# QUANTUM COMPUTING APPLICATION FOR COST REDUCTION TECHNOLOGY USING REVERSIBLE LOGICAL SYNTHESIS

Dikshant Sharma
Mtech Stucent, YMCA University of Science & Technology

Sonam Khera
Assistant Professor, YMCA University of Science & Technology

Neelam Turk
Professor, Deptt of ECE, YMCA University of Science & Technology

**Abstract**— Using Toffoli and Feynman gates, Peres gate, and SMG gate, three prototypes of reversible two's complement adder / subtractor are planned and compared. Following that, three reversible circuit designs for implementing a 2's complement circuit of adder / subtractor including an overflow detection is built in and got it checked too. Then, a reversible form of BCD circuit is implemented and is created and checked at the same time. For the previous designs, the binary coded decimal adder designs are checked and quantitatively evaluated. Quantum cost, delay, and transistor cost are all considered when evaluating reversible circuits.

Classical graph theory algorithms are considered, and quantum techniques such as the quantum minimum finding algorithm are applied to them in order to solve them in a quantum computer and thus compute their query complexity. The median of a graph and the middle of a graph are determined using quantum query complexity in both a classical and quantum method to define the 'Service Facility Location Problem' and the 'Emergency Facility Location Problem,' respectively.

The related project parameters should be handled proactively, and a technical manager should be able to predict the project outcome. Technical administrators are mostly concerned with the end result. Instead, they should concentrate on proactively monitoring and mitigating defect leakage at the outset. In a quantum project functionally the predictive models are applicable to test the defects, that overall helps in the reduction of residual defects.

**Keywords: Quantum cost; Quantum gates; prototypes**

## I. INTRODUCTION

Over the previous few decades, pc users became familiar with associate new increase in process speed and power. Gordon Moore found in 1965 that chip power doubled last year. whereas the speed of growth has slowed to "only" doubling each eighteen months, "Moore's law" has projected a geometrical rise for over forty years. High-end PCs these days have the process capability of devices that were once known as supercomputers. code advancements are equally forceful, however most notably within the style of camera work, that is maybe commonest to the typical client. The near-photorealistic graphics of today's video games and films have replaced the primitive coloured dots and flat polygons of pc games from twenty years agone.

Complicated code utilized in pc animations, biological science analysis, machine fluid dynamics, world climate and economic modelling, worldwide mastercard process, and a spread of different advanced applications necessitates an enormous quantity of computing capability. because of the stress of those drawback domains, researchers have developed distributed computing systems that mix the process power of thousands, if not millions, of processors into clusters. However, there square measure some limitations to the present strategy. Adding additional processors will increase the process power of those clusters linearly, whereas the scale of their inputs grows exponentially, that may be a important drawback. The machine demands of those issues appear to be inherent within the issues themselves, i.e., no algorithmic rule workable on a computer, the standard model of computing, will solve the matter with less exponential resources in time, memory, and processors, per the overwhelming agreement.

Moore's law has enabled scientists to tackle way larger issues than within the past by doubling computing power each eighteen months, however even Moore's law has its limits. The doubling of capability with every new chip generation implies that concerning [*fr1] as several atoms square measure used per little bit of info. once compute into the longer term, this pattern hits a saturation of 1 atom per little bit of information between 2020 and 2030. This doesn't essentially imply that machine progress would stall at that stage. Quantum computing may be a trendy technology that has the flexibility to greatly increase the pace of development in computing potency. A quantum computer's feature is that it deviates from the computer paradigm of computation. There square measure many functions which will be computed additional expeditiously on a quantum pc than on a conventional pc. This outstanding reality underpins quantum computing's monumental strength (Manin).

## II. What is Quantum Computing

Simulating quantum mechanical systems takes a great deal of classical energy, however if quantum effects may be controlled, they'll do a great deal of classical computation, in line with Feynman (1982). Peter Shor showed that quantum computers ar capable of expeditiously factorization giant numbers (Ekert & Jozsa 1996). this is often notably exciting since it's normally assumed that no effective factorization algorithmic program for classical computers is feasible. Grover (1997) planned a quantum algorithmic program for determination the overall search drawback in O(n).

Quantum computing's attract was additionally hampered by one flaw. Quantum effects ar terribly delicate. Noise distorts quantum behavior and suppresses non-classical phenomena even at atomic scales. To smooth random noise, ancient computers use millions or perhaps billions of atoms or electrons. Tests and compares bits on the manner in communication, storage, and process to spot and proper minor errors till they accumulate and cause incorrect results, inclined messages, or perhaps device crashes. once a quantum system is measured, it falls into one among the measure bases, exploit no proof of previous superpositions (Nielsen & Chuang 2016).

Steane (1997) created a major breakthrough once he discovered ways for coding quantum bits, or "Qubits," and measurement cluster properties, permitting even minor errors to be corrected.

## III What are Quantum Bits

Consider the binary strings 011, 111, that correspond to the numbers three and seven in binary. In general, 3 physical bits are often designed in 23=8 other ways to represent integers starting from zero to seven. At any given time, a register created of 3 classical bits will solely hold one variety.

A quantum system during which the Boolean states zero and one ar portrayed by a such as try of normalised and reciprocally orthogonal quantum states labeled as 1> (Steane 1997). the other (pure) state of the Qubit are often written as a superposition for a few |0> + β |1> for a few for a few for a few that |α|2 + |β|2 = one. A quabit may be a unit of data.

A molecule, a nuclear spin, or a polarised gauge boson ar samples of microscopic particles. A Quantum Register of Size n may be a set of n Qubits.

Assume that the info is hold on in binary type within the registers. as an example, a register in state |1> cnot |1> cnot |0> represents the quantity half dozen. in additional compact notation, a represents a quantum register ready with the worth a 20a0 21a1... 2n1an1 and represents the tensor product |an-1> cnot |an-2>... |a1> cnot |a0>, wherever ai E . There ar 2n states of this sort, that represent all binary strings of length n or numbers from zero to 2n-1 and function a helpful procedure foundation.

As shown in Equations, a quantum register of size 3 will store individual numbers like three or seven. (1.1 and 1.2).

$$|0\rangle \otimes |1\rangle \otimes |1\rangle \equiv |011\rangle \equiv |3\rangle \tag{1.1}$$

$$|1\rangle \otimes |1\rangle \otimes |1\rangle \equiv |111\rangle \equiv |7\rangle \tag{1.2}$$

But, it can also store the two of them simultaneously. Consider tl first Qubit, instead of setting it to $|0\rangle$ or $|1\rangle$, prepare a superpositic $1/\sqrt{2}(|0\rangle+|1\rangle)$, obtained as given in Equation (1.3).

$$\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle)\otimes|1\rangle\otimes|1\rangle \equiv \frac{1}{\sqrt{2}}(|011\rangle+|111\rangle) \equiv \frac{1}{\sqrt{2}}(|3\rangle+|7\rangle) \qquad (1.3)$$

This register can in fact be prepared in a superposition of all eig numbers, which can be done by putting each Qubit into the superpositic $1/\sqrt{2}(|0\rangle+|1\rangle)$, which results in Equation (1.4).

$$\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle)\otimes\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle)\otimes\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle) \qquad (1.4)$$

This can also be written in binary (ignoring the normalization constant $2^{-3/2}$) as given in Equations (1.5 to 1.7).

$$|000\rangle+|001\rangle+|010\rangle+|011\rangle+|100\rangle+|101\rangle+|110\rangle+|111\rangle \qquad (1.5)$$

$$\equiv |0\rangle+|1\rangle+|2\rangle+|3\rangle+|4\rangle+|5\rangle+|6\rangle+|7\rangle \qquad (1.6)$$

$$\equiv \sum_{x=0}^{7}|x\rangle \qquad (1.7)$$

## IV. About Quantum Gates

Digital computers were created from thousands or several straightforward gates, starting from microprocessors to supercomputers, from little chips in wristwatches and microwaves to continent-spanning distributed networks that handle world mastercard transactions. every gate performs one operation, like manufacturing a one once all of its inputs ar one, or inverting a one to a zero and a zero to a one. Engineers use these basic gates to form additional advanced circuits that add or multiply 2 numbers, decide a memory location, or opt for that directions to execute next supported the result of AN operation. Engineers produce progressively advanced modules from these circuits, leading to computers, CD players, craft navigation systems, optical maser printers, and mobile phones. whereas vital challenges stay, like transmission signals at GHz speeds, having 1,000,000 gates to work in good lockstep, or storing 10 billion bits while not missing one one, once an easy gate is meant, the remainder is "merely" a style. A quantum gate (Vedral & Plenio 1998) may be a system that performs a hard and fast unitary operation on a group of Qubits over a planned time span.

## V. Single qubit gates

The following are some examples of single-qubit quantum gates that can be useful (Meglicki).

- Identity Transformation I

|0> → |0>

|1> → |1>

Matrix Model

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

The circuit representation of the *I* gate is given in Figure 1.1.

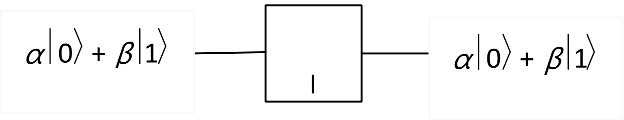

$\alpha|0\rangle + \beta|1\rangle$
I
$\alpha|0\rangle + \beta|1\rangle$

**Figure 1.1. I gate**

- Not Gate - X Gate

|0> → |1>

|1> → |0>

Matrix Model

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

The circuit representation of the *X* gate is given in Figure 1.2.

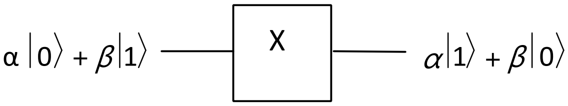

$\alpha|0\rangle + \beta|1\rangle$
X
$\alpha|1\rangle + \beta|0\rangle$

**Figure 1.2. X gate**

- Y Gate, Y=ZX

$$|0\rangle \rightarrow -|1\rangle$$
$$|1\rangle \rightarrow |0\rangle$$

For a Qubit in state 0, the magnitude of the amplitude is converted to 1 and the phase of the amplitude is inverted.
Matrix Model

$$\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

The Circuit representation of Y gate is given in Fig 1.3.

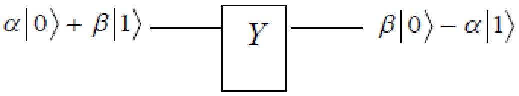

$\alpha|0\rangle + \beta|1\rangle$
$Y$
$\beta|0\rangle - \alpha|1\rangle$

**Figure 1.3. Y gate**

- Z Gate

$$|0\rangle \rightarrow |0\rangle$$
$$|1\rangle \rightarrow -|1\rangle$$

Matrix Model

$$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

The circuit representation of Z gate is given in Fig 1.4.

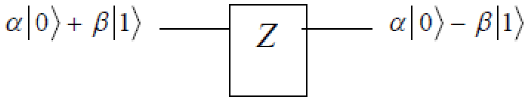

$\alpha|0\rangle + \beta|1\rangle$
$Z$
$\alpha|0\rangle - \beta|1\rangle$

**Figure 1.4. Z gate**

All the above gates are unitary (Shende), for example, consider *Y*Gate, which is shown in Equation (1.8).

$$YY^{*} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} = I \tag{1.8}$$

- Transformation of Walsh-Hadamard

The Hadamard Transformation, described by, is another essential single-bit transformation.

$$H : |0\rangle \rightarrow \frac{(|0\rangle + |1\rangle)}{\sqrt{2}}$$

$$|1\rangle \rightarrow \frac{(|0\rangle - |1\rangle)}{\sqrt{2}}$$

The circuit representation of *H* gate is given in Figure 1.5.

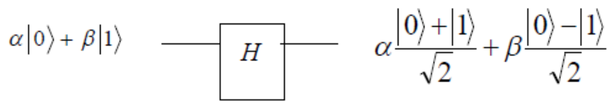

$\alpha|0\rangle+\beta|1\rangle$
H
$\alpha\frac{|0\rangle+|1\rangle}{\sqrt{2}}+\beta\frac{|0\rangle-|1\rangle}{\sqrt{2}}$

**Figure 1.5. H gate**

The transformation *H* has a number of important applications. When applied to|0>, H creates a superposition state.

$$\frac{(|0\rangle+|1\rangle)}{\sqrt{2}}$$

The transformation that applies *H* to *n* bits is called the Walsh or Walsh Hadamard transformation *W*.

### 1.2.3.2 Multiple qubit gates

Multiple Qubit Gates (Meglicki) area unit offered within the following examples.

- Controlled-NOT Gate

It is a kind of gate that's controlled by the user. it is also called the CNOT gate. The management Qubit and therefore the Target Qubit area unit the 2 input Qubits for this gate. Figure 1.6 depicts the CNOT gate's circuit illustration.

The goal Qubit is painted by rock bottom line, whereas the management Qubit is painted by the highest line. The gate's operation is summarised as follows. The goal Qubit is left alone if the management Qubit is ready to zero. The goal Qubit is flipped if the management Qubit is ready to one. The CNOT transformation is delineated as follows.

$$\begin{gathered}|00\rangle\to|00\rangle\\|01\rangle\to|01\rangle\\|10\rangle\to|11\rangle\\|11\rangle\to|10\rangle\end{gathered}$$

Matrix Model

$$\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{pmatrix}$$

The CNOT gate may be thought of as a generalisation of the classical logic gate |A, B> □ |A, B xor A> , wherever is that the addition modulo two, that is strictly what the logic gate accomplishes.

