UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
PROGRAMA DE PÓS-GRADUAÇÃO EM MICROELETRÔNICA

QUATERNARY CLB FOR A FAULT TOLERANT QUATERNARY FPGA

Eduardo Luis Rhod

Monografia elaborada para defesa de doutorado sob a orientação do professor Luigi Carro, em preenchimento parcial aos requisitos necessários para a obtenção do grau de Doutor pelo Pograma de Ps-Gaduação em Microeletrônica.

Porto Alegre

2012

CIP - Catalogação na Publicação

Rhod, Eduardo
QUATERNARY CLB FOR A FAULT TOLERANT QUATERNARY FPGA / Eduardo Rhod. -- 2012.
90 f.

Orientador: Luigi Carro.

Tese (Doutorado) -- Universidade Federal do Rio Grande do Sul, Instituto de Informática, Programa de Pós-Graduação em Microeletrônica, Porto Alegre, BR-RS, 2012.

1. Tolerância a Falhas. 2. Lógica Quaternária. 3. FPGA. 4. CLB. I. Carro, Luigi, orient. II. Título.

Elaborada pelo Sistema de Geração Automática de Ficha Catalográfica da UFRGS com os dados fornecidos pelo(a) autor(a).

DEDICATÓRIA

Dedico este trabalho a todos os meus familiares, aos que estão entre nós e também aos que já se foram. Em especial, dedico aos meus pais, que desde cedo me incentivaram e mostraram o valor transformador da educação, pois nunca mediram esforços para me garantir uma educação de qualidade. Principalmente dedico a eles por me educarem através da melhor técnica conhecida pela humanidade, a educação pelo exemplo de bom caráter e valores humanos que sempre demonstraram e continuam demonstrando até hoje.

Dedico também a minha mulher e companheira de todos os momentos, e aos seus familiares que sempre pude considerar como a minha segunda família.

AGRADECIMENTOS

Ao Programa de Pós-Graduação em Microeletrônica, PGMicro, e a todos os funcionários da secretaria do curso Pós-Graduação em Microeletrônica, pelo profissionalismo e seriedade com que exerceram suas atividades. Agradeço em especial às secretárias do PGMicro, Zíngara Lubaszewski e Gabriela Ouriques. Ao professor Henri Boudinov, pelos ensinamentos relacionados à física de semicondutores. Agradeço também a todos os professores e colegas de turma que contribuíram enormemente na minha formação durante o mestrado, professores Sergio Bampi, Marcelo Lubaszewski, Ricardo Reis e Fernanda Lima Kastensmidt. Em especial, agradeço ao professor e colega de pesquisa Carlos Arthur Lang Lisbôa. Agradeço também e em caráter decisivo para a qualidade da pesquisa desenvolvida ao longo do doutorado e principalmente pelo meu crescimento profissional aos colegas de laboratório Mateus Beck Rutzig, Gabriel Nazar, Ulisses Brisolara Corêa, Antonio C. S. B. Filho, Mônica Magalhães, Ronaldo Ferreira e Fabio Iturriet pelas enormes contribuições e ajudas fornecidas durante os trabalhos de pesquisa. Ao CNPQ e a Capes pela provisão da bolsa de mestrado, sem a qual não teria seguido com os estudos.

Agradeço em especial ao meu orientador Luigi Carro, por acreditar no meu trabalho e por conduzir exemplarmente a minha orientação, guiando-me na busca por um trabalho de qualidade, mas ao mesmo tempo dando espaço para que eu exercesse a minha pesquisa com criatividade e liberdade.

Agradeço a todos os meus familiares e amigos de Lajeado pelo apoio e compreensão da minha ausência nos diversos momentos em que estive dedicado aos meus estudos. Agradeço a minha esposa Alexandra Barcelos Rhod e seus familiares pelo companheirismo e amor. Em especial agradeço aos meus pais Pedro Valentin Rhod e Marta Regina Blasi Rhod pela educação e por todas as oportunidades que me deram sem nunca medir esforços. Agradeço os meus irmãos Guilherme Blasi Rhod e Amanda Rhod Rocha por fazerem parte na minha vida.

RESUMO

A diminuição no tamanho dos transistores vem aumentando cada vez mais o número de funções que os dispositivos eletrônicos podem realizar. Apesar da diminuição do tamanho mínimo dos transistores, a velocidade máxima dos circuitos não consegue seguir a mesma taxa de aumento. Um dos grandes culpados apontados pelos pesquisadores são as interconexões entre os transistores e também entre os componentes. O aumento no número de interconexões dos circuitos traz consigo um significativo aumento do cosumo de energia, aumento do atraso de propagação dos sinais, além de um aumento da complexidade e custo do projeto dos circuitos integrados. Como uma possível solução a este problema é proposta a utilização de lógica multivalorada, mais especificamente, a lógica quaternária. Os dispositivos FPGAs são caracterizados principalmente pela grande flexibilidade que oferecem aos projetistas de sistemas digitais. Entretanto, com o avanço nas tecnologias de fabricação de circuitos integrados e diminuição das dimensões de fabricação, os problemas relacionados ao grande número de interconexões são uma preocupação para as próximas tecnologias de FPGAs. As tecnologias menores que 90nm possuem um grande aumento na taxa de erros dos circuitos, na lógica combinacional e sequencial. Apesar de algumas potenciais soluções começara a ser investigadas pela comunidade, a busca por circuitos tolerantes a erros induzidos por radiação, sem penalidades no desempenho, área ou potência, ainda é um assunto de pesquisa em aberto. Este trabalho propõe o uso de circuitos quaternários com modificações para tolerar falhas provenientes de eventos transientes. Como principal contribuição deste trabalho destaca-se o desenvolvimento de uma CLB (do inglês *Configurable Logic Block*) quaternária capaz de suportar eventos transientes e, na possibilidade de um erro, evitá-lo ou corrigi-lo.

Palavras-chaves: Arquiteturas tolerantes a falhas, circuitos quaternários, técnicas de detecção de erros, taxa de *soft error, FPGAs*.

ABSTRACT

The decrease in transistor size is increasing the number of functions that can be performed by the electronic devices. Despite this reduction in the transistors minimum size, the circuit's speed does not follow the same rate. One of the major reasons pointed out by researchers are the interconnections between the transistors and between the components. The increase in the number of circuit interconnections brings a significant increase in energy consumption, propagation delay of signals, and an increase in the complexity and cost of new technologies IC designs. As a possible solution to this problem the use of multivalued logic is being proposed, more specifically, the quaternary logic. FPGA devices are characterized mainly by offering greater flexibility to designers of digital systems. However, with the advance in IC manufacturing technologies and the reduced size of the minimum fabricated dimensions, the problems related to the large number of interconnections are a concern for future technologies of FPGAs. The sub 90nm technologies have a large increase in the error rate of its functions for the combinational and sequential logic. Although potential solutions are being investigated by the community, the search for circuits tolerant to radiation induced errors, without performance, area, or power penalties, is still an open research issue. This work proposes the use of quaternary circuits with modifications to tolerate faults from transient events. The main contribution of this work is the development of a quaternary CLB (Configurable Logic Block) able to withstand transient events and the occurrence of soft errors.

Keywords: Fault tolerant architectures, quaternary circuits, error detection techniques, soft error rate, FPGAs

SUMMARY

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL 1
1 Introduction 14
2 Context of the Research 17
2.1 Quaternary Logic Background 17
2.1.1 Current-Mode Multivalued-Circuits 19
2.1.2 Voltage-Mode Multivalued Logic Circuits 21
2.1.3 Voltage-Mode Multivalued Logic Circuits Applied to FPGAs 23
2.2 Quaternary Base Circuits 24
2.3 Fault Tolerance Background 26
2.3.1 Radiation Sources and their Effects 27
2.3.2 Effects of SEUs and SETs in Digital Circuits 28
2.3.3 Metrics to Evaluate the Vulnerability of Circuits to Soft Errors 29
2.3.4 Mitigation Techniques for SEUs and SETs 32
3 Fault Tolerant Quaternary Circuits 38
3.1 Fault Tolerant Quaternary Lookup Table (FT-QLUT) and the Error Avoiding Quaternary Lookup Table (EA-QLUT) 38
3.1.1 QLUT Fragilities 39
3.1.2 Proposed Architecture 40
3.1.3 Fault Injection Experiments 42
3.1.4 FT-QLUT Improvement: The EA-QLUT 46
3.2 The Fault Tolerant Quaternary Latch and Flip-Flop (FT-QL and FT-QFF) 48
3.2.1 The Fault Tolerant Quaternary Latch (FT-QL) 52
3.2.2 The Fault Tolerant Quaternary Flip-Flop (FT-QFF) 55
3.3 Variability Impact Analysis 58
4 Area, Performance and power dissipation overheads 60
4.1 FT-QLUT and EA-QLUT Overheads 60
4.2 FT-QL and FT-QFF Area Results, Analysis and Discussion 65

5 The Fault Tolerant Quaternary Configurable Logic Block (FT-QCLB) 68
5.1 Fault Injection Experiments 70
5.2 Simulation Results, Overheads and Results Analysis 71
5.3 Area Results 74
6 conclusion and future work 76
7 references 78
Apendix A: Different bases and its relations 84
Apendix B: Contributions 88

LIST OF FIGURES

Figure 2.1. Quaternary Inverter proposed by YASUDA 21
Figure 2.2. Standard QLUT schematic circuit with 1 quaternary input 25
Figure 2.3. Quaternary inverter (a) and quaternary inverter truth table (b). 26
Figure 2.4. Boron fission induced by low energy neutron. 28
Figure 2.5. Sequential circuit. 28
Figure 2.6. Combinational circuit without radiation (a) and with radiation (b). 29
Figure 2.7. SRAM cell hardened by the inclusion of two feedback resistors 34
Figure 2.8. Combinational circuit (b) by using space (or hardware) redundancy. 35
Figure 2.9. Use of time redundancy to detect an SET in a combinational circuit. 35
Figure 2.10. Use of space redundancy to detect an SET in a combinational circuit. 36
Figure 2.11. TMR with time redundancy. 37
Figure 3.1. Labeled quaternary lookup table with 1 input 40
Figure 3.2. QLD quaternary decoder with its error detection circuit 41
Figure 3.3. Fault tolerant QLUT circuit 42
Figure 3.4. Binary LUT equivalent to the 1 input quaternary LUT 43
Figure 3.5. Complete binary LUT with the DWC technique equivalent to the FT-QLUT of Figure 3.3. 43
Figure 3.6. Approximated waveform of the double exponential current source (a) and indication of the waveform injections in an NMOS transistor (b). 44
Figure 3.7. Waveforms of a fault injection simulation at the inverter of the QLD3 signal (Figure 3.2). 45
Figure 3.8. Error Avoiding Quaternary Lookup Table (EA-QLUT) 46
Figure 3.9. Quaternary latch and quaternary inverter 48
Figure 3.10. Example of a pulse that did not generate an error (a) and example of a pulse that generated an error (b) 50
Figure 3.11. Fault Tolerant Quaternary Latch. 53
Figure 3.12. FT-Q-Latch Error detection circuit 54
Figure 3.13. Error detection and correction circuit correcting a SET. 54
Figure 3.14. Quaternary master-slave flip-flop 55
Figure 3.15. the fault detection and correction of used in the FT-QFF 57

Figure 3.16. Histogram for the 2QLUT in 90nm technology. 58
Figure 4.1. 2QLUT: 2 input QLUT...61
Figure 5.1. Standard Binary CLB composition..68
Figure 5.2. Mixed Binary and Quaternary FPGA 69
Figure 5.3. Binary to Quaternary circuit decoder 70
Figure 5.4. Quaternary to Binary circuit decoder 70

LIST OF TABLES

Table 2.1. Binary Logic Functions with 2 Inputs 18
Table 2.2. Threshold Voltage of the Different Transistors 25
Table 2.3. Truth table of the down literal circuits. 25
Table 2.4. Architectural-vulnerability-factor (AVF) estimation approaches 31
Table 3.1. QLD Truth Table 41
Table 3.2. Percentage of Errors Observed with Injections in Transistors T1, T2, T4 and T6 51
Table 3.3. Percentage of Errors Observed with Injections at Transistor T3 51
Table 3.4. Percentage of Errors Observed with Injections at Transistor T5 51
Table 3.5. Overall Percentage of Errors Observed in Each Transistor 51
Table 3.6. Quaternary Decoder Truth Table 54
Table 4.1. Area results comparing the QLUT with the binary one without the fault detection circuits 61
Table 4.2. Area results comparing the FT-QLUT with the Binary LUT using the DWC technique 62
Table 4.3. Propagation time for different sizes of the proposed QLUT and its binary correspondent. 62
Table 4.4. Propagation time (in ps) for different sizes of the proposed QLUT and its binary correspondent obtained by increasing transistor size. 63
Table 4.5. Power Dissipation (in μW) for Different sizes of the FT-QLUT and the Binary Correspondent 64
Table 4.6. Area Results of the EA-QLUT and the Two Binary Correspondents 64
Table 4.7. Propagation time (in ps) for different sizes of the proposed EA-QLUT and its binary correspondent protected with the TMR technique. 64
Table 4.8. Power dissipation (in μW) for different sizes of the proposed EA-QLUT and its binary correspondent. 65
Table 4.9. Area Results in Number of Transistors 66
Table 4.10. Area Results for the FT-QFF in Number of Transistors 67
Table 5.1. Error Rate for the Quaternary and Binary Circuits with no FT 72
Table 5.2. Error Rate by Component for the Quaternary CLB and the Binary Correspondent with no FT. 73
Table 5.3. Error Rate for the FT-Q-CLB and the Binary Correspondent. 73
Table 5.4. Area of Quaternary and Binary CLBs with no FT Technique. 74
Table 5.5. Area of Quaternary and Binary CLBs with FT Technique. 74

Table 5.6. Area Overhead Comparing the Quaternary and the Binary FT Versions 74

LIST OF ABREVIATIONS

AVF	Architectural Vulnerability Factor
ALU	Arithmetic and Logic Unit
BPSG	Boron Phospho-Silicate Glass
CLB	Configurable Logic Block
CMOS	Complementary Metal-Oxide-Semiconductor
DLC	Down Literal Circuit
DWC	Duplication with Comparison
FIT	Failures in Time
IC	Integrated Circuit
LET	Linear Energy Transfer
LUT	Lookup Tables
MOS	Metal-Oxide-Semiconductor
MTTF	Mean Time to Failure
MVL	Multiple-Valued Logic
N-MR	Modular Redundancy of order N
SE	Soft Error
SER	Soft Error Rate
SETs	Single Event Transient
SEUs	Single Event Upsets
SoC	System-on-a-Chip

SOI	Silicon-on-Insulator
SRAM	Static Random Access Memory
TMR	Triple Modular Redundancy
TVF	Timing Vulnerability Factor

1 INTRODUCTION

In computer engineering and digital systems the subject of the system radix is recurrent. The question of what radix, is optimum for a system, if there is only one possible answer? Is the well known radix-2, pushed forward by the continuous development and ability of microelectronics to provide efficient two-state devices and circuits? Some practical engineering systems would point to the radix-3, for example, a system that has the possible states of forward/stop/reverse. Others would suggest the use the radix-10 as we already do in our 10 fingers based nature or even the biggest radix off all which can be performed by the analog circuits. The decision of the radix of the system is closely related to the pros and cons offered by the radix. In digital systems, the chosen radix reflects at the data length, circuit complexity, interconnection complexity, the total pins of a component, the address bus width, the data bus width, the number of representations, power consumption, the speed of arithmetic operations among others.

Nowadays engineers are facing several problems related to the increase in the number of interconnections and pins of the integrated circuits (IC), increasing the complexity of the designs and all the other related costs, like power consumption and delay. As a possible solution to reduce the number of interconnections, the use of multiple-valued logic (MVL), which means the ability to transport more than 2 logic values in the same wire, may lead to a meaningful reduction in the number of wires required to transport the same information. By reducing the number of wires it is possible to reduce several critical aspects of a circuit, such as the critical path delay, the total area and also the total power and energy demanded. In the MVL field, quaternary logic has been reported as a way to gain area and power when using quaternary circuits instead of binary ones (MAHAPATRA; IONESCU, 2005; SHEIKHOLESLAMI; YOSHIMURA; GULAK, 1998; ZILIC; VRANESIC, 1993). Today, the memory manufacturers are the principal users of multiple-valued circuits, since they are providing drastic improvement of the storage capacity when multiple-valued technology is used (BAUER et al., 1995). Space companies in particular have started the

investigation of the possible adoption of FPGAs (GAC, 2003) based on quaternary logic in the field for space missions without any formal publishing up to this time.

The short time to market provided by FPGAs, plus the vast quantity of tools one can find to assist a design project, together with the great flexibility offered in an FPGA chip make them the first choice of designers to implement the hardware of different systems, targeting low and even medium fabrication volumes. However, with the introduction of advanced technology nodes, interconnection delays are becoming one of the main concerns when it comes to the global circuit delay, together with area and power consumption. In FPGAs the huge amount of interconnections plays an even more important role in the circuit delay (PRAGASAN, 2001), power consumption (FEI et al., 2005) and area (SINGH; PARTHASARATHY; MAREK-SADOWSKA, 2002), as one can have as much as 90% of the chip area devoted for the interconnection of the blocks.

FPGAs allow designers to implement hardware systems together with the possibility to explore any level of parallelism available in the application, with excellent time-to market and reduced non-recurring engineering costs. On the other hand, this flexibility is paid with a huge amount of circuit area, delay and power, taken by all the routing switches and wires. In order to reduce these interconnection related costs, the use of multi-valued logic, more precisely quaternary logic applied to FPGAs is an open subject with almost no meaningful research or results.

Although the appealing aspects offered, the new generation of quaternary-levels technology is resulting more sensitive to process variations due to the reduced transistor dimensions. Furthermore, these technologies will be extremely sensitive to high-energy particle strikes such as neutrons from cosmic rays and alpha particles, which represent a concern not only for spatial applications, but also for general-purpose applications working at low altitude and even at the sea level. In today’s circuits, the Soft Error Rate (SER) which represent a measurement of the radiation sensitivity of a given technology or device, is becoming one of the principal aspects in designing integrated circuits, together with performance or power characteristics. Several research works deal with the SER characterization and mitigation for binary circuits, while for multiple-value logic like quaternary logic this remains an open subject that needs to be addressed.

This thesis shows that quaternary logic circuits can have appealing characteristics for space applications, not only because quaternary circuits reduce the interconnection related costs, like for instance, number of interconnections, logic area, power consumption, but also because of some specific characteristics of the quaternary circuits that can be applied to fault tolerant circuits. The quaternary circuits presented in this thesis are a result of the study of the quaternary circuits’

vulnerabilities throughout extensive fault injection experiments taking into account the architectural and timing vulnerability aspects of the proposed circuits. After the evaluation of the quaternary circuits, modifications as well as new topologies of circuits were proposed and evaluated by comparing the proposed quaternary circuits with the binary version protected with the most faire technique, as will be explained in the experimental results analysis.

In this thesis, the reader is conducted to the subject of quaternary circuits applied to the construction of a fault tolerant CLB, which is the main block of an FPGA. At Chapter 2, it is presented the background and basic concepts that are important to the understanding of the contributions of this work, composed by the quaternary logic background and the basic quaternary circuits that have been used as starting point of this work, followed by the fault tolerance subject background. At the Chapter 3, it is presented the first contributions of this thesis trough the process followed to develop the fault tolerant quaternary circuits that are used in the construction of the fault tolerant CLB. Chapter 4 presents the area, power and performance overheads of the circuits presented at Chapter 3 while Chapter 5 presents the Fault Tolerant Quaternary Configurable Logic Block (FT-QCLB), which is the main contribution of this work. Chapter 6 presents the conclusions and open subjects that can be foreseen for the work presented in this thesis.

2 CONTEXT OF THE RESEARCH

This chapter presents the context of the research developed during this thesis and is divided in three sections. The first one presents the basis of quaternary logic followed by the quaternary circuits already developed by other researchers. In the second section, the most commonly found sources of radiation and their effects in digital circuits are presented. After that, some of the most used and known techniques applied to detect and mitigate errors in digital circuits are presented and analyzed, together with a brief discussion of the positive and negative aspects of each technique.

2.1 Quaternary Logic Background

When working with multiple-valued logic, an important metric to evaluate the logic capacity is the number of possible functions that one can obtain with a given number of inputs. This number can be calculated with the following equation (1):

$$F = B^{B^n} \quad (1)$$

Where 'B' is the radix or the base, and 'n' is the number of variables that compound the function. As a practical example let us consider a binary logic function with 2 inputs. In this case the number of functions that can be obtained can be calculated as follows (equations 2 and 3):

$$F = 2^{2^2} \quad (2)$$

$$F = 16 \quad (3)$$

Bellow one can see in Table 2.1 the 16 possible functions that one can have considering the previous example of binary logic functions with 2 inputs (A and B in table below).

Table 2.1. Binary Logic Functions with 2 Inputs

A	B	f1	f2	f3	f4	f5	f6	f7	f8	f9	f10	f11	F12	f13	f14	f15	f16
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Source: Created by the author.

One can easily identify at Table 2.1 the corresponding function implemented by the basic binary logic gates, the 2 inputs NAND gate at the f15 column, as well as the one implemented by the NOR gate, in the f9 column, and the 2 inputs XNOR gate, in the f11 column of the same table. The other functions can be obtained by a combination of the basic binary gates and the inverter gate like the AND function, presented in column f2.

In multiple valued-logic such as quaternary logic, one can have 4 possible logic values (here expressed by the numbers 0, 1, 2 and 3) and therefore, the number of quaternary functions that one can have is exponentially greater than the binary one with the same number of inputs. In order to a binary function have the same capacity as a quaternary one, it is necessary the combination of two binary circuits with the double of the number of inputs as the quaternary one. For instance, if one has a quaternary LUT with 1 quaternary input, one would need the combination of 2 binary LUTs with 2 binary inputs to have the same number of functions. This can be explained mathematically by the reasoning that follows.

Let us consider a binary function called 'F_B' with 'k' binary inputs and a quaternary function called 'F_Q' with 'j' quaternary inputs. The binary function equivalent, here called 'F_Beq', to the quaternary one is given by the following equations:

$$F_B eq = F_Q \tag{4}$$

As one needs the combination of any two binary functions to have the same capacity as the quaternary one,

$$F_B . F_B = F_Q \tag{5}$$

With equation 1,

$$2^{2^k}.2^{2^k} = 4^{4^j} \tag{6}$$

$$2^{2^{k+1}} = 2^{2.2^{2.j}} \tag{7}$$

$$2^{2^{k+1}} = 2^{2^{2.j+1}} \tag{8}$$

Which gives us,

$$k = 2.j \tag{9}$$

Which means that the number of binary inputs 'k' must be equal to the double of the number of quaternary inputs 'j' in order to have the same logic capacity as the quaternary circuit. This prove that the use of quaternary logic can reduce the number of interconnections in a circuit, and consequently reduce all the related costs like propagation delay, area and energy consumption of a quaternary integrated circuit (IC) based system. Not only the number of interconnections but also the total number of transistors of an integrated circuit can be reduced when one use multivalued circuits, especially quaternary circuits. This assumption will be proven at Chapter 4. The Appendix A is referred for more information about the binary to quaternary conversion applied to lookup tables.

