IT’S ALL ABOUT THE SIGNAL ROUTING: UNDERSTANDING THE RELIABILITY OF QCA CIRCUITS AND SYSTEMS

A Dissertation

Submitted to the Graduate School
of the University of Notre Dame
in Partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy

by

Timothy J. Dysart

Peter M. Kogge, Director

Graduate Program in Computer Science and Engineering
Notre Dame, Indiana
July 2009
Much has been written and predicted about the demise of Moore’s law in advancing computing technology. While many of these predictions have fallen by the wayside, the end may be approaching since the ”red brick wall” (areas where scaling progress may end if breakthroughs are not identified) is barely being pushed further into the future. As a result, numerous nanoelectronic devices are being investigated to determine if they can augment or replace silicon-based transistors. Due to a variety of factors, a significant challenge when building a system from any of these devices is how to produce a reliable, predictable system from unreliable components with unpredictable behavior?

In this dissertation, the reliability aspects of a specific nanoelectronic architecture – Quantum-dot Cellular Automata (QCA) – are explored. QCA is a unique architecture to explore since the same basic device, a simple cell, is used to implement both logic and interconnect components. The device itself can be implemented using different physical properties, which in turn, influences how the signal routing within the QCA circuits and systems should be designed. This dissertation computes the reliability of QCA circuits and systems based on the reliabilities of the underlying components. This is used to identify how reliable
the components should be to achieve a target circuit or system reliability and to identify which components are the most critical to circuit reliability. Additionally, various techniques for improving reliability through the use of hardware redundancy are evaluated to determine how a reliable QCA system should be designed. Lastly, various organizations for a specific interconnect component, a straight wire segment, built from a specific device type, are compared to determine how reliable a component can be. Throughout this dissertation, a constant theme is observed: the organization and reliability of the signal routing has a major impact on circuit and system reliability.
To Pam for her love and patience.
CONTENTS

FIGURES .................................................. vii
TABLES .................................................. xi
SYMBOLS ............................................... xii
ACKNOWLEDGMENTS .................................... xiii

CHAPTER 1: INTRODUCTION ................................. 1
  1.1 Motivation ........................................... 1
  1.2 The Key to a Reliable QCA System ................. 4
  1.3 Results and Contributions ........................... 7
    1.3.1 Activities ..................................... 8
    1.3.2 Contributions .................................. 9
    1.3.3 Insights ...................................... 10
  1.4 Dissertation Roadmap ............................... 11
  1.5 Definitions ....................................... 12

CHAPTER 2: BACKGROUND MATERIAL ..................... 14
  2.1 QCA Basics .......................................... 14
    2.1.1 ESQCA Devices ................................ 14
    2.1.1.1 Metal-dot Devices ........................... 17
    2.1.1.2 Semiconductor Devices ....................... 18
    2.1.1.3 Molecular Devices ........................... 19
    2.1.2 MQCA Devices ................................ 20
    2.1.2.1 Basic Operation ............................. 21
    2.1.2.2 Experimental Work ........................... 23
    2.1.3 QCA Clocking .................................. 23
    2.1.4 Signal Routing Differences ..................... 26
  2.2 Nanoelectronic System Reliability Enhancement .... 27
    2.2.1 Configurable Architectures ..................... 27
2.2.2 Hardware Redundancy ........................................... 29
2.2.2.1 von Neumann Multiplexing ................................. 29
2.2.2.2 N-modular Redundancy .................................... 30
2.3 QCA Centric Reliability ........................................... 36
2.3.1 QCA Defects and Faults ....................................... 36
2.3.2 QCA Component Reliability Enhancement .................... 37
2.3.3 QCA System Reliability Enhancement ......................... 39

CHAPTER 3: MODELING TECHNIQUES ................................. 41
3.1 Probabilistic Transfer Matrices ................................. 42
3.1.1 PTMs and QCA .................................................. 44
3.2 Linear Regression Analysis ....................................... 46
3.3 Geometric Analysis ................................................ 48
3.3.1 Missing Cell Patterns .......................................... 49
3.3.2 Tool Operation ................................................... 51
3.3.3 Computational Speedup ........................................ 51
3.3.4 Extendibility .................................................... 52
3.4 3-State Coherence Vector ......................................... 52
3.5 Other Modeling Techniques ........................................ 55
3.5.1 ESQCA Modeling Techniques ................................. 55
3.5.1.1 2-State Coherence Vector ................................ 55
3.5.1.2 Statistical Mechanical Modeling ......................... 55
3.5.1.3 Bayesian Macromodeling ................................... 57
3.5.1.4 MAquinas ..................................................... 58
3.5.2 MQCA Modeling Techniques ................................. 59
3.6 Correlating Analytical and Physical Modeling ................. 59

CHAPTER 4: CIRCUIT LEVEL ANALYTIC MODELING ............. 62
4.1 Coming Attractions ................................................ 62
4.1.1 Circuit Design Principles ..................................... 62
4.1.2 Fault/Error Rates ............................................. 63
4.1.3 Reading the Results .......................................... 64
4.2 Circuit Designs ..................................................... 64
4.2.1 ESQCA Logical Wire Crossing Circuit ....................... 64
4.2.2 Parity Tree ....................................................... 65
4.2.3 Multiplexers ..................................................... 66
4.2.4 Adder ............................................................ 67
4.2.5 Multiplier ......................................................... 68
4.3 Circuit Complexity ................................................ 69
4.4 Uniform Component Error Rates ............................... 70
4.5 Circuit Complexity versus Reliability ......................... 75
4.6 Perturbing Component Error Rates .............................. 77
### FIGURES

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Fundamental ESQCA cells and components.</td>
</tr>
<tr>
<td>2.2</td>
<td>Additional ESQCA components used in this work.</td>
</tr>
<tr>
<td>2.3</td>
<td>MQCA devices and components.</td>
</tr>
<tr>
<td>2.4</td>
<td>ESQCA clocking example. Subfigures (b) and (c) courtesy of K. Walus, U. of British Columbia.</td>
</tr>
<tr>
<td>2.5</td>
<td>Electrostatic wire crossing.</td>
</tr>
<tr>
<td>2.6</td>
<td>Gate level layout of wire crossing circuit.</td>
</tr>
<tr>
<td>2.7</td>
<td>von Neumann multiplexing techniques.</td>
</tr>
<tr>
<td>2.8</td>
<td>Triple modular redundancy (TMR) configurations.</td>
</tr>
<tr>
<td>2.9</td>
<td>These graphs show ((R_{Rep} - R_{NR})) for cascades of (N) wire slices where both (R_{Rep}, R_{NR} \in [0, 1]). The region in black is where the difference is 0. <strong>Note:</strong> The scale is in steps of 0.1 and changes in each subfigure.</td>
</tr>
<tr>
<td>2.10</td>
<td>Logic to determine a single bit output in QMR.</td>
</tr>
<tr>
<td>2.12</td>
<td>Regular and thick ESQCA wire segments.</td>
</tr>
<tr>
<td>3.1</td>
<td>Majority gate transfer matrices.</td>
</tr>
<tr>
<td>3.2</td>
<td>XOR gate layout and PTM derivation.</td>
</tr>
<tr>
<td>3.3</td>
<td>Sample failure patterns.</td>
</tr>
<tr>
<td>3.4</td>
<td>Calculating circuit reliability from a PTM.</td>
</tr>
<tr>
<td>4.1</td>
<td>Gate layout of ESQCA logical wire crossing circuit.</td>
</tr>
<tr>
<td>4.2</td>
<td>Parity Tree designs.</td>
</tr>
<tr>
<td>4.3</td>
<td>Multiplexer designs.</td>
</tr>
<tr>
<td>4.4</td>
<td>2-Adder design.</td>
</tr>
<tr>
<td>4.5</td>
<td>Adder circuit designs used in the multiplier.</td>
</tr>
</tbody>
</table>
4.6 4x2 multiplier design. ........................................ 70
4.7 Number of components v. number of inputs. ................. 71
4.8 Circuit reliabilities with uniform component error rates. ....... 72
4.9 Circuit reliability as a function of the number of circuit components. Lines are for uniform error rates. .................. 76
4.10 2-Mux reliability as the error rate of specific components is modified by an order of magnitude. ......................... 78
4.11 4-Adder reliability as the error rate of specific components is modified by an order of magnitude. ......................... 79
4.12 Circuit reliabilities with single perfect component as listed in legend. Other components have a uniform error rate. .......... 90
4.13 Circuit reliabilities as the wire component error rate changes (see legend for this rate); other components have a uniform error rate (x-axis). ........................................ 97
5.1 Full adder. .................................................................. 99
5.2 PLA construction. .......................................................... 100
5.3 Circuit reliabilities with uniform component error rates. ....... 102
5.4 Reliability of a two-input, one-output parity tree PTree from Sec. 4.4 and the reliabilities of TMR based systems using a generic two-input, one-output module with a reliability of 0.9999. .......... 109
5.5 NMR reliabilities for modules with \( R_0 = 1 - 1 \times 10^{-4} = 0.9999 \). If the NMR reliability is below \( R_0 \), the redundant system has a lower reliability than the non-redundant system. PIC refers to Perfect InterConnect as is assumed by classical reliability models. For clarity, QMR, PIC is not shown since it overlaps with TMR, NR, PIC. ................................................................. 111
5.6 Cascaded TMR with replicated voters. Repeated from Fig. 2.8(b). 113
5.7 System reliability of a cascaded TMR system with a non-module component error rate of \( 1 \times 10^{-4} \) and \( R_0 = 0.9999 = 1 - 1 \times 10^{-4} \) and \( R_0 = 0.9 \) as shown in the legend. .......................... 114
5.8 Wire Slice Models .......................................................... 123
5.9 Construction of \( PTM_{WS} \) from five unit lengths. ............... 123
5.10 Fault rates for single segment wire slices as the number of unit lengths in the wire segment increases. The fault rate per unit length in the segment is \( 1 \times 10^{-8} \). ................................. 127
5.11 Fault rates for chains of wire slices with a fault rate per unit length of \( 1 \times 10^{-8} \) ....................................................... 129
5.12 Fault rates for wires of a fixed length given a fault rate per unit length of $1 \times 10^{-8}$. Note that the x-axis of these graphs is not to scale.

5.13 Fault rates for wires of a fixed length given a fixed interconnect and voting component fault rate of $1 \times 10^{-4}$. The number in the legend entries is the fault rate per unit length.

6.1 Regular and thick ESQCA wire segments.

6.2 Memory structure segments and sample.

6.3 Yield estimates for various memory structures by aspect ratio.

6.4 Yield estimates for various memory structures by aspect ratio.

6.5 Sample cell (from [61]) that is used here.

6.6 Maximum displacement and rotation defects when at least 90 passing tests were observed for wires of length 25 with various percentages of missing cells. The larger the maximum angle of rotation, the more robust the wire segment.

6.7 Maximum displacement and rotation defects when at least 90 passing tests were observed for wires of length 50 with various percentages of missing cells. The larger the maximum angle of rotation, the more robust the wire segment.

6.8 Maximum displacement and rotation defects when at least 90 passing tests were observed for wires of length 75 with various percentages of missing cells. The larger the maximum angle of rotation, the more robust the wire segment.

6.9 Maximum displacement and rotation defects when at least 90 passing tests were observed for wires of length 100 with various percentages of missing cells. The larger the maximum angle of rotation, the more robust the wire segment.

6.10 Number of passing tests for a range of displacement and rotation defect magnitudes as the percentage of dropped cells changes in a (25,3) wire. *Note:* the range of the x-axis changes in each subfigure.

6.11 Number of passing tests for a range of displacement and rotation defect magnitudes as the percentage of dropped cells changes in a (100,3) wire. *Note:* the range of the x-axis changes in each subfigure.

6.12 Number of passing tests for a range of displacement and rotation defect magnitudes as the percentage of dropped cells changes in a (100,5) wire. *Note:* the range of the x-axis changes in each subfigure.
6.13 Number of passing tests for a range of rotation defect magnitudes for given fixed maximum displacements (labeled MD) as the percentage of dropped cells changes in (25, 1), (25, 3), and (25, 5) wires.

6.14 Number of passing tests for a range of rotation defect magnitudes for given fixed maximum displacements (labeled MD) as the percentage of dropped cells changes in (100, 1), (100, 3), and (100, 5) wires.

6.15 Number of passing tests for a range of rotation defect magnitudes for a given fixed maximum displacement (labeled MD) as the percentage of dropped cells changes in all four lengths of a three wide wire.
2.1 NUMBER OF WIRE CROSSINGS REQUIRED FOR VARIOUS NMR METHODS.  

4.1 UNIFORM ERROR RATE CORRESPONDING TO FIVE NINES RELIABILITY  

4.2 LINEAR REGRESSION RESULTS  

4.3 NORMALIZED LINEAR REGRESSION RESULTS  

4.4 COMPONENT IMPORTANCE ORDERINGS (IN DESCENDING ORDER).  

4.5 MAGNETIC MULTIPLIER RELIABILITY WITH ZERO OR ONE PERFECT COMPONENT.  

4.6 ELECTROSTATIC MULTIPLIER RELIABILITY WITH ZERO OR ONE PERFECT COMPONENT.  

5.1 CUSTOM LOGIC AND PLA ADDER COMPONENT COUNTS.  

5.2 LARGEST NON-MODULE COMPONENT ERROR RATE WHERE THE RELIABILITY OF THE NMR VERSION IS GREATER THAN $R_{gm} = 0.9999$.  

5.3 REGRESSION COEFFICIENTS FOR TWO INPUT MODULES.  

5.4 REGRESSION COEFFICIENTS FOR THREE INPUT MODULES.
## SYMBOLS

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASIC</td>
<td>Application Specific Integrated Circuit</td>
</tr>
<tr>
<td>ESQCA</td>
<td>Electrostatic QCA</td>
</tr>
<tr>
<td>FPGA</td>
<td>Field-Programmable Gate Array</td>
</tr>
<tr>
<td>MQCA</td>
<td>Magnetic QCA</td>
</tr>
<tr>
<td>NMR</td>
<td>N-Modular Redundancy</td>
</tr>
<tr>
<td>PLA</td>
<td>Programmable Logic Array</td>
</tr>
<tr>
<td>QCA</td>
<td>Quantum-dot Cellular Automata</td>
</tr>
<tr>
<td>QMR</td>
<td>Quintuple Modular Redundancy</td>
</tr>
<tr>
<td>RCR</td>
<td>Random Circuit Reliability</td>
</tr>
<tr>
<td>TMR</td>
<td>Triple Modular Redundancy</td>
</tr>
</tbody>
</table>
ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Peter Kogge, for his support and guidance while I pursued this research. In particular, thank you for ensuring that I saw the big picture and allowing me the freedom to study the occasional stray, yet interesting, path.

An additional thank you to Dr. Jay Brockman, Dr. Sharon Hu, and Dr. Michael Niemier for serving on my committee. Your insights and suggestions have strengthened the content of this work. Similar acknowledgments are due to Dr. Eugene Henry and Dr. Marya Lieberman for their support when I initially proposed this work.

I would also like to recognize Dr. Douglas Thain and the staff of Notre Dame’s Center for Research Computing for providing assistance as I used and abused the various computing resources available on campus.

To the staff of the CSE department, particularly Curt Freeland, MD McNally, Jane Wroblewski, and Joyce Yeats: thank you for ensuring that our computing systems run smoothly and that the hassles of paperwork and bureaucracy were non-existent.

A special thank you is also due to my parents and my wife whose love, patience, and support enabled me to survive nearly a quarter-century of schooling. While you may have thought at times that this day may never come, it is here.

Additionally, a thanks to the many denizens of Cushing 222 over the years for
your thoughts and discussions – from work related topics to Notre Dame football to the occasional vortex.

Lastly, I would like to thank the many sponsors of my research over the years. These have included the Jet Propulsion Laboratory, the Semiconductor Research Corporation (grant 2000HJ836), the National Science Foundation (grant CCR-0210153), and the U.S. Department of Defense.
CHAPTER 1

INTRODUCTION

1.1 Motivation

Much of the progress of digital electronics over the last forty plus years has been driven by achieving a common corollary of Moore’s law stating that the number of transistors per die (chip) should double every 18 to 24 months. To continually reach these targets, only two options are available: enlarge the die or make the transistors smaller. Because of the negative impacts that can occur, specifically lower yields, when the die size is increased, the semiconductor industry has focused on decreasing the size of transistors. As a result of this continual march, transistors are now well into the nanoscale and can easily be considered nanoelectronic devices\(^1\).

As a result of this scaling, at least for the context of this work – building reliable circuits and systems, there are two significant challenges. The first of these is related to the yield (percentage of properly functioning die) that can be achieved during the manufacturing process. Due to the size of the transistors and the fabrication processes required, placing perfectly sized and shaped devices exactly where they should be located is extremely challenging. There is an entire field of research, design for manufacturing (aka design for yield), dedicated to

\(^1\)This assumes the common definition of nanotechnology as dealing with structures under 100nm in size
reducing the impact of this inherent uncertainty in fabricating integrated circuits. In essence, the goal of design for manufacturing is to build defect tolerance into the transistors.

The second significant challenge is that once these integrated circuits have been proven to function correctly, they need to continue operating correctly (reliably) in the field. Due to the size of the transistors, only a few tens of electrons may actively participate in the computation thus leaving the transistors vulnerable to strikes from energetic outside particles. If these strikes cause an error in the computation a single event upset occurs; without any way to identify or correct for these errors, the impacts can vary from harmless to invalid simulation results to an entire system crash. By introducing fault detection and/or tolerance into the system, such as the error correcting codes used with memory, the impacts of these single event upsets can be reduced or eliminated.

One of the major limitations of this continued transistor scaling is that it may end in the near future. If this happens, it is more likely to be the result of fundamental physical limits rather than an engineering challenge because some of the components in the transistors (e.g. gate oxides) are already a few atomic layers thick which leaves almost no material to be shrunk. A large number of semiconductor manufacturers have worked together to develop a roadmap, the International Technology Roadmap for Semiconductors (ITRS) [47], which details the fabrication parameters that are required in order to keep scaling transistors in accordance with Moore’s law. Within the ITRS, the parameters are shown in tabular form and are labeled according to whether a manufacturing (high speed/throughput) solution is known or not; if a solution is not known the parameter is labeled with a red box. The “red brick wall,” or when manufacturing solutions for most of the
parameters in a table are not known, is relatively near for certain aspects of the
fabrication process. When this red brick wall is reached for most aspects of the
fabrication process, continued transistor scaling is likely to cease.

Scaling transistors has not only caused the cost per transistor to be reduced,
but it has also enabled significant performance increases. With more transistors
on a die, more computations can be performed. A smaller transistor (generally)
requires less power to perform a computation than a larger one. Performance as-
pects such as these have been another major driver for the semiconductor industry,
and even if transistor scaling ceases, the industry will continue to investigate de-
vices/methods for improving the performance of integrated circuits. Performance
in this context could be raw computational power, but it could also include metrics
such as joules per operation or manufacturing cost per logic gate.

A range of nanoelectronic devices are being investigated to determine how they
may enable performance improvements if they are used to augment or replace
silicon-based integrated circuits. As discussed in [8], these devices may “allow
orders-of-magnitude improvements in device density and complexity.” Bahar et
al. also list three critical questions regarding these devices:

• How will we use these huge numbers of devices?

• How must we modify and improve design tools and methodologies to accom-
  modate radical new ways of computing?

• Can we produce reliable, predictable systems from unreliable components
  with unpredictable behavior?

The last of these questions is a concern because many of the devices being in-
vestigated utilize optic, molecular, or quantum principles to operate. Additionally,
these devices may be integrated with nanoscale silicon which introduces process variability (e.g. size of doped regions may vary across a die) as another source of unpredictable components. As a result, it may not be unusual for nanoelectronic systems to have defect and fault rates that are several orders of magnitude larger than that of today's silicon-based systems.

Overall, the last question from Bahar et al. really asks, for a given nanoelectronic system, what does a reliable system look like at the component, circuit, and system levels? What forms of defect and fault tolerance should be included to ensure reliable computation? This dissertation will examine and answer this question for systems built from quantum-dot cellular automata (QCA) devices.

QCA is a nanoelectronic device technology that utilizes arrays of locally interacting cells to perform computation. These cells are simple devices that interact with one another via Coulombic (electrostatic) or magnetic forces rather than carrying current (as transistors do). By placing cells in specific arrangements, they can perform either logic functions or act as interconnect (wire) components. These properties, coupled with how QCA circuits are manufactured, have unique implications when examining how to produce a reliable QCA system.

1.2 The Key to a Reliable QCA System

The major result of this dissertation is summarized in its title “It’s all about the signal routing.” Two questions result from this conclusion: 1) what is signal routing? and 2) why is it unique in QCA?

Signal routing, in simpler terms, is simply interconnect, or in even more basic terms, wires. In terms of what happens in a computational structure, signal routing is what takes a set of bits from one set of locations, e.g. the outputs of a
set of gates, to another set of locations, e.g. the inputs to the next set of gates. In CMOS circuits, signal routing is accomplished through layers of metal, polysilicon, and vias. Obviously, this network of metal layers and vias is more complex than, but functionally equivalent to, a set of wires. In QCA systems, signal routing is just a specifically arranged group of cells, but that imposes its own challenges.

There are several aspects that make signal routing unique in QCA. First, QCA systems, as currently envisioned, are planar (two-dimensional); rather than being able to use multiple layers of metal as is done in CMOS circuits, all of the QCA cells exist on the same plane. Second, the structure of the interconnect may be different for circuits built from electrostatic QCA (ESQCA) cells than for circuits built from magnetic QCA cells (MQCA). Third, within a QCA circuit, the logic devices and the interconnect devices are the same; this implies that at a fundamental level logic and interconnect are equivalent. Finally, unlike CMOS, the clocking needed to control complex QCA circuits is fundamentally different and separate from the processes that perform the logic. This is different from the CMOS model which uses transistors for logic and clocking, and layers of metal and vias for interconnect.

The challenge with having two-dimensional, rather than three-dimensional, circuits is that when two signals (bits) need to cross, they need to occupy the same space at the same time. In a three-dimensional circuit, these signals could occupy the same location in the plane and be in different layers of metal to avoid needing to share the same space at the same time. Obviously, for QCA circuits, some mechanism for crossing wires is necessary.

For ESQCA circuits, this mechanism consisted of a wire of rotated cells passing through a gap in a wire of unrotated cells. To achieve room temperature operation
in ESQCA circuits, molecular devices will be needed. For this crossing to function properly using molecular cells sub-Angstrom precision in placing the cells would be necessary. Since this level of control is unlikely to be feasible, a logical wire crossing (combinational circuit) that does not require rotated cells can be used instead.

For MQCA circuits, a single magnet may be sufficient to cross wires in the plane. This may greatly simplify MQCA circuits designs when compared to ES-QCA circuit designs (using the logical crossing), and as will be shown in this dissertation, can have a significant impact on reliability between the two QCA implementations.

This impact on reliability exists because of the device sharing between logic and interconnect in QCA circuits. The classical reliability models, such as those for N-modular redundancy, generally treated logic as potentially faulty and signal routing to be either perfect or, if it was potentially faulty, rolled it into the fault rates of the logic components. More recent modeling regarding signal routing has focused on when the wires will break due to factors like electromigration [9]. This type of modeling fails to treat signal routing as another logic component as needs to be done in QCA circuits.