- Controlled-Controlled-NOT

Toffoli gate is another name for it. It's created from 3 Qubits. If and on condition that the primary 2 bits square measure each one, it negates the last little bit of 3. the subsequent could be a illustration of it.

$$\begin{gathered}|000\rangle\to|000\rangle\\|001\rangle\to|001\rangle\\|010\rangle\to|010\rangle\\|011\rangle\to|011\rangle\\|100\rangle\to|100\rangle\\|101\rangle\to|101\rangle\\|110\rangle\to|111\rangle\\|111\rangle\to|110\rangle\end{gathered}$$

Matrix Model

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

A circuit of the form shown in Figure 1.7 is commonly used to describe the controlled-controlled-NOT gate.

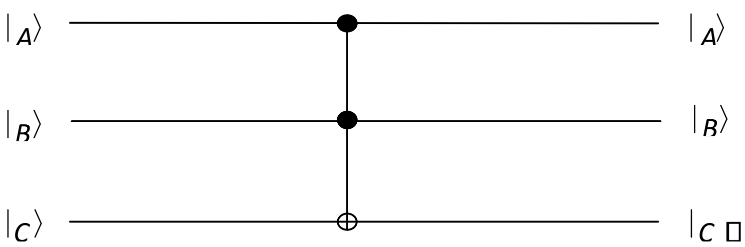


**Figure 1.7. Controlled-controlled-NOT gate**

## 1.2.4. QUANTUM PARALLELISM

In many quantum algorithms, quantum similarity (Nielsen & Chuang) could be a key feature. Quantum similarity allows quantum computers to check a operate f(x) for a large vary of x values at identical time.

Assume that f (x) : □ could be a one-bit domain operate which a large selection contemplate a two-qubit quantum machine that begins within the state | x, y > so as to work out this operate (Amiri).

This state are often remodeled into | x, y xor f(x)> with associate applicable sequence of logic gates, wherever xor indicates addition modulo a pair of, the primary register is termed the 'data' register, and therefore the second register is termed the 'target' register.

Let U f be the name of the transformation outlined by the map |x, y> → | x, y xor f (x)>. When y = 0, the second Qubit's final state is that the worth f (x).

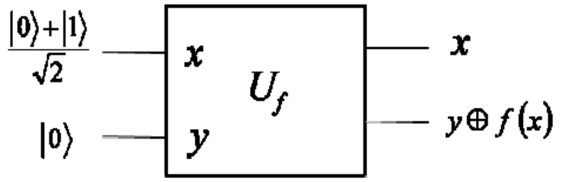


**Figure 1.8. Circuit evaluating *f (0)* and *f (1)*simultaneously**

In Figure one.8, Uf is applied to Associate in Nursing input that's not within the machine basis, however the info register is ready within the superposition, which might be generated with a Hadamard gate operative on |0>. After that, Uf is used, leading to the state shown in Equation(1.9).

$$\frac{|0, f(0)\rangle + |1, f(1)\rangle}{\sqrt{2}} \qquad (1.9)$$

This is a rare state of affairs. information is enclosed within the varied words. It's virtually as if f(x) has been checked for 2 totally different values of x at identical time for each f (0) and f (1). Quantum similarity is that the name for this perform.

Multiple circuits ar designed and dead at the same time in classical similarity to reckon f (x). However, by employing a quantum computer's capability to be in superpositions of various states, one f (x) circuit is employed to check the perform for many values of x at identical time.

This procedure may be extended to figure on any item. The Hadamard remodel could be a general procedure which will be wont to scale back the quantity of bits. This method consists of n Hadamard gates operative on n Qubits in parallel. it's shown sure n = a pair of with Qubits ready as |0> at the beginning.

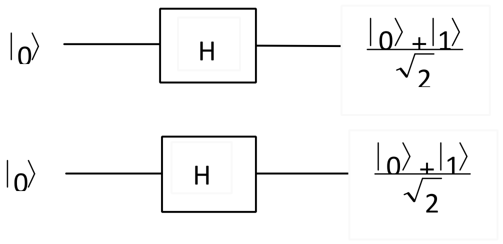

$|0\rangle$
H
$\frac{|0\rangle+|1\rangle}{\sqrt{2}}$
$|0\rangle$
H
$\frac{|0\rangle+|1\rangle}{\sqrt{2}}$

**Figure 1.9. Hadamard ransform $H^{\text{xor}2}$on two qubits**

The output will be $\left(\frac{|0\rangle+|1\rangle}{\sqrt{2}}\right)\left(\frac{|0\rangle+|1\rangle}{\sqrt{2}}\right) = \frac{|00\rangle+|01\rangle+|10\rangle+|11\rangle}{2}$, $H^{\otimes 2}$ denotes the parallel action of two Hadamard gates and □ is called as 'tensor'. In general, the result of performing the Hadamard transform on $n$ Qubits initially all in |0> state is given in Equation (1.10).

$$\frac{1}{\sqrt{2^n}}\sum_x |x\rangle \qquad (1.10)$$

In Equation (1.10), the sum is over all possible values of $x$ and it is denoted as $H^{\text{xor } n}$. Hence, the Hadamard transform produces an equal superposition of all computational basis states. It produces a superposition of $2^n$ states using just $n$ gates.

Quantum parallel evaluation of a function with an $n$ bit input $x$ and 1 bit output $f(x)$ can be performed in the following manner. Prepare $n$ □1 Qubit state 0 0 , and then apply the Hadamard transform to the first $n$ Qubits, followed by the quantum circuit implementing $U_f$. This produces the state given in Equation (1.11).

$$\frac{1}{\sqrt{2^n}}\sum_x |x\rangle|f(x)\rangle$$

(1.11)

Hence, quantum parallelism enables all possible values of the function $f$ to be evaluated simultaneously, even though $f$ is evaluated only once.

## VI Quantum Circuits

Since quantum circuits square measure the same as the logic style level of classical computation, all quantum computation is modelled at the quantum circuit level during this study. The Qubits (Postulate 1) and therefore the operators or gates square measure the 2 main elements of a quantum circuit (Postulate 2). The values of the Qubits square measure discovered by mensuration (Postulate 3), and therefore the tensor product will be wont to categorical multiple Qubits and gates (Postulate 4). Clearly, quantum physics postulates have a whole set of properties that to conduct logic style subject to the no-cloning theorem's fanout restriction. 2 tiny quantum circuit examples square measure thought-about within the remainder of this segment to acquaint you with the traditional quantum circuit notation.

### *A. To Swap the state of two qubits*

Consider Figure one.10, that depicts an easy quantum circuit with 3 quantum gates. 3 CNOT gates frame the circuit. every wire within the quantum circuit is diagrammatical by a line within the circuit. This wire might or might not represent a physical wire; it's going to represent the passage of your time or a physical particle like a gauge boson, a lightweight particle, move from one position to a different through house.

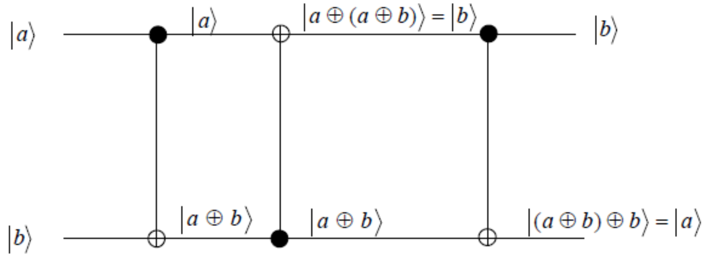


**Figure 1.10. Circuit depicting the swapping of two qubits**

The circuit in Figure one.8 switches the states of the 2 Qubits (Nielsen & Chuang), and therefore the gate sequence has the subsequent effects on a procedure basis state a,b, as shown in Equation (1.12).

$$|a,b\rangle \rightarrow |a,a\oplus b\rangle$$

$$\rightarrow |a\oplus(a\oplus b),a\oplus b\rangle = |b,a\oplus b\rangle$$

$$\rightarrow |b,(a\oplus b)\oplus b\rangle = |b,a\rangle$$

(1.12)

Here, xor denotes addition modulo 2. The effect of the circuit, therefore is to interchange the state of the two Qubits.

### *B. Bell States*

Figure 1.11 shows a quantum circuit for making Bell states. The circuit transforms the four process basis states in line with Table one.1 and consists of a Hadamard gate followed by a CNOT gate.

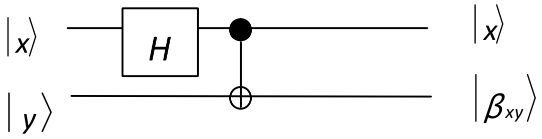


**Figure 1.11: A quantum circuit for producing Bell states**

The Hadamard remodel, for example, places the highest Qubit during a superposition. this is often then used as an impression input to the CNOT gate, and therefore the target is simply inverted once the management is ready to

$$|\beta_{00}\rangle = \frac{|00\rangle+|11\rangle}{\sqrt{2}}$$

$$|\beta_{01}\rangle = \frac{|01\rangle+|10\rangle}{\sqrt{2}}$$

$$|\beta_{10}\rangle = \frac{|00\rangle-|11\rangle}{\sqrt{2}}$$

$$|\beta_{11}\rangle = \frac{|01\rangle-|10\rangle}{\sqrt{2}}$$

(1.13)

The Bell states, also known as EPR states or EPR pairs, are named after Bell and Einstein, Podolsky and Rosen (Nielsen & Chuang).

## VII OBJECTIVE AND SCOPE

The research's main goal is to use reversible logic circuits to resolve Associate in Nursing application of quantum computing for value reduction. an outsized variety of quantum laptop reversible circuits are developed and synthesised. The circuits ar planned and checked for all attainable values, and that they ar compared to existing circuits in terms of the amount of constant inputs used, the amount of garbage outputs, quantum value, delay, and semiconductor device value. the subsequent may be a list of reversible circuits that has been developed.

- Design of a brand new reversible gate

- Design of a reversible two's complement adder/subtractor
- Design of a reversible two's complement adder/subtractor with overflow detection logic
- Design of Reversible binary coded decimal adder style
- The sophistication of a quantum question to guage the graph's median and middle
- Statistical strategies for predicting defects in quantum code comes

### 1.1. TOOLS USED FOR SIMULATION

➢ The following tools were used for simulation of reversible circuits within the current analysis.

➢  Quantum laptop soul (QCE) is employed to emulate numerous hardware styles of Quantum Computers. The QCE simulates the physical processes that govern the operation of the hardware quantum processor, strictly consistent with the laws of quantum physics. The QCE additionally provides AN surroundings to right and execute quantum algorithms beneath realistic experimental conditions. The version of QCE used is ten.11

➢  QuIDDPro, it's a quick, scalable, and easy-to use process interface for generic quantum circuit simulation. It supports state vectors, density matrices, and connected operations exploitation the Quantum info call Diagram (QuIDD) organisation. QuIDDPro could be a tool that's accustomed check the input and output of a circuit. It additionally provides some vital info like runtime in seconds, variety of gates applied, average runtime per gate in seconds, base memory usage in MB and peak further memory usage in MB. The version of QuIDDPro used is three.8

➢  QCViewer (Parent & Parker) could be a tool for displaying, editing, and simulating quantum circuits. It permits users to check new circuit styles and build publication quality diagrams with a simple to use graphical interface. Supported options additionally embody simulation of the circuit whereas diagrammatically displaying the present state. it's additionally helpful for viewing terribly large/complex circuits with the employment of sub circuit abstraction. The QCViewer provides a take a look ating choice to test the correctness of the circuits. By means that of providing inputs to the circuit and supportive whether or not the output obtained matches the desired results, every circuits are often tested severally and effectively by providing numerous inputs, any variety of times. The version of QCViewer used is one.2

## VII REVERSIBLE CIRCUITS

In literature, several mixtures of reversible gates and approaches area unit offered to style a traditional circuit exploitation reversible logic (Rangaraju). during this section, 3 styles for reversible two's complement adder/subtractor and 3 styles for reversible two's complement adder/subtractor with overflow detection logic area unit planned to realize the optimized reversible circuits. These styles are simulated exploitation QCviewer (Parent & Parker). The performances of the circuits area unit analyzed on the premise of total range of gates used, quantum value, and delay.

### A. Reversible Two's Complement Adder/Subtractor - style I

In style I, the total adders were enforced exploitation Toffoli and nuclear physicist reversible gates. To add/subtract 2 four bit binary numbers A3A2A1A0 and B3B2B1B0, the input to the circuit as in Figure four.3 was given within the following order |A0>|B0>|S>|S>|0>|A1>|B1>|S>|0>|A2>|B2>|S >|0>|A3> |B3>|S>|0> and also the outputs S0, S1, S2, S3 were measured in Qubit4, Qubit8, Qubit12, Qubit16, severally. Carry, C4 was measured in Qubit17. the quantity of inputs/outputs in style I is seventeen, when S=1, the circuit performs addition and once S=0, it performs subtraction. The reversible circuit for a four bit Two's Complement Adder/subtractor of this style is given in Figure four.3. the quantity

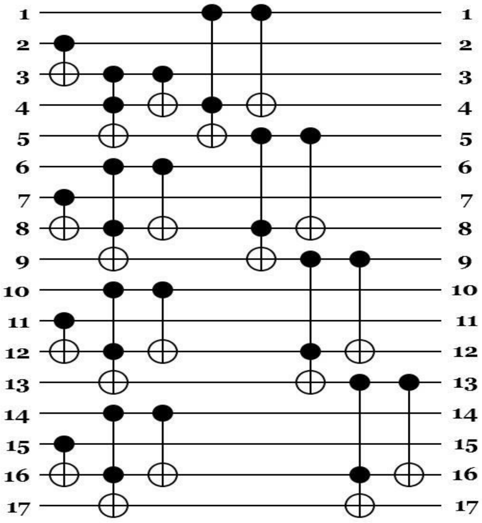


**Figure 2.3. Reversible two's complement adder/subtractor - Design I**

The quantum price and delay of reversible n-bit two's complement adder/subtractor - style I is calculated as follows,

- The step one of style I, wants n CNOT gates operating in parallel, hence, this step has the quantum price of n and delay of one one
- The step a pair of has n CCNOT gates. Thus, this step incorporates a quantum price of five.n and delay of one one
- The step three has n CNOT gates. Hence, the quantum price is n and delay is one one
- The step four has n CCNOT gates and n CNOT gates. Hence, the quantum price is (5n + n =6n) and (n-1) CNOT and (n-1) CCNOT gates work parallel, thence the delay for this step is (1+n-1+1 = n+1 Δ)

Thus, the overall quantum price of reversible n-bit two's complement adder/subtractor - style I is n+5n+n+6n=13.n, the propagation delay is 1Δ+1Δ+1Δ+(n+1) Δ = (n+4) Δ. Hence, the quantum price of the four-bit circuit given in Figure 4.3 is fifty two and therefore the propagation delay is eight eight.

