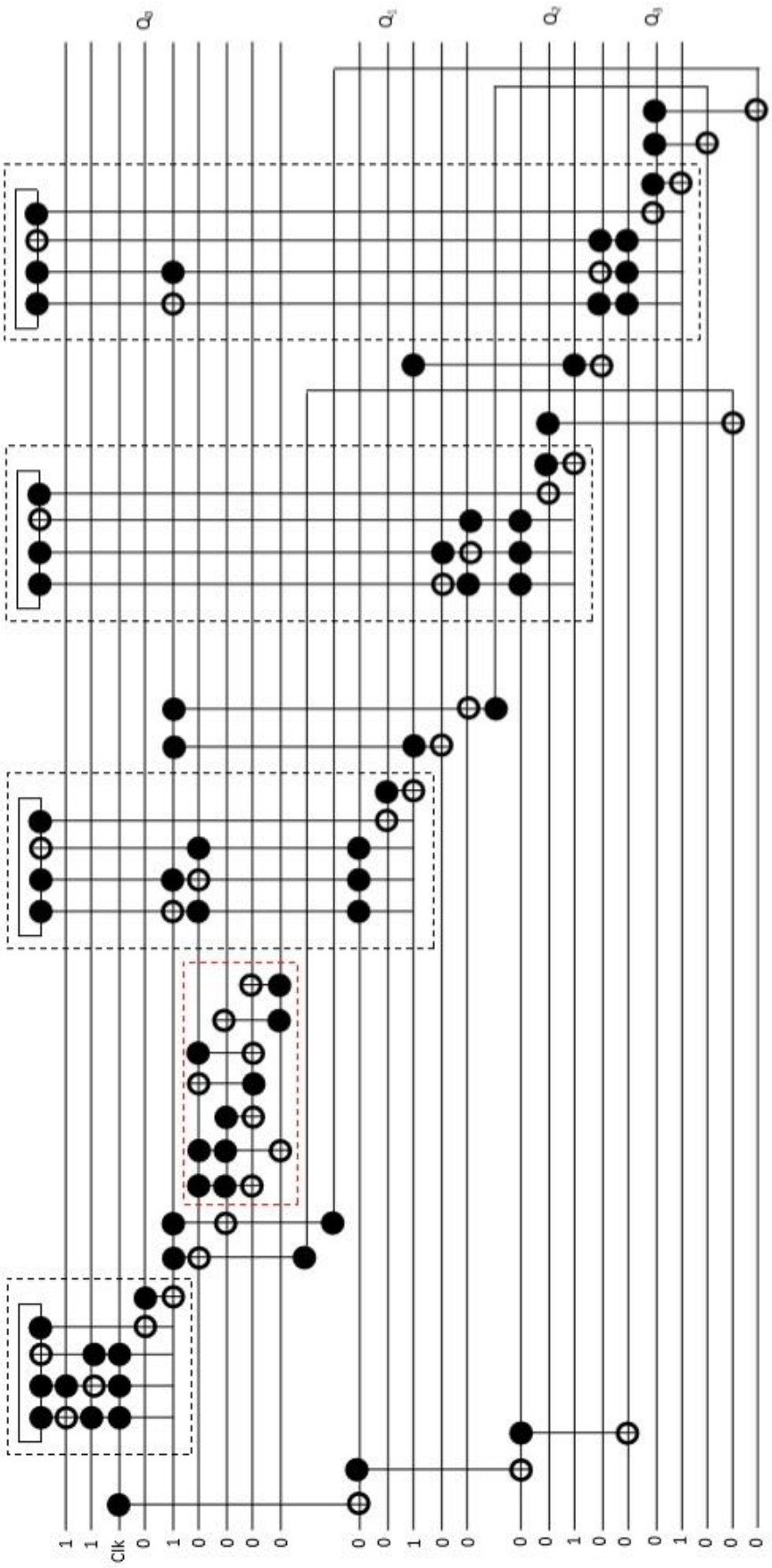


Figure 5.11. Reversible Synchronous Decimal Down Counter (Toffoli Netlist)

5.4 SYNCHRONOUS REVERSIBLE DECIMAL COUNTER (DESIGN D3)

The complete design implementing the synchronous reversible decimal counter has been generated from both the designs, Design D1 and Design D2. The selection of a particular design among D1 and D2 is done by a control signal $DN/\overline{UP}$ which selects one of the outputs from the up or the down counter. The selection procedure is conducted using a Fredkin gate array. The design is shown in Figure 5.12. The dotted part on the right exhibits the Fredkin gate array. Basically, the Fredkin gate array basically works as a 2:1 array-multiplexer. It selects one of the two output vectors generated from up counter and down counter to the output depending upon the control signal $DN/\overline{UP}$. The control signal acts as the select line to the multiplexer. Hence, the design in Figure 5.12 behaves according to the condition below.

$$D3 = \begin{Bmatrix} Up\ Counter\ if DN/\overline{UP} = 0 \\ Down\ Counter\ if DN/\overline{UP} = 1 \end{Bmatrix}$$

5.5 COMPARATIVE ANALYSIS

The comparative analysis has been done with literary proposals with respect to Gate Count, Number of 2-Qubit Gates, and Quantum Cost. These results have been presented in Table 5.3. Table 5.4 provides the analysis based on $V - V^+ - TOFF2$ gates for the decomposed architectures.

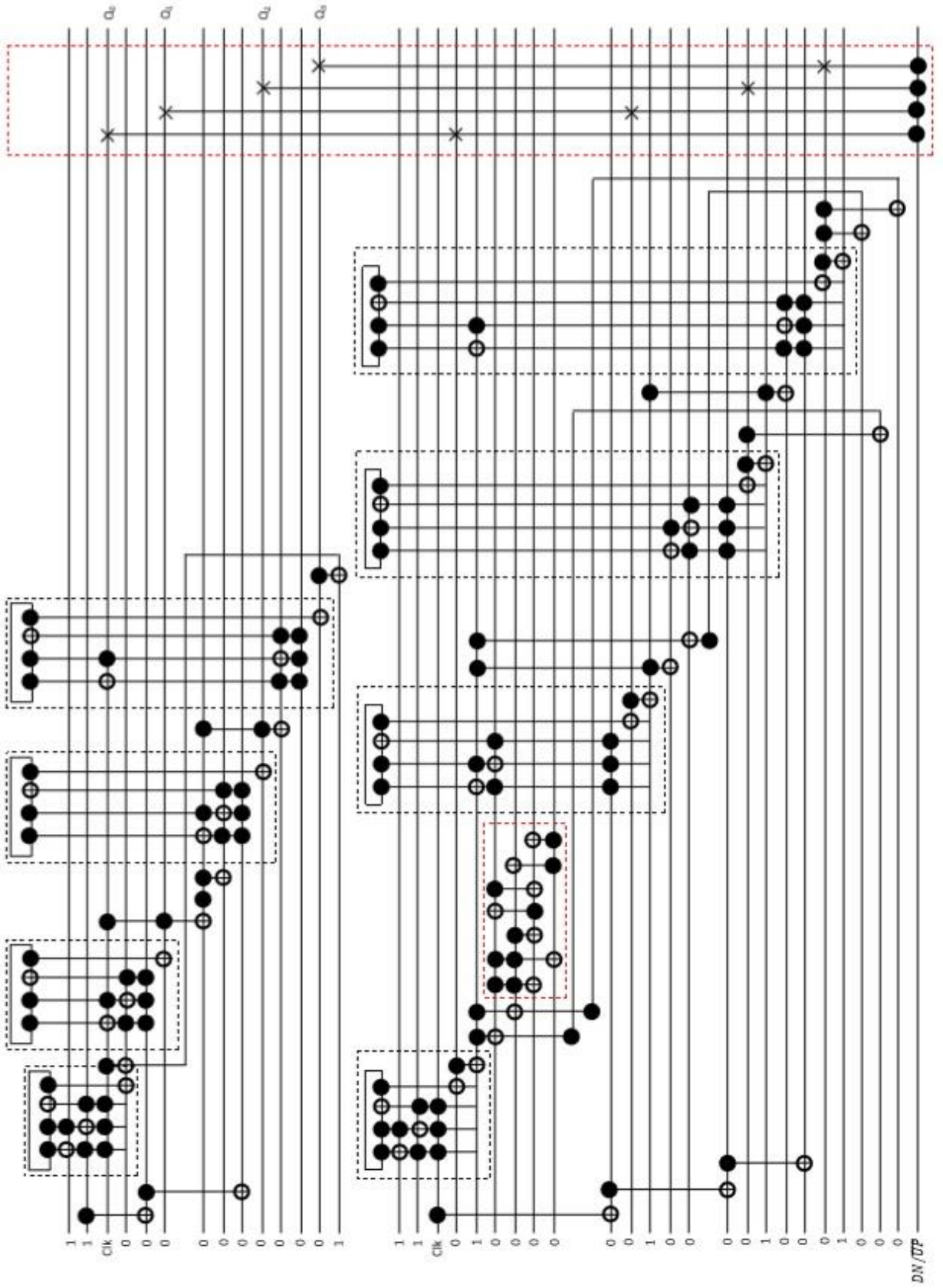


Figure 5.12. Reversible Decimal Counter (Synchronous) (Toffoli Netlist)

TABLE 5.3. QUANTUM METRIC ANALYSIS (UP COUNTER, DOWN COUNTER, UP/DOWN COUNTER)

Design	Toffoli Gate Count	Quantum Cost	Two-Qubit Gate Count
Up Counter (Part – I)	6	34	18
Up Counter (Part – II)	16	108	60
Complete Up Counter (Figure 7) including TOF2 between Part – I and Part - II	23	143	79
Down Counter (Figure 11)	38	178	114
Reversible Up/Down Counter (Figure 12)	65	349	213

Table 5.3 and Table 5.4 reflect that quantum cost is better when the proposed designs are implemented using Toffoli Gates rather than when implemented using $V - V^{+} - TOFF2$ gates.

TABLE 5.4. QUANTUM METRIC ANALYSIS (DECOMPOSED DESIGNS)

Decomposed Design	Gate Count	Quantum Cost	Two-Qubit Gate Count
Up Counter (Part – I, Figure 8)	36	36	36
Up Counter (Part – II, Figure 9)	114	114	114
Complete Up Counter including TOF2 between Part – I & Part - II	151	151	151
Down Counter	186	186	186
Fredkin Gate Array	28	28	28
Reversible Up/Down Counter	365	365	365

Comparative analysis of all the counter proposals in literature has been provided in Table 5.5. It can be observed from Table 5.5 that multiple designs have been proposed for Mod-16 counter. Moreover very few proposals contain Toffoli gate implementations. The proposed design in this chapter has been compared with [**63**] [**64**] [**65**] [**66**] [**67**] [**79**] [**80**].

