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

RAUL DARIO CHIPANA QUISPE

SINGLE EVENT TRANSIENT EFFECTS IN CLOCK DISTRIBUTION NETWORKS

Thesis presented in partial fulfillment of the requirements for the degree of Doctor of Microelectronics

Prof. Dra. Fernanda Lima Kastensmidt
Advisor

Porto Alegre, December 2014.

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

CHIPANA QUISPE, RAUL DARIO.

Single Event Transient Effects in Clock Distribution Networks / Raúl Darío Chipana Quispe. – Porto Alegre: PGMICRO - UFRGS, 2014.

119 f.: il.

Orientador: Fernanda Lima Kastensmidt.

Thesis (Ph.D.) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Microeletrônica. Porto Alegre, BR – RS, 2014.

1. Single Event Transient. 2. Clock distribution networks 3. Clock extraction. 4. Clock tree. 5. Clock mesh. I. Kastensmidt, Fernanda G. L. II. Single Event Transient Effects in Clock Distribution Networks.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Reitor: Prof. Carlos Alexandre Netto
Vice-Reitor: Prof. Rui Vicente Oppermann
Pró-Reitor de Pós-Graduação: Prof. Vladimir Pinheiro do Nascimento
Diretor do Instituto de Informática: Prof. Luís da Cunha Lamb
Coordenador do PGMICRO: Prof. Gilson Inácio Wirth
Bibliotecária-Chefe do Instituto de Informática: Beatriz Regina Bastos Haro

ABSTRACT

Technology scaling to semiconductor has increased the radiation-induced susceptibility of electronic devices. Single Event Transient (SET) are becoming increasingly problematic for integrated circuits (ICs). Radiation effects may occur in the clock distribution networks of the ICs. During the strike of an ionizing particle, charge may be collected on the output node of the clock buffer, provoking a clock glitch, clock jitter and clock skew. As consequence of the impact, it is possible to notice errors in the control flow or data flow of the system. This work investigates the SET susceptibility in the clock distribution network of the circuit. We are interested in the most sensitive paths of the network and registers that are most likely to flip in the clock network. Some bit-flips are most likely to provoke a fault in the IC output once a failure occur in those elements. In the present work we propose a new methodology to identify the most sensitive nodes and to calculate the soft error rate due to SET in clock distribution network. This new methodology uses a tool developed in this thesis named EXT-CLK. The tool extracts the clock network from layout design files, to perform different simulations of SET injection in electrical and logic level. The SRAM arbiter circuit has been chosen as a case study. Thousands of electrical simulations have been performed in order to identify the sensitive nodes of the clock network. Results show that 17 registers of SRAM arbiter exhibit high vulnerability factor. This information can help the designers to use some mitigation techniques on those registers before the manufacturing process.

RESUMO

A redução na escala dos semicondutores tem aumentado a suceptibilidade de componentes eletrônicos a radiação. *Single event transient* (SET) afeta cada vez mais os circuitos integrados. Os efeitos da radiação podem afetar as redes de relógio dos circuitos integrados. Durante o impacto de uma partícula ionizada, a carga pode ser coletada na saída do *buffer* da rede de relógio e provocar um *clock glich*, *clock jitter* e *clock skew*. Como consequência do impacto, é possivel notar erros no fluxo do controle e no fluxo de dados do sistema. A presente tese investiga a suscetibilidade ao SET nas redes de relógio dos circuitos. Nós estamos interessados nos caminhos mais sensíveis da rede e nos registros que apresentam mais probabilidade de mudar de estado (*bit-flip*). Alguns *bit-flips* tem mais probabilidade de provocar uma falha na saída do circuito, enquanto outros podem ser mascarados pela aplicação. Nesta tese propomos uma nova metodologia para identificar os nós mais sensíveis e calcular o *soft error rate* causado pelo SET nas redes de relógio. Nossa metodologia utiliza uma ferramenta desemvolvida para esta tese chamada EXT-CLK, a ferramenta extrai a rede de relógio dos archivos de desenho do circuito para realizar diferentes simulações de injeção de SET. Como estudo de caso foi selecionado o circuito *SRAM arbiter*. Centenas de simulações foram feitas com o intuito de identificar os nós mais sensíveis da rede de relógio. Os resultados mostram 17 registros do ciruito *SRAM arbiter* terem alto índice de suscetibilidade. A informação encontrada nos resultados poderão ajudar os desenhadores a escolher a técnica de mitigação mais apropriada para o circuito antes de ser fabricado.

ACKNOWLEDGMENTS

I dedicate my sincere thanks:

- To my lovely wife Raquel for helping me in everything.
- To my parents Darío and Ana for giving me the life and take care of me.
- To my sisters Helena, Ana and Norma for encouraging and supporting my decisions.
- To my advisor Prof. Fernanda Lima Kastensmidt for giving me the chance to research at the PGMICRO.
- To my friends Jorge, Eduardo, Walter, Jimmy, Lucas, William, José, Felipe, Samuel, Gracieli, Anelise, Calebe, Guillerme, Jerson and Gustavo Wilke.

LIST OF FIGURES

Figure 1.1 General clock distribution networks for processors (Wilke, 2008). 15
Figure 1.2 Radiation induced jitter and race (Seifert et al., 2005). 16
Figure 2.1 Van Allen radiation belts. 19
Figure 2.2 Ionizing Dose: Cumulative effects... 20
Figure 2.3 Charge distribution in a gate oxide after expose to radiation (Lacoe, 2008). 21
Figure 2.4 Edge transistor current (Lacoe, 2008)... 23
Figure 2.5 I_D-V_G curves as a function of total dose for 0.35 μm minimum geometry NMOS transistor (Lacoe, 2008)... 24
Figure 2.6 I_D-V_G curves as a function of total dose for 0.18 μm (Lacoe, 2008)............ 24
Figure 2.7 Mechanism for heavy ions and protons in semiconductors (Lacoe, 2008)... 26
Figure 2.8 Energetic particle strike a p-n junction extending the depletion region as a funnel (Lacoe, 2008).. 27
Figure 2.9 SRAM core under ion strike. 28
Figure 2.10 Ionization upset threshold in SRAM core (hspice simulation 65nm)......... 28
Figure 2.11 SET propagation in an inverter chain (Dodd et al., 2004). 29
Figure 2.12 Logic masking.. 30
Figure 2.13 Electrical masking... 30
Figure 2.14 Latching-window masking... 31
Figure 2.15 PNPN structure in CMOS technology (Deepon Saha, 2010) 31
Figure 2.16 Equivalent circuit of CMOS latchup.. 32
Figure 2.17 Logic path mapping in the Flash-based FPGA. 33
Figure 2.18 Propagation delay of an inverter gate.. 34
Figure 3.1 Clock skew definition. 37
Figure 3.2 Clock skew in circuits.. 37
Figure 3.3 Jitter in clock waveform... 38
Figure 3.4 Floorplanning of structured custom VLSI circuit (Friedman, 2001). 39
Figure 3.5 General clock distribution networks in integrated circuit (Friedman, 2001). 39
Figure 3.6 Geometric matching for zero skew (Friedman, 2001) 40
Figure 3.7 Symmetric H-tree distribution (Wilke, 2008). 41
Figure 3.8 Clock mesh structure.. 42
Figure 3.9 Global mesh using H-tree distribution (Cadence, 2003)............................. 43
Figure 3.10 Effects of radiation particle strike on a regular clock path (Dash et al., 2009). 46
Figure 3.11 Inverter proposed: (a) conventional inverter, (b) hardened dual port inverter (Ming e Shanbhag, 2005). 47
Figure 3.12 Hardened dual port inverter as clock path (Ming e Shanbhag, 2005)......... 47
Figure 3.13 SEU hardened clock leaf inverter (Mallajosyula e Zarkesh-Ha, 2008). 48
Figure 3.14 TMR clock regenerator circuit (Dash et al., 2009). 48
Figure 3.15 Split-output inverter based clock regenerator hardening (Dash *et al.*, 2009). 49
Figure 3.16 Split-output SEU tolerant inverter 50
Figure 4.1 Proposed methodology to analyze SET susceptibility in clock networks..... 53

Figure 4.2 Output of EXT-CLK tool: clock network netlist and netlist separated in sections 55
Figure 4.3 Flowchart to logic cell subcircuits statement 56
Figure 4.4 Representation of a net 57
Figure 4.5 Example of a net description in DEF file 58
Figure 4.6 Algorithm to separate nets concerning the clock network 58
Figure 4.7 Asterisk convention used in net description 59
Figure 4.8 Algorithm to replace asterisks by numbers 60
Figure 4.9 Writing the net subcircuit headline. 60
Figure 4.10 Pi model for wire simulation in hspice (Rabaey et al., 1999) 61
Figure 4.11 Net description in DEF syntax (left) and Net description in netlist hspice (right). 62
Figure 4.12 Algorithm to create the net subcircuit. 63
Figure 4.13 Flowchart of the algorithm to call the subcircuit of components 65
Figure 4.14 Flowchart of the algorithm to call the subcircuit of nets. 67
Figure 4.15 Algorithm to define initial condition. 68
Figure 4.16 Algorithm to measure statement 69
Figure 5.1 Example of exponential source function 71
Figure 5.2 Clock Tree Network in SRAM arbiter circuit. 72
Figure 5.3 Minimum charge injected in the output of buffers to produce a bit-flip in at last one register. 74
Figure 5.4 Clock path in SRAM arbiter to be analyzed. 75
Figure 5.5 Critical charge results in the SRAM arbiter clock path 75
Figure 5.6 Result of the SET simulation replacing small devices 77
Figure 5.7 Percentage of registers failed due to induced charge injected in the clock source. 78
Figure 5.8 Bit-flip rate in registers due to SET injection by electrical simulation. 79
Figure 5.9 Error rate the circuit outputs due to SEU injections in the registers by logic simulation. 81
Figure 5.10 Soft error rate due to SET in the clock tree network. 82
Figure 5.11 SRAM arbiter layout using clock tree topology at 65nm. 83
Figure 5.12 Profile SRAM arbiter using clock tree topology. 83
Figure 5.13 Electric model of mesh. 84
Figure 5.14 SRAM arbiter layout using clock mesh. 85
Figure 5.15 Profile SRAM arbiter using clock mesh. 85
Figure 5.16 Comparison between clock tree and clock mesh 86

LIST OF TABLES

Table 2.1 Results of SET fault injection in FPGA prototype in critical path 33
Table 2.2 Result of SET propagation in critical path at hspice level......................... 34
Table 5.1 Types of Buffers 72
Table 5.2 Details of SET simulation replacing the small devices 76

LIST OF ABBREVIATIONS AND ACRONYMS

ASIC	Application Specific Integrated Circuits
BICS	Build-In Current Sensor
CMOS	Complementary Metal Oxide Semiconductor
CTS	Clock Tree Synthesis
DEF	Design Exchange Format
FF	Flip-Flop
FIT	Failure In Time
FPAA	Field Programmable Analog Array
FPGA	Field Programmable Gate Array
IC	Integrated Circuit
IGBT	Insulated Gate Bipolar Transistor
LEF	Library Exchange Format
LET	Linear Energy Transfer
LVS	Layout vs. Schematic
MBU	Multi-bit Upset
MOS	Metal Oxide Semiconductor
PIPB	Propagation Induced Pulse Broadening
PLL	Phase Locked Loop
PTM	Predictive Technology Model
SEB	Single Event Burnout
SER	Soft Error Rate
SEGR	Single Event Gate Rupture
SET	Single Event Transient
SEU	Single Event Upset
SEL	Single Event Latch
SEE	Single Event Effects
SPICE	Simulation Program with Integrated Circuits Emphasis
SRAM	Static Random Access Memory
TID	Total Ionizing Dose
TMR	Triple Modular Redundancy

VLSI	Very Large Scale Integration

TABLE OF CONTENTS

ABSTRACT 3
RESUMO 4
LIST OF FIGURES 6
LIST OF TABLES 8
LIST OF ABBREVIATIONS AND ACRONYMS 9
1 Introduction 14
1.1 Motivation 16
1.2 Thesis Proposal 16
2 Radiation Environment and Effects on Semiconductors Devices 18
2.1 Total Ionizing Dose (TID) 20
2.2 Gate-oxide effects mechanism 20
2.2.1 Threshold voltage shift 22
2.2.2 Radiation-induced edge leakage 23
2.3 Single Event Effects (SEE) 25
2.3.1 Single Event Upset (SEU) 26
2.3.2 Single Event Transient (SET) 28
2.3.3 Single Event Latchup (SEL) 31
2.4 Propagation induced pulse broadening (PIPB) 32
3 Clock Distribution Networks 36
3.1 Definitions 36
3.1.1 Skew 36
3.1.2 Jitter 37
3.2 Clock Distribution Architectures 38
3.2.1 Buffered Clock Tree Networks 39
3.2.2 H-Tree Clock Distribution Networks 41
3.2.3 Clock Mesh Distribution 41
3.3 Power Dissipation in Clock Distribution Networks 44
3.4 Radiation Effects on Clock Tree Networks 45
3.5 State-of-the-Art Mitigation Techniques to Clock Tree Networks 46
3.5.1 Hardened Dual Port Inverter 46
3.5.2 SEU Hardened Clock Leaf Inverter 47
3.5.3 TMR Clock Regenerator Circuit 48
3.5.4 Split-output Inverter Based Clock Regenerator Hardening 49

3.6 Problem Definition 51
4 Proposed Methodology to analyze the set susceptibility in clock networks.... 52
4.1 Proposed Methodology....................52
4.2 EXT-CLK tool....................53
4.2.1 Logic cell subcircuits statement.................... 55
4.2.2 Nets subcircuit statement.................... 56
4.2.3 Net subcircuit headline 60
4.2.4 Call subcircuit of components 64
4.2.5 Call statement of net 66
4.2.6 Initial condition statement 67
4.2.7 Measure statement 68
5 Fault Injection Simulations.................... 70
5.1 Case study: SRAM arbiter....................72
5.2 SET injection in every node of the clock tree 73
5.3 SET injection in a particular path of the clock tree....................75
5.4 Replacing small devices on clock tree 76
5.5 SET injection in the clock source....................77
5.6 Soft Error Rate methodology 79
5.6.1 Soft error rate due to SET in clock tree networks.................... 81
5.7 Clock Mesh vs Clock Tree 82
6 Conclusion 87
References.................... 88
Apêndice A: 94
Análise de single event effect em redes de relógio.................... 94
A.1 Introdução....................94
A.1.1 Motivação 95
A.2 Efeitos da radiação em componenes semicondutores....................96
A.2.1 Total Ionizing Dose.................... 96
A.2.2 Mecanismo de TID sobre o gate 97
A.2.3 Deslocamento da voltagem umbral.................... 98
A.2.4 Corrente de fuga nas bordas.................... 99
A.2.5 Single event effects (SEE) 99
A.3 Redes de relógio em semicondutores 102
A.3.1 Árvore de relógio 102
A.3.2 Rede de distribuição H.................... 103

A.3.3 Rede de relógio em malha (mesh) 104
A.4 Metodologia proposta para a análise do SET em redes de relógio..................105
A.4.1 Metodologia proposta 105
A.4.2 Ferramenta EXT-CLK 106
A.5 Simulação de injeção de falhas...108
A.5.1 Injeção de SET em cada nó.. 109
A.5.2 Injeção de SET num caminho da rede 110
A.5.3 Substituição de buffers pequenos por maiores....................................... 111
A.5.4 Injeção de SET no início da rede 112
A.5.5 Soft error rate na árvore de relógio 113
A.5.6 Comparação entre rede de árvore e rede de malha 115
A.6 Conclusões...118

1 INTRODUCTION

The semiconductor manufacturing is becoming a major industry in the world. Success in the industry requires constant attention to the state of art in each stage. Techniques in chemistry and physics must be improved over the time. Semiconductor industry makes possible the production of integrated circuits (IC) to all electronic devices, like computers, mobile phones, tablets and other digital home appliances that are now inextricable parts of the modern society.

In the early days of integrated circuits, only a few transistors could be placed on a chip, because the scale used was large due to the contemporary technology and manufacturing yields were low compared to the current standards. As the level of integration was small, the design process was relatively simple. Over time, the numbers of transistors placed in the same area was improved to millions and billions. Then, in the early 1980s the term Very Large Scale Integration (VLSI) was introduced as the final step in the development process of design. This new process allows the creation of complex design such as Applications Specific Integrated Circuit (ASIC) or System-on-chip (SoC) that have multiple interfaces running with multiple asynchronous clocks, having frequency as high as multiple gigahertz.

Any error introduced in any stage of the manufacturing process may appear on the final electronic testing, and consequently the device may be lost. There are other errors that can be introduced when the IC is in operation, during its lifetime. In radiation environment the presence of heavy ions, protons and electrons, may introduce new type of errors. Particle strikes can ionize directly or indirectly the ICs provoking transient and permanent faults (Schrimpf, 2007).

