EFFICIENT FAULT TOLERANT BASED SCL IMPLEMENTATION USING QCA AND GDI

1. N S S R DEEPIKA PATNALA, 2.ANALA NARAYANA MURTHY
1. Assistant Professor, Dept. of ECE, VSM College of Engineering, Ramachandrapuram, A.P
2. Assistant Professor, Dept. of ECE, VSM College of Engineering, Ramachandrapuram, A.P

ABSTRACT:

The semiconductor era has been thriving since the past four decades but as we continually attain smaller chip sizes comprising of millions of transistors, the problems grow at an unmitigated speed too. Testing such huge circuits poses a huge problem unless tackled prudently. The best case scenario given the current circumstances would be to create a favorable test environment on-chip by implementing Design for Test (DFT) techniques. Asynchronous digital design methodologies are currently gaining popularity since they enjoy the benefits of reduced area, power and low EMI. In spite of several advantages, due to the absence of the global clock and the presence of more state holding gates, testing these circuits presents a challenge to the designer. A DFT implementation for one such asynchronous class known as Sleep Conventional Logic (SCL) circuits along with Gate diffusion input with quantum cellular automata has been proposed in this paper. The methodology discussed, exploits the technique of test points' insertion in order to improve the controllability of difficult feedback paths. Enhanced observability of long paths in the circuits is achieved by introducing of balanced tree structures or scan latches. This approach has been automated and works with industry standard tool suites, such as Tanner tools16.0

KEYTERMS: Design for Test, Null Conventional Logic, Sleep Conventional Logic, quantum cellular automata, Gate diffusion input.

INTRODUCTION:

Software development processes mainly focus on controlling and reducing errors, Identifying and rectifying software faults that do occur, and support to provide high quality software [2]. It is well understood that delivering quality software is no longer an advantage but is a necessary factor. So we can say that acceptance and success of any software product depends on its quality. The quality can be measured in terms of attributes of the system. Unfortunately, most of the industries not only fail to deliver a quality product to their consumers, but also do not understand the significant quality attributes for ensuring the software quality; testing is the main activity in software development process. Software testing is an important discipline of software engineering, and consumes significant amount of time and effort. An appropriate approach is required to perform testing activities properly and effectively. Software testability always supports the testing process and facilitates the creation of better quality software within given time and budget. Testability is a quality factor; its measurement or eval However, testability has always been an elusive concept and its correct measurement or evaluation is a difficult exercise. Most of the studies measure testability or precisely the quality attributes that have impact on testability at the source code level. It has been inferred from the literature survey on testability factors that there is an acute need of proposing a commonly accepted minimal set of the factors affecting software testability [4, 2]. Estimating testability at a later stage leads to the late arrival of desired information, leading to late decisions about changes in design. This greatly increases total cost and rework. Therefore, early evaluation of testability in the development process may enhance quality and reduce testing efforts and costs. SLEEP convention logic (SCL), also known as multithreshold NULL convention logic (MTNCL) [1]–[4], is a variant of NULL convention logic (NCL) [5], [6] that takes advantage of the Multi-Threshold Complementary Metal Oxide Semiconductor (MTCMOS) power-gating technique [7], [8] to further reduce the power consumption. The application of MTCMOS to the NCL circuits comes with interesting architectural changes that