The signal routing components (e.g. fanouts, wire segments) need to be treated in the same manner as logic components when computing reliability because they are functionally equivalent; both types of components are built from cells arranged in specific patterns. Additionally, since faults in QCA components tend to be inversions there may be situations where two inversions in a circuit cancel each other out (i.e. two wrongs make a right). The analytic modeling technique used in this work is fully capable of treating interconnect components like logic components
and accounting for two wrongs making a right.

1.3 Results and Contributions

The overarching theme and main contribution of this dissertation is that signal routing is the key driver for determining the reliability of QCA circuits and systems.

While this work is in many ways a continuation of the work done in [23] which advanced the state of QCA design tools and analyzed how various defects impacted ESQCA wire segments, several questions remained. These wire segments varied in their width, which is a potential mechanism for increasing their robustness (tolerance to defects), and the major question that remained was: how much robustness is necessary? If a five cell wide wire is more robust than a three cell wide wire, which one should actually be used? In other words, when attempting to make a component more robust, at what point is it reliable enough, i.e. is there a target level of reliability that needs to be reached?

Answering this question was the major reason for doing the analytic modeling with probabilistic transfer matrices (PTMs) that is described in this work. PTMs are utilized because they are capable of treating logic and interconnect equivalently, a necessary requirement when computing QCA circuit and system reliabilities. Using PTMs also enabled rapid studies of circuit reliability and an understanding of how reliable the circuit components need to be to achieve a target circuit reliability. By extending this technique and using regression analysis, the component(s) most critical to the reliability of a circuit could be determined. By comparing ESQCA circuits with logical wire crossings to their MQCA counterparts using wire crossing devices, it is clear that introducing additional complexity
into the signal routing of a circuit has a noticeably negative effect on circuit reliability.

Since the defect rates for nanoelectronic circuits are likely to be quite high as compared to those for CMOS, it may be necessary to utilize some form of fault tolerance. These techniques, commonly involving configurable systems and/or hardware redundancy, not only provide fault tolerance while the circuit is operating, but also provide a secondary layer of defect tolerance. In this work, two forms of fault tolerance are analyzed using PTMs: N-modular redundancy and programmable logic arrays. N-modular redundancy is a natural fit for QCA logic since it utilizes majority voting logic to determine an output and the basic gate in QCA is a three-input majority gate. The results contained in this work suggest that these techniques may be of limited value since they are heavily dependent on large amounts of signal routing, particularly as larger systems are considered.

The modeling with PTMs at the circuit and system level answered how reliable components should be for a target circuit/system reliability and identified which components are most critical within the circuit or system. In the latter part of this work, a critical component, the wire segment, is physically modeled in the presence of defects to determine how reliable a component can be.

1.3.1 Activities

The list below summarizes the activities contained in this dissertation.

- A variety of combinational circuits implemented in ESQCA and MQCA were modeled via PTMs to calculate their reliability when built from faulty components.
- The most critical component(s) of a variety of combinational circuits imple-
mented in ESQCA and MQCA is determined.

- The reliability of ESQCA and MQCA systems that implement N-modular redundancy with generic modules is computed via PTMs. For this contribution several types of N-modular redundancy are investigated.

- Initial results comparing the reliability of custom logic circuits to programmable logic circuits for ESQCA and MQCA are presented.

- Three potential methods for organizing long MQCA wires are evaluated.

- Reliability estimates for memory-like ESQCA wire segments with missing cells are calculated via a scalable new method.

- Reliability estimates for ESQCA wire segments with multiple defect types are computed via detailed physical simulation.

1.3.2 Contributions

The following contributions will be presented in this dissertation.

- An analytic modeling method from the literature is utilized to compute QCA circuit and system reliabilities when both logic and interconnect components are faulty.

- A method for identifying the critical components of a QCA circuit is presented. This method will show that wires and majority gates are the most critical components in QCA circuits.

- Preliminary results will suggest that custom logic circuits may be preferable to PLA based circuits from the reliability perspective.
• NMR will be shown to improve the reliability of MQCA systems but is unlikely to improve the reliability of ESQCA systems.

• Guidelines for designing highly reliable long MQCA wires are provided.

• A new modeling method for quickly estimating the reliability of an ESQCA multi-bend wire is used to show that the reliability of the wire is independent of the wire segment length between bends.

• Physical simulation is used to identify the robustness of various ESQCA wire segments in the presence of multiple defect types.

1.3.3 Insights

Several insights regarding QCA circuit modeling and reliability have been developed during the course of this work. These are listed below.

• Mathematical (analytic) modeling of QCA circuits is feasible and methods for identifying critical components have been developed. This type of modeling should be applicable to other nanoelectronic devices.

• The use of system level reliability techniques, such as space redundancy and regular logic, may have a negative effect on reliability. As such, new methods or blends of current techniques may be required depending on the technology utilized.

• The study of nanoelectronic circuits can be done with various abstractions. In particular, applying the lessons learned from ULSI/VLSI circuit design and development can prevent the need to reinvent the wheel.
• ESQCA wires can operate properly in the presence of manufacturing defects and it is possible to make them more robust by increasing the width of the wire. Additionally, bent ESQCA wires are very sensitive to defects in the inner corner region.

• The major insight that should be taken away from this work is that signal routing is THE driver for QCA circuit reliability.

1.4 Dissertation Roadmap

This dissertation is divided into seven chapters, including this one. The remaining chapters are outlined below.

• Chapter 2 introduces the basic principles of QCA, techniques for providing reliability enhancement in other nanoelectronic technologies, and summarizes previous work on QCA based reliability enhancement.

• Chapter 3 describes the various modeling techniques used in this dissertation along with a brief overview of other QCA modeling techniques. The techniques described in the first two sections of this chapter are utilized in Chapters 4 and 5 while the techniques described in the succeeding two sections (3.3 and 3.4) are used in Chapter 6.

• Chapter 4 develops an understanding of QCA circuit reliability, via analytic modeling, for a variety of circuits built from faulty components. Additionally, the circuits are analyzed to determine which component(s) have the largest impact on circuit reliability. This chapter includes the results from [24, 25, 27].
• Chapter 5 moves beyond the circuit level and examines how various system reliability techniques may impact reliability via analytic modeling. Additionally, guidelines for designing a reliable MQCA wire are identified. This chapter includes results from [26, 28–30].

• Chapter 6 investigates the robustness of straight ESQCA wire segments of varying lengths as the width of segments changes via physical modeling. The results of Chapters 4 and 5 aid in determining how reliable a component should be to achieve a specific circuit/system reliability, but do not answer how reliable a component can be. The answer to this question, how reliable can a component be, is examined in this chapter. This chapter includes results from [32, 33]

• Chapter 7 concludes this dissertation and identifies directions for future work.

1.5 Definitions

The following terms are used in this work and are defined here for simplicity.

• A device is the fundamental building block of an electronic system. Transistors and QCA cells are examples of devices.

• A component is built from one or more devices and implements a simple, fundamental logical function. Wire segments and logic gates are examples of components.

• A circuit is a collection of components that are arranged to compute a specific logical function. Adders and multiplexers are examples of circuits.
• For this work, a system is a collection of multiple circuits that are combined together into a higher level function. In general, system has been used as a catch-all term to describe a structure that is more complex than a circuit.

• A defect generally occurs at the device level and is defined to be the difference between what was intended to be manufactured and what was manufactured.

• Fault and error have generally been used interchangeably in this work. Both of these terms describe the case when the output of the component, circuit, or system is different from the one desired. Similarly, fault rate and error rate are used to describe how frequently a fault (error) may occur.

• The terms fail and failure are used to describe a component or circuit under test that has shown an error.

• Reliability is a measure of how frequently the desired output of a component, circuit, or system is, or expected to be, observed. Reliability can also be considered to be (1 - fault rate).

• Yield is a measure of how many good components or circuits result from a manufacturing process.

• Defect tolerance is used to describe a device, component, circuit, or systems that will not fail in the presence of a defect or defects. Typically, this term is used at the device and component level.

• Fault tolerance is used to describe a circuit or system capable of correcting or hiding a fault or error.

• Robustness is used to describe the defect tolerance of a device or component.
CHAPTER 2
BACKGROUND MATERIAL

This chapter has three major sections dedicated to the background material needed to understand this work. The first major section will describe the basics regarding QCA. The second major section will describe common reliability enhancement techniques proposed/used in nanoelectronic circuits and systems. The last major section will survey previous efforts to understand and improve reliability in QCA.

2.1 QCA Basics

This section will cover the basics regarding QCA devices and architectures. The first two subsections describe the basic device types as based on the physical state variable: electrostatic QCA (ESQCA) and magnetic QCA (MQCA). The third subsection describes how QCA circuits are clocked in order to transmit information. The last subsection highlights the potential differences in implementing signal routing for electrostatic QCA and magnetic QCA if room temperature operation is desired.

2.1.1 ESQCA Devices

In the first part of this section, the basic operation of ESQCA cells is described. The following subsections will describe experimental efforts on metal-dot, semi-
conductor, and molecular ESQCA devices.

In ESQCA, the arrangement of electronic charge is used to store and transmit information [55]. A single cell has four active quantum dots in the corners of the cell. Two free electrons are then able to locate on the quantum dots. Due to Coulombic interactions, these electrons will tend to occupy diagonal corners in the cell. Since these two diagonal configurations are stable, binary computing can occur. The term polarization \( P \) is used to describe how the electrons are arranged in the cell. Two ESQCA cells, one with each polarization, are shown in Fig. 2.1(a). For simplicity, a cell with \( P = -1 \) is considered a binary 0 and a cell with \( P = 1 \) considered a binary 1.

Depending on the specific device type, a mechanism for controlling the movement of the free electrons is utilized. Additionally, an extra pair of quantum dots may be present in each column of cells; these are used to hold the cell in a null (non-polarized) state. For metal-dot based cells, a clock-controlled potential barrier determines if the electrons are free to move between dots or if the electrons are localized on separate dots. For molecular cells, the null dots are likely to be below the active dots (forming a V-shaped column of dots), and thus a clock-controlled electric field can either pull the electrons into the null dots or push the electrons into the active dots.

To create the basic components needed to transmit information, cells are placed in close proximity to one another. This enables neighboring cells to interact via Coulombic forces. The specific arrangement of cells determines which basic component is formed. The three main components, which form a functionally complete logic set, are shown in Figs. 2.1(b)-2.1(d).

The first component, a wire segment is shown in Fig. 2.1(b). The second
component, the majority gate of Fig. 2.1(c), is the fundamental gate used in QCA, and implements the voting logic function $AB + BC + AC$. Holding one of the inputs to zero forms an AND gate, while holding an input to one forms an OR gate for the remaining two inputs. Since cells that are diagonal from one another tend to hold opposing polarizations, inverters can also be constructed as is shown in Fig. 2.1(d). A variety of other components are also used throughout this work, and are shown below in Fig. 2.2.

The literature regarding systems designed using ESQCA cells is quite rich and includes combinational circuits, several forms of memory, programmable logic structures, and microprocessors [18, 35, 48, 56, 71, 97, 102]. The cited papers
should not be considered an all-inclusive list, but just a sampling of the literature.

2.1.1.1 Metal-dot Devices

Much of the early work on experimental QCA devices utilized metal-dot cells. These metal-dot systems use Al islands with Al/AlO$_x$/Al tunnel junctions that were pioneered by Fulton and Dolan [37]. Once the detection of a single electron moving between dots could be obtained [4], an individual QCA cell was built and tested [5, 75, 86, 87]. The next steps developed a QCA wire [76] and the majority gate [6]. The first clocked QCA circuit, a cell, was tested in [3]. All of
these previous experiments had source and drain leads for the dots, whereas the cell tested in [7] removed these leads. This change forced the cell to utilize the single free electron, while simplifying the manufacturing process and enabling a greater output polarization. A clocked cell acting as a latch was tested in [74]. More recent progress has demonstrated clocked shift registers (wires) [54, 77] and a clocked fanout structure [107].

These metal dot systems have several downsides. For example, the cells tend to be large (5-10$\mu$m$^2$ in [54]), challenging to fabricate, require extremely low temperature operation (generally < 100mK), and need to be in the presence of a magnetic field to suppress the superconductivity of Al. These difficulties have led researchers to investigate other potential QCA devices, particularly since room temperature operation is desired.

2.1.1.2 Semiconductor Devices

Recently, a group has reported the operation of a Si-based QCA cell [65]. This cell is similar in nature to that of the metal dot cells tested above, but uses heavily doped ($n^+$) Si regions for the dots rather than Al dots. These dots are approximately 500nm x 80nm in size. As in the metal-dot case, a low temperature and the presence of a magnetic field were necessary to ensure the operation of the cell.

In addition to Si-based QCA cells, another group has reported QCA-like operation in a cell using GaAS/AlGaAs dots [38]. As in the metal-dot case, low temperature operation was required.
2.1.1.3 Molecular Devices

Molecular based QCA devices have been investigated due to a (potentially) simpler manufacturing process, small size (high device density), potential for room temperature operation, and potential for high speed circuits. Early descriptions of molecular QCA can be found in [57, 60]. Room temperature operation of molecular QCA is described in [103] and the potential for high speed clocking is discussed in [93].

There have been two major tracks for developing molecular QCA devices and circuits: 1) a method for attaching QCA molecules to a substrate and 2) developing the QCA molecules themselves. These tracks are outlined below.

The proposed method for attaching QCA molecules to a substrate involves a mix of bottom-up self assembly and top-down lithography processes. In the bottom-up process, DNA tiles create a grid-like surface, or scaffolding, and then groups of these tiles are joined together to form a “nanobreadboard.” A molecular QCA circuit can then be created by allowing the QCA molecules to self-assemble on the DNA structure, thus forming a self-assembled monolayer (SAM).

DNA based structures have been proposed for use in the manufacture of several molecular electronic systems [22, 81]. These structures are used because DNA can provide a regular pattern for the attachment of molecules with approximately 3.5nm between attachment sites in the horizontal direction and 2nm in the vertical direction. These DNA-based structures, commonly called tiles, can also be joined together to form larger structures [106] that are typically called grids or rafts.

Using DNA tiling for molecular QCA has focused on creating DNA rafts and precisely placing these rafts onto a silicon substrate. [84] first creates a raft of 4 DNA tiles while [44] places these rafts into a silicon trench. The rafts are
approximately 8 nm x 37 nm while the trenches can be made 5-10 nm wide. This combination suggests that natively defect-tolerant components, such as wide wires (described later in this chapter), can be built.

The top-down assembly will be required to imprint a pattern onto silicon, most likely using electron beam lithography (EBL), and then placing material on the silicon in the desired locations. These methods have successfully attached gold, QCA-like molecules, and DNA rafts [41, 44, 79] to silicon based substrates.

The other track, developing the QCA molecules, has focused on the development of candidate molecules. Several have been developed and analyzed in [49, 50, 58, 59, 64]. An exciting development for molecular QCA systems is the controlled switching of a potential QCA molecule [80].

There are a variety of issues that are common in molecular electronics. The first challenge is in designing and producing an appropriate molecule. Another challenge is in developing the input and output structures required to interact with the molecules. Additionally, there must be enough control over the fabrication process to ensure the precise orientation and alignment of the molecules, particularly in QCA.

2.1.2 MQCA Devices

In this section, the principles behind the operation of magnetic QCA devices will be explored. Additionally, experimental work demonstrating MQCA operation will be described.
2.1.2.1 Basic Operation

MQCA uses the alignment of magnetic dipoles, rather than the configuration of electronic charge, to store and transmit information among neighboring cells. In general, rectangular nanomagnets are used as the basic devices. Due to the shape anisotropy\(^1\) of the rectangular magnet, the magnetic dipole (magnetization vector) is in its lowest energy state when it is aligned along the long axis of the rectangle. Since this is the lowest energy state, it results in a stable, non-volatile value for the magnet. A cartoon of the basic device is shown in Fig. 2.3(a). In this figure, both possible polarizations are shown and the long axis of the rectangle is vertical.

---

\(^1\)Shape anisotropy occurs when the length of one dimension of the magnet differs from another dimension of the magnet. For example, a square does not demonstrate shape anisotropy, but a rectangle does.

---

Figure 2.3. MQCA devices and components.
If a group of magnets is aligned horizontally, as shown in Fig. 2.3(b), a wire can be formed. This wire shows magnets that are anti-ferromagnetically coupled together. If the magnets were aligned in the form of a vertical wire, the magnets would be ferromagnetically coupled (i.e. each magnet would have the same magnetization vector). If a horizontal wire consists of an even number of magnets, it operates like a wire; however, if it were to have an odd number of magnets, it will operate like an inverter.

The last component needed for general purpose computation is an AND or OR gate. Similar to the ESQCA case, the basic logic gate for MQCA is the majority gate and is shown in Fig. 2.3(c). However, in this case, the middle input \( B \) is inverted since it is anti-ferromagnetically coupled to the center magnet of the majority gate whereas the other inputs are ferromagnetically coupled to the center magnet of the gate. As before, an input can be specifically programmed to cause this gate to operate as an AND or OR gate.

In ESQCA, the clocking mechanism ensured that data was stored in a specific arrangement. In MQCA, the clocking mechanism is used to erase stored data rather than lock it into a specific configuration. Recall that without any external influences a rectangular magnet will hold a binary value indefinitely. To change the state of a magnet, it is first nulled via an external magnetic field (provided via the clocking mechanism) that forces the magnetization vector of the magnet to align with its short axis. When this external field is removed, the magnetization vector will align along its long axis based on the magnetic dipoles of its near neighbors.
2.1.2.2 Experimental Work

The first experimental work demonstrating the use of magnetic dots to implement the QCA paradigm was shown in [16]. In this work, a wire segment built from 69 circular dots (rather than rectangular magnets) is presented. These dots were approximately 110nm in diameter with approximately 25nm between dots (135nm pitch). The wire segment was tested and shown to work at room temperature.

A more recent work using oval/rectangular shaped magnets demonstrated the operation of both an MQCA wire and a majority gate [46]. In this work, the magnets are approximately 70nm by 135nm and are separated by 25nm (95nm pitch). Again, room temperature operation was demonstrated.

Further experimental work on MQCA is also described in [45].

2.1.3 QCA Clocking

To effectively move data across a QCA circuit, a clocking structure is necessary [56]. This structure, typically buried in the substrate below the QCA devices [43, 72], utilizes multiple clocking signals that are out of phase with one another, with only one signal allowing active computation at any given moment. Typically, four clock signals are utilized with each signal being in a unique phase. These phases are switch, hold, release, and relax. By organizing the clocking structure in this manner, QCA circuits operate as large shift registers; this implies that they can be viewed as very fine-grained pipelines utilizing the principle of pipelining-in-wire [70].

The clocking signals are assumed to operate in an adiabatic fashion. That is, the transition between the clock high and clock low states is slow (gentle) to
ensure that the system remains near its ground state energy at all times.

In ESQCA, a clock signal is an electric field perpendicular to the dots (cells) which controls the localization of the dots. The clock phases correlate to the state of the electric field acting on the cells and the phases are named for the state of the cells during the phase. During the hold phase, a positive electric field is maintained which causes the free electrons to remain localized on specific dots. This provides the cell a set polarization that can then be used to drive neighboring cells. Cells in this state, or cells with a static polarization, are commonly referred to as driver cells. During the release phrase, the electric field is lowered and a negative electric field is seen by the cells. The free electrons no longer remain localized during this phase. During the relax phase, the cells see a negative electric field and the free electrons remain unlocalized. The final phase, switch, raises the electric field and the cells begin to see a positive electric field. This causes the free electrons to localize on the dots and the hold phase begins anew.

An example of ESQCA clocking is shown in Fig. 2.4. A single clocking signal and its associated phases before and after a transition are shown in Fig. 2.4(a). The four clock signals used to move data in a wire are shown in Fig. 2.4(b). Fig. 2.4(c) shows how data moves in a wire that has been divided into eight clocking zones and operates in a shift-register like fashion using all four clock signals. This wire segment maintains two bits of data since two of the D-latches will be in the hold phase at any point in time.

Clocking in MQCA is slightly different in that rather than using a clock signal to latch data, the clock signal is used to erase data as described in Sec. 2.1.2.1. Additionally, the clock signal applies a magnetic field (rather than an electric field) across the magnets. For MQCA clocking, the labeling of Fig. 2.4(a) would swap...
the switch and release phases and similarly, the hold and relax phases would be
swapped. The operation of the clocked wire would then remain the same.

It should also be noted that since the clocking wires for both QCA technologies
are likely to be buried in a Si-based substrate, the size and spacing of the clocking
wires will be subject to CMOS fabrication process rules.
2.1.4 Signal Routing Differences

Signal routing, or moving a set of signals (bits) from a set of points to another set of points is a unique challenge in QCA. This challenge results from QCA being a planar technology, which means that signals cannot be routed up and around as they are in CMOS with vias and multiple layers of metal. This difference between CMOS and QCA means that two bits must be able to cross in the same layer of QCA cells. In early ESQCA circuit design, the wire crossing of Fig. 2.5 was proposed to enable two signals to cross in the plane. This crossing utilized a wire with unrotated (90 degree wire) and rotated cells (45 degree wire) where the value in the wire with unrotated cells “hopped” over the wire with rotated cells. The challenge with this crossing is that if molecular cells are used, then each cell will need to be placed with sub-Angstrom level precision for the crossover to function properly [14].

To alleviate this problem, a simple combinational circuit capable of crossing two signals that does not require rotated cells can be constructed [13]. This combinational circuit is described in more detail in Sec. 4.2.1 and computes two outputs: \((AXORB)XORA = B\) and \((AXORB)XORB = A\) (as shown in Fig. 2.6).

In MQCA circuit design, a specific arrangements of magnets capable of crossing

Figure 2.5. Electrostatic wire crossing.
wires is currently being investigated [72]. Having an explicit structure capable of crossing wires in the plane avoids having to add extra combinational logic to route signals. The impact of this difference is quantified in the later chapters of this dissertation.

2.2 Nanoelectronic System Reliability Enhancement

In many nanoelectronic systems, there is likely to be a non-trivial number of manufacturing defects capable of causing faults. As a result, many proposed nanoelectronic architectures incorporate some form of defect and/or fault tolerance to mitigate these problems. Two of the common techniques implemented in nanoelectronic architectures are configurable (aka programmable or reconfigurable) systems and the use of hardware replication and redundancy. These techniques are briefly described below.

2.2.1 Configurable Architectures

Reconfigurable or programmable architectures were originally developed for quickly and cheaply implementing different circuits with performance similar to
ASICs but without the design time or cost of ASICs. Common examples of configurable logic include programmable logic arrays (PLAs) which can program a sum-of-products function and field-programmable gate arrays (FPGAs) which are highly configurable devices capable of implementing everything from simple circuits to complex microprocessors. A general overview of configurable computing can be found at [98].

Memory systems are another example of configurable architectures. In the design and manufacturing process for a memory array, extra rows and/or columns are built. Once the manufacturing process is complete, the memory system is tested to determine if any of the original lines (row or column) is faulty. If so, the faulty line is swapped for one of the extra lines (assuming the spare line is non-faulty) and the memory system remains functional. This mechanism of building extra lines is typically referred to as sparing and exchanging the original (faulty) line for a spare line is typically referred to as sparing.