Cosmic radiation is gamma radiation that has emanated from the sun and other terrestrial bodies traveled to our planet. On its journey, cosmic rays have not to pass through any substantial matter, only space. Hence, it can travel large distances without being absorbed. Once the radiation reaches our atmosphere it begins to be absorbed by the air, thus reducing the radiation level on the ground. Radiation belts, solar flares and cosmic rays are the main source of the space radiation. Protons, neutrons, electrons and heavy ions in space, are responsible for interference and malfunction of transistors operating in space. These charged particles pass through the semiconductor junctions and liberate electrons from the atoms. The effects of radiation on semiconductors have been classified as Single-Event Effects (SEE) (Barnaby et al., 2008). SEE caused by atmospheric radiation have been recognized as a design issue for avionics systems, as well as for high reliability ground-based systems due to the interaction with neutrons. If this transient effect occurs inside a memory cell it is known as Single Event Upset (SEU) and if the effect occurs in a combinational logic it is known as Single Event Transient

(SET). When SET and SEU are not logically masked by the application, they can provoke faults in the circuit functional behavior (Wissel et al., 2009) (Battezzati et al., 2009).

Decreasing the size of the transistors decreases also their capacitance, making them more susceptible to disruption caused by charge generated by radiation. Studies have indicated that high frequency operations with supply voltage reduced exhibit higher soft error rates (SER) (Juhnke e Klar, 1995). All the circuitry present in the clock tree networks is becoming vulnerable to radiation effects as well. SET may also occur at the output node of the clock buffers (Chellappa et al., 2011). Transient pulse can propagate through the network depending on the width and the amplitude of the pulses (Lacoe, 2008). Figure 1.1 depicts the scheme of a complex clock distribution topology of a modern microprocessor which consists of a root and many branches and buffers.

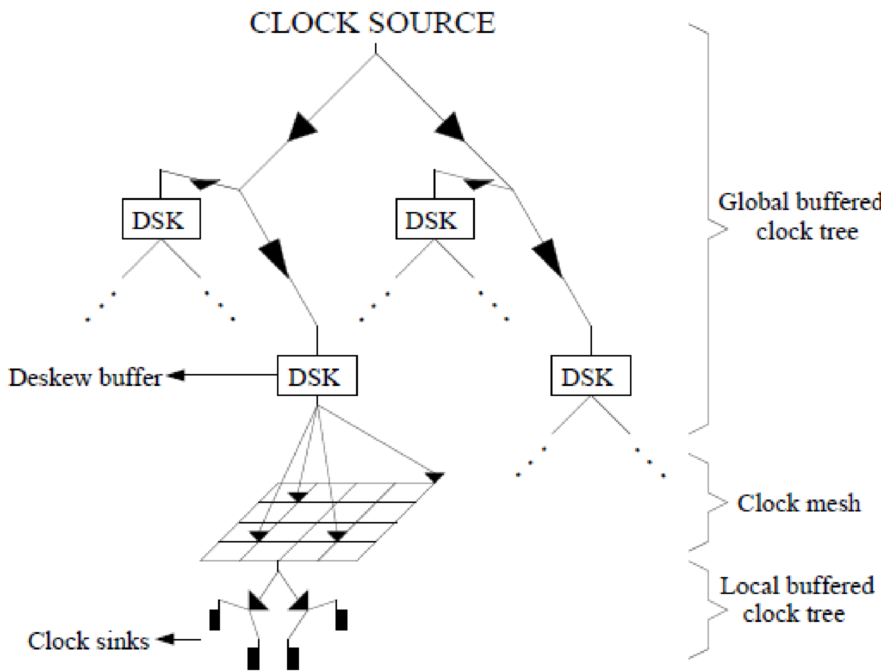


Figure 1.1 General clock distribution networks for processors (Wilke, 2008).

The effects of these transient pulses may be translated to clock glitches, clock jitter and clock skew (Ming e Shanbhag, 2005). Some authors classify this glitch as clock race (Seifert et al., 2005) (Dash et al., 2009). If the SET reaches one or more register elements, depending on the topology, those may have the stored value changed, resulting in a bit-flip.

In Figure 1.2, the additional clock pulse (race) results in a premature latching of the input data, which means that the output node will change the value early, becoming an erroneous data. This phenomenon may be a result of a particle strike. Depending on the type the application, these effects may alter the control or data flow of the circuit, resulting in an error.

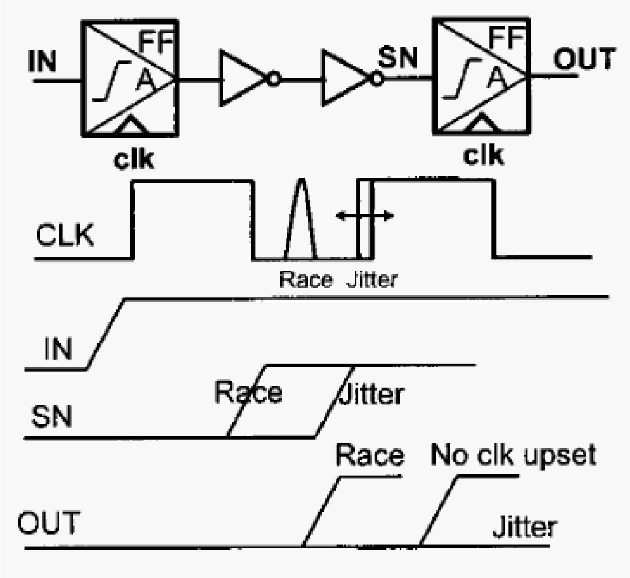


Figure 1.2 Radiation induced jitter and race (Seifert et al., 2005).

1.1 Motivation

As technology scaling is rapidly increasing, circuit designs become increasingly complex. Synchronizing each block or stage is a priority. Consequently clock distribution network is very relevant to ensure the correct behavior of the circuit when the clock signal must be delivered. Especially when the integrated circuit will work in aerospace, because radiation may generate single event effects over the clock network. This effect may be translated as a clock pulse and reach register elements storing incorrect data. There are techniques to analyze and improve the design against radiation effects, but there is a lack of techniques to analyze the susceptibility of single event effects focused in clock distribution networks which is our main motivation.

1.2 Thesis Proposal

Before developing any optimization technique for clock distribution networks resistant to radiation effects, it is necessary to develop an efficient analysis of susceptibility to single event effects (SET) in clock networks. Therefore, we investigate a method for accurate analysis of SET susceptibility in different clock network topologies. Then, we decide to elaborate a tool to extract the clock distribution network from the layout design and perform the analysis of susceptibility.

Hence, we evaluate the SET susceptibility in clock distribution networks of ASIC design. The main contributions of this work are presented below:

- Development of EXT-CLK tool, which allows the extraction the clock distribution network from the layout design to electrical model. This tool is able to extract different clock networks design such as clock tree, clock mesh and H-topology. The results obtained from this tool allow many types of SET simulations using hspice.

- Simulation of SET injection over different nodes in the clock distribution networks. These simulations will result in a profile of the susceptibility of the clock network to a specific circuit design. This profile helps us to identify vulnerable nodes in terms of critical charge (Q_c).
- Simulation of SET injection in a particular path of the clock network, which allows making a profile of a path of the clock signal, showing the relation between the charges needed to provoke a failure and size of the cell used in the path.
- SET injection simulation in the clock source. Through this simulation, we identify the susceptibility of each register of the circuit in response to a given value of charge injected in the root the network. The result shows the minimum value of charge to achieve a failure and the maximum value to get 100% of fail on the registers. The same simulation allows the identification of the percentage of fail to each register and identifying the most sensitive register.
- Replacing smaller buffers. We show a simple experiment to evaluate the relation between size and susceptibility of buffers used in the clock network. We replace smaller buffers by larger and compute the SET injection simulation.
- Simulation of soft error rate in clock networks. This simulation combines electrical level simulation and logic level simulation. The results obtained are the soft error rate of each register of the clock network taking into account the masking effects of the application level.
- Comparison between clock mesh and clock tree networks. We take a case study circuit in clock tree and developed the same circuit using clock mesh. After the extraction of the clock mesh, we make the profile of susceptibility and compare with the clock tree version. The result of the simulation shows a remarkable difference.

In this work, we present a new method for evaluating SET caused by radiation effects in clock distribution networks. In order to perform this method, we introduce EXT-CLK, a new tool developed in Bash command line for Linux environment and OS X environment (Mac OS). The main feature of EXT-CLK is the capability of performing several simulations to identify the susceptibility of the clock network. The simulations are focused in nodes, buffers, gates, and registers, which assists the production of a profile for the circuits. This tool can be used to perform accurate and efficient analyses of susceptibility of single-event effects (SET) in clock networks. As case study, SRAM arbiter circuit was selected to evaluate the proposed analysis method.

This work is organized as follows: the next chapter describes the radiation effects on semiconductor devices. In chapter 3 we discuss the types of clock distribution networks. In chapter 4 we describe in detail the EXT-CLK tool and how it works. In chapter 5 we show many SET simulation results obtained from the clock network. Finally, in chapter 6 we discuss the results and how this work can help designers to optimize the clock network for resisting against radiation effects.

2 RADIATION ENVIRONMENT AND EFFECTS ON SEMICONDUCTORS DEVICES

Looking back to the historical event, the first artificial satellite by US, Explorer I, launched on January 31, at 1958, carried a cosmic rays counter. The counter suddenly stopped to count cosmic rays when the spacecraft reached some altitude. It was later found that the counter was in fact saturated by an extremely high particle count rate. This was the day of the discovery of the Van Allen belts (Garber, 2007); that event was one of the outstanding discoveries of the International Geophysical Year. The evidence of trapped particles in Earth's radiation belts can be considered as the very first scientific outcome of the Space Age. The radioactive phenomena encountered in space are classified into four categories by source of origin: Radiation belts, solar flares, solar wind and cosmic rays.

- Radiation belts: The belt contains a combination of trapped electrons and protons from the outer belt to inner belt, as shown in Figure 2.3. The inner belt contains electrons with energy lower than 5 MeV and the outer belt may reach up to 7 MeV (Boudenot, 2007).
- Solar flares: Are a sudden brightening observed over the surface of the Sun, which is interpreted as a large energy release up to 6×10^{25} joules. Radiation is emitted across the entire electromagnetic spectrum from long wavelength up to gamma rays. Then, electrons, protons, and heavy ions are heated and accelerated in the solar atmosphere and reach the space environment.
- Solar wind: Is a stream of charged particles released from the upper atmosphere of the Sun. It mostly consists of electrons and protons with energy between 1.5 and 10 KeV. These particles can escape from the gravity of Sun caused by their high kinetic energy.
- Cosmic rays: Are very high-energy particles, mainly originated outside the Solar System. These particles may produce secondary particles, which penetrate and impact the Earth's atmosphere and sometimes even reach the surface of the Earth. Cosmic rays are originated from supernovas, but this is not thought to be their only source.

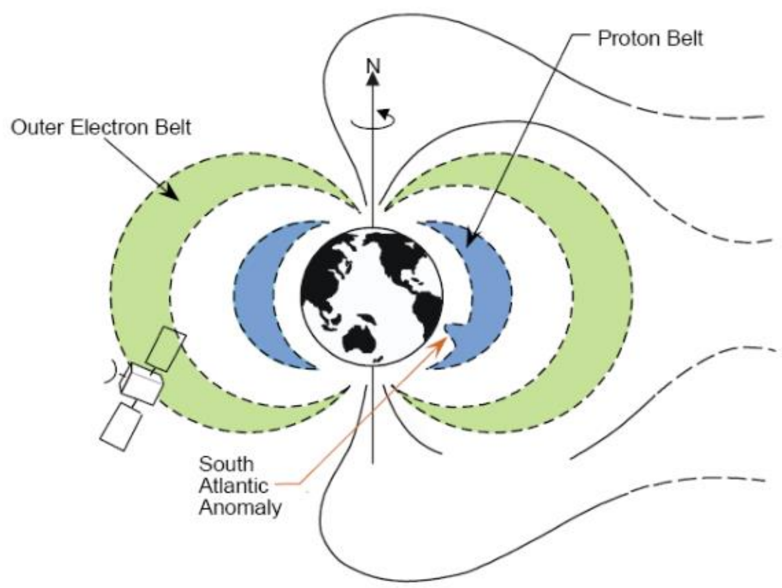


Figure 2.1 Van Allen radiation belts.

The flow of high-energy particles experienced in orbit and flight altitude demands the shielding of electronics with component redundancy, and radiation hardening. Thereby, this can increase their cost and reduce their performance. One of the effects of Total Ionizing Dose (TID) is an accumulative effect in the silicon oxides, which degrades the performance of the transistors, similar to an aging effect.

As the dimensions and operating voltage of commercial ICs are reduced to satisfy the increasing demand for high density, and low power, their sensitivity to radiation has to increase dramatically. Recently, soft errors have become a big concern in advanced commercial ICs because they can increase the product failure rate. In a qualified manufacturing silicon process, the typical failure rate for the hard reliability mechanisms is 10-100 failure-in-time (FIT), while the soft error rate can easily exceed 50,000 FIT (where one FIT is equivalent to one failure in a billion device hours) (Schrimpf e Fleetwood, 2004). For a single chip customer system even a soft error rate of 50,000 FIT is usually not problematic, but this failure rate is intolerant for high reliability systems with multi-chip assembly.

In the late 1970s, alpha particles emitted from the natural radioactivity decay of uranium, thorium and daughter isotopes present as impurities in package materials were found to be the predominant cause of soft error rate in DRAMs (Dodd e Massengill, 2003). During the same period, it was demonstrated that ionizing reaction products created from the interaction of cosmic neutrons with electronic material could cause soft error. In the mid-1990s, it has been established that the high energy cosmic radiation was the dominant error source in the DRAMs devices. Soft error was identified from low energy cosmic neutrons interactions with ^{10}B in devices material, especially in the borophosphosilicate glasses (BPSG), used extensively as insulating layers in integrated circuits. This mechanism has exhibited to be the dominant soft error rate in 0.25 and 0.18 µm CMOS SRAMs (Kobayashi et al., 2002). Despite of low levels of compounds found in package materials, sufficient alpha particles are generated to cause a significant rate of upsets in SRAM FPGAs (Actel, 2007). But after identifying the source of radiation on

package materials, high purity materials are employed to reduce the alpha emission. In devices requiring high reliability, uranium and thorium impurities have been reduced below one hundred parts per trillion. For conventional integrated circuits packaging, the reduction of material alpha emission was from 5-10 alphas/cm^2-hr to less than 0.001 alphas/cm^2-hr. To reduce soft error rate induced due to the ^{10}B activation by low energy neutrons, BPSG is replaced by other insulators that do not contain boron. When all these techniques are used, the soft error rate of the integrated circuit is reduced significantly, but ultimately limited by cosmic high-energy neutrons interactions that cannot be easily prevented.

The nature of the interaction between semiconductors and high-energy particles depends on the properties of the particle such as its charge, energy, momentum, and target material: density and mass. Other parameters that determine the damage are the flux and the fluency of particles over the device in exposition. Spacecraft anomalies can be divided mainly in two effects: Total Ionizing Dose (TID) affects associated with trapped particles in the magnetic Van Allen belts. And the second, Single Event Effect (SEE) resulting from the interaction of highly energetic particles (including protons, cosmic rays and other heavy particles). These can be a source of significant data corruption in the FPGAs (Actel, 2007) and multiple devices.

2.1 Total Ionizing Dose (TID)

Exposure to radiation produces long-term changes in device and circuit characteristics that may result in parametric degradation or functional failure. Total Ionizing Dose (TID) mainly affects insulating layers, because trapped charges degrade the electrical performance. Non-ionizing energy results in displacement damage and defects in both insulator and semiconductor regions. In older technologies, these effects were described by a spatially uniform representation of the cumulative amount of energy deposited. The accuracy of this description relies on the relatively large size of the devices and the energy deposit by individual particles or photons. In very small devices, less than 130 nm, this approach is no longer valid (Schrimpf, 2007). A simple way to explain the TID effects is shown in Figure 2.2. The consequence of TID in CMOS transistors are: threshold voltage shift, leakage current and isolation oxide. But first we will describe the mechanism of gate-oxide effects.

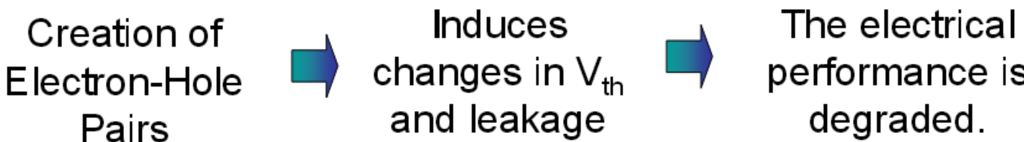


Figure 2.2 Ionizing Dose: Cumulative effects

2.2 Gate-oxide effects mechanism

The cumulative damage incurred in semiconductor due to exposure to radiation is referred as the total dose effect. This mechanism can be described in four steps. In the first step, the ionizing radiation creates electrons-holes pairs. When an energetic particle

(proton, electron, or heavy ion) impacts a semiconductor. The particle loses energy at a constant rate as it passes through the solid material, as long as it is not near the end of its range. The mechanism by which electrons and protons lose energy, is primarily by inelastic Coulomb scattering in which the incident particle ejects an outer shell electron from an atom. The incident particle repeats this scattering process as it continues through the solid producing a line of electrons-holes pairs. The rate at which a particle loses energy is normalized by the density of the material in which the energy is deposited, and it is referred as linear energy transfer (LET) rate. The elements most sensitive to total dose effects from ionizing radiation in a CMOS device are the gate and isolation oxides, which are most often fabricated with SiO_2.