ultimately results in area and performance advantages as well. Similar to most other asynchronous logic styles, SCL is not supported by the current synchronous Automatic Test Pattern Generation (ATPG) tools [9]. This isdue to a number of major changes in the circuit architecture, such as using asynchronous handshaking signals instead of a global clock for synchronization, using multirail encoding to represent signals, and using threshold gates instead of traditional Boolean gates. Design for testability (DFT) is a major concern in today's semiconductor industry because it is essential to reduce test time, increase test quality, and reduce the cost associated with generating and applying test vectors. In contrast to the NCL circuits, no DFT methodology has been developed for the SCL circuits so far. The current NCL specific DFT techniques cannot be directly used for the SCL circuits due to the structural differences caused by introducing the sleep mechanism for power-gating. The aim of this paper is to analyze the various stuck-at faults within an SCL pipeline and propose a comprehensive scan-based testing methodology that provides for a high test coverage at the cost of the usual area overheard caused by introducing the scan chain. While lowering the no of transistors and decreasing the frequency of switching activities are prevalent techniques of reducing power consumption and area in ALU. First, while supply voltage reduction effectively lowers power consumption, its application is limited to the functional units in the ALU circuits. The project proposes a design for low power area optimized consumption ALU that exploits thebenefits of offline software, which can work alone in delivering minimum power consumption or work alongside supply voltage reduction technology to deliver even lower power consumption. Our ALU architecture consists of a set of fast and slow functional units. There are many advantages and plus points to the design of our ALU. Not only does it consumeminimal power during runtime, it does not require real time process to monitor performance. Neither is a hardware circuit needed to tune the supply voltage. Compared with other models operating on the supply voltage reduction principle, the ALU we have designed is far simpler. A quantum cellular automaton or QCA is an abstract model of quantum computation, devised in analogy to conventional models of cellular automata introduced by von Neumann. The same name may also refer to quantum dot cellular automata, which are a proposed physical implementation of "classical" cellular automata by exploiting quantum mechanical phenomena. QCA have attracted a lot of attention as a result of its extremely small feature size (at the molecular or even atomic scale) and its ultra-low power consumption, making it one candidate for replacing CMOS technology. The Quantum Cellular Automaton (QCA) concept represents an attempt to break away from the traditional three-terminal device paradigm that has dominated digital computation. Since its early formulation in 1993 at Notre Dame University, the QCA idea has received significant attention and several physical implementations have been proposed. In electronics, an adder or summer is a digital circuit that performs addition of numbers. In many computers and other kinds of processors, adders are used not only in the arithmetic logic unit(s), but also in other parts of the processor, where they are used to calculate addresses, table indices, and similar operations. Although adders can be constructed for many numerical representations, such as binary-coded decimal or excess-3, the most common adders operate on binary numbers. In cases where two's complement or ones' complement is being used to represent negative numbers, it is trivial to modify an adder into an adder–subtractor. Other signed number representations require a more complex adder. In the context of models of computation or of physical systems, quantum cellular automaton refers to the merger of elements of both (1) the study of cellular automata in conventional computer science and (2) the study of quantum information processing. In particular, the following are features of models of quantum cellular automata: The computation is considered to come about by parallel operation of multiple computing devices, or cells. The cells are usually taken to be identical, finite-dimensional quantum systems (e.g. each cell is a qubit). Each cell has a neighborhood of other cells. Altogether these form a network of cells, which is usually taken to be regular (e.g. the cells are arranged as a lattice with or without periodic boundary conditions).

DESIGN FOR TESTABILITY:

Since asynchronous architecture is different from synchronous architecture it needs a testing procedure which is different from the normal synchronous testing methods. For that it uses scan cell based DFT methods.The components of

a scan cell are shown in Fig.The testing procedure is as follows.

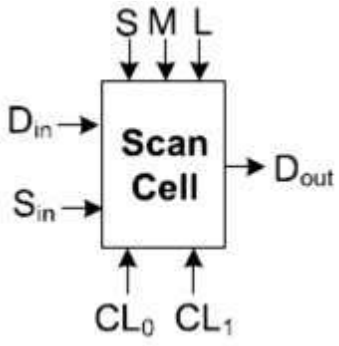


Fig: SCL Scan cell

1) All stuck-at faults on the inputs and output of all completion C-elements can be detected by applying a single {DATA, NULL} pair to the SCL pipeline and allow it to propagate through the whole circuitry.

2) SCL combinational logic block becomes a normal Boolean circuit when disabling the sleep signal, then it can be checked for stuck-at faults using the traditional combinational ATPG tools.

3) The stuck-at faults on the sleep signal forks within a combinational logic block are either untestable (stuck-at-0 faults) or can be ignored through fault collapsing (stuck-at-1 faults).

4) The stuck-at faults on the sleep signal forks within a completion detector block are either untestable (stuck-at-0 faults) or can be detected during the test of the completion C-elements (stuck-at-1 faults).

5) The scan chain design is used to test the stuck-at faults on the sleep signal forks within a register block. For this scan chain is also needed to apply the ATPG generated test patterns to the combinational logic blocks. The scan chain design of the SCL architecture is shown in Fig. A scan cell is needed to connect to a single rail in the circuitry. Since in SCL architecture each data is represented using dual rail encoding for each data bit there will be two scan cells connected. The overall circuit can be divided into many different functional blocks as shown in Fig.4 and then after each of these functional block a scan chain is there, that consists of as many scan cells as the number of rails and celement. Like that its goes on until the final functional block. Then as mentioned above in the points by applying a single {DATA, NULL} pair through the overall circuit the faults are detected.