An early effort to build a functional computer from faulty components was the Teramac project [42]. Teramac had approximately 7.6 million components, with 3% of them being defective. Nearly 900 FPGAs were used in this system for the processing elements and memory, with 75% of them defective. In spite of these defects, Heath et al. were able to create a functional computer system by configuring the Teramac to avoid using the defective locations in its processing logic and memory. A key result from this project is that these systems needed to provide a large amount of communications bandwidth between the processing elements and memory to ensure that these elements and memory did not become isolated and unusable.

In the context of developing defect and fault tolerant nanoelectronic architec-
tures, configurable techniques are favored when the underlying devices support a crossbar array with (individually) addressable crosspoints or when the device architecture is a transistor. Examples of reconfigurable nanoelectronic crossbar arrays have been presented in [20, 39, 89].

2.2.2 Hardware Redundancy

Within the set of hardware redundancy techniques are two methods described by von Neumann in [99]: multiplexing and N-modular redundancy. N-modular redundancy is described here in significant detail since it is used later in this work.

2.2.2.1 von Neumann Multiplexing

In [99] von Neumann proposed a multiplexing scheme that could be utilized with majority gates or NAND gates. With majority multiplexing, shown in Fig. 2.7(a), each input and gate is replicated multiple times in the execute component of this scheme. Since the probability of error in the outputs can grow using this method, a restoration stage is required to remove this effect. Within the restoration stage a random permutation of the wires occurs in the block labeled “U.” von Neumann also applied multiplexing to a system of NAND gates. NAND multiplexing, shown in Fig. 2.7(b), requires a second restoration stage to prevent an inversion of the desired function.

Several variations of von Neumann multiplexing have been developed for nanoelectronic architectures in [40, 82, 83]. The use of multiplexing for implementing fault tolerance in QCA is not discussed here since the results presented later in this work demonstrate that large amounts of signal routing can have a negative effect on system reliability. Since the wire permuters (blocks labeled “U”) are simply
large blocks of signal routing, the results presented later in this work suggest that multiplexing could cause significant reductions in system reliability.

2.2.2.2 N-modular Redundancy

N-modular redundancy (NMR) was proposed by von Neumann in [99]. In NMR, N instances of the same module perform the same computation and then a majority vote of the output(s) is taken. As long as ⌈N/2⌉ modules compute the output bit(s) properly, the system output is correct. This voting process enables a logical masking of potential errors in the modules. In this work, two versions of NMR, N = 3 and N = 5, also termed triple modular redundancy (TMR) and quintuple modular redundancy (QMR) respectively are considered.

The reliability equations in this section are considered the “classical” models
since they assume the signal routing to be perfect. Unless otherwise noted, the reliabilities presented in this dissertation for QCA circuits and systems will assume the signal routing to be faulty.

**Triple Modular Redundancy:**

[62] provides the background for the TMR models and reliability equations described here. The basic TMR model, also known as a trio, is shown in Fig. 2.8(a). If the original system reliability is low, then the system can be divided into slices that are cascaded (chained) together as shown in Fig. 2.8(b). In both cases, either a single voter or replicated voters can be utilized.

The system reliability for a single trio \( R_{T, NR} \) with non-replicated voters, given a module reliability of \( R_0 \) and a voting circuit reliability of \( R_v \) is

\[
R_{T, NR} = R_v * (3R_0^2 - 2R_0^3). \tag{2.1}
\]

If using replicated, rather than non-replicated voters, the system reliability for a single trio \( R_{T, Rep} \) is

\[
R_{T, Rep} = R_v 3R_0^2 - 2R_vR_0^3. \tag{2.2}
\]
If the reliability of the original module is low, then the reliability of the system can be improved by using cascaded TMR. If it is assumed that the original module can be divided into \( N \) slices with each partial module having the same reliability (thus, \( R_M = R_0^{1/N} \) where \( R_M \) is the reliability of each partial module) then Eq. 2.3 and Eq. 2.4 will compute the system reliability for systems implementing cascaded TMR using either non-replicated \( (R_{NR}) \) or replicated \( (R_{Rep}) \) voters respectively.

\[
R_{NR} = (R_v * (3R_M^2 - 2R_M^3))^N = (R_v * (3R_0^{2/N} - 2R_0^{3/N}))^N \quad (2.3)
\]

\[
R_{Rep} = (3R_v^2R_M^2 - 2R_v^3R_M^3)^N = (3R_v^2R_0^{2/N} - 2R_v^3R_0^{3/N})^N \quad (2.4)
\]

In this work, TMR-NR will refer to a cascaded TMR system with non-replicated voters and TMR-Rep will refer to a cascaded TMR system with replicated voters. Also note that if \( N = 1 \), the systems consist of just a single trio.

Figure 2.9 shows the difference in system reliability when comparing cascades of TMR systems when using non-replicated and replicated voters for a range of \( N \). For each of these figures, \( R_0 \) is on the x-axis, \( R_v \) is on the y-axis, and the shade shows the difference in system reliabilities rounded to the nearest tenth. Note that the scale used in the shading changes with each plot and that the contour shades represent steps of 0.1.

In Fig. 2.9, the black regions are where the reliability of the two implementations are approximately equal. The lighter the region, the higher the reliability TMR-Rep is over TMR-NR. The system reliabilities for both of these techniques is dependent on \( R_v \). This impact is seen when \( R_v = 1 \) because the reliability of each TMR implementation is equivalent; thus, there is a black region that spans the top of each graph. This trait also explains why the contour lines are more
Figure 2.9. These graphs show $(R_{Rep} - R_{NR})$ for cascades of $N$ wire slices where both $R_{Rep}, R_{NR} \in [0, 1]$. The region in black is where the difference is 0. **Note:** The scale is in steps of 0.1 and changes in each subfigure.

densely packed at the top ($R_v \approx 1$) rather than the bottom of each figure. In the $N = 1$ case, we see that TMR-Rep can only offer small reliability gains over TMR-NR. However as $N$ goes from 10 to 50, the reliability gains for TMR-Rep over TMR-NR can be far more significant. The downside to this improvement is that it occurs in a much smaller region as $N$ increases.

**Quintuple Modular Redundancy:**
Figure 2.10. Logic to determine a single bit output in QMR.

Only a single level of modules for QMR is developed in this work. Instead of the three copies of a module and a majority gate(s) acting as the voting logic, QMR consists of five copies of the module and the voting logic. A half adder and a full adder are used to implement the QMR voting logic. The final output value is selected using the logic shown in Fig. 2.10 and described by Eq. 2.5 where \( C \) represents the carry bit, \( S \) represents the sum bit, \( HA \) is for the half adder (2 input bits), and \( FA \) is for the full adder (3 input bits).

\[
Output_{QMR} = \text{maj}(C_{HA}, (S_{HA} + S_{FA}), C_{FA}) \tag{2.5}
\]

For a single output bit, \( Output_{QMR} \) is a logical one if the five modules generate at least three logical ones. If a carry bit from an adder is logical one, that means at least two votes for logical one are observed. If a sum bit from an adder is logical one, that means a single vote for logical one is observed. In order to reach three votes for logical one at least one carry bit and one sum bit must be logical one. If both sum bits are logical one, and neither carry bit is logical one, that represents
only two votes for logical one, thus $Output_{QM} = 0$.

**Circuit Complexity Differences** One of the major differences between each of the potential NMR methods is in the complexity of the signal routing needed to move signals from the inputs to the modules and then from the modules to the outputs. Table 2.1 shows the number of crossings required both into and away from a single layer of modules for the NMR techniques implemented in this work.

The input routing when using TMR is independent of the voter configuration; for the TMR output routing, non-replicated voters are marked as “TMR, NR” and replicated voters are labeled “TMR, Rep.” For the QMR output routing results, the crossovers used in the adder circuits have been included in the table. QMR with modules of three output bits and four input or output bits since they are not tested due to limitations in Matlab (discussed in Sec. 3.1).

**TABLE 2.1**

**NUMBER OF WIRE CROSSINGS REQUIRED FOR VARIOUS NMR METHODS.**

<table>
<thead>
<tr>
<th>Num. of bits</th>
<th>Input Routing</th>
<th>Output Routing</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TMR</td>
<td>QMR</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>13</td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>-</td>
</tr>
</tbody>
</table>
2.3 QCA Centric Reliability

In this section the potential defects in QCA circuits are presented. Previous work on enhancing reliability at the component and system level is then discussed.

2.3.1 QCA Defects and Faults

A variety of potential defects for ESQCA circuits, particularly molecular based circuits, are shown in Fig. 2.11. These defects are likely to result from the inherent imprecision in manufacturing these circuits. Modeling these defects when completing physical simulation is straightforward and not discussed here.

There is a rich body of work in the literature that has investigated the impact of various defects and temperature concerns in ESQCA components and circuits. A sampling of this work includes [11, 23, 34, 51, 61, 66, 85, 88, 90, 95, 103]. These papers have tended to find the same faults if the level of defects (or temperature) is high enough. The common faults are either 1) a weak or non-deterministic polarization and 2) an inversion of the desired output.

In MQCA circuits, the potential defects are likely to be displaced and misshapen magnets. A misshapen magnet could have either missing or extra magnetic material. These defects are likely to exist in almost every magnet since perfectly rectangular magnets placed exactly where desired will be nearly impossible to achieve.

As reported in [69], these defects are likely to cause stuck-at faults if the clock signal is not strong enough to null (erase the previous polarization) the magnet. If the defective magnet cannot be nulled, it will tend to stay in its previous state, thus causing a stuck-at-last fault. In some cases, defect tolerance can be provided by supplying a stronger clock signal, but at the cost of power dissipation. The
implications of this fault at the circuit level have yet to be explored.

2.3.2 QCA Component Reliability Enhancement

To date, much of the work on increasing the robustness of QCA components has investigated ESQCA majority gates and wires. There does not appear to be any literature presented to date that investigates improving the robustness of MQCA components.

Fijany and Toomarian [34] centered their efforts on detailing how the basic QCA majority gate can be modified into a “block majority gate” where multiple QCA cells are arranged in a rectangle and operate as a single gate. The block majority gate is found to be more resistant to defects like cell misalignment, cell rotation, and missing cells. This work also demonstrated that thick input wires may also be beneficial to the operation of a block majority gate. However, this work did not 1) identify an ideal block majority gate size, 2) provide any guidance.
as to error rates for the majority gate, nor 3) determine the ideal locations for the inputs of the majority gate.

![Regular and thick ESQCA wire segments.](image)

Figure 2.12. Regular and thick ESQCA wire segments.

References [23, 31, 51, 61, 96] examine how straight wires can be made more robust. The common trait among these works, similar to the results of [34], is that increasing the “thickness” of the wire by having multiple cells in parallel, increases its resilience to defects. Figure 2.12 shows both a single wide wire and a thick wire that is three cells wide. Reference [96] observes the impact of stray charge on a metal-dot cell wire. Reference [51] examines temperature and misalignment defects for metal-dot cells. Reference [61] considers rotations and displacements for clocked molecular cells. References [31] and [23] consider missing cells in straight and bent wires for clocked molecular cells. For [23, 31], error rates for the components have been estimated, but these results are limited since defect rates
for molecular QCA are not known and the tested wires were of a fixed length.

2.3.3 QCA System Reliability Enhancement

Two approaches, other than a direct application of NMR, have been proposed for increasing the reliability of QCA systems. The first, a programmable logic array (PLA) for QCA has been presented in [17, 18]. A method for estimating the yield of QCA based PLAs can be found in [17]. This work utilizes a new parameter, the effective defect rate (EDR), to capture the probability that a specific defect type (i.e., displacement) causes a QCA cell to malfunction since not all defects will cause a QCA cell failure. The EDR is then used to compute the probability that a defect of type $x$ will cause a fault in the PLA cell. The probability of a fault in the PLA cell is then computed using the probability of a fault for all defect types. The yield of the PLA is then computed as a function of the PLA cell probabilities. The results of [17] demonstrate that PLA yield can drop significantly once a certain EDR is reached. The EDR causing this drop varies based on the circuit size; larger circuits require lower EDRs than smaller circuits to have the same yield.

A second technique blends the use of TMR hardware redundancy with time redundancy of computing with shifted operands [78] for QCA based circuits was presented in [104]. In this technique, termed TMR with shifted operands (TMRSO), an add function is computed three times, but rather than doing the three additions in parallel, they are done in different time steps. As a result, the cost of replicated hardware is reduced, but the latency of the operation increases. The difficulty with this technique will probably not lie in the number of crossovers needed to implement it (estimated at 26 from the figure in [104]), but rather in the circuit floorplan since feedback loops are required. The implications of using
hardware redundancy in regards to the circuit design will be discussed later.
CHAPTER 3

MODELING TECHNIQUES

In this chapter, the modeling techniques used later in this dissertation are discussed. Other QCA modeling techniques that are being utilized in the literature are also summarized. The methods in Sections 3.1 and 3.2 are analytic modeling techniques that can be used for both ESQCA and MQCA circuits. The remainder of the techniques presented in this chapter are based on physical modeling for ESQCA, with the exception of OOMMF (Sec. 3.5.2) which is used to physically model MQCA. The final section of this chapter, Sec. 3.6 describes the results of an experiment run to correlate analytic modeling with physical modeling for ESQCA.

The publications associated with this dissertation are the first QCA-based applications of the probabilistic transfer matrix method (Sec. 3.1) and linear regression (Sec. 3.2). A large number of Matlab scripts have been created to utilize these methods. The geometric analysis method (Sec. 3.3) is a new modeling technique developed for this dissertation; a separate design tool has also been created to implement this method. A new simulator has been built to implement the last major modeling approach used in this work, the three-state coherence vector (Sec. 3.4).
3.1 Probabilistic Transfer Matrices

Probabilistic transfer matrices (PTMs) were recently described in [52, 53] as a method for accurately computing the reliability of circuits (or system) given error prone components. Each component has its own PTM, and by completing a sequence of matrix based operations, a PTM for a circuit is formed. This circuit PTM is then used to calculate the circuit’s reliability.

The PTM for a component, circuit, or system with \( m \) inputs and \( n \) outputs is a \( 2^m \times 2^n \) matrix that shows the relationship for all combinations of inputs and outputs. At location \((i, j)\) in the matrix, the specific relationship between input set \( i \) and output set \( j \) is defined. If the component produces an incorrect output with probability \( p \) (fault rate), regardless of input values, correct relationships between \( i \) and \( j \) are represented by \( 1 - p \) and erroneous relationships are represented by \( p/(2^n-1) \) in the PTM. If the PTM is fault-free \((p = 0)\), the matrix is called an Ideal Transfer Matrix (ITM) and the only non-zero entries are correct input/output combinations. A PTM for a majority gate with an error rate of \( p \) is shown in Fig. 3.1(a) while the ITM for the majority gate is shown in Fig. 3.1(b).

In a simplistic error model, if the component produces an incorrect output with probability \( p \) (error rate), regardless of input values, correct input/output combinations are represented by \( 1 - p \) and erroneous input/output combinations are represented by \( p/(2^n-1) \) in the PTM. To prevent a given input combination from having an output probability greater than one, the sum for the entries of a row in a PTM must be 1.

A circuit PTM is generated by first dividing the circuit into slices that are one component level each, i.e. no level has a series of components. Within a
level the components are in parallel, and the Kronecker product\(^1\) is calculated for the level, thus giving each level a PTM. These level PTMs are then multiplied, from circuit inputs to outputs, using matrix-matrix multiplication to generate the circuit PTM. At each stage of this process, the size of the PTM corresponds to the number of inputs and outputs as discussed above.

Once the PTM for the circuit has been generated, it is element-wise multiplied by its ITM so that only the non-faulty input/output combinations exist in the ETM (element-wise transfer matrix). An example ETM is shown in Fig. 3.1(c). Once the ETM is found, \(v^* ETM = v'\) is computed, where \(v\) is a vector consisting of the probability of each input occurring. The resulting vector, \(v'\) is the probability of each output occurring correctly. The exact reliability for the circuit is found by summing the elements of \(v'\).

\(^1\)The Kronecker product, denoted by \(\otimes\), is block matrix multiplication where with a \(m \times n\) matrix \(A\) and a \(p \times q\) matrix \(B\) results in a \(mp \times nq\) matrix. For example, consider \(A\) and \(B\) to each be a \(2 \times 2\) matrix. In this case a \(4 \times 4\) matrix of the following form results: \(A \otimes B = \begin{bmatrix} a_{1,1}B & a_{1,2}B \\ a_{2,1}B & a_{2,2}B \end{bmatrix}\)

Figure 3.1. Majority gate transfer matrices.
In many cases, a soft lower bound for circuit reliability is observed when component error rates are 50%. This lower bound is equivalent to $1/2^n$ and has been termed the *random circuit reliability (RCR)*. References [24, 25, 27, 52, 53] contain more detailed examples and discussion on this process.

The analytic modeling results presented in this work are generated via Matlab rather than the decision diagram method described in [52, 53]. References [52, 53] state that circuit width (number of bits in a slice) is the main limiting factor when using this approach due to the exponential space requirements of the matrices. These papers suggest a maximum width of approximately 40 using decision diagrams. Using Matlab limits the maximum circuit width to approximately 17 or 18 (best case) since a PTM of this size could contain over a billion entries which is too large for Matlab to handle.\(^2\) Having multiple slices with large PTMs can also significantly slow the time to result using this method since the matrices are continually being swapped in and out of memory. To improve performance when using Matlab with the PTM method, the use of sparse matrices should be considered (particularly when doing the Kronecker products within a slice).

3.1.1 PTMs and QCA

When calculating QCA circuit reliabilities with PTMs, both logic and interconnect must be treated as fault-prone since each is built from combinations of the same basic device. As a result, the following component PTMs are used throughout this work: AND, OR, and MAJ gates; inverters; wires; single bend wires; T-shaped fanouts; regular fanouts; and wire crossing devices (MQCA only). A simplifying assumption used here is that the fixed input value required to convert a majority gate into an AND or OR gate is error-free. This ensures that the error

\(^2\)For example, a PTM with 15 inputs and 15 outputs would have $2^{30}$ entries.
rates for all three logic gates are equivalent throughout this work.

In Fig. 3.2, an XOR gate built from QCA components and its corresponding circuit PTM are shown. This circuit consists of the following components: three AND gates, an OR gate, an inverter, seven straight wires, two single bend wires, and a T-fanout. A separate wire component has not been used between the output of the majority gate and the input of the inverter. Circuit outputs are considered fault-free since some of these outputs are then used as inputs into other circuits, and these inputs are considered fault-prone.

The division of the XOR gate into gate levels, with each level being denoted by LX, with X being an integer, and the construction of the XOR gate’s PTM, XOR\_PTM, is shown in Fig. 3.2(b). The symbol \( \otimes \) is used to denote the Kronecker product operation, I is the 2x2 identity matrix and is necessary to properly size the gate level matrices for matrix-matrix multiplication. I is required in gate levels 1-5 as a place holder for the rightmost A and B inputs.

Throughout this work, it has been assumed that the clocking structure does
not introduce any errors into the circuit logic. Additionally, the circuits developed for use in this work are laid out in an approximate fashion. The underlying assumption is that each instance of a straight or bent wire could be segmented as necessary for the circuit to be clocked in an orderly fashion.

Another simplifying assumption that has been made is that wires of different lengths are treated equivalently. While wire length will be a factor in its reliability (as shown later), this prevents needing to do a cell by cell layout of the circuit. Since most of the long wires that appear in the circuits used here have fanouts along their lengths, they are generally treated as multiple faulty segments (wires, fanouts, single bend wires). Dividing the wire into smaller multiple segments should have a similar effect as giving longer wires higher error rates.

3.2 Linear Regression Analysis

Linear regression is a common statistical approach for constructing an equation that can be used to predict an output (dependent variable) given a set of input values (independent variables) and for identifying how each independent variable influences the dependent variable. In this dissertation, linear regression is used to identify which QCA component(s), the independent variables, are most critical to the reliability of a circuit (or system), the dependent variable. By identifying which component(s) are most critical to reliability, efforts to increase reliability can focus on the components that will provide the greatest increase in reliability.

The regression method, which will be detailed below, utilizes results from the PTM modeling of various circuits and systems. Within much of the PTM modeling that is done here, the error rates for the components are equal (uniform). In computing the circuit reliabilities for the regression analysis, the error rates for
the components will vary by several orders of magnitude.

The linear regression analysis forms a linear function of the form \( a_1 x_1 + a_2 x_2 + \ldots + a_n x_n + c = r \). Within this linear function, \( n \) is the number of component types (i.e., a majority gate is a component type), the \( x_i \) values are the input variables, in this case the component error rates, and \( r \) is the circuit reliability that results from a set of specific set of \( x_i \) values. The regression coefficients, the \( a_i \) values, are directly proportional to the impact the variable \( x_i \) has on the circuit reliability. The constant \( c \) is an intercept term that ensures that the circuit reliability approaches one as the error rates decrease (i.e., if the component error rates were zero, then \( c = r = 1 \)).

The regression coefficients are computed using Matlab. The inputs to the regression function in Matlab (\texttt{regress}) are sets of component error rates (\( x_i \) values) and the circuit reliabilities (\( r \) values) that result from each set of component error rates. One of the outputs of the \texttt{regress} function is the “best fit” regression coefficients (\( a_i \) values). The larger the magnitude of the regression coefficient, the greater the impact of that component on the circuit reliability. For the work presented here, the most (least) critical component is the component that has the largest (smallest) regression coefficient. These regression coefficients, presented later in this work, are negative since the reliability of the circuit is reduced as the component error rate increases. It should also be noted that a set of regression coefficients is unique to a given circuit.

This dissertation will also discuss a variation of the regression analysis where the regression coefficients are divided by the number of instances of a given component in the circuit. For example, if the regression coefficient for the majority gate error rate is \(-5.3\) and there are 20 majority gates in the circuit, then the regres-
sion coefficient is normalized to \(-5.3/20 = -0.265\). These normalized regression coefficients are then ordered to determine the most critical component.

In order to compare whether the original regression or normalized ordering of components should be utilized, the circuit reliabilities are reanalyzed using the PTM method. In this comparison, the most critical component found using each method is considered perfect (one component at a time), and the resulting circuit reliabilities are computed. If the circuit reliabilities for when the most critical component found by the regression analysis is perfect are higher than when the most critical component found by the normalized ordering is perfect, then the regression ordering is favored. Else, the normalized ordering is favored. The results presented in Sec. 4.7 will show that the regression ordering is favored over the normalized ordering.

3.3 Geometric Analysis

One of the major challenges in evaluating the reliability of components or circuits is that the computational time required to evaluate a single component or circuit can be extremely high. One method of reducing computational time to evaluate the operation of a QCA is based on the principles of VLSI yield estimation and was presented in [33]. For the results presented in Sec. 6.1, wires of one to three cells wide are considered.

In VLSI yield estimation, the physics of the system was typically entirely abstracted away and only the system geometry was considered. One tool that did this, VLASIC [100], could identify the fault(s) caused by defects in the layout geometry of an integrated circuit. This tool was particularly focused on local, rather than global defects. When a circuit was to be tested, a distribution of
defects (based on fabrication data) was introduced into the circuit and then the
tool examined the resulting geometry of each layer to determine the existence of
a fault. Here, similar principles are applied by examining patterns of missing cells
that were found to cause wire failures when full quantum mechanical simulation
was utilized. By building a database of these patterns, larger circuits are tested
without having to do full physical simulation. This enables the quick testing of a
large number of circuits.