TABLE 5.5. COMPARATIVE ANALYSIS OF REVERSIBLE COUNTERS

Proposal	A	S	Mod-4	Mod-8	Mod-10 (Decimal)	Mod-16	D
Ref [**63**]	√	×	×	√	×	√	√
Ref [**64**]	√	√	×	×	×	√	×
Ref [**65**]	×	√	√	×	×	×	√
Ref [**66**]	√	√	×	×	×	√	×
Ref [**67**]	×	√	×	×	×	√	×
Ref [**79**]	√	×	×	×	×	√	×
Ref [**80**]	×	√	×	×	×	√	×
Proposed	×	√	×	×	√	×	√

A: Asynchronous, S: Synchronous, D: Design using Toffoli Gates

5.6 CONCLUSION

Counter designs in literature reflect numerous proposals for mod-4, mod-8, mod-16 counters. Extensive survey failed to provide any proposal for decimal counter. This chapter presents a reversible counterpart for the traditional decimal counter. Designs have been done primarily using Toffoli gates with Fredkin gates at the final level. Use of Toffoli and Fredkin gates assert the proposed designs to be reversible in nature. The complete design for the decimal counter has been approached in a threefold manner. First, design for decimal up counter in the reversible domain has been proposed. Secondly, decimal down counter proposal has been made using Toffoli gates. The final design incorporates the two designs to generate the complete decimal counter. The final design uses a Fredkin gate array to se-

lect the desired output from the two outputs from the up and the down counter respectively.

The study in this chapter can be further extended to incorporate the asynchronous counterpart. Also the counter proposal can be implemented using QCA cells.

CHAPTER 6

Taxonomy for research on Reversible Logic

Mahamuda Sultana, Diganta Sengupta, Atal Chaudhuri, "Taxonomy Proposal for Research on Reversible Logic", International Journal of Engineering and Advanced Technology, Volume 8, Issue 6. (SCOPUS).

6 TAXONOMY FOR RESEARCH ON REVERSIBLE LOGIC

Mushrooming publications in any specific research domain has made it time consuming and harsh for researchers to select articles of their choice of interest. While conducting literature survey, huge time and efforts are wasted in shortlist publications which form important in relevant field of study. It is in this interest that the present chapter provides a taxonomy for research on reversible logic.

Through the journey of the current research, it was observed that multiple numbers of articles had been considered before concentrating on those articles which formed aligned to the topic of this thesis. But none-the-less all those articles were traversed for gaining the knowledge. Hence, many articles were rejected from the vocabulary for this thesis which had been consulted and found inappropriate to the point of interest. Since the articles have been consulted and the important points noted, hence this chapter was designed to help future researchers with their literature survey and reduce their time for conducting their literature study.

Nowadays, massive collections of articles are observed in the internet and online indexing databases. Since it is tough to explore the complete literature, a taxonomy becomes handy in shortlisting the articles according to some topic nodes. Out of the many databases, this chapter has selected IEEE Xplore as the choice of database for generating the taxonomy. Taxonomy generation requires recognition of a vocabulary as has been illustrated in [**81**]. As discussed the vocabulary is IEEE Xplore for this chapter. IEEE Xplore has been chosen because there is an option is IEEE Xplore where bibliographic data can be easily downloaded as a .csv file. Other important databases comprise of SCOPUS, INSPEC etc. which can be consulted in future for generating the comprehensive taxonomy for reversible logic research. Initial taxonomy generating systems were inefficient as they could not retrieve information logically as discussed in [**82**]. Only facts were provided to the reader but with technological advancements, it is possible to get hold of proper systems and generate the taxonomy. Taxonomy used to be generally associated with biology and zoology [**83**] [**84**]. Technical taxonomies can be found in [**85**] [**86**]. A taxonomy proposal on reversible logic gates has been provided in [**87**] but is the partial view of the field. This chapter provides a wider view of the subject. A comprehensive survey of synthesis and optimization algorithms for reversible logic has been provided in [**88**] whereas a comprehensive survey of four input four output reversible gates has been found in [**89**]. These two articles have been consulted while generating the taxonomy as they provided the basic de-

tails regarding one dimension of research on reversible logic.

6.1 VOCABULARY CREATION

The vocabulary has been generated from IEEE Xplore using the seed term 'Reversible Logic'. Altogether 1071 articles were fetch which were grouped into seven different types as shown in Table 6.1. IEEE Xplore provided the bibliographic data in terms of 29 attributes. These 29 attributes are mentioned in Table 6.2

Table 6.1. TYPES OF PUBLICATIONS

Article Type	Article Count	Article Type	Article Count
Conferences	929	Courses	1
Journals	120	Books	5
Magazines	8	Standards	3
Early Access Articles	5		

Table 6.2. 29 ATTRIBUTES IN BIBLIOGRAPHIC FILE

Document Title	Abstract	Mesh_Terms
Authors	ISSN	Article Citation Count
Author Affiliations	ISBNs	Reference Count
Publication Title	DOI	License
Date Added To Xplore	Funding Information	Online Date
Publication_Year	PDF Link	Issue Date
Volume	Author Keywords	Meeting Date
Issue	IEEE Terms	Publisher
Start Page	INSPEC Controlled Terms	Document Identifier
End Page	INSPEC Non-Controlled Terms	

It can be observed from Table 6.1 that there are seven types of publications. Of these, this chapter concentrates on Journals only. It can be seen that the journal count is 120. Bibliog-

raphy for these 120 journals was shortlisted and then the taxonomy was generated from them. All these 120 articles have all the 29 attributes mentioned in Table 6.2. Hence a database was created to store all the bibliographic data for all these 120 journal articles. The database was then searched for errors and redundancies. 11 articles were found to be redundant. The articles were excluded from the database and fresh database was generated containing the remaining 109 articles. Also this database contained only seven attributes in contrast to the 29 initial attributes. Six of seven attributes were from the initial database. One more attribute was added to identify each article in the vocabulary. This new attribute was termed as PID (Paper Id). Hence, each of the 109 articles had a unique PID. This database was termed as 'metadata'. The seven attributes in 'metadata' are mentioned in Table 6.3. The PID has been assigned according to Equation (1).

$$PIDi = Taxonomy\ Bibliography \rightarrow i\ \forall \in [1{,}109] \quad (1)$$

Note: The taxonomy Bibliography is provided after the bibliography in the thesis.

Table 6.3. ATTRIBUTES IN 'METADATA'

PID	Abstract
Title	Author Keywords
Authors	Document Type
Year of publication	

Since, the taxonomy has been generated using journals only, hence the attribute 'Document Type' in Table 6.3 is redundant.

From 'metadata', the attribute 'year' helped in generating the research growth in reversible logic over the years. Figure 6.1 presents the research growth. Figure 6.1 has been generated from Table 6.4 which provides the year wise publication data.

It can be observed that there has been a positive research growth in the domain for the last three years. There has been a slight dip in growth in 2017, but again the growth has been positive in the last year. Therefore, it can be claimed that the topic is of interest to the research community in recent times. Figure 6.1 also reflects that the topic 'Reversible Logic' has gained momentum in the past decade only.

Table 6.4. YEAR WISE GROWTH OF RESEARCH

Year	Journals	Year	Journals	Year	Journals
1956	2	1991	1	2008	5
1958	1	1993	1	2009	2
1960	1	1996	1	2010	5
1962	1	1998	3	2011	5
1963	1	1999	4	2012	3
1965	1	2000	2	2013	4
1966	1	2001	1	2014	7
1969	1	2002	1	2015	8
1973	1	2003	3	2016	8
1974	1	2004	2	2017	4
1978	1	2005	4	2018	9
1984	1	2006	4		
1987	1	2007	3		

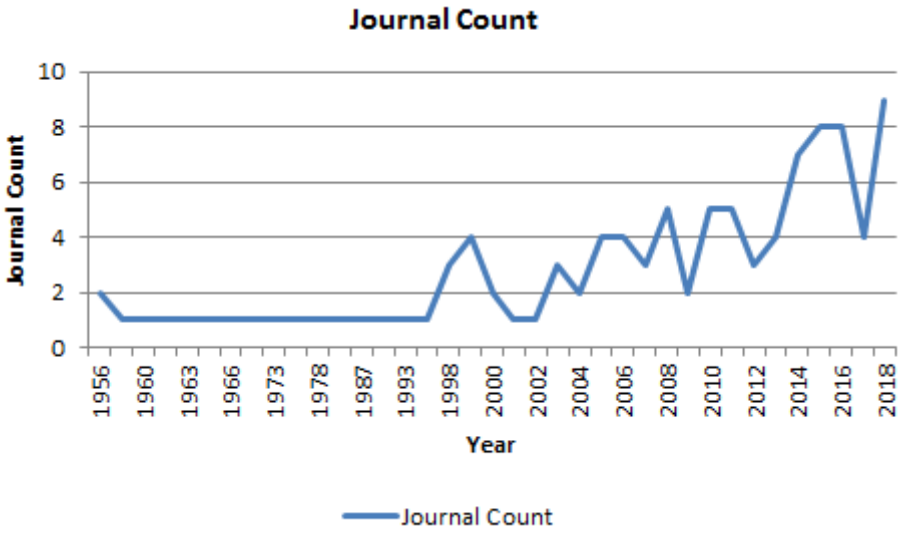


Figure 6.1. Research growth in ‘Reversible Logic’

6.2 GENERATION OF ‘KEYWORD’ AND ‘ABSTRACT’ DATABASE

After generation of ‘metadata’, two more databases were generated. One database stored the PIDs along with the abstracts. This database has been termed as ‘Abstract’ Database because it holds the abstract for all the 109 articles along with their paper Ids.

A third database named ‘keyword’ database was generated to store all the unique

Author Keyword from all the 109 articles. Altogether 221 unique Author Keywords were extracted from all the papers in metadata. After removing redundancies, only 12 keywords formed relevant. It is because Author Keywords like ‘reversiblelogic’, ‘reversible logic’, ‘reversible logic concept’, ‘Reversible Logic’, ‘Reversiblelogic’ etc. were found. These all were grouped into a single keyword (phrase) since all of them had identical meaning. These twelve keywords were provided with unique Ids known as ‘Taxonomy Node’.