The next section summarizes the most relevant research related to multivalued circuits and its pros and cons.

2.1.1 Current-Mode Multivalued-Circuits

Multivalued logic (MVL) have been the subject of research of several works in the past, motivated by the objectives of improving the quantity of information to be transmitted in a single wire, improving the capacity of storage devices and also to increase the performance of processing elements. According to Hurst (HURST, S. L., 1984), even before the advent of the semiconductor technology as we know it today, with the well consolidated CMOS technology, some multivalued logic systems were implemented with electromagnetic relay and its derivatives. Despite simple relays were binary in nature (energized or deenergized), there were also available multiple-position devices, such as ten-position uniselectors, three-position polarized relays with a central deenergized state, and other variants.

As the technology evolved to the planar MOS technology, the publishing in the multivalued logic field have also followed this new technology trend. From the late 1950's onwards, with the advent of the solid-state devices, in the form of transistors with more than two stable states, a

certain number of multivalued switching circuits, particularly ternary ones, started to be published (DRUZETA; VRANESIC; SEDRA, 1974; DUNDERDALE, 1970; HURST, STANLEY L., 1978). These first circuits were called current mode circuits because they used different level of current to emulate the different logic levels.

Only in 1983 with (FREITAS, D. A.; CURRENT, 1983a;1983b), after several attempts to develop multivalued circuits, the first large scale circuits started to be produced. By that time, the proposed non binary circuits did not intend to replace completely the binary ones, but only to offer an alternative with better performance in certain domain of applications and still being compatible with the traditional CMOS process. In certain domains of application like for instance, in the arithmetic circuits domain, the use of current-mode multivalued circuits presents excellent results in terms of performance and area (CURRENT, 1990; FRIEDMAN et al., 1977; KAWAHITO et al., 1994).

As it was stated before, these current mode circuits were designed only for specific functions that could be done in a more efficient way than the binary ones. They were intended to be only a part of a mixed, part binary and part multivalued, circuit and therefore needed converter circuits in the frontier with the binary logic. These converter circuits were inspired in the current comparator circuit and can be referred in (FREITAS, D. A.; CURRENT, 1983a;1983b; FREITAS, DAVID A.; CURRENT, 1984). Besides the arithmetic and converters circuits some work also proposed quaternary current-mode latches (CURRENT, 1989; CURRENT; HURLSTON, 1991), quaternary adder with latch (CURRENT, 1990) and current-mode quaternary analog to digital converter (ADC). As these basic current mode MVL circuits started to show up in literature, some other derivative circuits like the full adder with signed digit (GONZALEZ et al., 2000; GONZALEZ; MAZUMDER, 1998), the dual rail differential logic (HANYU; IKE; KAMEYAMA, 1999; HANYU; SAITO; KAMEYAMA, 1998) and the dual rail source coupled and the dynamic source-coupled logic (HANYU; MOCHIZUKI; KAMEYAMA, 2003).

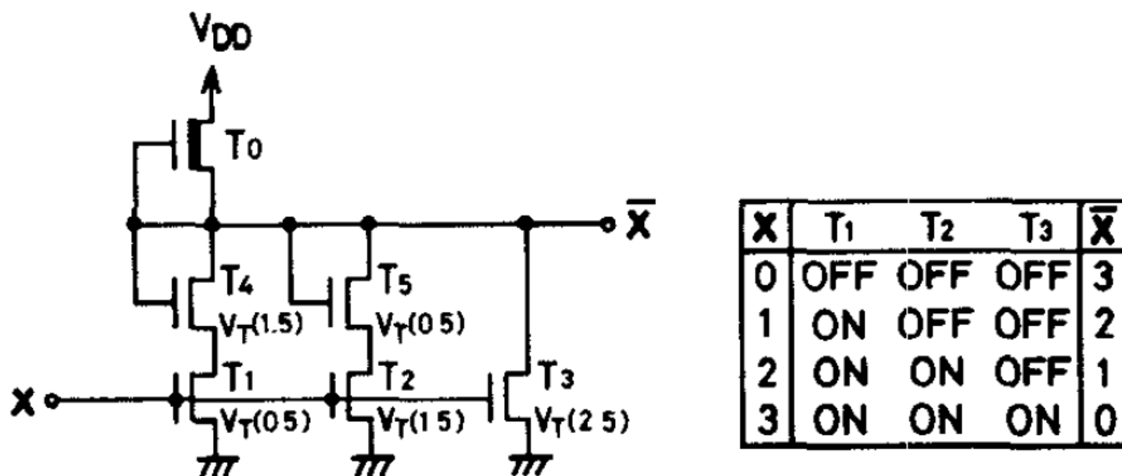
Although these multivalued current mode circuits had some advantages in comparison to the binary ones, especially in the arithmetic domain, these current mode circuits presented a high static power consumption, as they need a constant current flow to hold the different logic levels and keep the circuits working. Despite some of the works previously mentioned proposed the use of lower levels of current to reduce the static consumption, this solution not only did not solve the problem but also leaded to another important problem inherent to the multivalued logic subject, which is the reduction of the margin noise.

In the embedded system era there is no space for static consumption circuits and everything needs to be low power. This is a key reason why we do not see any current mode multivalued logic circuit working in large scale. Current mode circuits need a constant current flow that causes a high static consumption which is the reason why the current mode mutlivalued circuits were precluded and gave space to the voltage-mode multivalued circuits.

2.1.2 Voltage-Mode Multivalued Logic Circuits

In voltage mode multivalued logic circuits, the different logic levels are realized by different levels of voltage. These different levels can be implemented with voltage dividers and also dedicated sources of supply. The first work in the subject of quaternary logic using voltage-mode circuits started to be published in 1986 in (YASUDA et al., 1986). In that work, the authors presented the first version of the quaternary voltage-mode circuits called as “inverter” presented in Figure 2.1, equivalent to “NAND”, equivalent to “NOR” and “delta literal” circuits with NMOS technology. In multivalued logic, the name of the logic gates that are used in binary logic, which are “inverter”, “AND”, “OR” and their variations, lose their meaning. In multivalued logic the logic gates are usually called as “diametrical inverter”, instead of “inverter”, “MIN” instead of “AND” and “MAX” instead of “OR”.

Figure 2.1. Quaternary Inverter proposed by (YASUDA et al., 1986)



X	T1	T2	T3	X̄
0	OFF	OFF	OFF	3
1	ON	OFF	OFF	2
2	ON	ON	OFF	1
3	ON	ON	ON	0

(Source: (YASUDA et al., 1986))

One can see from Figure 2.1 that the inverter circuit proposed at (YASUDA et al., 1986) use three different enriched and one depletion transistor. The threshold voltage of each transistor is written in parenthesis. The NMOS based circuits have the drawback of the static consumption of

current. Also in (HEUNG; MOUFTAH, 1985) the authors proposed ternary circuits with lower consumption using both, PMOS and NMOS enriched and depletion transistors, as well as external sources of power supply. Also in the same work, the authors proposed the ternary "AND" and "OR" equivalent circuits.

In (THOIDIS, I. et al., 1998; THOIDIS, I. M. et al., 2001), the authors presented the low power "delta literal" circuits as well as converter circuits from binary to quaternary and vice versa using 2 PMOS and 2 NMOS transistors, both being one of enriched and another of depletion type, using 0.18µm technology with the four voltage levels defined as 0V, 1V, 2V and 3V representing the level 0, level 1, level 2 and level 3 respectively. These circuits were later used in (THOIDIS, IOANNIS M.; SOUDRIS; THANAILAKIS, 2004) to propose the first voltage-mode quaternary full adder presented in literature. Simulation results using the SPICE (NAGEL, 1973) simulator with 0.7 µm technology and presented a critical path delay of 15.9ns.

In (SHIBATA; OHMI, 1993) the authors propose the so called "Neuron MOS" transistors constructed with a double poly-silicon layer. This double poly-silicon layer allows one to set a certain threshold voltage (Vth) deviation in the transistor, which enables the construction of multi Vth transistors that are used in the implementation of the basic circuits used in the construction of the multivalued basic gates. Although this work proposes an alternative process to the traditional ion implantation, this method demands the "programming" of the additional poly-silicon layer of each transistor, with its different VT deviation, which is a big overhead. Using the proposed "Neuron MOS" transistors, the authors in (PARK et al., 2004) presented the quaternary MIN/NMIN, MAX/NMAX and the quaternary adder/subtractor circuits with 0.35µm technology simulated with the SPICE simulator.

In (LEBLEBICI; GURKAYNAK, 1998; OZDEMIR et al., 1996; SCHMID; LEBLEBICI, 2004) the authors proposed multivalued logic circuits equivalent to the inverter, NAND and NOR binary gates using the so called "Capacitive Threshold Logic".

Despite of the big number of publications related to the multivalued logic topic of research, none of the proposed work presented real advantages of using the multivalued version of circuits instead of the traditional binary ones. The current mode attempts, presented good performance results but with the cost of static current consumption, which precluded its usage. When it comes to the publications related to the voltage-mode circuits presented up to the year of 2004, none of the works presented were able to combine low power and high or even compatible performance to the binary circuits.

In (GONÇALVES DA SILVA, R. C.; BOUDINOV, H.; CARRO, L., 2006), the authors proposed a new family of voltage-mode quaternary CMOS circuits composed of a quaternary inverter, NMIN and NMAX circuits, with high performance and low power/energy consumption than any other quaternary voltage mode circuits proposed up to that time. The circuits proposed used three sources of supply, 3V, 2V and 1V, besides the reference 0V that formed the four logic levels of the quaternary circuits and where simulated at the TSMC 0.18 μm technology. To implement the quaternary circuits the authors used 3 PMOS and 3 NMOS with different voltage thresholds. Later on, the same authors proposed in (GONÇALVES DA SILVA, R. C.; BOUDINOV, H. I.; CARRO, L., 2006) a quaternary full adder with 76% reduction in power dissipation, 15% improvement in performance with 20% of area overhead in comparison to the equivalent circuit implemented in binary logic. The authors used the same TSMC 0.18 μm technology, transistors and voltage sources as in the previous work. In (CUNHA; BOUDINOV; CARRO, 2007), the same authors compared quaternary look-up tables with the correspondent binary ones. Simulation results showed that the quaternary LUT with 2 inputs had 49.8% less transistors, 70.3% less power dissipation and 5% better performance than its binary equivalent.

More recently in (VASUNDARA; GURUMURTHY, 2009), the authors presented the same circuits with the same topologies as proposed in (GONÇALVES DA SILVA, R. C. et al., 2006) with the only difference of the technology node used, which was the 90nm MOSIS. As expected the authors obtained faster and lower power circuits in comparison to the previous work. In (DATLA; THORNTON, 2010) the authors present quaternary serial and parallel multipliers with decreased dynamic power dissipation and improved timing performance when compared to binary circuits with equivalent word sizes, especially in the parallel version.

2.1.3 Voltage-Mode Multivalued Logic Circuits Applied to FPGAs

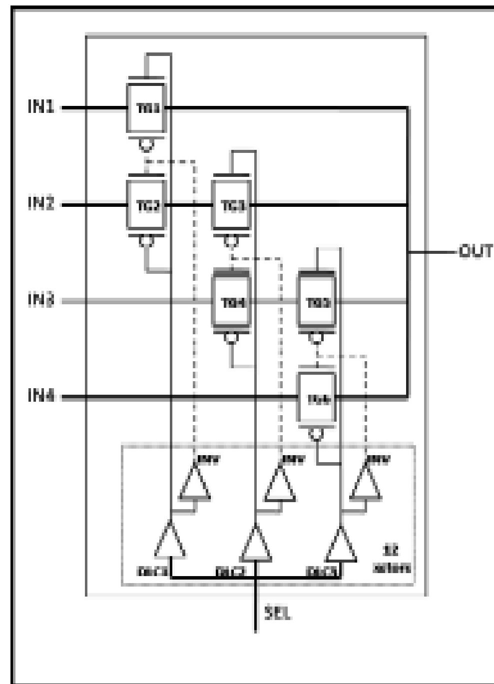
There is not much published work related to multivalued logic applied to FPGAs up to the time that this thesis was written. When it comes to the use of quaternary logic circuits to be used in an FPGA, the work proposed in (RITT et al., 2010) the authors show that when one use quaternary circuits to implement FPGAs one can have a reduction of 27% in the number of transistors and 19% in the number of nets in average. Also in (LAZZARI et al., 2010a;2010b) the authors show that when one use quaternary circuits to reduce the interconnection costs, especially critical in FPGAs, one can have an average gain in area of 33%, less power consumption of 16%, with a small penalty in performance of 11%, comparing to the binary equivalent. The most important result of the previously mentioned paper is that one can have a reduction of more than 50% in wire length and number of switches in average, for the benchmark chosen by the authors.

This work proposes a fault tolerant quaternary CLB (FT-Q-CLB) that can be used in the construction of a fault tolerant quaternary FPGA. The FT-Q-CLB of this work is constructed with quaternary logic circuits. Most of the fault tolerant quaternary circuits that are presented here were based on some of the works presented in the previous chapter, but with some hardware improvement and implemented using the 90nm technology with VCC of 1.2V. This way, the four logic values ‘0’, ‘1’, ‘2’ and ‘3’ have the corresponding voltage of ‘0V’, ‘0.4v’, ‘0.8V’ and ‘1.2V’ respectively. The following chapter presents the quaternary circuits used as base for the development of the fault tolerant components used in the construction of the FT-Q-CLB proposed in this work as well as the characterization of the SET phenomena that is considered in this work.

2.2 Quaternary Base Circuits

One of the main components of the quaternary CLB is the quaternary Lookup-Table proposed by here referred as QLUT that is illustrated in Figure 2.2. The quaternary lookup table (QLUT) of 0 is based on voltage-mode quaternary logic CMOS circuits. These circuits use several different transistors with different threshold voltages that are presented in Table 2.2, and operates with four voltage levels, corresponding to the logic levels ‘0’, ‘1’, ‘2’ and ‘3’. The QLUT cell is designed using Down Literal Circuits (DLCs), binary inverters and pass transistor gates. The Down Literal Circuits perform the decoding function shown in Table 2.3. There are three possible Down Literal Circuits in quaternary logic, named DLC1, DLC2 and DLC3. The Down Literal Circuits are designed in CMOS technology with 3 different threshold voltages for PMOS transistors, and 3 different threshold voltages for NMOS transistors.

Figure 2.2. Standard QLUT schematic circuit with 1 quaternary input



(Source: Created by the author)

One can understand the behavior of the QLUT of Figure 2.2 by looking at the DLC truth table at Table 2.3. More details on the QLUT circuit can be found in (CUNHA et al., 2007).

Table 2.2. Threshold Voltage of the Different Transistors

Name	T1	T2	T3	T4	T5	T6
Threshold Voltage	-1.0	1.0	0.6	0.2	-0.2	-0.6
Type	PMOS	NMOS	PMOS	NMOS	PMOS	NMOS

(Source: Created by the author)

Table 2.3. Truth table of the down literal circuits.

In (logic level)	DLC1 (logic level)	DLC2 (logic level)	DLC3 (logic level)
0	3	3	3
1	0	3	3
2	0	0	3
3	0	0	0

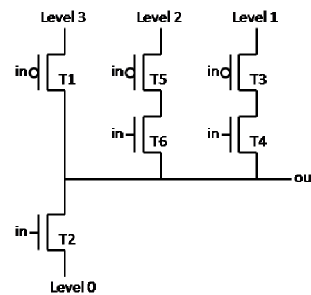
(Source: Created by the author)

Another important quaternary circuit that is used in the construction of the FT-Q-CLB is the voltage-mode quaternary inverter proposed by (GONÇALVES DA SILVA, R. C. et al., 2006). In quaternary logic, the inverter function has a different meaning as in the binary logic, in which the 2 existent logic values are each other's inversion. In the quaternary logic as there are 4 possible logic

values, the quaternary inverting function must follow a diametrical inversion. Figure 2.3(a) presents the quaternary inverter circuit implemented with the transistors constructed with the threshold voltages presented in Table 2.2. In the same figure one can find the truth table of the quaternary inverter.

Figure 2.3. Quaternary inverter (a) and quaternary inverter truth table (b).

in	out
0	3
1	2
2	1
3	0



(a) **(b)**

(Source: Created by the author)

The inverter circuit of Figure 2.3(b) is used to construct a fault tolerant quaternary flip flop that is an important component of the Fault Tolerant Quaternary CLB and will be presented latter on.

In order to have the capacity to tolerate faults, the proposed circuits of this work use variations and also some improvements of some of the well known fault tolerant techniques published so far. The next section introduces the fault tolerance subject starting with the most common sources of radiation, followed by the metrics used in this area and finally the most referenced fault tolerance techniques with a brief discussion of the drawbacks of each technique.

2.3 Fault Tolerance Background

This section presents a description of the different types of spatial radiation that can produce or stimulate bit flips in circuits. This section also discusses the most commonly found sources of radiation and their effects in digital circuits. Also, the most used metrics that are applied to measure the vulnerability of the circuits are described followed by some of the most used and known techniques applied to detect and mitigate errors in digital circuits. These different techniques are presented and analyzed, together with a discussion of the positive and negative aspects of each technique.

2.3.1 Radiation Sources and their Effects

There are different types of space radiation that can cause soft errors. In this section the most known and relevant types of radiation sources that can cause SEUs and SETs are presented. Also the effects that SEUs and SETs can cause and the conditions to an error in a certain circuit occur are discussed (HEIJMEN, 2002).

The main sources of radiation catered from space are:

a) alpha particles;
b) high energy cosmic neutrons;
c) boron fission induced by low energy neutrons.

There are other kinds of particles that can cause soft errors, like heavy ions for instance, but they will not be discussed here because they are only relevant for aero-space applications, due to their occurrence only in space or in the highest parts of the earth atmosphere.

2.3.1.1 Alpha Particles

An alpha particle is a doubly ionized helium atom, made of two protons and two neutrons. Alpha particles can be found in circuits packaging materials, solder points of the integrated circuits or in wafers, which are thin slices of semi-conductor material, upon which circuits are constructed. When an alpha particle hits a beta or gamma ray, it loses energy and generates transient current pulses that, depending on their intensity, can cause an SEU (single event upset) which can result in a soft error if it compromises the correct functionality of the circuit.

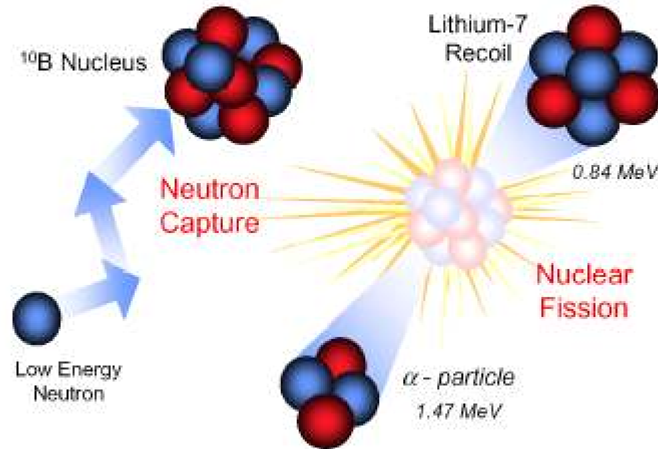
2.3.1.2 High Energy Cosmic Neutrons

This kind of particle is formed by the collision of galactic particles and solar wind particles with the terrestrial atmosphere. Most of cosmic rays are reflected or captured by the geomagnetic field of the earth, and only 1% of the high energy cosmic neutrons hit the earth surface, generating a flux of 25 neutrons/cm^2.hr (ZIEGLER; LANFORD, 1981) with energy higher than 1 MeV (1 million electron volt) at sea level. Only neutrons with 5 MeV or higher energy are capable of generating soft errors.

2.3.1.3 Boron Fission Induced by Low Energy Neutrons

Another form of radiation can occur when low energy neutrons interact with boron atoms (BAUMANN, R. et al., 1995). As a result, a lithium core and an alpha particle are generated by fission, as depicted in Figure 2.4. Both particles resulting from this reaction are capable of generating SEUs or SETs that can cause the undesired soft errors.

Figure 2.4. Boron fission induced by low energy neutron.

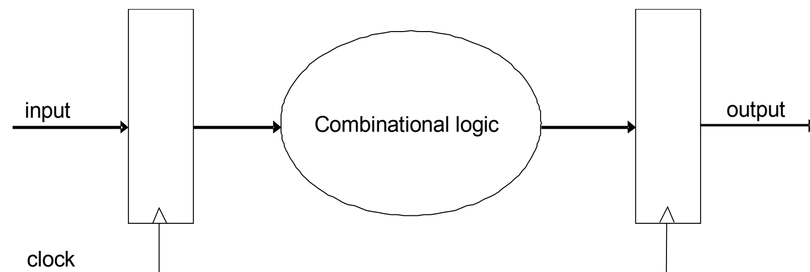


(Source: (BAUMANN, R. C., 2001))

2.3.2 Effects of SEUs and SETs in Digital Circuits

A particle hit can affect a combinational as much as a sequential part of a circuit (ALEXANDRESCU; ANGHEL; NICOLAIDIS, 2002). In sequential circuits, like the one shown in Figure 2.5, SEUs can occur only in memory elements (the registers in Figure 2.5). On the other hand, the combinational components can be affected by SETs which, given the right circumstances, can cause an error. The hit of a radiation particle in a memory element does not imply that an SEU will be registered. In order to an SEU occur, it is necessary that this particle has enough charge to create a significant current pulse. In other words, it is necessary that the charge generated by the particle is greater than or equal to the so called critical charge ($Q_{critical}$) of the hit element. The $Q_{critical}$ will be explained with more details in following paragraph.

Figure 2.5. Sequential circuit.

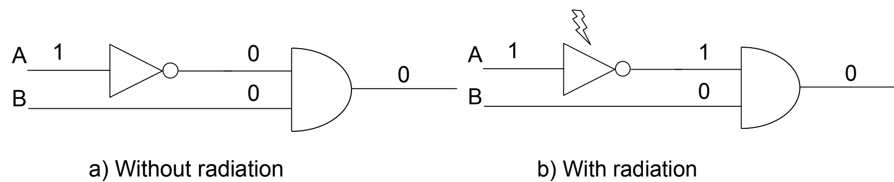


(Source: Created by the author)

The occurrence of an SET in combinational logic does not mean that an error will result. In order to an error occur, a combination of events must happen, allowing the SET to be captured or generate and erroneous operation. First, it is necessary that the charge generated by the radiation

source be equal or higher than the $Q_{critical}$ of the element that was hit. Second, the combinational circuit must be fast enough to propagate the error, and third, the logic of the architecture must allow that the wrong logic value that was generated propagates to some memory element during its latching window or generate an erroneous operation. In Figure 2.6, one can see an example in which the combinational circuit does not allow the SET propagation. The Figure 2.6 shows a small combinational circuit in two different situations. In the first situation (a), the circuit is free of the radiation effects, while in the second (b) the circuit is being affected by a source of radiation. One can see that in booth circuits the result is the same even in the presence of radiation.