SCL ARCHITECTURE:

The SCL framework is shown in Fig. 1. As in Null Convention Logic, in each pipeline stage of SCL architecture a combinational logic function block (Fi), a register block (Ri), and a completion detector block (CDi) are contained. And this SCL requires an additionalgate to synchronize between DATA and NULL phases of the circuit. This additional gate is a simple resettable C-element with inverted output and it is known as the completion C-element (Ci). Combinational logic blocks in the SCL are made of threshold gates [14], [15] which implement the unatefunctions and there is no logicinversions are allowed.

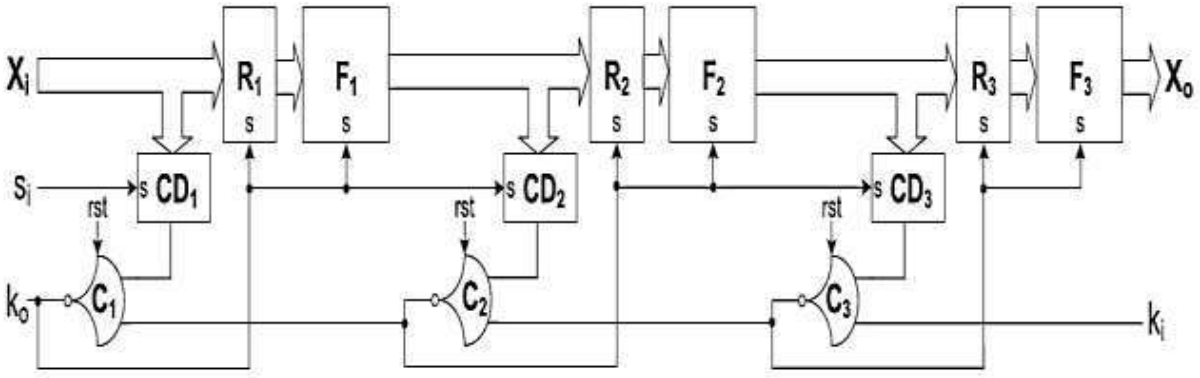


Fig-1: SCL architecture

An SCL gate is generally denoted as THmnWw1,…,wn where n is the number of inputs, m is the threshold of the gate, and w1, w2,... ,wn are the weights of inputs when the weights are > 1. If the inputs of the SCL gate are taken as x1,….,xn, the output of the SCL gate is logic 1 if x1w1 + · · · + xnwn ≥ m. For example, a TH23 gate has three inputs and its threshold is 2. In normal Boolean logic, the output of the TH23 gate can be expressed as Z = AB + AC + BC, here A, B, and C are its inputs. The transistor level design of this TH23 gate is shown in Fig.2.

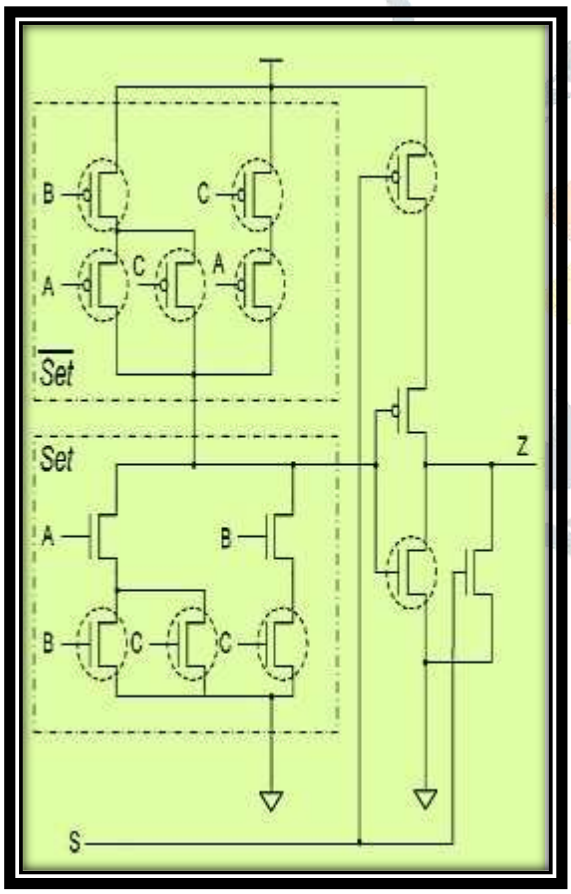


Fig- 2:SCL gate implementation of the TH23 gate.