Thus, the overall quantum price of reversible n-bit two's complement adder/subtractor - style I is n+5n+n+6n=13.n, the propagation delay is 1Δ+1Δ+1Δ+(n+1) Δ = (n+4) Δ. Hence, the quantum price of the four-bit circuit given in Figure 4.3 is fifty two and therefore the propagation delay is eight eight.

B. Reversible Two's Complement Adder/Subtractor - style II

In style II, the total adders area unit enforced mistreatment Peres gates. To add/subtract 2 four bit binary numbers A3A2A1A0 and B3B2B1B0, the input to the circuit as in Figure four.4 is given within the following order |A0>|B0>|S>|S>|0>|A1>|B1>|S>|0>|A2>|B2>| S>|0>|A3>|B3>|S>|0> and therefore the outputs S0, S1, S2, S3 area unit measured in Qubit4, Qubit5, Qubit9, Qubit13, severally.

The number of inputs/outputs in style II is seventeen, when S=1, the circuit performs addition and once S=0, it performs subtraction. The reversible circuit for a four bit Two's Complement Adder/subtractor of this style is given in Figure four.4. the overall range of reversible gates employed in style II is twelve.

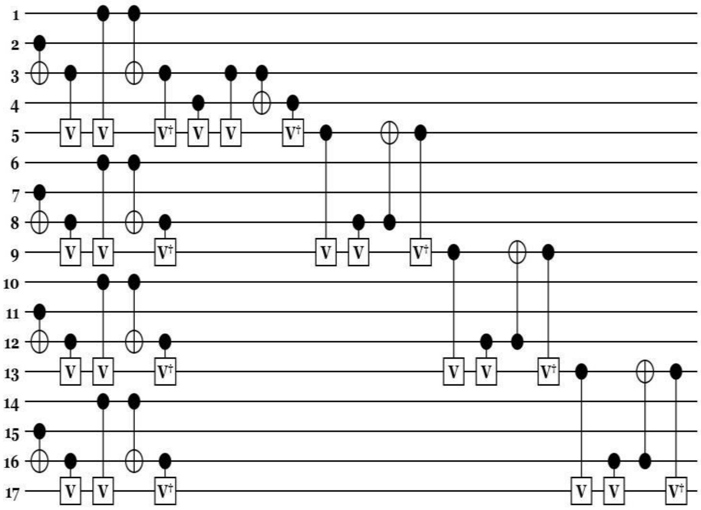


**Figure 4.4. Reversible two's complement adder/subtractor - Design II**

The quantum value and delay of reversible n-bit two's complement adder/subtractor - style II is calculated as follows,

- The step one of style II, wants n CNOT gates operating in parallel. Hence, this step has the quantum value of n and delay of one one
- The step a pair of has n Controlled V gates, operating in parallel. Thus, this step incorporates a quantum value of n and delay of one one
- The step three has n Controlled V gates, operating in parallel. Thus, this step incorporates a quantum value of n and delay of one one

□ The step four has n CNOT gates. Hence, the quantum value is n and delay is one one

- The step five has n Controlled V┼ gates, operating in parallel. Thus, this step incorporates a quantum value of n and delay of one one
- The step vi has a pair of Controlled V gates, a CNOT gate, and a Controlled V┼ gate, operating nonparallel for one bit. Hence, quantum value for n bit during this step is (n + n + n + n)= 4n and delay is (4n) Δ

Thus, the full quantum value of reversible n-bit two's complement adder/subtractor - style II is n+n+n+n+n+4n=9.n, the propagation delay is 1Δ+1Δ+1Δ+1Δ+1Δ+(4n) Δ =(4n+5)Δ. Hence, the quantum value of the four-bit circuit as given in Figure 4.4 is thirty six and therefore the propagation delay is twenty one twenty one.

**C Reversible Two's Complement Adder/Subtractor - style III**

In style III, the total adders ar enforced exploitation SMG gates. To add/subtract 2 four bit binary numbers A3A2A1A0 and B3B2B1B0, the input to the circuit as in Figure four.5 is given within the following order |S>|B0>|S>|A0>|0>|B1>|S>|A1>|0>|B2>|S>|A2>|0>|B3>|S>|A3>|0> and therefore the outputs S0, S1, S2, S3 ar measured in Qubit3, Qubit7, Qubit11, Qubit15, severally. the amount of inputs/outputs in style III is seventeen, when S=1, the circuit performs addition and once S=0, it performs subtraction. The reversible circuit for a four bit Two's Complement Adder/subtractor of this style is given in Figure four.5. the amount of reversible gates utilized in style III is eight.

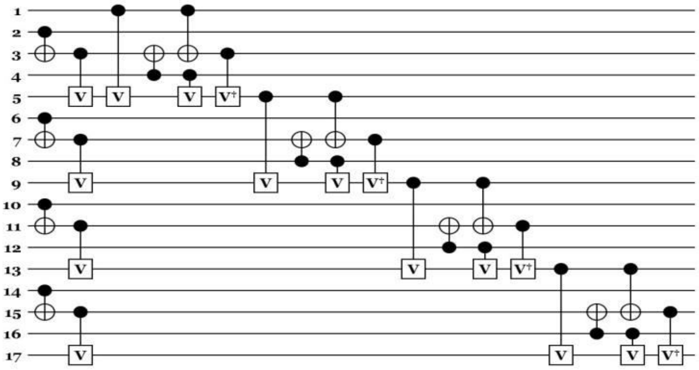


**Figure 4.5. Reversible two's complement adder/subtractor - Design III**

The quantum value and delay of reversible n-bit two's complement adder/subtractor with overflow detection logic -Design I is calculated as follows,

- The step one of style I, wants n CNOT gates operating in parallel. Hence, this step has the quantum value of n and delay of one one

- The step two has n CCNOT gates. Thus, this step includes a quantum value of five.n and delay of one one

- The step three has n CNOT gates. Hence, the quantum value is n and delay is one one

- The step four has n CCNOT gates and n CNOT gates. Hence, the quantum value is (5n + n =6n) and (n-1) CNOT and (n-1) CCNOT gates add parallel. Hence, the delay for this step is (1+n-1+1=n+1 Δ)

- The step five has one CNOT gates. Hence, the quantum value is one and delay is one one

Thus, the full quantum value of reversible n-bit two's complement adder/subtractor with overflow detection logic - style I is n+5n+n+6n+1=13n+1, the propagation delay is 1Δ+1Δ+1Δ+(n+1)Δ+1Δ

=(n+5) Δ. Hence, the quantum value of the four-bit circuit as given in Figure 4.6 is fifty three and therefore the propagation delay is nine nine.

### E. Reversible Two's Complement Adder/Subtractor with Overflow Detection Logic - style II

In style II, full adders created mistreatment Peres Gates are accustomed implement two's complement adder/subtractor with overflow detection logic. To add/subtract a four bit binary variety the input is |A0>|B0>|S>|S>|0>|A1>|B1>|S>|0>|A2>|B2>|S>|0>|A3>|B3>|S>|0> and therefore the outputs S0, S1, S2, S3 ar measured in Qubit4, Qubit5, Qubit9, Qubit11, severally. V is measured in Qubit13, V=1 states that associate overflow has occurred. The four-bit reversible circuit of reversible two's complement adder/subtractor with overflow detection logic - style II is given in Figure 4.7.

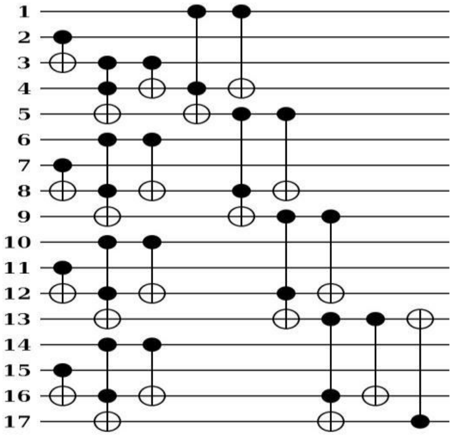


**Figure 4.6 Reversible 2's Complement Adder/Subtractor with Overflow Detection Logic - Design I**

The quantum price and delay of reversible n-bit two's complement adder/subtractor with overflow detection logic - style II is calculated as follows,

- The step one of style II, wants n CNOT gates operating in parallel. Hence, therefore this step has the quantum price of n and delay of one one

- The step a pair of has n Controlled V gates, operating in parallel. Thus, this step incorporates a quantum price of n and delay of one one

- The step three has n Controlled V gates, operating in parallel. Thus, this step incorporates a quantum price of n and delay of one one

- The step four has n CNOT gates. Hence, the quantum price is n and delay is one one

- The step five has n Controlled V┼ gates, operating in parallel. Thus, this step incorporates a quantum price of n and delay of one one

- The step six has a pair of Controlled V gates, a CNOT gate and a Controlled V┼ gate, operating asynchronous for one bit. Hence, quantum price for n bit during this step is (n + n + n + n)= 4n and delay is (4n) Δ

- The step seven has one CNOT gates. Hence, the quantum price is one and delay is one one

Thus, the whole quantum price of reversible n-bit two's complement adder/subtractor with overflow detection logic - style II is n+n+n+n+n+4n+1=9n+1, the propagation delay is 1Δ+1Δ+1Δ+1Δ+1Δ+(4n) Δ+ 1Δ =(4n+6)Δ. Hence, the quantum price of the four-bit circuit as given in Figure 4.7 is thirty seven and therefore the propagation delay is twenty two twenty two.

### F. Reversible Two's Complement Adder/Subtractor with Overflow Detection Logic - style III

In style III, full adder made exploitation SMG Gate has been used. To add/subtract a four bit binary variety, the input is |S>|B0>|S>|A0>|0>|B1>|S>|A1>|0>|B2>|S>|A2>|0>|B3>|S>|A3>|0> and the outputs S0, S1, S2, S3 square measure measured in Qubit3, Qubit7, Qubit11, Qubit15, severally. V is measured in Qubit13, V=1 states that associate overflow has occurred. The reversible circuit of style III is given in Figure four.8.

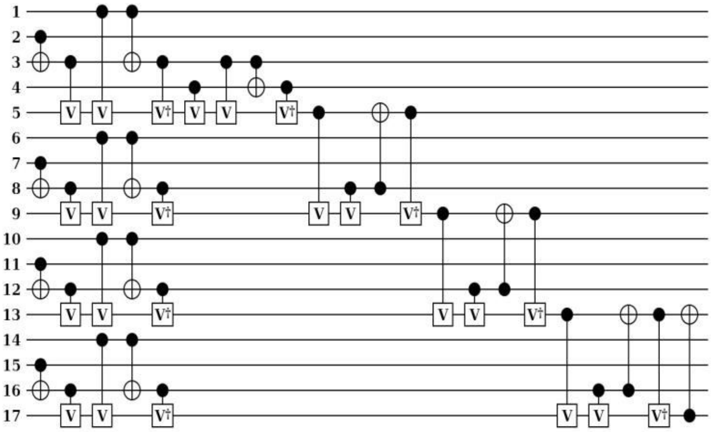


**Figure 4.7. Reversible 2's complement adder/subtractor with overflow detection logic- Design II**

The quantum cost and delay of reversible n-bit two's complement adder/subtractor with overflow detection logic - Design II is calculated as follows,

- The step 1 of Design II, needs n CNOT gates working in parallel. Hence, thus this step has the quantum cost of n and delay of 1 Δ
- The step 2 has n Controlled V gates, working in parallel. Thus, this step has a quantum cost of n and delay of 1 Δ
- The step 3 has n Controlled V gates, working in parallel. Thus, this step has a quantum cost of n and delay of 1 Δ
- The step 4 has n CNOT gates. Hence, the quantum cost is n and delay is 1 Δ
- The step 5 has n Controlled V┼ gates, working in parallel. Thus, this step has a quantum cost of n and delay of 1 Δ
- The step 6 has 2 Controlled V gates, a CNOT gate and a Controlled V┼ gate, working in series for 1 bit. Hence, quantum cost for n bit in this step is (n + n + n + n)= 4n and delay is (4n) Δ

- The step 7 has 1 CNOT gates. Hence, the quantum cost is 1 and delay is 1 Δ

Thus, the total quantum cost of reversible n-bit two's complement adder/subtractor with overflow detection logic - Design II is n+n+n+n+n+4n+1=9n+1, the propagation delay is 1Δ+1Δ+1Δ+1Δ+1Δ+(4n) Δ+ 1Δ =(4n+6)Δ. Hence, the quantum cost of the 4-bit circuit as given in Figure 4.7 is 37 and the propagation delay is 22 Δ.