3.3.1 Missing Cell Patterns

References [23, 31] examined the impact of multiple missing cells on straight
and bent wire segments of varying widths made from densely packed molecular
QCA cells with the two-state coherence vector method from QCADesigner [2] per-
forming full physical simulation. In these studies, a number of cells were selected
at random from the wire segment under test and removed from the circuit. If the
wire passed both a binary 0 and a binary 1 properly the wire was considered cor-
rect (passed), else the wire segment was considered to have failed. Upon analyzing
the failure cases certain patterns of missing cells appeared on multiple occasions.

Some of the common failure patterns are shown in Figure 3.3 (a)-(d). In many
of these patterns, there is a common factor: a diagonal arrangement of cells is
present. This diagonal arrangement is used for an inverter, thus it is unsurprising
that arrangements of this type cause inversion faults. In particular, in Fig. 3.3(b)
an inverter has effectively been built into the wire. One cell wide straight wires
are considered to fail when a gap of three or more cells exists. This is due to
the fact that the diagonal arrangement of cells cannot occur with this defect in
straight wire segments prevents inversion faults, but non-deterministic faults are
observed (output is random).

Many of the failure cases have been confirmed in MAquinas [12], but for many of the bent wire cases, MAquinas failed to generate reproducible results. As a result, the two and three cell wide corner regions are treated as follows: using Fig. 3.3(e) as a model of the two cell wide corner, we first determine if either cell marked with a star is missing. If so, a failure has occurred (inversion fault). If not, the method will look for straight wire patterns in the horizontal wire up to the line labeled X, and in the vertical wire starting at the line labeled with Y. This treatment of the corner regions is reasonable since previous work [23] has shown that the inner part of the corner (the starred cells) is far more sensitive to defects than the outer corner region (upper right portion of the corner). A more complete list of failure patterns and further discussion on how they were identified is available in [32].
3.3.2 Tool Operation

The geometric analysis tool uses circuit design files created by the QCADesigner version described in [23] as input. The circuit design is then placed on a grid structure with unique addresses for each cell in the design and null addresses for grid locations without cells. Once the design has been placed, the corner cells (marked by stars in Fig. 3.3(e)) are identified. For each test, a group of cells is then dropped at random from the design. These dropped locations then contain a common address that identifies them as missing cells.

To determine if a failure exists in the memory structure under test, the tool first checks the corner regions of the circuit. If no failures occur in the corner regions, the tool then identifies the input cell and then walks down the wire in the direction of the data flow one cell at a time. At each cell, the tool investigates the local neighborhood for that cell to identify if a failure pattern exists. If a failure pattern is identified, the test ends and the wire is marked as failed. If the walk reaches the end of the wire and a failure pattern has not been identified, the wire is functional (passes) and a new test begins.

3.3.3 Computational Speedup

As mentioned above, the major advantage of developing and using a tool of this type is that of computational speedup. For the wires that are presented in Sec. 6.1.1, if the physical simulation method utilized in [31] is used to determine the operation of a wire segment, generating a single result could take anywhere from a couple of hours to a week or more depending on the length and width of the wire. Using this method, only a few microseconds are required to generate a single result.
3.3.4 Extendibility

Another advantage of the geometric analysis method is that extending it to include other defect and fault models should be fairly straightforward. Additionally, extending this model to the other logic and interconnect components (i.e. majority gates and fanouts) should be straightforward as well. Examples of results in the literature that could be used to develop these extensions include [61, 66, 67, 85, 91].

3.4 3-State Coherence Vector

The 3-state coherence vector (3SCV) model is an approximation of a full quantum mechanical model for determining the dynamics of a QCA system. This approximation is necessary to make solving the dynamics of a system tractable. This approach was developed and utilized in [61, 93, 94] and is intended for clocked six-dot ESQCA cells coupled to a thermal bath. This approach considers the electrostatics of the system (including the clock) and the thermodynamics of the system.

This model assumes a three-state approximation of the cell, i.e. the cell is polarized to either +1 or -1 or is null (polarization of 0). Additionally, it is assumed that the cell cannot switch from +1 to -1 (or vice versa) without being in the null state. A density matrix formalism is used in this approach; this formalism provides a method for describing the statistical state of a quantum system. This representation can then be utilized to extract values of interest, such as the polarization of the cell.

The coherence vector $\vec{\lambda}$ is an eight-element vector that is the projection of the density matrix ($\hat{\rho}$) onto the generators of SU(3). The components of $\vec{\lambda}$ are
\[ \lambda_i = Tr\{\hat{\rho}\hat{\lambda}_i\} \quad (3.1) \]

with \( \hat{\lambda}_i \) being a specific generator of SU(3). The coherence vector approach is utilized because it provides a simple mechanism for including dissipative coupling between the system and a heat bath. For reference, the polarization of a cell is the inverse of the seventh element of the coherence vector \( (P = -\lambda_7) \).

For each cell \( r \) in the system (with neighboring cells \( s \)) the following Hamiltonian is constructed

\[
\hat{H}_r(t) = \begin{bmatrix}
-0.5 \cdot PEk & 0 & -\gamma \\
0 & 0.5 \cdot PEk & -\gamma \\
-\gamma & -\gamma & E_c
\end{bmatrix} \quad (3.2)
\]

where \( E_c \) is the energy seen on cell \( r \) due to the clock, \( \gamma \) is the tunneling energy between a polarized state and the null state, and PEk is found as follows

\[
PEk = \sum_{s=0}^{r.neighbors} s.Polarization \cdot \begin{cases} 
    r.NegEk[s] & \text{if } s.Polarization < 0 \\
    r.PosEk[s] & \text{if } s.Polarization > 0 
\end{cases} \quad (3.3)
\]

\( E_k \) is the kink energy between two cells and is roughly the energy cost of them having different polarizations or the same polarization. This energy is found as follows: \( E_k = E_{\text{diff}} - E_{\text{same}} \) where \( E_x \) is computed using Eq. 3.4. For this equation, \( q_{i(j)} \) is the electronic charge located on dot \( i(j) \), \( \epsilon_0 \) is the permittivity of the molecule, and \( d \) is the distance between dot \( i \) and dot \( j \). When \( x = \text{diff} \) cells \( r \) and \( s \) have different polarizations and when \( x = \text{same} \) cells \( r \) and \( s \) have the same polarization. To ensure that the cell is charge neutral, each dot contains a charge.
of $+e/3$.

$$E_x = \sum_{i=0}^{r\text{ dots}} \sum_{j=0}^{s\text{ dots}} \frac{q_i \cdot q_j}{4\pi \epsilon_0 d}$$

(3.4)

In calculating the values of $r.PosEk[s]$, the polarization of cell $r$ is -1,+1 and the polarization of cell $s$ is +1 for $diff$ and $same$. For $r.NegEk[s]$, the neighbor cells, $s$, are assumed to have a polarization of -1 (thus $r$ is +1,-1 in $diff$ and $same$).

With the Hamiltonian for each cell, the other components of the dissipative equation of motion can be found. The equation of motion is a set of coupled differential equations (each cell solves the equation separately and the system is moved forward in time as a single unit) as shown here:

$$\frac{\partial}{\partial t} \vec{\lambda} = \vec{\Omega} \vec{\lambda} - \frac{1}{\tau} [\vec{\lambda} - \vec{\lambda}_{ss}(t)].$$

(3.5)

In this equation, $\vec{\Omega}$ is a specifically defined matrix (see [93]), $\vec{\lambda}$ is the coherence vector for a cell, $\vec{\lambda}_{ss}$ is the instantaneous thermal equilibrium coherence vector for a cell, and $\tau$ is the energy relaxation time representing the strength of the coupling between the system and the thermal bath.

Equation 3.6 is used to find each of the eight elements ($j$) in $\vec{\lambda}_{ss}$ for each cell. In this function, the parameter $\hat{\lambda}_j$ is the $j$-th generator of SU(3).

$$\vec{\lambda}_{ss}^{(j)} = Tr\{ \hat{\rho}_{ss}(t) \hat{\lambda}_j \}$$

(3.6)

$$\hat{\rho}_{ss}(t) = \frac{tmp}{Tr[tmp]}$$

(3.7)

$$tmp = e^{-\hat{H}(t)/k_B T}$$

(3.8)

To solve this equation the initial value of $\vec{\lambda}$ is set to $\vec{\lambda}_{ss}$ at $t = 0$ for each
cell and then the system is marched forward, as a single unit, in time using the Dormand-Prince method. The Dormand-Prince method is an established differential equation solver from the Runge-Kutta family that utilizes a variable, rather than fixed, time-step. References [15, 105] are useful starting points for understanding the Dormand-Prince method.

3.5 Other Modeling Techniques

In this section, a brief overview of other QCA modeling techniques is presented. These are divided into ESQCA and MQCA sections.

3.5.1 ESQCA Modeling Techniques

3.5.1.1 2-State Coherence Vector

This technique is similar to the one presented above in Sec. 3.4, but instead of using six-dot ESQCA cells, this model uses four-dot ESQCA cells. The basic form for the equation of motion remains the same for this technique, but the mathematical components and solution do vary. This modeling technique is presented in [92] and is the most detailed method utilized in QCADesigner, a frequently used ESQCA CAD tool [2, 101].

3.5.1.2 Statistical Mechanical Modeling

Statistical mechanical modeling is used to find the lowest energy state and the probability of each output configuration occurring. This method does not account for any form of clocking. A four-dot ESQCA cell is assumed. The model discussed below is the six-state method which allows for the two free electrons in the cell to locate on any pair of dots. A simplification of this model utilizing only two cell
states (free electrons diagonally aligned) is also available, but is not shown. This modeling technique is presented in [68, 103].

The key aspect of this model is that it will compute the energy for all possible combinations of electron arrangements. Since each of the $N$ cells can have six possible electron arrangements, a total of $6^N$ energy values are calculated. The energy for a specific arrangement of cells, $E_i$ is:

$$E_i = \sum_{m=1}^{N-1} \sum_{n=m+1}^N E_{m,n}, \quad (3.9)$$

where $E_{m,n}$ is the energy between two electrons computed using Coulomb’s law which is

$$E_{m,n} = \frac{q^2}{4\pi\epsilon r_{m,n}}, \quad (3.10)$$

where $q$ is the elementary charge, $\epsilon$ is the permittivity of free space, and $r_{m,n}$ is the distance between the electrons.

The electron arrangement $i$ which corresponds to the lowest energy is called the ground state and its energy is referred to as $E_{gnd}$. For each combination $i$, the energy difference between it and the ground state is then computed (note that $\Delta E = 0$ for the ground state):

$$\Delta E_i = E_i - E_{gnd}, \quad (3.11)$$

With these energy differences computed, the Boltzmann sum is calculated as:

$$F_{tot} = \sum_{i=1}^{6^N} e^{-\frac{\Delta E_i}{k_B T}}, \quad (3.12)$$

where $k_B T$ is Boltzmann’s constant multiplied by the temperature. A partial
Boltzmann sum is then computed for each of the six cell states. For these partial sums, it is the state of the output cell which determines which partial sum it is added to. In equation form:

\[ F_j = \sum_{i=1}^{6} e^{-\frac{\Delta E_i}{k_B T}}, \]  

(3.13)

where \( j \) is a whole number between one and six and is the state of the output cell. The probability of output state \( j \) occurring is equivalent to \( F_j / F_{\text{tot}} \). The expected polarization \( (P_{ex}) \) of the output cell is then found using the following equation:

\[ P_{ex} = \sum_{j=1}^{6} w_j F_j, \]  

(3.14)

where \( w_j \) is determined by examining the dots on the right side of the cell. If both dots neither dot contains an electron \( w_j \) is 0, if only the upper dot contains an electron \( w_j \) is 1, if only the lower dot contains an electron \( w_j \) is -1. These weights correspond to a cell with no effective polarization, a polarization of 1 and a polarization of -1 respectively.

3.5.1.3 Bayesian Macromodeling

Bayesian networks have recently been proposed as a method for computing the probability of obtaining the correct output of an ESQCA circuit [11, 88]. In this technique, small neighborhoods of QCA cells are modeled using the two-state coherence vector approach. These small neighborhoods of cells are called a macromodel and are roughly equivalent to a component. When using the macromodels in the circuit level simulation, only the average behavior of the macromodel is con-
sidered, not each individual cell. The macromodels are the nodes of a Bayesian (probabilistic) network, and the interaction between macromodels is determined by the Bayesian modeling.

The main advantage of this modeling technique is that it is far more computationally efficient than the two-state coherence vector method in QCADesigner. However, the disadvantage of this modeling method is that physical parameters such as temperature, cell size and spacing, and clock strength are required. This implies that this technique is less general than the PTM approach discussed earlier in this chapter. The computational speedup provided by Bayesian modeling could make it extremely useful for quickly exploring the physical design space (i.e. cell size and spacing) and the error rate of perfect and defective components.

3.5.1.4 MAquinas

MAquinas is a physical ESQCA simulator geared towards molecular QCA [12]. In this simulator, a circuit and its clocking structure/signals are marched forward in time to determine the operation of the circuit. At each time step, the time-independent Schröedinger equation (TISE) is solved to find the ground state of each cell. The solution steps are repeated at each time step until a self-consistent (constant) solution is found. Finding the ground state at each time step is a valid option since, due to adiabatic clocking, it is assumed that the system will always be at or near its ground state energy.

A significant difference between MAquinas and several of the other physical modeling techniques described in this chapter is that in solving the TISE, only the electrostatics of the system are considered; the thermodynamic aspects of the system are ignored. In that regards, this solver is an intermediate step between
the statistical mechanical methods and the coherence vector methods. For the statistical mechanical models, the clock is ignored (but the thermodynamics are not entirely ignored) while in the coherence vector methods both the clock and the thermodynamics of the system are accounted for.

3.5.2 MQCA Modeling Techniques

The main physical modeling tool for MQCA based work is OOMMF, the Object Oriented MicroMagnetic Framework [21, 73]. This simulator solves the Landau-Lifshitz equation which describes how the magnetization of a solid material changes in time. OOMMF is widely used within the micro/nano-magnetic research community and several results have shown a strong correlation between experimental work and OOMMF simulations.

3.6 Correlating Analytical and Physical Modeling

In this section, we compare PTM results to those obtained by physical simulation for two straight wire segments connected in series. This comparison also provides a concrete example of PTM construction and computing the resulting circuit reliability. Before a PTM for a wire segment can be constructed, computing the wire error rate is required.

The experimental setup of Sec. 6.2, which utilizes the three-state coherence vector simulation method (Sec. 3.4), was used to determine the reliability of a short wire segment built from densely packed molecular QCA cells. In this experiment, physical simulation was used to determine the reliability of wire segments when cells are randomly dropped from the wire segments. The wire segment used is ten cells long and three cells wide. 20% of the cells were then randomly dropped from
the wire. Simulation was then performed to determine if the wire transmitted both a logical 0 and a logical 1 properly; if so, the wire passed, else it failed. After completing 1000 iterations, a passing rate of 76.8% (failure rate of 23.2%) was observed.

Using this failure rate, the reliability of two, ten cell long, wire segments connected serially (thus forming a twenty cell long wire) is calculated to be 64.36% using the PTM method described in Sec. 3.1. Figure 3.4 shows the wire segment PTM, circuit PTM, and final reliability calculation for this circuit. The ITM for the wire (used to calculate the ETM) is simply the $2 \times 2$ identity matrix. In this example, we have assumed that logical 0 and 1 have equal input probabilities.

A twenty cell long, three cell wide wire was then created for physical simulation in the same manner as the single wire segment. As was done for the 10 cell long wire, 20% of the cells were dropped from this wire segment. After completing 1000 iterations, a passing rate of 63.3% (failure rate of 26.7%) was observed. This is virtually identical to the pass rate of 64.36% that is predicted by the PTM modeling shown in Fig. 3.4.
Using a straightforward combinatorial model to estimate the passing rate of the 20 cell long wire would result in a passing rate of 58.98% (76.8% * 76.8%). Clearly, this model underestimates the reliability of the wire. This underestimation is a result of not being able to consider the case where two wrongs (two inversions) make a right (proper output) which can be captured by the PTM model. The possibility of two wrongs making a right is captured in the (1,1) and (2,2) locations of Fig. 3.4(b) with the $0.232^2$ terms. Overall, these results validate the use of PTM modeling to estimate the reliability of QCA circuits.
In this chapter a set of common circuits is introduced. The reliability of these circuits, using both ESQCA and MQCA designs, is modeled using the PTM approach outlined in Sec. 3.1. The regression modeling technique, outlined in Sec. 3.2 is then used to determine the critical components of these circuits. To conclude the chapter, an example of how improving the reliability of a critical component improves the circuit reliability is presented. A majority of this section is drawn from the work reported in [24, 25, 27].

4.1 Coming Attractions

This section is used to briefly describe several points that are helpful in understanding this and the following chapter.

4.1.1 Circuit Design Principles

The circuit and system designs presented here have been designed using a “stick-like” layout which is geared to approximate how data would need to flow in a clocked QCA circuit. It is also assumed that the circuit or system could be organized to use only straight wire and single bend wire segments. As a result, long and short wires are treated equivalently in this work; however, as discussed
earlier, many of the longer wires are multiple segments of shorter wires (due to
fanouts), thus they tend to have a higher error rate.

The circuits and systems, while designed by hand, attempt to use a minimum
number of wire crossings. In general, the number of crossovers has been reduced by
intelligently locating fanouts to avoid having the same signal cross another signal
multiple times. In many cases, a single circuit or system was designed multiple
times to identify the design that had the fewest crossovers or, if multiple designs
used the same number of crossovers, the design that minimized the maximum
number of crossovers a signal had to traverse was selected.

4.1.2 Fault/Error Rates

One of the major challenges that occurs when investigating reliability in nano-
electronic circuits is that hard data on error rates is not yet available. For QCA,
almost every device is likely to be defective in some way. To account for the pos-
sibility that the components could range from highly faulty to rather robust, the
error rates considered in this work range from $1 \times 10^{-8}$ to $1 \times 10^{-1}$.

For a majority of the results in the analytic modeling chapters, with the excep-
tion of the regression analysis results, every component has the same error rate.
When all of the components share an error rate, it is referred to as a uniform
ero rate. For the non-regression analysis sections, assume that a uniform error
rate is utilized unless explicitly stated otherwise. Additionally, in many cases, all
but a single component has the same error rate; when this occurs, the error rate
for this single component will be listed in the graph legend unless it is error-free
(perfect).
4.1.3 Reading the Results

A large majority of the graphs in this and the following chapter show how the reliability of a circuit or system changes as the component error rate changes. Note that these graphs are regularly semi-log plots. Frequently, multiple graphs are shown in a single figure to simplify the complexity of each graph.

Two general trends also appear in these graphs. First, for a given component error rate, as the size of the circuit or system increases, the reliability of the circuit or system tends to decrease. Second, when the reliability of the same circuit is computed for both ESQCA and MQCA, the MQCA version will have a higher reliability. This is strictly due to the differences in signal routing between them.

4.2 Circuit Designs

This section starts by presenting the ESQCA logical crossover circuit. The other circuits that are tested in this chapter are then presented. In general, the maximum size of the tested circuits is limited to the size of the PTMs that can be computed using Matlab.

4.2.1 ESQCA Logical Wire Crossing Circuit

The crossover circuit, first described in Sec. 2.1.4, is needed only for the electrostatic circuits. Recall that this circuit uses the axiom that \((A \text{ XOR } B) \text{ XOR } A = B\), and that \((A \text{ XOR } B) \text{ XOR } B = A\). This circuit, which can be built entirely from unrotated cells, is shown in Fig. 4.1 with a single XOR gate being identified in the dotted polygon. The XOR gate consists of the following components: three AND gates, an OR gate, an inverter, seven straight wires, two single bend wires, and a T-fanout. Having to use this logical circuit to cross wires significantly
increases the complexity of the electrostatic circuits as will be shown shortly.

Figure 4.1. Gate layout of ESQCA logical wire crossing circuit.

4.2.2 Parity Tree

The parity tree circuits used here calculate the even parity bit which is 0 if the number of ones in the input vector is even and 1 otherwise. The two input version of the parity tree is an XOR gate. The four, six, and eight input versions are shown in Fig. 4.2. Note that there are no mandatory wire crossings in any of these circuits, so the electrostatic and magnetic implementations are equivalent (technology independent).

Throughout the remainder of this chapter, a two input parity tree is an XOR,
and the remainder of the parity tree circuits are identified by the number of inputs. In some graphs the abbreviation \textit{Ptree} is used for the parity tree.

### 4.2.3 Multiplexers

Figure 4.3 shows the designs for the two and four input multiplexers. No figure for the eight input multiplexer is shown, but it is similar in nature to the four input version. An interesting comparison between these multiplexers is that the smallest uses only one wire crossing, the four input multiplexer uses eight wire crossings and the eight input multiplexer uses twenty-seven wire crossings. This explosion in the number of crossings needed has significant implications when considering the total number of components needed to implement each circuit with ESQCA. Throughout the rest of this chapter a 2-Mux refers to a two input multiplexer, a 4-Mux refers to a four input multiplexer, and an 8-Mux refers to an eight input multiplexer.
4.2.4 Adder

Figure 4.4 shows a ripple carry adder of two bit slices based on the design in [102]. Adders of different sizes are not shown since they are similar in nature to the one shown. Throughout the rest of this paper a 1-Adder refers to a single bit slice adder, a 2-Adder refers to a two bit slice adder, a 3-Adder refers to a three
bit slice adder, and a 4-Adder refers to a four bit slice adder.

Figure 4.4. 2-Adder design.

4.2.5 Multiplier

The multiplier circuit that was designed for this work requires both a half-adder and a full-adder. Both designs are presented in Fig. 4.5. Since the half-adder requires no crossovers, it is technology independent. The full adder is shown since, while similar to the adder presented in Fig. 4.4, it has been rearranged for use in the multiplier. This rearrangement slightly simplifies the component requirements by removing a crossover, but it would likely be more challenging to layout and probably more clocking zones and/or area than the one presented above.
Figure 4.5. Adder circuit designs used in the multiplier.

The multiplier, shown in Fig. 4.6, takes a four bit unsigned binary input and a two bit unsigned binary input and generates a six bit output. It is not a generalized slice of a larger multiplier. This circuit requires ten crossovers, shown as the dotted line boxes, in its interconnection network. When including the full adder circuits, a total of fourteen wire crossings are required.

4.3 Circuit Complexity

Figure 4.7 shows the number of components versus the number of input bits for the parity tree, multiplexors, and adders. For this graph, and others that consider circuit complexity, the select bits for the multiplexers are considered input bits. Thus, the 2-Mux has three inputs, 4-Mux has six inputs, and 8-Mux has eleven inputs. The multiplier is not included in Fig. 4.7 since only a single size is considered; in the MQCA design, 132 components are needed, in the ESQCA design, 334 components are needed. The difference in the number of components
needed for the ESQCA designs as compared to the MQCA designs is unsurprising given the number of logical wire crossings needed in the ESQCA designs.

4.4 Uniform Component Error Rates

In order to begin understanding the inherent reliability of QCA circuits, the first observation will consider circuit reliability when each component has the same error rate. Recall that the electrostatic circuits required six components and the magnetic circuits required seven components. Since no experimental data is available on the error rate for QCA components, the error rates tested in this section cover the range from $1.0 \times 10^{-8}$ to $1.0 \times 10^{-1}$.