Immediately after the creation of a line of electrons-holes pairs, a fraction of the pairs recombine. The temporal window during where recombination occurs is very short and limited by the time of electron to transit and be removed from the gate oxide (typically less than 0.1 ps). The recombination occurs in response of the gate electric field. Even with no voltage applied to the gate, recombination will be completed in at most a few picoseconds due to the built-in gate electric field. The amount of recombination depends on the density of electrons-holes pairs and the applied electric field. For high LET particles, the density of electrons-holes pairs is high, the Coulomb interaction between an isolated electrons-holes pair is effectively screened out, and the fraction of electrons-holes pairs that recombine can be high. For lower LET particles, the density of electrons-holes pairs is less, the interaction between different electrons-holes pairs is negligible, and the Coulomb interaction between the isolated electron and hole of a pair dominates the recombination process. The effect of an electric field on the recombination processes is to separate the electrons and holes, which will result in less recombination. Furthermore, both electrons and holes that are within a characteristic tunneling length of SiO_2, 4-5 nm are tunnel out of the gate oxide rapidly after a radiation pulse.

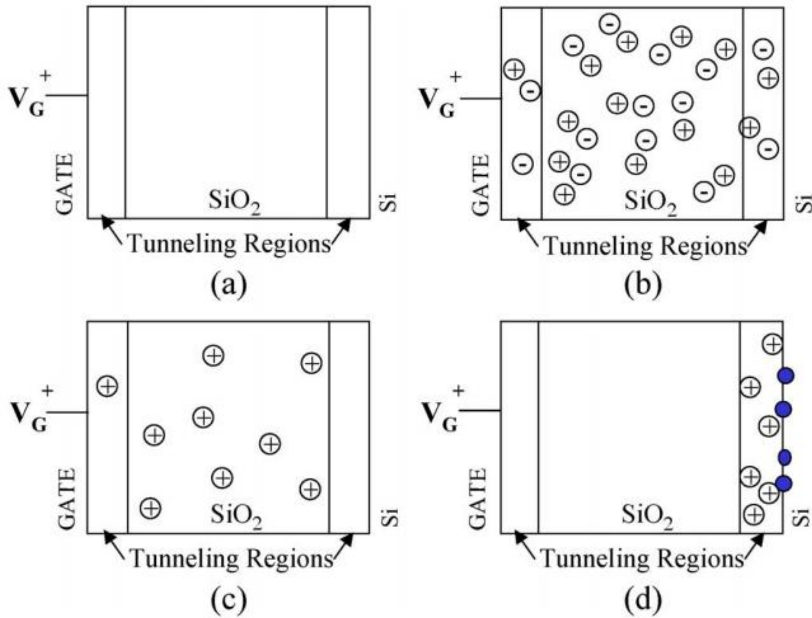


Figure 2.3 Charge distribution in a gate oxide after expose to radiation (Lacoe, 2008).

At this point, the only charges that left in the gate oxide are holes, as shows Figure 2.3 (c). The second step is the drift of the remaining holes to the oxide/silicon interface, due to electric field from the positively biased gate electrode. This process has been studied in depth, and it has been determined that the transport by holes is primarily by hopping between localized sites in the gate oxide. This process is both thermally and field-activated. The holes transported are highly dispersed, occurring over many decades in time after a radiation pulse. The transport of these holes to the interface occur via polaron hopping where the holes become self-trapped by their own deformation potential.

In the third step, a fraction of the holes are trapped at the interface. This oxide-trapped charges are positively charged, and can be neutralized over time by either electron tunneling from the silicon or by thermal emission of an electron. The activation energy for this process is relatively low, significant recovery can occur for radiation exposure over a long time period such as for a space mission

In the fourth step, the formation of “interstate traps” can occur. A simplistic view is that interface states are associated with dangling bonds between the silicon and the SiO_2. Interface states exist within the silicon band gap at the interface with SiO_2. For NMOS transistors, the interface states act as negative charges in the gate-oxide of a NMOS transistor, or positive charges in the gate-oxide of a PMOS transistor. These steps are described pictorially in Figure 2.3 (d).

2.2.1 Threshold voltage shift

The effect of introducing charges in the gate oxide and/or at the gate-oxide/silicon interface is to shift the threshold voltage ΔV_T in CMOS transistors. The shift in threshold voltage is determined by integrating the weighted additional charge density (ρ) over the oxide thickness (t_{ox}).

$$\Delta V_T = \Delta V_{ot} + \Delta V_{it} = \frac{-1}{C_{ox} t_{ox}} \int_0^{t_{ox}} x\, \rho(x) dx \qquad \text{Equation 2.1}$$

Equation 2.1 describes the shift threshold voltage, where $x = 0$ represents the polysilicon/gate oxide interface, and C_{ox} is the gate capacitance. The trapped holes charges in oxide (*ot*) is always positive. Hence, ΔV_{ot} for both NMOS and PMOS transistor are always negative. For NMOS transistors, interface trapping charge (*it*) is negative, while for PMOS transistors, interface trapping charge is positive. The net threshold voltage shift ΔV_T will be the sum of ΔV_{ot} and ΔV_{it}. The amount of charge generated in the oxide by radiation is proportional to the oxide thickness. In addition, the effect of the oxide charge on the threshold voltage is proportional to the distance of the charge from the gate electrode, as describes the Equation 2.2

$$\Delta V_T = -\frac{Q_{ot}}{C_{ox}} \propto x_{ox}^2 \qquad \text{Equation 2.2}$$

Thus, thin oxide, such as modern gate oxides, are much less sensitive to ionizing radiation than field oxides or older gate oxides. Thin gate oxides are even less sensitive to total ionizing dose than this estimate would suggest because most of the charge within 5 nm of the interface is quickly removed by tunneling (Schrimpf, 2007).

2.2.2 Radiation-induced edge leakage

Transistors are electrically isolated from one another by a thick field oxide (FOX). The edges for a standard transistor layout are defined by the gate oxide interfaces with the FOX. Figure 2.4 shows an off-axis view of a standard MOS transistor. Since the quality of the oxide near the FOX/gate interface is poorer than that of the gate oxide, it may be more efficient in trapping charge. Radiation-induced trapped holes in these edge transistors can result in a negative shift in the threshold voltage, then if this is large enough, the edge transistors becoming conductive resulting in a source-drain current in the transistor off state. That off-state current can increase due to radiation-induced edge effects. The consequences of increase off-state current may be result in signal corruption and reduced margins of work. In some cases this can lead to functional failure, as the transistors are always active. Edge leakage is not a problem for PMOS transistors, for PMOS transistors the effect of oxide-trapped charge and interface trapped charge are to shift an already negative threshold voltage even further in the negative direction.

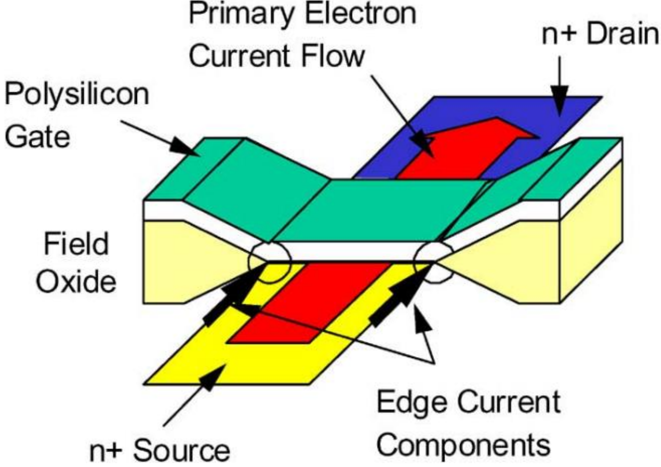


Figure 2.4 Edge transistor current (Lacoe, 2008).

Edge leakage in a commercial CMOS process could exceed 1 μA for total dose levels as low as 10 krad(Si). The I_D-V_G characteristics as a function of total dose for a 0.35 μm (minimum geometry) NMOS transistor fabricated by Chartered Semiconductor are show in Figure 2.5. This process uses local oxidation of silicon (LOCOS) and has a gate-oxide thickness of 7.6 nm. For exposure up to 50 Krad, the I_D-V_G characteristics no change. At 70 Krad, there is an increase in the off-state current of approximately 10^{-11} A. At 100 Krad, the off-state current has increased a little more than 1 nA. At 300 Krad the off-state current has increased approximately 5 times the drive current.

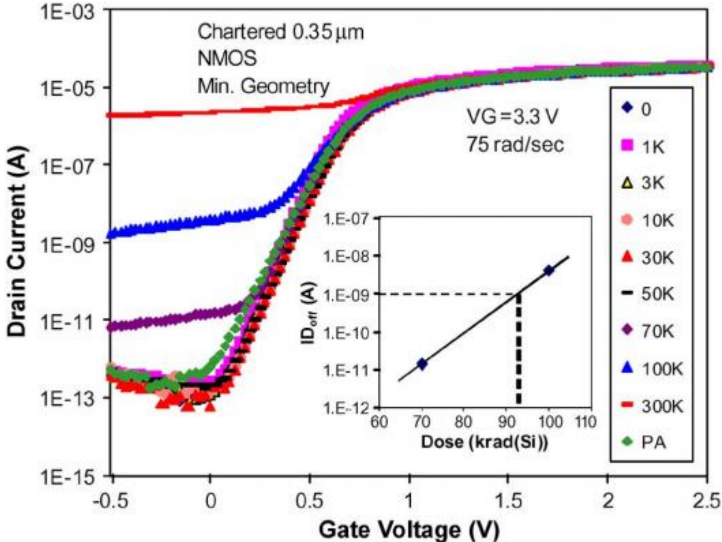


Figure 2.5 I_D-V_G curves as a function of total dose for 0.35 µm minimum geometry NMOS transistor (Lacoe, 2008).

Figure 2.6 shows the I_D-V_G characteristics as a function of total dose for a 0.18 µm (minimum geometry) NMOS transistor fabricated by TSMC. This process uses Shallow Trench Isolation (STI) technology and has a gate-oxide thickness of 3.2 nm. The curves are similar to 0.35 µm process, but for 0.18 µm the total-dose hardness level is approximately 345 krad under high dose rate. The drain current in 0.18 µm is 1 nA at approximately 350 krad. While, for 0.35 µm process, 1 nA is caused by 100 krad.

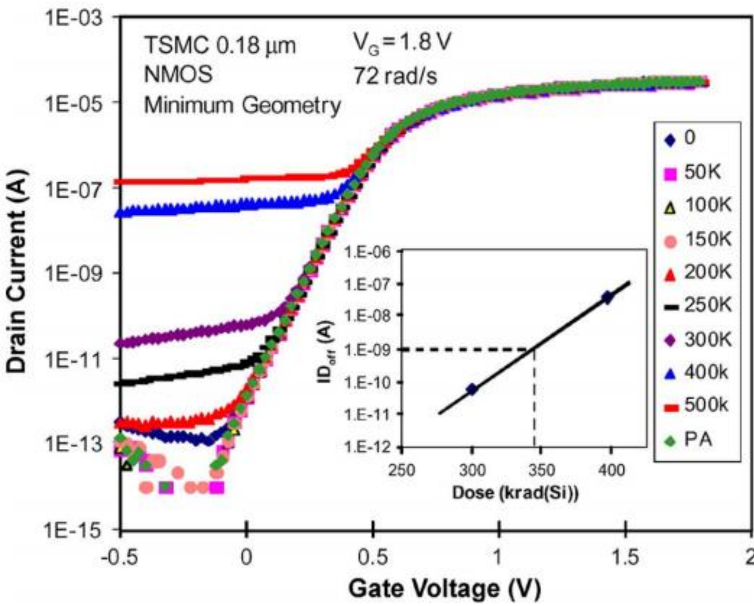


Figure 2.6 I_D-V_G curves as a function of total dose for 0.18 µm (Lacoe, 2008).

In older technologies, TID effects are described by the cumulative amount of energy deposited. In very small devices, less than 130 nm, this approach is no longer valid, as has been shown in Figure 2.5 and Figure 2.6 (Schrimpf, 2007).

After explaining the effects of TID on CMOS devices, the semiconductor industry is addressing this issue by researching potential new materials to replace SiO_2 as the gate dielectric material. The key is material with dielectric constant (K) values larger than SiO_2. This potential gate dielectric material is often referred as high-K dielectric. Because the dielectric constant is larger than SiO_2, the thickness can be increased to maintain the same capacitance while decreasing the tunneling current. Among the materials under consideration are hafnium oxide Hf-O/SiO_2/Si and aluminum oxides Al_2O_3/SiO_xN_y/Si (Lacoe, 2008). The TID effects on capacitors and transistors fabricated with high-K gate dielectrics indicate that while for thick oxides they can trapping efficiency. For test structures with dielectric thickness less than 5 nm, the amount of charge trapping is very small, resulting in very small shifts. Since the high-K dielectric thickness in advanced CMOS technologies be less than 5 nm, the results look encouraging using these technologies for space applications.

2.3 Single Event Effects (SEE)

Single event effects is a phenomena which result from the interaction between a single energetic particle and sensitive region of a microelectronic circuit. This particle can be energetic protons, neutrons or ions (Munteanu e Autran, 2008). These effects are classified in hard errors and soft errors. Hard errors are non-recoverable errors, while soft errors may be recoverable by a reset, a power cycle or simply rewriting the information. Hard errors such as Single Event Burnout (SEB) in power MOS, IGBT or power bipolar transistors, single event gate rupture (SEGR), micro-dose induced threshold voltage variation in MOS transistor, are not be discussed in this work, but they represent also an important issue in SEEs.

Soft errors include a great variety of manifestation depending upon the device considered. In memory devices, SEU is manifested in latch and register. This correspond to a bit-flip of the one cell. If more than one bit-flip occurred by a single event, a Multi-bit Upset (MBU) is obtained. In analog devices SETs are mainly transient pulses in operational amplifiers, comparator or reference voltage circuits. In combinatorial logic, SETs are transient pulses generated in a sensitive node that may propagate in a combinatorial circuit and eventually be latched in a storage cell. In bulk CMOS technology, PNPN parasitic structure may be triggered giving a Single Event Latch-up (SEL). SEL is associated with a strong increase of power supply current. SEL can be destructive by overheating of the structure and localized metal fusion.

In this section we will focus on devices and nondestructive SEEs. Also we will describe the mechanism of SEE with particular emphasis on SEU in static random access memories (SRAM) and SET in logic circuits. The only destructive SEE that we describe will be SEL.

2.3.1 Single Event Upset (SEU)

Single event upset (SEU) is defined by NASA as "radiation-induced errors in microelectronic circuits caused when charged particles lose energy by ionizing the medium through which they pass, leaving behind a wake of electron-hole pairs" (Baloch et al., 2006). SEUs are non-destructive events and a simple reset or rewriting results in normal device behavior. This phenomenon may occur in a digital circuit when a particle strike causes data to change states in a storage element such a flip-flop, latch or memory bit (Vargas e Nicolaidis, 1994). Direct ionization is the primary charge deposition mechanism for upsets caused by heavy ions, which are rather loosely defined as an ion with atomic number $Z \geq 2$. Lighter particles, such as protons and neutrons, usually do not produce enough charge by direct ionization to cause upsets in memory circuits, as show in Figure 2.7, although this can change with the technologies scale and the critical charge to create an upset decreases. These particles can, however, produce upsets due to indirect mechanisms. When a high-energy proton or neutron strikes a semiconductor, it may undergo an inelastic collision with a target nucleus. This can generate a nuclear reaction resulting in the emission of alpha particles or gamma particles and the recoil of a daughter nucleus, or a spallation reaction in which the target nucleus is broken into two fragments, each of which can recoil. Any of these reaction products can now deposit charge along their paths by direct ionization, because these particles are much heavier than the original protons and neutrons, they can deposit greater linear charge density as they travel through the semiconductor, which is more likely to cause upsets than the ionization created by the original particle. A depict of the mechanism is shown in Figure 2.8 (Lacoe, 2008).

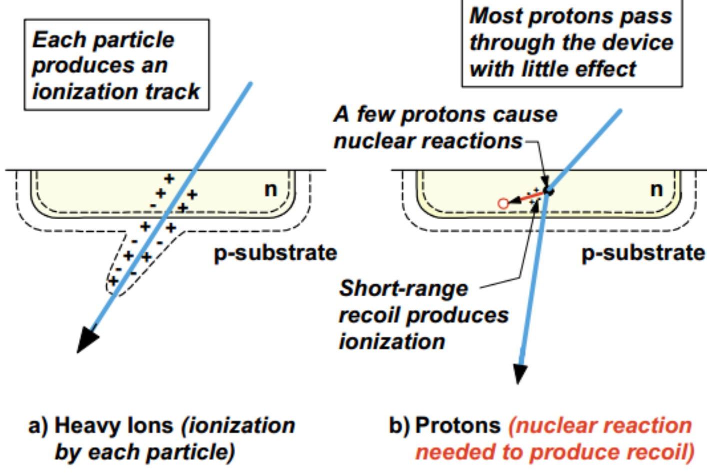


Figure 2.7 Mechanism for heavy ions and protons in semiconductors (Lacoe, 2008).