Figure 2.6. Combinational circuit without radiation (a) and with radiation (b).



(Source: Created by the author)

2.3.3 Metrics to Evaluate the Vulnerability of Circuits to Soft Errors

The vulnerability of a circuit to soft errors indicates the probability of the circuit to have an error. This probability indicates to the end user how much he can rely on the correct operation of the circuit. With the growing concern about circuit reliability, companies are using some metrics to evaluate their products. In this section, some of the most used metrics proposed by scientists and designers to evaluate the vulnerability of the circuits, which is known as the soft error rate (SER), are presented.

2.3.3.1 Failures in Time (FIT)

The fault rate of a circuit can be measured through the number of failures that occur in a certain period of time. This metric is known as *Failures in Time*, or *FIT*. If a circuit has a fault rate of 1 FIT, it means that in a period of 1 billion hours 1 fault will probably occur. Some companies, like IBM, are using this metric as a reference to the design of their products. IBM sets its target for undetected errors caused by SEUs to 114 FIT (BOSSEN, 2002), which means that 1 fault may occur in the time range of about 9 million (8.771.930 to be more precise) hours of device operation. The additive property of FIT makes it convenient for calculation of the fault rate of large systems, because the designer just needs to sum the FIT of all components that are part of the system to have the system FIT.

2.3.3.2 *Mean Time to Failure – MTTF*

Another metric that can be applied to measure the fault rate of a system is the mean time to failure. Differently from the FIT, the MTTF is more intuitive, because it indicates the mean time that will elapse before an error occurs. The MTTF has an inverse relation to the FIT, which is expressed by the following equation (10):

$$MTTF\left(hours\right)=\frac{10^{9}}{FIT} \tag{10}$$

2.3.3.3 *The Soft Error Rate Estimation*

The soft error rate (SER) of a system can also be expressed in terms of the nominal soft error rates of individual elements that are part of the system, such as SRAMs, sequential elements such as flip-flops and latches, combinational logic, and factors that depend on the circuit design and the microarchitecture (NGUYEN; YAGIL, 2003; SEIFERT et al., 2001), as follows in equation (11):

$$SER^{design}=\sum_{i} SER_{i}^{nominal}\times TVF_{i}\times AVF_{i} \tag{11}$$

where i stands for the ith element of the system.

The SER^{nominal} for the ith element is defined as the soft failure rate of a circuit or node under static conditions, assuming that all the inputs and outputs are driven by a constant voltage. The TVF_i, time vulnerability factor (also known as time derating) stands for the fraction of the time that the element is susceptible to SEUs that will cause an error in the ith element. The AVF_i, architectural vulnerability factor (also known as logic derating) represents the probability that and error in the ith element will cause a system-level error.

The SER^{nominal} is defined by the probability of occurrence of an SEU in a specific node of the element. This probability depends on the element type, transistor size, node capacitance and other characteristics of the element. For instance, to estimate the SER^{nominal} for a latch, one must know the $Q_{critical}$, which identifies the minimum charge necessary to cause the element to fail. This can be done by injecting waveforms of alpha and neutron particle hits on all relevant nodes. Then, it is necessary to evaluate the alpha and neutron flux to which the circuit is submitted. More details can be found in (SEIFERT et al., 2001).

The timing vulnerability factor can be summarized as the fraction of time that the element can fail. For example, the timing vulnerability factor of a latch is equal to the portion of the time that the latch is in its store mode. For combinational logic, the timing vulnerability factor depends

on its type, which can be data path or control path. More details on these and other TVF evaluation aspects can be seen in (NGUYEN; YAGIL, 2003; SEIFERT et al., 2001).

The architectural vulnerability factor of an element can be understood as the probability that a fault in that element causes an error in the system. In Table 2.4 some approaches to estimate the AVF, its major issues, advantages and disadvantages are presented.

Table 2.4. Architectural-vulnerability-factor (AVF) estimation approaches

Approach	Description	Major issues	Advantages	Disadvantages
Fault injection	Inject error(s) and simulate to see if injected error(s) cause(s) system-level error(s) by comparing the system response with simulated fault-free response	* Which inputs to simulate; * How many errors to inject; * Which signals to inject errors in; * Which signals to use for comparison.	* Applicable to any design; * Easy automation.	* Long simulation time (several days or weeks) for statistically significant results; * Dependence on chosen stimuli.
Fault-free Simulation	Perform architectural or logic simulation and identify situations that do not contribute to system- level errors, such as unused variables and dead instructions.	* Which inputs to simulate; * How to identify situations that do not contribute to system-level errors.	* Much faster compared to fault injection; * Easy automation.	* Applicable to very specific designs and not general enough; * Dependence on chosen stimuli.

(source: (MITRA et al., 2005))

The AVF value of an element depends on its inputs and also on how important that element is for the circuit considering its functionality. As an example, suppose that the contents of a flip-flop are erroneous. If the flip-flop output drives to an AND gate with another signal whose logic value is 0, the error will have no effect on the output of the AND gate.

The soft error experiments presented in this work are evaluated using an approximation to the methodology explained in this section, which is the soft error rate estimation. More details on the SER experiments will be explained in Chapter 3.

2.3.4 Mitigation Techniques for SEUs and SETs

In the first years of spatial exploration, the reliability of the circuits started to become an important concern for designers. At that time, the major technique used to protect circuits was shielding. This shielding technique worked by reducing the particle flow to smaller levels and consequently, reducing the number of errors caused by particle hit to zero. During many years this technique was widely used in aero-spatial applications and guaranteed the correct operation of the circuits. However, with the technology evolution up to nanometer scale, circuits became more susceptible to particle hits, making this shielding technique obsolete for special circuits and even for circuits to be used at sea level.

Trying to reach the level of reliability that once belonged to shielding, scientists have proposed several techniques in the past years, each one with its pros and cons, to mitigate SEUs and SETs. In this section, some of these techniques are presented and their costs, in terms of area and processing time overheads, are discussed.

2.3.4.1 Process Modification related techniques

Several process solutions have been proposed to reduce SER sensitivity of circuits, including the usage of well structures, buried layers, deep trench isolation, and implants at the most sensitive nodes. Also wafer thinning has been proposed as a way to reduce SEU sensitivity (DODD et al., 2001). It was shown that the overall SEU threshold LET (linear energy transfer) can be significantly increased if the substrate thickness is reduced to 0.5 μm. In practice, however, several criteria would have to be met to make the thinning of fully processed wafers possible. Another reduction of the SER can be achieved by reducing to almost zero the contribution of errors caused by the particles resulted by the boron fission reaction. This can be done by eliminating BPSG (boron phosphor-silicate glass) from the process flow. If the use of BPSG is necessary, enriched ^{11}B could be used in the BPSG layers (BAUMANN, R. C., 2001). Silicon-on-insulator (SOI) technologies are relatively insensitive to soft errors. Applying SOI technology instead of the corresponding bulk process improves the SER with a factor in the range of 2 to 8 (HARELAND et al., 2001). However, the cost of materials, especially of the wafers, is higher for SOI. In general, these process modification solutions are expensive and are applied just for a few designs.

2.3.4.2 Component Hardening Techniques

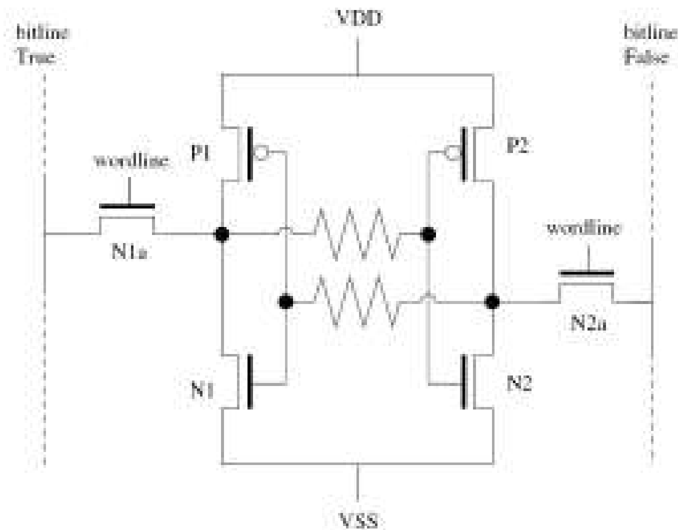
There are two basic approaches to improve SER sensitivity at the circuit level. On one hand, the components applied in the design can be modified such that they become less susceptible to soft errors. The main goal of this approach, often named design hardening, is to manufacture SER-reliable circuits using standard CMOS processing without additional masks (VELAZCO et al.,

1994). On the other hand, one can accept that soft errors occur at a certain rate and include extra circuitry to detect and correct them. Error detection and correction techniques are discussed in the next subsection.

Solutions to reduce the SER sensitivity of components can be categorized as techniques to increase the capacitance of the storage node, to reduce the charge collection efficiency, or to compensate for charge loss. The applied design style can have an important effect on SER. For instance, in (SEIFERT et al., 2001) it is demonstrated that level-sensitive latches using transmission gates are more sensitive than edge-triggered static latches, because the former use floating nodes to store information.

Another method to improve SER sensitivity is to enlarge the critical charges by increasing the capacitance of the storage nodes. In fact, if all critical charges are sufficiently large, alpha particles are not able to upset a circuit and neutrons are the only source of soft errors that can affect the circuit. In (KARNIK et al., 2001), an explicit feedback capacitor is added to the node capacitances. In (OOTSUKA et al., 1998), a SER-hardened SRAM cell used stacked cross-coupled interconnects to increase the capacitor area. Enlargement of the node capacitances are not only applied in memory design, but were also shown to be an efficient way to improve the SER sensitivity of sequential or domino nodes in high-performance circuits (KARNIK et al., 2002). The main drawback of increasing the node capacitances is that generally the cell area is increased affecting the memory overall area. The SER sensitivity of SRAM cells and latches can also be improved by adding feedback resistors between the output of one inverter and the input of the other, as shown in Figure 2.7. This SRAM cell topology was proposed in (SEXTON et al., 1991). The transient pulse induced by an ionizing particle is filtered by the two resistors, which slow down the circuit such that it does not have sufficient time to flip state. However, the inclusion of feedback resistors in a memory element has the drawback that the write speed is lowered (VELAZCO et al., 1994).

Figure 2.7. SRAM cell hardened by the inclusion of two feedback resistors



(Source: (SEXTON et al., 1991))

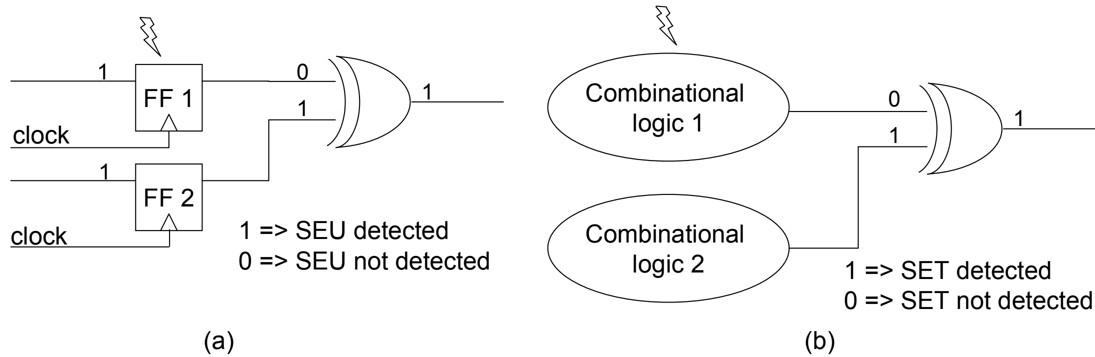
2.3.4.3 Circuit Design SEU and SET Hardware Mitigation Techniques

As stated in a previous subsection, process modification solutions are expensive and are used just in few designs with high volume. Also, component hardening techniques involve costs in energy, area and performance that sometimes may not be reasonable for manufacturers. Therefore, the development of techniques not related to the process variation or component modification has been stimulated during the past years, and some design based mitigation techniques have been proposed for the scientific community. In this section, some of the most know and widely used design techniques that have been proposed by researchers worldwide are presented. These techniques are divided into two main groups: error detection techniques and error detection and correction techniques.

2.3.4.4 Hardware Error Detection Techniques

The error detection techniques are based in redundancy to detect if an error has occurred. This redundancy can be hardware redundancy, also known as space redundancy, or time redundancy. The hardware redundancy approach called duplication with comparison (DWC) is based in the duplication of the module which failing behavior has to be detected, followed by the comparison of the outputs of both modules. If the results do not match, an error signal is activated. This technique can be used to detect either SETs in combinational circuits or SEUs in memory elements. Figure 2.8 illustrates these two situations, time (situation a) and space (situation b) redundancy, to detect SEU and SET, both with one error detected.

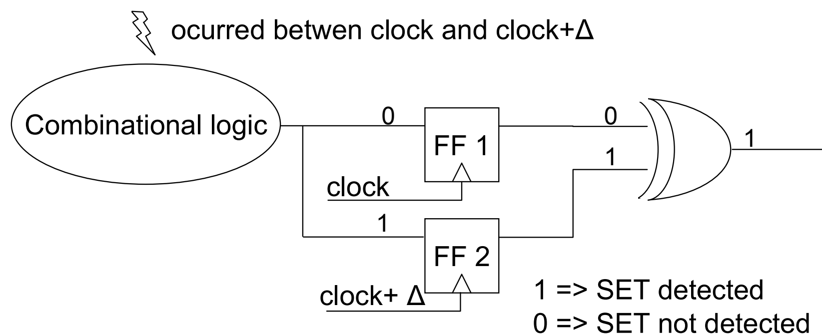
Figure 2.8. Combinational circuit (b) by using space (or hardware) redundancy.



(Source: Created by the author)

Time redundancy can be used to detect SETs in combinational logic. This technique detects SETs by capturing the output of the combinational circuit in two different moments in time. The two captured values are compared, and in case of different values, an SET detection is indicated. Figure 2.9 illustrates the use of time redundancy to detect an SET in a combinational circuit.

Figure 2.9. Use of time redundancy to detect an SET in a combinational circuit.



(Source: Created by the author)

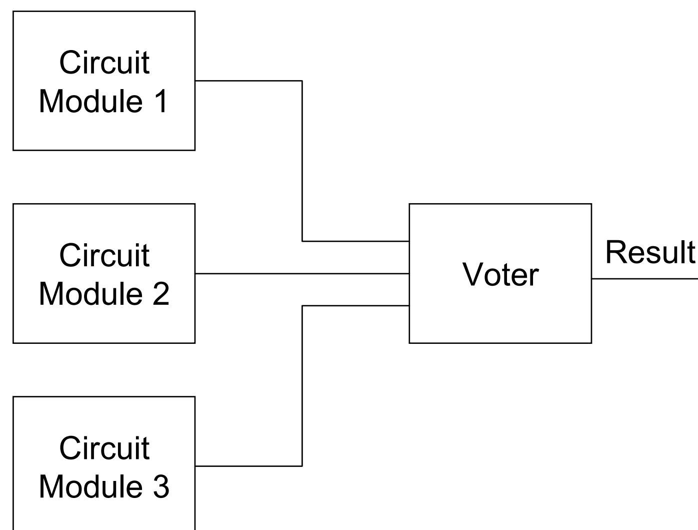
The circuit designer must set the "Δ" time wide enough to allow the SET propagation, but also short enough not to lose the pulse. If a particle hits one of the memory elements used to capture the values, an SEU will be registered and an SET will be erroneously detected. The main drawbacks of detection techniques based on duplication are: the hardware area is more than doubled, and they are only able to detect the events, and not to avoid the occurrence of an error. This way, if the designer wants the circuit to operate correctly, it is necessary that the event detection flag indicates that the operation needs to be repeated and the wrong value must be discarded.

2.3.4.5 Hardware Error Detection and Correction Techniques

With the necessity of not only detecting but also correcting the soft errors, researches have proposed some detection and correction techniques based on redundancy of modules. The most

commonly used and known hardware error detection and correction technique is the so called Triple Modular Redundancy, or TMR. The triple modular redundancy (JOHNSON, 1994) first proposed by Von Neumann in 1956, uses the redundancy of modules to guarantee the correct functionality of the circuits in which it is implemented. This technique is based on the triplication of the protected module in a way that, if any of the three modules fails, the other two will guarantee the correct operation of the system. The redundancy used in this technique can be time redundancy or space redundancy. In Figure 2.10, the use of space redundancy of the component that is being protected, together with a voter block, is illustrated. The voter is the module that votes, or chooses, for the majority result from the component blocks to be the circuit result.

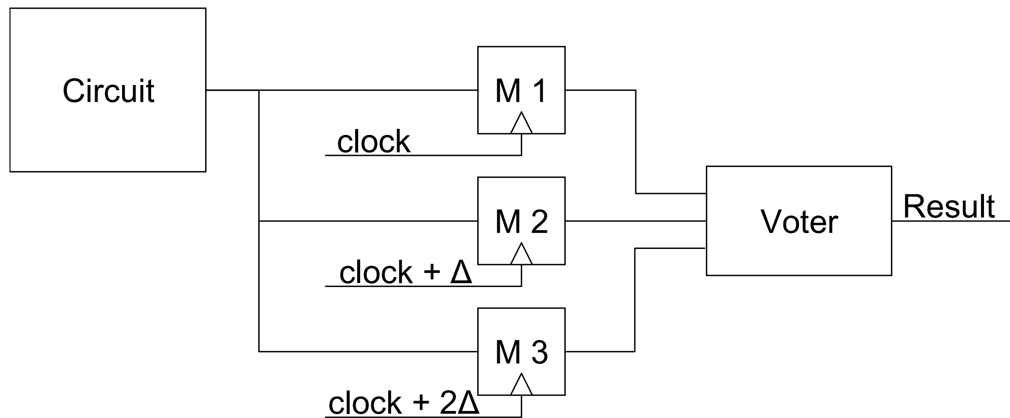
Figure 2.10. Use of space redundancy to detect an SET in a combinational circuit.



(Source: Created by the author)

Since all the three modules operate in parallel, this technique corrects any failure in one of the three modules with the performance penalty of the voter delay. On the other hand, the area overhead is more than 200%, due to the triplication of the protected module and the voter. Depending on the size of the module, this area penalty can be a price that the designer cannot afford. In Figure 2.11, the use of TMR with time redundancy to correct a fault in one module is illustrated. The TMR with time redundancy only triplicates the memory elements responsible for capturing the result of the circuit at different moments in time. If we compare the area of both TMRs, the time and the space one, we can say that the time TMR has the lower area overhead if the size of the circuit is smaller than the memory element. On the other hand, the time redundancy TMR will have bigger performance penalty due to the different need to capture the circuit values at three different moments in time. Also, the clock circuit with the two "Δ" delays adds some extra complexity to the circuit design.

Figure 2.11. TMR with time redundancy.



(Source: Created by the author)

However, the voter is not free of faults and if a fault hits the voter, the system reliability can be compromised. It is important to mention that the TMR technique is only effective against single faults and in case of a double faults, which means two faults affecting each one a different module, the voter can choose a wrong answer as if it were correct. To guarantee the system reliability against multiple faults, the redundancy has to be increased. This way, N-MR - Modular Redundancy of order N, uses a higher number of modules to guarantee that the majority of the modules operate correctly. In case of double faults, the number of duplicated modules must be five. This way, if two blocks fail, the other three will operate correctly and the voter will be able to choose the right result from the majority. Despite its tolerance to multiple faults, the N-MR has a huge area overhead, which gets to more than 400% for the 5-MR, due to the addition of four copies of the protected module and the voter block. Also, the size of the voter grows geometrically when compared to the TMR version. Since the voter is sensitive to faults, the reliability of the system can be compromised if the size of the voter grows too much.

There are also a lot of software based fault tolerant techniques that can be applied in software based systems. These software based techniques generally have lower area overhead in comparison to the previously mentioned hardware techniques but on the other hand, these techniques end up adding non recurring engineering cost and usually demand considerable extra memory space for the software redundancy. Also there are some errors detecting and correcting codes applied mainly to memory components that are not in the scope of this work and therefore will not be presented here.

3 FAULT TOLERANT QUATERNARY CIRCUITS

This section presents the fault tolerant quaternary circuits that were developed based on the voltage-mode quaternary circuits presented in (CUNHA et al., 2007; GONÇALVES DA SILVA, R. C. et al., 2006). The circuits presented in this section were designed, simulated and analyzed separately to, latter on, be assembled together to form the main result of this work, which is the fault tolerant quaternary configurable logic block (FT-QCLB). In order to evaluate the variability impact of the quaternary circuits, Mote Carlo simulation results are presented for the Quaternary LUT cell with 2 inputs.

The first fault tolerant quaternary circuit developed was the quaternary lookup-table (FT-QLUT), which is presented in the following subsection.

3.1 Fault Tolerant Quaternary Lookup Table (FT-QLUT) and the Error Avoiding Quaternary Lookup Table (EA-QLUT)

In this subsection it is presented the FT-QLUT that was designed to detect and tolerate single faults by indicating whenever a single event generates a soft error. In order to become fault tolerant, the Quaternary LUT presented in Figure 2.2 was studied in the search of its fragilities. After finding all the weakness of the QLUT, it was proposed a fault tolerant technique to overcome each of them. The next subsections present first, the evaluation of the weak points of the QLUT of 0 followed by the proposed architecture to overcome its weakness and the fault injection experiments that were taken to evaluate the proposed architecture.