The graphs of Fig. 4.8 show the estimated circuit reliability when all components have the same error rate. The results are shown only for error rates of $1.0 \times 10^{-4}$ and higher for clarity. Results for lower error rates will be discussed shortly. The reliability axis for the parity tree and multiplexer graphs starts at 0.5 since this value is the RCR (random circuit reliability) for all of these cir-
circuits. For the adder circuits, the RCR values are 0.25, 0.125, 0.0625, and 0.03125 for 1-Adder, 2-Adder, 3-Adder, and 4-Adder respectively. The circuit reliability approaches this value for all of the adder circuits at high error rates.

The graphs in Fig. 4.8 provide working ranges of expected circuit reliability when faulty components are present. As components are initially designed, built, and tested these results can be used to identify the types of circuits that can be prototyped and how reliable these prototypes might be. Another observation is that circuit reliabilities tend to plummet as the error rate increases beyond 0.1% ($1.0 \times 10^{-3}$), thus this value could be considered a reasonable lower bound on component reliability. Additionally, for larger circuits, the circuit reliabilities quickly approach their RCR values. This trend is quite pronounced in the electrostatic circuits as the error rate increases above 1%. Lastly, when considering these graphs as a whole, it is apparent that magnetic circuits are more reliable
than their electrostatic counterparts. This is a direct result of the wire crossing method necessary for each implementation.

Rather than show data for component error rates below $1.0 \times 10^{-4}$, Table 4.1 shows the component error rate required for each circuit to have a reliability of five nines (0.99999) or better. For each fewer (greater) nine of reliability that is acceptable, the uniform error rate necessary to achieve that reliability increases (decreases) by an order of magnitude. Considering the two input parity tree, four nines of reliability is observed at a uniform error rate of $1.0 \times 10^{-5}$ and three nines
of reliability is observed at a uniform error rate of $1.0 \times 10^{-4}$. 
# TABLE 4.1

**UNIFORM ERROR RATE CORRESPONDING TO FIVE NINES RELIABILITY**

<table>
<thead>
<tr>
<th>Parity Tree</th>
<th>Adders</th>
<th>Multiplexers</th>
<th>Multiplier</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>EStatic</td>
<td>Magnetic</td>
<td>EStatic</td>
</tr>
<tr>
<td>2 Inputs</td>
<td>1.0×10⁻⁶</td>
<td>1.0×10⁻⁷</td>
<td>5.0×10⁻⁷</td>
</tr>
<tr>
<td>4 Inputs</td>
<td>2.5×10⁻⁷</td>
<td>2-Adder</td>
<td>5.0×10⁻⁸</td>
</tr>
<tr>
<td>6 Inputs</td>
<td>1.0×10⁻⁷</td>
<td>3-Adder</td>
<td>2.5×10⁻⁸</td>
</tr>
<tr>
<td>8 Inputs</td>
<td>1.0×10⁻⁷</td>
<td>4-Adder</td>
<td>2.5×10⁻⁸</td>
</tr>
</tbody>
</table>
4.5 Circuit Complexity versus Reliability

In this section circuit reliability is measured as a function of circuit complexity (excluding the multiplier circuit). Figure 4.9 plots the circuit reliabilities as a function of the number of circuit components. Graphs could be generated by comparing the reliability to the number of input bits, but they would be nearly identical and would preclude the inclusion of a reference line.

For these graphs, the component error rate is assumed to be uniform and is shown in the legend. Additionally, these graphs display a reference line, labeled $Ref.$ which is computed as $(1 - err)^n$ where $err$ is the uniform error rate and $n$ is the number of components. These reference lines fall below those of the circuits since they do not consider the circuit structure, two wrongs making a right, or any potential fault masking.

In the electrostatic graph, the multiplexers are more reliable than the adder throughout. In the magnetic graph, the parity tree and adder reliabilities tend to nearly overlap (particularly at the lower error rates) and that there is some flux in which circuit is most reliable when few components are needed. However, once approximately 50 components are necessary, the multiplexer has the highest reliability and the parity tree the lowest. The lack of fault masking is likely the main cause for the low reliability of the parity tree.

The linear trends shown in both of these graphs suggests that increasing the circuit size and complexity will not have a compounding negative affect on circuit reliability. This impact may enable fewer large logic blocks to be more effective than a greater number of smaller logic blocks. Another expected trend would be that circuits with similar complexities would have similar reliabilities, which generally is not the case here. This suggests that two factors may be quite influential:
Figure 4.9. Circuit reliability as a function of the number of circuit components. Lines are for uniform error rates.

1) fault masking and 2) the number of outputs of the circuit. Estimating the impacts of these factors is left for future work.

The results in Sec. 4.4 and here provide a good starting point for identifying the component error rates necessary to build reliable circuits and how circuit complexity may impact reliability. However, these results do not consider how
circuits may be impacted by having individual component error rates that may differ by several orders of magnitude. In the next section, the error rates of some components are varied to determine how circuit reliability changes if components have different error rates.

4.6 Perturbing Component Error Rates

In the previous section, the error rate for all of the components was equivalent. This section investigates how changes in the reliability of one or two components impacts circuit reliability. In particular, the components that are impacted are the wire segments (both straight and bent) and majority gates. Since these components are the most frequently used, modifying their error rates should impact circuit reliability the most.

Figures 4.10 and 4.11 show the circuit reliabilities that result for 2-Mux and 4-Adder when the error rate for the wire segments (straight and bent, marked as WSB) and majority gates (marked Maj) are modified by an order of magnitude from the remainder of the components. The Lower curves refer to the case when the error rate for the marked components is one-tenth that of the component error rate shown on the x-axis. Similarly, the Higher curves refer to the case when the error rate for the marked components is ten times that of the component error rate shown on the x-axis. The Orig curve leaves the component error rate uniform (same results as Sec. 4.4).

The easiest result to see in Figs. 4.10 and 4.11 is that when the error rate of either the wire segments or majority gates improves (declines), the circuit reliability increases (decreases). Additionally, the change in reliability is greater for the wire segments than for the majority gates. The interesting aspect of these results is
that the change in reliability is greater when either of the component types (WSB or Maj) has an error rate higher than the remainder of the components rather than when one of the component types has an error rate lower than the other components. This suggests that if a component has an error rate higher than the other components, it should be made more reliable before trying to improve the
reliability of other components. In essence, there should be no weak links in the reliabilities of the circuit components.

Should multiple components be weak links or should the components demonstrate a nearly uniform reliability, a mechanism for selecting which component to improve is needed. The mechanism for doing this should identify which compo-
ponent(s) are most critical to circuit reliability in order to see the largest gains in circuit reliability. A method for identifying the critical component(s) of a circuit is demonstrated in the next section.

4.7 Linear Regression Analysis

A major goal for this chapter is not only to understand QCA reliability at the circuit level, but also to develop a guide for improving circuit reliability by enhancing device and/or component reliability. However, to successfully improve circuit reliability, the following questions must be answered: 1) which component is the most important and 2) how much enhancement is required.

In this section, two different methods (described in Sec. 3.2) are compared to determine which best identifies the most critical component(s) of the circuit. Both of these methods are based on the results of linear regression which generates a regression coefficient that represents the impact each component has on the reliability of the circuit. In the first method, the magnitude of the regression coefficient is directly used to determine the most critical component. A possible limitation of this method is that it may underestimate the impact of a component that is not frequently used. The second method accounts for this potential limitation by normalizing the regression coefficients by the number of instances of the component in the circuit. These two methods are compared by separately making the most critical component identified by each method perfect and calculating the reliability of the circuit. The method that generates the higher circuit reliability is favored.

One of the concerns outlined above was that component error rates that differ by several orders of magnitude were not considered. Using regression analysis,
this concern is alleviated by computing the circuit reliability for all combinations of component error rates that vary by several orders of magnitude. In particular, each component has its error rate stepped from $1.0 \times 10^{-8}$ to $1.0 \times 10^{-2}$ by orders of magnitude for the parity tree, multiplexer, and adder circuits. This results in $7^6$ (7 is the number of component error rates tested and 6 is the number of components) data points for each of the electrostatic circuits and $7^7$ data points for each of the magnetic circuits. Each data point consists of an error rate for each component and the resulting circuit reliability. The error rates are not increased above $1.0 \times 10^{-2}$ since this has a significantly negative impact on the quality of the regression results. This decrease in quality is a result of the circuit reliabilities staying near their RCR values rather than decreasing to zero at high error rates.

The results of this regression analysis (without normalization) are shown for the tested circuits in Table 4.2. The regression coefficients in the table are negative because increasing the component error rate, its $x_i$ value, decreases the circuit reliability. When examining a row of the table, the magnitude of the coefficient is what determines the impact of that variable on the circuit’s reliability. The larger the magnitude of the coefficient, the greater the impact of that variable’s influence on the circuit’s reliability. For the two input parity tree, this means that the majority gate (MAJ) has the greatest impact on reliability, followed by the straight wire (W), the single bend wire (SB), the inverter (NOT), and lastly, the T-fanout (TFAN). Regular fanouts are under the column FAN, while the magnetic crossover component is under the column XV.

Since the regression analysis only considers the data points for one circuit at a time, the coefficients of different circuits are not correlated in any way and comparing them has limited value. The only conclusion that can be drawn from
comparing the coefficients between circuits is that the small changes in component reliability have a more significant impact on circuit reliability as the circuit size increases; i.e. the magnitude of the regression coefficient increases as the circuit size increases.

In Table 4.3, the regression coefficients have been normalized by the number of instances of each component. If the components were ordered in this manner for the two-input parity tree, the ordering would have the majority gate as the most important component followed, in order, by the inverter, single bend wires, T-fanout, and the straight wires. For this specific circuit, this ordering places more emphasis on the inverter and significantly less on the straight wires.

To determine whether the regular or normalized regression results are best at selecting the most important component, the most critical component selected by each method is made perfect (non-faulty) and then compute the circuit reliability when the other components are faulty. A simple weighted method has been used to order the components when all of the circuits are considered. In this weighted method, the components are scored from 6 (for electrostatic circuits) or 7 (for magnetic circuits) going down to 1, with the most important component having the highest score. The component scores are then averaged for each technology implementation of the circuits. The component orderings resulting from this weighted method are shown in Table 4.4. For the parity tree circuits, the lack of wire crossings has no impact on these results being included in the electrostatic orderings. In the magnetic orderings, the lack of crossover components has no impact for the regression ordering when the parity tree circuits are not included and in the normalized ordering the NOT and XV would swap when the parity tree circuits are not considered.
Figure 4.12 shows the circuit reliabilities for these cases with the symbols and legend stating which component is perfect. For the electrostatic circuits, the majority gate was selected as the regression component and the inverter as the normalized component. In both Fig. 4.12(a) and Fig. 4.12(c), keeping the majority gate perfect improves circuit reliability the most. For the magnetic circuits, the wire was selected as the regression component and the majority gate was the normalized component. In Fig. 4.12(b) and the two larger circuits in Fig. 4.12(d) the perfect wire offers the highest circuit reliability. For the 2-Mux case in Fig. 4.12(d), the perfect majority gate is better, but for this specific circuit the regression coefficient for the majority gate is higher than the regression coefficient for the wire, thus it still draws us towards selecting the regression coefficient. For the parity tree circuits of Fig. 4.12(e) if the electrostatic component orderings (they nearly match the actual parity tree circuit orderings) are utilized, then the regression ordering is a greater reliability improvement than the normalized ordering. These results lead to the conclusion that the regression ordering is favored over the normalized ordering.

While no regression analysis has been done on the multiplier circuits, they have been analyzed with a single component being perfect and the remainder of the components having a uniform error rate. Tables 4.5 (magnetic version) and 4.6 (electrostatic version) show the reliabilities, given a constant error rate for the components in the leftmost column, when none of the components are perfect in the next column (which is equivalent to the results of Fig. 4.8(d)), and the remainder of the columns show the circuit reliability with each component individually being perfect. The columns with perfect components are sorted by higher reliabilities on the left and lower reliabilities on the right at the $1.00E-04$
error rate line.
<table>
<thead>
<tr>
<th>Circuit Size</th>
<th>Circuit Type</th>
<th>Components</th>
<th>Constant</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>MAJ</td>
<td>NOT</td>
</tr>
<tr>
<td>Parity Tree</td>
<td>2 Inputs</td>
<td>-3.11</td>
<td>-0.73</td>
</tr>
<tr>
<td></td>
<td>8 Inputs</td>
<td>-15.68</td>
<td>-4.09</td>
</tr>
<tr>
<td>Electrostatic Adders</td>
<td>1-Adder</td>
<td>-14.50</td>
<td>-3.95</td>
</tr>
<tr>
<td></td>
<td>3-Adder</td>
<td>-32.34</td>
<td>-9.72</td>
</tr>
<tr>
<td></td>
<td>4-Adder</td>
<td>-37.71</td>
<td>-11.89</td>
</tr>
<tr>
<td>Electrostatic Muxes</td>
<td>2-Mux</td>
<td>-6.52</td>
<td>-1.46</td>
</tr>
<tr>
<td></td>
<td>4-Mux</td>
<td>-18.52</td>
<td>-4.91</td>
</tr>
<tr>
<td></td>
<td>8-Mux</td>
<td>-23.63</td>
<td>-8.31</td>
</tr>
<tr>
<td>Circuit Size</td>
<td>Circuit Type</td>
<td>Components</td>
<td>Constant</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------</td>
<td>------------</td>
<td>----------</td>
</tr>
<tr>
<td></td>
<td>Magnetic Adders</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1-Adder</td>
<td>MAJ -2.63</td>
<td>NOT -1.20</td>
</tr>
<tr>
<td></td>
<td>2-Adder</td>
<td>MAJ -5.08</td>
<td>NOT -2.33</td>
</tr>
<tr>
<td></td>
<td>3-Adder</td>
<td>MAJ -7.37</td>
<td>NOT -3.40</td>
</tr>
<tr>
<td></td>
<td>4-Adder</td>
<td>MAJ -9.51</td>
<td>NOT -4.41</td>
</tr>
<tr>
<td></td>
<td>Magnetic Muxes</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2-Mux</td>
<td>MAJ -2.42</td>
<td>NOT -0.36</td>
</tr>
<tr>
<td></td>
<td>4-Mux</td>
<td>MAJ -5.52</td>
<td>NOT -0.68</td>
</tr>
<tr>
<td></td>
<td>8-Mux</td>
<td>MAJ -9.99</td>
<td>NOT -1.33</td>
</tr>
</tbody>
</table>
### TABLE 4.3

NORMALIZED LINEAR REGRESSION RESULTS

<table>
<thead>
<tr>
<th>Circuit Size</th>
<th>Circuit Type</th>
<th>Components</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>MAJ</td>
</tr>
<tr>
<td>Parity Tree</td>
<td>2 Inputs</td>
<td>-0.77</td>
</tr>
<tr>
<td></td>
<td>8 Inputs</td>
<td>-0.56</td>
</tr>
<tr>
<td>Electrostatic Adders</td>
<td>1-Adder</td>
<td>-0.37</td>
</tr>
<tr>
<td></td>
<td>2-Adder</td>
<td>-0.32</td>
</tr>
<tr>
<td></td>
<td>3-Adder</td>
<td>-0.28</td>
</tr>
<tr>
<td></td>
<td>4-Adder</td>
<td>-0.24</td>
</tr>
<tr>
<td>Electrostatic Muxes</td>
<td>2-Mux</td>
<td>-0.43</td>
</tr>
<tr>
<td></td>
<td>4-Mux</td>
<td>-0.17</td>
</tr>
<tr>
<td></td>
<td>8-Mux</td>
<td>-0.067</td>
</tr>
<tr>
<td>Circuit Size</td>
<td>Circuit Type</td>
<td>Components</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------</td>
<td>------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAJ</td>
</tr>
<tr>
<td>Magnetic Adders</td>
<td>1-Adder</td>
<td>-0.88</td>
</tr>
<tr>
<td></td>
<td>2-Adder</td>
<td>-0.85</td>
</tr>
<tr>
<td></td>
<td>3-Adder</td>
<td>-0.82</td>
</tr>
<tr>
<td></td>
<td>4-Adder</td>
<td>-0.79</td>
</tr>
<tr>
<td>Magnetic Muxes</td>
<td>2-Mux</td>
<td>-0.81</td>
</tr>
<tr>
<td></td>
<td>4-Mux</td>
<td>-0.50</td>
</tr>
<tr>
<td></td>
<td>8-Mux</td>
<td>-0.37</td>
</tr>
</tbody>
</table>
TABLE 4.4

COMPONENT IMPORTANCE ORDERINGS (IN DESCENDING ORDER).

<table>
<thead>
<tr>
<th>Regression Ordering</th>
<th>Normalized Ordering</th>
</tr>
</thead>
<tbody>
<tr>
<td>Electrostatic</td>
<td>Magnetic</td>
</tr>
<tr>
<td>MAJ</td>
<td>W</td>
</tr>
<tr>
<td>W</td>
<td>MAJ</td>
</tr>
<tr>
<td>SB</td>
<td>SB</td>
</tr>
<tr>
<td>FAN</td>
<td>FAN</td>
</tr>
<tr>
<td>NOT</td>
<td>XV</td>
</tr>
<tr>
<td>TFAN</td>
<td>NOT</td>
</tr>
<tr>
<td>TFAN</td>
<td></td>
</tr>
</tbody>
</table>

Of particular interest in the above tables is the ordering of the perfect components. For the magnetic multiplier, having a perfect majority gate will improve the reliability more than having a perfect wire component. This changes slightly for the electrostatic multiplier at error rates at and below $5.00E-03$ when a perfect wire would be better than a perfect majority gate. However, in both cases, the wire and majority gate have a more significant impact than the other components. While this ordering deviates slightly from the orderings shown in Table 4.4, it remains clear that the regression analysis is the preferred method for determining
Figure 4.12. Circuit reliabilities with single perfect component as listed in legend. Other components have a uniform error rate.
which component is most critical to the reliability of a given circuit.

The results of Tables 4.5 and 4.6 also demonstrate how circuit reliability can be improved just by improving the reliability of a single component. These gains are investigated in more depth in the next section.
# Table 4.5

Magnetic multiplier reliability with zero or one perfect component.

<table>
<thead>
<tr>
<th>Error Rate</th>
<th>None</th>
<th>Maj. Gate</th>
<th>Wire</th>
<th>Reg. Fan.</th>
<th>Crossover</th>
<th>Single Bend</th>
<th>Inverter</th>
<th>T-Fanout</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00E-07</td>
<td>0.999992</td>
<td>0.999994</td>
<td>0.999994</td>
<td>0.999993</td>
<td>0.999993</td>
<td>0.999992</td>
<td>0.999992</td>
<td>0.999992</td>
</tr>
<tr>
<td>1.00E-06</td>
<td>0.999916</td>
<td>0.999938</td>
<td>0.999936</td>
<td>0.999934</td>
<td>0.999927</td>
<td>0.999924</td>
<td>0.999919</td>
<td>0.999918</td>
</tr>
<tr>
<td>1.00E-05</td>
<td>0.999159</td>
<td>0.999379</td>
<td>0.999365</td>
<td>0.999337</td>
<td>0.999267</td>
<td>0.999237</td>
<td>0.999187</td>
<td>0.99918</td>
</tr>
<tr>
<td>1.00E-04</td>
<td>0.991624</td>
<td>0.993812</td>
<td>0.993668</td>
<td>0.993395</td>
<td>0.992702</td>
<td>0.992396</td>
<td>0.9919</td>
<td>0.991835</td>
</tr>
<tr>
<td>2.50E-04</td>
<td>0.97921</td>
<td>0.984613</td>
<td>0.984255</td>
<td>0.983582</td>
<td>0.981872</td>
<td>0.981115</td>
<td>0.97989</td>
<td>0.979731</td>
</tr>
<tr>
<td>5.00E-04</td>
<td>0.958918</td>
<td>0.969502</td>
<td>0.96879</td>
<td>0.967476</td>
<td>0.964126</td>
<td>0.96264</td>
<td>0.96024</td>
<td>0.959933</td>
</tr>
<tr>
<td>7.50E-04</td>
<td>0.939111</td>
<td>0.954663</td>
<td>0.953601</td>
<td>0.951674</td>
<td>0.946753</td>
<td>0.944567</td>
<td>0.941039</td>
<td>0.940594</td>
</tr>
<tr>
<td>1.00E-03</td>
<td>0.919776</td>
<td>0.940089</td>
<td>0.938681</td>
<td>0.936171</td>
<td>0.929744</td>
<td>0.926886</td>
<td>0.922275</td>
<td>0.921704</td>
</tr>
<tr>
<td>2.50E-03</td>
<td>0.813029</td>
<td>0.857954</td>
<td>0.854547</td>
<td>0.849093</td>
<td>0.834895</td>
<td>0.828528</td>
<td>0.818308</td>
<td>0.817172</td>
</tr>
</tbody>
</table>
TABLE 4.5

*Continued*

<table>
<thead>
<tr>
<th>Error Rate</th>
<th>None</th>
<th>Maj. Gate</th>
<th>Wire</th>
<th>Reg. Fan.</th>
<th>Crossover</th>
<th>Single Bend</th>
<th>Inverter</th>
<th>T-Fanout</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.00E-03</td>
<td>0.665699</td>
<td>0.739189</td>
<td>0.732786</td>
<td>0.724155</td>
<td>0.700988</td>
<td>0.690459</td>
<td>0.673667</td>
<td>0.672145</td>
</tr>
<tr>
<td>7.50E-03</td>
<td>0.54907</td>
<td>0.639617</td>
<td>0.630651</td>
<td>0.620415</td>
<td>0.591973</td>
<td>0.578872</td>
<td>0.55808</td>
<td>0.55661</td>
</tr>
<tr>
<td>1.00E-02</td>
<td>0.456312</td>
<td>0.555904</td>
<td>0.544801</td>
<td>0.53402</td>
<td>0.502879</td>
<td>0.488348</td>
<td>0.465349</td>
<td>0.464168</td>
</tr>
<tr>
<td>2.50E-02</td>
<td>0.177947</td>
<td>0.263942</td>
<td>0.247001</td>
<td>0.240787</td>
<td>0.21517</td>
<td>0.20235</td>
<td>0.181514</td>
<td>0.182539</td>
</tr>
<tr>
<td>5.00E-02</td>
<td>0.068488</td>
<td>0.110357</td>
<td>0.095211</td>
<td>0.09501</td>
<td>0.084534</td>
<td>0.078796</td>
<td>0.067419</td>
<td>0.069124</td>
</tr>
<tr>
<td>7.50E-02</td>
<td>0.043302</td>
<td>0.06674</td>
<td>0.054789</td>
<td>0.055511</td>
<td>0.051306</td>
<td>0.048742</td>
<td>0.041803</td>
<td>0.043065</td>
</tr>
<tr>
<td>1.00E-01</td>
<td>0.03374</td>
<td>0.049933</td>
<td>0.040034</td>
<td>0.040346</td>
<td>0.038716</td>
<td>0.037399</td>
<td>0.032554</td>
<td>0.03344</td>
</tr>
</tbody>
</table>
**TABLE 4.6**