### F. Reversible Two's Complement Adder/Subtractor with Overflow Detection Logic - Design III

In Design III, full adder constructed using SMG Gate has been used. To add/subtract a four bit binary number, the input is $|S>|B_0>|S>|A_0>|0>|B_1>|S>|A_1>|0>|B_2>|S>|A_2>|0>|B_3>|S>|A_3>|0>$ and the outputs $S_0$, $S_1$, $S_2$, $S_3$ are measured in Qubit3, Qubit7, Qubit11, Qubit15, respectively. V is measured in Qubit13, V=1 states that an overflow has occurred. The reversible circuit of Design III is given in Figure 4.8.

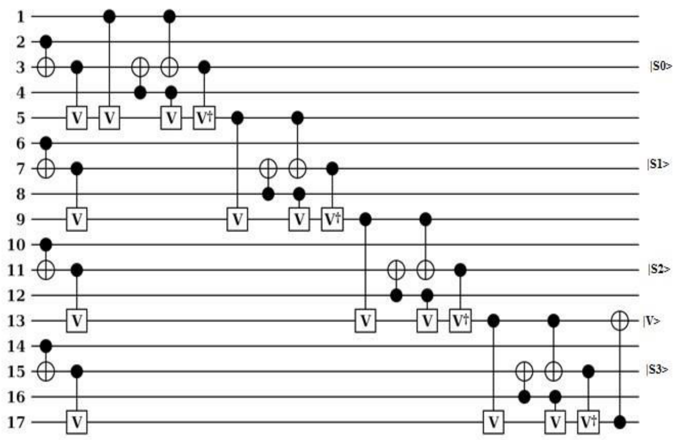


**Figure 4.8. Reversible 2's complement adder/subtractor with overflow detection logic - Design III**

The quantum value and delay of reversible n-bit two's complement adder/subtractor with overflow detection logic - style III is calculated as follows,

- The step one of style III, wants n CNOT gates operating in parallel. Hence, so this step has the quantum value of n and delay of one one

- The step a pair of has n Controlled V gates, operating in parallel. Thus, this step incorporates a quantum value of n and delay of one one

- The step three wants n SMG operating nonparallel, so causative to the quantum value of n(6-1)=5.n and delay of n(5-1)=4n Δ

- The step four has one CNOT gates. Hence, the quantum value is one and delay is one one

Thus, the overall quantum value of reversible n-bit two's complement adder/subtractor with overflow detection logic - style III is n+n+5n+1=7n +1, the propagation delay is 1Δ+1Δ+(4n) Δ +1 Δ =(4n+3)Δ. Hence, the quantum value of the four-bit circuit as given in Figure 4.8 is twenty nine and therefore the propagation delay is nineteen Δ.

## IX. SIMULATION RESULTS

The projected reversible circuits ar enforced victimisation QCviewer. Figure 4.9 offers the input screen to feature the numbers nine(1001) and 3(0011), and therefore the price of S is zero. Hence, the corresponding input for |A0>|B0>|S>|0>|A2>|B2>|S>|0>|A3>|B3>|S>|0>|A4>|B4>|S>|0> is given as |0>|1>|0>|1>|0>|1>|0>|0>|0>|0>|0>|0>|0>|0>|0>|1>|0> and therefore the outputs S0, S1, S2, S3 ar measured in Qubit3, Qubit7, Qubit11, Qubit15, severally. the complete output is 01011100101000110 as given in Figure four.10. measurement the desired Qubits, the output is 1100(12), Qubit15 is one, Qubit11 is one and

Qubit7 and Qubit3 ar zero.

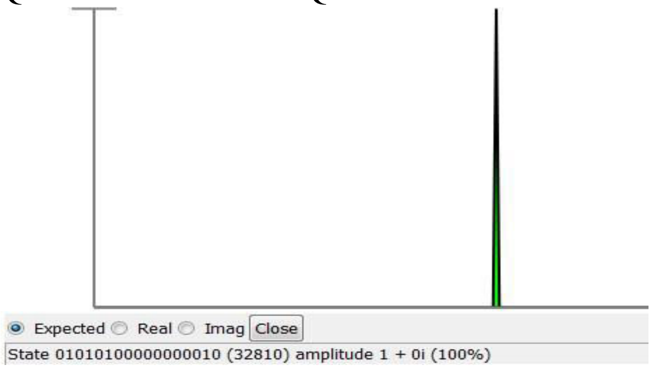

● Expected ○ Real ○ Imag Close
State 01010100000000010 (32810) amplitude 1 + 0i (100%)

**Figure 4.9. Sample input screen for reversible two's complement adder/subtractor - Design III**

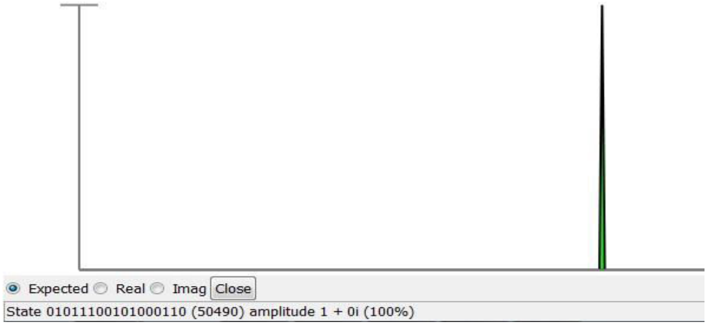

● Expected ○ Real ○ Imag Close
State 01011100101000110 (50490) amplitude 1 + 0i (100%)

**Figure 4.10. Sample output screen for reversible two's complement adder/subtractor - Design III**

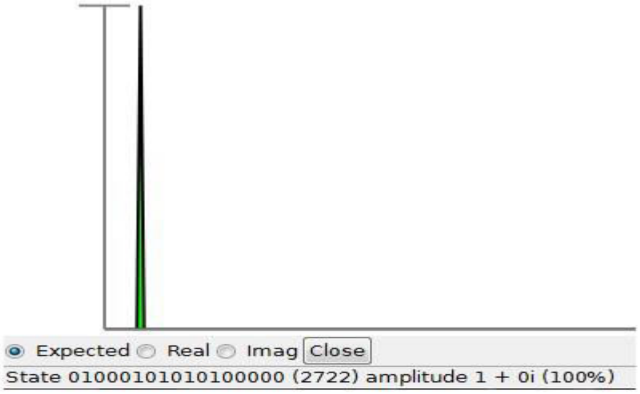

● Expected ○ Real ○ Imag Close
State 01000101010100000 (2722) amplitude 1 + 0i (100%)

**Figure 4.11. Sample input screen for reversible two's complement adder/subtractor with overflow detection logic - Design III**

Figure 4.11 provides the input screen to feature the numbers 6(0110) and 7(0111), and the worth of S is zero, thence the input is |0>|1>|0>|0>|0>|1>|0>|1>|0>|1>|0>|1>|0>|0>|0>|0>|0> and also the outputs S0, S1, S2, S3 square measure measured in Qubit3, Qubit7, Qubit11, Qubit15, severally. the whole output is 01100101111110100 as given in Figure four.12. activity the specified Qubits, the output is 1101 (13), Qubit15 is one, Qubit11 is one, Qubit7 is zero and Qubit3 is one, price of V is measured in Qubit13, which is 1, indicating Associate in Nursing overflow has occurred.

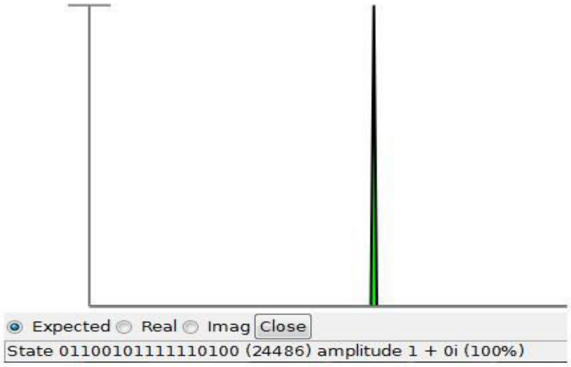

● Expected ○ Real ○ Imag Close
State 01100101111110100 (24486) amplitude 1 + 0i (100%)

**Figure 4.12. Sample output screen for reversible two's complement adder/subtractor with overflow detection logic - Design III**

## A. A. RESULTS AND DISCUSSIONS

SMG full adder gate has been utilized in implementation of the reversible 2's complement adder/subtractor and 2's complement adder/subtractor with overflow detection logic.

### A.1. Reversible Two's Complement Adder/Subtractor

3 styles for reversible four-bit two's complement adder/subtractor were simulated. style i used to be created mistreatment Toffoli and nuclear physicist Gates, style II mistreatment Peres gates and style III mistreatment SMG gate were designed and tested for all doable values. The distinction is highlighted supported the amount of reversible gates used, quantum price and delay, as shown in Table four.1. the amount of reversible gates required to construct style I is twenty, thus its quantum price is fifty two. the amount of reversible gates required to construct style II is twelve and its quantum price is thirty six. Whereas, variety of reversible gates required to construct style III is eight, thus its quantum price is twenty eight.

**Table 4.1. Comparison of reversible four-bit two's complement adder/subtractor**

| | No of Reversible gates | Quantum cost | Delay |
|---|---|---|---|
| D I | 20 | (13*n)=(13*4)=52 | (n+4) Δ= 8 Δ |
| D II | 12 | (9*n)=(9*4)=36 | (4*n+5) Δ=21 Δ |
| D III | 8 | (7*n)=(7*4)=28 | (4*n+2) Δ=18 Δ |

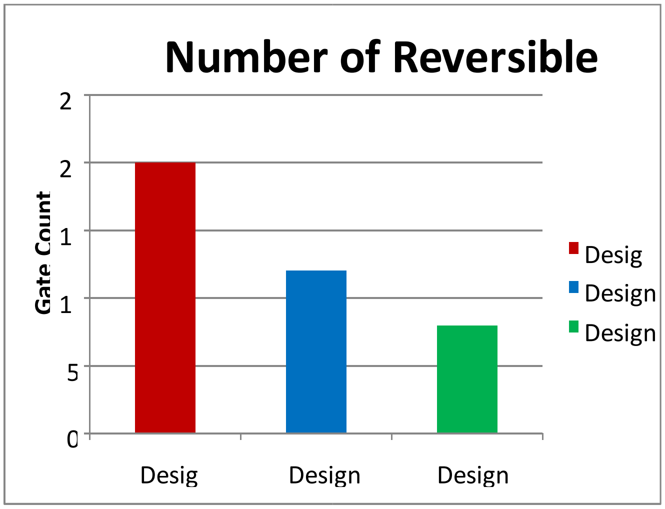


**Figure 4.13. Comparison of reversible four-bit two's complement adder/subtractor in terms of number of reversible gates**

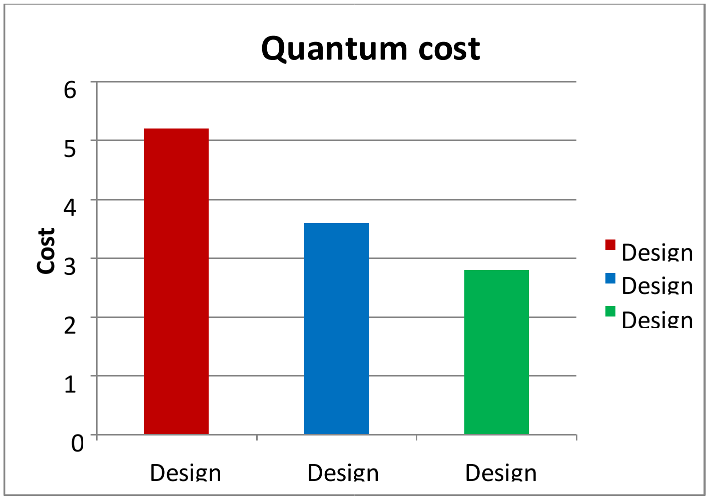


**Figure 4.14. Comparison of reversible four-bit two's complement adder/subtractor in terms of quantum cost**

Comparison in terms of range of reversible gates needed is shown pictorially in Figure four.13 and comparison in terms of quantum value is pictorially represented in Figure four.14. share of improvement once scrutiny style III enforced victimisation SMG Gate with style I enforced victimisation Toffoli and Richard Feynman gates is forty sixth and therewith of style II victimisation Peres Gate is twenty second. The projected circuits are simulated victimisation QCviewer and obtained results of simulation showed the proper operation of circuits. The projected reversible 4-bit two's complement adder/subtractor was additionally accustomed build AN n-bit two's complement adder/subtractor circuit.

### B. Reversible Two's Complement Adder/Subtractor with Overflow Detection Logic

Three designs for reversible four-bit two's complement adder/subtractor with overflow detection logic were simulated. Design I was constructed using Toffoli and Feynman Gates, Design II using Peres gates, and Design III using SMG gate and tested for all possible values. The contrast is being highlighted based on the number of reversible gates used, quantum cost, and delay, as shown in Table 4.2. The number of reversible gates needed to construct Design I is 21, hence its quantum cost is 53. The number of reversible gates needed to construct Design II is 13, hence its quantum cost is 37. Whereas, number of reversible gates needed to construct Design III is 9, hence its quantum cost is 29.