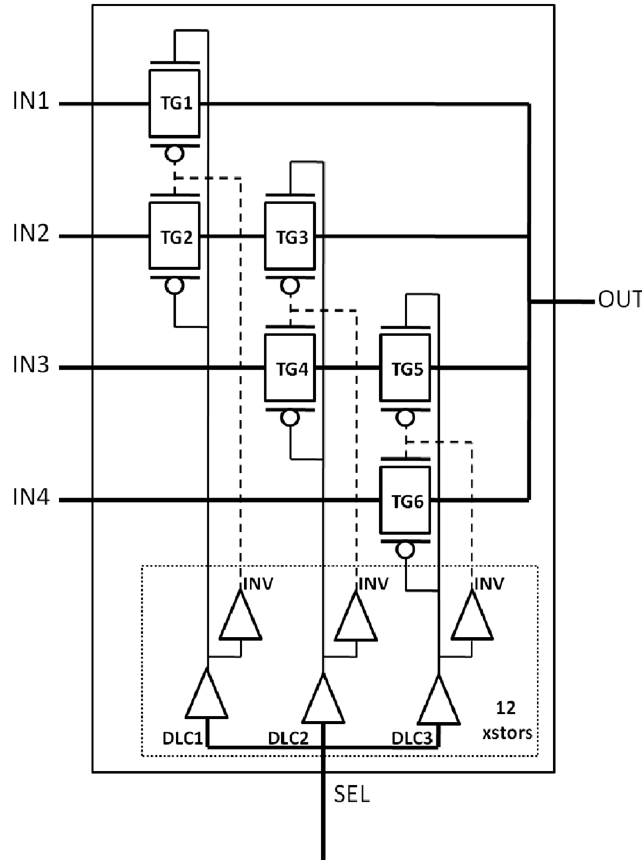
3.1.1 QLUT Fragilities

In order to propose fault tolerant version of the QLUT, it was started a study to raise the fragilities of the QLUT cell first presented in 0 and represented bellow in Figure 3.1 with some important components and connections labeled. As one can see from Figure 3.1, the quaternary input (labeled as “SEL” at the figure) is decoded by the 3 types of DLCs in binary values according to the table already presented (Table 2.3). These decoded binary values, together with its inverted values are used to control the so called transmission gates (TGs) to connect the desired quaternary signal to the circuit output. In a free of faults environment, the correct behavior of the cell is guaranteed, but in the presence of faults caused by an event like a high energy particle strike at a critical component of the QLUT like, for instance, one DLC, could compromise the QLUT functionality. In order to illustrate this scenario, let’s consider this example in which the SEL input holds the quaternary level of ‘0’. In a free of faults environment, this would mean that the value present at the input ‘IN1’ would be connected to the QLUT output. On the other hand, if a fault event would affect the DLC1 component with enough charge to provoke its incorrect behavior, this would result in an inversion of its output value, which in our example would be from quaternary level ‘3’ to quaternary level ‘0’. This event would change the selected value to be connected to the QLUT output from the value connected to ‘IN1’ to the value connected to ‘IN2’, which would characterize an error at the QLUT. The origin of this error can be understand by looking at the DLC decoding table (Table 2.3) and comparing the decoding of level ‘0’ and level ‘1’. One can see that only the DLC1 component contributes for the difference between the two decoding values and in case that the DLC1 fails, the ‘IN2’ connected value can be erroneously selected instead of the ‘IN1’ one, and vice-versa. The same behavior can be observed in case that the input level is equal to quaternary ‘1’ or ‘2’, in which only the DLC2 component is responsible for the differentiate between the value connected to ‘IN2’ or ‘IN3’. At the same way, the DLC3 component is the only one to differentiate between the quaternary input levels ‘2’ and ‘3’. The conclusion of this analysis is that the decoding signals generated by the DLCs are not robust enough to tolerate transient faults. Even if one use a circuit to detect errors in the decoding, this circuit would not be able to differentiate between an erroneous code and a valid one as the erroneous one could also be valid as they are equal to valid codes.

Besides the DLCs, there are other sources of vulnerability that can be pointed which are the transmission gates TG1 and TG6 of Figure 3.1. The TG1 and the TG6 are lonely in the connection path between an input ‘IN1’ and ‘IN4’ respectively and the output. This way if a faulty event happens to cause the undesired conduction of one of these transistors at the same time that another

input is selected, and if the two paths have different values, this would be characterized as an error in the QLUT circuit.

Figure 3.1. Labeled quaternary lookup table with 1 input



(Source: Created by the author)

After identifying the vulnerable points of the QLUT circuit, it was proposed some modifications in the design of the QLUT cell in order to present a more reliable functionality. The next subsection presents the proposed FT-QLUT architecture.

3.1.2 Proposed Architecture

In order to improve the reliability of the QLUT circuit, it was necessary a new decoding circuit to control the transmission gates of the QLUT as well as more redundancy in the paths of the TG1 and TG6 that were identified as vulnerable.

The problem of the DLC decoding circuit was that there were only one component that could differentiate between two inputs and in case of a failure in that component, an erroneous input could be connected to the QLUT output. In order to solve that fragility, it was proposed a decoding circuit that has more than one, in this case two different components that need to have its value changed in order to connect each selectable input to the output. This decoding circuit is called as Quaternary Level Decoder (QLD) and has its truth table presented at Table 3.1.

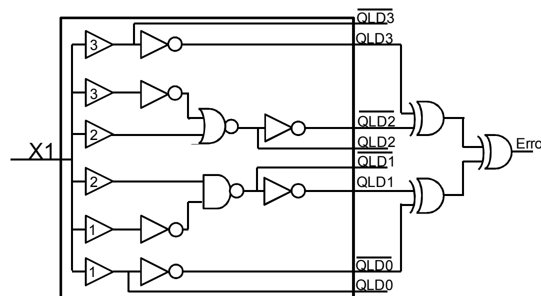
Table 3.1. QLD Truth Table

Input level	QLD0	QLD1	QLD2	QLD3
0	3	0	0	0
1	0	3	0	0
2	0	0	3	0
3	0	0	0	3

(Source: Created by the author)

One can see from the table Table 3.1 that for any quaternary input level the generated code has only one out of the four values equal to ‘3’ (the binary equivalent to High). If one of the QLDs changes its value, the generated code will always be different than any other existing and possible code. This way, one can use a detection circuit to detect the presence of an invalid code. The circuit that implements the quaternary level decoder is composed basically by DLCs, and traditional binary logic gates like INVERTER, NAND and NOR gates and is presented in Figure 3.2 already connected to the error detection circuit.

Figure 3.2. QLD quaternary decoder with its error detection circuit



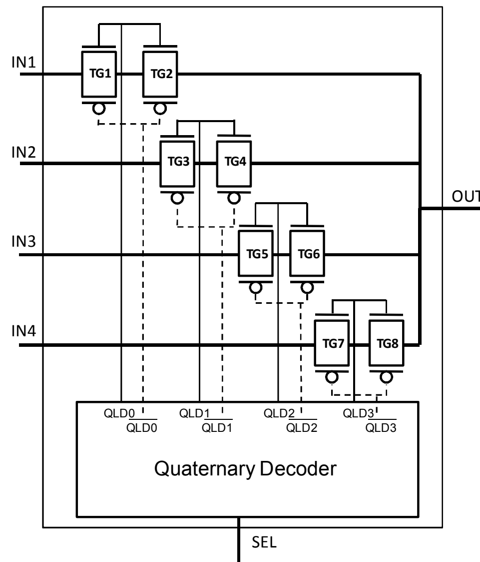
(Source: Created by the author)

As one can see from Figure 3.2 that the QLD circuit is illustrated inside the square while the detection circuit is outside of the same square and is constructed with 3 XOR gates.

The second vulnerability presented by the QLUT is caused by the lack of redundancy in two paths of the QLUT. One can easily correct this vulnerability by adding an extra transmission gate at the two paths that has only one TG. This way if one of the two transistors erroneously starts to conduct, the other transistor in the path will block this conduction, working as a failure avoiding mechanism.

After modifying the design of the traditional QLUT cell by adding error detection and avoiding mechanisms, Figure 3.3 presents the final design of the FT-QLUT cell.

Figure 3.3. Fault tolerant QLUT circuit



(Source: Created by the author)

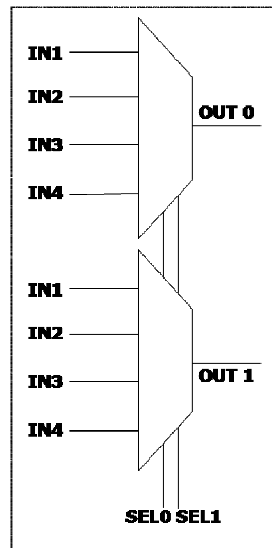
In order to evaluate the developed FT-QLUT, fault injection experiments were taken to evaluate and compare the proposed quaternary circuit with its binary correspondent. The next subsection presents these fault injection experiments.

3.1.3 Fault Injection Experiments

In the fault injection experiments the FT-QLUT circuit was described and simulated in the HSPICE tool (SYNOPSYS, 2009) with CMOS models for the 180nm technology node. In order to have the exact idea of the overheads of the proposed circuit, it was compared to the equivalent binary LUT with the well known DWC (duplicate with comparison) technique, which is equivalent to the proposed quaternary circuit and well known by the scientific community.

As it was already demonstrated mathematically in the previous chapter, in order to a binary LUT have the same selection capacity or the same number of functions to implement as the quaternary one, it is necessary that the binary circuit has the double of the number of inputs of the quaternary one. Also, as the binary LUT works with binary values, it is necessary the double number of binary LUTs to have the same output value as the quaternary one. Figure 3.4 presents the binary equivalent circuit to the 1 input quaternary LUT.

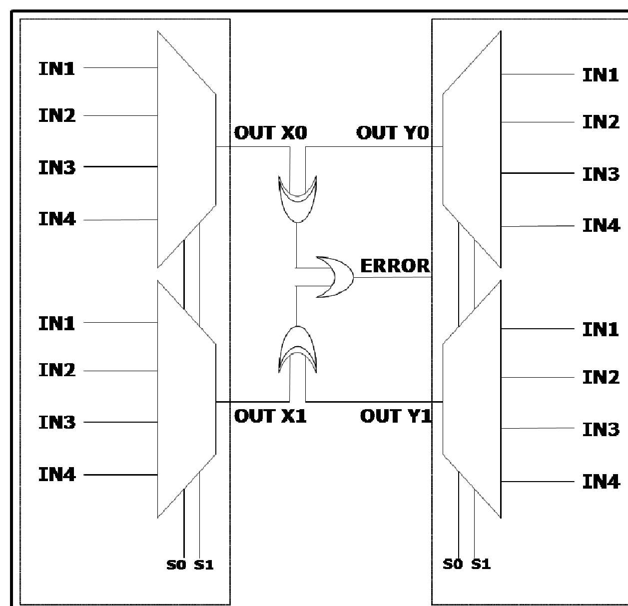
Figure 3.4. Binary LUT equivalent to the 1 input quaternary LUT



(Source: Created by the author)

The Duplicate With Comparison technique used to protect the binary circuit demands that the binary circuit be duplicated and have its duplicated answer compared. Figure 3.5 illustrates the resultant binary circuit that was described and simulated at the HSPICE simulator to compare with the proposed FT-QLUT.

Figure 3.5. Complete binary LUT with the DWC technique equivalent to the FT-QLUT of Figure 3.3.

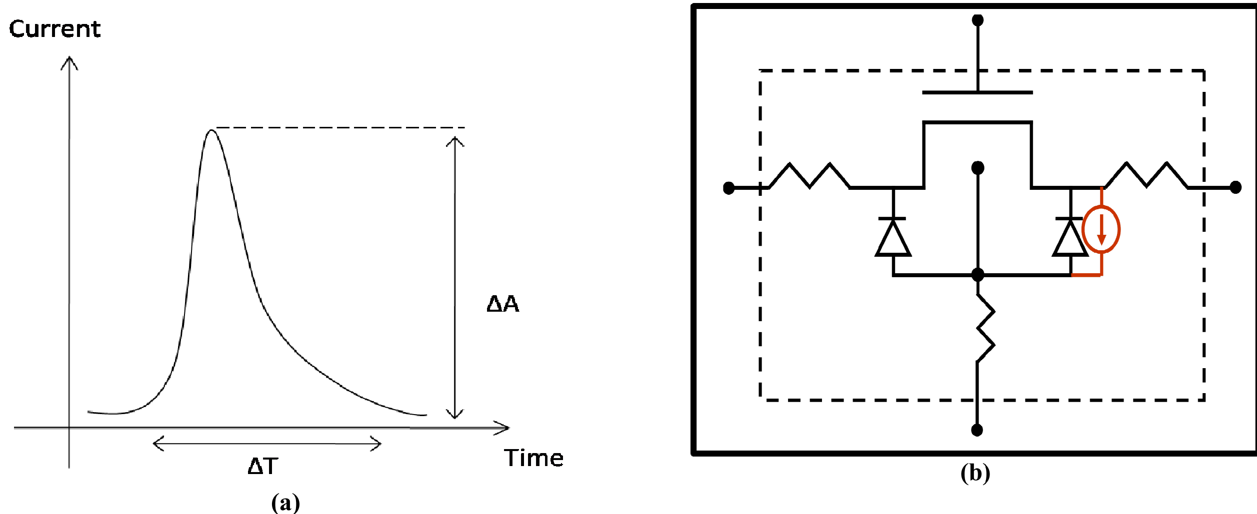


(Source: Created by the author)

In order to simulate the electrical charges injection that cause the SETs, sources of current in the form of double exponential waves were injected in all transistors of both circuits, the proposed

quaternary and the binary correspondent. Figure 3.6 presents the indication of the current source injection (a) as well as an illustration of the current source wave with the ΔA indicating the amplitude variation and the ΔT indicating the pulse width variation (b) that were used in the experiments.

Figure 3.6. Approximated waveform of the double exponential current source (a) and indication of the waveform injections in an NMOS transistor (b).

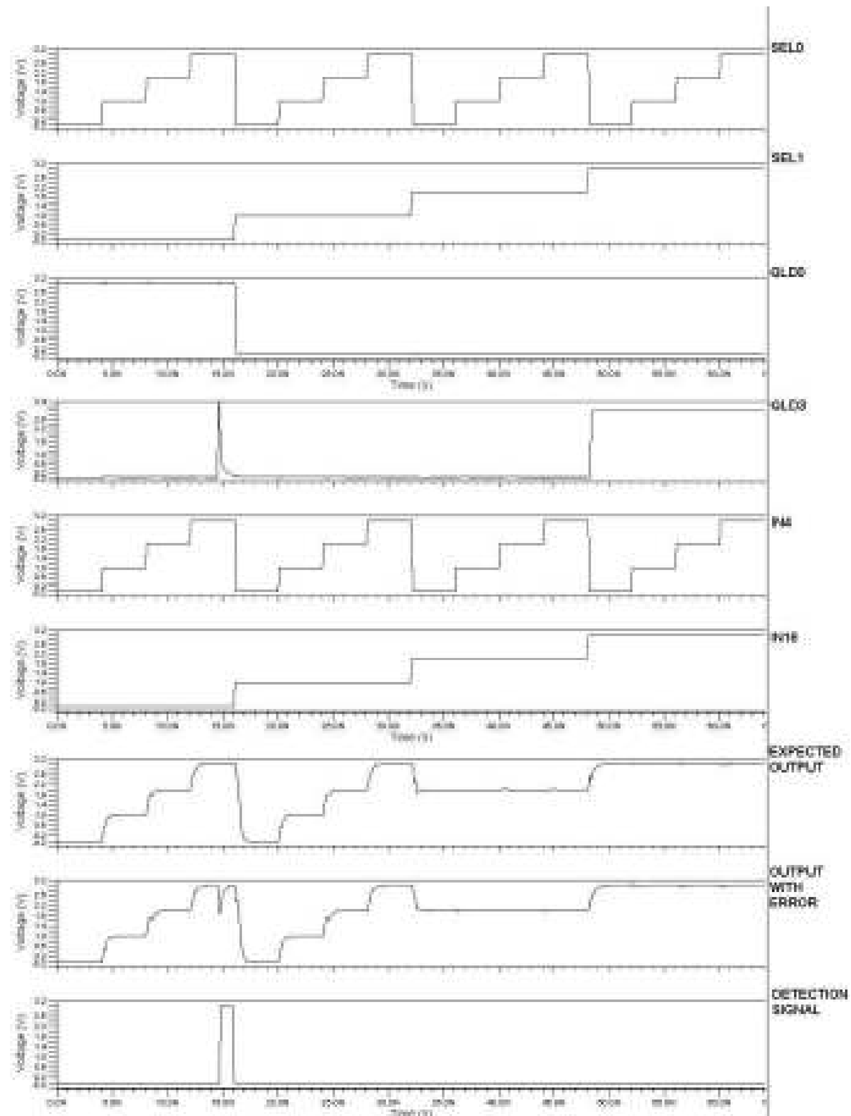


(Source: Created by the author)

During the fault injection experiments it was taken exhaustive simulations with pulses injected in all transistors with all combinations of possible values of inputs in both LUT circuits, the quaternary and the binary ones. The minimum charge of the source of current necessary to stimulate an SE was of 2.061 pC. After the injection of all faults, the simulation results showed that all faults were detected or avoided by both, the quaternary and the binary circuits.

Figure 3.7 presents an example of the waves generated an analyzed during the fault injection experiments in a 2 inputs QLUT. In Figure 3.7 one can see a case in which the event generated an error in the circuit. In this example it was chosen as fault injection target a transistor from the inverter of the QLD3 control signal. From the top down, one can see the two quaternary input signals (also called selection signals), followed by the QLD0 (with no errors), the QLD3 presenting the event, the IN4 signal, which is the correct input to be driven to the output in this example, the IN16, which is the wrong input selected by the event pulse, the expected output, the output with the error caused by the event and the error detection signal.

Figure 3.7. Waveforms of a fault injection simulation at the inverter of the QLD3 signal (Figure 3.2).



(Source: Created by the author)

Although the proposed FQ-QLUT detects and indicates whenever an error occurs in the circuit, in real life applications the ideal situation is to avoid all possible errors to happen or, when it is not feasible to avoid all errors, one needs to correct the error as soon as possible with minimum overhead.

In order to improve the fault tolerance of the FT-QLUT of Figure 3.3, this work proposes some additional circuit that gives the FT-QLUT the ability to avoid all events that would cause an error in the correct functionality of the QLUT. The next section presents the circuit modifications

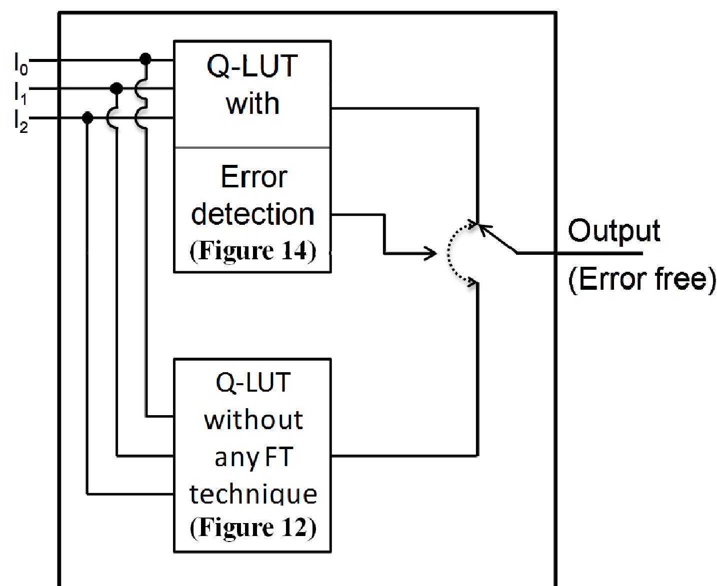
necessary to improve the FT-QLUT and gave origin to the Error Avoiding QLUT circuit (EA-QLUT).

3.1.4 FT-QLUT Improvement: The EA-QLUT

In order to be error free, the quaternary lookup table proposed in this section is able to detect and avoid errors. This error detection and avoiding scheme is done by the combination of two mechanisms. The first one duplicates a strategic portion of the circuit in order to avoid single errors and provide a redundant answer of the QLUT. The second mechanism is exactly the same circuit that detects errors in the decoder circuit of the QLUT that was previously presented at Figure 3.3. Whenever the error detection circuit detects an error, it automatically selects the redundant answer from the error free copy. This circuit is summarized at Figure 3.8.

The proposed EA-QLUT of this work is based in the work proposed in (RHOD, E. L.; CARRO, 2009) where the authors developed a quaternary lookup table that is able to detect or avoid the errors caused by SETs in all transistors of the circuit.

Figure 3.8. Error Avoiding Quaternary Lookup Table (EA-QLUT)



(Source: Created by the author)

One can see from Figure 3.8 that the proposed EA-QLUT is constructed with two types of QLUT, one is the error detection version presented in Figure 3.3 and the other is the QLUT of Figure 3.2 proposed in (CUNHA et al., 2007) that does not use any fault tolerant technique. Alone, both circuits have limited or no fault tolerance at all. The error detection version, as its name already says, has only capacity to detect errors but not to avoid them to happen. The second QLUT, the one with no FT capabilities, cannot neither detect nor correct any faulty event. This circuit's

innovation comes from the fact that by associating these two circuits with a simple on/off switch, which can be implemented with a couple of transmission gates, one obtains a circuit with the ability not only to detect, but also to avoid SETs that can compromise the correct functionality of the circuit. The EA-QLUT of this work works as follows. When the circuit is working without any errors being detected, the QLUT selected output is the one with the error detection version. Whenever an SET is observed by the error detection circuit, the output of the switch immediately to select the output from the circuit with no FT technique. This means that if an SET is being detected in the error detection circuit, the circuit with no FT is free of faults. Also, even if the switching mechanism is the target of a failure and erroneously switches from one circuit to the other, there will be no error in the answer because the fault will be in the switching circuit and not in the selected one. It is important to mention that this scenario is acceptable when one only considers single events happening at the same time. When dealing with multiple events at the same instant one needs to use different FT techniques to assure the circuit reliability, which are not in the scope of this work.

In order to validate the proposed EA-QLUT, the circuit of Figure 3.8 was described and simulated in HSPICE using as transistor models the ones obtained with the PTM web tool, which is a web tool created at the Arisona State University that can generate CMOS transistors models for different technology nodes and Vth parameters (CAO, 2007). The proposed EA-QLUT was compared with the binary correspondent circuit presented to the one presented in Figure 3.8, which was also described and simulated using the same technology at the same tool. As was already stated, the binary multiplexer can only produce binary values and therefore, as the binary multiplexer should have the same capacity as the quaternary multiplexer, one must use two binary multiplexers. After being duplicated, the binary LUT was hardened using the same technique as it was used with the quaternary circuit presented at Figure 3.8. One may question why the technique used in comparison was not the TMR, instead of using the same approach as the quaternary one. The reason of this choice will be explained and analyzed at Chapter 4.