GATE DIFFUSION TECHNIQUE:

The GDI method is based on the use of a simple cell as shown in below Fig.. At a first glance the basic cell resembles the standard CMOS inverter, but there are some important differences: GDI cell contains three inputs – G (the common gate input of

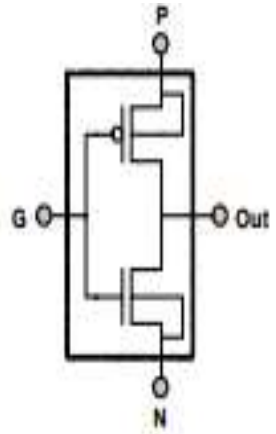
the nMOS and pMOS transistors), P (input to the outer diffusion node of the pMOS transistor) and N (input to the outer diffusion node of the nMOS transistor). The Out node (the common diffusion of both transistors) may be used as input or output port, depending on the circuit structure. The GDI cell is similar to a CMOS inverter structure. In a CMOS inverter the source of the PMOS is connected to VDD and the source of NMOS is grounded. But in a GDI cell this might not necessarily occur. There are some important differences between the two. The three inputs in GDI are namely-

1) G- common inputs to the gate of NMOS and PMOS

2) N- input to the source/drain of NMOS

3) P- input to the source/drain of PMOS

Bulks of both NMOS and PMOS are connected to N or P (respectively), that is it can be arbitrarily biased unlike in CMOS inverter. Moreover, the most important difference between CMOS and GDI is that in GDI N, P and G terminals could be given a supply 'VDD' or can be grounded or can be supplied with input signal depending upon the circuit to be designed and hence effectively minimizing the number of transistors used in case of most logic circuits (eg. AND, OR, XOR, MUX, etc). As the allotment of supply and ground to PMOS and NMOS is not fixed in case of GDI, therefore, problem of low voltage swing arises in case of GDI which is a drawback and hence finds difficulty in case of implementation of analog circuits.



The QCA cell:

In contrast to electronics based on transistors, QCA does not operate by the transport of electrons, but by the adjustment of electrons in a small limited area of only a few square nano meters. QCA is implemented by quadratic cells, the so-called QCA cells. In these squares, exactly four potential wells are located, one in each corner of the QCA cell (see figure 1). In the QCA cells, exactly two electrons are locked in. They can only reside in the potential wells. The potential wells are connected with electron tunnel junctions. They can be opened for the electrons to travel through them under a particular condition, by a clock signal. A later chapter will cover this in more detail. Without any interaction from outside, the two electrons will try to separate from each other as far as possible, due to the Coulomb force that interacts between them. As a result, they will reside in diagonally located potential wells, because the diagonal is the largest possible distance for them to reside.

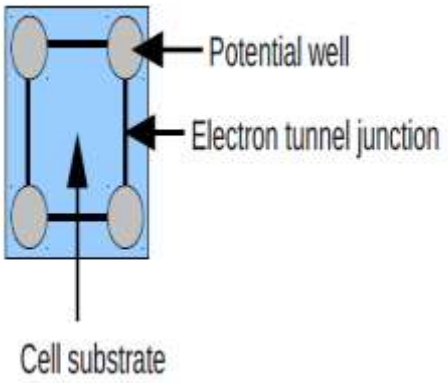


Figure : Anatomy of a QCA cell

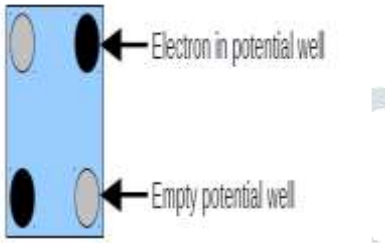


Figure : Electrons in potential wells

There are two diagonals in a square, which means the electrons can reside in exactly two possible adjustments in the QCA cell. Regarding these two arrangements, they are interpreted as a binary '0' and binary '1', i.e. each cell can be in two states. The state '0' and the state '1', as shown in figure. A binary system is something familiar, as Boolean logic is used already in today's computers. There, a high voltage is often interpreted as binary '1' and a low voltage as binary '0'.