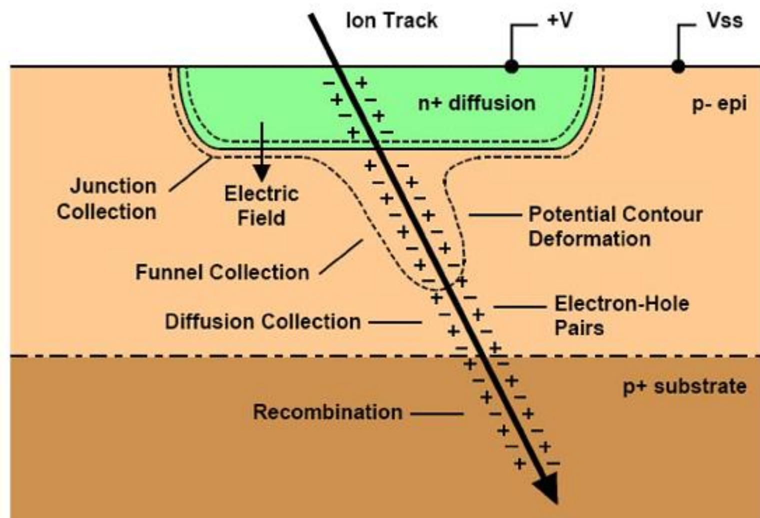


Figure 2.8 Energetic particle strike a p-n junction extending the depletion region as a funnel (Lacoe, 2008).

When the heavy ion strikes a p-n junction, the most sensitive sites for upset in bulk CMOS, creating a funnel extension of the depletion region under the junction. The funnel is caused by the projection of the electrostatic potential of the heavily doped surface electrode along the wire charge. In response, the current flow creates an effective short across the p-n junction for the period of time needed to dissipate the generated charge, typically a few nanoseconds. The later contribution of the current flow is associated whit diffusion of the charge toward the critical node. The impact of this current on the circuit operation depends strongly on the location and angle of the particle trajectory though the circuit, and the circuit response of an inadvertent current pulse at the particle location. The SRAM is a useful example of an SEU sensitive component.

The upset in SRAMs is due to the active feedback in the cross-coupled inverter pair that forms a typical SRAM memory cell, shown in Figure 2.9. When an energetic particle strikes a sensitive location in a SRAM (typically the reverse-biased drain junction of a transistor biased in the "off" state, for example the "off" n-channel transistor shown in Figure 2.9, charge collected by the junction results in a transient current in the struck transistor. As this current flows through the struck transistor, the restoring transistor ("on" p-channel transistor in Figure 2.9.) attempt to balance the particle-induced current. Unfortunately, the restoring transistor has a finite amount of current drive, and equally importantly, a finite channel conductance. The current flows through the restoring transistor therefore induces a voltage drop at its drain. This voltage transient in response to the single-event current transient is actually the mechanism that can cause upset in SRAM cells (Dodd e Massengill, 2003). The voltage transient is essentially similar to a write pulse and can cause the wrong memory state to be locked into the memory cell.

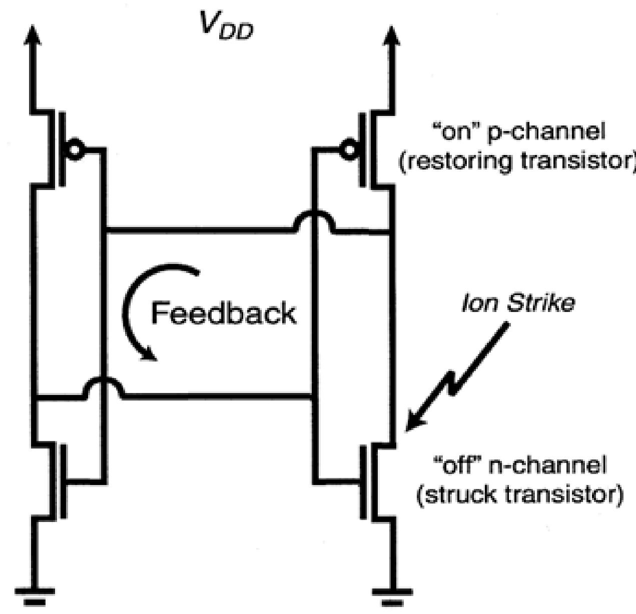


Figure 2.9 SRAM core under ion strike.

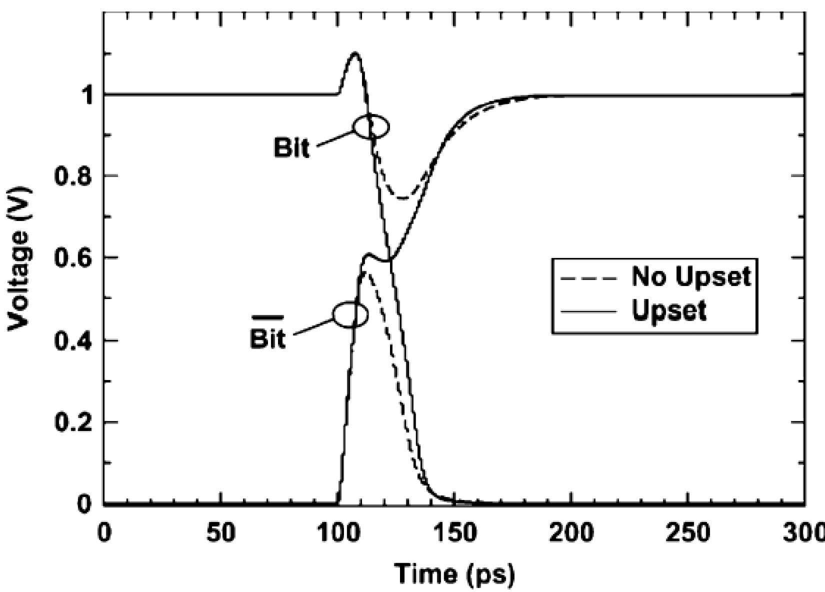


Figure 2.10 Ionization upset threshold in SRAM core (hspice simulation 65nm).

Figure 2.10 shows the drain voltage waveform of two side of the SRAM cell. The value data change (upset) when the ionization density reach the critical charge. The dashed lines represent when the ionizing density is slightly lower (no upset) and this cannot change the data.

2.3.2 Single Event Transient (SET)

A single event transient (SET) is the result of a voltage disturbance on a signal line caused by a particle strike. This single event occurs on non-latched elements such as

combinatorial logic and clock line or global control lines (Loveless et al., 2012). The voltage transient generated by this phenomena can propagate any significant distance through the combinatorial logic depending of the width of the transient pulse and the capacitance of the transistor (Wirth et al., 2007). Combinatorial circuit can interpret a SET voltage as a high-level state by the short time and produce unwanted behavior in the circuit. With the technology scaling the combinatorial circuit are more sensitive to transient pulse width narrow. The width of the SET is one of the main factors that determine whether an SET will result in an upset (fault) or not (Narasimham et al., 2009). According to Dodd (Dodd et al., 2004), which recall the SET as digital-SET (DSET) at least there are four criteria that must be met for a phenomena to result in a circuit error.

- The particle strike must be generate a transient pulse capable to propagating through the circuit.
- There must be an open logic path through which the DSET can propagate to arrive at a latch or other memory element.
- The DSET must be of sufficient amplitude and duration to change the state of the latch or memory element.
- In synchronous logic, the DSET must arrive at the latch during the latching edge of the clock.

The probability that transient pulse glitches will be captured as valid data in combinational logic increase linearly with frequency because the frequency of clock edges increases. As circuit speeds increase, it is also expected that the ability of a given transient pulse to propagate increases, however, one might also conjecture that the duration of transient pulse decreases. Due to both their greater ability to propagate in high speed circuits and their higher probability of capture by subsequent storage elements, such as latches, this phenomena have been predicted to became a very important issue in deep submicron digital ICs.

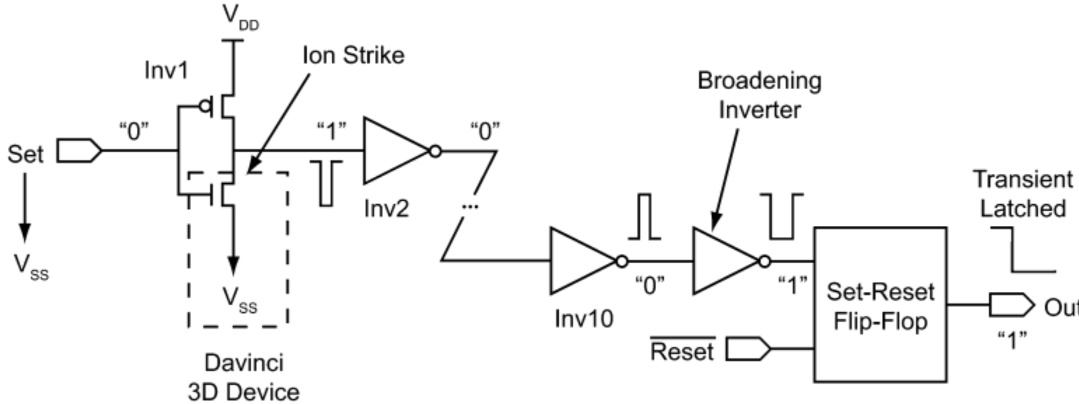


Figure 2.11 SET propagation in an inverter chain (Dodd et al., 2004).

Figure 2.11 shows an example of combinational circuit that is under ion strike in the off-biased n-channel transistor. The transient pulse generated in the first inverter travel from the second inverter through the chain of inverter to flip-flop. Depending of the

properties of each inverter the transient pulse may be broadening or attenuated and finally reaching the flip-flop and storing a data invalid.

On the other hand, a transient pulse will not affect the results of a computation unless it is captured in a memory circuit. Therefore, Shivakumar (Shivakumar et al., 2002) defines a soft error in combinational logic as a transient error in the result of a logic circuit that is subsequently stored in a memory circuit. A transient pulse in a logic circuit might not be captured in a memory circuit because it could be masked by one of the following three phenomena.

- **Logic masking** occurs when a particle strikes a portion of the combinational logic that is blocked from affecting the output due to a subsequent gate whose result is completely determined by its other input values. An example is shown in Figure 2.12. Other example of logic masking is when the combinational logic have a clock gate blocking the signal in case of clock networks.

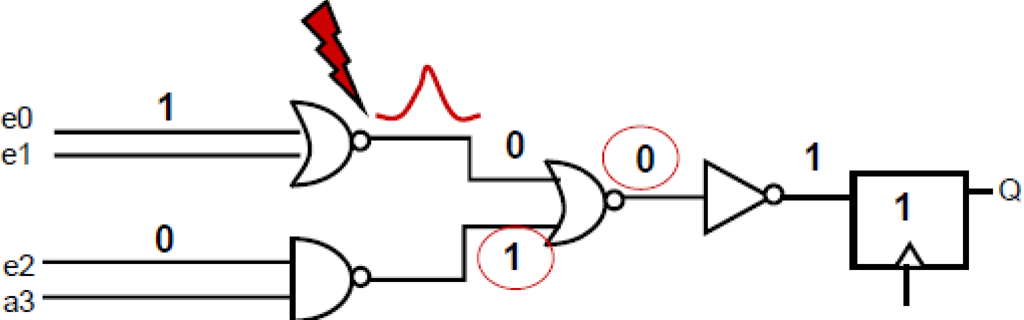


Figure 2.12 Logic masking.

- **Electrical masking** occurs when the transient pulse, resulted from a particle strike, is attenuated by subsequent logic gates due to the electrical properties of the gates, to the point that it does not affect the result of the circuit. Figure 2.13 shows an example of electrical masking.

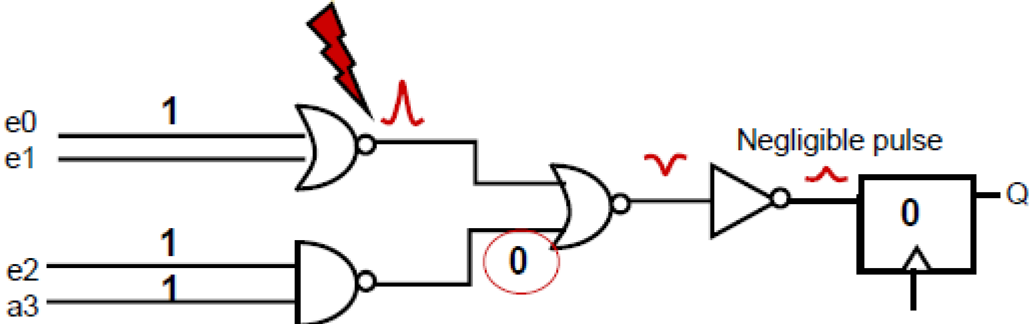


Figure 2.13 Electrical masking.

- **Latching-window masking** occurs when the transient pulse, generated from a particle strike, reaches the latch, but not at the clock transition where the latch captures its input value. Figure 2.14 shows how the transient pulse reach the flip-flop but it is ignored because the clock edge is not available.

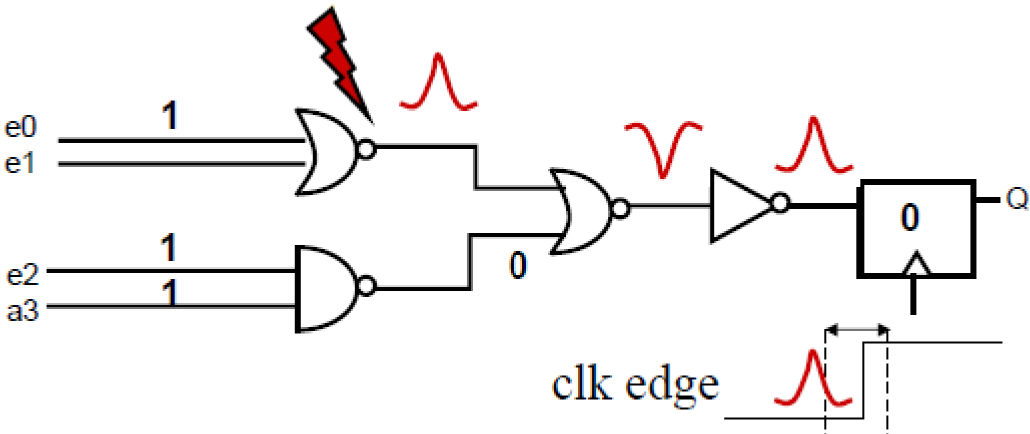


Figure 2.14 Latching-window masking.

2.3.3 Single Event Latchup (SEL)

Single Event Latchup is an abnormal high-current state in a device caused by the passage of a single energetic particle through sensitive regions resulting in the loss of device functionality. Adjacent n-type and p-type regions in CMOS circuits may form a parasitic thyristor composed of two pairs of parasitic bipolar transistors. Some PNPN structure can switch from a high impedance to a low impedance. A spurious current spike in one of these transistors may be amplified by the large positive feedback of the thyristor to cause a virtual short between V_{DD} and ground (Nicolaidis, 2006). If power is not removed quickly, catastrophic failure may occur due to excessive heating or bond wire failure.

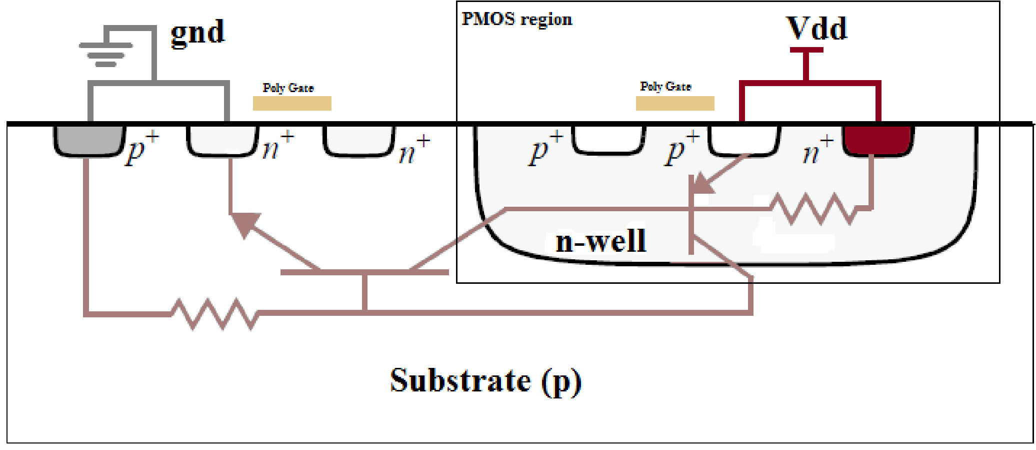


Figure 2.15 PNPN structure in CMOS technology (Deepon Saha, 2010)

Consider the n-well structure in the Figure 2.15. The NPNP structure is formed by the source of the NMOS, the p-substrate, the n-well and the source of the PMOS. A circuit equivalent is also shown in Figure 2.16. When one of the two bipolar transistors gets forward biased (due to current flowing through the well, or substrate), it feeds the base of the other transistor. This positive feedback increases the current until the circuit fails or burns out (Rabaey et al., 1999).

