Synthesis Of Reversible Universal Logic Around QCA With Online Testability

QUANTOM DOT CELLULAR AUTOMATA (QCA)-DESIGNER

S20220020274 G.Sai Kiran

ABSTRACT:

Quantum-dot Cellular Automata (QCA) hold promise as a viable technology for chip multi-processors (CMPs) with thousands of processors, offering energy-efficient digital circuit design through reversible logic. However, challenges such as the need for a large number of logic gates and a high defect rate hinder the performance of QCA-based designs. This study introduces an innovative approach to synthesize a reversible universal QCA logic gate (RUG), aiming to minimize garbage outputs and reduce the number of logic gates required for implementation, while simultaneously enhancing defect tolerance. Additionally, a concurrent error detection mechanism is proposed to enable online testing of circuits built using the RUG. Experimental results demonstrate that the RUG facilitates energy-efficient, cost-effective, and testable QCA circuit designs.

INTRODUCTION:

Modern digital design techniques focus on achieving energy-efficient implementation of complex logic circuits with high device density. Quantum-dot Cellular Automata (QCA) is emerging as a promising technology to meet these objectives, offering device densities as high as devices/cm². It features a switching speed of 10 ps and power dissipation in the range of 100 W/cm² [1].

The 3-input majority gate (or majority voter) forms the foundational unit of QCA-based design. However, as the majority gate is not functionally complete, a combination of majority gates and inverters, known as MI, is utilized to realize QCA designs. Despite this, the need for numerous logic gates and the resulting garbage outputs limit the efficiency of nano-scale QCA designs.

To address this, universal gate structures such as the AND-OR-Inverter (AOI) [2] have

been proposed to achieve cost-effective digital designs. However, these gates implement irreversible logic, making them unsuitable for energy-efficient designs. Landauer [3] demonstrated that irreversible computations result in information loss, generating joules of heat energy per bit transition, as defined by the Shannon-von Neumann-Landauer (SNL) limit:

where is the Boltzmann constant and. This represents the minimum energy required to process a bit. Reversible computation, however, allows for operation below the SNL limit, as Bennett [4] showed that zero power dissipation is achievable when circuits are built using reversible logic gates.

Reversible logic conserves energy through charge recovery in CMOS circuits, while QCA circuits leverage a clocked, information-preserving mechanism [5]. This enables QCA circuits to dissipate energy significantly below , making QCA technology appealing for energy-efficient chip multi-processors (CMPs). However, the high defect rate of QCA circuits remains a major challenge [2].

These challenges drive the need for a novel gate structure in QCA that is both reversible and capable of realizing universal logic functions. Such a structure must optimize the number of logic gates, minimize garbage outputs, and ensure defect tolerance while maintaining energy efficiency. To address these requirements, we propose a Reversible Universal Gate (RUG), which reduces the gate count and garbage outputs in digital designs.

The RUG enables area-efficient implementation of complex logic circuits while ensuring near-zero energy loss. Its effectiveness is demonstrated through the realization of standard and symmetric functions. Additionally, a concurrent error detection scheme is proposed for RUG-based designs, eliminating the need for comparators typically used in reversible circuit testing. The following section delves into the fundamentals of QCA and reversible gate design.

THE QCA BASICS

A quantum dot is a confined region where an electron is restricted by quantum-mechanical effects, as illustrated in Fig. 1(a). A quantum cell comprises four such quantum dots arranged at the corners of a square and houses two free electrons [1]. These electrons can quantum-mechanically tunnel between the dots, stabilizing in either polarization (representing logic 0) or (representing logic 1), as depicted in Fig. 1(b). Synchronization and timing in QCA are achieved through a cascaded clocking mechanism involving four distinct, periodic phases [1].

Quantum-Dot Cellular Automata CELL (QCA CELL):

A **QCA Cell** is the basic building block of Quantum-Dot Cellular Automata, representing binary data through the quantum-mechanical properties of confined electrons.

1.Structure of a QCA Cell

- A QCA cell consists of **four quantum dots** arranged at the corners of a square.
- It contains **two free electrons**, which can tunnel between the dots.
- Coulombic repulsion ensures that the electrons occupy different dots.

2.Binary Representation (Polarization)

- Binary digits are represented by the **polarization** of the cell:
- **Polarization:** Electrons align diagonally, representing logic 0.
- **Polarization:** Electrons align in the opposite diagonal, representing logic 1.

3.Clocking and Synchronization

- QCA cells rely on a **clocking mechanism** to control electron positions.
- The clocking system uses **four periodic phases** to ensure proper timing and synchronization of operations.

This design enables QCA cells to perform dense, energy-efficient digital computations.

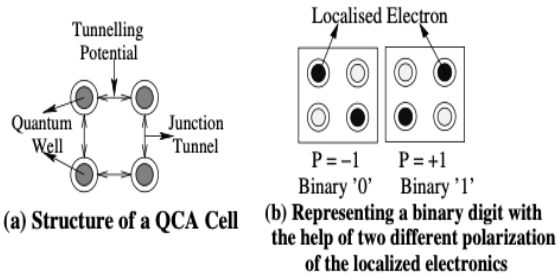


Figure 1. A QCA cell

QCA Logic Gate:

The fundamental structure implemented in QCA is the **3-input majority gate**, defined as (Fig. 2(a)). To achieve universal functionality, a **CMVMIN gate structure** has been proposed [6], which simultaneously realizes 3-input majority logic (MV) and minority logic (MIN) through its two outputs, and (Fig. 2(b)).

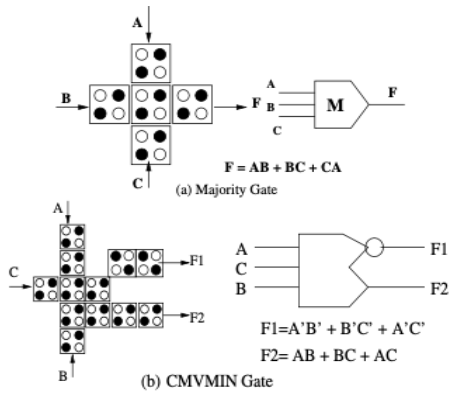


Figure 2. A Majority gate and a CMVMIN gate

An **inverter** in QCA can be implemented in two distinct orientations, as shown in Fig. 3(a). Additionally, two types of QCA wires are feasible for logic implementation. The only allowable wire crossing in QCA-based designs is depicted in Fig. 3(b). This crossing employs two orientations: a 90-degree structure (x-cell) and a 45-degree structure (+-cell). However, fabricating nano-scale cells with these two orientations presents significant challenges in QCA manufacturing.

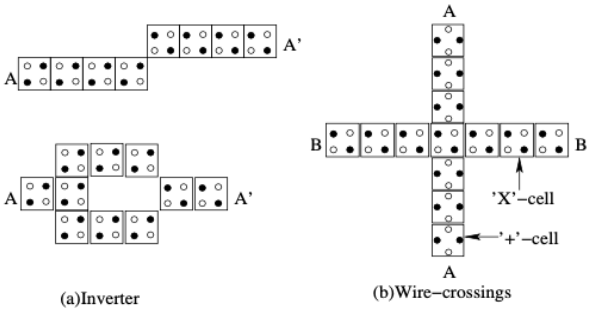


Figure 3. Wire-crossing and inverter

A comprehensive list of reversible gates is provided in [7], with some of the most significant examples including the **NOT gate**, **CNOT (Feynman) gate**, **Toffoli gate**, **Fredkin gate**, and **Peres gate** [9], as illustrated in Fig. 4.

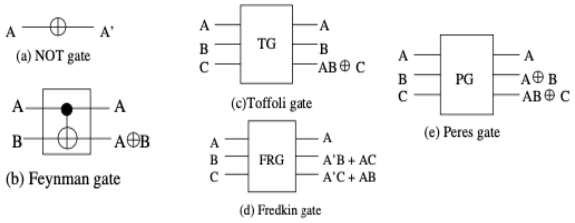


Figure 4. Reversible gates

Reversible Logic Gate:

A logic gate is considered **reversible** if its input-to-output mapping is bijective, meaning each unique input

(Inputs: I = (I1, I2, I3, ..., IN)

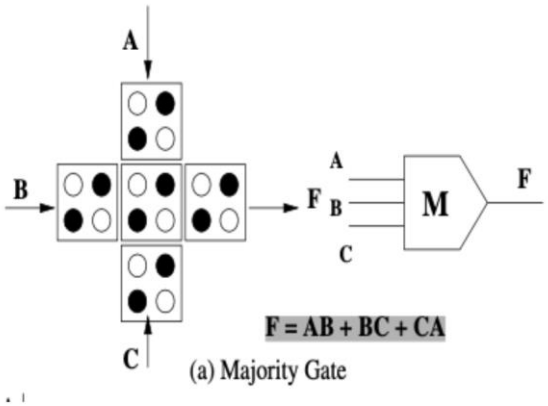
Outputs: O = (O1, O2, O3, ..., ON))

corresponds to a unique output (), and the number of inputs equals the number of outputs. A reversible gate is defined with inputs and outputs.