The fault injection experiments that were taken to evaluate the proposed EA-QLUT and the binary correspondent followed the same strategy as explained in the previous section (section 3.1.3. After the extensive fault injection, all injected faults that caused errors were immediately corrected by the FT technique used in both circuits. As one can imagine, the voter in the TMR technique is not protected against SET and in the case of an event hit the voter, an uncorrectable error can occur

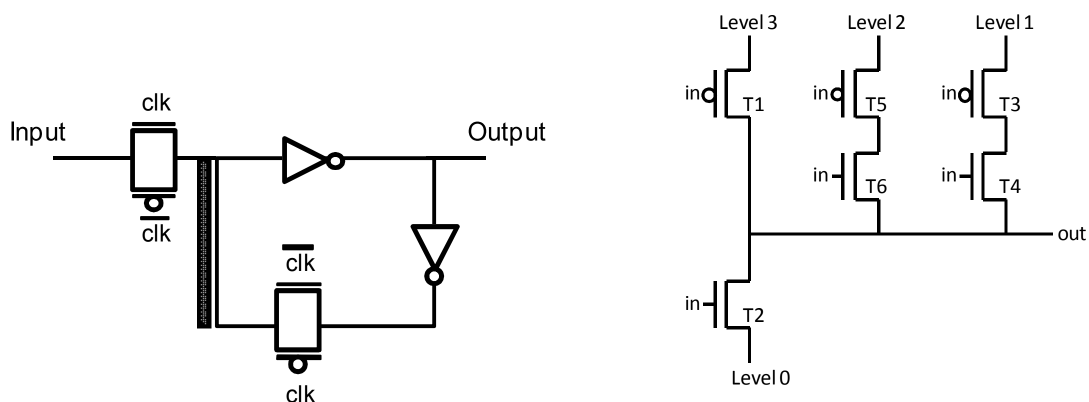
in the binary circuit. In order to protect the voter, the strategy known transistor sizing (hardened by design) was applied, which consist of increasing the size of the critical transistors of the circuit in order to increase the critical charge to higher levels than the SET can generate. As a drawback, the transistor sizing technique presented an increase in the total area of the binary version. These area results together with the other overhead figures of all the circuits proposed in this thesis will be presented and carefully analyzed in the next chapter (Chapter 4).

This subsection presented the EA-QLUT which is the main processing element of the objective of this work, which is the FT-QCLB. In order to a configurable logic block synthesize synchronous sequential circuits, it is mandatory the present of memory elements. The most common and elementary memory component is the *Flip-Flop*. The next section presets the development of the first voltage-mode fault tolerant quaternary flip-flop developed in the scope of this thesis.

3.2 The Fault Tolerant Quaternary Latch and Flip-Flop (FT-QL and FT-QFF)

The Flip-Flop presented in this work is constructed using two latches each one working with opposite active levels, also known as Master-Slave Flip-Flop. One way of constructing a Latch which has very low area is by using two inverters connected by a feedback loop like it is presented in Figure 3.9.

Figure 3.9. Quaternary latch and quaternary inverter



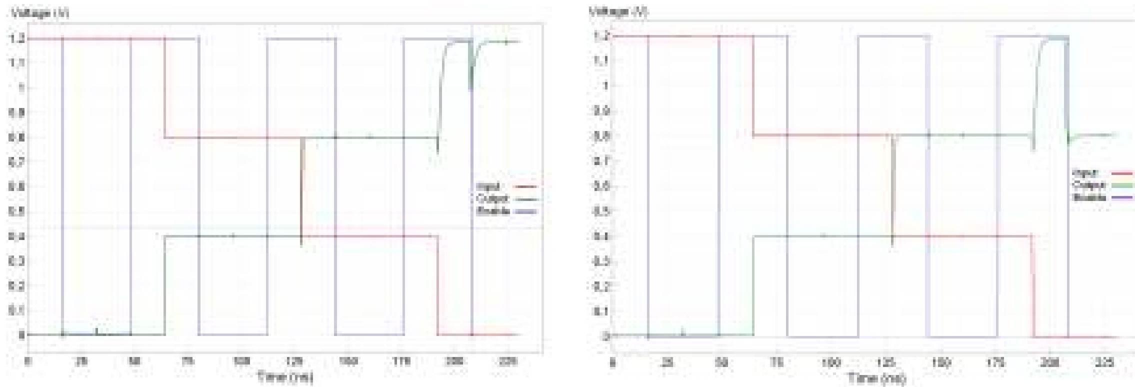
(Source: Created by the author)

The quaternary latch used in this work to construct the Flip-Flop is a memory element that is able to store information using a feedback system to keep the information level stable and is constructed basically by using two quaternary inverters, like the one presented in Figure 3.9, as one can see in Figure 3.9, and two transmission gates to control the input and to avoid short circuit. As one can see from the Figure 3.9, the quaternary latch holds the information through the inverters

feedback system. This way if a heavy energy particle strikes any of the components provoking an SET, depending on the SET characteristics, the inverters feedback can sustain the SET long enough to storage an erroneous information and transform the SET in an SEU.

In order to evaluate this faulty scenario and to be able to propose fault tolerance strategies to protect the circuit of the latch, extensive fault injection simulations were taken place in the quaternary latch circuit of Figure 3.9. The quaternary latch circuit was described and simulated using the HSPICE simulation tool using the transistor models obtained from the PTM web tool for 90nm and Vdd of 1.2V. To have a characterization as complete as possible the latch circuit was submitted to current sources in the form of a double exponential wave with three different intensities of amplitude (ΔA), each of this with 10 different variations of pulse length (ΔT). The ΔA values used in the experiments were of 10μA, 50μA and 100μA with ΔT of 200ps, 400ps, 600ps, 800ps, 1000ps, 1200ps, 1400ps, 1600ps, 1800ps and 2000ps. In order to consider the architectural vulnerability factor, all waves were injected in all transistors using as input the four possibilities of quaternary levels (0, 1, 2 and 3). Not only positive pulses like the one illustrated in figure 4 but also negative pulses were injected at the transistors of the circuit. At the end, each transistor was injected with 240 sources of current during the HSPICE simulation process. The fault injection process started by injecting all the 240 different waves in each of the two inverters of Figure 3.9. It is important to emphasize that all waves were injected at the time that the circuit was more susceptible to errors, which means, the time when the enable transmission gate from the input is not conducting anymore and feedback loop is active. In order to consider whether a pulse caused an error, the output value must be altered to an erroneous value and remain stabilized in that level by the latch. For instance Figure 3.10 (a) shows an example where the injected pulse with ΔA=10μA and ΔT=400ps does not generate and error while in Figure 3.10 (b) when a pulse with ΔA=10μA and ΔT=800ps is injected, the inverter has its output altered from level 3 to the level 2.

Figure 3.10. Example of a pulse that did not generate an error (a) and example of a pulse that generated an error (b)



(Source: Created by the author)

One can see from Figure 3.9 that the transistors T1, T2, T4 and T6 are all connected to the same node, which is the output node of the inverter circuit. This way, all injections with the same amplitude and length of pulse resulted in equal behavior at the inverter output. To simplify the results presentation, the percentage of the transistors T1, T2, T4 and T6 are presented at the same table, Table 3.2. Table 3.3 presents the percentage of errors observed with the transistor T3 as injection target while Table 3.4 presents the percentage of errors with the transistor T5 as target.

The second column of Table 3.2, Table 3.3 and Table 3.4 presents the injection of positive pulses at the inverter of number 1, while the third column shows the percentage of errors with positive pulses at the inverter of number 2. The fourth and fifth columns show the percentage of errors for the transistors of the inverter of number 1 and 2 respectively but with negative pulses being injected. The higher the percentage value presented in tables 3, 4 and 5, the smaller is the amplitude or width of the pulse necessary to provoke an error.

One can observe that in some cases none of the 240 injections provoked an error. Most of these cases can be explained because depending on the input stimulus, the injected pulse does not alter the output of the circuit. For instance, when looking at the second row and second column of table 3, the percentage presented is 0%. That happened because as the input stimulus is equal to the level 0, the logic of the inverter drives the output to the level 3, which is the highest level of the circuit and cannot be increased by the positive pulse. As the second inverter receives the input from the first inverter, the input stimulus that does not cause any error is the opposite as the one at the inverter 1.

Table 3.2. Percentage of Errors Observed with Injections in Transistors T1, T2, T4 and T6

Input Stimulus	Inverter1 x positive	Inverter2 x positive	Inverter1 x negative	Inverter2 x negative
Level 0	0%	50%	93.33%	0%
Level 1	56.67%	33.33%	60%	0%
Level 2	60%	33.33%	6.67%	50%
Level 3	83.33%	0%	0%	70%

(Source: Created by the author)

Table 3.3. Percentage of Errors Observed with Injections at Transistor T3

Input Stimulus	Inverter1 x positive	Inverter2 x positive	Inverter1 x negative	Inverter2 x negative
Level 0	0%	50%	0%	0%
Level 1	0%	0%	0%	0%
Level 2	0%	0%	3.33%	0%
Level 3	86.67%	0%	0%	0%

(Source: Created by the author)

Table 3.4. Percentage of Errors Observed with Injections at Transistor T5

Input Stimulus	Inverter1 x positive	Inverter2 x positive	Inverter1 x negative	Inverter2 x negative
Level 0	0%	50%	56.67%	0%
Level 1	0%	33.33%	60%	0%
Level 2	60%	0%	6.67%	50%
Level 3	83.33%	0%	0%	16.67%

(Source: Created by the author)

Table 3.5. Overall Percentage of Errors Observed in Each Transistor

Transistor	Inverter1 x positive	Inverter2 x positive	Inverter1 x negative	Inverter2 x negative
T1	50%	29.17%	40%	30%
T2	50%	29.17%	40%	30%
T3	21.67%	12.5%	0.83%	0%
T4	50%	29.17%	40%	30%
T5	35.83%	20.83%	30.83%	16.67%
T6	50%	29.17%	40%	30%

(Source: Created by the author)

One can also see that the second inverter has always a smaller percentage of errors. That can be explained by the fact that the second inverter has a bigger fanout, or in other words, drives its signal to a higher quantity of transistors than the first inverter, and therefore has a higher capacitance to be overcome by the injected pulse. It is important to mention here that the critical time of each transistor, i.e. the time that a specific transistor is more susceptible to generate an error when stimulated by an event, can vary significantly from transistor to transistor.

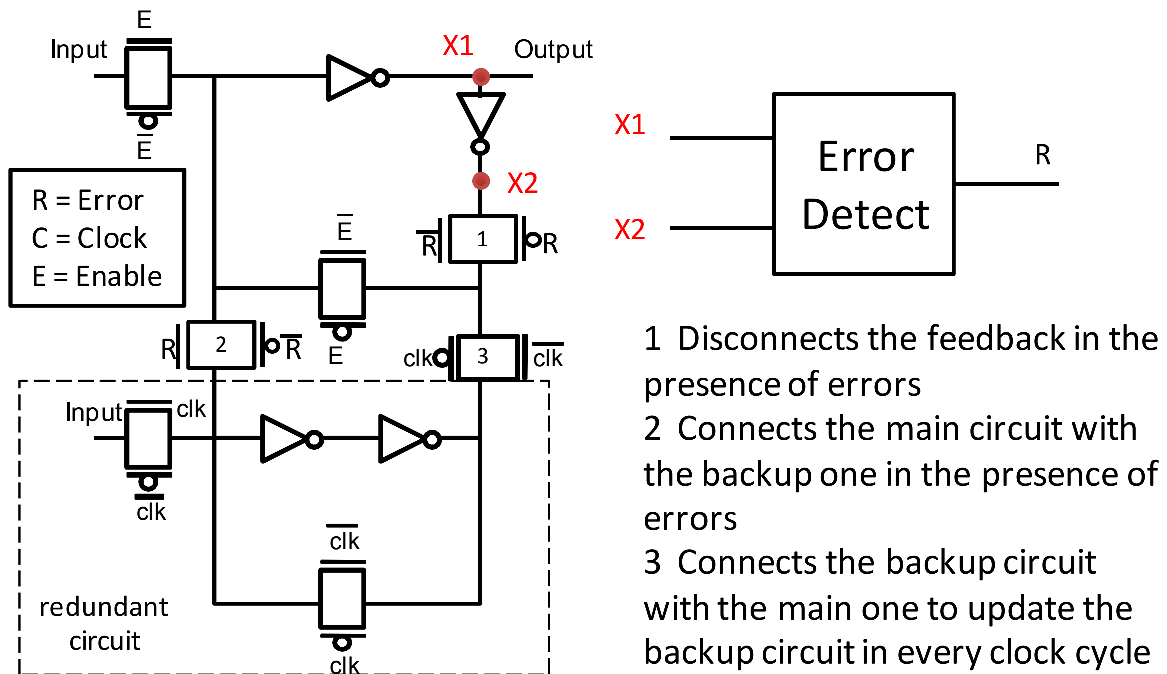
Table 3.5 presents the overall percentage of errors in all transistors. One can see from Table 3.5 that the transistors T3 and T5 have less percentage of errors due to the fact that they have another transistor in serial that can block the error to reach the output.

3.2.1 The Fault Tolerant Quaternary Latch (FT-QL)

In the previous section it was presented the quaternary latch characterization when submitted to several types of current sources used to emulate the hit of heavy particles. After the presented results, one can conclude that the quaternary latch can be very sensitive to SETs. One of the main reasons of the high quaternary sensitivity to SET is the low voltage distance between each adjacent logic level.

This subsection presents some circuit modifications in the traditional latch design together with the addition of an error detection circuit to cope with SETs of all intensities and length. The basic idea of the SET robustness comes from the use of a redundant latch circuit that stores a so called "redundant copy" of the main latch. This way if an error in the main circuit is detected, the redundant circuit updates the main circuit with the redundant and free if errors copy. To keep the redundant circuit always free of errors, the main circuit, when in the presence of no errors, updates the redundant circuit every clock cycle and whenever there is no new input value indicated by the enable signal. The Fault Tolerant Latch and its simplified functionality are illustrated in Figure 3.11.

Figure 3.11. Fault Tolerant Quaternary Latch.



(Source: Created by the author)

In Figure 3.11, the redundant circuit is indicated by the dashed rectangle and in case the main circuit is submitted to any kind of event that can cause an error, the error detection circuit turns off the transmission gate 1 and turns on the transmission gate 2 to disconnect the main circuit feedback and connect the redundant circuit to the main one. This way, the redundant circuit restores the correct level in the main circuit. In order to keep the redundant copy always updated, the transmission gate with the number 3 connects the main circuit to the redundant one in every clock cycle when there is no error. This update mechanism works perfectly in the presence of single events in the same clock cycle.

The error detection circuit works by analyzing the outputs of the inverters in the main latch circuit. If both values respects the inverter truth table presented in Figure 3.9, this indicates that the main circuit is free of errors. The complete error correction latch design is presented in Figure 3.11 and the error detection circuit design is illustrated in the Figure 3.12.

Figure 3.12. FT-Q-Latch Error detection circuit

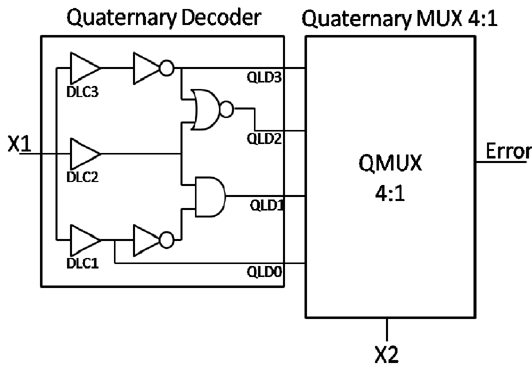


Table 3.6. Quaternary Decoder Truth Table

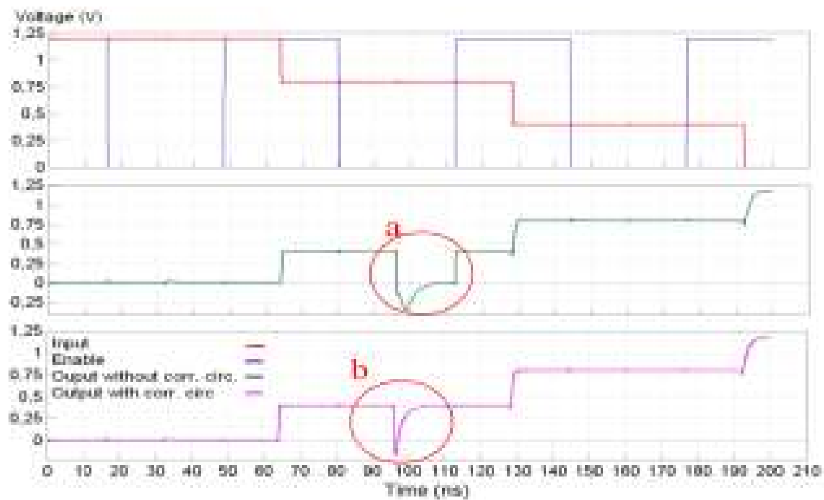
In (logic level)	QLD0 (logic level)	QLD1 (logic level)	QLD2 (logic level)	QLD3 (logic level)
0	3	0	0	0
1	0	3	0	0
2	0	0	3	0
3	0	0	0	3

(Source: Created by the author)

As we can see from Figure 3.12, the error detection circuit is composed of a quaternary decoder circuit and a quaternary 4:1 multiplexer like the one illustrated in Figure 3.1. The quaternary decoder circuit, as its name says, decodes the quaternary input in 4 binary signals according to the truth table presented in Table 3.6.

In order to test the error detection and correction circuit, all HSPICE simulations with the fault injection scheme presented in section 3.1.3 were repeated another time with the latch protected with the error detection circuit of Figure 3.12. The experiments showed that the combination of the error detection with the redundant circuit to correct the main one prevent all pulses from becoming a permanent error in the latch. Figure 3.13. **Error detection and correction circuit correcting a SET.** illustrates the error detection and correction circuit working.

Figure 3.13. Error detection and correction circuit correcting a SET.



(Source: Created by the author)

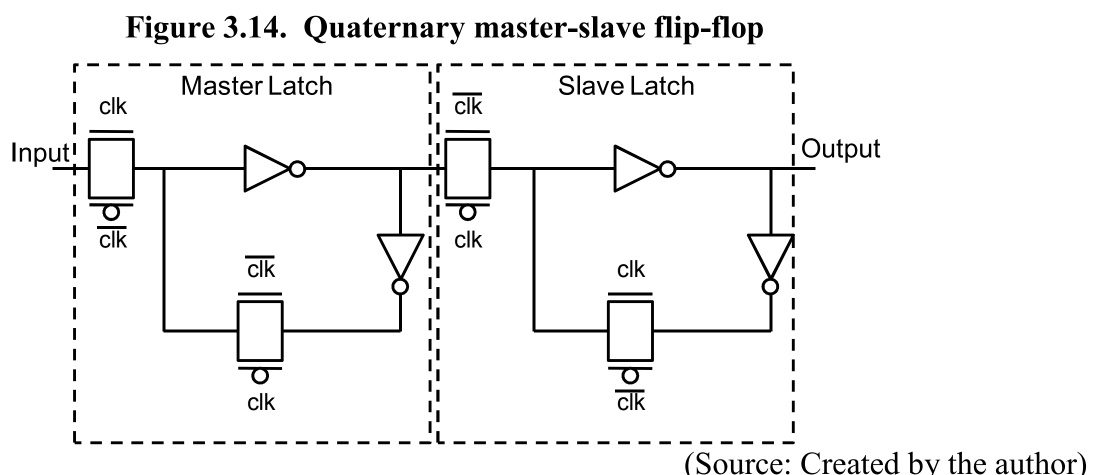
Figure 3.13 illustrates a certain pulse injected into the latch circuit of Figure 3.9 that causes a single event to be captured as an error by the latch (indicated by the "a" in Figure 3.13). This error is sustained by the feedback connection of the latch until a new value arrives and the enable signal is activated. When simulating the same event at the latch circuit with the correction mechanism (presented in Figure 3.11), one can see from Figure 3.13 indicated by the "b" circle that the event do not become an error because the correction mechanism opens the feedback connection of the latch and the redundant circuit corrects the main one. Even if the redundant circuit is the target of any event, the main circuit corrects the redundant one every clock cycle.

After studying the vulnerabilities of the latch and proposing the FT-QL circuit, the next subsection presents the FT-QFF developed using the FT-QL presented in this subsection (Figure 3.11).

3.2.2 The Fault Tolerant Quaternary Flip-Flop (FT-QFF)

The FT-QFF of this work is constructed by using the traditional master/slave method, using two latches connected in sequence, one working as the master and the other as the slave. Each latch, as it was already presented, is constructed with the quaternary inverters that were presented in a previous section (Figure 3.9) and some transmission gates. Figure 3.14 presents the QFF design.

Figure 3.14. Quaternary master-slave flip-flop



(Source: Created by the author)

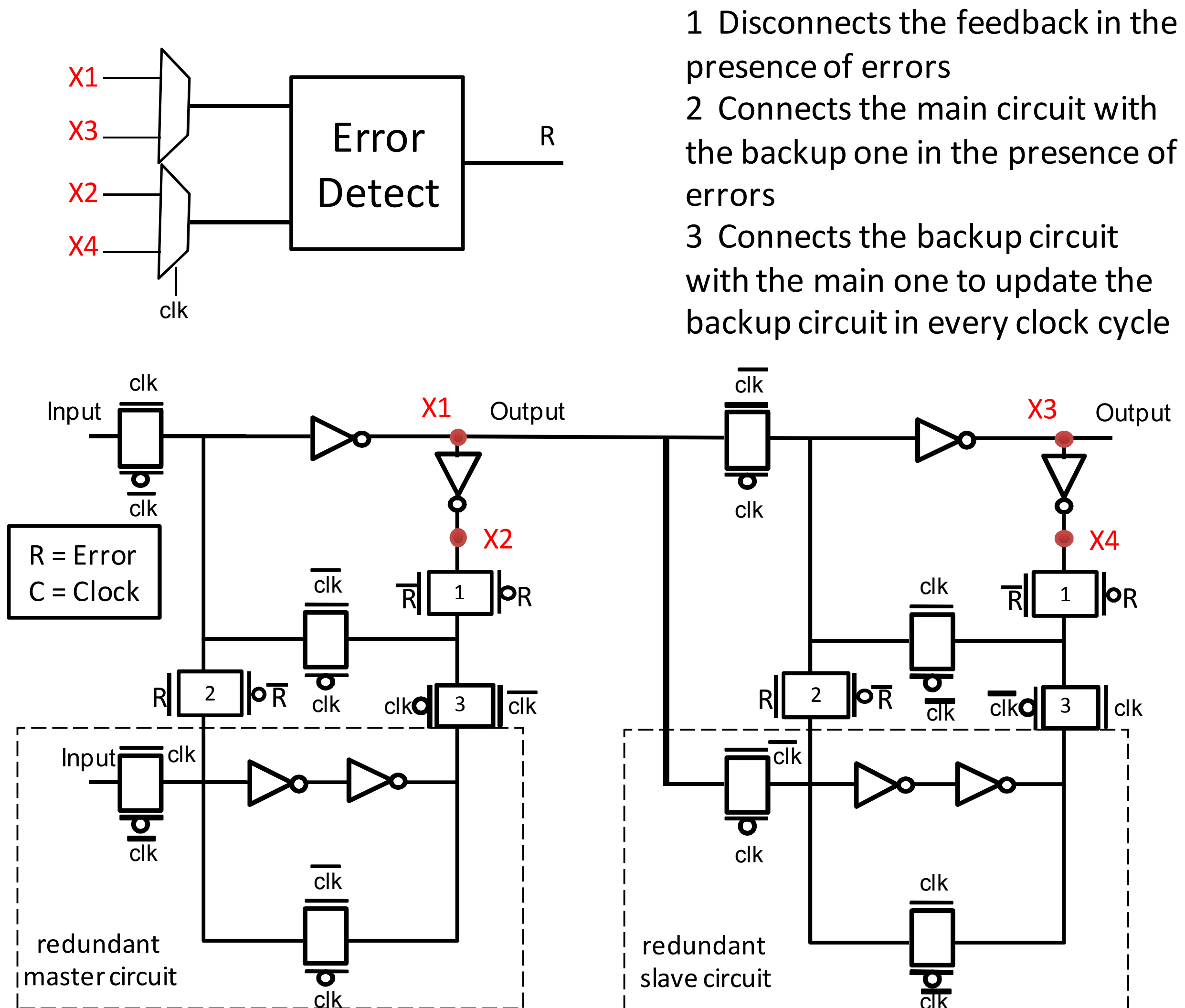
One can see from Figure 3.14 that the proposed QFF works as the traditional master-slave flip-flop controlled by the clock signal. In the circuit presented in Figure 3.14 the "master" passes

the stored quaternary signal to the "slave' in the falling edge of the clock. The FF fault detection and correction design is based in the work that proposes a fault tolerant quaternary latch (FT-QL) (RHOD, E.; STERPONE; CARRO, 2010), presented at Figure 3.11. In that work, the so called FT-QL uses a detection mechanism that compares the output of both inverters of the latch and, in case of an incorrect value caused by an SET, a golden copy of the correct value is used to correct the erroneous one. To keep the golden copy always free of errors, whenever the main circuit is error free, the clock signal updates the golden copy with the main circuit's error free value.