**Table 4.2. Comparison of reversible four bit two's complement adder/subtractor with overflow detection logic**

| | No of<br>Reversible gates | Quantum cost | Delay |
|---|---|---|---|
| D I | 21 | (13*n)+1=(13*4)+1=53 | (n+5) Δ= 9 Δ |
| D II | 13 | (9*n)+1=(9*4)+1=37 | (4*n+6) Δ=22 Δ |
| D III | 9 | (7*n)+1=(7*4)+1=29 | (4*n+3) Δ=19 Δ |

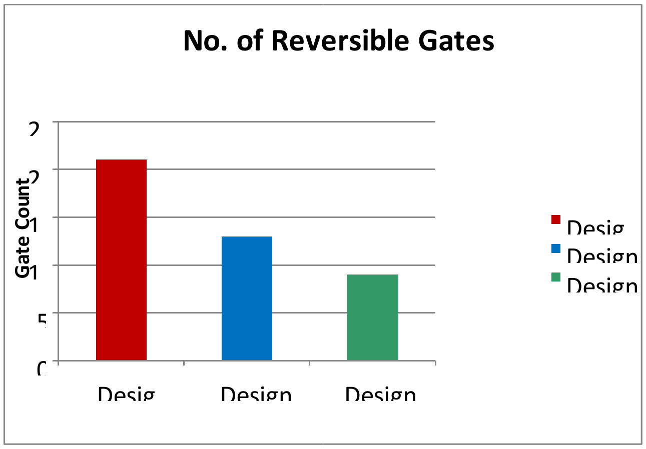


**Figure 4.15. Comparison of reversible four-bit two's complement adder/subtractor with overflow detection logic in terms of number of reversible gates**

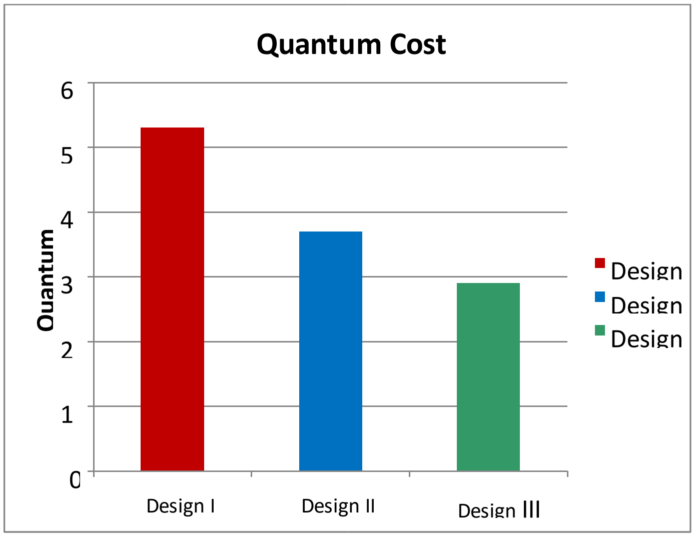


**Figure 4.16. Comparison of reversible four-bit two's complement adder/subtractor with overflow detection logic in terms of quantum cost**

Comparison in terms of number of reversible gates required is shown pictorially in Figure 4.15 and comparison in terms of quantum cost is pictorially depicted in Figure 4.16. Percentage of improvement when comparing Design III implemented using SMG gate with Design I implemented using Toffoli and Feynman gates is 45% and with that of Design II using Peres Gate is 21%. The proposed circuits have been simulated using QCviewer and obtained results of simulation showed the correct operation of circuits. The proposed reversible 4-bit two's complement adder/subtractor with overflow detection logic was also used to build an n-bit two's complement adder/subtractor with overflow detection logic circuit.

## X. CONVENTIONAL BINARY CODED DECIMAL ADDER

Binary coded decimal or BCD representation uses four bits for each decimal digit. Though 16 distinct digits can be represented in 4 bit binary numbers, only the first 10 numbers (0000 to 1001) are valid in BCD system. For example, 0101 0010 is a valid BCD number, that is equivalent to '52' in decimal system, but 0010 1011 is not valid as 1011 and cannot be represented with the help of one decimal digit.

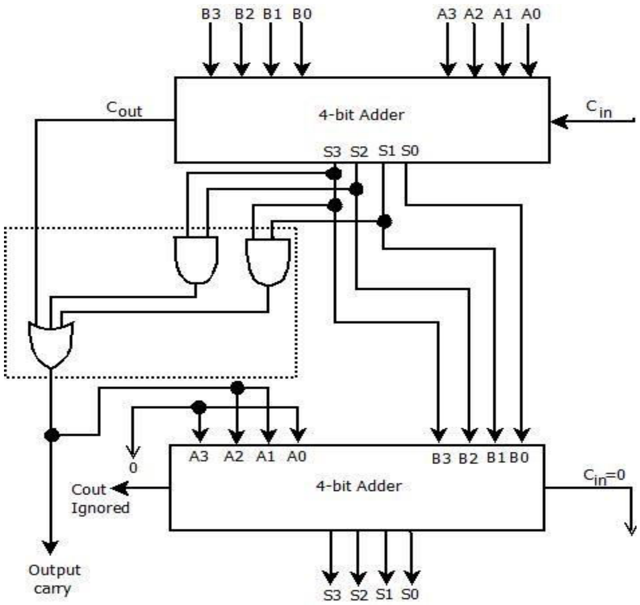


**Figure 5.1. Conventional BCD adder**

Conventional BCD adder is summarized as follows:

- Add two BCD numbers using ordinary binary addition
- If the four-bit sum is less than or equal to 9, no correction is needed
- If the four-bit sum is greater than 9 or if a carry is generated from the four-bit sum, the sum is invalid, then a correction is needed
- To correct the invalid sum, add 01102 to the four-bit sum

Figure 5.1 illustrates three parts of a BCD adder; 4 bit binary adder, over 9 detection unit and correction unit. The first part is a binary adder which works on two four bit BCD digits and a one bit carry input. In the second part, the over 9 detector recognizes, if the result of the first part is more than 9 or not. Finally, if the output of the detector unit is 1, the sum is added by $6(0110)_2$, else nothing is done.

## 5.2. REVERSIBLE BINARY CODED DECIMAL ADDER

In this section, three designs for reversible BCD adder are proposed, the circuits are evaluated in terms of quantum cost, delay, and transistor cost and are compared with the existing ones in the literature.

### *A. Binary Coded Decimal Adder - Design I*

In Design I, to implement reversible BCD adder the requirements are summarized as follows:

- 4-bit binary adder for initial addition, to perform this SMG full adder Gate was used
- Logic circuit to detect sum greater than 9, to perform this Detector unit designed using Toffoli Gate shown in Figure 5.2 was used
- One more four-bit adder to add 01102 to the sum, if sum is greater than 9 or if carry is 1, again SMG full adder Gate was used

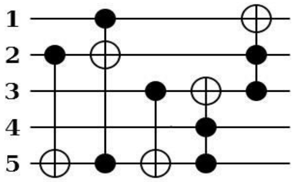


**Figure 5.2. Detector unit using Toffoli gate**

The quantum cost and delay of detector unit designed using Toffoli gate is calculated as follows:

- The step 1 of Figure 6.2, has 1 CNOT gate, thus this step has the quantum cost of 1 and delay of 1 Δ
- The step 2 has 1 CCNOT, thus this step has a quantum cost of 5 and delay of 1 Δ
- The step 3 has 1 CNOT gate, thus this step has the quantum cost of 1 and delay of 1 Δ
- The step 4 has 1 CCNOT gates, hence the quantum cost is 5 and delay is 1 Δ
- The step 4 has 1 CCNOT gates, hence the quantum cost is 5 and delay is 1 Δ

Thus, the total quantum cost of the detector unit designed using Toffoli gate is 1+5+1+5+5=17, the propagation delay is 1Δ+1Δ+1 Δ +1 Δ +1 Δ =5Δ.

The four-bit reversible circuit of Binary Coded Decimal Adder Design I is given in Figure 5.3, the circuit consists of 19 Qubits, the input is given in the following order, |cin>|$b_0$>|$a_0$>|0>|$b_1$>|$a_1$>|0>|$b_2$>|$a_2$>|0>|$b_3$>|$a_3$> |0>|0>|0>|1>|cin>|0>|0> and the outputs $C_0$, $S_3$, $S_2$, $S_1$, $S_0$ are measured in Qubit16, Qubit19, Qubit8, Qubit5, and Qubit2 respectively. The number of constant inputs in the circuit is 9 and number of garbage outputs are 14.

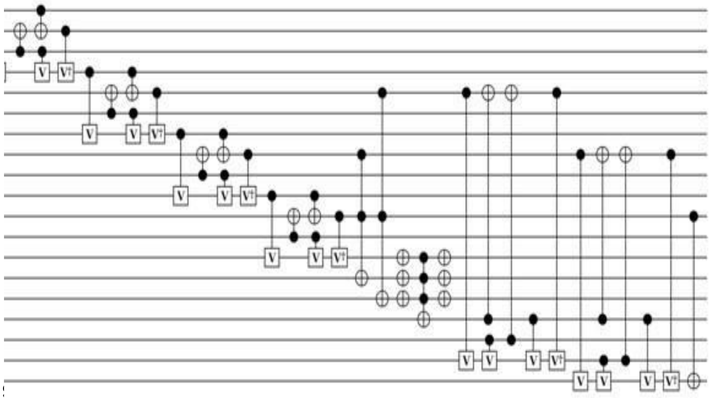


**Figure 5.3. Reversible BCD adder - Design I**

The quantum cost and delay of n-bit reversible BCD adder - Design I is calculated as follows:

- The step 1 of Design I has n Controlled V gate acting in parallel. Thus, this step has the quantum cost of n and delay of 1 Δ
- The step 2 has n SMG gate working in series. Thus, this step has a quantum cost of (6-1)n=5n and delay of (5-1)n Δ=4n Δe step 3 has 1 CCNOT gate. Thus, this step has the quantum cost of 5 and delay of 1 Δ
  - The step 4 has 1 CCNOT gates. Hence, the quantum cost is 5 and delay is 1 Δ
  - The step 5 has 3 NOT gates working in parallel. Hence, the quantum cost is 3 and delay is 1 Δ
  - The step 6 has 1 CCCNOT gate. Hence, the quantum cost is 13 and delay is 1 Δ
  - The step 7 has 3 NOT gates working in parallel. Hence, the quantum cost is 3 and delay is 1 Δ
  - The step 8 has Controlled V gate. Hence, the quantum cost is 1 and delay is 1 Δ
  - The step 9 has CNOT gate and Controlled V gate, acting parallel. Hence, the quantum cost is 2 and delay is 1 Δ
  - The step 10 has 1 CNOT gates. Hence, the quantum cost is 1 and delay is 1 Δ
  - The step 11 has Controlled V gate. Hence, the quantum cost is 1 and delay is 1 Δ

- The step 12 has Controlled V† gate. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 13 has Controlled V gate. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 14 has CNOT gate and Controlled V gate, acting parallel. Hence, the quantum cost is 2 and delay is 1 Δ
- The step 15 has 1 CNOT gates. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 16 has Controlled V gate. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 17 has Controlled V† gate. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 18 has 1 CNOT gates. Hence, the quantum cost is 1 and delay is 1 Δ

Thus, the total quantum cost of n-bit reversible binary coded decimal adder - Design I is n+5.n+5+5+3+13+3+1+2+1+1+1+1+2+1+1+1+1=6n+42, the propagation delay is 1Δ+(4n)Δ+16 Δ =(4n+17)Δ. Quantum cost of 4-bit reversible Binary Coded Decimal adder-Design I, the circuit as given in Figure 5.3 is 66, with a delay of 33 Δ and the number of control lines, s=44, hence its transistor cost is 8 * 44 = 352.

### *5.2.2. Binary Coded Decimal Adder - Design II*

In Design II, to implement reversible BCD adder, the requirements are :

- 4-bit binary adder for initial addition. To perform this, SMG full adder Gate was used
- Logic circuit to detect sum greater than 9. To perform this, Detector unit designed using Fredkin Gate as shown in Figure 5.4 was used
- One more four-bit adder to add 01102 in the sum, if sum is greater than 9 or carry is 1. Again SMG full adder Gate was used

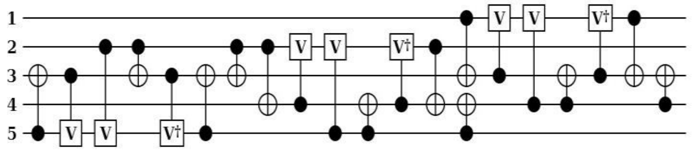


**Figure 5.4. Detector unit using Fredkin gate**

The quantum cost and delay of detector unit designed using Fredkin gate is calculated as follows:

- The step 1 of Figure 5.4 has 1 CNOT gate. Thus, this step has the quantum cost of 1 and delay of 1 Δ
- The step 2 has 1 Controlled V gate. Thus, this step has a quantum cost of 1 and delay of 1 Δ
- The step 3 has 1 Controlled V gate. Thus, this step has a quantum cost of 1 and delay of 1 Δ
- The step 4 has 1 CNOT gate. Thus, this step has the quantum cost of 1 and delay of 1 Δ
- The step 5 has 1 Controlled V† gate. Thus, this step has a quantum cost of 1 and delay of 1 Δ
- The step 6, 7, 8 has 1 CNOT gate each. Hence, this step has the quantum cost of 1 each and delay of 1 Δ each
- The step 9, 10 has 1 Controlled V gate each. Thus, this step has a quantum cost of 1 each and delay of 1 Δ each
- The step 11 has 1 CNOT gate. Thus, this step has the quantum cost of 1 and delay of 1 Δ
- The step 12 has 1 Controlled V† gate. Thus, this step has a quantum cost of 1 and delay of 1 Δ
- The step 13 has 1 CNOT gate. Thus, this step has the quantum cost of 1 and delay of 1 Δ
- The step 14 has 2 CNOT gate acting in parallel. Thus, this step has the quantum cost of 2 and delay of 1 Δ
- The step 15 has 1 Controlled V gate. Thus, this step has a quantum cost of 1 and delay of 1 Δ
- The step 16 has 1 Controlled V gate. Thus, this step has a quantum cost of 1 and delay of 1 Δ
- The step 17 has 1 CNOT gate. Thus this step has the quantum cost of 1 and delay of 1 Δ
- The step 18 has 1 Controlled V† gate. Thus, this step has a quantum cost of 1 and delay of 1 Δ
- The step 19 and step 20 has 1 CNOT gate each. Hence, this step has the quantum cost of 1 each and delay of 1 Δ each

Thus, the total quantum cost of the detector unit designed using Fredkin gate is 21, the propagation delay is 20 Δ.