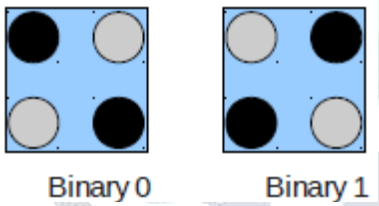


Figure: Binary interpretation of adjustments

Basic QCA elements and gates

So far, we know how to interpret and transport information with QCA cells, but yet we lack the possibility for computations. For QCA cells the basic gate is a three-input majority vote. It is built from five cells, arranged as a cross.

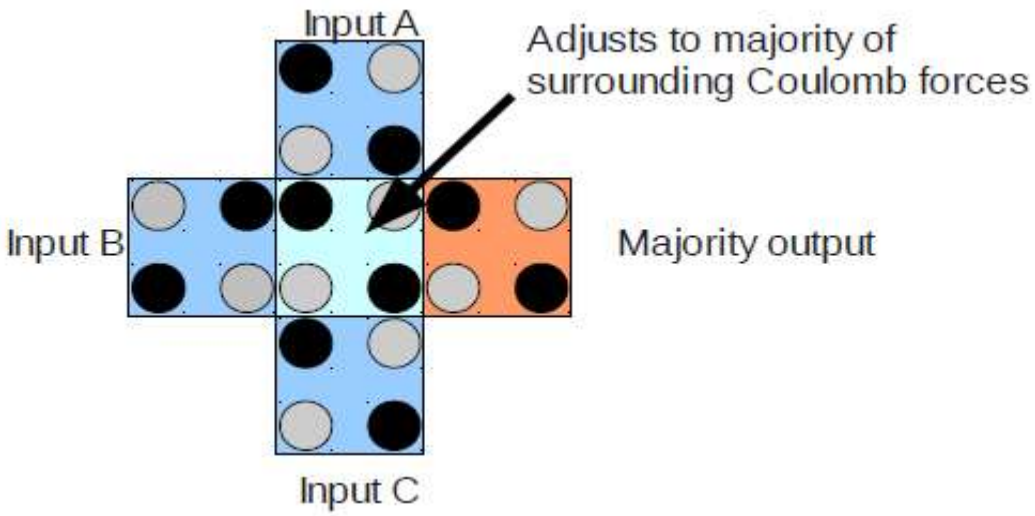


Figure : QCA majority voter

QCA Majority voter

From physics, it is know that the Coulomb forces of several electrons sum up. The majority voter takes advantage of this effect. The cells on top, at the left and at the bottom work as input connection cells. As the Coulomb forces of the electrons of all input cells sum up, the middle cell adjusts to the majority of adjustments of the input connection cells. Finally the output cell adjusts to the middle cell and the resulting state of the majority vote can be obtained from the output cell.

PROPOSED TECHNIQUE:

QCA BASED SCL WITH GDI:

The projected width of a QCA cell for room temperature operation is somewhere in the 5nm realm. For cells of this size it is not likely that we will be able to have small enough clocking zones (three cell width in our adder, which will be approximately 20nm, at most) to have working majority gates. Also, in previously proposed adders the clocking zones are non-uniform in that they do not follow the constriants of the proposed clocking scheme. They do not have uniform, parallel, vertical clocking zones that are required by the use of wires running under the array. They are also very inflexible in that if clocking zones are offset by one or more cells the array will not properly function. These problems can be solved by creating a majority gate that will operate in a single clocking zone regardless of the radius of effect of the cells. Therefore, we now have motivation to construct a majority gate that will operate correctly under a single clock control within multiple radius of effect distances. The construction is not complex, in fact it mearly involves adding (or subtracting) a number of cells to the gate in order to even out the three inputs' interactions with the device cell(s). First discussed will be the majority gate that will be used to handle a radius of effect of up to dNN . It can be seen that the output for a radius of effect of dNN results is a correct output for the gate and also works for d1, both with a single clock. This tolerance is facilitated by the addition of only two cells. However, this configuration does not operate correctly with a radius of effect of d2. For radius of effect of d2 we need to add more cells to the configuration. Once again, the addition of one cell to the design creates a structure that will operate correctly with radius of effects dNN and d2.