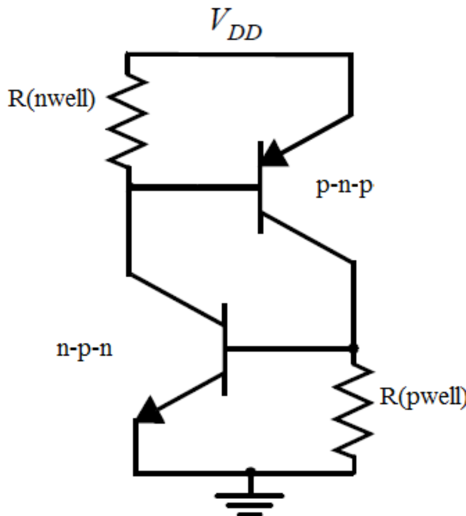


Figure 2.16 Equivalent circuit of CMOS latchup.

An ideal PNPN structure is active if the gain product of the NPN and PNP transistors is $\beta 1 \times \beta 2 > 1$, where $\beta 1$ and $\beta 2$ are the current gains Ic/Ib of the bipolar transistors. Other important parameters that influence the sensitivity are the resistivity and depth of the well and the resistivity of the substrate. To reduce the base-emitter resistor can be used a thin epitaxial layer on a high doped substrate. Multiple substrate and well contacts also reduce the base-emitter resistors.

2.4 Propagation induced pulse broadening (PIPB)

Propagation Induced Pulse Broadening (PIPB) is another concern to take into account in the design. Since an initial transient pulse is broadened enough through the logical circuit that have more probability of generating a data corrupted (Mogollon et al., 2009) (Ferlet-Cavrois et al., 2007). A logical circuit, as a chain of inverters or structure of the clock tree network, is susceptible to glitch voltage and a transient pulse with a few picoseconds can be broadened to about a few nanoseconds. Several simulations have been performed in (Mogollon et al., 2009) to show the PIPB effects dependence on the V_T voltage. They show that in a chain of two identical inverters after the first inverter the transient pulse is broadened, however, when passing through the second inverter the effect is reverted and the initial transient pulse width is recovered. Any SET broadening or attenuation will arrive from the imbalance between tp_{HL} (propagation time from high to low) and tp_{LH} (propagation time from low to high).

Sterpone *et al.* investigated the effects of PIPB on SET in Flash-based FPGA (Sterpone et al., 2011) using as case study ProASIC3 family manufactured in 130mn technology. The circuit under experiment was a logic path of a Pipelined Multiplier including logic gates (2-bits NOR, 2-bits AND and 2-bits EX-OR), as shown in Figure 2.17.

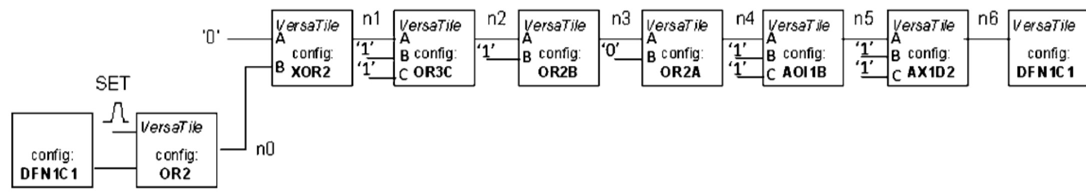


Figure 2.17 Logic path mapping in the Flash-based FPGA.

The result of the experiment is shown in Table 2.1, the measurement have been performed considering the average measurement over 15 electrical injections. Table 2.1 shows two types of SET injection simulations, SET pulse 0-1-0 and SET pulse 1-0-1. The SET pulse 0-1-0 was attenuated and filtered, in some cases it is possible to confirm that shorter pulses are more susceptible to filtered effects. On the other hand, when the SET pulse 1-0-1 is considered, the effect is quite opposite. The SET pulse are broadened. The broadening effect is more relevant for shorter pulses, reaching the 82% of effect PIPB for a 350 ps of injected pulse.

Table 2.1 Results of SET fault injection in FPGA prototype in critical path

node	SET width	PIPB Effect %	PIPB factor [ps / logic path]	SET width	PIPB Effect %	PIPB factor [ps / logic path]	SET width	PIPB Effect %	PIPB factor [ps / logic path]
SET (0-1-0 pulse)	350 ps			700 ps			1,200 ps		
n6	0 ps	filtered	-350	355 ps	-50%	-355	876 ps	-27%	-324
SET (1-0-1 pulse)	350 ps			700 ps			1,200 ps		
n6	640 ps	+82%	+290	888 ps	+27%	+188	1,410 ps	+18%	+210

In order to characterize the PIPB effect, the same critical path was modeled at transistor level using hspice at 130 nm of Predictive Technology Model (NIMO, 2012). Each logic gates of the critical path was modeled according to Versatile from ProASIC3, more details in (Sterpone et al., 2011). Table 2.2**Error! Reference source not found.** shows the simulation of PIPB effects in SET propagation at hspice based on Figure 2.17. Note that at each Versatile output SET may be filtered or broadened according to the SET width and its direction (1-0-1 or 0-1-0). Note also that the PIPB effects may vary along the logic path.

Table 2.2 Result of SET propagation in critical path at hspice level

node	SET Width (ps)	PIPB Effect %	SET Width (ps)	PIPB Effect %	SET Width (ps)	PIPB Effect %
SET ⎍	350	-	700		1200	
n0	276	-21%	630	-10%	1132	+5
n1	108	-60%	477	-24%	981	-13
n2	filtered	-100%	422	-11%	928	-5
n3	filtered	filtered	249	-40%	760	-18
n4	filtered	filtered	216	-13	734	-3
n5	filtered	filtered	261	+25	783	+6
n6	filtered	filtered	355	+36 (-50% effect)	876	+11 (-27% effect)
SET ⊔						
n0	428	+22%	779	+11	1281	+6
n1	569	+32%	918	+17	1424	+11
n2	611	+7%	964	+5	1477	+3
n3	775	+26%	1130	+17	1642	+11
n4	794	+2%	1148	+1.5	1662	+1.2
n5	738	-7%	1086	-5	1609	-3
n6	640	-13% (+82% effect)	888	-22 (+27% effect)	1513	-6 (+25% effect)

An important contribution to understand the PIPB effects was presented in (Wirth et al., 2008) where he describes a model to induce pulse broadening. This model is divided in four regions, according to the relationship between τ_n (duration of the transient pulse at the n-th logic stage) and the gate delay t_p. Where t_p will be equal to t_{pHL} for 0-1-0 transient at the n-th node, or equal to the t_{pLH} for 1-0-1 transition. An example of gate delay is shown in Figure 2.18. In the same Figure 2.18, the authors claim that "if the propagation delay for the first transition is shorter than the propagation delay for the second transition, the pulse is broadened" i.e. if $t_{pHL} < t_{pLH}$ then the pulse in the output of the gate will be broadened.

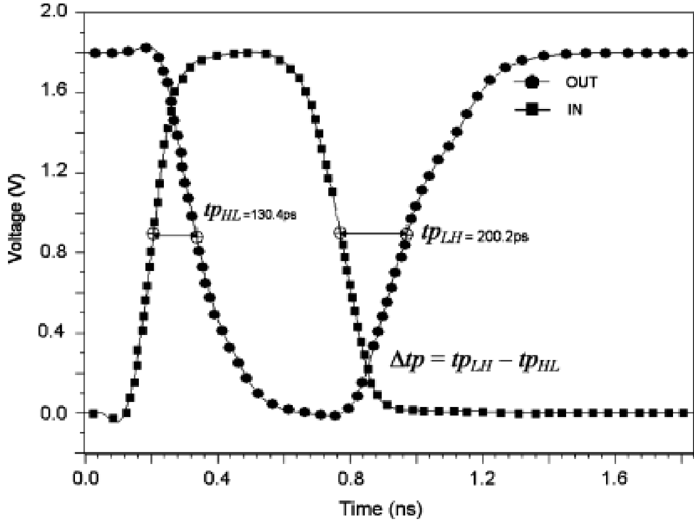


Figure 2.18 Propagation delay of an inverter gate

The first region of the PIPB model presented by Wirth represent the situation where the transient pulse is filtered out. The model described in the equation 2.3 evaluates the relationship between the width of the transient pulse (τ_n) at the input of the gate and the delay of the gate (t_p). Then, for a transient pulse to be propagated to the next logic stage, the ratio τ_n / t_p, must be equal or greater than k. Where k is a fitting parameter which depends on the technology of interest. For transient pulse with duration τ_n times t_p, the voltage at the output of the gate will be less than VDD/2, and it is considered filtered.

$$if\ (\tau_n < k * t_p) \rightarrow \tau_{n+1} = 0 \qquad \text{Equation 2.3}$$

The second region, described in Equation 2.4, represents the simulation in which the transient pulse is not degraded by electrical masking, but it may change as broadened or attenuated while propagating through the logic path due to unbalance between t_{pHL} and t_{pLH}. This occurs if the input pulse has duration (τ_n) greater than (k+3) times t_p.

$$if\ (\tau_n > (k+3) * t_p) \rightarrow \tau_{n+1} = \tau_n + \Delta t_p \qquad \text{Equation 2.4}$$

The third region represents the situation in which electrical masking occurs. It is found that the pulse may degrade faster in the last stage before being filtered out. The model for the third region, described in Equation 2.5, was obtained by curves fitting.

$$if\ \left((k+1) * t_p < \tau_n < (k+3) * t_p\right) \rightarrow \tau_{n+1} = \frac{{\tau_n}^2 - {\tau_p}^2}{\tau_n} + \Delta t_p \qquad \text{Equation 2.5}$$

The fourth region models the situation where stronger degradation occurs. In the last stages the pulse degrades faster, compared to the pulse decreasing in early stages and the model for the third region loses its validity. The model for the fourth region is:

$$if\ (k * t_p < \tau_n < (k+1)t_p) \rightarrow \tau_{n+1} = (k+1)t_p\left(1 - e^{(k - \tau_n/t_p)}\right) + \Delta t_p$$
Equation 2.6

The model presented by Wirth is validated by comparing the results obtained using model equations to hspice circuit simulation for the 0.25 μm Bulk technology node. More details in (Wirth et al., 2008).

3 CLOCK DISTRIBUTION NETWORKS

In a synchronous digital system, the clock signal is used to define a time reference for the flow of data and controls. Clock signals and networks distribution are vital functions to the operation of a synchronous system (Friedman, 2001). Clock signals are typically loaded with the greatest fan-out, travel over the longest distance, and operate at the highest speeds of any signal, either control or data. Since the data signals are provided with a temporal reference by the clock signal, the clock waveforms must be particularly clear and sharp. Furthermore, these clock signals are particularly affected by technology scaling, process variability and also radiation issues.

Ideally, the clock signal should appear to all processing elements at the same time. However, since the clock signal must be routed via a distribution network that includes clock distribution logic (buffers, clock gates) and interconnects, this signal arrive at the inputs of the processing elements at different times. This difference in arrival times is defined as clock skew. If the skew is greater than a certain threshold, may cause incorrect operation of the circuit (Ramanathan et al., 1994). Moreover, the propagation delay of the signal is defined by the distance that a clock signal must travel through the network. Jitter is another concern that affects the waveform of the clock signal. An excessively long signal path will introduce large delays that will significantly degrade the performance of the circuit. To deal with drawbacks different topologies have been adopted. In this chapter we are going to describe the most important topologies of the clock distribution networks such as buffered clock tree, H-tree, clock mesh and issues related to power consumption.

3.1 Definitions

A clock distribution network is developed taking into account some parameters of design like area, floorplan, and power consumption. To choose the appropriate distribution of the clock signal it is necessary to define some terms such as skew, jitter, and how to deal with these.

3.1.1 Skew

Clock skew is defined as the maximum difference of the signal reaches to every sequential element connected through some clock path. Since data signals must travel long distances, are inevitably delayed by combinational logic and interconnect. Some points on the chip are susceptible to short path errors caused by clock skew. Figure 3.1 represents a concept of clock skew. It is especially important, for the clock distribution, to achieve low skew among nearby points. A typical way to avoid fast-path errors is adding delay to short path on the chip. This is usually done by adding buffers to delay such signals, as see in the Figure 3.2. This method of protection, against the possibility of fast-path error, obviously increases chip area, power consumption and design time. But it reduces a clock skew for better performance of the chip (Restle et al., 2000).

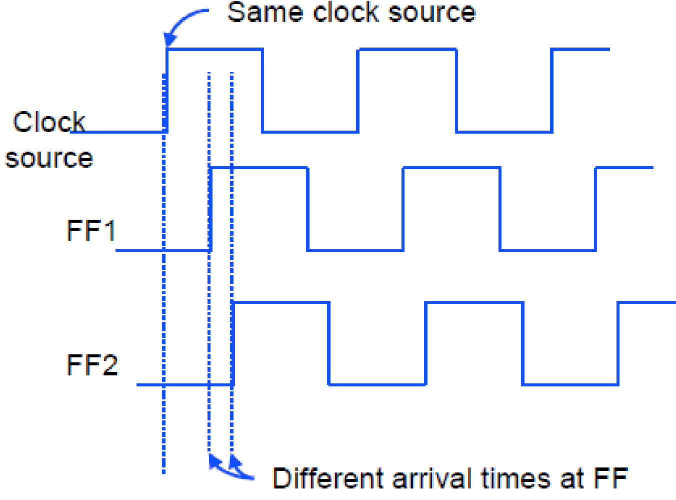


Figure 3.1 Clock skew definition.

An ideal global clock distribution achieves skew zero to everywhere. Clock Tree Synthesis (CTS) is a stage of the design flow, which uses an algorithm to insert buffers in specific place and reduce the delay difference of path. The goal of CTS is to reach the minimum low skew.

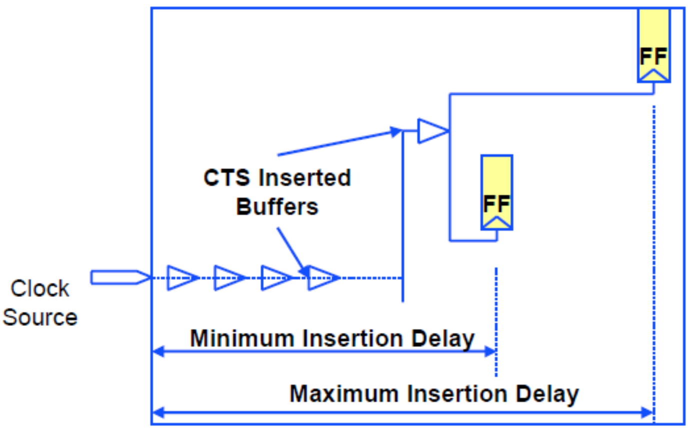


Figure 3.2 Clock skew in circuits.

3.1.2 Jitter

Jitter is defined as undesired variations in the cycle period. Jitter generates timing uncertainty in the signal because some cycles are reduced or enlarged. This undesired variation can generate negative or positive clock jitter, as shows the Figure 3.3. Generally a phase-locked loop (PLL) is used to generate a clock on chip. In the past, much of clock jitter came from the PLL and its sensitivity to power-supply noise. However, PLL designs have continued improving such that PLL jitter has scaled well with chip cycle time. The total delay of the global clock distribution has not scaled with chip cycle time. In fact, the total delay, which is now interconnect dominated, has tended to increase with increasing chip size and the number of clocked circuits. Since the global clock is buffered at many

places on the chip, it uses the regular noisy V_{DD} and GND, then, much of the local clock jitter is from buffer delay variations due to power supply noise. In addition to the requirement of a low jitter PLL, the jitter from the clock distribution must be controlled. This can be done through optimal buffer and transmission line design, and by minimizing non-periodic power noise.

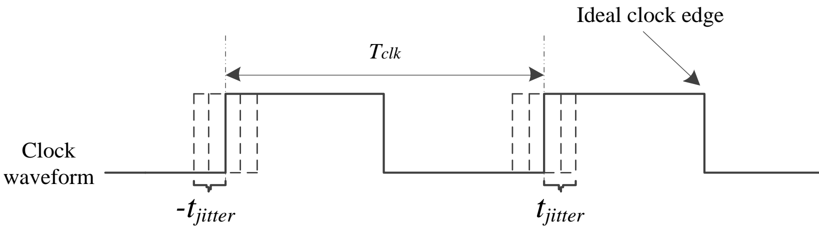


Figure 3.3 Jitter in clock waveform.

Variation in capacitive load also contributes to clock jitter. We can consider two major sources of capacitive-load variation; coupling between the clock lines or adjacent signal wires, and variations in gates capacitance. The clock network includes both interconnections and the gate capacitance of registers. Any coupling between the clock wire and adjacent signal results in jitter. Since the adjacent can transition in arbitrary directions and at arbitrary times, the exactly coupling to the clock network is not fixed from cycle-to-cycle. Variation in the gate capacitance related to the sequential elements is another source of clock jitter, because the load capacitance is highly non-linear and depends on the applied voltage. In many latches and registers this translates to the clock load being a function of the current state of the latch/register and depend of the values stored on the internal nodes of the circuit (Rabaey et al., 2003).