The reversible circuit of Design II is given in Figure 5.5. The circuit consists of 17 Qubits. The input is given in the following order $|0>|cin>|b_0>|a_0>|0>|b_1>|a_1>|0>|b_2>|a_2>|0>|b_3>|a_3>|0>|0>|0>|0>$ and the outputs $C_0$, $S_3$, $S_2$, $S_1$, $S_0$ are measured in Qubit14, Qubit17, Qubit9, Qubit6, and Qubit3, respectively. The number of constant inputs in the circuit is 8 and number of garbage outputs are 12.

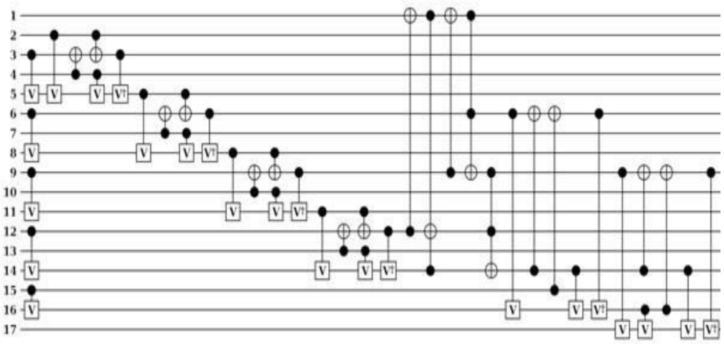


**Figure 5.5 Reversible BCD adder - Design II**

The quantum cost and delay of n-bit reversible BCD adder - Design II is calculated as follows:

- The step 1 of Design II has n+1 Controlled V gate acting in parallel. Thus, this step has the quantum cost of n+1 and delay of Δ
- The step 2 has n SMG gate working in series. Thus, this step has a quantum cost of (6-1)n=5n and delay of (5-1)n Δ=4n Δ
- The step 3 has 1 CNOT gate. Thus, this step has the quantum cost of 1 and delay of 1 Δ
- The step 4 has 1 CCNOT gates. Hence, the quantum cost is 5 and delay is 1 Δ
- The step 5 has 1 CNOT gate. Thus, this step has the quantum cost of 1 and delay of 1 Δ
- The step 6 has 1 CCNOT gates. Hence, the quantum cost is 5 and delay is 1 Δ
- The step 7 has 1 CCNOT gates. Hence, the quantum cost is 5 and delay is 1 Δ
- The step 8 has Controlled V gate. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 9 and step 10 has 1 CNOT gate each. Thus, this step has the quantum cost of 1 each and delay of 1 Δ each
- The step 11 has Controlled V gate. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 12 has Controlled $V^{\dagger}$ gate. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 13 has Controlled V gate. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 14 has CNOT gate and Controlled V gate, acting parallel. Hence, the quantum cost is 2 and delay is 1 Δ
- The step 15 has 1 CNOT gates. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 16 has Controlled V gate. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 17 has Controlled $V^{\dagger}$ gate. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 18 has 1 CNOT gates. Hence, the quantum cost is 1 and delay is 1 Δ

Thus, the total quantum cost of n-bit reversible binary coded decimal adder –Design II is 6.n+30, the propagation delay is 1Δ + (4n)Δ + 16 Δ =(4n+17) Δ. Quantum cost of 4-bit reversible Binary Coded Decimal adder - Design II, the circuit as given in Figure 5.5, is 54, with a delay of 33 Δ and the number of control lines is 45, hence its transistor cost is 8 * 45 = 360.

### *C. Binary Coded Decimal Adder - Design III*

In Design III, to implement reversible BCD adder the requirements are :

- 4-bit binary adder for initial addition, to perform this SMG full adder Gate is used
- Binary to BCD converter using Toffoli and Peres Gates as shown in Figure 5.6 is used

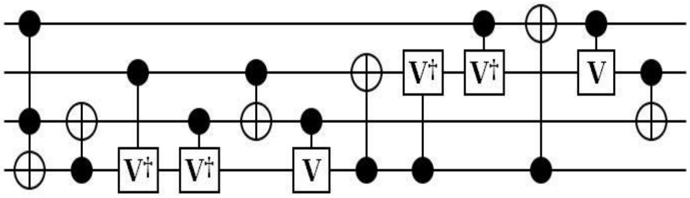


**Figure 5.6 Binary to BCD converter using Toffoli and Peres gates**

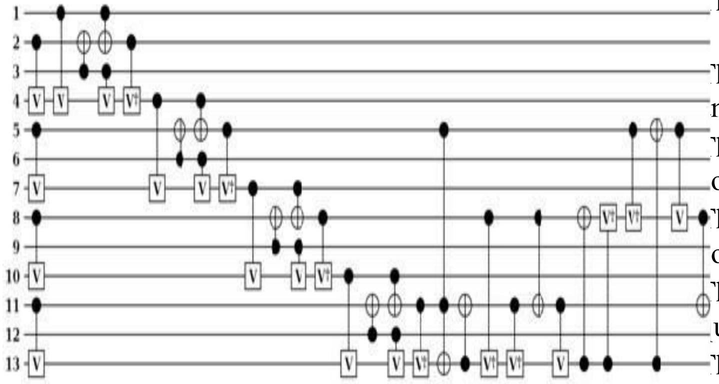


**Figure 5.7 Reversible BCD adder - Design III**

The reversible circuit of Design III is given in Figure 5.7. The circuit consists of 13 Qubits. The input is given in the following order |cin>|b$_0$>|a$_0$>|0>|b$_1$>|a$_1$>|0>|b$_2$>|a$_2$>|0>|b$_3$>|a$_3$>|0> and the outputs C$_0$, D$_3$, D$_2$, D$_1$, D$_0$ are measured in Qubit13, Qubit11, Qubit8, Qubit5, and Qubit2, respectively. The number of constant inputs in the circuit is 4 and number of garbage outputs are 8. The quantum cost and delay of n-bit reversible BCD adder- Design III is calculated as follows:

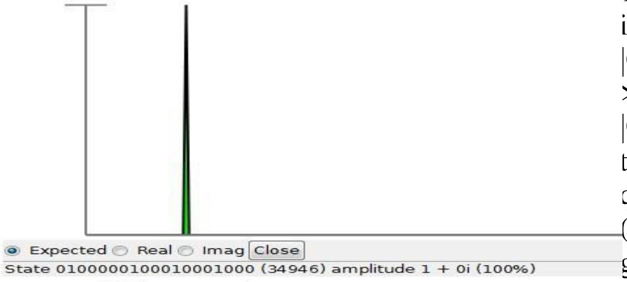
- The step 1 of Design III has n Controlled V gate acting in parallel. Thus, this step has the quantum cost of n and delay of 1 Δ
- The step 2 has n SMG gate working in series. Thus, this step has a quantum cost of (6-1)n=5n and delay of (5-1)n Δ=4n Δ
- The step 3 has 1 CCNOT gate. Thus, this step has the quantum cost of 5 and delay of 1 Δ
- The step 4 has 1 CNOT gates. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 5 has Controlled $V^{+}$ gate. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 6 has Controlled $V^{+}$ gate. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 7 has 1 CNOT gate. Thus, this step has the quantum cost of 1 and delay of 1 Δ
- The step 8 has Controlled V gate. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 9 has 1 CNOT gates. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 10 has Controlled $V^{+}$ gate. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 11 has Controlled $V^{+}$ gate. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 12 has 1 CNOT gate. Thus, this step has the quantum cost of 1 and delay of 1 Δ
- The step 13 has Controlled V gate. Hence, the quantum cost is 1 and delay is 1 Δ
- The step 14 has 1 CNOT gates. Hence, the quantum cost is 1 and delay is 1 Δ

Thus, the total quantum cost of n-bit reversible binary coded decimal adder –Design III is 6n+16, the propagation delay is (4n) Δ+13 Δ = (4n+13) Δ. Quantum cost of 4-bit reversible Binary Coded Decimal adder Design III, the circuit as given in Figure 5.7, is 40, with a delay of 29 Δ and the number of control lines is 37, hence its transistor cost of 8 * 37 = 296.

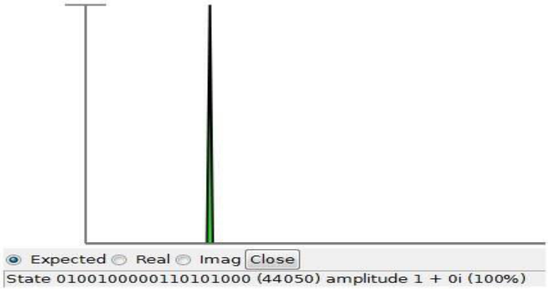
## XI. SIMULATION RESULTS

In this chapter, three designs for reversible BCD adder were designed. To add 8(1000) and 5(0101) using Design I, the general input is |cin>|b$_0$>|a$_0$>|0>|b$_1$>|a$_1$>|0>|b$_2$>|a$_2$>|0>|b$_3$>|a$_3$>|0>|0>|0>|1>|cin>|0>|0>, the actual input is |0>|1>|0>|0>|0>|0>|0>|1>|0>|0>|0>|1>|0>|0>|0>|1>|0>|0>|0> and the measured output is 0100100000110101000, where $q_{16}$=1, $q_{19}$=0, $q_8$=0, $q_5$=1, $q_2$=1, which is equivalent to 0001 0011 (13).

Figure 5.8 gives the input screen and Figure 5.9 gives the output screen for Design I.

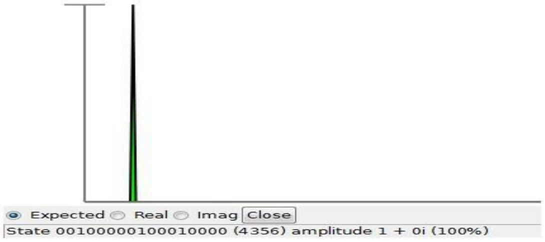


State 0100000100010001000 (34946) amplitude 1 + 0i (100%)



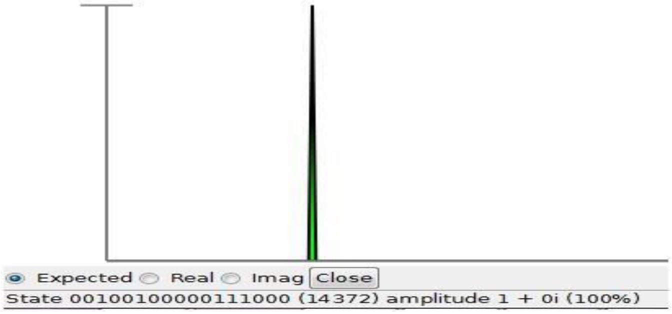
State 0100100000110101000 (44050) amplitude 1 + 0i (100%)

**Figure 5.8. Sample input screen - BCD adder Design I**



State 00100000100010000 (4356) amplitude 1 + 0i (100%)

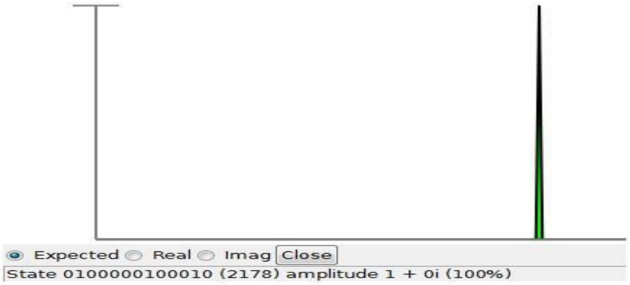
**Figure 5.10 Sample input screen - BCD adder design II**



State 00100100000111000 (14372) amplitude 1 + 0i (100%)

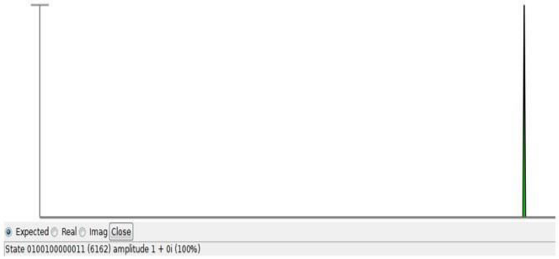
**Figure 5.11 Sample output screen - BCD adder design II**

To add 8(1000) and 5(0101) using Design II, the general input is $|0>|cin>|b_0>|a_0>|0>|b_1>|a_1>|0>|b_2>|a_2>|0>|b_3>|a_3>|0>|0>|0>|0>$ and the actual input is |0>|0>|1>|0>|0>|0>|0>|0>|1>|0>|0>|0>|1>|0>|0>|0>|0> and the measured output is 00100100000111000, where $q_{14}$=1, $q_{17}$=0, $q_9$=0, $q_6$=1, $q_3$=1, which is equivalent to 0001 0011 (13). Figure 5.10 gives the input screen and Figure 5.11 gives the output screen for Design II.