Creating QCA structures that will operate at room temperature will require reducing the scale of cells to the molecular level, giving cells sizes of around 2nm [24]. Molecular cells are constructed by connecting redox sites, which can hold a charge, by ligands that allow tunneling between the sites. A simple example of such a molecule is shown in [25] (1,4-diallyl butane radical cation) and has two allyl groups which are connected by a butyl bridge which facilitates the tunneling of electrons and, therefore, the switching of the molecule between basis states. The size of this molecule is 7˚A in length (0.7nm). Placing two of these molecules side by side creates a cell with a total of four allyl groups. These four allyl groups act a the dots which contain charges.

The two-molecule cell is approximately 1nm by 1nm and has the two basis states ("0" and "1") that we need for a typical QCA cell. Cells of this size also have erroneous output in simulation when configured into a left-to-right majority gate.

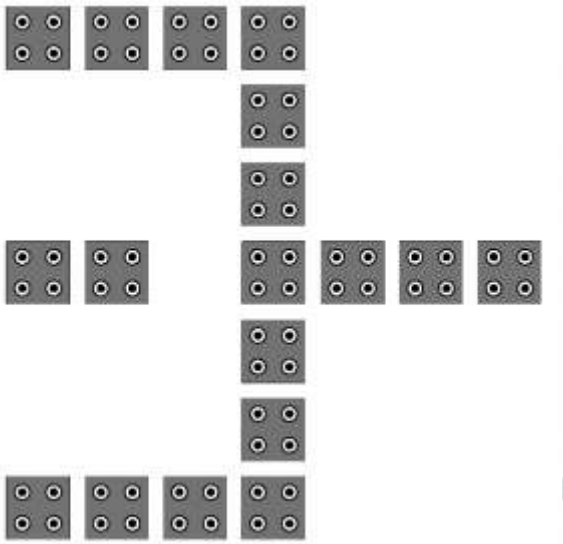
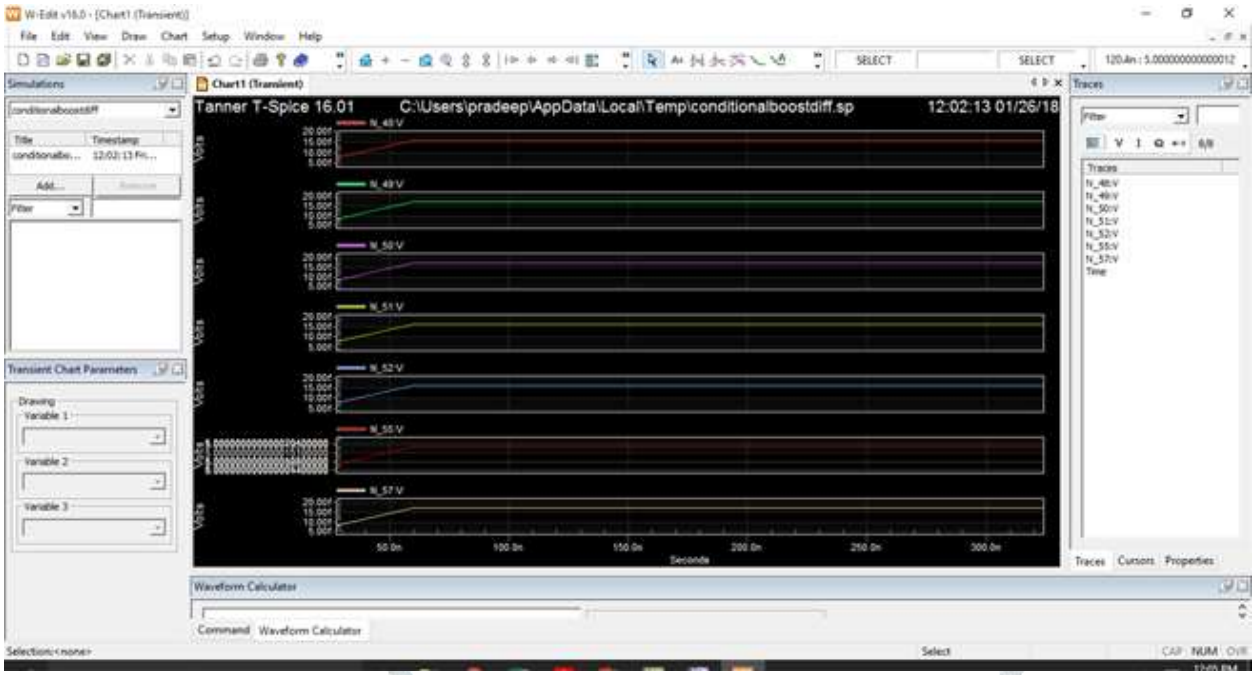
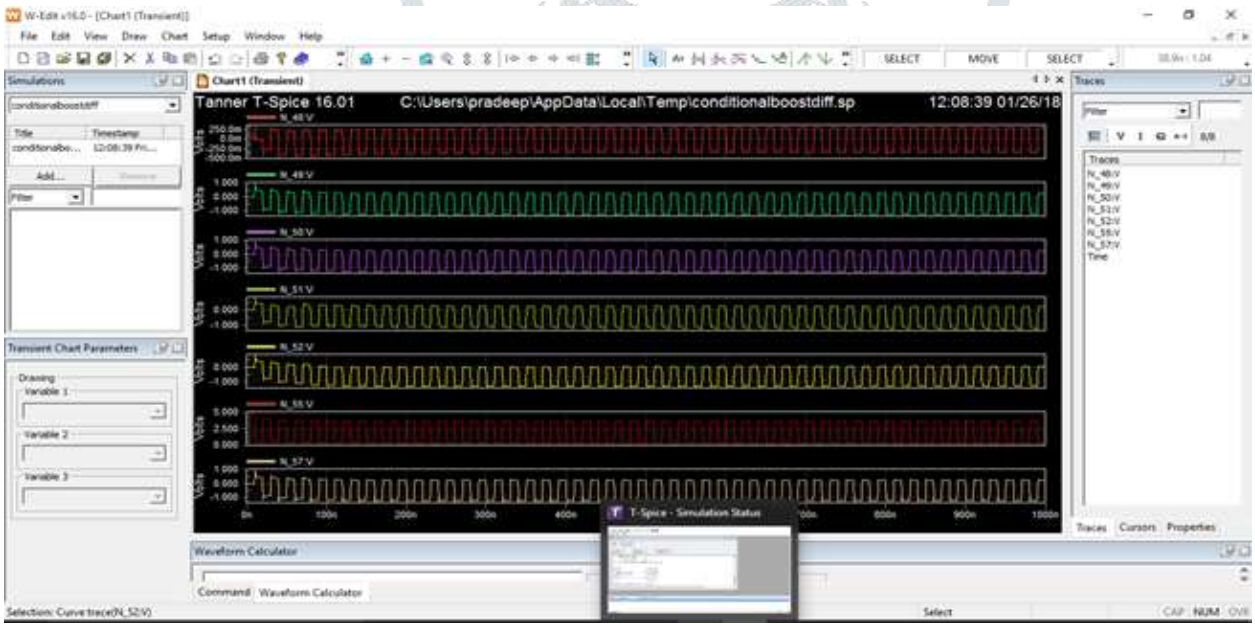