Hence, the ‘keyword’ database contained two attributes – ‘Taxonomy Node’ and ‘Keyword’. The term ‘keyword’ can be better referred to as key-phrase as a single keyword may contain more than one word. Table 6.5 provides the ‘keyword’ along with those Author Keywords which fall within their perspective. These tables also resemble the level of redundancies in the author keywords

Table 6.5. AUTHOR KEYWORDS CLUBBED INTO ‘KEYWORD’

‘keyword’	Author Keywords
Reversible Logic	logic reversible logic adiabatic logic adiabatic computing reversibility reversible computing conservative logic logic design reversible logic synthesis reversible network reversible lanes reversible data hiding logic synthesis
Power	low power differential power analysis power amplifier low power dissipation power analysis attack power distribution

Miscellaneous	transform silicon atpg microring resonators onoff ratio sidechannel attacks nems fault coverage ctestable thinfilm transistors fault diagnosis test generation plasma oxidation resistive switching serpent cipher bias temperature instability memory cellular automata operational procedures radar inalngan robot skin
Digital Circuits	reversible circuits reversible circuit sequential circuits integrated circuits circuit synthesis synchronous sequential circuit minimal garbage garbage

	garbage outputs testing registers full adder counters minimization lookahead flipflop postsynthesis optimization ring oscillator ro
Gates	toffoli gate fredkin gate toffoli gates quantum gates swap gates gate sinking negative control paritypreserving
Emerging Technologies	optical computing cryptography emerging technologies nanotechnology image encryption nanocomputing serpent cipher image encryption
Design	boolean functions reliability threshold voltage satisfiability decision diagrams energy efficiency fault coverage nearest neighbor

	fault diagnosis test generation failure physical analysis supercapacitors
Quantum Computing	quantum computing cellular automata quantum circuits quantum circuit quantum computation quantumdot quantumdot cellular automata qudits entanglement

6.3 CREATION OF 'INDEX' DATABASE FROM 'KEYWORD' AND 'ABSTRACT' DATABASE

Using the information in the 'keyword' and the 'Abstract' database, the index database has been generated. This database has four attributes – TN (term Id), Keyword, PIDs, Frequency.

TN stands for the unique term id assigned to the final 12 keywords in the 'keyword' database. Keyword also refers to the unique keywords in the 'keyword' database. Now each keyword is taken from the 'keyword' database and searched in the 'Abstract' database for occurrences. All those PIDs are noted where that particular 'keyword' was identified. All these PIDs are stored in the PIDs field of 'index' database. The number of PIDs in the 'index' database is given in the Frequency attribute in the 'index' database.

After generation of the 'index' database, the database is sorted in descending order so that the keyword with the highest frequency is at the top. Table 6.6 provides the keywords with their frequencies.

Table 6.7 provides the PIDs associated with each keyword in the 'index' database. These serve as the ready-reckoner for researchers to get hold of the references according to

their choice of interest.

Table 6.6. KEYWORD TERMS WITH TERM ID AND FREQUENCY

Taxonomy Node ID	Node	Frequency
TN1	Reversible Logic	70
TN2	Digital Circuits	44
TN3	Quantum Comp	29
TN4	Gates	22
TN5	Miscellaneous	22
TN6	Design	21
TN7	Emerging Tech	17
TN8	Algorithm	15
TN9	Power	12
TN10	Complexity	11
TN11	Memory	10
TN12	Security	6

Table 6.7. KEYWORD AND MAPPED PIDS

Taxonomy Node	PIDs
Reversible Logic	PID1;PID2;PID3;PID4;PID5;PID6;PID7;PID9;PID10;PID11;PID12;PID13;PID14;PID15;PID16;PID17;PID18;PID19;PID20;PID21;PID22;PID23;PID24;PID25;PID27;PID28;PID29;PID30;PID33;PID34;PID35;PID36;PID37;PID42;PID43;PID44;PID45;PID46;PID48;PID49;PID50;PID51;PID53;PID54;PID56;PID57;PID58;PID60;PID65;PID66;PID72;PID73;PID76;PID83;PID84;PID88;PID89;PID91;PID92;PID96;PID99;PID100;PID108;PID109;PID38;PID67;PID55;PID40;PID47;PID63
Gates	PID6;PID22;PID36;PID45;PID50;PID53;PID64;PID68;PID3;PID13;PID16;PID33;PID28;PID30;PID31;PID66;PID41;PID70;PID40;PID69;PID34;PID106

Security	PID14;PID15;PID63;PID65;PID74;PID97
Memory	PID13;PID26;PID55;PID65;PID73;PID78;PID79;PID85;PID88;PID94
Digital Circuits	PID3;PID8;PID12;PID16;PID26;PID27;PID31;PID33;PID41;PID51;PID53;PID54;PID67;PID70;PID11;PID28;PID29;PID32;PID34;PID61;PID66;PID10;PID13;PID58;PID85;PID7;PID18;PID23;PID35;PID38;PID37;PID73;PID79;PID87;PID89;PID99;PID9;PID56;PID90;PID14;PID39;PID22;PID69;PID97
Quantum Comp	PID6;PID12;PID22;PID26;PID31;PID34;PID45;PID48;PID61;PID68;PID3;PID13;PID23;PID37;PID38;PID55;PID74;PID78;PID28;PID41;PID69;PID70;PID103;PID36;PID64;PID51;PID67;PID89;PID77
Miscellaneous	PID30;PID39;PID103;PID5;PID104;PID70;PID87;PID40;PID42;PID73;PID15;PID25;PID13;PID36;PID79;PID32;PID74;PID75;PID78;PID76;PID106;PID107
Design	PID14;PID18;PID28;PID38;PID44;PID32;PID37;PID65;PID79;PID99;PID75;PID106;PID48;PID70;PID87;PID104;PID24;PID47;PID69;PID97;PID98
Emerging Tech	PID22;PID48;PID50;PID31;PID65;PID67;PID45;PID69;PID3;PID76;PID23;PID16;PID24;PID14;PID26;PID59;PID78
Algorithm	PID6;PID14;PID16;PID22;PID30;PID31;PID35;PID45;PID53;PID60;PID66;PID67;PID78;PID87;PID89
Power	PID14;PID22;PID25;PID32;PID35;PID55;PID57;PID74;PID15;PID104;PID105;PID98
Complexity	PID13;PID19;PID26;PID32;PID33;PID45;PID50;PID53;PID63;PID74;PID89

6.4 COSINE SIMILARITY FUNCTION

After generation of the ‘index’ database, the matrices are formed for implementation of the Cosine Similarity function. Cosine Similarity function has been used in this chapter to create the taxonomy.

Basically the taxonomy has been formed taking into account the distance between two Author Keyword, or more precisely, between two taxonomy nodes. Cosine Similarity function provides similarity between two terms. Another similarity measuring function is the Google Similarity Distance [**90**]. The Google Similarity Distance finds the similarity between the hit counts between two counts when searched via a search engine. Since, the process used in this chapter is offline, hence Cosine Similarity distance was chosen. Also, Cosine Similarity function forms the basis for other taxonomy generation algorithms. Hence the findings in this chapter can be utilized to generate more robust taxonomies in future if the present taxonomy is generated using Cosine Similarity function. Equation (2) presents the Cosine Similarity function.

$$Cosine\ Similarity = \frac{n_{x,y}}{\sqrt{n_x}\sqrt{n_y}} \quad (2)$$

$Where\ n_{x,y} = Number\ of\ articles\ containing\ both\ term\ 'x'\ and\ term\ 'y',$

$n_x = Number\ of\ articles\ containing\ only\ term\ 'x', and$

$n_y = Number\ of\ articles\ containing\ only\ term\ 'y'.$

If the two nodes are similar, then the angle between them tends to 0°. Therefore $Cos\ 0° = 1$, and if the distance between the two nodes are very high, then the two nodes can be said to be orthogonal to each other. In that case, $Cos\ 90° = 0$. Hence the Cosine Similarity value for each group of nodes lies between 0 and 1. If the value nears 0 then the two nodes are highly dissimilar, and is the value approaches 1, the nodes are highly identical and bear a string relationship between them.

6.4.1 GENERATION OF TERM CO-OCCURRENCE MATRIX

The Term Co-occurrence Matrix contains the count of occurrences of two keywords taken together. Hence, this matrix generates the numerator for Equation (2). Table 6.8 provides the Term Co-occurrence Matrix.

After generation of the Term Co-occurrence Matrix, next the denominator of Equation (2) is generated. This matrix contains the square root values for the frequencies of each node multiplied by the square root of the second node under consideration. This ma-

trix is shown in Table 6.9.

After generation of the two matrices in Table 6.8 and 6.9, the Cosine Similarity Matrix is generated and is shown in Table 6.10.

Table 6.8. TERM CO-OCCURRENCE MATRIX

R/C	TN1	TN2	TN3	TN4	TN5	TN6	TN7	TN8	TN9	TN10	TN11	TN12
TN1	-	30	17	15	10	11	12	12	7	8	5	4
TN2	30	-	18	13	7	12	9	10	4	6	5	2
TN3	17	18	-	14	6	6	10	7	3	5	4	1
TN4	15	13	14	-	6	4	7	8	1	5	1	0
TN5	10	7	6	6	-	7	2	3	5	3	4	2
TN6	11	12	6	4	7	-	5	2	4	1	2	3
TN7	12	9	10	7	2	5	-	7	2	3	3	2
TN8	12	10	7	8	3	2	7	-	3	3	1	1
TN9	7	4	3	1	5	4	2	3	-	2	1	3
TN10	8	6	5	5	3	1	3	3	2	-	2	2
TN11	5	5	4	1	4	2	3	1	1	2	-	1
TN12	4	2	1	0	2	3	2	1	3	2	1	-

Table 6.9. DENOMINATOR VALUES FOR COSINE SIMILARITY FUNCTION

R/C	TN1	TN2	TN3	TN4	TN5	TN6	TN7	TN8	TN9	TN10	TN11	TN12
TN1	-	55.5	45.06	39.24	39.24	38.34	34.5	32.4	28.98	27.75	26.46	20.49
TN2	55.5	-	35.72	31.11	31.11	30.4	27.35	25.69	22.98	22	20.98	16.25
TN3	45.06	35.72	-	25.26	25.26	24.68	22.2	20.86	18.65	17.86	17.03	13.19
TN4	39.24	31.11	25.26	-	22	21.49	19.34	18.17	16.25	15.56	14.83	11.49
TN5	39.24	31.11	25.26	22	-	21.49	19.34	18.17	16.25	15.56	14.83	11.49
TN6	38.34	30.4	24.68	21.49	21.49	-	18.89	17.75	15.87	15.2	14.49	11.22
TN7	34.5	27.35	22.2	19.34	19.34	18.89	-	15.97	14.28	13.67	13.04	10.1
TN8	32.4	25.69	20.86	18.17	18.17	17.75	15.97	-	13.42	12.85	12.25	9.49
TN9	28.98	22.98	18.65	16.25	16.25	15.87	14.28	13.42	-	11.49	10.95	8.49
TN10	27.75	22	17.86	15.56	15.56	15.2	13.67	12.85	11.49	-	10.49	8.12
TN11	26.46	20.98	17.03	14.83	14.83	14.49	13.04	12.25	10.95	10.49	-	7.75
TN12	20.49	16.25	13.19	11.49	11.49	11.22	10.1	9.49	8.49	8.12	7.75	-

Table 6.10. COSINE SIMILARITY MATRIX

R/C	TN1	TN2	TN3	TN4	TN5	TN6	TN7	TN8	TN9	TN10	TN11	TN12
TN1	-	0.54	0.38	0.38	0.25	0.29	0.35	0.37	0.24	0.29	0.19	0.2
TN2	0.54	-	0.5	0.42	0.23	0.39	0.33	0.39	0.17	0.27	0.24	0.12
TN3	0.38	0.5	-	0.55	0.24	0.24	0.45	0.34	0.16	0.28	0.23	0.08
TN4	0.38	0.42	0.55	-	0.27	0.19	0.36	0.44	0.06	0.32	0.07	0
TN5	0.25	0.23	0.24	0.27	-	0.33	0.1	0.17	0.31	0.19	0.27	0.17
TN6	0.29	0.39	0.24	0.19	0.33	-	0.26	0.11	0.25	0.07	0.14	0.27
TN7	0.35	0.33	0.45	0.36	0.1	0.26	-	0.44	0.14	0.22	0.23	0.2
TN8	0.37	0.39	0.34	0.44	0.17	0.11	0.44	-	0.22	0.23	0.08	0.11
TN9	0.24	0.17	0.16	0.06	0.31	0.25	0.14	0.22	-	0.17	0.09	0.35
TN10	0.29	0.27	0.28	0.32	0.19	0.07	0.22	0.23	0.17	-	0.19	0.25
TN11	0.19	0.24	0.23	0.07	0.27	0.14	0.23	0.08	0.09	0.19	-	0.13
TN12	0.2	0.12	0.08	0	0.17	0.27	0.2	0.11	0.35	0.25	0.13	-

6.5 TAXONOMY GENERATION

Table 6.10 serves to generate the taxonomy. The process is described in the following pseudo code.