**ELECTROSTATIC MULTIPLIER RELIABILITY WITH ZERO OR ONE PERFECT COMPONENT.**

<table>
<thead>
<tr>
<th>Error Rate</th>
<th>None</th>
<th>Wire</th>
<th>Maj. Gate</th>
<th>Reg. Fan.</th>
<th>Single Bend</th>
<th>Inverter</th>
<th>T-Fanout</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00E-07</td>
<td>0.999954</td>
<td>0.999967</td>
<td>0.999967</td>
<td>0.999963</td>
<td>0.99996</td>
<td>0.99956</td>
<td>0.99956</td>
</tr>
<tr>
<td>1.00E-06</td>
<td>0.999538</td>
<td>0.999673</td>
<td>0.999666</td>
<td>0.999632</td>
<td>0.9996</td>
<td>0.999563</td>
<td>0.999557</td>
</tr>
<tr>
<td>1.00E-05</td>
<td>0.995393</td>
<td>0.996737</td>
<td>0.996667</td>
<td>0.996329</td>
<td>0.996004</td>
<td>0.995635</td>
<td>0.995583</td>
</tr>
<tr>
<td>1.00E-04</td>
<td>0.955035</td>
<td>0.967918</td>
<td>0.967255</td>
<td>0.964</td>
<td>0.960884</td>
<td>0.957334</td>
<td>0.956844</td>
</tr>
<tr>
<td>2.50E-04</td>
<td>0.891978</td>
<td>0.922014</td>
<td>0.920502</td>
<td>0.912838</td>
<td>0.905566</td>
<td>0.897258</td>
<td>0.896139</td>
</tr>
<tr>
<td>5.00E-04</td>
<td>0.797517</td>
<td>0.851039</td>
<td>0.848456</td>
<td>0.834575</td>
<td>0.821598</td>
<td>0.806688</td>
<td>0.804763</td>
</tr>
<tr>
<td>7.50E-04</td>
<td>0.714793</td>
<td>0.786403</td>
<td>0.783109</td>
<td>0.764232</td>
<td>0.746845</td>
<td>0.726748</td>
<td>0.724265</td>
</tr>
<tr>
<td>1.00E-03</td>
<td>0.642234</td>
<td>0.727499</td>
<td>0.723788</td>
<td>0.700944</td>
<td>0.680214</td>
<td>0.656099</td>
<td>0.653253</td>
</tr>
<tr>
<td>2.50E-03</td>
<td>0.356589</td>
<td>0.467401</td>
<td>0.464734</td>
<td>0.432053</td>
<td>0.40495</td>
<td>0.37186</td>
<td>0.369038</td>
</tr>
</tbody>
</table>
TABLE 4.6

Continued

<table>
<thead>
<tr>
<th>Error Rate</th>
<th>None</th>
<th>Wire</th>
<th>Maj. Gate</th>
<th>Reg. Fan.</th>
<th>Single Bend</th>
<th>Inverter</th>
<th>T-Fanout</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.00E-03</td>
<td>0.164901</td>
<td>0.247046</td>
<td>0.249321</td>
<td>0.220952</td>
<td>0.200779</td>
<td>0.173161</td>
<td>0.172206</td>
</tr>
<tr>
<td>7.50E-03</td>
<td>0.0965</td>
<td>0.148306</td>
<td>0.15368</td>
<td>0.132772</td>
<td>0.120093</td>
<td>0.09998</td>
<td>0.100149</td>
</tr>
<tr>
<td>1.00E-02</td>
<td>0.067221</td>
<td>0.100084</td>
<td>0.10663</td>
<td>0.091204</td>
<td>0.083273</td>
<td>0.068415</td>
<td>0.069043</td>
</tr>
<tr>
<td>2.50E-02</td>
<td>0.030388</td>
<td>0.036385</td>
<td>0.041216</td>
<td>0.035668</td>
<td>0.034816</td>
<td>0.029647</td>
<td>0.030284</td>
</tr>
<tr>
<td>5.00E-02</td>
<td>0.022581</td>
<td>0.024694</td>
<td>0.027984</td>
<td>0.024247</td>
<td>0.024651</td>
<td>0.021868</td>
<td>0.02231</td>
</tr>
<tr>
<td>7.50E-02</td>
<td>0.020037</td>
<td>0.021581</td>
<td>0.024486</td>
<td>0.020838</td>
<td>0.021574</td>
<td>0.019446</td>
<td>0.019816</td>
</tr>
<tr>
<td>1.00E-01</td>
<td>0.0185</td>
<td>0.019952</td>
<td>0.022695</td>
<td>0.018921</td>
<td>0.019825</td>
<td>0.01803</td>
<td>0.018348</td>
</tr>
</tbody>
</table>
4.8 Improving Circuit Reliability

Having identified which component was most critical in each circuit with linear regression, the potential reliability improvements by making a single component more robust can be quantified. From the results in Sec. 4.7 either the majority gate or wire could be selected as the most critical component. Selecting the wire over the majority gate as the most critical component is advantageous since enhancing its reliability is straightforward, particularly for ESQCA.

Figure 4.13 shows how the reliabilities for the parity tree, multiplexer, and adder circuits changes as the reliability of the straight wire component varies by several orders of magnitude. These graphs clearly show that an extremely high error rate for the wire causes the circuit to be nearly useless. As the error rate for the wire is reduced by two orders of magnitude, the circuit reliability increases significantly. Further reductions in the wire error rate are excluded in Fig. 4.13(a) through Fig. 4.13(e) since they are nearly indistinguishable from the perfect wire case. Figure 4.13(f) is a zoomed in version of Fig. 4.13(a) to show the impact of lowering the wire error rate beyond $1.0 \times 10^{-3}$. The $1.0 \times 10^{-3}$ line for the 4-Adder is not visible since its highest reliability is approximately 0.915 and the lowest reliability shown on this graph is 0.97.
Figure 4.13. Circuit reliabilities as the wire component error rate changes (see legend for this rate); other components have a uniform error rate (x-axis).
The previous chapter developed an understanding of reliability at the circuit level in QCA circuits; in particular, the previous chapter identified target reliabilities for the components. In this chapter, various system level approaches to improving reliability are evaluated. These approaches, programmable and redundant architectures, were introduced in Sec. 2.2.

A programmable architecture, the programmable logic array (PLA), is evaluated first. The second section investigates how reliability is impacted for several variants of NMR for generic modules. The third section evaluates three potential MQCA wire organizations and provides a set of guidelines as to which organization should be favored when only reliability is considered.

In this chapter, as in the previous chapter, the ESQCA circuits and systems utilize the logical wire crossing circuit while the MQCA circuits and systems utilize a wire crossing component. Additionally, the PTM method is used to generate the results in this chapter.

5.1 Programmable Logic Arrays

This section presents demonstrates the difference in reliability for a simple full adder designed using custom logic and implemented on a PLA.
5.1.1 Adder Implementations

The custom logic implementation of the full adder is shown in Fig. 5.1. This adder has three single-bit inputs and two single-bit outputs.

The QCA implementation of a PLA is capable of computing multiple functions in sum of products form [72]. The PLA operates by computing a series of product terms (inputs that are ANDed together, also known as minterms) in the AND plane and then generates a set of output functions by summing these product terms (products are ORed together) in the OR plane. Figure 5.2(c) shows the basic structure of a QCA-based PLA that has two input bits (literals), computes two products, and is capable of computing two output functions.

The AND plane consists of a series of AND cells. An AND cell, shown in Fig. 5.2(a) is capable of acting like a wire or contributing to the product. If the select bit of the AND cell is set to 0, the product is ANDed with the literal (input); if the select bit is 1, the product is passed through (AND cell acts like a wire).
(a) PLA AND cell.  
(b) PLA OR cell.  
(c) Small PLA. Black dots are wire crossings.

Figure 5.2: PLA construction.

Similarly, the OR cell consists of a series of OR cell. An OR cell, shown in Fig. 5.2(b) can act like a wire or contribute to the function output. If the select bit of the OR cell is 1, the output function is ORed with the minterm; if the select bit is 0, the OR cell acts like a wire.

Rather than model the process of routing a zero and a one throughout the entire PLA, a simple reduction is used at each cell to compute its PTM. This reduction first computes, as a unique circuit, the reliability of the literal (implicant), select bit, and the OR (AND) gate within the AND (OR) cell. Once this reliability is computed, a new wire like element is created and used as an input, along with the implicant (function), to the second gate in the AND (OR) cell. All inputs to the PLA from outside of the AND and OR planes are assumed to be error-free. These inputs are the literals (function input bits), the minterm lines
which are initialized to 1, and the sum lines which are initialized to 0.

The PLA for the full adder uses six literals (three input bits and their complements), seven minterm lines, and two function lines. The functions computed by the PLA are shown in Eq. 5.1 and 5.2. Implementing the adder on the PLA requires 42 AND cells and 14 OR cells, which is a significant increase in logic over the custom implementation.

\[
f_1 = \text{sum} = ab'c'_\text{in} + a'bc'_\text{in} + a'b'c_\text{in} + abc_\text{in}
\]

\[
f_2 = c_\text{out} = ab + ac_\text{in} + bc_\text{in}
\]

5.1.2 Results

The methodology of Sec. 4.4 for calculating the reliability of circuits with uniform component error rates is used in this section. Additionally, the custom logic adder results from Sec. 4.4 are reused here.

The main result of this work is presented in Fig. 5.3. In Fig. 5.3(a), the results are for low component error rates, while the results for higher component error rates are in Fig. 5.3(b). These graphs have been divided to show different reliability ranges for the adder designs. The circuit reliabilities have a soft floor of 0.25 since this is the RCR for a two-bit output circuit.

From the graphs, two major trends are observed: 1) the custom logic implementations show a higher reliability than the PLA implementations and 2) the circuits using MQCA demonstrate a higher reliability. In particular, note that for component error rates of greater than $5.0 \times 10^{-3}$ for the ESQCA PLA and $5.0 \times 10^{-2}$ for the MQCA PLA these circuits provide nearly random outputs. These results are unsurprising once the component counts of Table 5.1 are considered since the
circuits with the highest component counts have the lowest reliabilities. This is mainly a result of having to use a logical crossing circuit for the ESQCA implementations and the increase in logic and interconnect needed to implement a regular architecture like a PLA.

5.1.3 Limitations

While the results of this section suggest that custom logic is preferable to programmable logic, there are three important factors that prevent drawing a strong conclusion. These factors are 1) comparing the adder circuit is harsh on the PLA since there is no overlap in the sum and carry out functions; 2) the adder circuit maps extremely well to majority-based logic; and 3) this work has not considered the potential (dis)advantages that may be present when spare rows and columns are available in a PLA. Extending this work to consider circuits from the LGSynth and ISCAS benchmark suites may allow for a stronger conclusion to be made. Analyzing these benchmark circuits may require a method like the
one proposed in [10] since the PTMs required to test the PLA are close to the maximum size that Matlab can handle.

### 5.2 N-Modular Redundancy with Generic Modules

A system that implements NMR has a base module (circuit) that is replicated \( N - 1 \) times, thus \( N \) copies of the module are in the system. The same output bit from each module is then routed into a voter (voting logic) that will select the majority value of the \( N \) copies of the output bit that the voter sees, provided that \( N \) is odd. In this section, several variations of NMR are investigated to determine if the reliability of a system using NMR is higher than that of the single base module. A comprehensive review of NMR techniques is provided in Sec. 2.2.2.2.

In this section, the base module is simply a black-box circuit that can implement any function. This type of module is termed a “generic module” for this...
section. A module that implements a specific function (such as an adder or multiplexer) is termed a “real circuit” in this section. These module types, generic and real circuits, must be treated differently because the reliability of the NMR system can vary as the function implemented by a module changes. The system reliability can change as the function implemented by the module changes because, with faulty routing components, the reliability of each input bit when it enters the module can be different. Similar behavior is seen between the modules and the voters as well.

To minimize the impact of different module functions causing different system reliabilities, multiple ITMs and PTMs are generated for a generic module of a given size. For this section, the size of a module refers to the number of input and output bits the module has. The final system reliability is an average of the system reliabilities generated by each ITM (and PTM) combination. Since smaller modules cannot implement a wide range of functions, the averaging process can cause roughness in the resulting reliability curves.

For reference, the term “non-module components” collectively refers to the signal routing and voting components needed to implement NMR. Throughout this section, the error rate for these components is uniform (unless otherwise noted).

The first subsection examines the potential reliability gains that can be achieved using only a single layer of replicated modules. In the second subsection, cascaded TMR implementations are considered. The third subsection performs a regression analysis on the non-cascaded modules to determine which components are most critical. Lastly, the implications of using NMR on factors like power, area, and speed are discussed.
For this section, the following labels are used:

- TMR-NR: TMR with non-replicated voters.
- TMR-Rep: TMR with replicated voters.
- QMR: Quintuple Modular Redundancy

5.2.1 Non-Cascaded NMR

This subsection examines basic forms of NMR where only a single set of replicated modules and voters exist. In particular, TMR-NR, TMR-Rep, and QMR are studied in this subsection. The first step taken in this subsection is that regions where the non-module component error rate is low enough for an NMR system to have a higher reliability than a generic module with a fixed reliability are identified. These regions are compared to a subset of the results in Sec. 4.4 to extrapolate how NMR may impact real circuits. Lastly, the system reliabilities that result from each NMR variant are compared to determine which variant is favored from a reliability perspective.

5.2.1.1 NMR Reliability Gains

For NMR to be effective, the reliability of an NMR system (with $N$ total copies of the generic module) must be higher than the reliability of a generic module. To determine if NMR is effective, the reliability of the generic module is held constant at $R_{gm} = 0.9999 = 1 - 1 \times 10^{-4}$ and the error rate for the non-module components is reduced until the reliability of the NMR system is higher than that of the generic module. This test identifies ranges of non-module component error rates where the NMR system has a higher reliability than the generic module.
Table 5.2 identifies the highest component error rate where the NMR-based system has a higher reliability than the generic module for a variety of module sizes. Mathematically, this is when the reliability of the NMR system has a higher reliability than the generic module, i.e. $R_{\text{NMR}} \geq (R_{gm} = 0.9999)$. For each of the NMR techniques, there are two notable trends in this table: 1) as the number of inputs increases, the component error rate required for NMR to be better decreases and 2) a similar trend is observed as the number of outputs increases. This is unsurprising since increasing the number of inputs or outputs in the module(s) increases the signal routing complexity of the system. Another observation is that the above trends are generally more pronounced for ESQCA than MQCA, and MQCA can tolerate higher component error rates. These observations are expected because the only difference between the two implementations is in their signal routing which is more complex (more components) for ESQCA than for MQCA.
TABLE 5.2

LARGEST NON-MODULE COMPONENT ERROR RATE WHERE THE RELIABILITY OF THE NMR VERSION IS GREATER THAN $R_{gm} = 0.9999$.

<table>
<thead>
<tr>
<th>NMR Technique</th>
<th>Num. Inputs</th>
<th>QCA Implementation &amp; Num. Output Bits</th>
<th>ESQCA</th>
<th>MQCA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TMR-NR</td>
<td>2</td>
<td>1×10^{-5}</td>
<td>2.5×10^{-6}</td>
<td>2.5×10^{-5}</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1×10^{-6}</td>
<td>1×10^{-6}</td>
<td>1×10^{-6}</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>7.5×10^{-7}</td>
<td>5×10^{-7}</td>
<td>5×10^{-7}</td>
</tr>
<tr>
<td>TMR, REP</td>
<td>2</td>
<td>5×10^{-6}</td>
<td>2.5×10^{-6}</td>
<td>5×10^{-5}</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1×10^{-6}</td>
<td>1×10^{-6}</td>
<td>1×10^{-6}</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>1×10^{-6}</td>
<td>5×10^{-7}</td>
<td>5×10^{-7}</td>
</tr>
<tr>
<td>QMR</td>
<td>2</td>
<td>5×10^{-6}</td>
<td>2.5×10^{-6}</td>
<td>2.5×10^{-5}</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>2.5×10^{-6}</td>
<td>1×10^{-6}</td>
<td>2.5×10^{-5}</td>
</tr>
</tbody>
</table>
5.2.1.2 Real Modules v. Generic Modules

The previous section assumed a specific value for the reliability of the generic module \((R_{gm})\) and identified a region of non-module component error rates where the NMR system had a higher reliability than the generic module. In this section, we will compare the reliability of a circuit tested in Sec. 4.4 to \(R_{gm}\) and the NMR system reliabilities of the previous section. This is done to extrapolate how NMR may impact reliability for a specific circuit based on the system reliabilities generated when using generic modules.

To accomplish this extrapolation, two non-module component error rates will need to be identified. The first is the non-module component error rate, \(F_1\), when the reliability of the real circuit equals that of the generic module; i.e., what is the non-module component error rate where \(R_{real} = R_{gm}\). The second value we need is the non-module component error rate, \(F_2\), when the reliability of the NMR system using a generic module is equivalent to the reliability of the generic module; i.e. what is the non-module component error rate where \(R_{NMR} = R_{gm}\). When \(F_1 > F_2\), the NMR technique is not effective (reduces reliability); if \(F_1 < F_2\) then the NMR technique is effective.

This comparison is illustrated via the results in Fig. 5.4 which shows the system reliabilities that result from using TMR-NR with a generic module in both QCA technologies (lines marked TMR, NR, ESQCA and TMR, NR, MQCA), the reliability of a generic two-input, one-output module \(R_{gm} = 0.9999\) (line marked as Module), and the reliability, from Sec. 4.4 of the two-input parity tree with all of its components having a uniform error rate (line marked as 2 Input PTree). For this figure, \(F_1 \simeq 1.0 \times 10^{-5}\) and there are two values of \(F_2\). \(F_{2,E}\) is used to denote \(F_2\) for the ESQCA TMR implementation and \(F_{2,M}\) is used to denote \(F_2\) for the
MQCA TMR implementation. Respectively, \( F_{2,E} \simeq 7 \times 10^{-6} \) and \( F_{2,M} \simeq 5 \times 10^{-5} \).

Having identified the \( F_1 \) and \( F_2 \) values, we can determine if a specific NMR implementation is effective. For this example, since \( F_1 > F_{2,E} \), the reliability of an ESQCA-based TMR-NR system with the module being a two-input, one-output parity tree will be lower than the reliability of the parity tree itself. However, with \( F_1 < F_{2,M} \), the reliability of an MQCA-based TMR-NR system with the module being a two-input, one-output parity tree will be higher than the reliability of the parity tree itself.

![Figure 5.4. Reliability of a two-input, one-output parity tree PTree from Sec. 4.4 and the reliabilities of TMR based systems using a generic two-input, one-output module with a reliability of 0.9999.](image-url)
If the component error rates identified in Table 5.2 are compared to the circuit reliabilities calculated in Sec. 4.4, an estimate as to whether NMR might be effective for real circuits can be made. To achieve a $R_{\text{real}} = 0.9999$ for a two-input (plus one select bit), one-output multiplexer implemented in ESQCA and MQCA non-module component error rates ($F_1$ values) of $2.5 \times 10^{-6}$ and $1 \times 10^{-5}$ are respectively needed. For a single bit slice of an adder (three-input, two-output) component error rates of $1 \times 10^{-6}$ and $5 \times 10^{-6}$ are needed if the adder is implemented in ESQCA and MQCA respectively.

By comparing these $F_1$ values to the $F_2$ values in Table 5.2, we observe that the basic NMR configurations considered in this section are, at best, ineffective for improving reliability in ESQCA. However, each of these basic NMR techniques improves the reliability of the circuits presented here if MQCA is utilized. Having shown that NMR can be utilized to improve the reliability of both generic modules and real circuits, we now determine which NMR technique(s) should be favored from a reliability perspective.

5.2.1.3 Comparing NMR Configurations

Figure 5.5 presents the system reliabilities that result when using generic modules for each of the NMR techniques considered in this section. The NR and Rep labels respectively refer to non-replicated and replicated voter implementations of TMR. Additionally, the PIC labels refer to Perfect InterConnect and are used as a reference line to demonstrate the system reliabilities that would result if classical models of computing reliability were used. Recall that these classical models, such as those presented in Sec. 2.2.2.2 and [62], consider the interconnect to be perfect.

The first pair of results, Figs. 5.5(a) and 5.5(b), considers three-input, one-
Figure 5.5: NMR reliabilities for modules with $R_0 = 1 - 1 \times 10^{-4} = 0.9999$. If the NMR reliability is below $R_0$, the redundant system has a lower reliability than the non-redundant system. *PIC* refers to Perfect InterConnect as is assumed by classical reliability models. For clarity, *QMR, PIC* is not shown since it overlaps with *TMR, NR, PIC*.

Output module using ESQCA and MQCA respectively. For this pair, QMR has the highest reliability while TMR-NR has the lowest. The second pair of results, Figs. 5.5(c) and 5.5(d), considers a three-input, two-output module implemented in both technologies. For the ESQCA version, QMR has the highest reliability while TMR-Rep is slightly higher than TMR-NR until the non-module component error rate is $\geq 2.5 \times 10^{-6}$ and then TMR-NR is better. For the MQCA version, QMR and TMR-Rep are nearly equivalent in reliability with QMR being just
a bit lower than TMR-Rep and TMR-NR having the lowest reliability. As the number of inputs and outputs increases, TMR-Rep is likely to outperform QMR since the number of crossovers for QMR grows more quickly than it does for TMR-Rep.

From a reliability perspective, it appears that QMR and TMR-Rep tend to provide similar results, at least for the module sizes tested here. As mentioned, TMR-Rep is likely to be the better selection for modules with large numbers of inputs and outputs due to its simpler signal routing. However, outside of reliability, factors such as area and speed may play a role in selecting an NMR technique if one is desired. The impact of these factors is addressed later.

5.2.2 Cascaded TMR

Another common method for improving system reliability with NMR, particularly for TMR, is to divide the original module into smaller sub-modules and to use TMR between the sub-modules. This form of TMR is termed cascaded TMR, with Fig. 5.6 showing a cascaded version of TMR with replicated voters. In this section, results for one-input, one-output modules and two-input, two-output modules are compared. Only modules with equal numbers of inputs and outputs are considered due to the matrix multiplication steps required by using PTMs.

For this section, a segment is a replicated set of modules and the voter(s) that immediately follow the replicated set of modules.

Figure 5.7 shows the impact of varying the reliability for a single module with no redundancy, denoted $R_0$ (value of which is shown in each legend entry), while maintaining a modest non-module component error rate of $1 \times 10^{-4}$ (non-module
component reliability of 0.9999). For this figure, \( NR \) refers to non-replicated voters and \( Rep \) refers to replicated voters. Since no crossovers are necessary for the non-replicated voters case in Fig. 5.7(a), the ESQCA and MQCA lines would be equivalent. The lines marked as \( Thy \) assume the interconnect to be perfect, thus only the modules and voters are faulty. When using replicated voters, a final voter is necessary to reduce the replicated outputs to a single output. As a result, a final multiplication by \( R_v \) is done in each of these cases.

For the results presented in Fig. 5.7(a), when \( R_0 \) and the non-module component reliability are similar, using cascaded TMR is of no value. However, when \( R_0 \) is significantly lower than the non-module component reliability, dividing the system into segments can provide significant reliability improvements.