Figure : Functioning left-to-right majority gate for cells of size 1, 2, or 4nm.

Cells of width 1, 2 and 4nm were tested in left-to-right majority gates. The results for these configurations were the same for all three cell sizes and radius of effects d1 to d3. All tests resulted in an output equal to the middle input cell. This indicates that the middle input cell overpowers the other inputs at the device cell, switching the device cell to the middle input value at all times. This is not unlike the errors that occur in configurations with larger cells. To overcome this erroneous functioning we have constructed a majority gate that uses one less cell that functions correctly for all three cell sizes and for radius of effect greater than or equal to dNN . For these simulations we used the coherence-vector simulation as with testing of larger cells. The simulations show that this configuration works only for radii of effect greater than or equal to dNN . This fact is obvious since the middle input will only be able to interact with the device cell if it can interact with cells that are dNN away, due to the missing cell in the middle input wire. Through simulation, we have found that this configuration will work for all radii of effect of concern, and beyond. More accurate simulations were done for molecular implementations of quantumdot cellular automata. The coherence-vector simulation was used to simulate the previously discussed molecular construction using 1,4-diallyl butane. It is stated in [8] that molecular implementations will have a kink energy (Ek) greater than 500 meV. With this in mind a kink energy of Ek = 629.45 (relative permittivity of 0.3) was chosen as an apporximate value for the molecule in question. The value was calculated using equation 10. The simulations were performed at approximate room temperature (300 K). The cell height and width was 1 nm with dot diameter and uniform cell spacing of 0.25 nm.