Begin:

$L1$ $for\ \boldsymbol{i} \in [\mathbf{1}, \mathbf{12}]$

$L2$ $for\ \boldsymbol{j} \in [\mathbf{1}, \mathbf{12}]$

$L3$ $scan\ \boldsymbol{R[i]C[j]}\ for\ the\ highest\ value\ \forall\ \boldsymbol{i \neq x \cap j \neq y}$

$L4$ $\boldsymbol{x = i\ for\ highest\ R[i]C[j]}$

$L5$ $\boldsymbol{y = j\ for\ highest\ R[i]C[j]}$

$L6$ $if\ (\boldsymbol{TN[y]is\ not\ related\ to\ TN[x]previously})$

$L7$ $\boldsymbol{TN[x]is\ the\ child\ of\ TN[y]}$

$L8$ $else$

$L9$ $goto\ L2$

End

In other words, each row is scanned from left to right. Initially since TN1 has the highest frequency, hence it becomes the root node. Starting from TN2, each row is scanned

horizontally and the column which gives the highest value is recorded. The TN corresponding to this column becomes the parent node for the initial node for which scanning was done. Post scanning, the child parent relationship is given in Table 6.11.

Table 6.11. PARENT CHILD RELATIONSHIP

Parent	Child	Parent	Child
TN1	TN2	TN7	Leaf Node
TN2	TN6, TN11	TN8	TN4
TN3	TN7	TN9	TN12
TN4	TN3, TN10	TN10	Leaf Node
TN5	Leaf Node	TN11	Leaf Node
TN6	TN5, TN9	TN12	Leaf Node

From Table 6.11, the taxonomy is generated. Figure 6.2 presents the taxonomy.

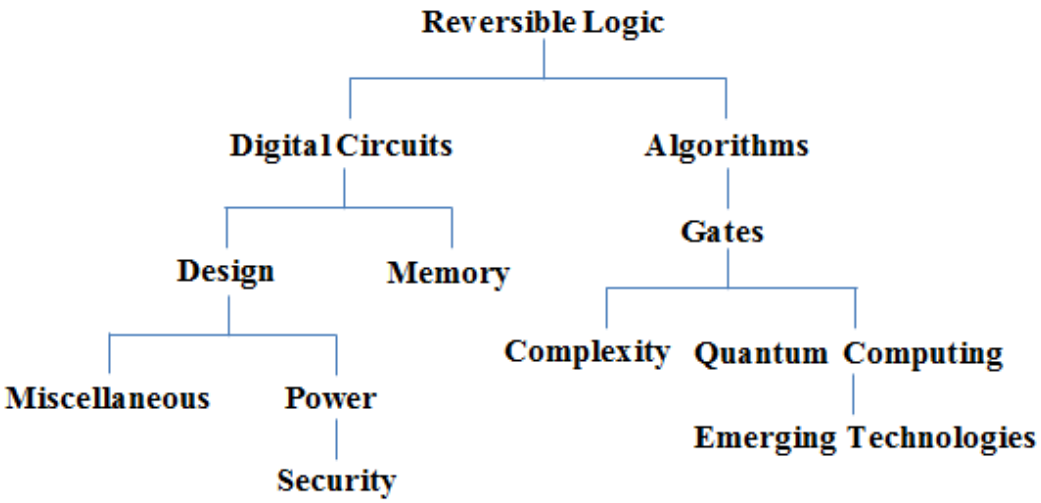


Figure 6.2. Taxonomy for research on Reversible Logic

6.6 CONCLUSION

This chapter has proposed a taxonomy for research on Reversible Logic. It has stemmed out of literature conducted on the huge number of articles. The taxonomy is believed to

help amateur researchers as well as existing researchers who wish to point out a certain reference out of the massive collection of research articles.

The next step in this taxonomy would be to include more online databases into the vocabulary. Also further algorithms for taxonomy generation can be explored for more robust taxonomy in future.

Chapter 7

Conclusion

7 CONCLUSION

This thesis is an attempt to address green computing through concept of reversible logic. With rising atmospheric temperature, it has become highly important to reduce the temperature. Since, modern CPUs dissipate enormous amount of heat, hence heat reduction has become a topic of interest among digital architects. Information lossless computation is possible only if the operations are reversible. Also conservation of intermediate information restrains heat dissipation. Also with advancing technology, the demand has driven higher clocking processors. But the present speed of processor vary within GHz. The need is of THz. To address these two issues, this chapter proposes reversible architectures which can be implemented using zero heat dissipation and which operates in the range of THz frequencies.

7.1 CHAPTER 2 SUMMARY

Before exploring, it is important to know the existing facts about research domains. It is this very issue that Chapter 2 addresses. It presents a brief analysis of existing reversible gates having four inputs. The chapter also proposes Toffoli gate realizations for all the existing reversible gates (four variables) and also presents the QCA implementations for the same. The chapter summarizes and provides a benchmark for future reversible gate proposals.

7.2 CHAPTER 3 SUMMARY

Having provided the benchmark in Chapter 2, Chapter 3 proposes a new reversible gate having four inputs and four outputs. It has been shown that the proposed gate fares better than most of the other literary proposals. The single standalone proposed gate can also realize a full adder and a full subtractor.

7.3 CHAPTER 4 SUMMARY

Chapter 4 provides a synthesis algorithm for synthesizing reversible gates. The al-

gorithm is a library based algorithm which matches Control Lime Set templates and does the synthesis. Comparative analysis has proven the algorithm to be better than an existing counterpart.

7.4 CHAPTER 5 SUMMARY

Chapter 5 proposes a decimal counter based on reversible Toffoli gates. The proposed counter is synchronous. Further exploration in the topic of the chapter can be to realize the asynchronous version. The decimal counter proposal is believed to be the very first proposal in the domain.

7.5 CHAPTER 6 SUMMARY

Chapter 6 provides a taxonomy based on reversible logic research. The taxonomy is supposed to aid amateur researchers and also those experts who wish to identify very accurately a certain publication which falls in their domain of interest. The taxonomy has been created using Author Keywords in published literature. The taxonomy has been created using the Cosine Similarity function.

7.6 FUTURE EXTENSION PROSPECTS

The library based algorithm utilizes only Toffoli Gates. It can be further extended to accommodate Toffoli-Fredkin combinations. The algorithm presently handles four variables. It can be redesigned to handle any number of variables. The thesis can be further extended to realize asynchronous counters and also implement the proposed counter using QCA. The taxonomy can be further generated using a more robust algorithm. Also more online databases can be used to make the taxonomy more comprehensive. In the present case, only journal articles have been used. In future, conference publications may also be included to generate such a taxonomy.

BIBLIOGRAPHY

TAXONOMY BIBLIOGRAPHY

1. Joonho Lim, Dong-Gyu Kim and Soo-Ik Chae, "A 16-bit carry-lookahead adder using reversible energy recovery logic for ultra-low-energy systems," in IEEE Journal of Solid-State Circuits, vol. 34, no. 6, pp. 898-903, June 1999.
2. D. P. Vasudevan, P. K. Lala, Jia Di and J. P. Parkerson, "Reversible-logic design with online testability," in IEEE Transactions on Instrumentation and Measurement, vol. 55, no. 2, pp. 406-414, April 2006.
3. H. Thapliyal and N. Ranganathan, "Reversible Logic-Based Concurrently Testable Latches for Molecular QCA," in IEEE Transactions on Nanotechnology, vol. 9, no. 1, pp. 62-69, Jan. 2010.
4. Yibin Ye and K. Roy, "Energy recovery circuits using reversible and partially reversible logic," in IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, vol. 43, no. 9, pp. 769-778, Sept. 1996.
5. P. Sethi and S. Roy, "All-Optical Ultrafast Switching in 2 × 2 Silicon Microring Resonators and its Application to Reconfigurable DEMUX/MUX and Reversible Logic Gates," in Journal of Lightwave Technology, vol. 32, no. 12, pp. 2173-2180, 15 June15, 2014.
6. P. Gupta, A. Agrawal and N. K. Jha, "An Algorithm for Synthesis of Reversible Logic Circuits," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 25, no. 11, pp. 2317-2330, Nov. 2006.
7. D. Maslov and G. W. Dueck, "Reversible cascades with minimal garbage," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 23, no. 11, pp. 1497-1509, Nov. 2004.

8. A. Zulehner and R. Wille, "One-Pass Design of Reversible Circuits: Combining Embedding and Synthesis for Reversible Logic," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 37, no. 5, pp. 996-1008, May 2018.

9. Joonho Lim, Dong-Gyu Kim and Soo-Ik Chae, "nMOS reversible energy recovery logic for ultra-low-energy applications," in IEEE Journal of Solid-State Circuits, vol. 35, no. 6, pp. 865-875, June 2000.

10. M. H. A. Khan, "Design of Reversible Synchronous Sequential Circuits Using Pseudo Reed-Muller Expressions," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 22, no. 11, pp. 2278-2286, Nov. 2014.

11. S. N. Mahammad and K. Veezhinathan, "Constructing Online Testable Circuits Using Reversible Logic," in IEEE Transactions on Instrumentation and Measurement, vol. 59, no. 1, pp. 101-109, Jan. 2010.

12. V. K. Semenov, G. V. Danilov and D. V. Averin, "Classical and Quantum Operation Modes of the Reversible Josephson-Junction Logic Circuits," in IEEE Transactions on Applied Superconductivity, vol. 17, no. 2, pp. 455-461, June 2007.