State 0100000100010 (2178) amplitude 1 + 0i (100%)

**Figure 5.12 Sample input screen - BCD adder design III**



State 0100100000011 (6162) amplitude 1 + 0i (100%)

**Figure 5.13 Sample output screen - BCD adder design III**

To add 8(1000) and 5(0101) using Design III, the general input is $|cin>|b_0>|a_0>|0>|b_1>|a_1>|0>|b_2>|a_2>|0>|b_3>|a_3>|0>$ and the actual input is 0>|1>|0>|0>|0>|0>|0>|1>|0>|0>|0>|1>|0> and the measured output is 0100100000011; where $q_{13}$=1, $q_{11}$=0, $q_8$=0, $q_5$=1, $q_2$=1, which is equivalent to 0001 0011 (13). Figure 5.12 gives the input screen and Figure 5.13 gives the output screen for Design III.

## XII. RESULTS AND DISCUSSIONS

Table 5.1 shows the comparison results of the proposed circuits with the existing ones in the literature in terms of quantum cost, delay, and transistor cost and also pictorially represented in Figure 5.14. The delay and transistor cost of are not known. It can be observed from Table 5.1, the proposed designs are better than the other designs in terms of quantum cost, delay, and transistor cost. The Quantum cost of Design III is the optimized one, compared to all other existing designs. The Transistor cost is calculated for the first time in the Literature. The percentage of improvement when comparing the designed circuits with Thapliyal & Ranganathan, with respect to quantum cost and delay is shown in Table 5.2.

**Table 5.1. Comparison of four-bit reversible BCD adders**

| Designs | Quantum Cost | Delay Δ | Transistor Cost |
|---|---|---|---|
| | | | |
| (Biswas et al 2008) | 55 | - | - |
| (Thomsen & Gluck 2008) | 169 | - | - |
| (Mohammadi et al 2009) | 103 | - | - |
| (Thapliyal & Ranganathan 2011 | 70 | 57 | - |
| Design I | 66 | 33 | 352 |
| Design II | 54 | 33 | 360 |
| Design III | 40 | 29 | 296 |

**Table 5.2. Comparison of reversible BCD adders - improvement ratio**

| | Quantum Cost | Delay |
|---|---|---|
| Existing Circuit<br>(Thapliyal & Ranganathan 2011) | 70% | 57 |
| Designed Circuit – Design III | 40% | 31 |
| Improvement Ratio | 42% | 45% |

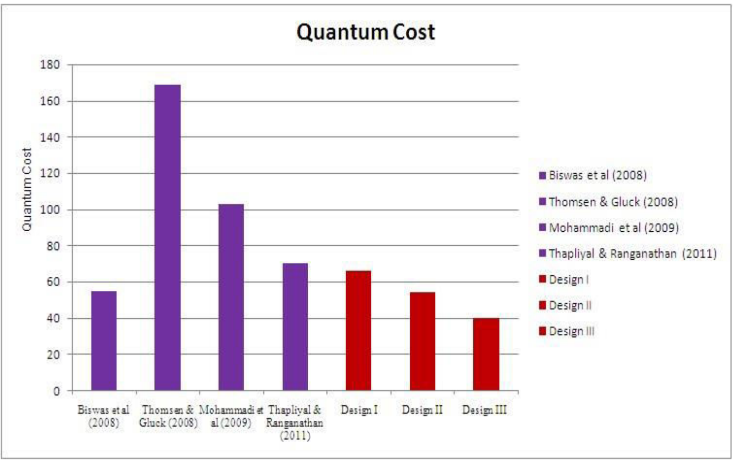


**Figure 5.14 Comparison of BCD adders in terms of quantum cost**

In this chapter, three designs, Design I, Design II and Design III were designed to implement reversible Binary Coded Decimal Adder. In Design I, SMG full adder Gate and detector unit designed using Toffoli Gates were used, whereas in Design II, SMG full adder Gate and detector unit designed using Fredkin Gates were used. In Design III, SMG full adder Gate along with Binary to BCD converter was used.

The circuits were evaluated in terms of quantum cost, delay, and transistor cost. Transistor cost was calculated for the first time in literature. It has been shown that the proposed circuits are more optimal in terms of quantum cost, delay, and transistor cost. The proposed reversible Binary Coded Decimal Adder was also used to build reversible n-bit Binary Coded Decimal adder.

## CONCLUSION

First, a reduced implementation of the four X four reversible full adder gate utilizing quantum primitive gates was designed, that was Associate in Nursing improvement in terms of quantum value, delay, and electronic transistor value over the previous implementations mistreatment Toffoli & Richard Feynman gates and Peres gate.

The new gate was able to turn out all the specified logical outputs, AND, NAND, OR, NOR, XOR, XNOR with none extra logical structures. Next, style and implementation of reversible two’s complement adder/subtractor was designed and simulated. 3 styles were projected, 1st one supported Toffoli and Richard Feynman gates, second supported the Peres gate, and therefore the third one supported the new SMG gate. The styles were compared to quantum value, delay, and electronic transistor value and has been verified that the reversible circuit designed mistreatment SMG gate is perfect. Next, style and implementation of reversible two’s complement adder/subtractor with overflow detection logic has been simulated. 3 styles were projected, 1st style supported Toffoli and Richard Feynman gates, second style supported Peres gate, and third style supported SMG gate.

Next, the new gate was utilized in the implementation of reversible binary coded decimal adder and 3 styles were simulated. These new BCD adder styles verified to be advantageous to antecedently printed add implementations that favor low quantum value, delay, and electronic transistor value. Next, classical algorithms for graph theory issues were thought of and quantum techniques were applied to those algorithms for finding them in an exceedingly quantum laptop, and therefore reason their question complexness. Quantum question complexness was thought of each in an exceedingly classical and in an exceedingly quantum system to work on median of a graph and center of a graph.

A key soul in an exceedingly competitive market place is client satisfaction. client satisfaction is taken into account as a vicinity of business strategy. The associated project parameters ought to be pro-actively managed and therefore the project outcome has to be foreseen by a technical manager. Technical managers chiefly focus solely on the tip state. Instead their focus ought to be proactively managing and minimizing defect escape at the first stages. during this work, sensible relevance of prognostic models and use of those models in an exceedingly quantum project to predict system testing defects, therefore serving to to scale back residual defects were mentioned.

Quality of a reversible circuit is typically calculable by Gate Count (GC) or by a metric known as Quantum value (QC). a lot of less effort has been dedicated to decrease of QC in reversible circuits. For any reversible perform there exist several reversible circuits implementing it. Thus, a value perform must be outlined to guage the standard of a circuit. Usually, additive value functions ar applied. Therefore, adding a gate to a circuit ends up in increasing its value. a standard follow in synthesis of reduced quantum value reversible circuits is to 1st realize a gate count best circuit so map the ensuing circuit into quantum gates. However, as showed during this work, this approach doesn't cause stripped quantum value circuits. It needs considering circuits having larger range of gates than the stripped one to be able to realize precise stripped quantum value circuits. analysis bestowed here will be extended in numerous directions. the present irreversible technologies dissipates ton of warmth and may cut back the lifetime of the circuit, however reversible logic operations don't erase (lose) data and dissipates a lot of less heat. The SMG full adder gate has been utilized in implementation of the reversible circuits. The quantum value, delay, and electronic transistor value were additionally calculated for the projected circuits. The projected circuits are simulated mistreatment Quantum laptop aper, QCviewer, QuIDDPro supported the prevailing follow and therefore obtained the results of simulation, that showed the attainable right operation of circuits used for quantum computers. The projected reversible 4-bit circuits was generalized for reversible n-bit reversible circuits, therefore the aim would be achieved. this may show the approach for the planning of a reversible logic circuit processor unit. Hence, it will be ended that days don't seem to be too behind to realize the target, in order that a whole reversible circuit processor may be designed to realize the value reduction.

**References:**

1. Ambainis, A 2003, ‘Quantum Walks and their Algorithmic applications’, International Journal of Quantum Information, vol. 1, no. 4, pp. 507-518.

2. Amiri, PK 2003, ‘Quantum Computers’, IEEE Potentials, vol. 21, no. 5, pp. 6-9.

3. Aradhya, RHV, Chinmaye, R & Muralidhara, KN 2012, ‘Design, Optimization and Synthesis of Efficient Reversible Logic Binary Decoder’, International Journal of Computer Applications, vol. 46, no. 6, pp. 45-51.

4. Barnum, H, Bernstein, HJ & Spector, L 2000, ‘Quantum Circuits for OR and AND of ORs’, Journal of Physics A: Mathematical and General, vol. 33, pp. 8047-8057.

5. Bennett, CH 1973, ‘Logical Reversibility of Computation’, IBM Journal of Research and Development, vol. 17, pp. 525-532.

6. Bhagyalakshmi, HR & Venkatesha MK 2010b, ‘Optimized reversible BCD adder using new reversible logic gates’, Journal of Computing, vol. 2, no. 2, pp. 28-32.

7. Bhagyalakshmi, HR & Venkatesha, MK 2010a, ‘An Improved Design of a Multiplier using Reversible Logic Gates’, International Journal of Engineering Science and Technology, vol. 2, no. 8, pp. 3838-3845.

8. Biswas, AK, Hasan, MM, Chowdhury, AR & Babu, HMH 2008, ‘Efficient approaches for designing Reversible Binary Coded Decimal adders’, Microelectronics Journal, vol. 39, no. 12, pp. 1693-1703.

9. Boroumand, S 2012, ‘A Novel Nanometric Reversible Four-bit Signedmagnitude Adder/Subtractor’, Life Science Journal, vol. 9, no. 3, pp. 1646-1655.

10. Boyer, M, Brassard, G, Hoyer, P & Tapp, A 1996, ‘Tight bound on quantum searching’, Progress of Physics, vol. 46, no. 4-5, pp. 493-505.

11. Breyfogle, FW 2003, ‘Implementing Six Sigma: Smarter Solutions Using Statistical Methods’, John Wiley & Sons, 2nd Edition.

12. Chuang, IL & Modha, DS 2000, ‘Reversible Arithmetic Coding for Quantum Data Compression’, IEEE transactions on Information theory, vol. 46, no. 3, pp. 1104- 1116.

13. Clark, J & Holton, DA 1995, A first look at Graph Theory, Allied Publishers Ltd.

14. Coghlan, D & Brannick, T 2002, Doing Action Research in Your Own Organization, Sage, London, UK.

15. Desoete, B & Vos, AD 2002, 'A reversible carry-look-ahead adder using control gates', Integration, The VLSI Journal, vol. 33, no. 1/2, pp. 89-104.

16. Drechsler, R & Wille, R 2012, 'Reversible Circuits: Recent Accomplishments and Future Challenges for an Emerging Technology', Progress in VLSI Design and Test Lecture notes in Computer Science, vol. 7373, pp. 383-392.

17. Dueck, GW & Miller, DM (Eds.), 2013, 'Garbage-Free Reversible Constant Multipliers for Arbitrary Integers', Reversible Computing 2013, LNCS 7948, pp. 70-83.

18. Durr, C, Heiligman, M, Hoyer, P & Mhalla, M 2004, 'Quantum Query Complexity of Some Graph Problems', Automata, Languages and Programming Lecture Notes in Computer Science, vol. 3142, pp. 481493.

19. Ekert, A & Jozsa, R 1996, 'Quantum computation and Shor's Factoring algorithm', Review of Modern Physics, vol. 68, no. 3, pp. 733-753.

20. Feynman, R 1982, 'Simulating Physics with Computers', International Journal of Theoretical Physics, vol. 21, no. 6&7, pp. 467-488.

21. Fortnow, L 2003, 'One complexity theorist's view of Quantum Computing', Theoretical Computer Science, vol. 292, pp. 597-610.

22. Fredkin, E & Toffoli, T 1982, 'Conservative Logic', International Journal of Theoretical Physics, vol. 21, no. 3-4, pp. 219-253.

23. Grover, LK 1997, 'Quantum Mechanics Helps in Searching for a Needle in a Haystack', Physical Review Letters, vol. 79, no. 2 pp. 325328.

24. Gummeson, E 2010, 'Qualitative Methods in Managing Research', SAGE, Thousand Oakes, California, USA.

25. Haghparast, M & Navi, K 2008, 'Design of a Novel Fault Tolerant Reversible Full Adder for Nanotechnology Based Systems', World Applied Sciences Journal, vol. 3, no. 1, pp. 114-118.

26. Haghparast, M, Jassbi, SJ, Navi, K & Hashemipour, O 2008, 'Design of a Novel Reversible Multiplier Circuit using HNG Gate in Nanotechnology', World Applied Sciences Journal, vol. 3, no. 6, pp. 974-978.

27. Hans, DR, Hams, AH, Michielsen, K & Koen, DR 2000, 'Quantum Computer Emulator', Computer Physics Communications, Available from: http://www.rugth30.phys.rug.nl/qce [November 2010]

28. Hari, SKS, Shroff, S, Mahammad, SN & Kamakoti, V 2006, 'Efficient Building Blocks for Reversible Sequential Circuit Design', 49th IEEE International Midwest Symposium on Circuits and Systems, pp. 437441.

29. Horowitz, E & Sahni, S 1990, Fundamentals of Data Structures, Galgotia Publications, New Delhi.

30. Horowitz, E, Sahni, S & Rajasekaran, S 2004, Fundamentals of Computer Algorithms, Galgotia Publications, New Delhi.

31. Hossain, MS, Rakib, MRH, Rahman, MM, Hossain, ASMD & Hasan, MM 2011, 'A New Design Technique of Reversible BCD Adder based on NMOS with Pass Transistor Gates', International Journal of VLSI design & communication Systems, vol. 2, no. 4, pp. 15-26.