3.2 Clock Distribution Architectures

Historically, the primary goal in clock distribution networks is to ensure that a clock signal arrives at every register within the entire synchronous system at the same time. This concept of zero clock skew has been extended to provide either a positive or a negative clock skew at a magnitude depending upon the temporal characteristics of each local data path.

The trade-off between speed, physical die, and power dissipation are critically affected by the clock distribution network. The design methodology and structural topology of the clock distribution network should be considered in the development of a circuit. Therefore, several clock distribution strategies have been developed. The most common approach to improve clock distribution is the use of buffers such as buffered clock tree, H-trees and clock mesh and combinations of these.

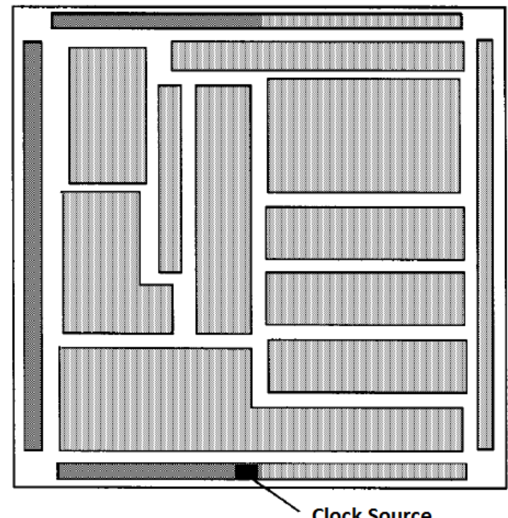


Figure 3.4 Floorplanning of structured custom VLSI circuit (Friedman, 2001).

3.2.1 Buffered Clock Tree Networks

Clock tree is the most common strategy for distributing clock signals in VLSI-based systems. This strategy consists in insert buffers at the clock source and/or along a clock path, forming a tree structure. Thus, the unique clock source is frequently described as the root of the tree, the initial portion of the tree as the trunk, individual paths driving each register as the branches, and the registers being driven as the leaves. This metaphor for describing a clock distribution network is commonly accepted and used throughout the literature and is illustrated in Figure 3.5. Occasionally, a mesh version of the clock tree structure is used in which shunt paths are placed to minimize the interconnect resistance within the clock tree.

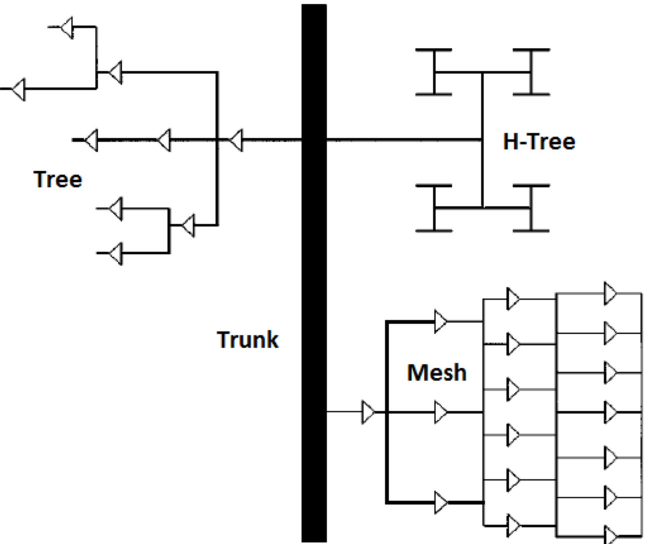


Figure 3.5 General clock distribution networks in integrated circuit (Friedman, 2001).

If the interconnect resistance of the buffer at the clock source is small as compared to the buffer output resistance, a single buffer is often used to drive the entire clock

distribution network. This strategy may be appropriate if the clock is distributed entirely on metal, making load balancing of the network less critical. The primary requirement of a single buffer system is that the buffer should provide sufficient current to drive the network capacitance, while maintaining high-quality waveform shapes and minimizing the effects of the interconnection by ensuring that the output resistance of the buffer is much greater than the resistance of the interconnect section being driven.

An alternative approach to using only a single buffer at the clock source is to distribute buffers throughout the clock distribution network, as shows the tree section of the Figure 3.5. This approach requires additional area but greatly improves the precision and control of the clock signal. The distributed buffers serve as double function, amplify the clock signals degraded by the distributed interconnect impedances and isolate the local clock nets from upstream load impedances. In this approach a single buffer drives multiple clock paths. The number of buffer stages between the clock source and each clocked register depends upon the total capacitance in the form of registers and interconnect, and the permissible clock skew. The maximum number of buffers driven by a single buffer is determined by the current drive of the source buffer and the capacitive load of the destination buffers.

The area of layout in clock distribution network has been developed by algorithms which carefully control the variations in delay between clock signals net so as to minimize clock skew. The strategy used is to construct binary tree-like structure with clock pins at the registers nodes. The structure are created using a recursive bottom-up approach. At each clock-input registers, a clock net is defined. The point where two zero-skew are symmetric is chosen such that the effective delay from that point to each clocked register is identical. This process continues up the clock distribution tree, then the point of connection of each new branch being chosen to satisfy the zero-skew design goal. The layout process ends when the root of the clock tree is reached. Figure 3.6 shows the geometric matching process of the algorithm. Thus, the algorithm attempts balance the delay of each clock branch in a recursive manner from the registers to the clock source.

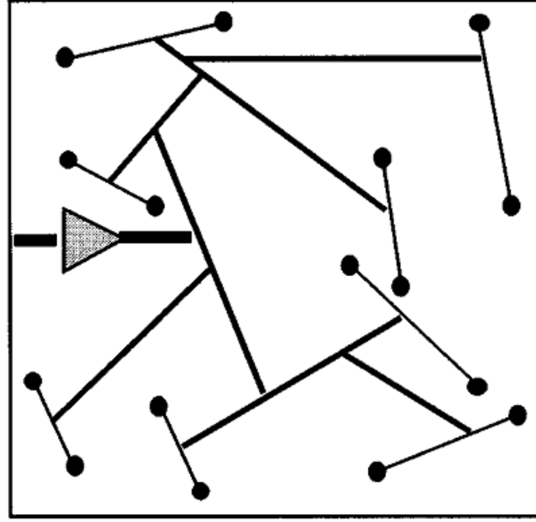


Figure 3.6 Geometric matching for zero skew (Friedman, 2001)

3.2.2 H-Tree Clock Distribution Networks

An H-tree topology is a symmetric architecture to ensure zero clock skew by preserving the distributed interconnect. Buffers often are identical from the clock signal source to the clocked register of each clock path. In this distribution, the primary clock driver is connected to the center of the main H structure and the clock signal is transmitted to the four corners of the main H, see Figure 3.7. These four identical clock signals provide the inputs to the next level of the H-tree hierarchy, represented by the four smaller H structure. The distribution process continues through several levels of progressively smaller H structures. The final destination points of the H-tree are used to drive the local registers or are amplified by local buffers which drive the local registers. Thus, each clock path from the clock source to a clocked register has practically the same delay. The primary delay between the clock signal paths is due to variations in process parameters that affect the interconnect impedance and, in particular, any active distributed buffer amplifiers. The amount of clock skew within an H-tree structure is strongly dependent of physical size, variation process, and how registers are distributed within the H-tree structure.

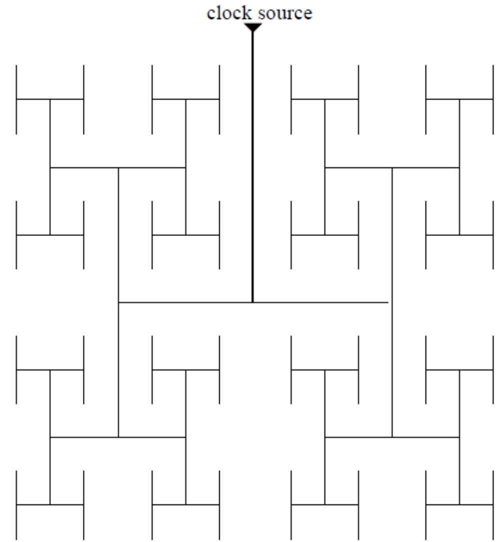


Figure 3.7 Symmetric H-tree distribution (Wilke, 2008).

3.2.3 Clock Mesh Distribution

Another important clock distribution network is the mesh structure which allows almost zero skew to the circuit. Clock Mesh is a grid composed by wires which sequential elements (registers) may be directly connected. Mesh is typically used in the final stage of clock network and widely used in the design of microprocessors. A major advantage of mesh structure is easily accessible at various points on the die to take the clock signal if the design change.

As clock trees, clock meshes typically use a multilevel structure to improve the clock signal from the root to all the clock input pins (flip-flops, memories, and so on). One of

the powerful design tool is Encounter from Cadence and this creates multilevel distribution that we can see in Figure 3.8 and that include the following sections:

■ Top-level chain

The top-level chain is a cascaded buffer chain from the mesh root to the first level of mesh pre-drive buffers. The chain of buffers can be used either to supply a suitable input transition to the mesh pre-drive, or to pad the mesh with extra insertion delay.

The routing for mesh chain nets is handled entirely by the router tool using a combination of special and regular routes. Meshes are not required to include a top-level chain. The root can connect directly to the first pre-drive stage, if it is suitable.

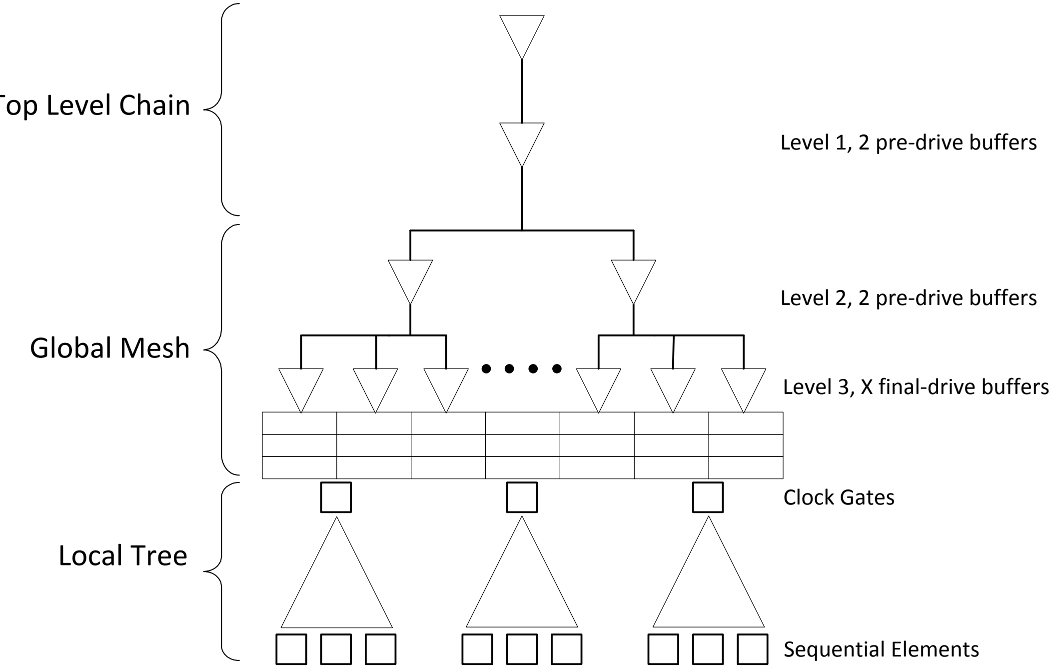


Figure 3.8 Clock mesh structure

■ Global mesh

The purpose of the global mesh is to distribute a single clock signal across the entire clock domain with good insertion delay and skew control. The global mesh consists of multiple (zero or more) pre-drive stages, followed by a single final-drive stage. While the pre-drive stages can have multiple nets at a given level (for example in an H-tree), the final stage always drives a single final global mesh net. This final mesh net can connect directly to the clock input pins, or there can be an additional level of local distribution, an example of global mesh is shown in Figure 3.9.

■ Local distribution

Local distribution is an optional section in which multiple small trees distribute the single global signal of the final mesh drive net to individual flip-flops or memory inputs.

The simplest form of local distribution consists of multiple small single-buffer (local trees), each one driving a cluster of flip-flops.

Using Encounter, NanoRoute tool handles all of the local-level routing. Using local distribution can have advantages over direct mesh-to-flip-flop connections in the final stage. For example, local distribution significantly reduces the loading in the final mesh stage, and can allow for a mesh structure with lower overall power consumption.

Also, for large clock domains with non-uniform flip-flop distributions, adding extra local buffers can help to balance the load by the mesh, and allow for better skew control.

Global mesh uses a multilevel H-tree pre-drive, followed by a general mesh final stage. Although the pre-drive is a tree structure, it can still employ multiple drivers on a single net. The final stage consists of a rectangular grid of final drivers feeding a rectangular mesh grid of trunks and branches.

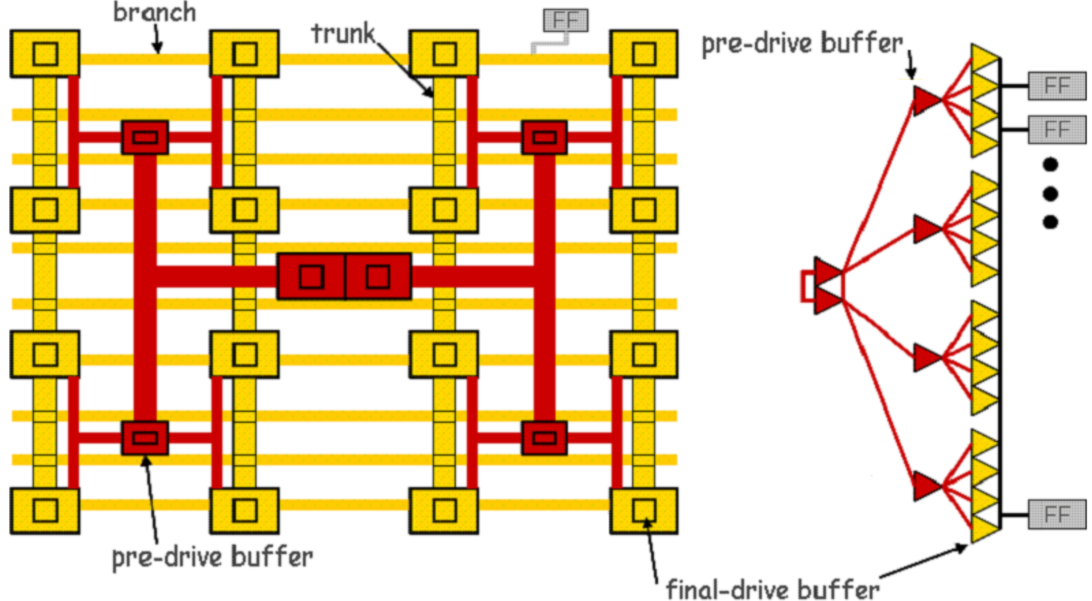


Figure 3.9 Global mesh using H-tree distribution (Cadence, 2003).

Implementing a clock mesh involves various tasks, such as inserting drivers into the netlist, placing and fixing the drivers, creating mesh routes, and so on. The Encounter clock mesh feature includes the following major implementation capabilities:

■ Synthesis

The synthesize ClockMeshcommand implements the clock mesh according to the current specification, by inserting and placing buffers, and creating mesh special routes as needed.

■ Mesh routing

The route ClockMeshcommand uses the native NanoRoute to complete detailed routing connections for driver and flip-flop inputs, top-level cascade chains, and local trees.

■ Wire trimming

After mesh routing is complete, the trim ClockMeshcommand removes unused portions of the mesh trunks and branches, eliminates antenna violations, and reduces overall mesh capacitance.

The advantage to this style is that the pre-drive is highly symmetrical and can achieve good skew control for large clock domains with many flip-flops. The drawback is that it may require higher power than other mesh styles.

Among different methods suggested for skew reduction, the mesh with a top level tree has been shown to be very effective (Rajaram e Pan, 2010). Unfortunately, the penalty is the power dissipation since the structure has a lot of redundancy interconnects. Also, the convergence of the paths may produce short circuit currents among the mesh drivers and they are, along with the high capacitance associated with the mesh wire structure, responsible for the higher power consumption in comparison to clock tree networks (Minsik et al., 2010).

3.3 Power Dissipation in Clock Distribution Networks

In a modern VLSI system, the clock distribution network may drive thousands of registers, creating a large capacitive load that must be efficiently sourced. Furthermore, each transition of the clock signal changes the state of each capacitive node within the clock distribution network, in contrast with the switching activity in combinational logic blocks, where the change of logic state is dependent on the logic function. The combination of large capacitive loads and a continuous demand for higher clock frequencies has led to an increasingly larger proportion of the total power of a system dissipated within the clock distribution network, in some applications the power consumption is greater than 25% of the total (Hirotsugu et al., 1994). The primary component of power dissipation in most CMOS-based digital circuits is dynamic power.

Most methods reported in the literature on minimizing dynamic power in sequential circuits are generally based on the following techniques: clock gating and reduced swing.