13. H. Thapliyal, N. Ranganathan and S. Kotiyal, "Design of Testable Reversible Sequential Circuits," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 21, no. 7, pp. 1201-1209, July 2013.

14. M. Morrison and N. Ranganathan, "Synthesis of Dual-Rail Adiabatic Logic for Low Power Security Applications," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 33, no. 7, pp. 975-988, July 2014.

15. M. A. Morrison, N. Ranganathan and J. Ligatti, "Design of Adiabatic Dynamic Differential Logic for DPA-Resistant Secure Integrated Circuits," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 23, no. 8, pp. 1381-1389, Aug. 2015.

16. 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.

17. J. E. Rice, "An Introduction to Reversible Latches," in The Computer Journal, vol. 51, no. 6, pp. 700-709, Nov. 2008.

18. D. Maslov, "Efficient reversible and quantum implementations of symmetric Boolean functions," in IEE Proceedings - Circuits, Devices and Systems, vol. 153, no. 5, pp. 467-472, October 2006.

19. T. Chattopadhyay, "Optical logic circuits using double controlled logic gate," in IET Optoelectronics, vol. 7, no. 5, pp. 99-109, October 2013.

20. Joonho Lim, Kipaek Kwon and Soo-Ik Chae, "Reversible energy recovery logic circuit without non-adiabatic energy loss," in Electronics Letters, vol. 34, no. 4, pp. 344-346, 19 Feb. 1998.

21. Kipaek Kwon and Soo-Ik Chae, "Simple reversible energy recovery logic using NMOS switch networks with cross-coupled PMOS pair," in Electronics Letters, vol. 34, no. 23, pp. 2215-2216, 12 Nov. 1998.

22. D. Maslov, G. W. Dueck and D. M. Miller, "Synthesis of Fredkin-Toffoli reversible networks," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 13, no. 6, pp. 765-769, June 2005.

23. L. Touil and B. Ouni, "Design of hardware RGB to HMMD converter based on reversible logic," in IET Image Processing, vol. 11, no. 8, pp. 646-655, 8 2017.

24. J. F. Chaves, M. A. Ribeiro, F. S. Torres and O. P. V. Neto, "Designing Partially Reversible Field-Coupled Nanocomputing Circuits," in IEEE Transactions on Nanotechnology, vol. 18, pp. 589-597, 2019.

25. S. Houri, G. Billiot, M. Belleville, A. Valentian and H. Fanet, "Limits of CMOS Technology and Interest of NEMS Relays for Adiabatic Logic Applications," in

IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 62, no. 6, pp. 1546-1554, June 2015.

26. G. Yang, X. Song, W. N. N. Hung and M. A. Perkowski, "Bi-Directional Synthesis of 4-Bit Reversible Circuits," in The Computer Journal, vol. 51, no. 2, pp. 207-215, March 2008.

27. H. M. Gaur and A. K. Singh, "Design of reversible circuits with high testability," in Electronics Letters, vol. 52, no. 13, pp. 1102-1104, 23 6 2016.

28. W. N. N. Hung, Xiaoyu Song, Guowu Yang, Jin Yang and M. Perkowski, "Optimal synthesis of multiple output Boolean functions using a set of quantum gates by symbolic reachability analysis," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 25, no. 9, pp. 1652-1663, Sept. 2006.

29. J. Ren and V. K. Semenov, "Progress With Physically and Logically Reversible Superconducting Digital Circuits," in IEEE Transactions on Applied Superconductivity, vol. 21, no. 3, pp. 780-786, June 2011.

30. D. Maslov, G. W. Dueck and D. M. Miller, "Toffoli network synthesis with templates," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 24, no. 6, pp. 807-817, June 2005.

31. V. V. Shende, A. K. Prasad, I. L. Markov and J. P. Hayes, "Synthesis of reversible logic circuits," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 22, no. 6, pp. 710-722, June 2003.

32. A. N. Nagamani, S. N. Anuktha, N. Nanditha and V. K. Agrawal, "A Genetic Algorithm-Based Heuristic Method for Test Set Generation in Reversible Circuits," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 37, no. 2, pp. 324-336, Feb. 2018.

33. T. Chattopadhyay, "All-Optical Modified Fredkin Gate," in IEEE Journal of Selected Topics in Quantum Electronics, vol. 18, no. 2, pp. 585-592, March-April 2012.

34. K. Datta, I. Sengupta and H. Rahaman, "A Post-Synthesis Optimization Technique for Reversible Circuits Exploiting Negative Control Lines," in IEEE Transactions on Computers, vol. 64, no. 4, pp. 1208-1214, 1 April 2015.

35. H. M. Hasan Babu, N. Saleheen, L. Jamal, S. M. Sarwar and T. Sasao, "Approach to design a compact reversible low power binary comparator," in IET Computers & Digital Techniques, vol. 8, no. 3, pp. 129 -139, May 2014.

36. Y. Chou, I. Tsai and S. Kuo, "Quantum Boolean Circuits are 1-Testable," in IEEE Transactions on Nanotechnology, vol. 7, no. 4, pp. 484-492, July 2008.

37. B. Debnath, J. C. Das and D. De, "Reversible logic-based image steganography using quantum dot cellular automata for secure nanocommunication," in IET Circuits, Devices & Systems, vol. 11, no. 1, pp. 58-67, 1 2017.

38. A. Roohi, R. Zand, S. Angizi and R. F. DeMara, "A Parity-Preserving Reversible QCA Gate with Self-Checking Cascadable Resiliency," in IEEE Transactions on Emerging Topics in Computing, vol. 6, no. 4, pp. 450-459, 1 Oct.-Dec. 2018.

39. B. Schaeffer, "Product Transformation and Heuristic EXOR-AND-OR Logic Synthesis of Incompletely Specified Functions," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 36, no. 11, pp. 1831-1841, Nov. 2017.

40. T. Chattopadhyay, "All-Optical Reversible Network Design Using Microring Resonators," in IEEE Journal of Quantum Electronics, vol. 51, no. 4, pp. 1-8, April 2015, Art no. 6500108.

41. D. Maslov and M. Saeedi, "Reversible Circuit Optimization Via Leaving the Boolean Domain," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 30, no. 6, pp. 806-816, June 2011.

42. S. Nakaharai et al., "Electrostatically Reversible Polarity of Dual-Gated Graphene Transistors," in IEEE Transactions on Nanotechnology, vol. 13, no. 6, pp. 1039-1043, Nov. 2014.

43. E. Mizraji, "Vector Logic: A Natural Algebraic Representation of the Fundamental Logical Gates," in Journal of Logic and Computation, vol. 18, no. 1, pp. 97-121, Feb. 2008.

44. C. Bandyopadhyay, R. Das, A. Chattopadhyay and H. Rahaman, "Design and synthesis of improved reversible circuits using AIG- and MIG-based graph data structures," in IET Computers & Digital Techniques, vol. 13, no. 1, pp. 38-48, 1 2019.

45. C. A. Lungarzo, "Many-valued Logics in Classical and Quantum Gates," in Logic Journal of the IGPL, vol. 13, no. 1, pp. 127-138, Jan. 2005.

46. Suhwan Kim, C. H. Ziesler and M. C. Papaefthymiou, "Charge-recovery computing on silicon," in IEEE Transactions on Computers, vol. 54, no. 6, pp. 651-659, June 2005.

47. J. R. D. Frejo, I. Papamichail, M. Papageorgiou and E. F. Camacho, "Macroscopic Modeling and Control of Reversible Lanes on Freeways," in IEEE Transactions on Intelligent Transportation Systems, vol. 17, no. 4, pp. 948-959, April 2016.

48. D. Große, R. Wille, G. W. Dueck and R. Drechsler, "Exact Multiple-Control Toffoli Network Synthesis With SAT Techniques," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 28, no. 5, pp. 703-715, May 2009.

49. J. Lim, D. -. Kim and S. -. Chae, "Reduction in energy consumption by bootstrapped n MOS switches in reversible adiabatic CMOS circuits," in IEE Proceedings - Circuits, Devices and Systems, vol. 146, no. 6, pp. 327-333, Dec. 1999.

50. S. K. Garai, "Novel method of designing all optical frequency-encoded Fredkin and Toffoli logic gates using semiconductor optical amplifiers," in IET Optoelectronics, vol. 5, no. 6, pp. 247-254, December 2011.

51. V. K. Semenov, G. V. Danilov and D. V. Averin, "Negative-inductance SQUID as the basic element of reversible Josephson-junction circuits," in IEEE Transactions on Applied Superconductivity, vol. 13, no. 2, pp. 938-943, June 2003.

52. D. Maslov and D. M. Miller, "Comparison of the cost metrics through investigation of the relation between optimal NCV and optimal NCT three-qubit reversible circuits," in IET Computers & Digital Techniques, vol. 1, no. 2, pp. 98-104, March 2007.

53. G. W. Dueck, "Challenges and advances in Toffoli network optimisation," in IET Computers & Digital Techniques, vol. 8, no. 4, pp. 172-177, July 2014.

54. C. Lu, S. Wang and S. Kuo, "An Extended XQDD Representation for Multiple-Valued Quantum Logic," in IEEE Transactions on Computers, vol. 60, no. 10, pp. 1377-1389, Oct. 2011.

55. M. Ottavi et al., "Partially Reversible Pipelined QCA Circuits: Combining Low Power With High Throughput," in IEEE Transactions on Nanotechnology, vol. 10, no. 6, pp. 1383-1393, Nov. 2011.

56. O. Mukhanov, V. Semenov and K. Likharev, "Ultimate performance of the RSFQ logic circuits," in IEEE Transactions on Magnetics, vol. 23, no. 2, pp. 759-762, March 1987.

57. A. Sarker, H. M. Hasan Babu and S. M. M. Rashid, "Design of a DNA-based reversible arithmetic and logic unit," in IET Nanobiotechnology, vol. 9, no. 4, pp. 226-238, 8 2015.

58. V. K. Kaplunenko, M. I. Khabipov, D. Y. Khokhlov, A. F. Kirichenko, V. P. Koshelets and S. A. Kovtonyuk, "Experimental implementation of SFQ NDRO cells

and 8-bit ADC," in IEEE Transactions on Applied Superconductivity, vol. 3, no. 1, pp. 2662-2665, March 1993.

59. X. Zhang, J. Long, Z. Wang and H. Cheng, "Lossless and Reversible Data Hiding in Encrypted Images With Public-Key Cryptography," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 26, no. 9, pp. 1622-1631, Sept. 2016.

60. Euisu Park, D. M. Tilbury and P. P. Khargonekar, "A modeling and analysis methodology for modular logic controllers of machining systems using Petri net formalism," in IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 31, no. 2, pp. 168-188, May 2001.