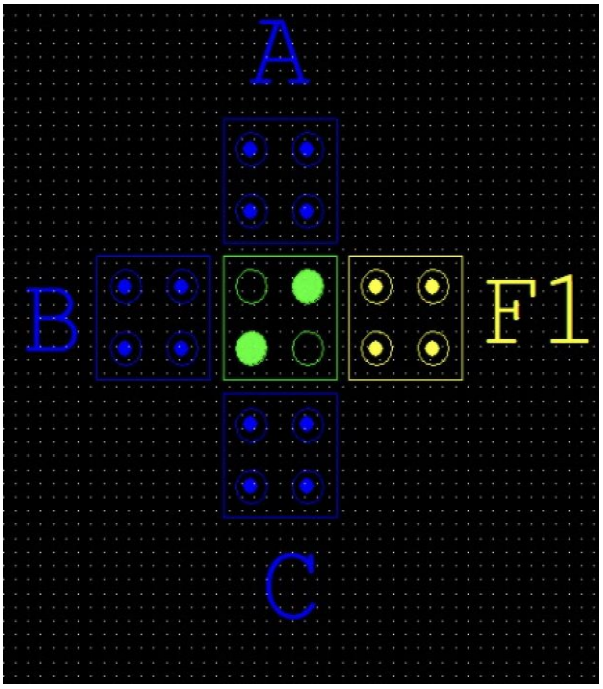
LIST OF ALL REVERSIBLE GATES DESIGNED IN THIS PAPER:

- Majority Gate
- CMVMIN Gate
- TOFFOLI Gate
- PERES Gate
- Wire Crossing & Inverter
- FEYNMAN GATE
- QCA Cell Layouts of RUG

1) MAJORITY GATE:



(a) Majority Gate



QCA DESIGN OF MAJORITY GATE

The Majority gate is a fundamental logic gate used in Quantum-dot Cellular Automata (QCA) design. It performs a critical role in digital circuits by implementing a three-input majority logic function.

Functionality:

The Majority gate operates with three inputs (A, B, and C) and computes the output as:
This expression evaluates to 1 when the majority of the inputs are 1.

Key Features:

1. **Inputs**:

- Three binary inputs (A, B, and C).

2. **Output**:

- A single binary output (F), which reflects the majority of the inputs being 1.

QCA Layout:

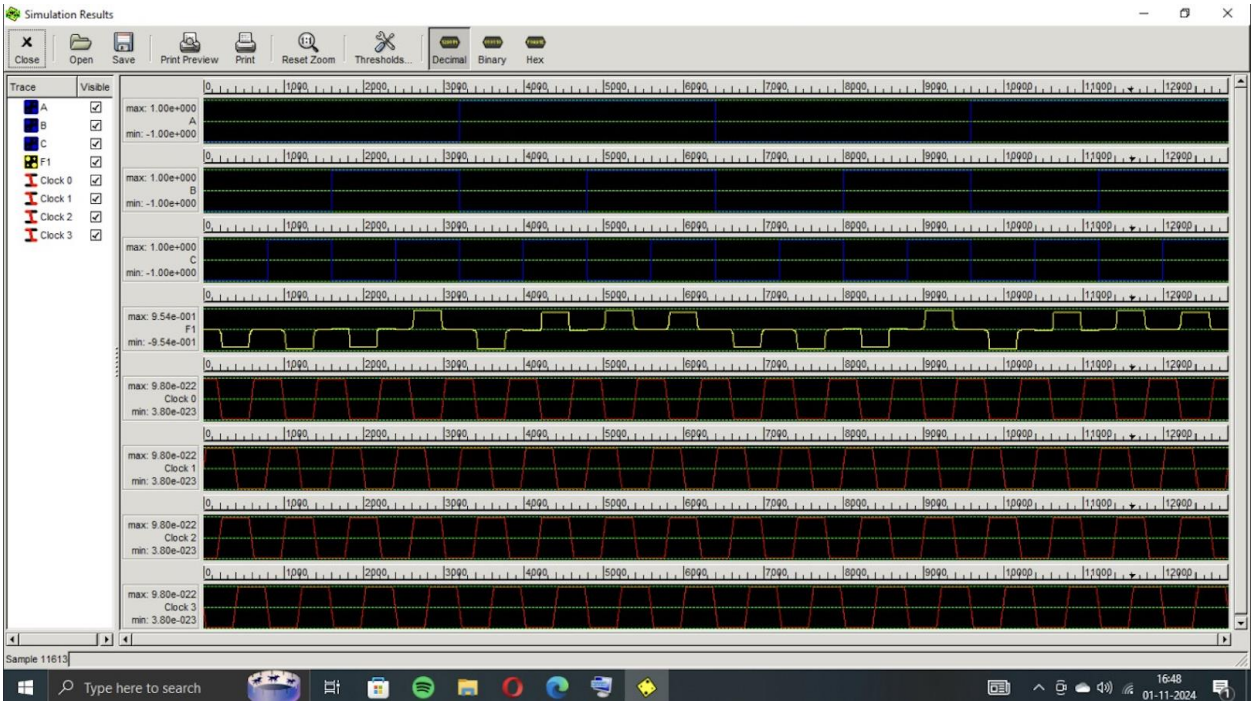
- The QCA layout for the Majority gate uses a compact arrangement of quantum-dot cells to represent the function.
- It ensures an optimal design with minimal cell count and efficient utilization of area.

Advantages:

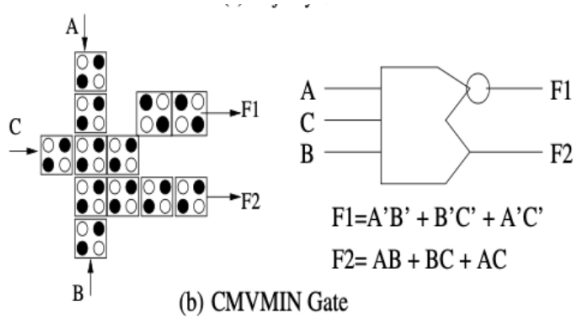
- Avoids unnecessary complexity.
- No clock cycle delays, enabling faster computation in circuits.

Verification:
Simulation results of the Majority gate using QCA Designer software verify its functionality. The design produces the correct output for all input combinations, ensuring reliability in QCA-based computations.

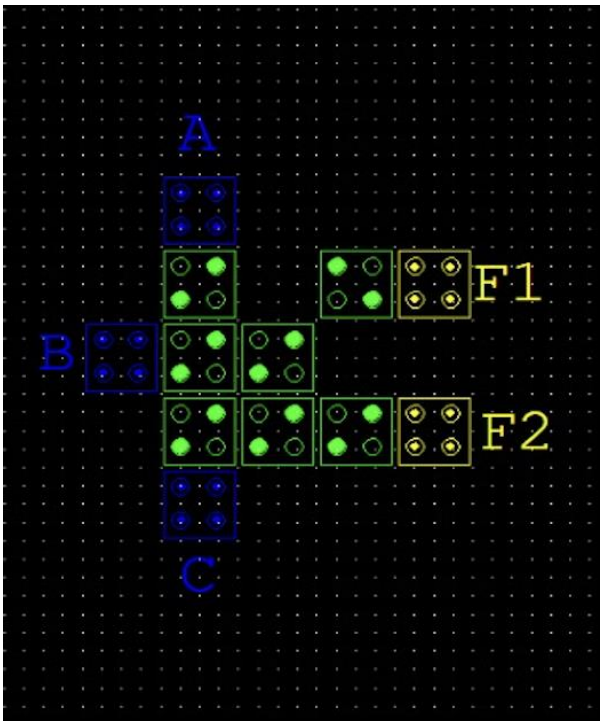
SIMULATION RESULT OF MAJORITY GATE:



2) CMVMIN GATE:



(b) CMVMIN Gate



QCA DESIGN OF CMVMIN GATE

The CMVMIN (Complemented Majority Voter-Minority) gate is a versatile logic gate in Quantum-dot Cellular Automata (QCA) circuits. It simultaneously computes two outputs, a complemented majority function (F1) and a regular majority function (F2), making it highly efficient for QCA-based designs.