A) **Clock gating** is an effective technique for minimizing dynamic power in sequential circuits, this technique consists in freezing the clock signal in regions of the chip that are not being used. Regions where clock is frozen are said to be on sleep mode. When clock is not switching dynamic power consumption is reduced to zero since no transitions occur in these regions. Clock signal can be set either to zero or one in sleeping regions. All regions in sleep mode are unable to process any data (Wilke, 2008).

B) **Reduced swing** is another effective technique to reduce the power consumption. The total power dissipation in a clock network, consists of three components: leakage, short-circuit, and dynamic power. The leakage current is dependent on the technology and is relatively small component in a clock network. Similarly, keeping proper rise and fall times throughout the clock tree may also minimize the short-circuit power component. The clock network has high switching activity; therefore the dynamic power consumption is the dominant factor (Haj Ali Asgari e Sachdev, 2004). Ignoring leakage and short-circuit power contributions, the clock network power dissipation is given as Equation 3.1.

$$P = f\, C_L\, V_{DD}\, V_{SW} \qquad \text{Equation 3.1}$$

Where f is the clock frequency, C_L is the total capacitance of the clock network, V_{DD} and V_{SW} are respectively the supply voltage and the output swing of the buffer. If the output of the buffer swings from GND to V_{DD}, then $V_{SW} = V_{DD}$, and the formula reduces to Equation 3.2.

$$P = f\, C_L\, (V_{DD})^2 \qquad \text{Equation 3.2}$$

Then, reducing V_{DD} dynamic power consumption is reduced quadratically. Dynamic power consumption could be reduced in a linear fashion by reducing only V_{SW}. A better approach is to change V_{DD} and V_{SW} only for the clock distribution network. Since clock devices are not going to be affected by the voltage reduction.

3.4 Radiation Effects on Clock Tree Networks

It has been shown that logical paths, such as clock distribution networks, are vulnerable to transient faults known as SETs. Technology scaling leads to decrease in transistor nodes capacitance and to raise circuit speed, which helps to increase SET susceptibility. During the strike of an ionizing particle, charge may be collected on the off transistor at the output of the clock buffers, which creates a transient pulse that can propagate through the network provoking clock glitches and/or clock jitter or clock skew. Figure 3.10 shows the effects of particle strike on clock path. If the SET pulse reaches one or more register elements depending on the topology, one or more elements may have the stored value changed, resulting in abnormal behavior on the system. Another consequence is the possibility of the SET can be broadened enough and increasing probability of being stored by edge triggered registers or other digital blocks giving place a data corruption. There are many types of clock distribution networks and many techniques have been proposed about radiation hardening in clock buffers. In this thesis we investigate how susceptible is a circuit before be manufactured. We will find the sensible nodes and what registers are susceptible to SETs. Eventually each circuit has been designed with different topology and in this thesis we investigate the radiation effects using clock tree and clock mesh distribution networks.

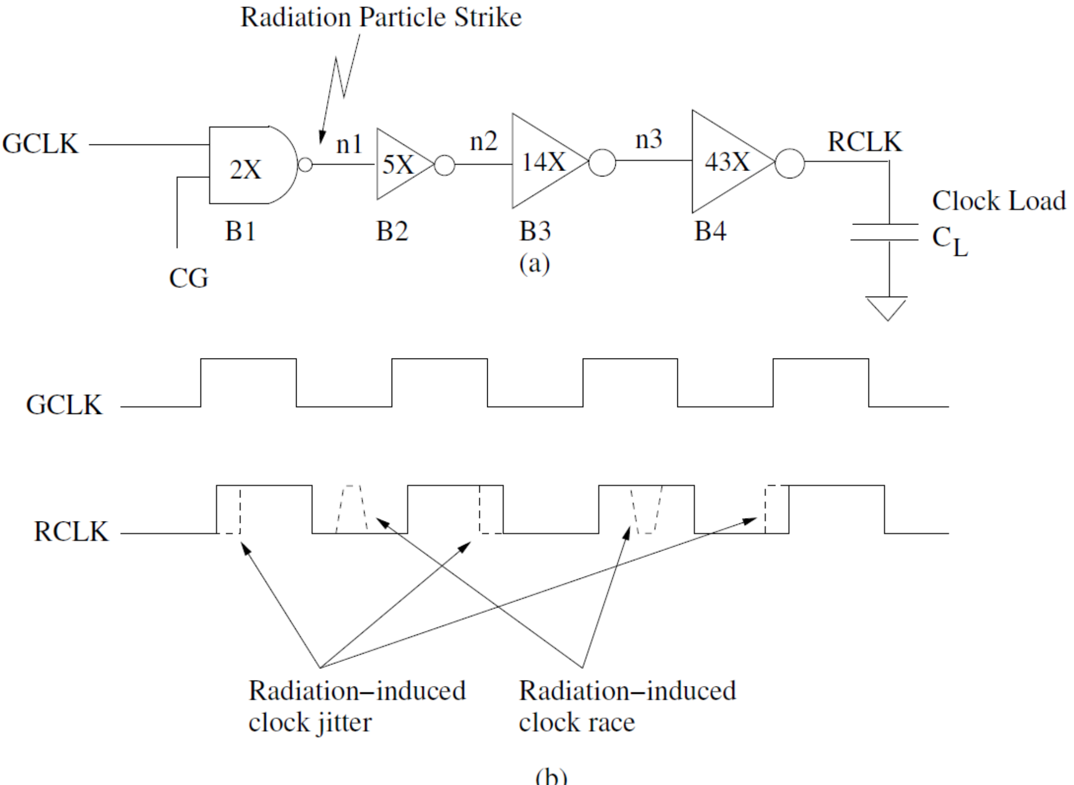


Figure 3.10 Effects of radiation particle strike on a regular clock path (Dash et al., 2009).

3.5 State-of-the-Art Mitigation Techniques to Clock Tree Networks

In this section, we describe some techniques, which help to improve the clock distribution networks against radiation effects. The first technique that we found in the literature is hardened dual port inverter as buffer. The second technique used to deal with radiation effects is introducing delay elements as part of buffers to filter glitch voltage. TMR clock regenerator circuit and Split-output inverter based clock regenerator are other techniques that we will describe.

3.5.1 Hardened Dual Port Inverter

A hardened dual port inverter found in (Ming e Shanbhag, 2005) is an interesting proposal to use in clock distribution networks, a distinct feature of this circuit is its dual input and output ports as shown in Figure 3.11 (b).

The proposed inverter adds two transistors to get the hardened goal. Both transistors have its body terminal tied to the corresponding source. The input ports IN_P and IN_N in Figure 3.11 (b) carry the same logic value to the PMOS and NMOS transistors. If we assume both input ports are in logic 1, only the drain node of PMOS transistor P1 is a sensitive state. A particle strike may induce a 0-1-0 transient pulse at the output port OUT_P. This transient voltage does not affect the output of a succeeding inverter, because OUT_P is connected to PMOS transistors of the succeeding dual port inverter, as shown in Figure 3.12. During the 0-1-0 transient pulse the OUT_P is in high impedance. The

transient voltage at OUT_N will increase much less due to the voltage division of NMOS transistors N2 and N1, as consequence the next dual port inverter will not be affected. A similar analysis exists when IN_P and IN_N are at logic 0.

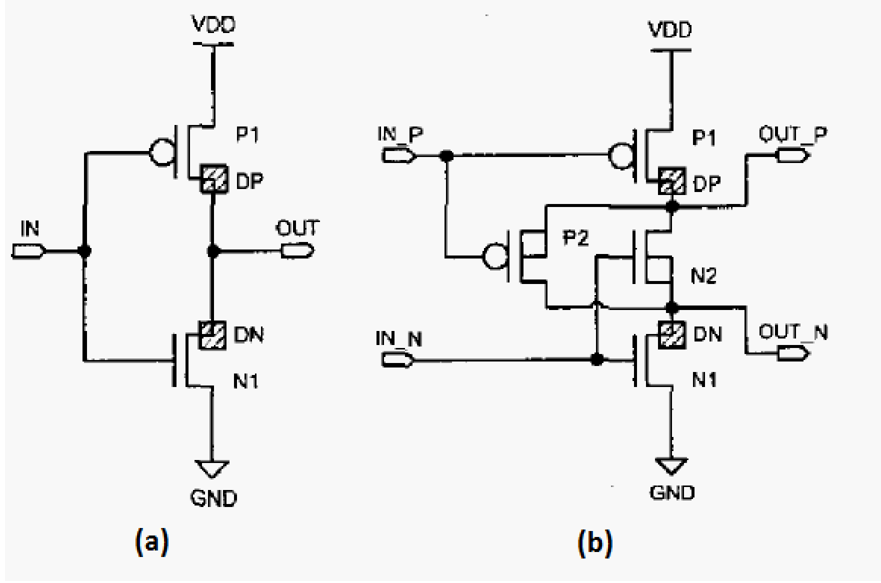


Figure 3.11 Inverter proposed: (a) conventional inverter, (b) hardened dual port inverter (Ming e Shanbhag, 2005).

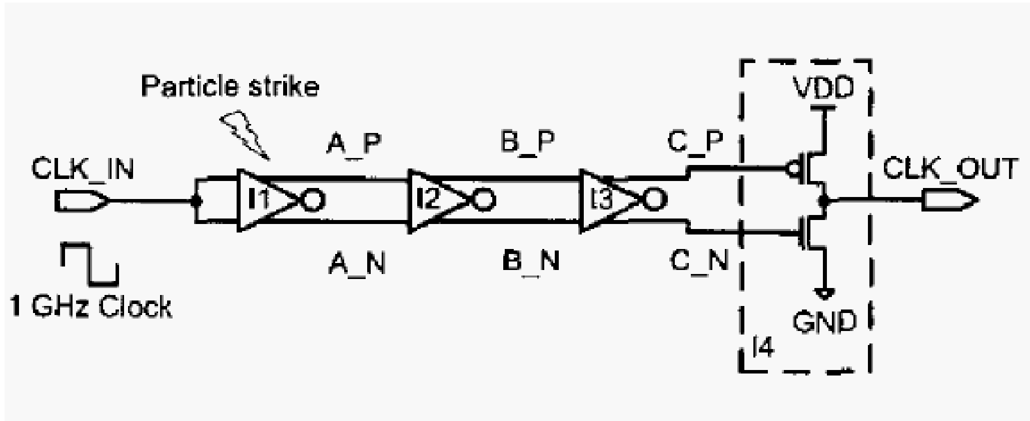


Figure 3.12 Hardened dual port inverter as clock path (Ming e Shanbhag, 2005).

In concordance with the results of the simulations by the author hardened dual port inverter it is possible use as buffer in the clock distribution networks.

3.5.2 SEU Hardened Clock Leaf Inverter

The proposal in (Mallajosyula e Zarkesh-Ha, 2008) introduce a leaf inverter with a C-element. This circuit works as an inverter under normal operation. This circuit checks the consistency of the clock signal, and checks the delay (T_D) introduced by the delay element. Any transient pulse width (T_{SEU}) < (T_D) is filtered. The maximum delay (T_D) for

the transient pulse is filtered at the input of the C-element is *T*/2, when *T* is the clock period. The delay element, showed in Figure 3.13, is designed to obey timing constraints of the clock network and it can be easily implemented by a chain of even numbered inverters. This inverter is designed to be placed at the end of the clock path because it can filter the transient pulse.

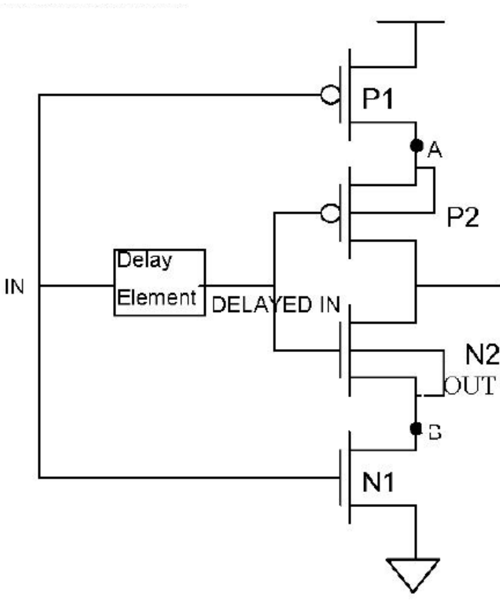


Figure 3.13 SEU hardened clock leaf inverter (Mallajosyula e Zarkesh-Ha, 2008).

3.5.3 TMR Clock Regenerator Circuit

In (Dash et al., 2009) have been proposed two hardening technique for regional clock regenerator. One approach uses Triple Modular Redundancy (TMR) for radiation hardening, while the other approach uses radiation hardening split-output inverter. According to the author, the TMR approach eliminates the risk of radiation induced clock glitches in the clock nodes up to 150 fC. The TMR regenerates the clock signal using three parallel regenerator circuits shown in Figure 3.14. The clock regenerator circuit drives a load of around 64 flip-flops in the maximum case. And in each flip-flop, the clock input drives a load of a 2X inverters.

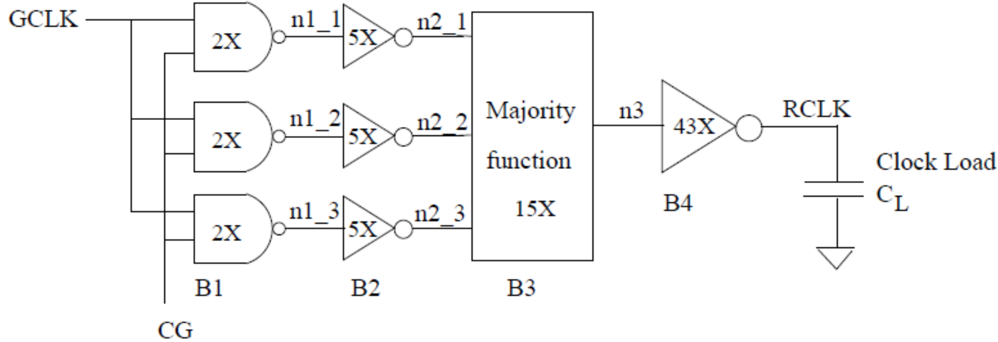


Figure 3.14 TMR clock regenerator circuit (Dash et al., 2009).

The output of the majority function gate is used to drive the 43X inverter (B4) which is immune to particle strikes effects. The majority function gate implements voting logic to compute the correct output. The voting logic function with three inputs a, b, c is $\overline{ab + ac + ca}$. In eventual particle strike at node ($n2_1$) be affected but two other inputs of the majority function ($n2_2$ and $n2_3$) will remain at their correct value. Thus the majority function gate can compute the correct input value for B4.

Additionally, the TMR clock regenerator circuit has an area overhead 282% and increase dynamic power consumption by 116% as compared to regular clock regenerator.

3.5.4 Split-output Inverter Based Clock Regenerator Hardening

Other hardening technique found in (Dash et al., 2009) is the clock regenerator based on Split-output inverter. The circuit shown in Figure 3.15 is tolerant to radiation particle strikes at any node and at any time according with simulations by the author. This clock regenerator drives 64 flip-flops like TMR clock regenerator.

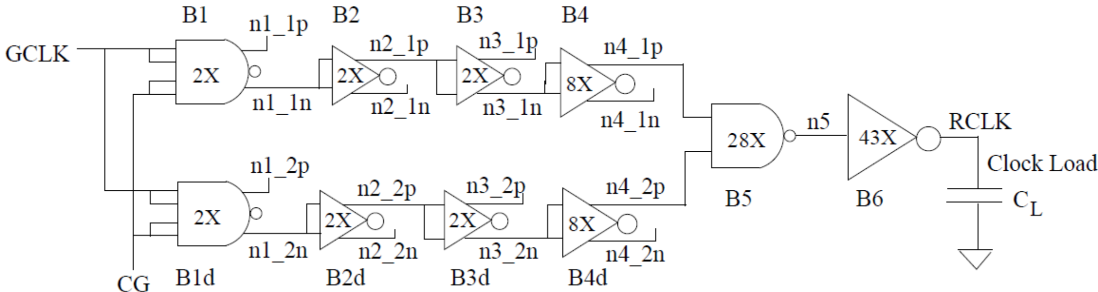


Figure 3.15 Split-output inverter based clock regenerator hardening (Dash *et al.*, 2009).

The inverters used is named split-output SEU tolerant inverter (Garg e Khatri, 2008). For understanding how the split-output inverter works, we consider two regular inverters shown in Figure 3.16 (a). To improve the INV1 and ensure its output does not affect the voltage at the output of INV2, INV1 is modified as shows the Figure 3.16 (c). INV1 has two inputs (inp and inn) and two outputs (out1p and out1n). The INV2 is also modified such that two different signals drive the transistors M3 and M4 in Figure 3.16 (b).