61. C. Bandyopadhyay, S. Parekh and H. Rahaman, "Improved circuit synthesis approach for exclusive-sum-of-product-based reversible circuits," in IET Computers & Digital Techniques, vol. 12, no. 4, pp. 167-175, 7 2018.

62. B. Aiazzi, L. Alparone and S. Baronti, "Fuzzy logic-based matching pursuits for lossless predictive coding of still images," in IEEE Transactions on Fuzzy Systems, vol. 10, no. 4, pp. 473-483, Aug. 2002.

63. S. Xiang and X. Luo, "Reversible Data Hiding in Homomorphic Encrypted Domain by Mirroring Ciphertext Group," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 28, no. 11, pp. 3099-3110, Nov. 2018.

64. D. Maslov and G. W. Dueck, "Improved quantum cost for n-bit Toffoli gates," in Electronics Letters, vol. 39, no. 25, pp. 1790-1791, 11 Dec. 2003.

65. B. Liu and B. Wang, "Reconfiguration-Based VLSI Design for Security," in IEEE Journal on Emerging and Selected Topics in Circuits and Systems, vol. 5, no. 1, pp. 98-108, March 2015.

66. G. Yang, X. Song, M. A. Perkowski, W. N. N. Hung, J. Biamonte and Z. Tang, "Four-level realisation of 3-qubit reversible functions," in IET Computers & Digital Techniques, vol. 1, no. 4, pp. 382-388, July 2007.

67. K. N. Patel, J. P. Hayes and I. L. Markov, "Fault testing for reversible circuits," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 23, no. 8, pp. 1220-1230, Aug. 2004.

68. C. Moraga, "Aspects of Reversible and Quantum Computing in ap-Valued Domain," in IEEE Journal on Emerging and Selected Topics in Circuits and Systems, vol. 6, no. 1, pp. 44-52, March 2016.

69. A. Kole, K. Datta and I. Sengupta, "A Heuristic for Linear Nearest Neighbor Realization of Quantum Circuits by SWAP Gate Insertion Using$N$-Gate Lookahead," in IEEE Journal on Emerging and Selected Topics in Circuits and Systems, vol. 6, no. 1, pp. 62-72, March 2016.

70. D. Bera, "Detection and Diagnosis of Single Faults in Quantum Circuits," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 37, no. 3, pp. 587-600, March 2018.

71. S. Fukushima and T. Kurokawa, "Optical parallel processor for binary images with cascaded bipolar-operational spatial light modulators," in IEEE Photonics Technology Letters, vol. 3, no. 7, pp. 682-684, July 1991.

72. B. Aiazzi, P. Alba, L. Alparone and S. Baronti, "Lossless compression of multi/hyper-spectral imagery based on a 3-D fuzzy prediction," in IEEE Transactions on Geoscience and Remote Sensing, vol. 37, no. 5, pp. 2287-2294, Sept. 1999.

73. H. B. Lv et al., "Resistive Memory Switching of $\hbox{Cu}_{x}\hbox{O}$ Films for a Nonvolatile Memory Application," in IEEE Electron Device Letters, vol. 29, no. 4, pp. 309-311, April 2008.

74. W. Liu, S. Srivastava, L. Lu, M. O'Neill and E. E. Swartzlander, "Are QCA cryptographic circuits resistant to power analysis attack?," in IEEE Transactions on Nanotechnology, vol. 11, no. 6, pp. 1239-1251, Nov. 2012.

75. A. Guo and J. A. del Alamo, "Unified Mechanism for Positive- and Negative-Bias Temperature Instability in GaN MOSFETs," in IEEE Transactions on Electron Devices, vol. 64, no. 5, pp. 2142-2147, May 2017.

76. G. Parise, E. Hesla and R. M. Rifaat, "Genetic Code of Electrical Operational Procedures: Lock-Out Systems, Simulators, and Training," in IEEE Transactions on Industry Applications, vol. 46, no. 2, pp. 569-574, March-april 2010.

77. P. Niemann, R. Wille, D. M. Miller, M. A. Thornton and R. Drechsler, "QMDDs: Efficient Quantum Function Representation and Manipulation," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 35, no. 1, pp. 86-99, Jan. 2016.

78. A. M. del Rey, G. R. Sánchez and A. de la Villa Cuenca, "A protocol to encrypt digital images using chaotic maps and memory cellular automata," in Logic Journal of the IGPL, vol. 23, no. 3, pp. 485-494, June 2015.

79. F. Adriyanto, C. Yang, T. Yang, C. Wei and Y. Wang, "Solution-Processed Barium Zirconate Titanate for Pentacene-Based Thin-Film Transistor and Memory," in IEEE Electron Device Letters, vol. 34, no. 10, pp. 1241-1243, Oct. 2013.

80. S. Banerjee, T. P. Chow and R. J. Gutmann, "1300-V 6H-SiC lateral MOSFETs with two RESURF zones," in IEEE Electron Device Letters, vol. 23, no. 10, pp. 624-626, Oct. 2002.

81. D. L. Mcmurtrie and J. P. Ward, "Triggering in the reversible half-wave magnetic amplifier," in Transactions of the American Institute of Electrical Engineers, Part I: Communication and Electronics, vol. 77, no. 5, pp. 598-605, Nov. 1958.

82. D. L. Mcmurtrie, "Magnetic amplifier circuits - A classification of half-wave and full-wave nonreversible and reversible self-saturating circuits," in Transactions of the American Institute of Electrical Engineers, Part I: Communication and Electronics, vol. 78, no. 6, pp. 739-751, Jan. 1960.

83. D. L. Duff and A. Ludbrook, "Reversing Thyristor Armature Dual Converter with Logic Crossover Control," in IEEE Transactions on Industry and General Applications, vol. IGA-1, no. 3, pp. 216-222, May 1965.

84. G. Parise, E. Hesla and R. M. Rifaat, "Architecture Impact on Integrity of Electrical Installations: Cut&Tie Rule, Ring Configuration, Floating Node," in IEEE Transactions on Industry Applications, vol. 45, no. 5, pp. 1903-1909, Sept.-oct. 2009.

85. D. Mange, M. Sipper, A. Stauffer and G. Tempesti, "Toward robust integrated circuits: The embryonics approach," in Proceedings of the IEEE, vol. 88, no. 4, pp. 516-543, April 2000.

86. H. F. Wong et al., "Gate-Controlled Transport Properties in Dilute Magnetic Semiconductor (Zn, Mn)O Thin Films," in IEEE Transactions on Magnetics, vol. 54, no. 11, pp. 1-4, Nov. 2018, Art no. 4200104.

87. P. -. Yeh, B. -. Ye, S. -. Kuo and I. -. Chen, "Effective design-for-testability techniques for H.264 all-binary integer motion estimation," in IET Circuits, Devices & Systems, vol. 4, no. 5, pp. 403-413, September 2010.

88. K. Karda, C. Mouli, S. Ramanathan and M. A. Alam, "A Self-Consistent, Semiclassical Electrothermal Modeling Framework for Mott Devices," in IEEE Transactions on Electron Devices, vol. 65, no. 5, pp. 1672-1678, May 2018.

89. T. Hey, "Quantum computing: an introduction," in Computing & Control Engineering Journal, vol. 10, no. 3, pp. 105-112, June 1999.

90. N. Bonelli, C. Callegari and G. Procissi, "A Probabilistic Counting Framework for Distributed Measurements," in IEEE Access, vol. 7, pp. 22644-22659, 2019.

91. W. W. Gaertner, "Adaptable Large-Scale-Integrated Military and Space Systems," in IEEE Transactions on Aerospace and Electronic Systems, vol. AES-2, no. 6, pp. 135-141, Nov. 1966.

92. L. Driessen, "On an infinite series of[4n, 2n]binary codes (Corresp.)," in IEEE Transactions on Information Theory, vol. 30, no. 2, pp. 392-395, March 1984.

93. R. Young, L. Su, M. Ieong and S. Kapur, "A possible mechanism for reconciling large gate-drain overlap capacitance with a small difference between polysilicon gate length and effective channel length in an advanced technology PFET," in IEEE Electron Device Letters, vol. 19, no. 7, pp. 234-236, July 1998.

94. M. Michael and W. C. Lin, "A nonvolatile memory circuit-a novel approach," in IEEE Journal of Solid-State Circuits, vol. 4, no. 5, pp. 288-291, Oct. 1969.

95. N. Raghavan et al., "Very Low Reset Current for an RRAM Device Achieved in the Oxygen-Vacancy-Controlled Regime," in IEEE Electron Device Letters, vol. 32, no. 6, pp. 716-718, June 2011.

96. G. F. Quittner, "Quo Vadis Automation?," in IEEE Transactions on Industrial Electronics and Control Instrumentation, vol. IECI-21, no. 4, pp. 215-221, Nov. 1974.

97. A. Maiti and P. Schaumont, "The Impact of Aging on a Physical Unclonable Function," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 22, no. 9, pp. 1854-1864, Sept. 2014.

98. H. Zhang, F. Mollet, C. Saudemont and B. Robyns, "Experimental Validation of Energy Storage System Management Strategies for a Local DC Distribution System of More Electric Aircraft," in IEEE Transactions on Industrial Electronics, vol. 57, no. 12, pp. 3905-3916, Dec. 2010.

99. D. Loev, W. Miehle, J. Paivinen and J. Wylen, "Magnetic Core Circuits for Digital Data-Processing Systems," in Proceedings of the IRE, vol. 44, no. 2, pp. 154-162, Feb. 1956.

100. S. Cova, D. Dotti and S. M. Tenconi, "Automated Digital Measurements of CV Curves and Derivatives, and of Capacitance Transients for Semiconductor Evalua-

tion," in IEEE Transactions on Instrumentation and Measurement, vol. 22, no. 4, pp. 347-355, Dec. 1973.

101. J. A. Baldwin, "Flux Reversal in Three-Rung Laddics," in IRE Transactions on Electronic Computers, vol. EC-11, no. 5, pp. 664-676, Oct. 1962.

102. E. X. Zhang et al., "Bias-Temperature Instabilities in 4H-SiC Metal–Oxide–Semiconductor Capacitors," in IEEE Transactions on Device and Materials Reliability, vol. 12, no. 2, pp. 391-398, June 2012.

103. H. J. García and I. L. Markov, "Simulation of Quantum Circuits via Stabilizer Frames," in IEEE Transactions on Computers, vol. 64, no. 8, pp. 2323-2336, 1 Aug. 2015.

104. F. Temcamani, J. Fonder, O. Latry and C. Duperrier, "Electrical and Physical Analysis of Thermal Degradations of AlGaN/GaN HEMT Under Radar-Type Operating Life," in IEEE Transactions on Microwave Theory and Techniques, vol. 64, no. 3, pp. 756-766, March 2016.