Functionality:
The CMVMIN gate operates with three inputs (A, B, and C) and computes two outputs:

- F1 = A'B' + B'C' + A'C'

(Complemented majority function)

- F2 = AB + BC + AC

(Majority function)

Key Features:

1. **Inputs**:

- Three binary inputs (A, B, and C).

2. **Outputs**:

- : Outputs 1 when the majority of the inputs are 0.
- : Outputs 1 when the majority of the inputs are 1.

QCA Layout:

- The QCA layout for the CMVMIN gate uses a compact configuration of quantum-dot cells to compute both outputs simultaneously.
- This dual-output design reduces the overall complexity and cell count in circuit implementations.

Advantages:

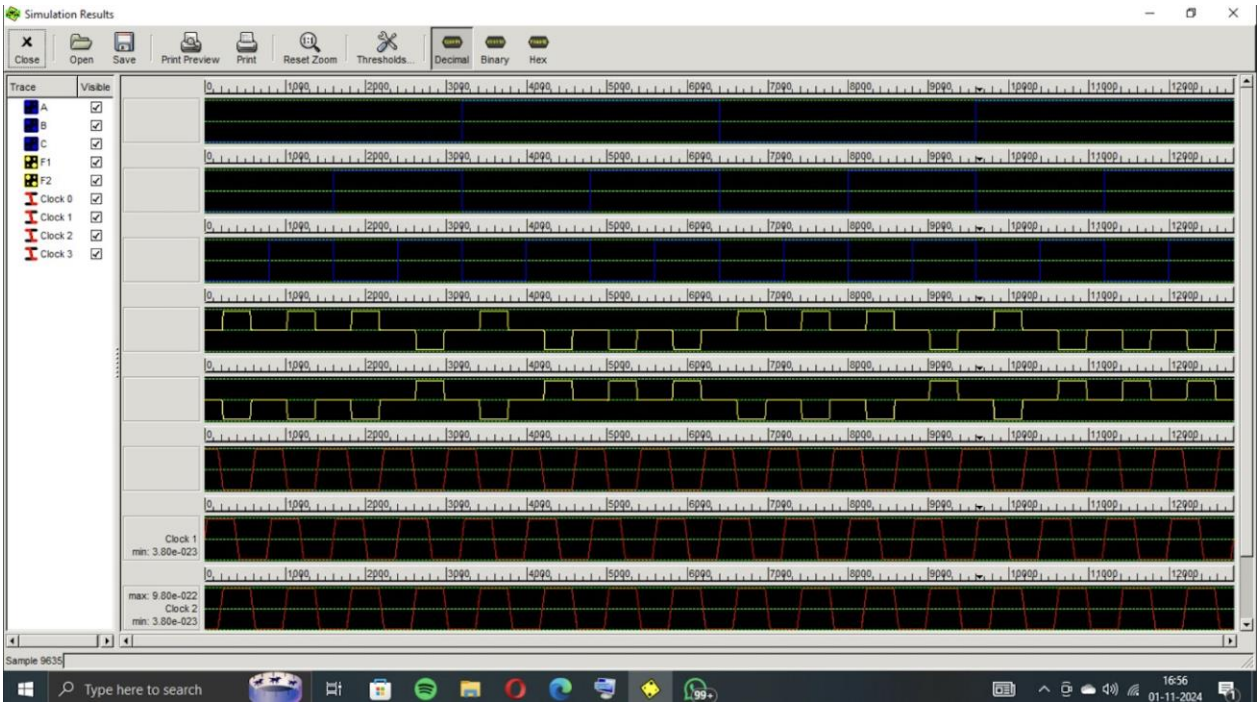
- Minimal layout design with efficient use of area.
- Supports both majority and complemented majority functions in a single gate.
- Eliminates clock delays, allowing for faster operation.

Verification:
Simulation results of the CMVMIN gate using QCA Designer software confirm its correct functionality. The design outputs the correct values for and across all input

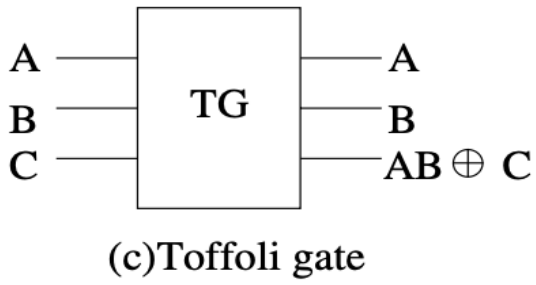
combinations, verifying its reliability for QCA applications.

Let me know if further adjustments are required!

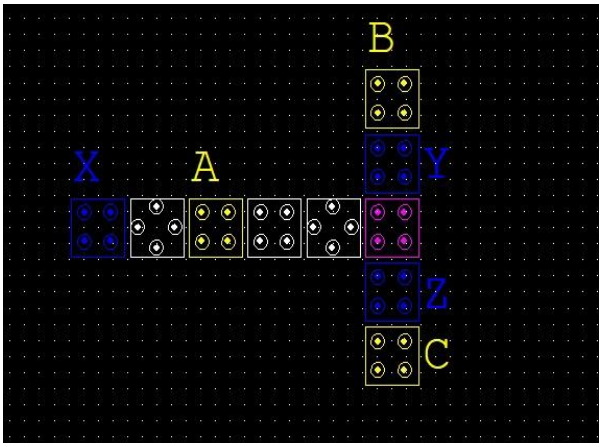
SIMULATION RESULT OF CMVMIN GATE:



3) TOFFOLI GATE:



(c)Toffoli gate



OVERVIEW:

The **Toffoli gate**, also known as the Controlled-Controlled-NOT (CCNOT) gate, is a universal reversible logic gate. It is an essential component for reversible and quantum computing, capable of implementing any Boolean

function. The gate operates on three inputs:

- Two control bits (Y and Y)
- One target bit (Z).

QCA DESIGN FOR TOFFOLI GATE

- The proposed design uses **10 quantum-dot cells** with **1 clock cycle delay**, significantly improving efficiency in terms of cell count and delay compared to existing designs.
- Displays the **simulation results**, verifying the functionality of the Toffoli gate. It confirms that the target bit is inverted correctly based on the state of the control bits.

Advantages of the Proposed Design

1. **Compact Layout**: Requires fewer cells, reducing the circuit area.
2. **Low Delay**: Operates with just 1 clock cycle delay, enabling faster computation.
3. **Energy Efficiency**: Minimizes power consumption due to the QCA technology.

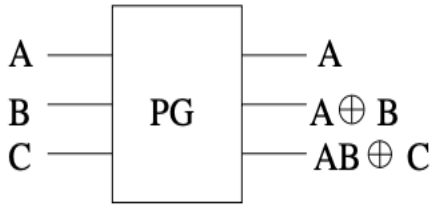
Applications

- **Quantum Computing**: Used to construct quantum algorithms and entangle qubits.
- **Universal Logic Design**: Can implement any Boolean function, making it a universal reversible gate.