RESULT:





CONCLUSION:

It has been shown that radius of effect faults occur in the simplest of structures in quantum-dot cellular automata. Under one clock cycle majority gates will provide erroneous results and, therefore, will limit the clocking scheme when placed in larger arrays. To counter these faults we have made minor adjustments to the majority gate. It has been shown that these changes, which are made according to the radius of effect of each individual cell, result in functioning majority gates. It has also been shown that, under simulation, the radius of effect fault-tolerance majority gates will operate for molecular implementations of QCA, which is important due to the fact that a molecular level cell will be needed to create arrays that will function at room temperature. The use of the modified gates in larger arrays of cells was also shown to be successful and benificial to the larger array. These gates will allow for faulttolerant designs with respect to radius of effect as we all facilitate more flexable clocking zone placement and sizing. This research will aid in the creation of large QCA designs that are more physically implimentable than ever before.

REFERENCES:

[1] (2004) QCADesigner: The QCA simulator homepage. [Online]. Available: http://www.atips.ca/projects/qcadesigner

[2] (2004) QCADesigner. [Online]. Available: http://www.qcadesigner.ca

[3] A. Fijany and B. N. Toomarian, "New disign for quantum dots cellular automata to obtain fault tolerant logic gates," J. Nanoparticle Research, 3, (Nov. 2001): 27–37.

[4] J. Timler and C. Lent, "Power gain and dissipation in quantum-dot cellular automata," J. Appl. Phys., 91, (Jan. 2002): 823–831.

[5] G. Toth and C. Lent, "Quantum computing with quantum-dot cellular automata," A Physical Review, 63, (2001)

[6] R. Zhang and K. Walus and W. Wang and G. Jullian, "A Majority Reduction Technique for Adder Structures in Quantum-dot Cellular," SPIE International Symposium on Optical Science and Technology, 5559-10, (2004)

[7] (2004) Basic Tutorial On Quantum-Dot Cellular Automata (QCA). [Online]. Available: http://www.qcadesigner.ca/tutorials/QCATutorial.html

[8] J. Timler and C. Lent, "Maxwell's demon and quantum-dot cellular automata," J. Appl. Phys., 94, (Jul. 2003): 1050–1060.

[9] C. S. Lent and B. Isaksen, "Clocked Molecular Quantum-Dot Cellular Automata," IEEE Trans. Electron Devices, 50(9), (Sep. 2003): 1890–1896.

[10] A. I. Csurgay and W. Porod and C. S. Lent, "Signal Processing with Near-NeighborCoupled Time-Varying Quantum-Dot Arrays," IEEE Trans. Circuits Syst. I, 47(8), (Aug. 2000): 1212–1223.

[11] P. D. Tougaw and C. S. Lent, "Logical devices implemented using quantum cellular automata," J. Appl. Phys., 75, (feb. 1994): 1818–1825.

[12] M. T. Niemier and P. M. Kogge, "Logic in wire: using quantum dots to implement a microprocessor," Proc. IEEE ECS'99, 3, (Pafos, Cyprus, Sep. 1999): 1211–1215. [13] M. T. Niemier, A. F. Rodrigues and P. M. Kogge, "A Potentially Implementable FPGA for Quantum Dot Cellular Automata," Proc. NSC, 1, (Boston, MS, Feb. 2002)

BIBLIOGRAPHY:

N S S R DEEPIKA PATNALA working as Assistant professor at VSM College of Engineering, has completed her M. Tech from VSM College of Engineering with 72% aggregate. She finished her Bachelors from Chaitanya Institute of Engineering and technology with 71% aggregate. She is very much enthusiastic about new trends and technology, participated in many college level competitions and won prizes.

ANALA NARAYANA MURTHY working as Assistant professor at VSM College of Engineering, has completed his M. Tech from Prasiddha College Of Engineering Technology with Distinction. He finished his Bachelors from Adithya college of Engineering with good aggregate. He is very much enthusiastic about new practical exposers, participated in many college level competitions and won prizes.