32. Islam, MS, Rahman, MM, Begum, Z & Hafiz, MZ 2009, 'Efficient Approaches for Designing Fault Tolerant Reversible Carry Look-Ahead and Carry Skip Adders', MASAUM Journal of Basic and Applied Sciences, vol. 1, no. 3, pp. 354-360.

33. James, G, Witten, D, Hastie & Tibshirani, R 2013, 'An Introduction to Statistical Learning with Applications in R', vol. 103 of Springer Texts in Statistics, Springer, New York, NY, USA.

34. Jeffery, S, Kothari, R, Gall, FL & Magniez, F 2015, 'Improving Quantum Query Complexity of Boolean Matrix Multiplication Using Graph Collision', Algorithmica, Springer, DOI: 10.1007/s00453-0159985-x

35. Johnson, M & Gustafsson, A 2010, 'Improving Customer Satisfaction, Loyalty and Profit', Jossey-Bass.

36. Kan, SH 2002, Metrics and Models in Software Quality Engineering, 2nd Edition.

37. Kasse, T 2008, Practical Insight 14 into CMMI, Artech House, 2nd Edition.

38. Krishnaveni, D & Priya, GM 2011, 'A Novel design of Reversible Serial and Parallel Adder/Subtractor', International Journal of Engineering Science and Technology, vol. 3, no. 3, pp. 2280-2288.

39. Kulpa, MK & Johnson, KA 2008, 'Interpreting the CMMI: A Process Improvement Approach', 2nd edition.

40. Landauer, R 1961, 'Irreversibility and Heat Generation in the Computing Process', IBM Journal of Research and Development, vol. 5, pp. 183-191.

41. Leach, DP, Malvino, AP & Saha, G 2006, Digital Principles and Applications, Tata McGraw-Hill, New Delhi.

42. Manin, YI 1999, 'Classical Computing, Quantum Computing and Shor's factoring algorithm', Seminaire Bourbaki, vol. 41, pp. 375-404.

43. Manjula, GS & Prabhakar, R 2008, 'Quantum Query Complexity to determine radius of a Graph', i-manager's Journal on Software Engineering, vol. 2, no. 3, pp. 64-69.

44. Maslov, D & Dueck, GW 2004, 'Reversible Cascades with Minimal Garbage', IEEE Transactions on Computer-aided design of integrated circuits and systems, vol. 23, no. 11, pp. 1497-1509.

45. Maslov, D 2005, Reversible Logic Synthesis Benchmarks Page, Available from: <http://www.cs.uvic.ca/~dmaslov>.

46. McConnell, B & Huba, J 2013, Creating Customer vangelists: How Loyal Customers Become a Volunteer Sales, Kaplan Professional.

47. Meglicki, Z 2002, Introduction to Quantum Computing (M743), Indiana University.

48. Mermin, ND 2003, 'From Cbits to Qbits: Teaching Computer Scientists Quantum Mechanics', American Journal of Physics, vol. 71, no. 1, pp. 23-30.

49. Mishra, UK, Sudhan, KH & Gupta, S 2011, 'Establishing process performance baselines and models for statistical control of software projects', Proceedings of the ASCNT, Centre for Development of Advanced Computing (CDAC), Noida, India, 2011.

50. Mohammadi, M, Haghparast, M, Eshghi, M & Navi, K 2009, 'Minimization and Optimization of reversible BCD-Full adder/subtractor using Genetic Algorithm and don't care concept', International Journal of Quantum Information, vol. 7, no. 5, pp. 969989.

51. Moore, GE 1965, 'Cramming More Components onto Integrated Circuits', Reprinted from Electronics Magazine, vol. 38, no. 8, pp. 114117.

52. Morris, M 2001, Digital Design, Prentice Hall, 3rd Edition.

53. Nannarelli, A & Lang, T 1999, 'Low-Power Divider', IEEE Transaction on Computers, vol. 48, no. 1, pp. 2-14.

54. Nielsen, MA & Chuang, IL 2003, Quantum Computation and Quantum Information, Cambridge University Press, Cambridge.

55. Niknafs, A & Mohammadi, M 2013, 'Synthesis and optimization of multiple-valued combinational and sequential reversible circuits with don't cares', Integration, The VLSI Journal, vol. 46, pp. 189-196.

56. Oliveira, DS & Ramos, RV 2007, 'Quantum bit string comparator: circuits and applications', Quantum Computers and Computing, vol. 7, no. 1, pp. 17-26.

57. Parent, A & Parker, J 2012, 'QCviewer0.8', Available from: <http://qcirc.iqc.uwaterloo.ca> [May 2012].

58. Peppers, D & Rogers, M 2004, Managing Customer Relationships: A Strategic Framework, John Wiley & Sons, New York, NY, USA.

59. Peres, A 1985, 'Reversible Logic and Quantum Computers', Physical Review A, vol. 32, no. 6, pp. 3266-3276.

60. Perkowski, M 2000, 'Reversible computation for beginners', Lecture series, Portland state university, Available from: <http://www.ee.pdx.edu/_mperkowsS>.

61. Perkowski, M, Jozwiak, L, Kerntopf, P, Mishchenko, A, Al-Rabadi, Coppola, A, Bulleer, A, Song, X, Khan, MMHA, Yanushkevich, S, Shmerko, VS & Jeske, C 2001, 'A general decomposition for reversible logic', Proceedings of R M 2001.

62. Prashant, RY & Sujata, SC 2011, 'Introduction to Reversible Logic gates and its Application', International Journal of Computer Applications, pp. 5-9.

63. Praveen, B & Vinay, KSB, 2012, '$2^n$:1 Reversible Multiplexer and its Transistor Implementation', International Journal of Computational Engineering Research, vol. 2, no. 1, pp. 182-189.

64. Preskill, J 1997, 'Quantum Computing: Pro and Con', Proceedings of the Royal Society, vol. 454, no. 1969, pp. 469-486.

65. Rabaey, JM & Pedram, M (eds.) 1995, 'Low Power Design Methodologies', The Springer International Series in Engineering and Computer Science, vol. 336, Boston, Kluwer Academic Publisher/Springer.

66. Rangaraju, HG, Venugopal, U, Muralidhara, KN & Raja, KB 2010, 'Low Power Reversible Parallel Binary Adder/Subtractor', International Journal of VLSI design & Communication Systems, vol.1, no. 3, pp. 1-12.

67. Rekha, KJ, Shahana, TK, Jacob, KP & Sasi, S 2007, 'Performance Analysis of Reversible Fast Decimal Adders', Proceedings of the World Congress on Engineering and Computer Science, WCECS, October 24th-26th, 2007, San Francisco, USA.

68. Remenyi, D, Money, A & Twite, A 2012, Effective Measurement & Management of IT Costs & Benefits, Butterworth-Heinemann, Oxford, UK.

69. Rentergem, YV & Vos, AD 2005, 'Optimal Design of a Reversible Full Adder', International Journal of Unconventional Computing, vol. 1, pp. 339-355.

70. Rieffel, EG & Polak, W 2000, 'An Introduction to Quantum Computing for Non-Physicists', ACM Computing Surveys, vol. 32, no. 3, pp. 300335.

71. Saeedi, M & Markov, IL 2013, 'Synthesis and optimization of reversible circuits - a survey', ACM Computing Surveys, vol. 45, no. 2, pp. 1-34.

72. Saeedi, M, Sedighi, M & Zamani, MS 2010, 'A library-based synthesis methodology for reversible logic', Microelectronics Journal, vol. 41, pp. 185-194.

73. Saiful, I & Rafiqul, I 2005, 'Minimization of Reversible Adder Circuits', Asian Journal of Information Technology, vol. 4, no. 12, pp. 1146-1151.

74. Saiful, I, Rahman, MM, Begum, Z & Hafiz, MZ 2010, 'Realization of a Novel Fault Tolerant Reversible Full Adder Circuit in Nanotechnology', The International Arab Journal of Information Technology, vol. 7, no. 3, pp. 317-323.

75. Self, B, Roche, G & Hill, N 2000, Customer Satisfaction Measurement for ISO 9000: 2000.

76. Shende, VV, Bullock, SS & Markov, IL 2006, 'Synthesis of quantumlogic circuits', IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 25, no. 6, pp. 1000-1010.

77. Shende, VV, Prased, AK, Markov, IL & Hayes, JP 2003, 'Synthesis of Reversible Logic Circuits', IEEE Transactions on Computer-aided design of integrated circuits and systems, vol. 22, no. 6, pp. 710-722.

78. Shor, PW 1994, 'Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer', SIAM Journal of Computing, vol. 26, no. 5, pp. 1484-1509.

79. Smolin, JA & DiVincenzo, DP 1996, 'Five Two-bit Quantum Gates Are Sufficient to Implement the Quantum Fredkin Gate', Physical Review A, vol. 53, no. 4, pp. 2855-2856.

80. Steane, A M 1997, 'Quantum Computing', SIAM Journal on Computing, vol. 26, no. 5, pp. 1484-1509.

81. Stoddard, RW 2007, CMMI process performance models and reliability, IEEE Reliability Society Annual Technology report, Carnegie Mellon University.

82. Stoddart, B & Lynas, R 2010, 'A Virtual Machine for supporting Reversible Probabilistic Guarded Command Languages', Electronic Notes in Theoretical Computer Science, vol. 253, pp. 33-56.

83. Stutzke, RD 2005, Estimating Software - Intensive Systems: Projects, Products, and Processes, Addison-Wesley.

84. Sudharshan, M, Bhagyalakshmi, HR &Venkatesha, MK 2012, 'Novel Design of one digit high speed Carry select BCD Subtractor using Reversible logic gates', International Journal of Emerging Technology and Advanced Engineering, vol. 2, no. 7, pp. 376-381.

85. Sutikno, T 2010, 'An optimized Square Root Algorithm for implementation in FPGA Hardware', Telkomnika, vol. 8, no. 1, pp. 1-8.

86. Syamala, Y & Tilak, AVN 2011, 'Reversible Arithmetic Logic Unit', 3rd International Conference on Electronics Computer Technology, 8th10th, April 2011.

87. Thapliyal, H & Ranganathan, N 2009, 'Design of Efficient Reversible Binary Subtractors Based on A New Reversible Gate', Proceedings of the IEEE Computer Society Annual Symposium on VLSI, pp. 229-234.

88. Thapliyal, H & Ranganathan, N 2011, 'A New Reversible Design of BCD Adder', Design, Automation & Test in Europe, pp. 1-4.

89. Thapliyal, H & Zwolinski, M 2006, 'Reversible Logic to Cryptographic Hardware: A New Paradigm', 49th IEEE International Midwest Symposium on Circuits and Systems, vol. 1, pp. 342-346.

90. Thapliyal, H, Srinivas, MB & Mandalika, B 2005, 'Novel design and reversible logic synthesis of multiplexer based full adder and multipliers', Forty Eighth Midwest Symposium on Circuits and Systems, vol. 2, pp. 1593-1596.

91. Thomsen, MK & Gluck, R 2008, 'Optimized reversible binary-coded decimal adders', Journal of Systems Architecture, vol. 54, no. 7, pp. 697-706.

92. Thomsen, MK, Gluck, R & Axelsen, HB 2010, 'Reversible arithmetic logic unit for quantum arithmetic', Journal of Physics A: Mathematical and Theoretical, vol. 43, pp. 1-10.

93. Toffoli, T 1980, Reversible Computing' Technical Report MIT/LCS/TM-151.

94. Vandana, S, Singh, OP, Mishra, GR & Tiwari, RK 2013, 'Design of a 4-bit 2's Complement Reversible Circuit for Arithmetic Logic Unit Applications', IJCA Special Issue on International Conference on Communication, Computing and Information Technology, vol. ICCCMIT, no. 2, pp. 1-5, Published by Foundation of Computer Science, New York, USA.

95. Vedral, V & Plenio, MB 1998, 'Basics of Quantum Computation', Progress in Quantum Electronics, Elsevier Science, vol. 22, no. 1, pp. 1-39.

96. Viamontes, GF, Markov, IL & Hayes, JP 2009, 'Quantum Circuit Simulation', Springer.

97. Vos, A D 2010, 'Reversible Computer Hardware', Electronic notes in Theoretical Computer Science, vol. 253, pp. 17-22.

98. Wille, R & Drechsler, R 2010, 'Effect of BDD Optimization on Synthesis of Reversible and Quantum Logic', Electronic Notes in Theoretical Computer Science, vol. 253, pp. 57-70.

99. Wille, R, Grobe, D, Frehse, S, Dueck, GW & Drechsler, R 2011,'Debugging reversible circuits', Integration, The VLSI Journal, Elsevier, vol. 44, pp. 51-61.

100. Wille, R, Soeken, M, Miller, DM & Drechsler, R 2014, 'Trading off circuit lines and gate costs in the synthesis of reversible logic', Integration, The VLSI Journal, vol. 47, no. 2, pp 284-294.

101. Yokoyama, T 2010, 'Reversible Computation and Reversible Programming Languages', Electronic Notes in Theoretical Computer Science, vol. 253, pp. 71-81.

102. Zadeh, RP & Haghparast, M 2011, 'A new Reversible/Quantum Ternary Comparator', Australian Journal of Basic and Applied Science, vol. 5, no.12, pp. 2348-2355.

103. Zeitzoff, P & Chang, J 2005, 'A perspective from the 2003 ITRS', IEEE Circuits & Devices Magazine, vol. 21, pp. 4-15.