SIMULATION RESULT OF TOFFOLI GATE:

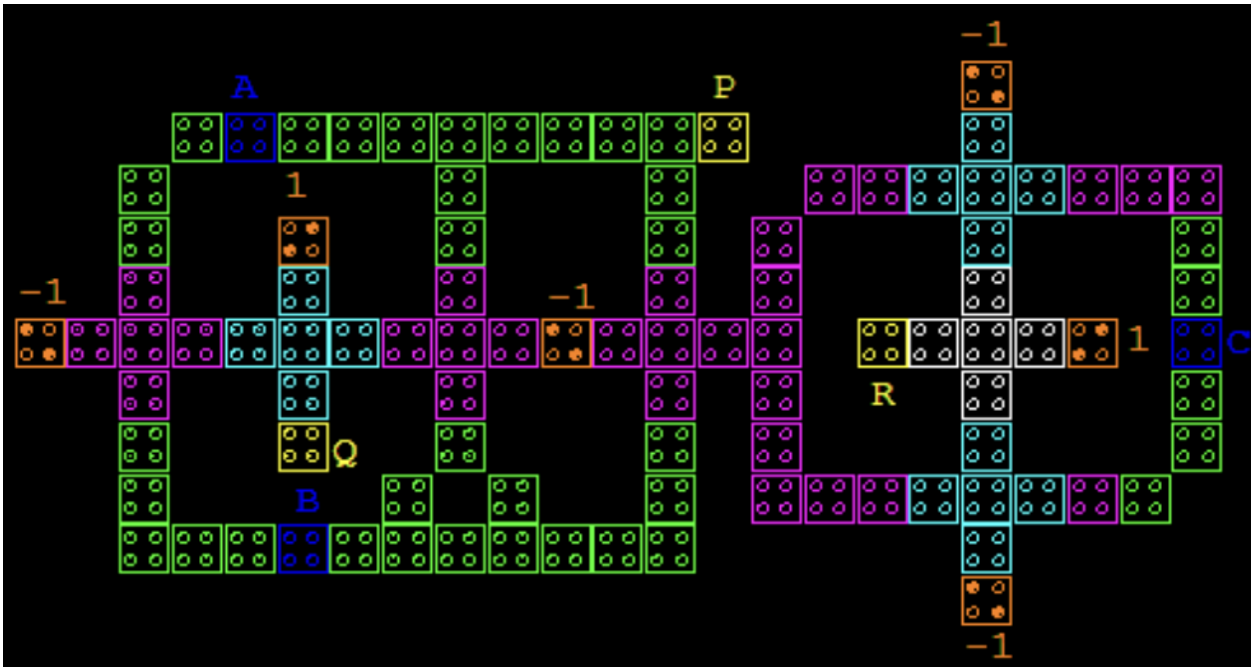

A
B
C
X
max: 1.00e+000
X
min: -1.00e+000
Y
max: 1.00e+000
Y
min: -1.00e+000
Z
max: 1.00e+000
Z
min: -1.00e+000

4) PERES GATE:



(e) Peres gate

QCA DESIGN OF PERES GATE:



OVERVIEW:

The Peres gate is a key reversible logic gate used in reversible computing and quantum circuits. It efficiently computes a combination of reversible logic functions with minimal overhead, making it ideal for QCA implementations.

Functionality:
The Peres gate operates with three inputs (A, B, and C) and produces three outputs:

•The first input is directly passed to the output.

•The XOR operation of inputs A and B.

•The XOR operation of the AND of A and B with input C.

Key Features:

1.**Inputs**:

•Three binary inputs (A, B, and C).

2.**Outputs**:

- Output 1: (unchanged).
- Output 2: $A \oplus B$
- Output 3: $AB \oplus C$

QCA Layout:

•The Peres gate's QCA layout is designed to minimize cell count while maintaining functionality.

•Its compact design ensures efficient utilization of space and resources in reversible circuit implementations.

Advantages:

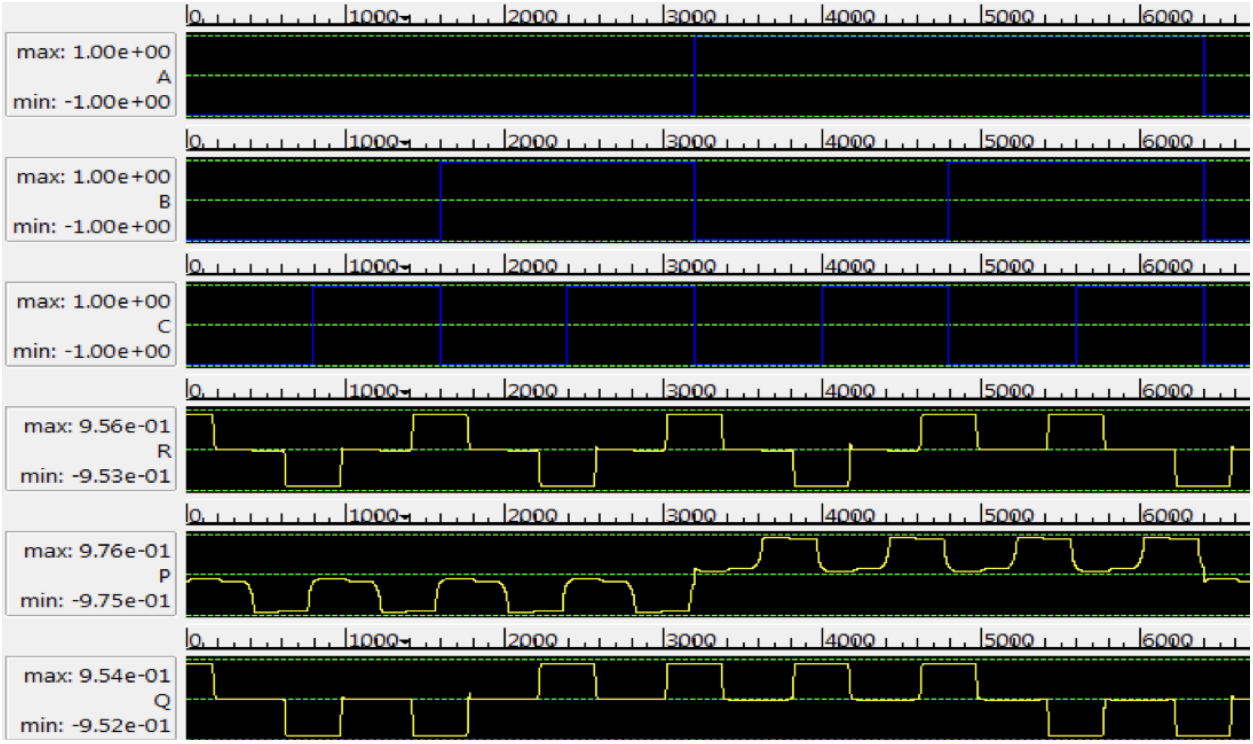
•Provides multiple useful logic functions in a single gate.

•Reversible logic operation minimizes energy dissipation.

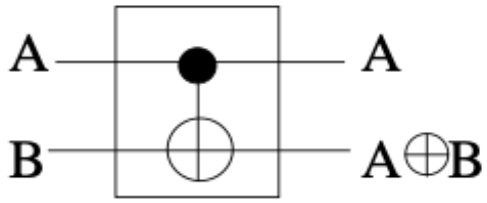
•Optimized for compact and efficient QCA circuit designs.

Verification:
Simulation results using QCA Designer software validate the correctness of the Peres gate. The outputs for all input combinations match the expected logic functions, confirming its suitability for QCA applications.

SIMULATION RESULT OF PERES GATE:

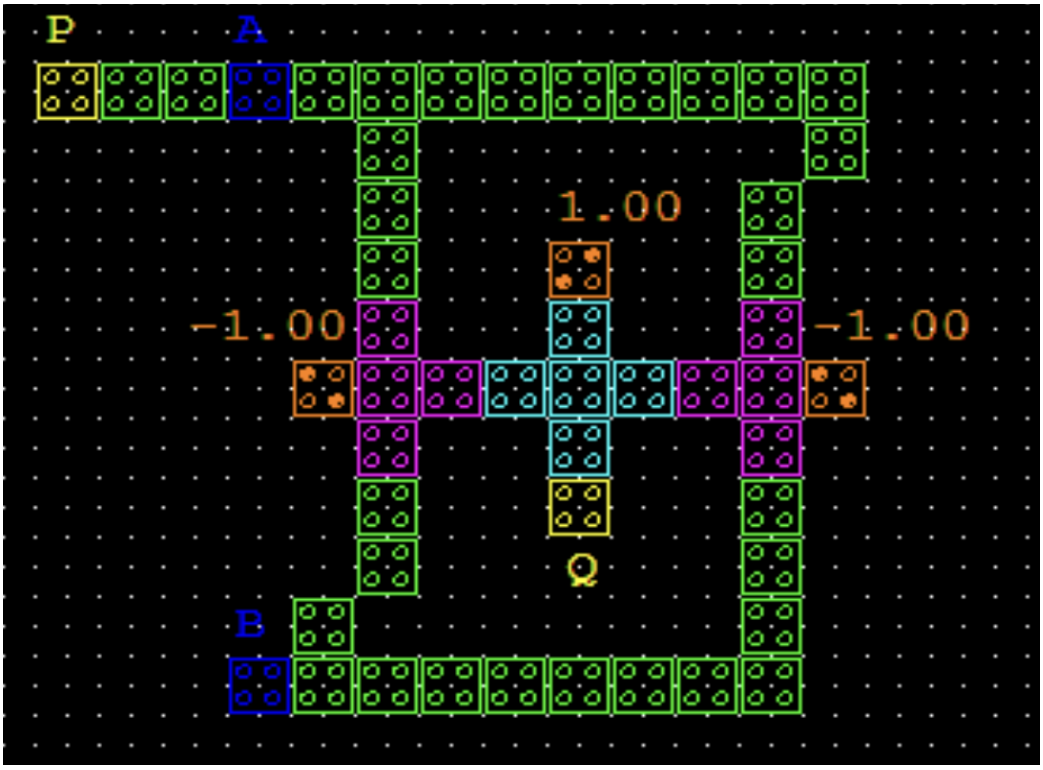


5) FEYNMAN GATE:



(b) Feynman gate

QCA DESIGN OF FEYNMAN CIRCUIT:



QCA DESIGN OF FEYNMAN GATE

The Feynman gate, also known as the Controlled-NOT (CNOT) gate, is a fundamental reversible logic gate used in reversible computing and quantum computing. It is a two-input, two-output gate where:

- One input serves as the control bit (A).
- The other input serves as the target bit (B).