105. A. Lavi and R. W. Roberts, "Static amplidyne: A new high-capacity push - Pull power amplifier I - Resistive-inductive loads," in Transactions of the American Institute of Electrical Engineers, Part I: Communication and Electronics, vol. 82, no. 4, pp. 503-508, Sept. 1963.

106. Y. Wu and J. A. del Alamo, "Electrical Degradation of InAlN/GaN HEMTs Operating Under ON Conditions," in IEEE Transactions on Electron Devices, vol. 63, no. 9, pp. 3487-3492, Sept. 2016.

107. P. Cosseddu, A. Loi, L. Basiricò, S. Lai and A. Bonfiglio, "Organic bendable and stretchable field effect devices for sensing applications," SENSORS, 2012 IEEE, Taipei, 2012, pp. 1-4.

108. A. K. Chattopadhyay, "An Adjustable-Speed Induction Motor Drive with a Cycloconverter-Type Thyristor-Commutator in the Rotor," in IEEE Transactions on Industry Applications, vol. IA-14, no. 2, pp. 116-122, March 1978.

109. V. L. Newhouse and N. S. Prywes, "High-Speed Shift Registers Using One Core Per Bit," in IRE Transactions on Electronic Computers, vol. EC-5, no. 3, pp. 114-120, Sept. 1956.

BIBLIOGRAPHY

[1] Rolf Landauer, "Irreversibility and Heat Generation in the Computing Process," *IBM Journal of Research and Development*, vol. 5, no. 3, pp. 183-191, July 1961.

[2] Antoine Bérut et al., "Experimental verification of Landauer's principle linking information and thermodynamics," *Nature*, vol. 483, no. 7388, pp. 187-189, March 2012.

[3] V V Zhirnov, R K Cavin, J A Hutchby, and G I Bourianoff, "Limits to Binary logic Switch Scaling - A Gedaken Model," *Proceedings of IEEE*, vol. 91, no. 11, pp. 1934-1939, November 2003.

[4] R Compaño, L Molenkamp, and D J Paul, "Technology Roadmap for Nanoelectronics," in *European Commission IST programme Future and Emerging Technologies*, 1999.

[5] C H Bennett, "Logical Reversibility of Computation," *IBM Journal of Research and Development*, vol. 17, no. 6, pp. 525 - 532, Nov 1973.

[6] A-M Visan, K Arya, G Cooperman, and T Denniston, "URDB: a universal reversible debugger based on decomposing debugging histories," in *PLOS '11 Proceedings of the 6th Workshop on Programming Languages and Operating Systems*, NY, USA, 2009.

[7] J C Das and D De, "Optimized design of reversible gates in quantum dot-cellular automata: a review," *Reviews in Theoretical Science*, vol. 4, no. 3, pp. 279-286, September 2016.

[8] J C Das and D De, "Circuit switching with Quantum-Dot Cellular Automata," *Nano communication networks*, vol. 14, pp. 16-28, December 2017.

[9] J C Das and D De, "Novel low power reversible binary incrementer design using quantum-dot cellular automata," *Microprocessors and Microsystems*,

vol. 42, pp. 10-23, May 2016.

[10] B Singh, R K Sarin, and B Raj, "Design and analysis of area efficient QCA based reversible logic gates," *Microprocessors and Microsystems*, vol. 52, pp. 59-68, July 2017.

[11] J Ren and V K Semenov, "Progress With Physically and Logically Reversible Superconducting Digital Circuits," *IEEE Transactions on Applied Superconductivity*, vol. 21, no. 3, pp. 780-786, June 2011.

[12] S Bandyopadhyay, A Balandin, V P Roychowdhury, and F Vatan, "Nanoelectronic implementations of reversible and quantum logic," *Superlattices and Microstructures*, vol. 23, no. 3-4, pp. 445-464, March 1998.

[13] C Taraphdar, T Chattopadhyay, and J N Roy, "Mach–Zehnder interferometer-based all-optical reversible logic gate," *Optics and Laser Technology*, vol. 42, no. 2, pp. 249-259, March 2010.

[14] S Karunamurthi and V K Natarajan, "VLSI implementation of reversible logic gates cryptography with LFSR key," *Microprocessors and Microsystems*, vol. 69, pp. 68-78, September 2019.

[15] T Purkayastha, D De, and K Das, "A novel pseudo random number generator based cryptographic architecture using quantum-dot cellular automata," *Microprocessors and Microsystems*, vol. 45, no. Part A, pp. 32-44, August 2016.

[16] E Fredkin and Thomas Toffoli, "Conservative Logic," *International Journal of Theoretical Physics*, vol. 21, pp. 219-253, 1982.

[17] Richard P Feynman, "Simulating Physics with Computers," *International Journal of Theoretical Physics*, vol. 21, no. 6, pp. 467-488, July 1982.

[18] Asher Peres, "Reversible Logic and Quantum Computers," *Physical Review A*, vol. 32, no. 6, p. 3266, Dec 1985.

[19] Thomas Toffoli, "Reversible Computing," MIT Lab for Computer Science, Tech Memo MIT/LCS/TM-151, 1980.

[20] Dmitri Maslov and Gerhard W. Dueck, "Improved Quantum Cost for n-bit Toffoli Gates," *IEEE Electronics Letters*, vol. 39, no. 25, pp. 1790 - 1791, Dec 2004.

[21] Dmitri Maslov and D Michael Miller, "Comparison of the Cost Metrics for Reversible and Quantum Logic Synthesis," *IET Computers & Digital Techniques, 1(2):98-104*, vol. 1, no. 2, pp. 98-104, February 2008.

[22] Dmitri Maslov and Gerhard W Dueck, "Reversible Cascades With Minimal Garbage," *IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS*, vol. 23, no. 11, pp. 1497-1509, November 2004.

[23] KETAN N0. PATEL, IGOR L. MARKOV, and JOHN P. HAYES, "Optimal Synthesis of Linear Reversible Circuits," *Quantum Information and Computation*, vol. 8, no. 3-4, pp. 282-294, 2008.

[24] Dmitri Maslov, Gerhard W Dueck, and D Michael Miller, "Synthesis of Fredkin–Toffoli Reversible Networks," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 13, no. 6, pp. 765-769, June 2005.

[25] S Banerjee et al., "Toffoli Netlist Based Synthesis of Four Variable Reversible Functions," in *Third IEEE International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN 2017)*, Kolkata, 2017, pp. 315-320.

[26] Dmitri Maslov, Gerhard W. Dueck, and D. Michael, "Toffoli Network Synthesis With Templates," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 6, pp. 807 - 817, June 2005.

[27] Daniel Große, Robert Wille, Gerhard W. Dueck, and Rolf Drechsler, "Exact Multiple-Control Toffoli Network Synthesis With SAT Techniques," *IEEE*

Transactions On Computer-Aided Design Of Integrated Circuits And Systems, vol. 28, no. 5, pp. 703 - 715, May 2009.

[28] Mehdi Saeedi, Mehdi Sedighi, and Morteza Saheb Zamani, "A library-based synthesis methodology for reversiblelogic," *Microelectronics Journal*, vol. 41, no. 4, pp. 185-194, March 2010.

[29] D Maslov, G W. Dueck, and D. Michael, "Toffoli Network Synthesis With Templates," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 6, pp. 807 - 817, June 2005.

[30] D M Miller, D Maslov, and G W Dueck, "A transformation based algorithm for reversible logic synthesis," in *Design Automation Conference*, 2003, pp. 318 - 323.

[31] V V. Shende, A K. Prasad, I L. Markov, and J P. Hayes, "Synthesis of Reversible Logic Circuits," *IEEE Transactions On Computer-Aided Design Of Integrated Circuits And Systems*, vol. 22, no. 6, pp. 710 - 722, June 2003.

[32] M Soeken, G W Dueck, and D M Miller, "A Fast Symbolic Transformation Based Algorithm for Reversible Logic Synthesis," in *International Conference on Reversible Computation*, 2016, pp. 307-321.

[33] C Y Lu and S A Wang, "Reversible Logic Synthesis Based on a Modified Tabulation Method," in *International Conference on Multimedia Technology*, 2011, pp. 2541-2544.

[34] Y Pang, J Lin, S Sultana, and K Radecka, "A Novel Method of Synthesizing Reversible Logic," in *IEEE International Symposium of Circuits and Systems (ISCAS)*, 2011, pp. 2857-2860.

[35] M Rawski and P Szotkowski, "Reversible Logic Synthesis of Boolean Functions Using Functional Decomposition," in *22nd International Conference Mixed Design of Integrated Circuits & Systems*, 2015, pp. 380-385.

[36] MEHDI SAEEDI, MORTEZA SAHEB ZAMANI, MEHDI SEDIGHI, and

ZAHRA SASANIAN, "Reversible Circuit Synthesis Using a Cycle-Based Approach," *ACMJournal onEmerging Technologies inComputing Systems*, vol. 6, no. 4, pp. 13-13:26, December 2010.

[37] M Lukac et al., "Evolutionary Approach to Quantum and Reversible Circuits Synthesis," *Artificial Intelligence Review*, vol. 20, no. 3, pp. 361-417, December 2003.

[38] K Datta, B Ghuku, D Sandeep, I Sengupta, and H Rahaman, "A Cycle based Reversible Logic Synthesis Approach," in *Third International Conference on Advances in Computing and Communications*, 2013, pp. 316-319.

[39] D V Zakablukov, "Application of Permutation Group Theory in Reversible logic Synthesis," in *International Conference on Reversible Computation*, 2016, pp. 223-238.

[40] R Wille, D GroBe, G W Dueck, and R Drechsler, "Reversible logic synthesis with output permutation," in *22nd International Conference on VLSI Design*, New Delhi, 2009, pp. 189-194.

[41] Ashis Kumer Biswas, Md. Mahmudul Hasan, Ahsan Raja Chowdhury, and Hafiz Md. Hasan Babu, "Efficient approaches for designing reversible Binary Coded Decimal adders," *Microelectronics Journal, Elsevier*, vol. 39, no. 12, pp. 1693-1703, December 2008.

[42] K Datta, I Sengupta, and H Rahaman, "A Post-Synthesis Optimization Technique for Reversible Circuits Exploiting Negative Control Lines," *IEEE Transactions On Computers*, vol. 64, no. 4, pp. 1208 - 1214, April 2015.

[43] Matthew Morrison and Nagarajan Ranganathan, "A Novel Optimization Method for Reversible Logic Circuit Minimization," in *IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, 2013, pp. 182-187.

[44] Md. S Islam, M M Rahman, and Z Begum, "Fault tolerant reversible logic synthesis: Carry look-ahead and carry-skip adders," in *International Conference on Advances in Computational Tools for Engineering Applications*,

2009. ACTEA '09., 2009, pp. 396-401.

[45] K Datta et al., "Exploiting Negative Control Lines in the Optimization of Reversible Circuits," in *International Conference on Reversible Computing*, 2013, pp. 209-220.