The FT-QFF of this work uses the same strategy as the FT-QL with only a small but significant difference. 0 presents the fault detection and correction circuit used in the FT-QFF.

One can see that the circuit presented in Figure 3.15 is almost the same as 2 FT-QLs. The only difference is that the FT-QFF uses only one error detection circuit instead of two, as it would be expected. This can be done because the error detection circuit only needs to detect the error at the time that the circuit is vulnerable. As the proposed Flip-Flop is constructed with latches working in the master/slave methodology and, in this condition, each latch is vulnerable only during 50% of the time, which is the time that the feedback loop is closed. This way, both latches can be monitored using only one error detection circuit. The only control circuit necessary to make the error detection circuit work is two multiplexers that selects whenever the master or the slave latch is active.

Figure 3.15. the fault detection and correction of used in the FT-QFF



(Source: Created by the author)

In order to evaluate the FT-QFF the circuit of Figure 3.15 was described and simulated using the HSPICE simulation tool with the transistor models obtained from the PTM web tool for 90nm and Vdd of 1.2V. As it was done in the Latch characterization, the FT-QFF circuit was submitted to current sources in the form of a double exponential wave with three different intensities of amplitude (ΔA), each of this with 10 different variations of pulse length (ΔT). As expected, all events that had enough energy to cause an SEU were immediately detected by the error detection circuit and corrected by the redundant circuit.

The next subsection presents the variability analyses that were taken in order to evaluate the impact of the Vt variations for the quaternary circuits.

3.3 Variability Impact Analysis

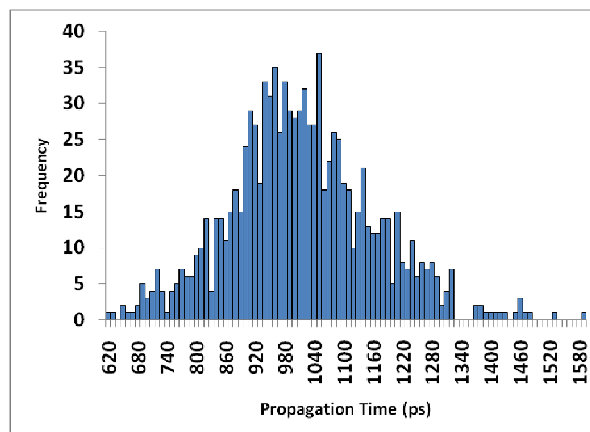
In this work the variability effects in the Vt of the proposed quaternary circuits are evaluated using Monte Carlo simulation with the HSPICE simulator from Synopsys. In order to evaluate how the proposed design will behave with the process variability, the QLUT design was described and simulated using Monte Carlo simulation with the HSPICE simulator. It was used the PTM (Predictive Technology Model) for 90nm technology with power supply of 1.2V to evaluate how the scaling affects our design.

As our quaternary circuits need multiple Vts (a total of 6), and as there is no known and available process from with the information about the standard deviation Vt, σVt, this work used the Predictive Technology Models to generate the models for each transistor and to apply at each model the equation (8) from (BERNSTEIN, 2006).

$$\sigma Vt = 3.19\,x\,10^{-8} \cdot \frac{t_{(ox)} \cdot N_A^{0.4}}{\sqrt{(L_{eff} \cdot W_{eff})}} \quad (8)$$

Using the equation (8), it is possible to estimate the σVt values for each transistor and use them in the Monte Carlo experiments. The Monte Carlo simulation were taken for 1,000 steps and at all cases for all technology nodes the proposed quaternary circuits produced no errors with just an acceptable performance degradation. As one can see in Figure 3.16, in 90% of the circuits have propagation time from 770 ps up to 1.25 ns.

Figure 3.16. Histogram for the 2QLUT in 90nm technology.



(Source: Created by the author)

The simulations result presented in this section depends heavily on the transistors models that were used during the simulation. It is important to emphasize that the real variability figures

depend on the process that are used in the IC fabrication and, in order to obtain a certain process variability, one would need the transistor models of the corresponding process. In the case of this thesis, as there are no available models to be used, it was used the PTM models already referred previously. One can conclude that the variability results shown here are just for the sake of illustration and evaluation of the predictive models that were used in this work. They cannot assure that quaternary circuits can be successfully fabricated.

This chapter presented the fault tolerant quaternary circuits that were specially designed for the Fault Tolerant Quaternary CLB, the EA-QLUT and the FT-QFF. Each of the proposed fault tolerant technique was selected in order to avoid or correct the all the possible faults. In order to have the exact figures of each developed circuit here presented, the next chapter presents the overhead results in terms of area, performance and power dissipation of the proposed quaternary circuits in comparison to the binary equivalent version.

4 AREA, PERFORMANCE AND POWER DISSIPATION OVERHEADS

In this chapter it is presented the overhead results of the proposed fault tolerant quaternary circuits that were presented in the previous chapter. When it comes to fault tolerance circuits, one must always evaluate all the drawbacks that a specific fault tolerant solution have. The expression "There is no free lunch" fits perfectly to the fault tolerance area and explains that when one is proposing any tolerant technique to a certain circuit, there will be a price to be paid. For the circuits proposed in this work, the price paid for the addition of the fault tolerance is in the form of extra area, extra performance, extra power dissipation, or even a combination of more than one of those. In order for a fault tolerant technique to be considered efficient, it must have the same or even more fault tolerance, with less overhead in comparison to the techniques that have already been proposed.

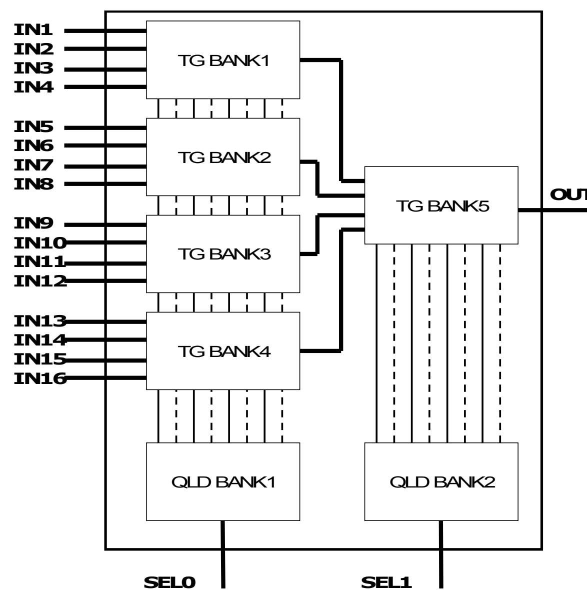
This chapter presents the overheads of the proposed quaternary circuits in comparison to its binary equivalent. The next section presents the Fault Tolerant Quaternary LUT overheads that are explained in details.

4.1 FT-QLUT and EA-QLUT Overheads

As explained in a previous section, a LUT can be easily constructed with a multiplexer and a memory element. By associating multiples quaternary one input QLUTs, one can construct a 2 input QLUT, as shown in Figure 4.1. In order to construct the corresponding binary LUT to the 2 inputs QLUT presented in Figure 4.1, one would need a 16:1 binary multiplexer, and to have the same fault correction capability as the quaternary version implemented using the technique presented in Figure 3.8, one would need to triplicate this binary area (using the TMR technique). As one increases by one the number of inputs in a QLUT, the corresponding binary LUT has the number of input increased by 2. Besides that, the number of the internal transmission gates (TGs) for the binary LUT grows equally in the quaternary version. When comparing both the quaternary and the

binary equivalent circuits, an interesting data is the percentage of the decoder circuit importance, which in the quaternary version with 1 input is very big, reaching 50% of the total area. As the number of input grows, this contribution grows almost arithmetically for the quaternary version. In the binary one, the area contribution of the decoder circuit starts small with 2 binary inputs (25%) but grows geometrically (times 2) with the increase of the input number. These results can be observed in Table 4.1.

Figure 4.1. 2QLUT: 2 input QLUT.



(Source: Created by the author)

In Table 4.1 one can see that the QLUT has 33.33% less transistors than the binary LUT with one input, 61.90% less transistors with 2 inputs and 83.33% less transistors with 3 inputs. Clearly, the more complex the circuit, the more area gain can be obtained by the usage of QLUTs. In Table 4.2 the area results for the QLUT with fault detection is compared with the corresponding binary LUT with the DWC technique.

Table 4.1. Area results comparing the QLUT with the binary one without the fault detection circuits

# of Quaternary Inputs	[1] - # of Transistors in the QLUT. (TGs + decoder)	[2] # of Transistors in the binary LUT (TGs + decoder)	Ratio % ([2]/[1] -1)
1	(12 + 12) = 24	(24 + 8) = 32	33.33
2	(60 + 24) = 84	(120 + 16) = 136	61.90
3	(252 + 36) = 288	(504 + 32) = 528	83.33

(Source: Created by the author)

In Table 4.2 one can see that the proposed error correction technique is more efficient when comparing to the binary correspondent approach using the TMR technique. One can see from the Table 4.2 that the difference between the area of the proposed QLUT and the binary correspondent starts from 48.39% less area with one input and reaches up to 139.05% less area with 3 inputs.

Table 4.2. Area results comparing the FT-QLUT with the Binary LUT using the DWC technique

# of Quaternary Inputs	# of Transistors in the QLUT [1]	# of Transistors in the binary LUT [2]	Ratio % ([2]/[1] -1)
1	62	92	48,39
2	178	340	91,01
3	486	1164	139,05

(Source: Created by the author)

In order to evaluate the propagation time of the proposed cell, both the quaternary and the binary error correcting circuits were described using the PTM (Predictive Technology Models) and simulated using the HSPICE simulator. In Table 4.3 the proposed error correcting QLUT from figure 3 is compared in terms of propagation time with its binary version using the TMR technique. The QLUT with 1 input is compared with the binary LUT with 2 inputs and the results show that the binary version is more than 1.8 times faster than the quaternary version. That happened because the proposed quaternary LUT has the QLD bank, which has a several times bigger propagation path than the binary one (compounded by just a few transmission gates). When one compares the propagation time for the 3 input QLUT and its binary correspondent, one can see that the difference is increased to 2.28 times faster. That can be explained by the reduction in the contribution of the voter. In the smaller LUT the voter has greater contribution to the total propagation time than in the bigger LUT.

In Table 4.3 the QLUT is compared with its binary correspondent for 1, 2 and 3 inputs quaternary inputs.

Table 4.3. Propagation time for different sizes of the proposed QLUT and its binary correspondent.

# of Quaternary Inputs	QLUT propagation time (in ps) [1]	Binary LUT propagation time (in ps) [2]	Ratio ([1]/[2]-1)
1	411	71	4.78
2	1050	279	2.76
3	2299	625	2.67

(Source: Created by the author)

One can see that the QLUT is more than 4.7 times slower than the binary one. That happened because the proposed quaternary LUT has the QLD bank, which has a several times bigger propagation path than the binary one (compounded by just a few transmission gates). When one compares the propagation time for the 3QLUT and its binary correspondent, one can see that the difference is reduced to 2.6 times faster. That happened because the number of transmission gates in the propagation path of the QLUT grows 2 times slower than the binary one. One can understand that difference by noticing that from the QLUT to the 3QLUT the number of TGBANK chain in the propagation path is 3 while from the 2LUT to the 6LUT the number of TGBANK chain in the propagation path is 6. On the other hand, the proposed QLUT offers the possibility to bigger reductions in the propagation time by adjusting the size of the transistors, as the results presented in Table 4.3 were obtained using transistor with the minimum size. For the sake of illustration, in Table 4.4 one can see the reduction in the propagation time and the area overhead that one can have to reduce propagation time, trading area and power for speed.

Table 4.4. Propagation time (in ps) for different sizes of the proposed QLUT and its binary correspondent obtained by increasing transistor size.

	QLUT x 2LUT	2QLUT x 4LUT	3QLUT x 6LUT
Propagation Time after resizing	330 x 71	812 x 268	1607 x 588
% of reduction	20 x 0	23 x 4	30 x 6
% area overhead	12.1 x 5.7	8.6 x 2.9	4.8 x 1.1
Final area	74 x 74	202 x 286	522 x 1074

(Source: Created by the author)

The results presented in Table 4.4 where obtained by increasing the transistor size of the critical areas of both the binary and the quaternary versions of LUT. We can see from the same table that the difference between the propagation time of the 3QLUT and the 6LUT has decreased from 2.6 times (2299/625) to 1.7 times (1607/588), which means a reduction of 90%, with and area increase of only 4.8%. These results show that by small changes in the size of the QLUT transistors, one can reduce the propagation time of the proposed QLUT.

In order to evaluate the power dissipation of the proposed FT-QLUT and its binary correspondent, simulations using the HSPICE tool were taken with all possible input variations with the same frequency of operation (100 MHz) in order to normalize the power results. The obtained results are presented in Table 4.5 for different sizes of the FT-QLUT and the correspondent binary version protected with the DWC technique.

Table 4.5. Power Dissipation (in µW) for Different sizes of the FT-QLUT and the Binary Correspondent

# of Quaternary Inputs	FT-QLUT [1]	Binary LUT with DWC [2]	Ratio % ([2]/[1] -1)
1	0.74	1.33	79.19
2	2.58	8.54	231.00
3	8.78	37.04	321.87

(Source: Created by the author)

One can see from table Table 4.5 that the proposed FT-QLUT dissipates from 79%, for the smaller version, up to 322% less power for the bigger one when comparing to its binary correspondent.

When it comes to the EA-QLUT, the same experiments were taken to evaluate the area, propagation time and power dissipation results as for the FT-QLUT. The area results are presented at Table 4.6 with the two possible techniques that one could use to protect the binary circuit.

Table 4.6. Area Results of the EA-QLUT and the Two Binary Correspondents

# of Quaternary Inputs	EA-QLUT [1]	Binary with TMR	Binary with same as EA-QLUT [2]	Ratio % ([2]/[1] -1)
1	92	172	130	41.30
2	268	484	442	64.92
3	780	1660	1618	107.44

(Source: Created by the author)

One can see from Table 4.6 that binary circuit protected with the TMR technique (3rd column) has bigger area than the binary protected with the same technique as the quaternary one (4th column). That happened because the detection circuit used at the DWC technique and the switch area are smaller than the voter circuit of the TMR technique. It is important to mention that the TMR voter here considered had its transistor size duplicated in order to increase its critic charge (Qcritic) and tolerate SETs.

The propagation time results are presented at Table 4.7.

Table 4.7. Propagation time (in ps) for different sizes of the proposed EA-QLUT and its binary correspondent protected with the TMR technique.

# of Quaternary Inputs	Quaternary LUT [1]	Binary LUT [2]	Ratio % (1-[1]/[2])
1	417	147	183.67
2	1057	356	196.91
3	2306	701	228.96

(Source: Created by the author)

It is important to mention here that the all simulation that were taken to evaluate the propagation time of the EA-QLUT presented at Table 4.7 were obtained using transistors with the minimum size. Experimental results presented at Table 4.4 indicates that if one increase the size of the transistors used to construct the QLD cell, one can have a significant reduce in the propagation time of the QLUT version with a small area overhead. Also the mean power dissipation for different sizes of the QLUT and its binary correspondent were evaluated in the 90nm technology using the PTM models. Table 4.8 summarized the mean power dissipation results obtained for different sizes of the Error Avoiding QLUT and its binary correspondent implemented with the TMR technique.

Table 4.8. Power dissipation (in μW) for different sizes of the proposed EA-QLUT and its binary correspondent.

# of Quaternary Inputs	Quaternary LUT [1]	Binary LUT [2]	Ratio % (1-[2]/[1])
1	1.49	3.05	104.70
2	5.24	16.14	208.02
3	17.83	61.86	246.94

(Source: Created by the author)

From Table 4.8 one can see that the quaternary solution dissipates from 1.04 to 2.46 less power than its binary version, from the smallest to the biggest sizes respectively. These differences can be explained to the less quantity of transistors that the quaternary version has, but mainly to the fact that as the quaternary circuits work with 4 different levels of VCC, to go from one level to the next does not require a full charge or discharge of the capacitor. For instance to go from the level 0 to the level 1, the quaternary circuits charge 1/3 of VCC while the binary version charges all VCC value.

4.2 FT-QL and FT-QFF Area Results, Analysis and Discussion

The area overhead of the proposed Fault Tolerant Quaternary Latch (FT-QL) is presented in Table 4.9.

In Table 4.9, one can see at the second column the area results of the binary version latch circuit. The third column presents the number of transistors necessary to implement the same binary circuit equivalent to the quaternary one. As the binary latch can only store 2 values, ‘0’ or ‘1’, it is necessary to have two binary latches combined in order to have the same storage capacity as the

quaternary circuit (presented at the 4th column). The fifth column presents the difference ratio between the binary equivalent circuit (presented at the 3rd column) and quaternary one (presented at the 4th column). The third row presents the area results with the fault tolerant circuit. One can see that the quaternary circuit illustrated at Figure 3.8 has 22% fewer transistors than the binary circuit protected with the TMR technique. That happened because the detection and correction circuit proposed in this work adds less proportional overhead than the TMR technique. This ratio can be observed at the 4th row of Table 4.9. The main responsible for the high overhead of the TMR technique is the voter. One can also see at the FT Overhead row that the impact of the fault tolerance technique was bigger for the quaternary than in the binary solution. That happened mainly because the error detection circuit costs more than 50% of the total area. It is important to mention that even if the overhead was bigger, the final area is what really counts as the cost of the solution. As it was mentioned previously, the quaternary solution presented 22% less total area in comparison to the binary counterpart.

Table 4.9. Area Results in Number of Transistors

	Binary version	Binary version equivalent	Quaternary	Diff. Ratio
Area without fault tolerant circuit	10	10 x 2 = 20	16	25%
Area with fault tolerant circuit	TMR + voter 3 x 10 + 20 = 50	2 x (TMR + voter) 50 x 2 = 100	Error Detec. and Correct. 44 + 38 = 82	22%
FT Overhead	400%	400%	412%	-12%

(Source: Created by the author)

When it comes to evaluate the performance overhead, the proposed circuit has only one extra transmission gate at the feedback path of the circuit as de decoder works in parallel and does not affect the delay of the latch. This way, the delay added by one transmission gate is negligible when compared to the voter overhead of the TMR technique.

For the FT-QFF the area results are presented in Table 4.10.

One can see from Table 4.10 that the FT-QFF has 38% fewer transistors than the binary version protected with the TMR technique. As it was expected the FT-QFF solution has even less transistors in comparison to the binary version than the FT-QL. That happened because the FT-QFF the overhead of the FT-QFF is 294% in comparison to 412% of the FT-QL. As it was already explained before, this difference comes from the percentage of the error detection circuit of the FT-QFF, which is 42% of the total area in comparison to 54% of the FT-QL.

Table 4.10. Area Results for the FT-QFF in Number of Transistors

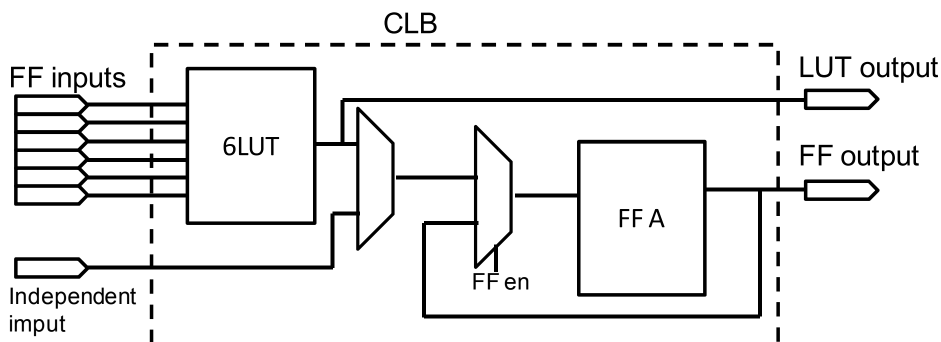
	Binary version	Binary version equivalent	Quaternary	Diff. Ratio
Area without fault tolerant circuit	20	20 x 2 = 40	32	25 %
Area with fault tolerant circuit	TMR + voter 3 x 20 + 20 = 80	2 x (TMR + voter) 2 x 80 = 160	Error Detec. and Correct. 54 + 72 = 126	38%
FT Overhead	300%	300%	294%	6%

(Source: Created by the author)

5 THE FAULT TOLERANT QUATERNARY CONFIGURABLE LOGIC BLOCK (FT-QCLB)

Basically a CLB is composed by 3 components: the Lookup table, responsible for the logic processing; a flip-flop, responsible for the data storage, important in the synthesis of sequential logic, and few multiplexers that give flexibility to the CLB cell by performing the signal routing. These components are illustrated in Figure 5.1.