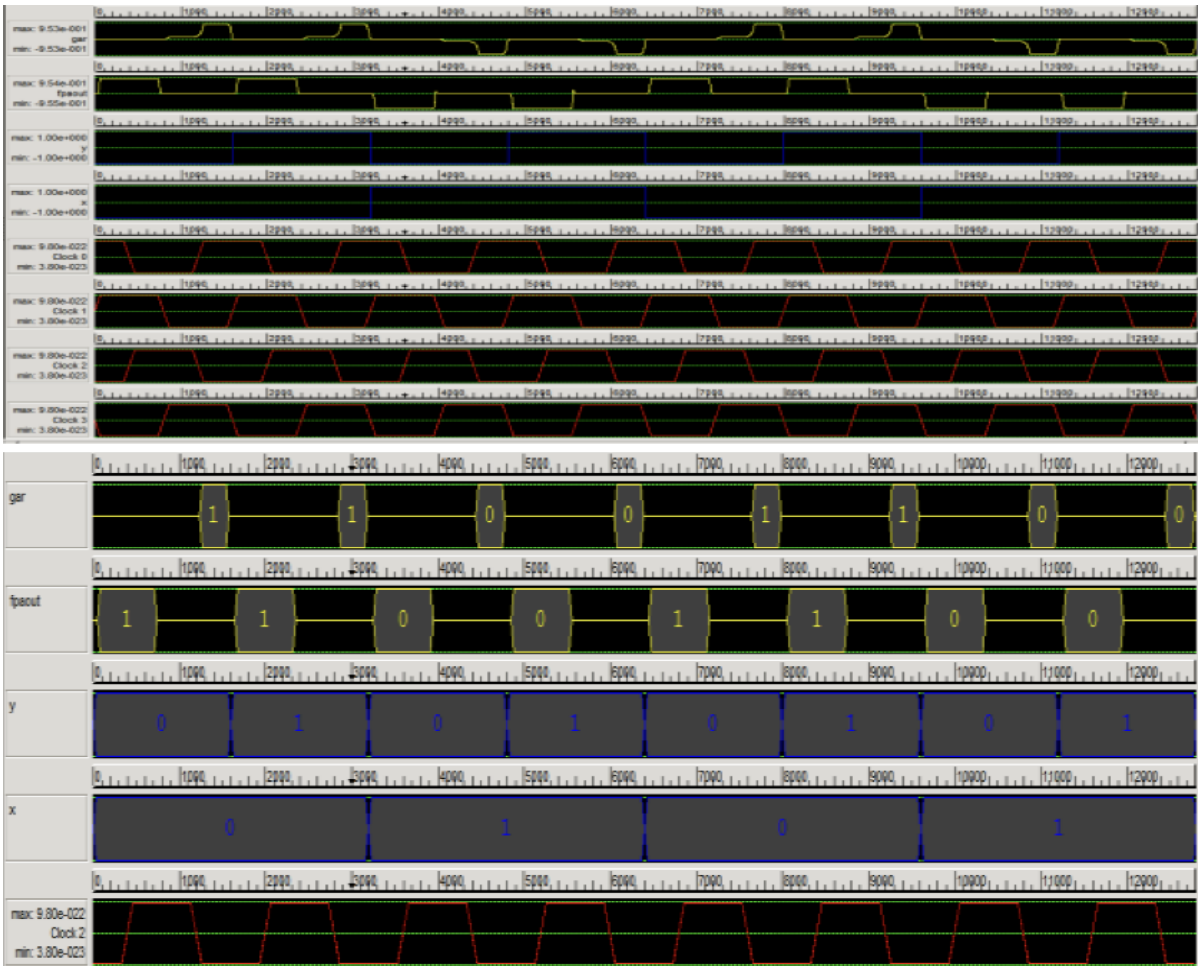
The target bit (B) is flipped (inverted) only when the control bit (A) is 1.

The QCA layout for the Feynman gate typically utilizes quantum-dot cellular automata (QCA) technology to minimize complexity and optimize area efficiency.

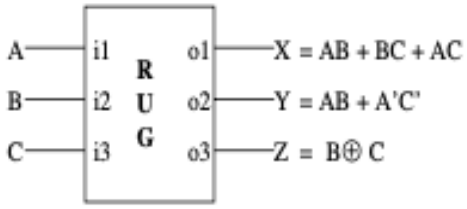
The design ensures fast computation by avoiding unnecessary clock cycle delays.

Simulation results of the Feynman gate using QCA Designer software confirm its functionality by demonstrating correct outputs for all possible input combinations.

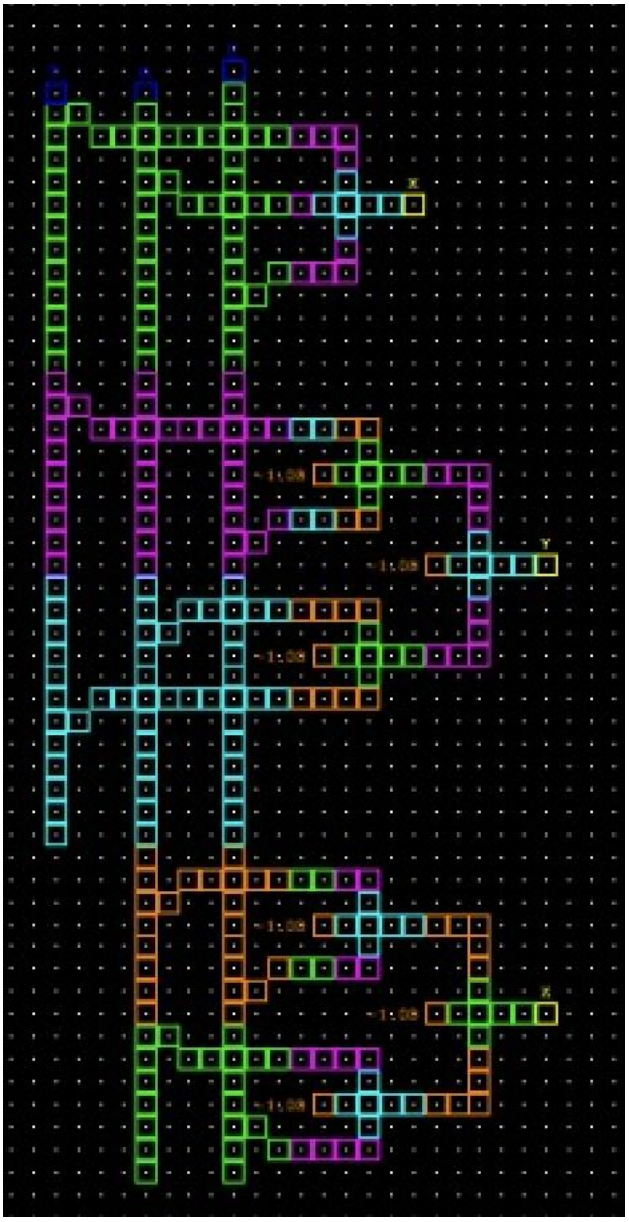
SIMULATION RESULT OF FEYNMAN GATE:



6) The Reversible Universal Gate (RUG):



QCA DESIGN OF Reversible Universal Gate CIRCUIT:



QCA Design of Reversible Universal Gate (RUG)

The Reversible Universal Gate (RUG) is a crucial 3x3 reversible logic gate that facilitates efficient implementation of various Boolean functions. It is widely applied in reversible computing and energy-efficient circuit designs. The RUG's operation is characterized by three outputs, each derived from the following logical functions:

```
// Output o1 (Majority Function)
assign o1 = (A & B) | (B & C) | (C & A);

// Output o2 (Universal Function)
assign o2 = (A & B) | (~A & ~C);

// Output o3 (XOR Function)
assign o3 = B ^ C;
```

The RUG is implemented using **Quantum-Dot Cellular Automata (QCA)** technology, which optimizes the design for reduced complexity and improved area efficiency. This gate achieves energy-efficient computation by preventing information loss, even in the realization of complex logic functions.

```
// Output o1 (Majority Function)
assign o1 = (A & B) | (B & C) | (C & A);


// Output o2 (Universal Function)
assign o2 = (A & B) | (~A & ~C);


// Output o3 (XOR Function)
assign o3 = B ^ C;
```

The universal function, $AB + A'C'$, simplifies the realization of XOR and XNOR operations, making the RUG a cost-effective solution for implementing intricate circuits.