[46] Kamalika Datta, Indranil Sengupta, and Hafizur Rahaman, "A Post-Synthesis Optimization Technique for Reversible Circuits Exploiting Negative Control Lines," *IEEE Transactions on Computers*, vol. 64, no. 4, pp. 1208 - 1214, April 2015.

[47] Dilip P. Vasudevan, Parag K. Lala, Jia Di, and J. Patrick Parkerson, "Reversible-Logic Design With Online Testability," *IEEE Transactions On Instrumentation And Measurement*, vol. 55, no. 2, pp. 406 - 414, April 2006.

[48] H Thapliyal and M B Srinivas, "Novel Reversible `TSG' Gate and Its Application for Designing Components of Primitive Reversible/Quantum ALU," in *Fifth International Conference on Information, Communications and Signal Processing, 2005* , 2005.

[49] Goutam Kumar Maity and Santi P Maity, "Implementation of HNG using MZI," in *Third International Conference on Computing Communication & Networking Technologies (ICCCNT), 2012* , 2012, pp. 1-6.

[50] Diganta Sengupta, Mahamuda Sultana, and Atal Chaudhuri, "Realization of a Novel Reversible SCG Gate and its Application for Designing Parallel Adder/Subtractor and Match Logic," *International Journal of Computer Applications*, vol. 31, no. 9, pp. 30-35, October 2011.

[51] Rekha K James, K Poulose Jacob, and Sreela Sasi, "Design of compact reversible decimal adder using RPS gates," in *World Congress on Information and Communication Technologies (WICT), 2012*, 2012, pp. 344 - 349.

[52] Majid Haghparast and Keivan Navi, "A Novel Reversible Full Adder Circuit for Nanotechnology Based Systems," *Journal of Applied Sciences*, vol. 7, no.

24, pp. 3995-4000, 2007.

[53] Majid Haghparast and Keivan Navi, "A Novel Reversible BCD Adder For Nanotechnology Based Systems," *American Journal of Applied Sciences*, vol. 5, no. 3, pp. 282-288, 2008.

[54] Md. Saiful Islam, Muhammad Mahbubur Rahman, and Zerina Begum, "Fault tolerant reversible logic synthesis: Carry look-ahead and carry-skip adders," in *International Conference on Advances in Computational Tools for Engineering Applications, 2009. ACTEA '09.*, 2009, pp. 396-401.

[55] S B Rashmi, T G Umarani, and H K Shreedhar, "Optimized Reversible Montgomery Multiplier ," *International Journal of Computer Science and Information Technologies*, vol. 2, no. 2, pp. 701-706, 2011.

[56] M Arun and S Saravanan, "Reversible Arithmetic Logic Gate (ALG) for Quantum Computation," *International Journal of Intelligent Engineering and Systems*, vol. 6, no. 3, pp. 1-9, 2013.

[57] Papiya Biswas, Namit Gupta, and Nilesh Patidar, "Basic Reversible Logic Gates and It's QCA Implementation," *Int. Journal of Engineering Research and Applications*, vol. 4, no. 6, pp. 12-16, June 2014.

[58] S Banerjee, A K Pal, M Sultana, D Sengupta, and A Das, "Reversible Code Converters based on Application Specific Four Variable Reversible Gates," in *International Conference on Emerging Technologies in Data Mining and Information Security (IEMIS 2018), Springer*, Kolkata, 2018, p. In Press.

[59] M Sultana, A Chaudhuri, D Sengupta, and A Chaudhuri, "Logic Design and Quantum Mapping of a Novel Four Variable Reversible s2c2 Gate," in *Annual Convention of Computer Society of India (CSI 2017), Springer*, Kolkata, 2018, pp. 416-427.

[60] A Chaudhuri, M Sultana, D Sengupta, and A Chaudhuri, "A novel reversible two's complement gate (TCG) and its quantum mapping," in *Devices for*

Integrated Circuit (DevIC), IEEE, Kalyani, 2017, pp. 252-256.

[61] A Chaudhuri, M Sultana, D Sengupta, C Chaudhuri, and A Chaudhuri, "A Reversible Approach to Two's Complement Addition using a Novel Reversible TCG Gate and its 4 Dot 2 Electron QCA Architecture," *Microsystem Technologies, Springer*, vol. 25, pp. 1965-1975, July 2018.

[62] H R Bhagyalakshmi and M K Venkatesha, "Design of a Multifunction BVMF Reversible Logic Gate and its Applications," *International Journal of Computer Applications*, vol. 32, no. 3, pp. 0975 – 8887, October 2011.

[63] Md H A Khan and M Perkowski, "Synthesis of Reversible Synchronous Counters," in *41st IEEE International Symposium on Multiple-Valued Logic*, Tuusula, Finland, 2011, pp. 242-247.

[64] V Rajmohan and V Ranganathan, "Design of counters using reversible logic," in *3rd International Conference on Electronics Computer Technology*, Kanyakumari, India, 2011, pp. 138-142.

[65] M Morrison and N Ranganathan, "Design of a Moore finite state machine using a novel reversible logic gate, decoder and synchronous up-counter," in *11th IEEE International Conference on Nanotechnology*, Portland, OR, USA, 2011, pp. 1445-1449.

[66] R Singh and M K Pandey, "Design and optimization of sequential counters using a novel reversible gate," in *International Conference on Computing, Communication and Automation (ICCCA)*, Noida, India, 2016, pp. 1393-1398.

[67] G C Naguboina and K Anusudha, "Realization and Synthesis of Ring Counter and Twisted Ring Counter Using Reversible Logical Computation with Minimum Quantum Cost," in *International Conference on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, India, 2018, pp. 926-931.

[68] M Sultana, A Chaudhuri, D Sengupta, and A Chaudhuri, "Toffoli Netlist and QCA Implementations for Existing Four Variable Reversible Gates – A Comparative Analysis," *Microsystem Technologies, Springer*, vol. 25, pp.

1987-2009, July 2018.

[69] M Sultana, A Chaudhuri, D Sengupta, D De, and A Chaudhuri, "Design of Synchronous Decimal Counter using Reversible Toffoli-Fredkin Netlist," *Journal of Circuits, Systems, and Computers*, vol. Communicated, 2019.

[70] M Sultana, D Sengupta, and A Chaudhuri, "Taxonomy Proposal for Research on Reversible Logic," *INternational Journal of Engineering and Advanced Technology*, vol. 8, no. 6, pp. 2608-2613, August 2019.

[71] K Walus, T J Dysart, and G A Jullien, "QCADesigner: a rapid design and Simulation tool for quantum-dot cellular automata," *IEEE Transactions on Nanotechnology*, vol. 3, no. 1, pp. 26-31, March 2004.

[72] Kamalika Datta, Indranil Sengupta, and Hafizur Rahaman, "A Post-Synthesis Optimization Technique for Reversible Circuits Exploiting Negative Control Lines," *IEEE Transactions On Computers*, vol. 64, no. 4, pp. 1208 - 1214, April 2015.

[73] Mona Arabzadeh and Mehdi Saeedi. (2008-2013, version 2.5) RCViewer+: A viewer/analyzer for reversible and quantum circuits. [Online]. http://ceit.aut.ac.ir/QDA/RCV.htm

[74] H Maity, A K Barik, A Biswas, A K Bhattacharjee, and A Pal, "Design of Quantum Cost, Garbage Output and Delay Optimized BCD to Excess-3 and 2's Complement Code Converter," *Journal of Circuits, Systems and Computers*, vol. 27, no. 12, p. 1850184, 2018.

[75] A N Nagamani, S Ashwin, B Abhishek, K V Arjun, and V K Agrawal, "Design and Analysis of Multiple Parameters Optimized n-Bit Reversible Magnitude Comparators," *Journal of Circuits, Systems and Computers*, vol. 25, no. 09, p. 1650112, 2016.

[76] H V Jayashree, S Patil, and V K Agrawal, "Design Approaches for Resource and Performance Optimization of Reversible BCD Addition and Unified BCD Addition/Subtraction Circuits," *Journal of Circuits, Systems and Computers*,

vol. 27, no. 03, p. 1850048, 2018.

[77] P Murugesan, T Keppanagounder, and V Natarajan, "Design of Efficient Reversible BCD Adder–Subtractor Architecture and Its Optimization Using Carry Skip Logic," *Journal of Circuits, Systems and Computers*, vol. 25, no. 07, p. 1650076, 2016.

[78] M L Chuang and C Y Wang, "Synthesis of Reversible Sequential Elements," *ACM Journal on Emerging Technologies in Computing Systems*, vol. 3, no. 4, pp. 19:1-19:12, January 2008.

[79] P Joshi and I Sahu, "A Review Paper on Design of an Asynchronous counter using novel reversible SG gate," in *International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*, Bangalore, India, 2017, pp. pp.617-621.

[80] S Rakshith and T Rakshith, "Contemplation of Synchronous Gray Code Counter and its Variants using Reversible Logic Gates," in *IEEE Conference on Information & Communication Technologies*, Thuckalay, Tamil Nadu, India, 2013, pp. pp.661-665.

[81] S L Camina, "A Comparison of Taxonomy Generation Techniques Using Bibliometric," Massachusetts Institute of technology, EECS Thesis 2010.

[82] P H Raven, B Berlin, and D E Breedlove, "The Origins of Taxonomy," *Science*, vol. 174, no. 4015, pp. 1210-1213, 1971.

[83] V Blagoderov, I Brake, T Georgiev, L Penev , and et al., "Streamlining taxonomic publication: a working example with Scratchpads and ZooKeys," *Zookeys*, pp. 17-28, 2010.

[84] S A Thomson, R A Pyle, S T Ahyong, and et al., "Taxonomy based on science is necessary for global conservation," *Plos Biology*, vol. 16, no. 3, pp. 1-12, 2018.

[85] D Sengupta and M Sultana, "Taxonomy of Decimal Multiplier Research," in

Conference on Algorithms and Applications (ALAP 2018), Springer, Kolkata, 2018, pp. 3-21.

[86] A K Pal, S Banerjee, N Dey, and D Sengupta, "IoT Based Home Automation," in *IEEE 3rd International Conference for Convergence in Technology (I2CT)*, Pune, 2018, pp. 1-6.

[87] M Morrison and N Ranganathan, "Design of a Reversible ALU Based on Novel Programmable Reversible Logic Gate Structures," in *IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, Chennai, 2011, pp. 126-131.

[88] M Saeedi and I L. Markov, "Synthesis and Optimization of Reversible Circuits - A Survey," *ACM Computing Surveys*, vol. 45, no. 2, February 2013.

[89] M Sultana et al., "Comprehensive quantum analysis of existing four variable reversible gates," in *2017 Devices for Integrated Circuit (DevIC), IEEE*, Kolkata, 2017, pp. 116-120.

[90] R L Cilibrasi and P M B Vitanyi, "The google similarity distance," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 3, pp. 370-383, 2007.