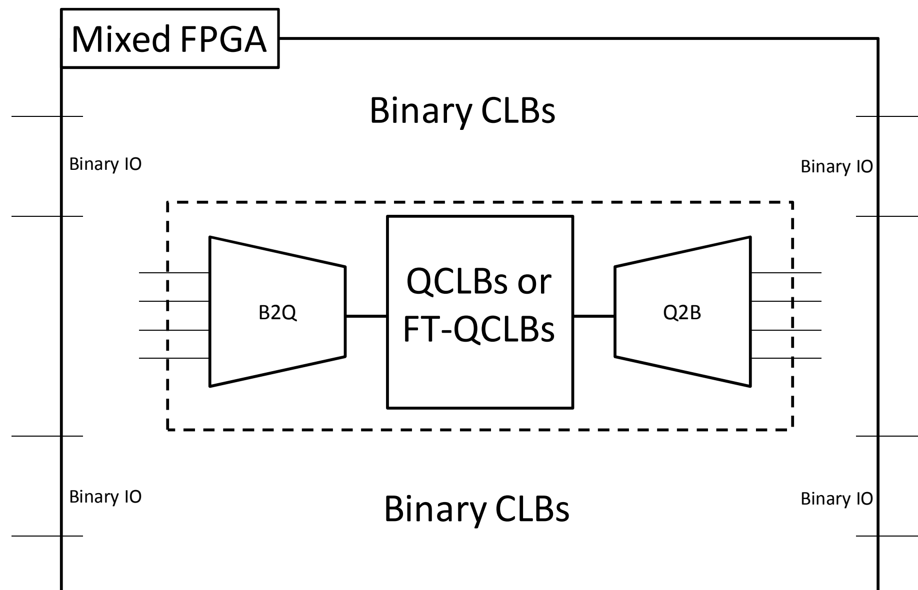
Figure 5.1. Standard Binary CLB composition



(Source: Created by the author)

In Figure 5.1 one can see the CLB cell that is used in this work to construct both the proposed quaternary CLB and the binary one that will be used to evaluate the proposed quaternary circuit. In order to have fault correction capability, each component of the FT-QCLB here proposed uses a different strategy to detect and correct faults that were presented in Chapter 3. Also it is important to mention here that the FT-CLB here proposed can be used to construct a complete quaternary FPGA or even a mixed quaternary and binary FPGA. In case of a mixed binary and quaternary CLB composition, in order to the quaternary signals coexist with binary ones it is necessary some coding and decoding circuits from quaternary to binary and vice-versa. Figure 5.2 illustrates the mixed quaternary and binary FPGA idea.

Figure 5.2. Mixed Binary and Quaternary FPGA

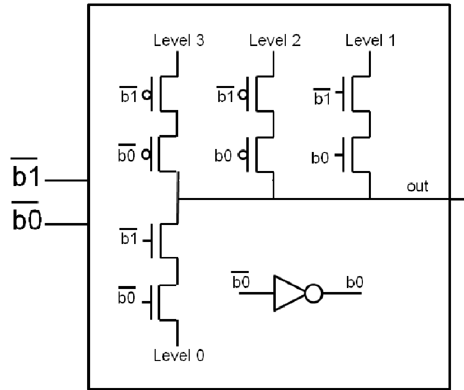


(Source: Created by the author)

One can see from the illustration of Figure 5.2 that the mixed FPGA is can be composed by a major binary area and some quaternary logic area which can be fault tolerant by using the FT-QCLB here proposed, or simply by quaternary CLBs with no FT technique, which can also be constructed with the circuits presented in this work. As it was mentioned before, in order to the multiple base signals coexists it is necessary some binary to quaternary decoder (called B2Q in Figure 5.3) and quaternary to binary decoder as well (called Q2B in Figure 5.4).

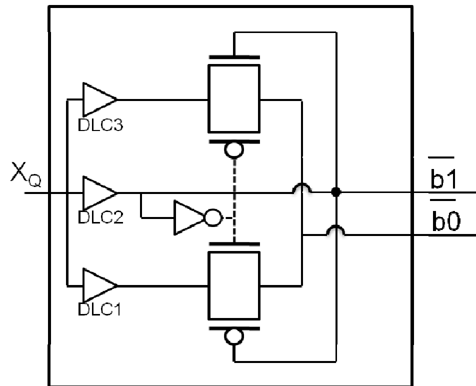
If one wants to invert the most and the least significant component of the quaternary exit signal one just need to invert the two quaternary inputs. In order to better understand how each projection component works, Figure 5.3 and Figure 5.4 presets the B2Q and the Q2B components respectively.

Figure 5.3. Binary to Quaternary circuit decoder



(Source: Created by the author)

Figure 5.4. Quaternary to Binary circuit decoder



(Source: Created by the author)

In some cases the use of traditional FT techniques, like for instance, the triple modular redundancy (TMR), represents an overhead in terms of area, or delay or in some cases, the addition of more sensitive area than before the use of the FT technique. This situation would happen, for example, if one would try to protect a small circuit like the multiplexers of this work using the TMR technique. In this particular case, the voter, which in the traditional TMR technique is sensitive to faults, would be bigger than the multiplexers circuit and therefore, the use of the TMR technique would imply in adding more unprotected area than before the technique was applied. In cases like this, the use of simple techniques like increasing the size of the transistors in the circuit would result in an increase in the value of the $Q_{critical}$ necessary to cause an SET in that component with a low increase in the total area.

5.1 Fault Injection Experiments

To have a characterization as complete as possible the FT-CLB circuit was submitted to current sources in the simulation form of a double exponential wave with 3 different intensities of amplitude, each of this with 10 different variations of pulse length. Figure 3.6 presents the

indication of the current source injection (a) as well as an illustration of the current source wave with the ΔA indicating the amplitude variation and the ΔT indicating the pulse width variation (b).

As it was done in the other circuits presented in this work, the ΔA values used in the experiments were of 10µA, 50µA and 100µA with ΔT of 200ps, 400ps, 600ps, 800ps, 1000ps, 1200ps, 1400ps, 1600ps, 1800ps and 2000ps. In order to consider the architectural vulnerability factor, all waves were injected in all transistors using as input the 4 possibilities of quaternary levels (0, 1, 2 and 3). The current sources injected at the transistors of the circuit were in the shape of positive and negative pulses. At the end, each node of the circuit was injected with 240 sources of current during the HSPICE simulation process. To take into account the time vulnerability factor, the time of all injection pulses was specially selected in order to consider the worst case scenario which is the time that the Flip Flop is storing the input.

In order to have a more realistic situation, low level electrical simulation (with HSPICE) was used to investigate the transient response of the proposed quaternary circuits with different input stimulus and with several different SETs configurations. To reduce the number of simulations, the simulated EA-QLUT circuit size and the binary correspondent one were simulated only with one quaternary input (corresponding to two binary inputs for the binary version). As the proposed technique is scalable, only small variances of the fault tolerance results are expected to be observed for the circuits with 2 and 3 inputs.

The next section presets the simulation results of the fault injection procedure that was described in this section. It highlights the percentage of SETs that caused errors and the correspondent area overheads of the proposed FT-Q-CLB and the binary correspondent protected with the TMR technique.

5.2 Simulation Results, Overheads and Results Analysis

This section presents the results of the fault injection experiments and the area overheads that resulted from the proposed approach and its binary correspondent.

In order to evaluate and to propose the fault tolerant techniques of this work, both quaternary and binary CLBs were simulated with current pulse injections as it was explained in a previous Chapter. Both quaternary and binary CLBs versions with no FT technique were simulated with 240 current pulses in all nodes of the circuit described in transistor level with the HSPICE simulator to identify which are the most sensitive nodes or regions of each circuit.

Table 5.1 presents the fault injection results of the quaternary and the binary CLBs with no FT techniques. In order to identify if a certain pulse caused an error, it was developed a C language program that analyzes the waveform of the output of the Flip-Flop and identify if the value corresponds to the expected one.

One can see from Table 5.1 that the second row presents the total number of injected nodes. The number of nodes is defined as the number of convergence points of one or more transistor drains. For example, if two circuits have the same number of transistors but one circuit has more transistors with its drains connected to the same point than the other, the one with more drains connected together will have fewer nodes than the other one. This relation explains why the number of injection nodes is not linearly proportional to the number of transistors of each circuit. Table 5.1 shows that the number of injection nodes of both quaternary and binary is almost the same, but when looking at the number of errors presented in the last row one can see that the quaternary version had more than 22% of injections resulting in comparison to only 6,97 of the binary one. That happened because as the quaternary circuits have two more intermediate levels between the ground and Vcc, the necessary charge or particle energy necessary to cause an error is smaller in the quaternary circuits than in the binary ones. Only the injection pulses with high amplitude and long duration of pulses caused errors in the binary CLB.

Table 5.1. Error Rate for the Quaternary and Binary Circuits with no FT

	Quaternary		Binary	
# of nodes injected	27		28	
# of pulses injected	12,960		13,440	
Pos. Pulses that caused Errors	1424	10,99%	476	3,54%
Neg. Pulses that caused Errors	1464	11,30%	460	3,42%
Total # and % of Errors	**2888**	**22,28%**	**936**	**6,97%**

(Source: Created by the author)

To illustrate the distribution of errors in each component of the circuit, Table 5.2 presents the error rate in each component of the CLB for both the quaternary and the binary versions with no FT technique.

Table 5.2. Error Rate by Component for the Quaternary CLB and the Binary Correspondent with no FT.

	Quaternary (%)	**Binary (%)**
LUT	5.20	0.83
Muxs	3.58	1.10
Flip Flop	13.50	5.03

(Source: Created by the author)

When one looks at Table 5.2 one can see that the majority of errors occur in the Flip Flop component of both, the quaternary and the binary circuits. That can be explained mainly for two reasons, first: the Flip Flop is the component that registers the error and therefore, if an SET occurs in the Flip Flop with energy enough to be sustained by the inverters loop that form the Flip Flop, the error will occur as there is no other component that can filter the error. The second reason comes from the fact that, as will be presented in the next section, the Flip Flop occupies a significant area of the CLB, and therefore the probability of a high energy particle strike the Flip Flop is bigger than any other component.

After characterizing the quaternary and the binary CLBs with no fault tolerant technique both circuits, the proposed FT-Q-CLB and the binary CLB protected with the well known TMR technique were simulated with the injection of current pulses in all circuit nodes, following the same methodology as with the circuits with no FT. Table 5.3 presents the error rate for the proposed FT-Q-CLB and its binary correspondent.

Table 5.3. Error Rate for the FT-Q-CLB and the Binary Correspondent.

	Quaternary		Binary	
# of nodes injected	85		93	
# of pulses injected	20,400		24,480	
Pos. Pulses that caused Errors	0	0%	44	0,39%
Neg. Pulses that caused Errors	0	0%	58	0,52%
Total # and % of Errors	**0**	**0%**	**102**	**0,41%**

(Source: Created by the author)

One can see from Table 5.3 that the quaternary FT techniques used in the proposed FT-Q-CLB detected and corrected all errors. When it comes to the binary version, all the 102 pulses that caused errors were located in the TMR voter. In order to reduce the binary version to 0 errors one

would just need to increase the size of the TMR voter to increase the value of the critic charge (Qcritic). This measure would consequently increase the total area of the binary version. The next section presents and discusses the area results and overhead of the proposed FT-QCLB.

5.3 Area Results

In order to demonstrate the area overhead of both fault tolerant versions, the quaternary and the binary ones, Table 5.4 illustrates the area of both CLBs without any FT technique.

Table 5.4. Area of Quaternary and Binary CLBs with no FT Technique.

	Quaternary				Binary				**Total**
# inputs	QLUT	Muxs	FF	Total	BLUT	Muxs	FF	Total	**Ratio**
1	24	12	42	78	32	24	32	88	**1.13**
2	88	12	42	142	136	24	32	192	**1.35**
3	240	12	42	294	456	24	32	512	**1.74**

(Source: Created by the author)

One can see from Table 5.4 that the quaternary CLB has 13%, 35% and 74% less area than its binary correspondent for 1, 2 and 3 inputs respectively.

Table 5.5 presents the proposed quaternary FT approach and its binary equivalent protected with the TMR technique.

Table 5.5. Area of Quaternary and Binary CLBs with FT Technique.

	Quaternary (FT-Q-CLB)				Binary (with TMR)				**Total**
# inputs	QLUT	Muxs	FF	Total	BLUT	Muxs	FF	Total	**Ratio**
1	96	60	130	286	96	72	96	304	**1.06**
2	274	60	130	464	408	72	96	616	**1.33**
3	788	60	130	978	1368	72	96	1576	**1.61**

(Source: Created by the author)

One can see from Table 5.5 that the quaternary FT approach has 6%, 33% and 61% less area than its binary correspondent for 1, 2 and 3 inputs respectively. In order to analyze the overhead of both approaches, Table 5.6 summarizes this information by combining the information of Table 5.4 and Table 5.5.

Table 5.6. Area Overhead Comparing the Quaternary and the Binary FT Versions

	Quaternary (FT-Q-CLB)				Binary (with TMR)				**Total**
# inputs	QLUT	Muxs	FF	Total	BLUT	Muxs	FF	Total	**Ratio**
1	72	48	88	208	64	48	64	176	**0.85**
2	186	48	88	322	272	48	64	384	**1.19**
3	548	48	88	684	912	48	64	1024	**1.50**

(Source: Created by the author)

One can see from Table 5.6 that the quaternary FT version with 1 input has more overhead than its binary version. That happened mainly because the overhead of the FT quaternary Flip-Flop is too significant for the total CLB overhead. For the circuits with 2 and 3 input sizes (quaternary equivalent) the overhead of the Flip-Flop becomes less significant to the total overhead. With 2 and 3 inputs the proposed quaternary solution has 19% and 50% less overhead than its binary counterpart.

6 CONCLUSION AND FUTURE WORK

This work presented the study of quaternary logic circuits from the fault tolerant perspective. By the presented results and the published work, one can conclude that quaternary circuits can have appealing characteristics for space applications, not only because it reduces the interconnection related costs, like for instance, number of interconnections (resulting in the reduction of the area occupied by the wiring), logic area, power consumption, but also because the quaternary circuits presented some interesting area reduction characteristics that consequently lead to a reduction of the power and total energy demanded by a circuit.

In this work the impact of Soft Errors such as Single Event Transients was targeted first, in small circuits with a few dozens of transistors like the lookup tables and flip-flops and lastly in a bigger circuit, the Q-CLB. Using the experience obtained in the fault injection experiments of the smaller circuits, this work proposed a quaternary CLB with fault tolerant properties, here called FT-Q-CLB, which detected and corrected all errors provoked by the SET injection in the circuit. All fault injection experiments were conducted with the precaution of being as fair as possible, explaining and showing why the compared binary strategy was the one that best fit to evaluate the proposed quaternary circuit.

By means of a low level electrical simulation the obtained results demonstrated that the quaternary fault tolerant techniques were able to effectively detect and correct all errors in the quaternary version, always having considerable lower overhead results in terms of transistor count and power consumption, as the results presented by the FT-QLUT at Chapter 4. It is important to mention that the proposed quaternary circuits have presented considerable disadvantages in terms of performance in comparison to the binary correspondent circuits. On favor of the quaternary circuits, we must point out that the transistor models used in the simulation experiments are just an approximation of the expected behavior of the multiple Vts transistors used in the quaternary logic.

The performance experimental results obtained with the transistor resizing showed us that one can increase the performance of the quaternary circuits with very low area overheads.

As a continuity of this work we intend to better evaluate the quaternary circuits in terms of the process variability impact for sub 65nm technologies mainly regarding the V_{th} and C_{ox} transistor parameters. The proposed FT-Q-CLB was not completely evaluated in terms of power consumption and performance overheads. These figures will also be targeted in future works. Also, the development of a complete quaternary FPGA is an open research project that can be seen as normal continuity of the work presented in this thesis. In order to do this, one can use the quaternary CLB proposed here but there are the other components of an FPGA that still need to be developed like a quaternary switching matrix and the I/O block compatibilities. Also, all the FPGA synthesis toolset that are used nowadays and are target for binary synthesis need to be adapted for the quaternary logic domain.

7 REFERENCES

ALEXANDRESCU, D.; ANGHEL, L.; NICOLAIDIS, M. New methods for evaluating the impact of single event transients in VDSM ICs. In: IEEE DEFECT AND FAULT TOLERANCE IN VLSI SYSTEMS WORKSHOP, DFT 2002, 17**. Proceedings of 17th IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems, 2002**. [S. l., s. n.]: IEEE, 2002. p.99-107. ISBN: 1063-6722

BAUER, M. et al. A multilevel-cell 32 Mb flash memory. In: IEEE INTERNATIONAL SOLID-STATE CIRCUITS CONFERENCE, ISSCC 1995, 42., 15-17 Feb 1995. **Digest of Technical Papers.** [S. l., s. n.]: IEEE, 1995. p.132-133, 351 1995. DOI: 10.1109/ISSCC.1995.535462

BAUMANN, R. et al. Boron compounds as a dominant source of alpha particles in semiconductor devices. In: IEEE RELIABILITY PHYSICS SYMPOSIUM, 1995, 33., 4-6 April 1995. **Proceedings...** [S. l., s. n.]: IEEE, 1995. p.297-302. DOI: 10.1109/RELPHY.1995.513695

BAUMANN, R. C. **Silicon amnesia: a tutorial on radiation induced soft errors**. In: IEEE INTERNATIONAL RELIABILITY PHYSICS SYMPOSIUM, 2001. [S. l., s. n.].

BERNSTEIN, K. F., D. J.; GATTIKER, A. E.; HAENSCH, W.; JI, B. L.; NASSIF, S. R.; NOWAK, E. J.; PEARSON, D. J.; ROHRER, N. J. High-performance CMOS variability in the 65-nm regime and beyond. **IBM Journal of Research and Development.** [S. l.] v. 50, n. 4.5, p. 433-449, 2006. DOI: 10.1147/rd.504.0433

BOSSEN, D. C. CMOS Soft Errors and Server Design. **Reliability Physics Tutorial Notes 2002**. [S. l., s. n.].

CAO, Y. **Predictive Technology Model (PTM)**. 06/01/2011 2007. Available at: < http://ptm.asu.edu/ >. Last Access: 20/05/2012.

CUNHA, R.; BOUDINOV, H.; CARRO, L. Quaternary Look-Up Tables Using Voltage-Mode CMOS Logic Design. In: IEEE INTERNATIONAL SYMPOSIUM ON MULTIPLE-VALUED LOGIC, ISMVL 2007, 37., 13-16 May 2007. **Proceedings...** [S. l., s. n.]: IEEE, 2007. p.56-56. ISBN: 0195-623X

CURRENT, K. W. A CMOS quaternary latch. In: IEEE INTERNATIONAL SYMPOSIUM ON MULTIPLE-VALUED LOGIC, ISMVL 1989, 19., 29-31 May 1989. **Proceedings...** [S. l., s. n.]: IEEE, 1989. p.54-57. ISBN: 0-8186-1947-3

CURRENT, K. W. A CMOS quaternary threshold logic full adder circuit with transparent latch. In: IEEE INTERNATIONAL SYMPOSIUM ON MULTIPLE-VALUED LOGIC, ISMVL 1990, 20., 23-25 May 1990. **Proceedings...** [S. l., s. n.]: IEEE, 1990. p.168-173. ISBN: 0-8186-2046-3

CURRENT, K. W.; HURLSTON, M. E. A bi-directional current-mode CMOS multiple valued logic memory circuit. . In: IEEE INTERNATIONAL SYMPOSIUM ON MULTIPLE-VALUED

LOGIC, ISMVL 1991, 21., 26-29 May 1991. **Proceedings...** [S. l., s. n.]: IEEE, 1991. p.196-202. ISBN: 0-8186-2145-1

DATLA, S. R. P. R.; THORNTON, M. A. Quaternary Voltage-Mode Logic Cells and Fixed-Point Multiplication Circuits. In: IEEE INTERNATIONAL SYMPOSIUM ON MULTIPLE-VALUED LOGIC, ISMVL 2010, 40., 26-28 May 2010. **Proceedings...** [S. l., s. n.]: IEEE, 2010. p.128-133. ISBN: 978-1-4244-6752-5

DODD, P. E. et al. Impact of substrate thickness on single-event effects in integrated circuits. In: IEEE TRANSACTIONS ON NUCLEAR SCIENCE, 2001. **Proceedings...** [S. l.]:IEEE, 2001. v. 48, n. 6, p. 1865-1871, 2001. ISSN 0018-9499.

DRUZETA, A.; VRANESIC, Z. G.; SEDRA, A. S. Application of Multithreshold Elements in the Realization of Many-Valued Logic Networks. **IEEE Transactions on Computers,** [S. l.] v. C-23, n. 11, p. 1194-1198, 1974. ISSN 0018-9340.

DUNDERDALE, H. Current-mode circuits for the unary functions of a ternary variable. **IET Electronics Letters,** [S. l.] v. 6, n. 1, p. 15-16, 1970. ISSN 0013-5194.

FEI, L. et al. Power modeling and characteristics of field programmable gate arrays. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems,** [S. l.] v. 24, n. 11, p. 1712-1724, 2005. ISSN 0278-0070.

FREITAS, D. A.; CURRENT, K. W. CMOS current comparator circuit. **Electronics Letters,** [S. l.] v. 19, n. 17, p. 695-697, 1983a. ISSN 0013-5194.

FREITAS, D. A.; CURRENT, K. W. A Quaternary Logic Encoder-Decoder Circuit Design Using CMOS. In: IEEE INTERNATIONAL SYMPOSIUM ON MULTIPLE-VALUED LOGIC, ISMVL 1983, 13. **Proceedings...** [S. l., s. n.]: IEEE, 1983. p 190-195.

FREITAS, D. A.; CURRENT, K. W. A simple high-gain CMOS voltage comparator circuit†. **International Journal of Electronics,** [S. l.] v. 57, n. 2, p. 195-198, 1984/08/01 1984. ISSN 0020-7217. Available at: < http://www.tandfonline.com/doi/abs/10.1080/00207218408938899 >. Last Access: 08/11/2012.

FRIEDMAN, N. et al. Realization of a multivalued integrated injection logic (MI/sup 2/L) full adder. **IEEE Journal of Solid-State Circuits,** [S. l.] v. 12, n. 5, p. 532-534, 1977. ISSN 0018-9200.

GAC, GERMAN AEROSPACE CENTER. **Applications of Nanotechnology in Space Developments and Systems. Technological Analysis**, [S. l., s. n.] p. 1, April 2003.

GONÇALVES DA SILVA, R. C.; BOUDINOV, H.; CARRO, L. A novel Voltage-mode CMOS quaternary logic design. **IEEE Transactions on Electron Devices ,** [S. l.] v. 53, n. 6, p. 1480-1483, 2006. ISSN 0018-9383.

GONÇALVES DA SILVA, R. C.; BOUDINOV, H. I.; CARRO, L. A low power high performance CMOS voltage-mode quaternary full adder. In: IEEE INTERNATIONAL CONFERENCE ON VERY LARGE SCALE INTEGRATION, IFIP 2006., 16-18 Oct. 2006. **Proceedings...** [S. l., s. n.]: IEEE, 2006. p.187-191. ISBN: 3-901882-19-7

GONZALEZ, A. F. et al. Standard CMOS implementation of a multiple-valued logic signed-digit adder based on negative differential-resistance devices. In: IEEE INTERNATIONAL SYMPOSIUM ON MULTIPLE-VALUED LOGIC, ISMVL 2000, 30. **Proceedings...** [S. l., s. n.]: IEEE, 2000. p.323-328. ISBN: 0-7695-0692-5

GONZALEZ, A. F.; MAZUMDER, P. Multiple-valued signed digit adder using negative differential resistance devices. **IEEE Transactions on Computers,** [S. l.] v. 47, n. 9, p. 947-959, 1998. ISSN 0018-9340.