One thing to note though is that the number of segments to use for peak (highest) reliability can vary. For example, in this graph, the \( Thy \) \( NR \), 0.9 result has a peak reliability at 20 segments and the \( QCA \) \( NR \), 0.9 result has a peak reliability at 15 segments. This peak reliability point is lower for the \( QCA \) \( NR \) case because it considers both the interconnect and voters to be faulty as opposed to the \( Thy \) \( NR \) case which consider the voters to be faulty but the interconnect to be perfect. Additionally, the reliability of the voter between each pair of segments
Figure 5.7. System reliability of a cascaded TMR system with a non-module component error rate of $1 \times 10^{-4}$ and $R_0 = 0.9999 = 1 - 1 \times 10^{-4}$ and $R_0 = 0.9$ as shown in the legend.

becomes the limiting factor in improving reliability since it acts as a single failure point. This trend is reduced for the replicated voters cases since there is not a single point (voter in this case) that puts an upper limit on reliability.

Similar trends are seen, albeit at lower system reliabilities, for the two-input, two-output modules of Fig. 5.7(b). When comparing these graphs, the number of
segments to use for peak reliability is lower for the two-input, two-output modules than the one-input, one-output modules. The majority of this difference is likely a result of the increased signal routing, but some is attributable to the use of more voters in implementing TMR.

Further study is necessary, but it is possible that larger modules may show that cascaded TMR offers no improvement to system reliability. Additionally, since the PTM modeling method requires matrix multiplication, it is not possible to investigating how reliability may be impacted when each segment may vary in the number of input and output bits that it has.

5.2.3 Linear Regression Results

As in Sec. 4.7, regression analysis is used to determine which part of a circuit is most sensitive to changes in reliability. Here, regression analysis is performed on two and three input modules with one to three outputs for a non-cascaded TMR-NR configuration. Rather than test each component individually, they are grouped together based on their purpose. For this section, the groupings are Logic (MAJ, AND, OR, NOT), Wiring (W, SB), Fanouts (FAN, TFAN, FAN3), Crossovers (for magnetic implementations), and Modules. The error rates for each of these groups has ranged from $1 \times 10^{-8}$ to $1 \times 10^{-2}$ by orders of magnitude (all combinations tested).

Tables 5.3 and 5.4 are the regression coefficients that result from this analysis. In these tables each column must be considered individually; the coefficients cannot be compared between columns since each column represents a different circuit. Recall that the larger the magnitude of the coefficient, the greater that group of components impacts reliability.
For all cases, the wiring and logic groupings have the largest impact on system reliability. These coefficients suggest that module reliability has almost no real impact on system reliability. Additionally, since this analysis is only conducted for the TMR-NR configuration the regression coefficient for the modules is likely to be smaller for the other NMR configurations. This possibility is due to TMR-NR having the smallest number of non-module components. Overall, this strongly suggests that the reliability of the logic and wiring between the modules will determine the overall system reliability.
### TABLE 5.3

REGRESSION COEFFICIENTS FOR TWO INPUT MODULES.

|                | Electrostatic | | Magnetic |
|----------------|---------------|-----------------|
|                | One Output    | Two Output      | Three Output | One Output | Two Output | Three Output |
| Logic          | -4.9783       | -11.5877        | -22.4486     | -0.9979    | -1.9776    | -2.9463      |
| Wiring         | -5.8052       | -15.4439        | -31.1184     | -0.5619    | -1.1027    | -1.9121      |
| Fanouts        | -3.9613       | -8.4430         | -15.9296     | -0.7723    | -1.1651    | -1.3698      |
| Crossovers     |               |                 |              | -0.4119    | -0.7010    | -1.0386      |
| Modules        | -0.0445       | -0.2165         | -0.3115      | -0.0438    | -0.0583    | -0.0885      |
| Constant       | 0.9987        | 1.0011          | 1.0034       | 1          | 1.0001     | 1.0004       |
### TABLE 5.4

REGRESSION COEFFICIENTS FOR THREE INPUT MODULES.

<table>
<thead>
<tr>
<th></th>
<th>Electrostatic</th>
<th></th>
<th>Magnetic</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>One Output</td>
<td>Two Output</td>
<td>Three Output</td>
</tr>
<tr>
<td>Logic</td>
<td>-11.0164</td>
<td>-21.2017</td>
<td>-32.6620</td>
</tr>
<tr>
<td>Fanouts</td>
<td>-8.8271</td>
<td>-16.5871</td>
<td>-25.0298</td>
</tr>
<tr>
<td>Crossovers</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Modules</td>
<td>-0.1325</td>
<td>-0.1830</td>
<td>-0.2606</td>
</tr>
<tr>
<td>Constant</td>
<td>0.9944</td>
<td>0.9932</td>
<td>0.9908</td>
</tr>
</tbody>
</table>
5.2.4 Implications

While NMR has been shown to improve reliability in certain scenarios, implementing NMR will impose costs on the system. Three of these costs: power, area, and speed, are discussed below.

One of the largest factors that will dictate how QCA technology will scale into large (i.e. multi-million devices) systems is how the devices interact with the clocking structure. These clocking structures are likely to be implemented in silicon technology, and as such, will be subject to the process technology rules regarding wire separation. In particular, it is unknown right now how many devices can be located, and how much logic can be computed, in a given clocking zone. The former is a potential concern, but [103] has shown that wire segments consisting of thousands of cells in length can operate correctly at room temperature. The latter questions is likely to be the main driver for system constraints such as power, speed, and area. As an example of how clocking may drive these factors, consider just a single bit-slice of the adder circuit shown in Fig. 4.4 (i.e. the lower left portion). For this adder, if the crossovers are equivalent to logic gates for clocking purposes in MQCA, then the longest path is five gate levels; for ESQCA, this increases to 19 gate levels (8 per crossover). If each gate level requires a unique clocking zone, then 5 and 19 clocking zones are required. If a simple change is possible, such as not requiring inverters to be in their own clocking zone, then only 4 and 14 clocking zones are required for MQCA and ESQCA respectively. If each clocking zone is of a minimum width, this change could enable area reductions of 20% and 26% for MQCA and ESQCA. Determining the amount of logic that can be located in a clocking zone is left for future work.
5.2.5 Power

Previous studies on QCA technology, both in ESQCA and MQCA, have shown that the power dissipated by the QCA devices is projected to be extremely low. Considering just the magnets in MQCA, the power dissipation has been estimated at 0.1 W for an array of of $10^{10}$ nanomagnets adiabatically switched $10^8$ times per second [19, 72]. For ESQCA, approximately $10^{12}$ devices in a square centimeter switching simultaneously would require approximately 100 W [92]. As a result, most of the power dissipated by a QCA circuit is likely to be a result of the clocking structure. Replicating a module is likely to have a minimal impact on power since the same clocking zones used in the original module can be used for the replicated modules as well. The main change in the number of clocking zones needed will occur in the input and output routing for the replicated modules; as shown above, this factor will be dictated by the amount of logic that can be located in a clocking zone.

5.2.6 Speed

In QCA systems, the clock rate is limited by the switching speed of the devices used to implement the system. In ESQCA, if molecular devices are utilized, then the speed of the system can approach the THz scale [92]. For MQCA, slower speeds are required since the magnets cannot change state as quickly as molecular devices, but at a speed of 100 MHz, the energy of an MQCA and a low-power CMOS adder are similar – and there is the possibility for further power reductions (or speed gains) in MQCA [72]. While the MQCA clock is slow compared to standard CMOS speeds, recall that a set of four contiguous clocking zones is equivalent to one pipeline stage. In essence, this means that QCA systems are
inherently deeply pipelined [63]. From a speed perspective, NMR will increase the latency to the first answer out of the system, but the throughput of the system will not be changed when using NMR.

5.2.7 Area

Considering just the modules, NMR requires at least N times the area that the original module does. However, since QCA is a planar technology, there will be an increase in the area due to the signal routing as well. As was shown in Table 2.1, the number of wire crossings required for each NMR configuration tended to grow quickly as the number of input and output signals for each module increased. In the system designs used here, an n input module requires \( n^2 - n \) wire crossings into the modules and 1.5 times that many crossings into the voters for the simplest design: TMR-NR. The maximum number of crossovers (path length) that a signal must pass through is \( z = 2 \times (n - 1) \) for aligning the inputs into the TMR modules. From the modules to the voters, the maximum path length is \( z \) for TMR with non-replicated voters and \( z + 4 \) for TMR with replicated voters.

Since the crossover device proposed for MQCA is roughly the size of a majority gate, the increase in area for signal routing will probably be small compared to the amount of extra logic used in replicating the modules. However, in ESQCA, a crossover circuit requires twelve majority gates and three inverters and the longest path through this circuit passes through six majority gates and two inverters. For modules with large numbers of inputs and outputs, this could cause a significant increase in area – provided that the module itself does not require a large number of crossovers.
5.3 MQCA Wires

This section investigates how MQCA wires should be organized for reliability. Three potential organizations are investigated: a single wire segment, a TMR-NR based organization, and a TMR-Rep based organization. Since cascades of each of these organizations are considered, the term slice is used to represent a single segment of each organization and the term wire segment is used to describe a section of wire within a slice. For the TMR based organizations, three wire segments are contained within each slice. Figure 5.8 shows a single slice of each configuration.

In this section, the fault rates of the wires are compared. The fault rate is simply computed as $1 - R$ where $R$ is the reliability of the wire structure as calculated by the PTM method. To compare wire segments of different lengths, the wire segments are given a fault rate per unit length of wire. In this modeling, the PTM for a wire segment, $PTM_{WS}$, is used to directly compute $R_w$ (reliability of a wire segment) using the procedure described in Sec. 3.1. $PTM_{WS}$ is found using Eq. 5.3 for a wire segment of length $l$ and a PTM for a unit length, $PTM_{UL}$. This construction of $PTM_{WS}$ is also shown graphically in Fig. 5.9 with $P = 1 - 5p + 20p^2 - 40p^3 + 40p^4 - 16p^5$ and $Q = 5p - 20p^2 + 40p^3 - 40p^4 + 16p^5$.

$$PTM_{WS} = PTM_{UL}^l$$ (5.3)

In addition to having fault-prone wire segments, the components needed to implement the TMR based organizations are also fault-prone. Specifically, there are fault-prone crossovers, fanouts, and voters. Rather than do a complete cell by cell layout for the wire segments between these components, it is assumed that the fault rates for these components include any potential faults in the wire segments.
(a) Straight Wire Slice.

(b) Single Voter Slice (TMR-NR).

(c) Replicated Voters Slice (TMR-Rep). Each wire crossing component is marked with an Xover box.

Figure 5.8. Wire Slice Models

Figure 5.9. Construction of $PTM_{WS}$ from five unit lengths.
used to connect these components to one another. This is a slight simplification from the methodology used previously which explicitly included these additional wire segments. Additionally, it is assumed that the clocking structure does not introduce any faults and the wire lengths are in increments of five unit lengths.

This section will first examine the fault rates for a single slice configuration. Then, multiple-slice wires will be considered in the next subsection when 1) there are a fixed number of slices and 2) when the total wire length is fixed. The third subsection will highlight how, for the TMR based organizations, the fault rate of the wire segment (equivalent to the module of the previous section) has almost no impact on the fault rate for the wire. Finally, a brief summary that can be used as a guide for determining the optimal wire organization will be presented.

5.3.1 Single Slice Fault Rates

In this section, the fault rate of single slices will be computed given a fixed fault rate of $1 \times 10^{-8}$ per unit length of wire and multiple fault rates for the other components. Fault rates per unit length other than $1 \times 10^{-8}$ have been tried, but the relationship between the fault rate per unit length and the fault rate for the other components is consistent as the fault rate per unit length varies. As a result, only the one fault rate per unit length is used.

Figure 5.10(a) presents the main result for this section. In this figure, Single Wire refers to a single wire with no replication, NR, $X$ refers to a triplicated wire with a single voter, and Rep, $X$ refers to a triplicated wire with triplicated voters. For both triplicated wire configurations, $X$ is the fault rate for the interconnect and voter components needed to implement TMR.

There are several trends in Fig. 5.10(a) that can be observed. First, when $X$ is
the same for the TMR based configurations, the TMR-NR configuration has the lower fault rate. Second, as the segment length increases, the fault rates for the TMR based configurations have an indistinguishable increase while the fault rate for the single wire configuration has a noticeable increase. This indistinguishable increase is a result of the fault rate for the TMR based configurations being significantly more dependent upon the fault rate of the TMR-needed interconnect and voting components rather than the fault rate of the module (single wire segment). This result was seen in the regression analysis of Sec. 5.2.3 and will be shown in more depth in Sec. 5.3.3.

Before discussing the results of Fig. 5.10(a) in depth, it should be noted that for TMR-Rep, there are three potential ways to compute the final fault rate of a single or multi-slice wire. The single-slice fault rates using each of these methods is shown in Fig. 5.10(b). The first two generate a single output from this wire by routing the outputs of the replicated voters to either a perfect voter (PMAJ) or a non-perfect voter (Match) that has the same fault rate as the other voters. The overlap between the Match results and the non-replicated voters results is due to the fact that in these methods, the reliability of the final majority vote puts an upper bound on the system reliability. The third method, used to generate the results in Fig. 5.10(a), assumes that the segment has three outputs with each one being the output of a voter (Wide). This last method is the most practical for building chains of slices using TMR-Rep, but it also generates the most pessimistic results for the single slice case as shown in Fig. 5.10(b). This pessimism results from not being able to “vote out” a fault on a single wire which would be done if the outputs from the replicated voters were routed through another majority gate.
Throughout the remainder of this topic, the fault rate per unit length \((1 \times 10^{-8})\) is labeled \(n\) and the fault rate for the components needed to implement TMR is \(X\). For Fig. 5.10(a), if \(X\) equals \(n\), then the replicated wire configurations have a lower fault rate. However, this scenario is improbable since this would mean that the fault rate for a multi-device component (i.e. majority gate) is equivalent to a single device. If \(X = 10 \times n = 1 \times 10^{-7}\), then the single wire slice configuration is better (lower fault rate) than TMR-NR until the segment length is between 15 and 20. Additionally, the single wire configuration is better than TMR-Rep until the segment length is between 35 and 40. If \(X = 100 \times n\), then the segment lengths at which the TMR-NR and TMR-Rep slice configurations are better than the single wire slice are 155-160 and 355-360 respectively.

In essence, as \(X\) increases by an order of magnitude in relation to \(n\), then the segment length needed for the TMR slice configurations to have a lower fault rate than the single wire slice configuration increases by approximately an order of magnitude. The approximate segment length needed for the TMR slice configurations to have a lower fault rate than the single wire slice is consistent as \(X\) and \(n\) vary by orders of magnitude from \(1 \times 10^{-11}\) to \(1 \times 10^{-4}\).
Figure 5.10. Fault rates for single segment wire slices as the number of unit lengths in the wire segment increases. The fault rate per unit length in the segment is $1 \times 10^{-8}$. 
5.3.2 Multi-Slice Wire Fault Rates

Having determined how a single wire slice should be configured to ensure reliable operation, the next exploration to undertake is how wires should be configured if multiple slices are chained together. Two main scenarios will be considered: 1) some number of slices are chained together and 2) a total fixed wire length is divided into a variable number of slices. Recall that for the single wire configuration, the slices are directly connected to one another; in the TMR-NR configuration, the output of the voter in each slice has to be fanned out to the triplicated wires of the next slice; and that in the TMR-Rep configuration, the output of the triplicated voters are directly connected to the triplicated wires of the next slice.

5.3.2.1 Fixed Number of Slices

Figure 5.11 shows the system fault rates for chains of 5 and 50 slices. The notation used in this section is the same as in the the previous section. For the TMR-Rep slice type, the Match method is used to generate a single output bit at the end. As before, the fault per unit length in the wire segment is $1 \times 10^{-8}$ and similar trends are seen as the fault rate per unit length varies by orders of magnitude.

One result, seen in both graphs of Fig. 5.11, is that TMR-Rep has a lower fault rate than TMR-NR for a given value of $X$. This differs from the single-slice results where the non-replicated voters method was better for a given value of $X$. For the 5 slice chain, if $X = 10 \times n$ than TMR-Rep is the most reliable. TMR-NR is an improvement on the single wire after the segment length is beyond 15-20. When $X = 100 \times n$, then TMR-Rep and TMR-NR are better than the single wire once the segment length is 30-35 and 155-160 respectively. Increasing $X$ by another order of
Figure 5.11. Fault rates for chains of wire slices with a fault rate per unit length of $1 \times 10^{-8}$. 

129
magnitude adds roughly another order of magnitude in the segment length needed for either TMR version to be an improvement on the single wire.

For the 50 slice chain, the relationship between \( X \), \( n \) and the segment length when TMR-NR is better than the single wire remains the same as they are for the 5 slice chain. However, the TMR-Rep shows a significant difference; in fact, adding additional slices to the chain has no significant impact on the fault rate as can be seen by comparing the TMR-Rep fault rates (assuming equivalent values of \( X \)) between Figs. 5.11(a) and 5.11(b). As a result, instead of having the single wire configuration be better than TMR-Rep for segment lengths of 30-35 at \( X = 100 \times n \), \( X \) can increase by another order of magnitude to reach the same point. These results imply, that for chains utilizing a large number of slices, TMR-Rep may be strongly favored.

5.3.2.2 Fixed Total Wire Length

Having considered single slice wires and wires of a fixed number of slices, wires with a fixed total length that are divided into a variable number of slices are examined. In Fig. 5.12 the system fault rates for wire lengths of 600 and 1200 units are presented using the same notation as before. Assuming that a unit length is a single nanomagnet in today’s technology, a non-replicated wire of this length would then be approximately 45\( \mu \)m and 90\( \mu \)m long if the nanomagnets have a pitch of 75nm ([72] assumes 60nm wide nanomagnets with 15nm between them). Since the TMR slice configurations add extra components, these wires would be longer.

As has been identified previously, there are ranges of segment lengths and TMR-required interconnect/voting component fault rates where both TMR slice
configurations can offer a lower fault rate than a single wire. A clear trend in these results, also noted in the previous subsection, is that as the number of slices is reduced, the fault rate for TMR-NR improves. While the fault rate per unit length and the fault rate of the components needed to implement TMR dominate which wire organization should be favored, these results show that factors like total wire length and segment length will also play a role in deciding which wire organization is best.

It is a near certainty that wires of the length considered here will need to be clocked across multiple clocking zones. Since this work utilizes a fault-rate per unit length of wire, these results are not dependent on the maximum length of wire that can be in a clocking zone. This can be demonstrated using the single-wire slice configuration for the shorter wire considered in this section. For example, if a maximum of 20 unit lengths can be in a clocking zone, the wire under consideration will require 30 clocking zones. The final PTM for 30 slices chained together with 20 unit lengths per slice is equivalent to the final PTM for 600 unit lengths as shown in Eq. 5.4 ($PTM_{UL}$ is the PTM for a unit length).

\[(PTM_{UL}^{20})^{30} = PTM_{UL}^{600}\]  \hspace{1cm} (5.4)
Figure 5.12. Fault rates for wires of a fixed length given a fault rate per unit length of $1 \times 10^{-8}$. Note that the x-axis of these graphs is not to scale.
5.3.3 Importance of Interconnect

Figure 5.13. Fault rates for wires of a fixed length given a fixed interconnect and voting component fault rate of $1 \times 10^{-4}$. The number in the legend entries is the fault rate per unit length.

In Sections 5.2.2 and 5.2.3 it was observed that, when using TMR configurations, it is the interconnect and voters that have the largest impact on reliability. A similar trend was identified in Sec. 5.3.1 and is more clearly shown in Fig. 5.13. In this figure, the wire length is fixed at 1200 and the interconnect and voting component fault rate is $1 \times 10^{-4}$. The system fault rate is then calculated for fault rates per unit length of $1 \times 10^{-5}$, $1 \times 10^{-8}$, and $1 \times 10^{-11}$. These fault rates are selected to show how reducing the fault rate per unit length by several orders of
magnitude has a minimal impact on the fault rate for the entire wire.

The single wire case where the fault rate per unit length is $1 \times 10^{-11}$ is not shown to highlight the other results; the fault rate for this case is $1.2 \times 10^{-8}$. Additionally, each of the TMR-NR lines overlap, as do the $Rep, 1e-8$ and $Rep, 1e-11$ lines. This demonstrates again that for both TMR slice configurations, improving the fault rate per unit length in the wire segment by several orders of magnitude will have little improvement on the wire fault rate. This result is expected, particularly for the TMR-NR case since the voting logic bounds the fault rate for a slice (note the location of $R_v$ in Eq. 2.3).
5.3.4 Guidelines for Organizing an MQCA Wire

To determine how wires should be organized for reliability, the fault rates for several wire organizations have been calculated. These fault rates are based on a fault rate per unit length in a wire and, for the TMR based approaches, the fault rates for the interconnect and voting components that are required to implement the TMR method have been varied as well. These results demonstrate that the relationship between the fault rate per unit length and the fault rate for the components needed to implement TMR (respectively, \( n \) and \( X \) as before) will be the primary factor in determining which wire configuration should be favored. Other factors, such as wire length and segment length will play a prominent, but lesser, role as well.

The salient points of this relationship are summarized below and provide a set of guidelines as to which wire organization is favored from a reliability perspective. For the lengths below both a multiple of the unit length and an actual length (assuming 75nm pitch between nanomagnets, as in Sec. 5.3.2.2).

- **Case 1, \( X \geq 1000 \times n \):** Unless the total wire length is extremely long, say over 1500 (\( \sim 113\mu m \)), the single wire configuration is favorable. If the wire has a length over 1500, then TMR-Rep is a better choice than TMR-NR if multiple slices are desired. If using just a single slice, then TMR-NR may be favorable. As discussed above, for both of these TMR configurations, the final voting gate places a bound on the maximum reliability of each technique.

- **Case 2, \( X \approx 100 \times n \):** For this case, the total wire length and number of slices will be the main driver for determining which configuration should be favored. For multi-slice wires or wires with a length over approximately 330
(~25µm), then TMR-Rep is likely the best choice. TMR-NR can also be practical if the slice length is over 150 (~11µm).

- Case 3, \(X \approx 10 \times n\): In this case, either version of TMR could be utilized and provide reliability improvements provided that the wire length is at least 15-20 (~11µm-15µm) for single slices using TMR-NR or at least 30-35 (~22µm-26µm) for single slices using TMR-Rep.

- Case 4, \(X \approx n\): This case was not studied in detail since it is unlikely to occur because the interconnect and voting components will likely require multiple unit lengths. However, as was shown in Fig. 5.10, if this does occur, TMR should be utilized unless the segment length is below 5 (~400nm).

Having identified how wires should be organized from an analytical perspective, the physical models that quantify the relationship between the fault rate for the components needed to implement TMR and the fault rate per unit length of wire need to be developed. Once this relationship is quantified, an optimal wire organization for reliability can be determined.
CHAPTER 6

COMPONENT MODELING

The analytic modeling of the previous two chapters has identified how reliable components should be in order to generate a reliable circuit. This chapter investigates how reliable components can be. This is done by physically modeling ESQCA wire structures built from molecularly sized cells. Wires are selected for this study since they are a frequently used component and the regression analysis of Sec. 4.7 found them to be the second most critical component in ESQCA circuits.

Recall from Sec. 2.3.2 that ESQCA components, particularly wires, can be made more robust by increasing the thickness (width) of the wire segments. Fig. 2.12 is reshown here in Fig. 6.1 as an example of a normal wire and a thick wire. For the results presented in this chapter, wire segments of multiple widths are tested.