Simulation results obtained through the **QCA Designer software** confirm the functionality of the RUG, producing correct outputs for all possible input combinations. Additionally, the reversible nature of the RUG ensures that every output uniquely maps to its corresponding input pattern, preserving information integrity.

A switching function f(x1, x2, ..., xn) is considered symmetric (or totally symmetric) with respect to its variables x1, x2, ..., xn if it remains invariant under any permutation of these variables. Symmetric functions are gaining significant attention from researchers in the field of VLSI design [11]. Various synthesis techniques for Boolean symmetric functions have been reported [11], tailored for different applications. A comparative performance analysis of the implementation of 2-variable and 3-variable symmetric functions is presented in Fig. 7. The analysis clearly shows that the implementation cost of symmetric functions is considerably lower in a RUG-based realization compared to other implementation methods.

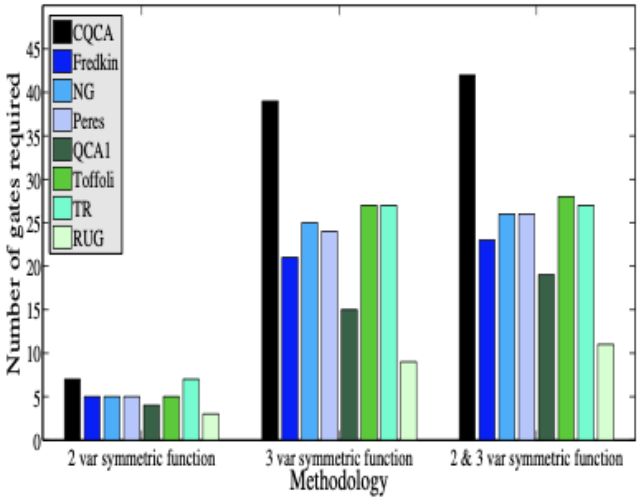
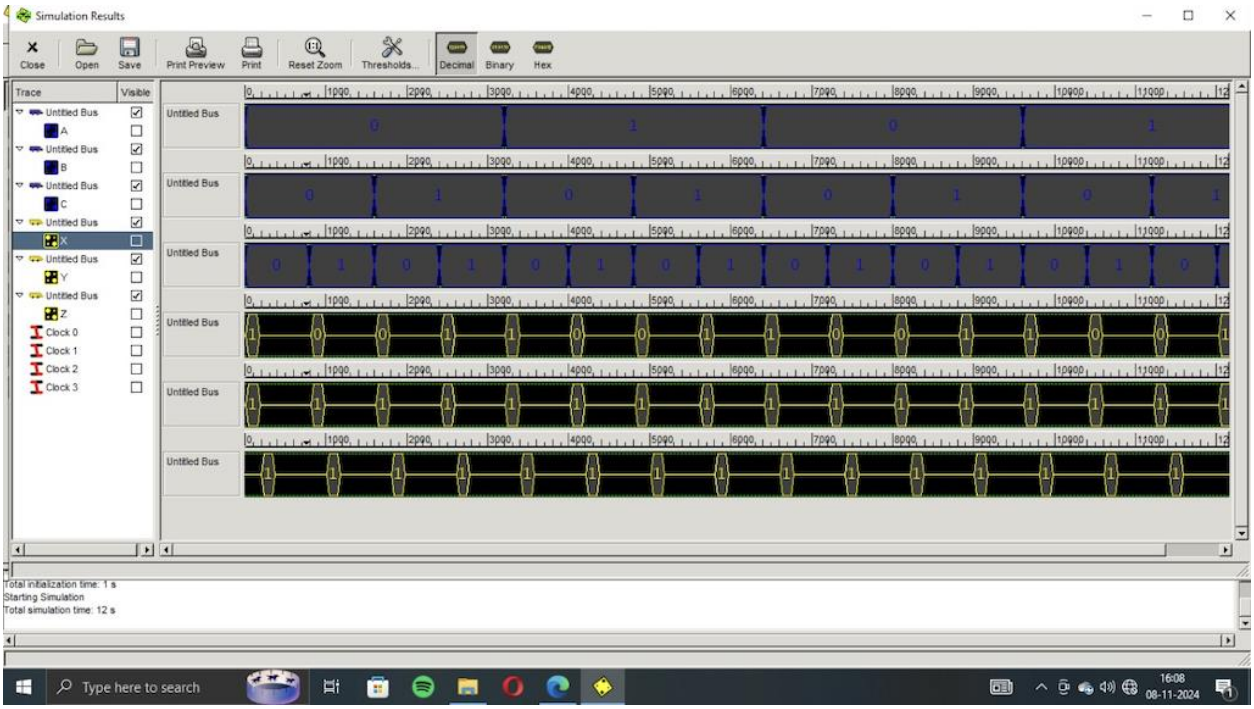


Figure 7. Performance of reversible gates in realizing symmetric functions

SIMULATION RESULT OF Reversible Universal Gate:



7) Wire Crossing & Inverter:

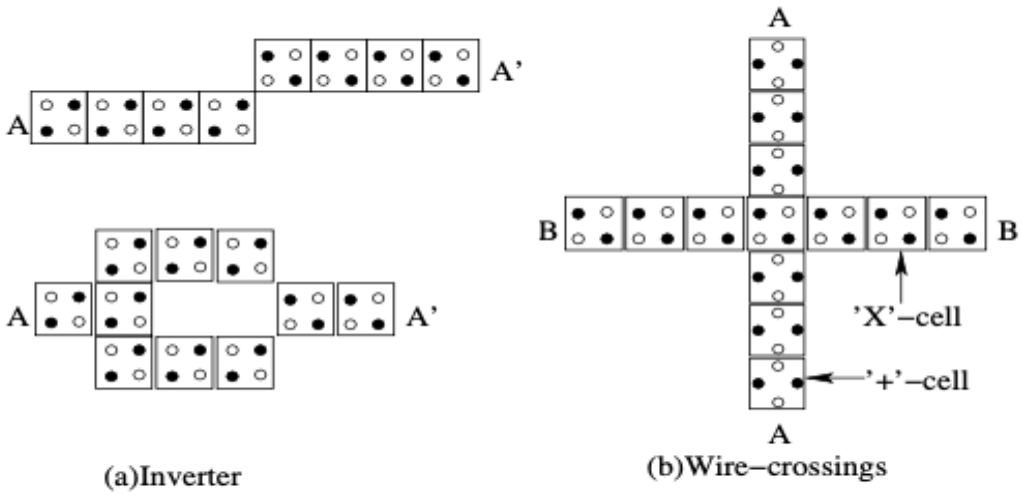
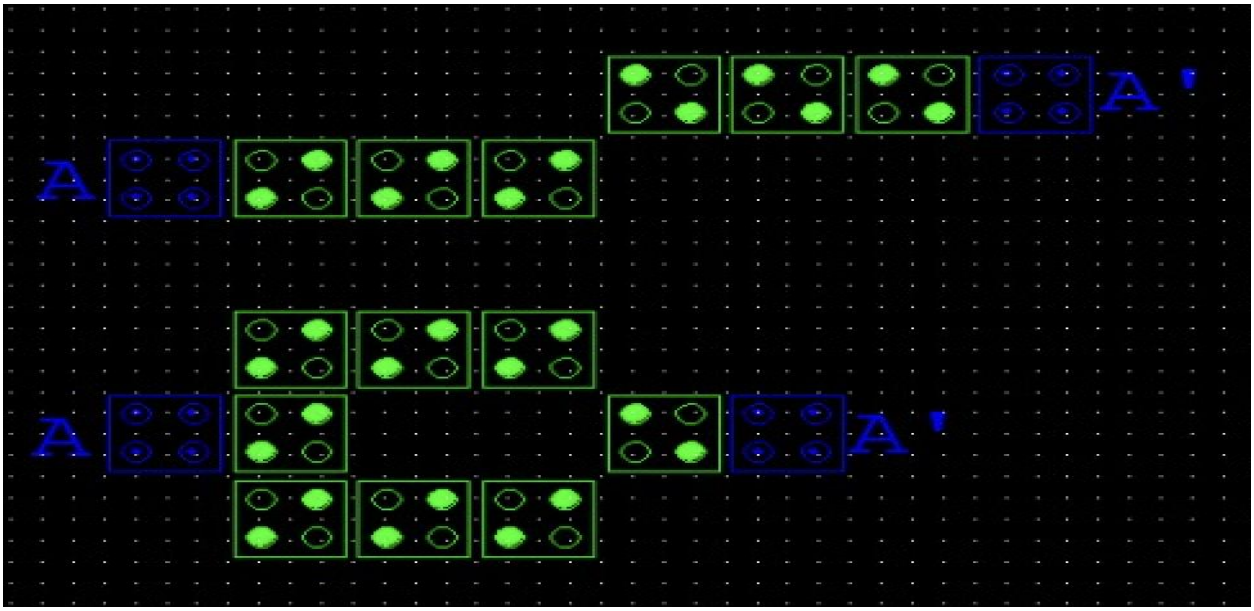
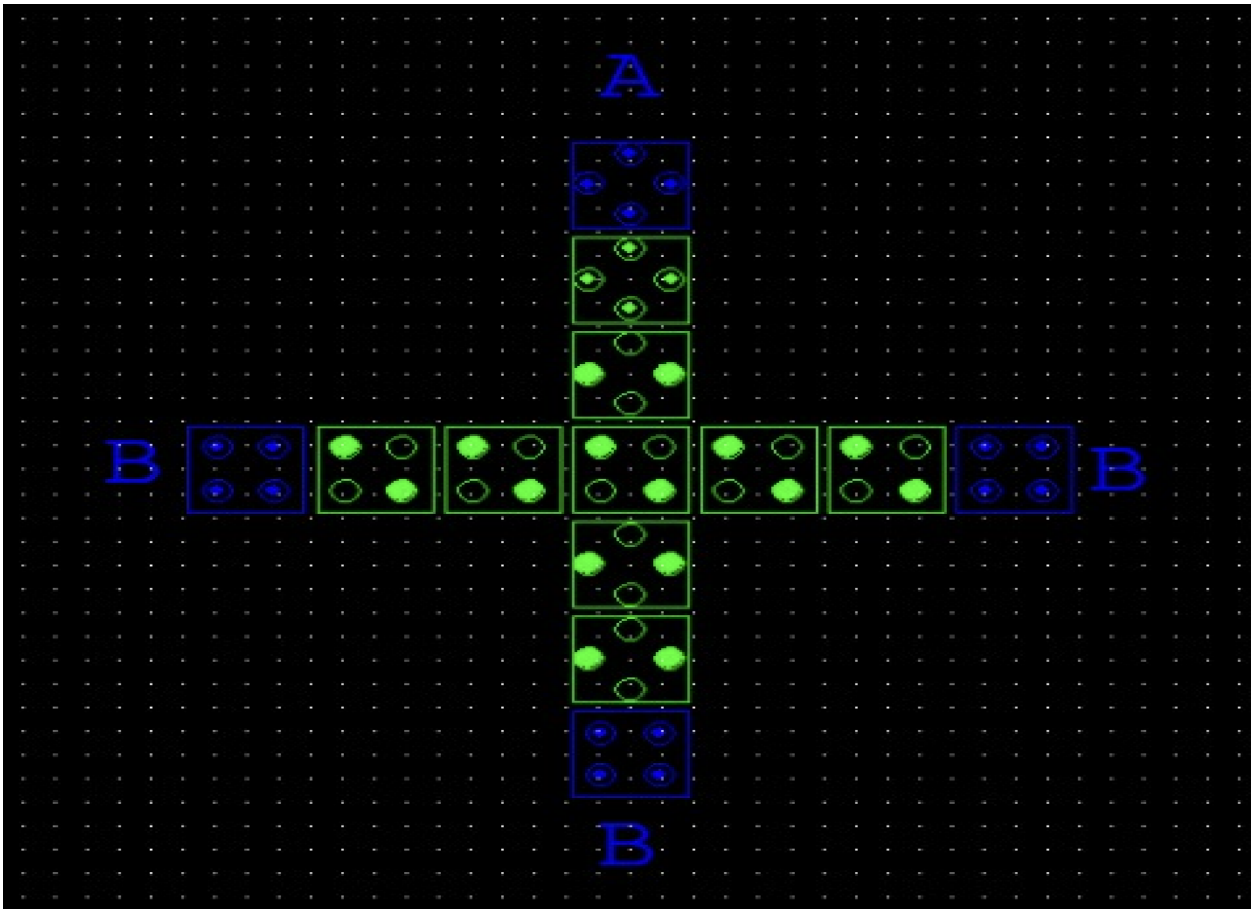


Figure 3. Wire-crossing and inverter

QCA DESIGN OF MAJORITY GATE





Wire Crossing in QCA

Functionality:

- QCA uses two types of wire orientations: **90-degree (x-cell)** and **45-degree (+-cell)**, which are needed for valid wire crossings.

Key Features:

1. **Wire Types:** Two orientations required for proper crossings.
2. **Crossing Mechanism:** Proper alignment ensures no signal interference.

QCA Layout:

- **x-cell** for 90° and **+-cell** for 45° orientations enable error-free wire crossings.

Advantages:

- Reliable signal transmission and compact design.

Verification:

- QCA Designer simulations confirm correct wire crossings without interference.

Inverter in QCA

Functionality:

- The inverter negates the input: $\bar{A}$ for input A.

Key Features:

1. **Input:** Single binary input.
2. **Output:** Logical negation of the input.

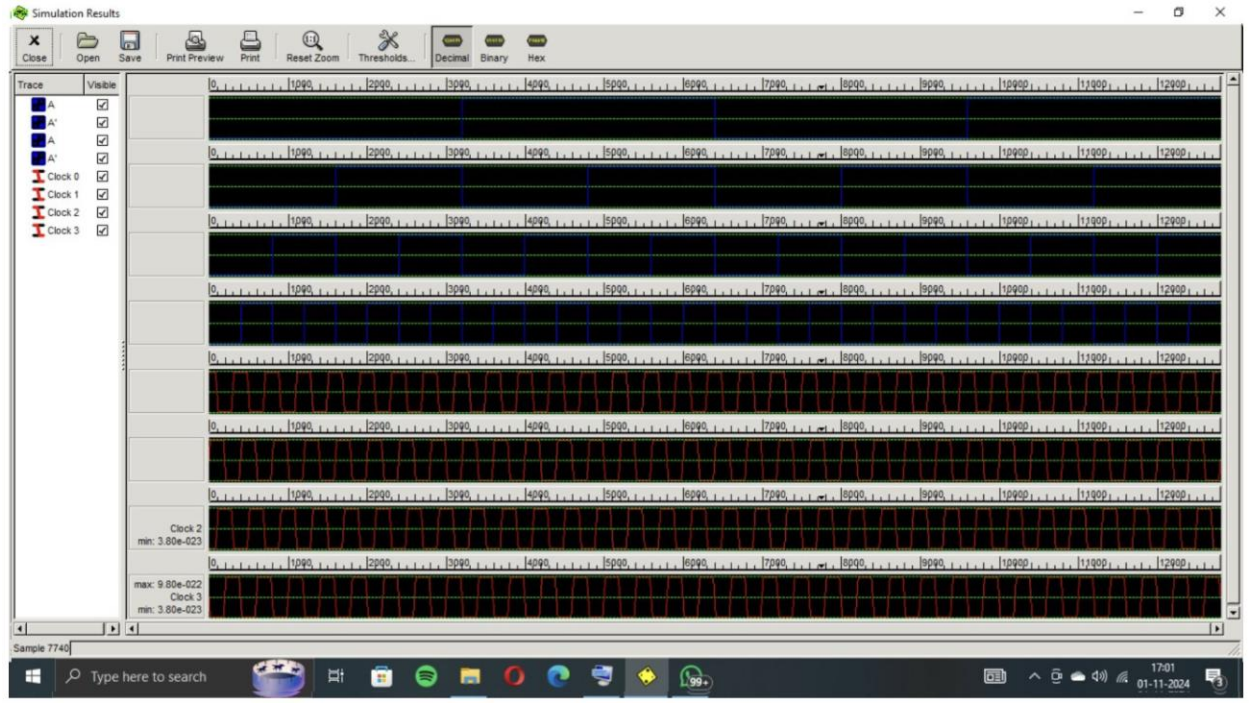
QCA Layout:

- Inverters are implemented using **x-cell** or **+-cell** arrangements for inversion.