HANYU, T.; IKE, T.; KAMEYAMA, M. Self-checking multiple-valued circuit based on dual-rail current-mode differential logic. In: IEEE INTERNATIONAL SYMPOSIUM ON MULTIPLE-VALUED LOGIC, ISMVL 1999, 29., **Proceedings...** [S. l., s. n.]: IEEE, 1999. p.275-279. ISBN: 0-7695-0161-3

HANYU, T.; MOCHIZUKI, A.; KAMEYAMA, M. Multiple-valued dynamic source-coupled logic. In: IEEE INTERNATIONAL SYMPOSIUM ON MULTIPLE-VALUED LOGIC, ISMVL 2003, 33., 16-19 May 2003. **Proceedings...** [S. l., s. n.]: IEEE, 2003. p.207-212.

HANYU, T.; SAITO, T.; KAMEYAMA, M. Asynchronous multiple-valued VLSI system based on dual-rail current-mode differential logic. In: IEEE INTERNATIONAL SYMPOSIUM ON MULTIPLE-VALUED LOGIC, ISMVL 1998, 28., 27-29 May 1998. **Proceedings...** [S. l., s. n.]: IEEE, 1998. p.134-139. ISBN: 0-7695-0161-3

HARELAND, S. et al. Impact of CMOS process scaling and SOI on the soft error rates of logic processes. In: IEEE Symposium on VLSI Technology, 2001. **Digest of Technical Papers**. [S. l., s. n.]: IEEE, 2001. p.73-74. ISBN: 4-89114-012-7

HEIJMEN, T. Radiation-induced soft errors in digital circuits - A literature survey. **Philips Electronics Nederland BV 2002.**, [S. l., s. n.] p. 7-20, 2002.

HEUNG, A.; MOUFTAH, H. T. Depletion/enhancement CMOS for a lower power family of three-valued logic circuits. **IEEE Journal of Solid-State Circuits,** [S. l.] v. 20, n. 2, p. 609-616, 1985. ISSN 0018-9200.

HURST, S. L. **The Logical Processing of Digital Signals**. Crane Russak & Co, 1978. 580 ISBN 978-0844809076.

HURST, S. L. Multiple-Valued Logic: its Status and its Future. **IEEE Transactions on Computers,** [S. l.] v. C-33, n. 12, p. 1160-1179, 1984. ISSN 0018-9340.

JOHNSON, B. W. **Design and Analysis of Fault Tolerant Digital Systems: Solutions Manual**. Addison-Wesley Publishing Company, 1994.

KARNIK, T. et al. Scaling trends of cosmic ray induced soft errors in static latches beyond 0.18 um. In: IEEE SYMPOSIUM ON VLSI CIRCUITS, 2001. **Digest of Technical Papers**. [S. l., s. n.] p.61-62. ISBN: 4-89114-014-3

KARNIK, T. et al. Selective node engineering for chip-level soft error rate improvement [in CMOS]. In: IEEE SYMPOSIUM ON VLSI CIRCUITS, 2002. **Digest of Technical Papers**. [S. l., s. n.] p.204-205. **ISBN:** 0-7803-7310-3

KAWAHITO, S. et al. High-speed area-efficient multiplier design using multiple-valued current-mode circuits. **IEEE Transactions on Computers,** [S. l.] v. 43, n. 1, p. 34-42, 1994. ISSN 0018-9340.

LAZZARI, C. et al. A new quaternary FPGA based on a voltage-mode multi-valued circuit. In: PROCEEDINGS OF THE CONFERENCE ON DESIGN, AUTOMATION & TEST IN EUROPE 2010a, 8-12 March 2010. **Proceedings…** [S. l., s. n.]: ACM, 2010. p.1797-1802. ISBN: 978-3-9810801-6-2

LAZZARI, C. et al. Voltage-mode quaternary FPGAs: An evaluation of interconnections. In: IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, ISCAS 2010b., May 30 2010-June 2. **Proceedings…** [S. l., s. n.]: IEEE, 2010. p.869-872. ISBN: 978-1-4244-5308-5

LEBLEBICI, Y.; GURKAYNAK, F. K. Modular realization of threshold logic gates for high performance digital signal processing applications. In: IEEE 11TH INTERNATIONAL ASIC CONFERENCE, 1998., 13-16 Sep 1998 **Proceedings…** [S. l., s. n.]: IEEE, 1998. . p.281-285. ISBN: 0-7803-4980-6

MAHAPATRA, S.; IONESCU, A. M. Realization of multiple valued logic and memory by hybrid SETMOS architecture. **IEEE Transactions on Nanotechnology,** [S. l.] v. 4, n. 6, p. 705-714, 2005. ISSN 1536-125X.

MITRA, S. et al. Robust system design with built-in soft-error resilience. **Computer,** [S. l.] v. 38, n. 2, p. 43-52, 2005. ISSN 0018-9162.

NAGEL, L. **Simulation Program with Integrated Circuit Emphasis.** 1973. Available at: < http://bwrc.eecs.berkeley.edu/classes/icbook/spice/ >. Last Access: 08/11/2012.

NGUYEN, H. T.; YAGIL, Y. A systematic approach to SER estimation and solutions. In: IEEE INTERNATIONAL RELIABILITY PHYSICS SYMPOSIUM, 2003, 41., 30 March-4 April 2003. **Proceedings…** [S. l., s. n.]: IEEE, 2003. p.60-70. ISBN: 0-7803-7649-8

OOTSUKA, F. et al. A novel 0.20 um full CMOS SRAM cell using stacked cross couple with enhanced soft error immunity. In: IEEE INTERNATIONAL ELECTRON DEVICES MEETING, IEDM '98., 6-9 Dec 1998. **Technical Digest.** [S. l., s. n.]: IEEE, 1998. p.205-208. ISBN: 0-7803-4774-9

OZDEMIR, H. et al. A capacitive threshold-logic gate. **IEEE Journal of Solid-State Circuits,** [S. l.] v. 31, n. 8, p. 1141-1150, 1996. ISSN 0018-9200.

PARK, S. J. et al. Design of quaternary logic gate using double pass-transistor logic with neuron MOS down literal circuit. In: IEEE INTERNATIONAL SYMPOSIUM ON MULTIPLE-VALUED LOGIC, ISMVL 2004, 34., 19-22 May 2004. **Proceedings…** [S. l., s. n.]: IEEE, 2004. p.198-203. ISBN: 0-7695-2130-4

PEREIRA, M. M.; RHOD, E. L.; CARRO, L. Fault Tolerant Design and Adaptability. In: BECK, A. C. S. L., CARLOS ARTHUR LANG; CARRO, LUIGI; (Ed.). **Adaptable Embedded Systems**: Springer, v.1, 2012. p.317. ISBN 978-1-4614-1745-3.

PRAGASAN, R. Spartan FPGAs – The Gate Array Solution. **Xilinx Application Notes**, [S. l., s. n.] p. 6, August 2001 2001.

RHOD, E.; STERPONE, L.; CARRO, L. A New Soft-Error Resilient Voltage-Mode Quaternary Latch. In: IEEE INTERNATIONA SYMPOSIUM ON DEFECT AND FAULT TOLERANCE IN

VLSI SYSTEMS, DFT 2010, 25., 6-8 Oct. 2010. **Proceedings…** [S. l., s. n.]: IEEE, 2010. p.200-208. ISBN: 978-1-4244-8447-8

RHOD, E. L.; CARRO, L. A Low Cost Low Power Quaternary LUT Cell for Fault Tolerant Applications in Future Technologies. In: IEEE INTERNATIONAL SYMPOSIUM ON VLSI, 2009. ISVLSI '09., 13-15 May 2009. **Proceedings…** [S. l., s. n.]: IEEE, 2009. p.292-297. ISBN: 978-1-4244-4408-3

RITT, M. et al. A Cost-Effective Technique for Mapping BLUTs to QLUTs in FPGAs. In: IEEE INTERNATIONAL CONFERENCE ON FIELD PROGRAMABLE LOGIC APPLICARIONS, FPL 2010., Aug. 31 2010-Sept. 2 2010. **Proceedings…** [S. l., s. n.]: IEEE, 2010. p.332-335. ISBN: 978-1-4244-7842-2

SCHMID, A.; LEBLEBICI, Y. Realisation of multiple-valued functions using the capacitive threshold logic gate. **IEE Proceedings - Computers and Digital Techniques,** [S. l.] v. 151, n. 6, p. 435-447, 2004. ISSN 1350-2387.

SEIFERT, N. et al. Historical trend in alpha-particle induced soft error rates of the Alpha™ microprocessor. In: IEEE INTERNATIONAL RELIABILITY PHYSICS SYMPOSIUM, 2001, 39., **Proceedings...** [S. l., s. n.]:IEEE, 2001. p.259-265. ISBN: 0-7803-6587-9

SEXTON, F. W. et al. SEU simulation and testing of resistor-hardened D-latches in the SA3300 microprocessor. **IEEE Transactions on Nuclear Science,** [S. l.] v. 38, n. 6, p. 1521-1528, 1991. ISSN 0018-9499.

SHEIKHOLESLAMI, A.; YOSHIMURA, R.; GULAK, P. G. Look-up tables (LUTs) for multiple-valued, combinational logic. In: IEEE INTERNATIONAL SYMPOSIUM ON MULTIPLE-VALUED LOGIC, ISMVL 1998, 28., 27-29 May 1998. **Proceedings...** [S. l., s. n.]: IEEE, 1998.. p.264-269. ISBN: 0-8186-8371-6

SHIBATA, T.; OHMI, T. Neuron MOS Voltage-Mode Circuit Technology for Multiple-Valued Logic. **IEICE TRANSACTIONS on Electronics,** [S. l.] v. E76-C, n. 3, p. 347-356, 1993. ISSN 0916-8516.

SINGH, A.; PARTHASARATHY, G.; MAREK-SADOWSKA, M. Efficient circuit clustering for area and power reduction in FPGAs. **ACM Transactions on Design Automation of Electronic Systems,** [S. l.] v. 7, n. 4, p. 643-663, 2002. ISSN 1084-4309.

SYNOPSYS. **HSPICE 2009**. 2009. Available at: < http://www.synopsys.com/Tools/Verification/AMSVerification/CircuitSimulation/HSPICE/Pages/default.aspx >. Last Access: 08/11/2012.

THOIDIS, I. et al. Quaternary voltage-mode CMOS circuits for multiple-valued logic. **IEE Proceedings -Circuits, Devices and Systems,** [S. l.] v. 145, n. 2, p. 71-77, 1998. ISSN 1350-2409.

THOIDIS, I. M. et al. The circuit design of multiple-valued logic voltage-mode adders. In: IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, ISCAS 2001, 6-9 May 2001. **Proceedings...** [S. l., s. n.]: IEEE, 2001. p.162-165 vol. 4. ISBN: 0-7803-6685-9

THOIDIS, I. M.; SOUDRIS, D.; THANAILAKIS, A. Voltage-Mode Multiple-Valued Logic Adder Circuits. **IEICE TRANSACTIONS on Electronics**, [S. l.] E87-C**:** 1054-1061 p. 2004.

VASUNDARA, P. K. S.; GURUMURTHY, K. S. Quaternary CMOS Combinational Logic Circuits. In: IEEE INTERNATIONAL CONFERENCE ON INFORMATION AND

MULTIMEDIA TECHNOLOGY, 2009. ICIMT '09.,16-18 Dec. 2009. **Proceedings...** [S. l., s. n.]: IEEE, 2009. p.538-542. ISBN: 978-0-7695-3922-5

VELAZCO, R. et al. Two CMOS memory cells suitable for the design of SEU-tolerant VLSI circuits. **IEEE Transactions on Nuclear Science,** [S. l] v. 41, n. 6, p. 2229-2234, 1994. ISSN 0018-9499.

YASUDA, Y. et al. Realization of quaternary logic circuits by n-channel MOS devices. **IEEE Journal of Solid-State Circuits,** [S. l.] v. 21, n. 1, p. 162-168, 1986. ISSN 0018-9200.

ZIEGLER, J. F.; LANFORD, W. A. The effect of sea level cosmic rays on electronic devices. **Journal of Applied Physics,** [S. l.] v. 52, n. 6, p. 4305-4312, 1981. ISSN 0021-8979.

ZILIC, Z.; VRANESIC, Z. G. Multiple-valued logic in FPGAs. In: MIDWEST SYMPOSIUM ON CIRCUITS AND SYSTEMS, 1993., 16-18 Aug 1993. **Proceedings…** , [S. l., s. n.]: IEEE, 1993. v. 2, p.1553-1556. ISBN: 0-7803-1760-2

APENDIX A: DIFFERENT BASES AND ITS RELATIONS

This appendix presents a brief guide for understanding the relationship beetween the binary and the quaternary logics, mainly focusing on the correlation between the quaternary and the binary logic that can be explored to reduce the CLB lookup table in terms of area, without reducing its functional capacity.

In order to better understand how the different ligic bases work, lets us consider the well know binary logic as a starting point. The number of logic functions tha one can have for a given 'n' input number in tha binary base is given by Equation A.1 bellow:

$$B^{B^n} \tag{A.1}$$

In binary logic, the number of the possible functions that one can have with only one input are only 4 as, using Equation A.1, $2^{2^1} = 4$. Table A.1 presents these 4 possible binary functions with one variable as input.

Table A.1: Binary Logic Functions with 1 Binary Input.

Variable	Function 1	Function 2	Function 3	Function 4
0	0	0	1	1
1	0	1	0	1

Among the function presented at Table A.1, one can find the inverter function presented at th Function 3 raw, which is the simplest function in binary logic. The most know binary logic functions have two variables as input. Table A.2 presents the 16 (2^{2^2}) possible functions that one can have with 2 binary inputs.

Table A.2: Binary Logic Functions with 2 Binary Inputs.

Variable A	Variable B	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15	F16
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Among the fucntions presented at Table A.2, one can highligth the AND (F2), OR (F8), NAND (F15), NOR (F9) and XOR (F7) binary functions. All digital electronic is conceived by these small set of functions. By combining other logic gates and construction more complex logic functions one can obtaing the other functions that are presented at Table A.2. When working with other bases, such like the ternary base, there are certain functions that cannot be obtained with the binary logic, which can serve in a more efficient way to describe other sistems.

The first non binary logic circuits were developed using the ternary logic, which in other words means the use of the numbers in base 3. In ternary logic a 1 input and 1 output logic function can have up to 27 (3^{3^1}) different functions. Table A.3 presents all the one input ternary logic functions using as notation the 0, 1 and 2 for the three logic values.

Table A.3: Ternary Functions for One Input and One Output

entrada	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	...
0	0	0	0	0	0	0	0	0	0	1	1	1	1	...
1	0	0	0	1	1	1	2	2	2	0	0	0	1	...
2	0	1	2	0	1	2	0	1	2	0	1	2	0	...

...	F14	F15	F16	F17	F18	F19	F20	F21	F22	F23	F24	F25	F26	F27
...	1	1	1	1	1	2	2	2	2	2	2	2	2	2
...	1	1	2	2	2	0	0	0	1	1	1	2	2	2
...	1	2	0	1	2	0	1	2	0	1	2	0	1	2

One can notice from Table A.3 that a hole new series of functions rises in comparison the the one input binary function of Table A.1. For instant the so called Successor function is presented at Function F16 and the Predecessor function at the F20 function of Table A.3. Whe one uses 2 ternary inputs and one ternary output the number of possible logic functions that one can have is

given by the expression $3^{(3^2)}$ which results in 19,683 functions and some new functions can be observed like for instance the MAX and the MIN functions, which gives as output the greater and the lesser value between the two inputs respectivelly.

When it comes for the quaternary logic, using four distinct values to represent the logic, which in this work are the numbers 0,1,2 and 3, the number of the possible obtained functions using one quaternary imput and one quaternary output is 256, given by the expression 4^{4^1}. For a function with 2 quanternary variables as input and one as output this number grows to 4^{4^2} which is equal to 4,294,967,296. As an exemplo of two quaternary functions with two quaternary inputs and one quaternary output, Figure A.1 presents the MAX and MIN quaternary functions.

Figure A.1: The MAX(a) and MIN(b) Quaternary Functions.

MAX (X,Y)

Y \ X	0	1	2	3
0	0	1	2	3
1	1	1	2	3
2	2	2	2	3
3	3	3	3	3

(a)

MIN (X,Y)

Y \ X	0	1	2	3
0	0	0	0	0
1	0	1	1	1
2	0	1	2	2
3	0	1	2	3

(b)

Table A.4 presents a few of the 256 quaternary functions with 1 quaternary input and one quaternary output.

Table A.4: Quaternary Logic Functions with One Quaternary Input and One Quaternary Output

input	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	...
0	0	0	0	0	0	0	0	0	0	0	0	0	0	...
1	0	0	0	0	0	0	0	0	0	0	0	0	0	...
2	0	0	0	0	1	1	1	1	2	2	2	2	3	...
3	0	1	2	3	0	1	2	3	0	1	2	3	0	...

...	F242	F243	F244	F245	F246	F247	F248	F249	F250	F251	F252	F253	F254	F255
...	3	3	3	3	3	3	3	3	3	3	3	3	3	3
...	3	3	3	3	3	3	3	3	3	3	3	3	3	3
...	0	0	1	1	1	1	2	2	2	2	3	3	3	3
...	2	3	0	1	2	3	0	1	2	3	0	1	2	3

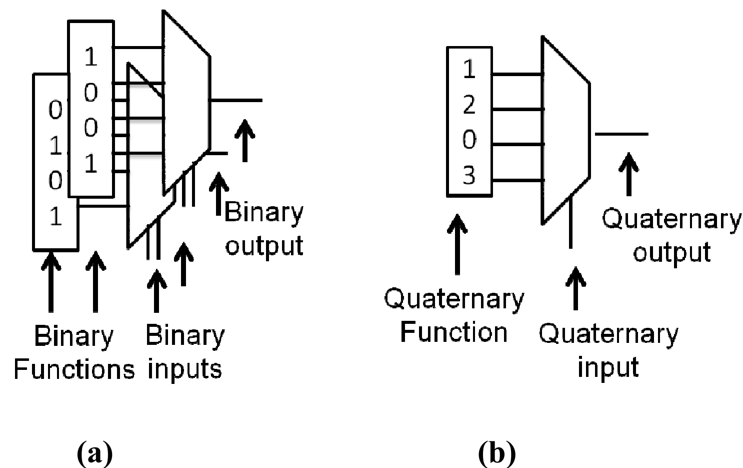
There is a close relation between the bases that are multiple by the same number , like for instance between the binary and quaternary bases or between the binary and the octal base also between the quaternary and the hexadecimal base. This relation makes easier the convertion between these bases. For instace, in order to convert a number from the binary to the quaternary base, one only needs to take every 2 bits, starting from the least significant ones, and convert to 1 quaternary digit according to Table A.5.

Table A.5: Quaternary and Binary Logic Conversion Table

Quaternary	Binary (Bit1)	Binary (Bit0)
0	0	0
1	0	1
2	1	0
3	1	1

By using this relation, one can conclude that when one uses the quaternary logic instead of the binary one, one can have two times more capacity of information in a quaternary digit in comparisson to a binary one. This work takes advantage of this relation by using the quaternary condensed way of expressing information in the place of the traditional binary one. More precisely this works uses the quaternary digits to replace the binary ones in lookup tables. Figure A.2 presents this relation.

Figure A.2: Relation Between the Binary(a) and the Quaternary(b) Lookup Tables.



(a) **(b)**

One can see from Figure A.2 (a) that the 2 input binary lookup table is equivalent to one input quaternary lookup table (Figure A.2 (b)). As the quaternary one drives twice more information, the binary one needs to be doubled in order to have the same capacity as the quaternary one.

APENDIX B: CONTRIBUTIONS

This apendix lists the papers and contributions of the author that were developed during the PhD study divided into the works that were acepted for publishing, not acepted for publishing, waiting response.

Acepted for publication

1. Rhod, E.L.; Rutzig, M.B.; Carro, L.; , "Binary translation process to optimize nanowire arrays usage," *Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on* , vol., no., pp.396-399, 18-21 May 2008.

2. Rhod, E. L. and Carro, L., “An Efficient Test and Characterization Approach for Nanowire-Based Architectures”. In *Proceedings of the 21st Annual Symposium on integrated Circuits and System Design,* SBCCI '08. Gramado, Brazil, Sep 01 - 04, 2008. ACM, New York, NY, 34-39. ISBN:978-1-60558-231-3

3. Rhod, E. L. ; Carro, L. ; Cota, E. “An Efficient Test and Characterization Approach for Nanowire-Based Architecturex.” In: 13th IEEE European Test Symposium, 2008, Torino. Proceedings of the 13th IEEE European Test Symposium. 2008.

4. RHOD, E. L., CARRO, L.” A Low Cost Low Power Quaternary LUT Cell for Fault Tolerant Applications in Future Technologies” In: IEEE Computer Society Annual Symposium on VLSI, 2009, Tampa. v.0. p.292 – 297

5. RHOD, E. L., STERPONE, L., CARRO, L. “A new RC design for mixed-grain based dynamically reconfigurable architectures” In: A new RC design for mixed-grain based dynamically reconfigurable architectures, 2010, Yasmine Hammamet. 16th IEEE International Conference on Electronics, Circuits, and Systems, ICECS 2010, p.984 – 987.

6. **<<Best Student Paper Award>>** RHOD, E. L., STERPONE, L., CARRO, L., “A New Soft-Error Resilient Voltage-Mode Quaternary Latch”.

7. **Book Chapter:** PEREIRA, M. M.; RHOD, E. L.; CARRO, L. Fault Tolerant Design and Adaptability. In: BECK, A. C. S. L.; LISBÔA, C. A. L.; CARRO, LUIGI; (Ed.). **Adaptable Embedded Systems**: Springer, v.1, 2012. p.317. ISBN 978-1-4614-1745-3.

Waiting for response

1. RHOD, E. L., CARRO, L., A Fault Tolerant Quaternary Configurable Logic Block (FT-Q-CLB) for Fault Tolerant Quaternary FPGAs. Submited to Microelectronics Reliabilities in may 17th 2011.

Submissions not acepted for publication

1. SEMISH 2009. [REJEITADO] Monica Magalhães Pereira, Eduardo Rhod, Carlos Arthur Lang Lisbôa, Thiago Berticelli Ló e Luigi Carro, "Lidando com os Desafios Impostos pelas Tecnologias Futuras em Diferentes Níveis de Abstração."

2. Eletronic Letters 2009 [REJEITADO] A Quaternary LUT Cell for Fault Tolerant Applications

3. DAC 2009. [REJEITADO] RHOD, E. L., CARRO, L, "A Low Cost Low Power Error Correction Quaternary LUT Cell for Fault Tolerant Applications".

4. DSD 2010. [REJEITADO] RHOD, E. L., STERPONE, L., CARRO, L., "A New Soft-Error Resilient Voltage-Mode Quaternary Latch".