The first section describes how missing cells can impact memory like segments. This study was completed using the geometric analysis method and program described in Sec. 3.3. The second and third sections examine the reliability of straight ESQCA wires in the presence of rotation, displacement, and missing cell defects. These sections utilize the three-state coherence vector simulation method described in Sec. 3.4.
6.1 Memory-like Wire Structures

In this section, memory-like ESQCA wire structures consisting of multiple wire segments and corner sections are investigated for their tolerance to missing cells. Both the length and the aspect ratio of the wire segments are varied to determine how these factors impact the potential yield (initial reliability) when building structures of this type. The first section describes the memory-like structure tested and the second section describes the results from the geometric analysis method.

6.1.1 Memory Structures

Figure 6.2 shows the multi-bend wire structure of [36] that can be used as a large shift register or memory storage element (if given more support) used in this section. Each structure is a chain of wire segments where each segment consists of 30 cells and two corners. The 30 cells are divided into a width and height, and varying these parameters changes the aspect ratio of the segment.
The chains tested here consist of two, four, and eight segments for one, two, and three cell wide memory structures. Additionally, aspect ratios of 9:1, 5:1, 2:1, 1:1 which evenly divide the 30 cells in each segment are tested. Example segments are shown in Figs. 6.2(a) and 6.2(b) and a chain of two segments is shown in Fig. 6.2(c). Varying the aspect ratio can provide insight as to how to organize multi-bend wires in an ESQCA circuit.

6.1.2 Results

A total of thirty-six different memory structures are tested. Figures 6.3 and 6.4 shows their yield as the percentage of cells removed is varied. The results have been divided into subgraphs based on aspect ratio. Each data point shown in Figs. 6.3 and 6.4 represents 1000 iterations of that specific configuration. For these results, a memory structure was considered failed if a failure pattern was observed. In Fig. 6.4(b) the line labeled “CV” shows the results from 500 simulations using the two-state coherence vector simulation (see Sec. 3.5.1.1 and [2]) at three dropout rates. This line is shown as to demonstrate the validity of the geometric analysis approach.

Clearly, the aspect ratio of the memory structure has no significant impact on the yield. This suggests that optimizing wire structures (interconnect or memory) by aspect ratio is unnecessary. These results also demonstrate that the longer the structure, the more likely it is to fail. This is to be expected since more single failure locations (corners) exist in the larger structures.

These graphs also show that, in general, the one cell wide structures have the highest yield followed by the three cell wide then the two cell wide structures. This result has been observed before in [31] and is caused by the fact that the
Figure 6.2: Memory structure segments and sample.
Figure 6.3. Yield estimates for various memory structures by aspect ratio.
Figure 6.4. Yield estimates for various memory structures by aspect ratio.
one cell wide structures generally only invert signals if a cell dropout occurs in a corner region. Single or double cell dropouts in straight sections do not cause inversion failures like double (or triple) cell dropouts do in thicker wire sections. However, the one cell wide wires are generally more susceptible to failure when other defect types (i.e. stray charges) are considered.

These yield estimates are likely pessimistic since the case of multiple failures existing in the structure is not considered. This is the most likely reason why the 'CV' results in Fig. 6.4(b) are higher than the results using our method. If an even number of failure patterns (all inverting) exist, the structure may operate properly. This behavior needs to be accounted for as the geometric analysis tool is matured.

6.2 Straight ESQCA Wire Segment Study Setup

This section describes the experimental setup and the wire segment structures that are being studied with the three-state coherence vector simulator. For this study, the tested wire segments vary in length from 25-100 cells and in width from 1-5 cells. Each of the wire segments will have some additional non-faulty cells near the input and output to ensure that the wire segment has the proper input value and that the output value converges. The notation \((x,y)\) is used to describe wire segments that are \(x\) cells long and \(y\) cells wide. The cell shown in Fig. 6.5 is used for this work with \(a = 1\)nm. For reference, the cell-center to cell-center distance is 2nm in both the horizontal and vertical directions.

Liu completed a study investigating the impact of rotations and disorder defects on (500,1) and (500,3) wires using the cell shown in Fig. 6.5 [61]. In this study, each cell in the wire was displaced in the horizontal and vertical directions.
while also being rotated. The displacement defects for a cell were in the range $[-\delta, \delta]$ while the rotation defects for a cell were in the range $[-\theta, \theta]$ with $\delta$ and $\theta$ being stepped from 0-0.35nm and 0-30 degrees respectively.

The results presented in [61] demonstrated that a one cell wide wire operated correctly when $\theta \leq 21^\circ$ no displacements are present. The same wire could tolerate displacements of $\delta \leq 0.35nm$ when no rotations were present. The three cell wide wire could tolerate a maximum rotation of approximately $5^\circ$ more than the one cell wide wire could when $\delta \leq 0.3nm$

The simulator developed for this dissertation has been validated by approximately reproducing the (500,1) results from [61].

6.2.1 Defect Ranges

For this study, the following defects are tested: displacements, rotations, and missing cells. The magnitude of a cell displacement, both horizontally and vertically, ranges from 0 to $0.30 \times a$. The range for a cell rotation is -30 to 30 degrees. The number of cells that could be dropped from any given wire segment ranges from 0 to 5%. In the simulator, the displacement and rotation defects are considered maximums whereas the percentage of missing cells is closer to the real value.
The procedure for injecting these defects into the wire segment is described below.

When displacement and rotation defects are introduced in the wire segments, each cell is given the prescribed defects of horizontal displacement, vertical displacement, and rotation based on two random values (unique to each defect type and cell): a sign value and a ratio value. The sign value determines which direction the defect is injected and the ratio value, in conjunction with the defect magnitude determines how the cell is manipulated. The sign value is designed to be +1 half the time, -1 the other half of the time. The ratio value ranges from [0,1] and is then multiplied by the maximum displacement distance or rotation angle. For example, if the maximum horizontal displacement $0.15 \times a$ the sign value is -1, and the ratio value is 0.65, the selected cell is then moved $0.65 \times 0.15 = 0.0975\text{nm}$ in the -x (to the left of the original location) direction.

For the missing cells, any non-input/fixed/output cell is eligible to be removed. A random number generator is used to determine how many cells should be removed if the percentage of cells to be removed is not a whole number (e.g. dropping 5% of 75 cells is 3.75). The remaining fractional part (e.g. 0.75) determines the percentage of the time that the number of cells to be dropped will be rounded up. For this case, there is a 75% chance that 4 cells will be dropped and a 25% chance that 3 cells will be dropped. Running multiple instances of a test with the same percentage of dropped cells would then create an average number of dropped cells per instance equal to the calculated number (3.75 in this case).

6.2.2 Simulation Parameters

An overview of the three state coherence vector method was presented in Sec. 3.4. The values for the simulation parameters shown below were used to
generate the results in [93, 94]. These values were selected to make the behavior of a wire readily apparent. The clock will be discussed after the list.

- Neighbor radius = 10nm. This value is used to determine the local neighborhood for a cell. Cells outside of this neighborhood do not influence the cell.

- $\tau = 10 \times \hbar/E_k$. $E_k$ is the ideal kink energy between two non-defective cells; for the cell used here $E_k$ is approximately 0.3 eV.

- $\gamma = 0.02\text{eV}$. This is approximately $0.15 \times E_k$.

- $T = 300\text{Kelvin}$.

- $T_c = 5\text{ps}$. This is the clock period.

The clock signal used in this simulator is not a direct sinusoidal wave as has been done in [61]. Instead, the clock wave used here is similar to any of the clock signals in Fig. 2.4(b) with each of the four parts (high, low, rising edge, falling edge) being 100nm long in the horizontal direction (this is roughly equivalent to a clocking zone being 100nm wide). Each cell in the wire segment will see this signal in its entirety only one time. The clock is initially low and at time $t$ the start of the rising edge of the clock will be $t/T_c \times z$ where $z$ is the horizontal range of the circuit with some buffering (extra space). The high and low clock signals are at $E_k$ and $-E_k$ respectively.

It should also be noted that each dot in a cell contains a charge of $+e/3$. This is done to evenly distribute the charge necessary to balance out the two free electrons, thus leaving the cell electronically neutral. Without these neutralizing charges, the simulator is unstable and tends to generate inaccurate results.
6.3 Straight ESQCA Wire Segment Study Results

For the results in this section, one hundred tests were run for a range of combinations that varied wire length, wire width, maximum displacement defect, maximum angle of rotation, and percentage of missing cells. Additionally, the maximum displacement defect is stepped by values of \(0.1 \cdot a\) (\(a = 1\text{nm}\)) and the maximum angle of rotation is stepped by values of two degrees. Since there are an infinite number of possible circuit arrangements at each configuration due to the randomness in how defects are injected, clearly only a small number of possible arrangements are tested. This limited testing accounts for much of the line roughness in the presented results.

For the purposes of this section, a passing wire is one that properly passes both a binary 0 and a binary 1 correctly. If either value does not move from the input to the output correctly, the wire is considered to have failed.

6.3.1 90% Pass Rates

Figures 6.6 through 6.9 identify the maximum displacement and angle of rotation for a wire (width marked in legend) to have a passing rate of 90%. The subgraphs differ in the percentage of missing cells. For the graphs in these figures, having a larger maximum angle of rotation means that the wire segment can tolerate a larger defect range (i.e. is more robust).

Within each of these figures, a specific trend is observed for all of the widths except for 2: as the maximum displacement and percentage of missing cells increase the maximum angle of rotation tends to decrease. In some cases, a larger angle can be tolerated when the maximum displacement increases, see the Wire 3 line in Fig. 6.6(b). This is due to the number of passing tests being 88 for a
displacement of 0.2 and angle of 20 degrees and the number of passes being 90 when the angle is the same, but the displacement is 0.3. More tests for each configuration would reduce this effect.

(a) No dropped cells.  
(b) 3% of cells dropped.  
(c) 5% of cells dropped.

Figure 6.6. Maximum displacement and rotation defects when at least 90 passing tests were observed for wires of length 25 with various percentages of missing cells. The larger the maximum angle of rotation, the more robust the wire segment.
Figure 6.7. Maximum displacement and rotation defects when at least 90 passing tests were observed for wires of length 50 with various percentages of missing cells. The larger the maximum angle of rotation, the more robust the wire segment.

An additional trend seen when comparing Figs. 6.6 through 6.9 is that increasing the wire length also decreases the maximum defect levels that can occur to have a passing rate of 90% for the wire segments. This is an expected result since longer wires offer more locations for the wire segment to fail.

In general, we also observe that increasing the thickness of the wire tends to increase the number of passing tests. However, a wire with a thickness of two tends
Figure 6.8. Maximum displacement and rotation defects when at least 90 passing tests were observed for wires of length 75 with various percentages of missing cells. The larger the maximum angle of rotation, the more robust the wire segment.

to create rather interesting behavior, particularly when 5% of the cells are missing. For small displacement ranges, the missing cells dominate and many of these wire segments act like inverters. This type of behavior was described in Sec. 3.3. However, when the maximum displacement range increases, the robustness of these wires is improved. This result is unexpected and needs to be studied in more depth. We estimate this behavior to be a result of either the large cell...
Figure 6.9. Maximum displacement and rotation defects when at least 90 passing tests were observed for wires of length 100 with various percentages of missing cells. The larger the maximum angle of rotation, the more robust the wire segment.

displacements preventing the formation of inverting structures within the wire or that an even number of inverting structures are formed within the wire.
6.3.2 Total Passing Tests

Figures 6.10 and 6.11 show the number of passing tests for the (25,3) and (100,3) wire as the maximum displacement, angle of rotation, and percentage of missing cells varies. If the number of passing tests was 100 for a given maximum angle of rotation and fixed displacement, it is assumed that all smaller maximum angles of rotation have 100 passing tests as well. A common trend in these results is that, for a fixed maximum displacement and percentage of missing cells, a slight change in the maximum angle of rotation can have a significant negative impact on the ability of the wire to transmit the proper value. As before, increasing the number of missing cells reduces the robustness of the wire as well.

An interesting trend occurring in the three wide wires is that the number of passing tests tends to reduce more slowly as the displacement magnitude increases. As a result, almost half of the wires pass the correct value when the maximum displacement is 0.3 and the maximum angle of rotation is greater than 25. When only rotational defects occur, there are very few passing wires when the maximum angle of rotation is greater than 25. This behavior is seen in the other wires as well, such as for the (100,5) wire, as shown in Fig. 6.12. While the exact cause for this is not yet known, it is likely to be the result of the large displacements either preventing the formation of inverting structures or that an even multiple of inverting structures exist within the wire.
Figure 6.10. Number of passing tests for a range of displacement and rotation defect magnitudes as the percentage of dropped cells changes in a (25,3) wire. Note: the range of the x-axis changes in each subfigure.
Figure 6.11. Number of passing tests for a range of displacement and rotation defect magnitudes as the percentage of dropped cells changes in a (100,3) wire. Note: the range of the x-axis changes in each subfigure.
Figure 6.12. Number of passing tests for a range of displacement and rotation defect magnitudes as the percentage of dropped cells changes in a (100,5) wire. Note: the range of the x-axis changes in each subfigure.
6.3.3 Comparing Wire Widths

Figures 6.13 and 6.14 show the number of passing tests as the maximum angle of rotation changes for wires that are 25 and 100 cells long (respectively) with maximum displacements (labeled \( MD \)) of 0 and 0.3. As before, the subgraphs show the results as the percentage of missing cells changes.

The results in these graphs demonstrate that, in general, a thicker wire is more robust than a thinner wire. As in the previous results, only having rotational defects enables a higher percentage of passing tests up to a given angle. Once this angle is reached, the number of passing tests plummets far more quickly than in the case when both rotational and displacement defects exist.
Figure 6.13. Number of passing tests for a range of rotation defect magnitudes for given fixed maximum displacements (labeled $MD$) as the percentage of dropped cells changes in (25, 1), (25, 3), and (25, 5) wires.
Figure 6.14. Number of passing tests for a range of rotation defect magnitudes for given fixed maximum displacements (labeled $MD$) as the percentage of dropped cells changes in (100, 1), (100, 3), and (100, 5) wires.
6.3.4 Comparing Wire Lengths

Figure 6.15 shows the number of passing tests for wires that are three cells wide as the maximum angle of rotation and wire length changes with maximum displacements (labeled $MD$) of 0 and 0.3. As before, the subgraphs show the results as the percentage of missing cells changes.

For these results, we observe that when only rotational defects are in the wire, there is a noticeable difference in the number of passing tests as the wire length increases. This result is to be expected because a longer wire offers more opportunities for an inverting structure to exist. However, when both rotational and displacement defects exist within the wire, the number of passing tests is essentially independent from wire length. This is an unusual result without any clear indication of why it occurs.

6.3.5 Summary of Results

Several conclusions can be drawn from the results presented above. First and foremost, a wire can have a wide range of manufacturing defects and still be functioning properly. This is likely to be beneficial in the manufacturing process since perfection will not be required for wires. For modest levels of defects, such as maximum rotations under 10 degrees, displacements under 0.1nm, and under 3% of the cells missing, almost any wire structure should be safe to use in shorter wire segments. However, should the level of defects be higher, building the shortest wire possible will be advantageous in most cases. Additionally, for higher defect ranges, thicker wires are a practical method for increasing robustness in a wire segment, provided that a width of two is NOT used if missing cells are possible.
Figure 6.15. Number of passing tests for a range of rotation defect magnitudes for a given fixed maximum displacement (labeled $MD$) as the percentage of dropped cells changes in all four lengths of a three wide wire.

6.3.6 Computation Time

In order to generate the results for this section, a total of 228,000 simulations were run. The total CPU time used to run these simulations was approximately 73,000 hours or 8.3 years. This represents an average of approximately 20 minutes per simulation. The smallest simulations, the (25,1) wires with no cells missing, took approximately 90 seconds per simulation while the largest simulations, the
(100,5) wires with no cells missing took approximately 63 minutes per simulation. These simulations were run using the Condor distributed system [1] which allowed for up to about 200 simulations to be run at a time.
CHAPTER 7

CONCLUSIONS AND FUTURE WORK

This dissertation has been designed to answer one of the questions posed by [8]: “Can we produce reliable, predictable systems from unreliable components with unpredictable behavior?” From the work contained herein, it certainly appears that the answer to this question is yes for QCA provided that reliable components can be built (for example, component error rates $\lesssim 1 \times 10^{-4}$). Without reliable components, there does not appear to be a path to a practical, reliable system because both the logic and interconnect will be faulty.

In order to build a reliable, predictable system, it will be necessary to approach reliability at the device, component, circuit, and system levels. The sections below highlight the takeaways from this work that could be of use to individuals working on the basic devices and components and to the circuit and system designers. After presenting these takeaways, the conclusions and insights presented in this work are summarized and a variety of future research directions are offered.

7.1 Takeaways for Device and Component Developers

Several results have been presented in this work that could be of use to individuals working on building devices and components in QCA. The first group of takeaways are developed from the analytic modeling results of Chapter 4 while the second group is developed from the ESQCA physical modeling results.
First, circuit reliabilities based on the reliability of the underlying components have been presented. These results are practical as manufacturing capabilities scale up because they provide a guideline as to how many samples of a circuit may need to be produced to ensure that several of them work. Secondly, it has been shown that no component should have a reliability much lower than the other components (i.e. no component is a weak link) because of the impact on circuit reliability. Third, the regression analysis method identified the impact each component had on reliability; assuming that there are no weak links, the most critical components should be targeted for reliability improvements first.

The following takeaways are strictly geared towards improving the robustness of ESQCA components. First, the potential for cells to rotate should be minimized. The results presented in this work strongly suggest that ESQCA components are more sensitive to cell rotations than cell displacements or missing cells. Second, a missing cell rate of under 3% should be targeted. Obviously, lower is better, but this rate may be a practical upper bound. Third, if building wires with sharp corners, the presence of defects in the inner corner region should be minimized. Lastly, it is clear that device level redundancy (thick components) can improve reliability and should be used.

Improving the reliability of MQCA devices and components is an open question and is left for future work.

7.2 Takeaways for Circuit and System Designers

In this section, the design of a large circuit, such as a 64-bit multiplier, for reliability is explored. This discussion is divided into ESQCA and MQCA design styles based on the wire crossing requirements of each. In both cases, it is ex-
pected that robust versions of the components will be utilized. Additionally, it is assumed that the designer will attempt to minimize factors like the number of total components wire, length of each wire, and the number of wire crossings required. Minimizing the number of wire crossings is far more important for ESQCA than MQCA, but should be done for both technologies.

Due to the impact signal routing has on reliability in both QCA technologies, along with the overall size of the multiplier, neither NMR (with the full multiplier being the copied module) nor configurable logic (entire multiplier on a PLA) is likely to improve reliability. Further guidelines, based on improving reliability in the sub-circuits of the multiplier to improve the entire multiplier’s reliability, are presented below.

In ESQCA, our results suggest that using NMR with almost any size module is likely to harm reliability, thus there is likely no value in using NMR at any scale within the multiplier. Our results also suggest (pending further study) that configurable logic is unlikely to be practical for ESQCA due to the increased amount of signal routing that would be required. Overall, these results suggest that a large ESQCA circuit designs should consist of custom logic using the most robust components available and following the assumptions listed above (i.e. minimizing the number of crossings).

In MQCA, our results demonstrate that NMR can be used to improve the reliability of small modules. This suggests that there should be a reliability benefit to using NMR on some of the smaller sub-circuits within the multiplier (e.g. use NMR on the adder portions of the multiplier). Larger sub-circuits, in terms of the number of inputs/outputs, are not likely to benefit from NMR due to the signal routing complexity, thus NMR should not be used with them. Given the amount
of signal routing necessary to implement a configurable system, it is unlikely that a small PLA within the multiplier logic would provide any benefit to reliability. A specific mechanism for improving the reliability of the multiplier would be to use the guidelines presented in Sec. 5.3.4 for long wires in the multiplier. Overall, the design of the multiplier will use custom logic, but there are likely to be some low-level circuits and long wires that can be replicated in order to improve reliability.

7.3 Contributions

The following contributions have been presented in this dissertation.

- An analytic modeling method from the literature is utilized to compute QCA circuit and system reliabilities when both logic and interconnect components are faulty.

- A method for identifying the critical components of a QCA circuit is presented. This method shows that wires and majority gates are the most critical components in QCA circuits.

- Preliminary results suggest that custom logic circuits may be preferable to PLA based circuits from the reliability perspective.

- NMR is shown to improve the reliability of MQCA systems but is unlikely to improve the reliability of ESQCA systems.

- Guidelines for designing highly reliable long MQCA wires are provided.

- A new modeling method for quickly estimating the reliability of an ESQCA multi-bend wire is used to show that the reliability of the wire is independent of the wire segment length between bends.
• Physical simulation is used to identify the robustness of various ESQCA wire segments in the presence of multiple defect types.

7.4 Insights

The following list identifies several broad conclusions that can be drawn from this work.

• Mathematical (analytic) modeling of QCA circuits is feasible and methods for identifying critical components have been developed. This type of modeling should be applicable to other nanoelectronic devices.

• The use of system level reliability techniques, such as space redundancy and regular logic, may have a negative effect on reliability. As such, new methods or blends of current techniques may be required depending on the technology utilized.

• The study of nanoelectronic circuits can be done with various abstractions. In particular, applying the lessons learned from ULSI/VLSI circuit design and development can prevent the need to reinvent the wheel.

• As the circuit or system size increases, the impact of small changes in the reliability of the basic components has an increasing effect on the reliability of the circuit or system.

• ESQCA wires can operate properly in the presence of manufacturing defects and it is possible to make them more robust by increasing the width of the wire. Additionally, bent ESQCA wires are very sensitive to defects in the inner corner region.
• The major insight that should be taken away from this work is that signal routing is THE driver for QCA circuit reliability.

7.5 Future Work

There are several areas for future work that could be pursued as a result of this dissertation. One of the key areas of future work will be to physically model ESQCA and MQCA components to determine the error rates for these components. This is necessary to refine the analytic modeling approach which frequently assumed a uniform error rate for the components. Additionally, the wire components were assumed to have the same error rate which was independent of length. As was shown in Chapter 6, this assumption is unlikely to hold.

Refining the analytic modeling will require having a more detailed layout of the circuits. This presents a significant challenge since factors like wire length will be heavily dependent upon how much computation can be completed in a clocking zone and how the clocking zones can be arranged and sized. These questions present an avenue for future research that needs to be pursued sooner rather than later.

Another avenue for future work that was discussed in Sec. 5.1.3 is that the comparison between PLA and custom logic reliabilities needs to account for swapping and sparing within the PLA. This is in addition to considering other benchmark circuits.

Models for computing circuit and system reliabilities from a set of equations, such as the classical TMR modeling equations (see Sec. 2.2.2.2) could be developed. This would allow for a much quicker comparison between various circuit and system designs and could also provide greater insight into how reliability changes
as module sizes scale.

The geometric analysis technique presented in Sec. 3.3 could easily be extended to account for other defect types and components as well. This would probably require some form of data mining on the results from Sec. 6.3 to determine when failures occur to develop the fault library. While this approach has only been developed for ESQCA at this point, there does not appear to be any substantial roadblocks that would prevent a similar tool from being created for MQCA circuits.
BIBLIOGRAPHY


95. P. D. Tougaw. Personal communication, 2005.