Advantages:

- Simple design and low power consumption.
  Verification:
- QCA Designer software verifies that the inverter produces correct outputs.

SIMULATION RESULT OF WIRE CROSSING AND INVERTER:


Simulation Results
Close
Open
Save
Print Preview
Print
Reset Zoom
Thresholds...
Decimal
Binary
Hex
Trace
Visible
A
A'
A
A'
Clock 0
Clock 1
Clock 2
Clock 3
Clock 2
min: 3.80e-023
max: 9.80e-022
Clock 3
min: 3.80e-023
Sample 7740
Type here to search
17:01
01-11-2024

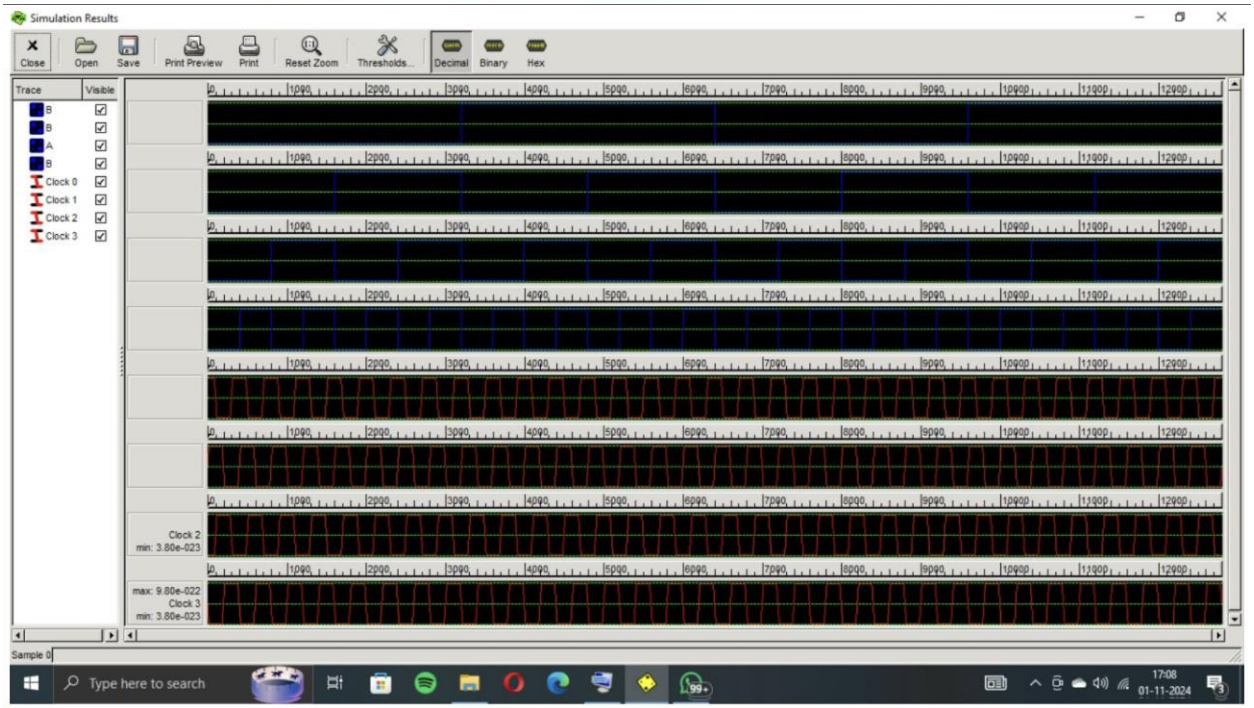

Simulation Results
Close
Open
Save
Print Preview
Print
Reset Zoom
Thresholds...
Decimal
Binary
Hex
Trace
Visible
B
B
A
B
Clock 0
Clock 1
Clock 2
Clock 3
Clock 2
min: 3.80e-023
max: 9.80e-022
Clock 3
min: 3.80e-023
Sample 0
Type here to search
17:08
01-11-2024

Table V
TRUTH TABLE FOR ERROR PROPAGATION AT X

A	B	C		X
D	1	0		D
D	0	1		D
0	D	1		D
1	D	0		D
0	1	D		D
1	0	D		D

D = Faulty value

The fault propagation in other outputs Y and Z are shown in Table VI) and Table VII respectively. These tables are utilised to generate test patterns for a RUG based design.

Table VI
TRUTH TABLE FOR ERROR PROPAGATION AT Y

A	B	C		Y
D	1	1		D
D	0	0		D'
1	D	ϕ		D'
0	ϕ	D		D'

D = Faulty value, D'= complement of D

Table VII
TRUTH TABLE FOR ERROR PROPAGATION AT Z

A	B	C		Z
ϕ	D	0		D
ϕ	D	1		D
ϕ	0	D		D
ϕ	1	D		D'

D = Faulty value, D'= complement of D

Table III
PERFORMANCE OF RUG IN REALIZING REVERSIBLE BENCHMARK

Benchmark	♯input	♯output	♯ Gates	♯ Garbage
RD32	3	2	3	4
Ham3	3	3	8	12
4mod5	4	1	11	19
Xor5	5	1	4	8
2of5	5	1	20	30
RD53	5	3	28	23

IV. CONCURRENT ERROR DETECTION

The last two rows in Table I represent that for any value of C and A=B=1, the output X is always 1. This is described in the first row of Table IV. All such combination of A, B, C that generate the constant values at X are summarised in Table IV.

Table IV
TRUTH TABLE SHOWING CONSTANT VALUES AT X

A	B	C		X
1	1	ϕ		1
ϕ	1	1		1
1	ϕ	1		1
0	0	ϕ		0
ϕ	0	0		0
0	ϕ	0		0

ϕ =Don't care value

CONCLUSION

This work presents the Reversible Quantum Cellular Automata (QCA) logic gate structure, referred to as the RUG, which serves as a universal reversible gate for energy-efficient logic design. The RUG is specifically designed to address the growing need for energy-saving mechanisms in modern digital circuits, especially in the context of quantum and nanoscale computing. It has been demonstrated that circuits based on the RUG-based QCA gates offer significant advantages in terms of both reduced gate count and minimized garbage outputs compared to traditional designs that rely on existing reversible gates.

The main advantage of the RUG gate lies in its cost-effectiveness. It reduces the complexity of the circuit by requiring fewer logic gates, which directly translates into lower power consumption and smaller circuit sizes. This is crucial for modern computing applications where power efficiency and space optimization are critical. Furthermore, the reduced garbage outputs—unnecessary outputs that do not contribute to the desired result but consume energy—are another key benefit of the RUG gate, which enhances the overall efficiency of the circuit.

Given its high device density, the RUG-based QCA circuits are particularly promising for use in Chip Multi-Processors (CMPs) with multiple processors. In CMP systems, the need for energy-efficient, compact, and cost-effective circuits is paramount, and the RUG-based approach provides a solution that can scale well in such multi-processor environments.

Additionally, the realization of symmetric functions and benchmark implementations using RUG has been explored, showcasing its potential for practical applications in reversible logic circuits. It has been established that a reversible gate capable of implementing universal, majority, and XOR functions can be utilized to build low-power QCA designs. This flexibility makes the RUG a highly versatile gate for various computational tasks, which are essential in quantum computing and other advanced technologies.In addition to the functional advantages, a novel methodology for concurrent error detection in reversible logic circuits has been proposed. Unlike conventional error detection techniques that rely on comparators, this approach eliminates the need for separate error detection modules, thereby reducing the circuit's complexity and improving overall system reliability. This innovation in error detection further strengthens the case for the adoption of RUG-based QCA circuits in high-performance and low-power computing environments.

THANK YOU