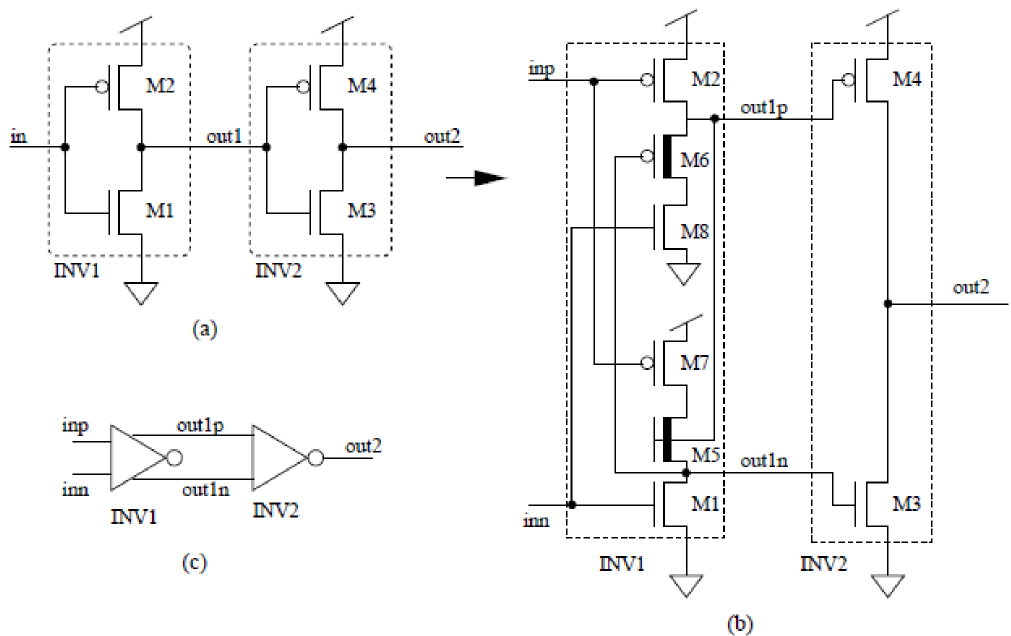


Figure 3.16 Split-output SEU tolerant inverter

To understand how it works, we assume that inn and inp are in logic 1, then in Figure 3.16 (b) the transistor M1 turns on and M2 turn off. As result, M1 pulls the node out1n down to logic 0, then turns on the transistor M6. Since M6 is turn on and M8 is also turn on, the out1p is driven to weak logic 0. Note that weak logic 0 is due to threshold voltage of the PMOS transistor M6. Then both out1p and out1n are now at logic 0, due to this, the output of INV2 (out2) goes to logic 1. Similar behavior occur at the node out1n, when the transistors M5 and M7 turn on, the node out1n is pulled to a weak logic 1 (V_{DD} - Vt). Note that the transistor M5 and M6 of INV1 in Figure 3.16 (b) are selected to be low threshold voltage (indicated by a thicker line).

In eventual particle strike at node out1p a transient pulse increases the voltage to V_{DD} due to the charge collection, as consequence, M4 of INV2 turns off and M5 turns on. However, the node out1n remains at GND value because M7 is in cutoff. Therefore M3 also remains off. Thus, the node out2 remains at the V_{DD} value, in a high impedance state. As consequence, radiation strike does not affect the voltage value of the *out*2 node.

Similarly, a particle strike at out1n does not affect the node out2 when inn and inp are at the GND value. A radiation particle strike at M8 can be significant only when out1p is at the V_{DD} value, since a radiation particle strike at the NMOS transistor can only result in a negative glitch. However, when out1p is at V_{DD}, M6 is turned off and hence a particle strike at M8 does not affect the node voltage of out1p. Similarly, a radiation particle strike at M7 does not affect the voltage of the out1n node. Then according to simulations the Split-output inverter is tolerant to radiation particle strikes.

3.6 Problem Definition

After review the radiation effects on clock distribution networks and see how the radiation effects can contribute to a failure of the circuit we reform the problem. How we can improve the clock distribution networks against radiation effects. Then at this point, we see the importance of a methodology to quantify the susceptibility of the circuit. In the next chapter we describe our methodology to quantify the susceptibility of the circuit from different point of view.

Techniques to deal with these transient faults exist, as we mentioned before, but come at a cost, using some protection technique probably it will increase the overhead. Designers clearly require accurate estimates of circuit error rates to make appropriate cost/reliability trade-offs. This work describes a method for generating these estimates and finally choose the best mitigation technique.

4 PROPOSED METHODOLOGY TO ANALYZE THE SET SUSCEPTIBILITY IN CLOCK NETWORKS

Soft errors caused by ionizing radiation have emerged as a major concern for current generation of technologies. High-energy neutrons or alpha particle from radioactive contaminants in packaging creates electron-hole pair in semiconductors. This electron-hole pair separates promptly in presence of an electric field and a temporary inversion layer may be created under the gate transistor. As mentioned in chapter 3, clock tree network may be affected by a temporary inversion.

In this chapter we introduce the proposed methodology to analyze the SET susceptibility in clock distribution networks with the aim to identify the most sensitive nodes and components of the circuit under test. A methodology proposed also cover topologies like clock tree and clock mesh described above.

4.1 Proposed Methodology

In order to develop any optimization technique for clock distribution network against radiation effects, it is necessary first to develop an accurate and efficient analysis of susceptibility in clock networks. We wondering how to make an accurate analysis if each design have a particular clock network. Then, we decide to elaborate a methodology to analyze any circuit design from the layout files. This methodology uses different environment to get an accurate analysis.

The proposed methodology offers different ways of evaluating the circuit. From the simulation point of view, we can do two type of simulation: simulation in logic level and simulation in electrical level. With the simulation in logic level we obtain the soft error rate in clock tree network. Simulation in electrical level allows to perform different types of SET simulation at hspice environment. As mentioned in chapter 1, we can do SET injection in every node of the clock network, SET injection in particular path, SET injection in the clock source, and we take a risk of replacing the small buffers on clock tree by large buffers. Also we make a comparison between clock tree and clock mesh from the point of view of susceptibility. All of these simulations are described in the next chapter.

The proposed methodology, shown in Figure 4.1 takes DEF, LVS and LEF files as input and extract the clock distribution network using the EXT-CLK tool to spice model. The main contribution of the method is EXT-CLK, tool developed in Bash command line in Linux environment, and it does not need to install any library and run also in OS X environment (Mac OS). After extraction of the clock network to spice model, we need to evaluate the model before performing any kind of SET simulation. We use the Predictive Technology Model (NIMO, 2012) at all simulations also to evaluate the model extracted. The evaluation of the model is not complex, we need to ensure that the clock signal reaches all registers of the circuit.

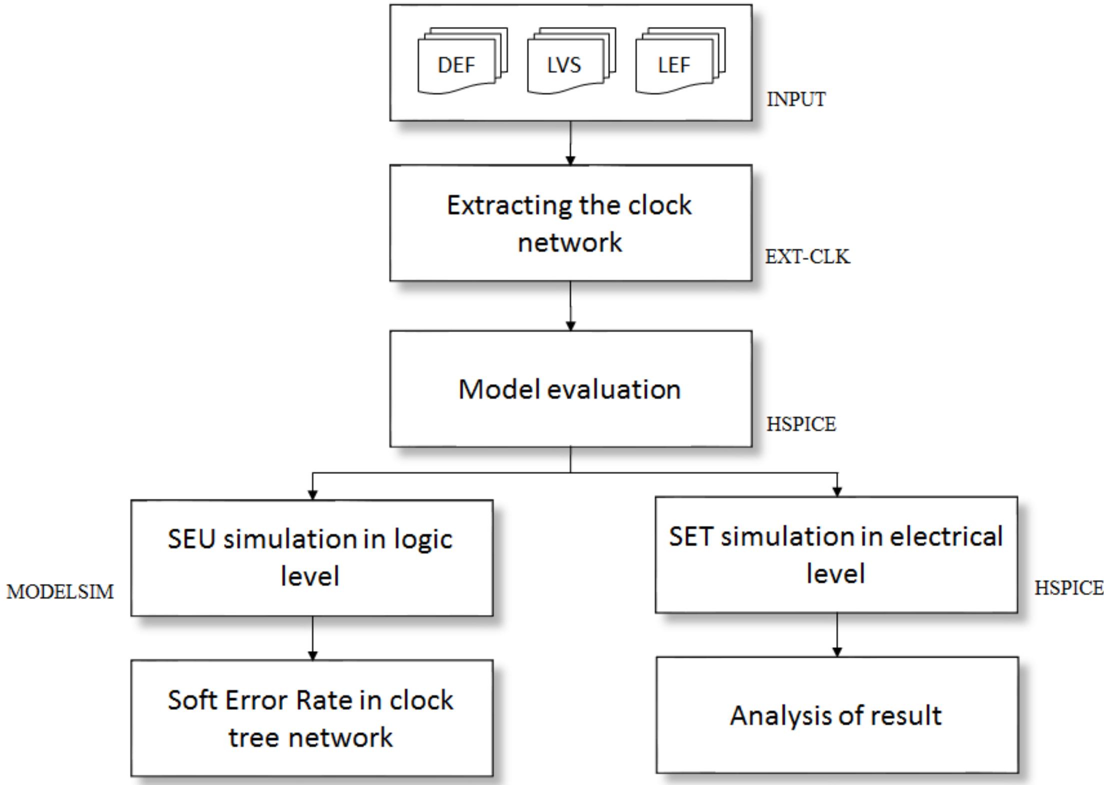


Figure 4.1 Proposed methodology to analyze SET susceptibility in clock networks.

4.2 EXT-CLK tool

EXT-CLK is a tool that extracts the clock tree network from layout design to netlist hspice. EXT-CLK tool takes three input files of the layout design and generates the clock network description in netlist hspice. The tool identifies buffers (inverters, not inverters), clock gates and flip-flops inside every clock path. Regarding to wire connection, EXT-CLK seeks each segment of wire and describes in pi model using the resistance and capacitance of each layer metal. Finally, we obtain the complete clock tree described in transistor level ready to simulate the effects of particle strikes.

It is important to note that circuits have many registers (hundred or thousand). As consequence, in eventual simulation of a particle strike in the clock network, we have to monitor each register, which is very complex. Therefore it is not enough to make a complete description of the clock tree without measuring points. Then, EXT-CLK tool generates the measure statements into the netlist in order to deliver a file with results. On the other hand, hspice can take long time to simulate a complex circuit whether no initial condition is used. Initial conditions are values used in hspice as part of solution in initial execution, and our tool EXT-CLK generates the initial condition value for each node.

The extraction of the clock tree network is performed taking into account the size of each buffer, the fan-out of each path, the place of clock gate and the resistance and capacitance of every wire connection.

The SRAM arbiter circuit was chosen as case study. The SRAM arbiter is the interface between the logic modules of a Gigabit Ethernet Switch with an external zero-bus turn

around (ZBT) SRAM memory. This circuit has been designed in 65 nm process using Encounter tool from Cadence (Tonfat, 2011).

The input files of EXT-CLK tool mentioned before are describe briefly:

DEF file: Design Exchange Format (DEF) file is an open specification for representing the physical layout of an integrated circuit in ASCII format. DEF conveys logical design and physical design data to place and route tools. Logic design data can include internal connectivity, grouping information, and physical constraints. Physical information includes placement locations and orientations, routing geometric data, and logical design change. In many cases, the DEF netlist information is in separate format, such as Verilog, or in a separate DEF file. It is also common to have a DEF file that only contains a components section to pass placement information. This file was created by Encounter tool. For more reference of DEF syntax see (Cadence, 2003).

LVS file: Layout vs. Schematic (LVS), is a class of verification software that determines whether a particular integrated circuit layout corresponds to the original schematic or circuit diagram of the design. LVS checking software recognizes the shapes of the layout that represent the electrical components of the circuit, as well as the connections between them. This netlist is compared against a similar schematic or circuit netlist. LVS file contains the netlist of every cell used in the integrated circuit.

LEF file: Library Exchange Format (LEF) contains library information for a class of designs. Library data includes layer, via, placement, and macro cell definitions. The LEF file is in ASCII representation using the syntax conventions. A single LEF file defines all the layout information. This may be complex and hard to manage. Instead, this information can be divided in two files, a "technology" LEF file and a "cell library" LEF file. A technology LEF file contains all of the LEF technology information for a design, such as placement and routing design rules, also process information for layers such as RESISTANCE and CAPACITANCE in specific unit that helps to calculate the wiring conductor model. A cell library contains the macro and standard cell information for a design.

After the briefly description of the input files. We describe the clock network netlist generated by EXT-CLK in spice model in *clock_network.sp*. The netlist is divided into six sections, such as standard cell statements, nets statements, call subcircuit, call nets, initial condition statements and measurement statements. Figure 4.2 shows an overview of the six blocks of the netlist. In order to create every section of the final netlist, several algorithms have been developed in Bash command line and here we explain these.

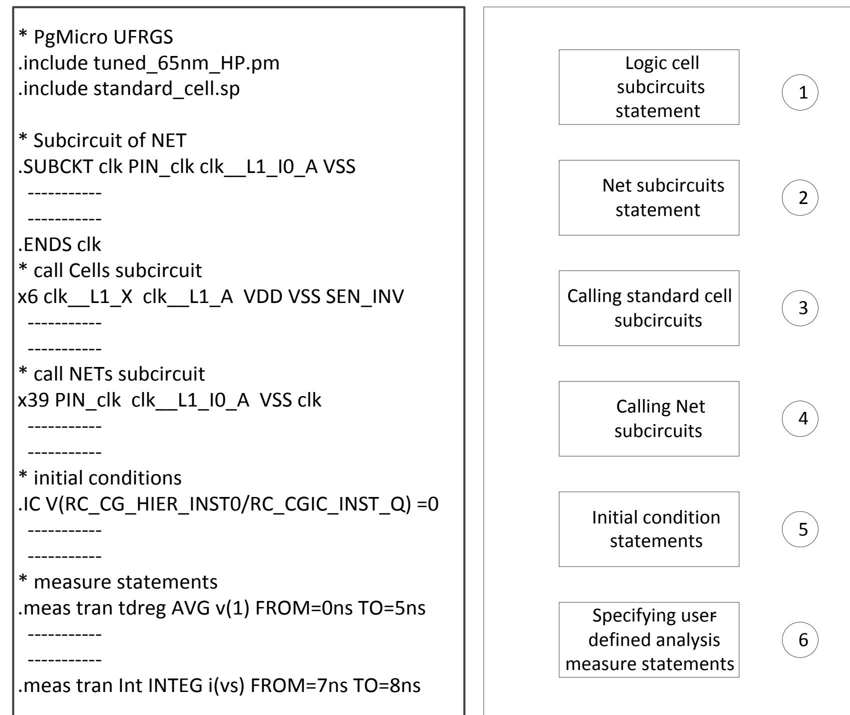


Figure 4.2 Output of EXT-CLK tool: clock network netlist and netlist separated in sections

4.2.1 Logic cell subcircuits statement

The first statement of the output netlist is about the logic cell. LVS file contains the netlist of cells used in specific design kit. Then, to elaborate the subcircuit statement from the LVS file, there are two ways to instantiate the standard cell subcircuits. The first way is instantiate each subcircuit that will be used directly in the netlist. And the second way is to attach a file that contains the subcircuit netlist. LVS files contain all gates as subcircuits but there are lots of them that are not necessarily used in the clock network. Then we take the simple way, attach the LVS file since the performance of the simulation is not affected. But before to attach the LVS file we need to modify some lines of the LVS files, adding the reference nodes "VDD" and "VSS" and rename some components. These reference nodes must be added at the end of the subcircuit statement inside the LVS files, like in example 1:

Reference nodes

.SUBCKT SEN_BUF_AS_16 X A VDD VSS

Example 1. Subcircuit statement adding reference nodes

The algorithm in Figure 4.3 searches for *subckt* word in the LVS file and add the VDD and VSS in the end of the line. Additionally we need to change the name of the transistors *nfet* and *pfet* by *nmos* and *pmos* respectively. The names of transistors are changed because those names must be the same of the technology file (PTM) otherwise we cannot simulate. After modifying the LVS file, we rename this to *standard_cell.sp* and include

this in the netlist of the clock network in *clock_network.sp*, as shows the third line in the left side of Figure 4.2.

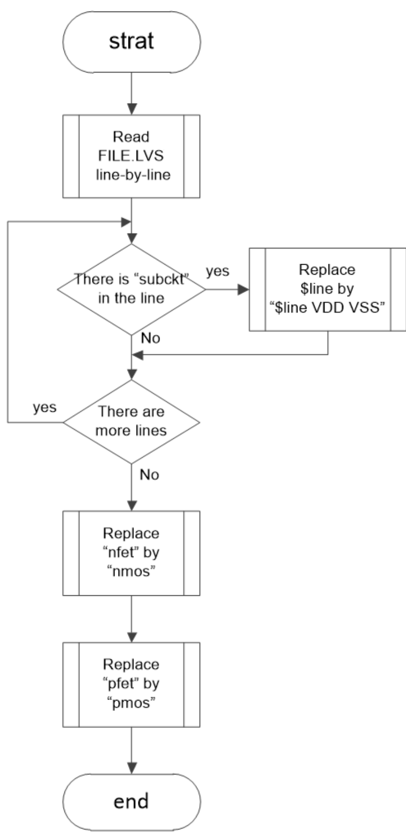


Figure 4.3 Flowchart to logic cell subcircuits statement

4.2.2 Nets subcircuit statement

Layout design has thousands of wiring conductors used to connect different gates. The net is composed of several segments of metals at different layers. We consider a net like a component which joins the logic cells by the same node. Figure 4.4 represents a net in a schematic circuit and layout circuit.

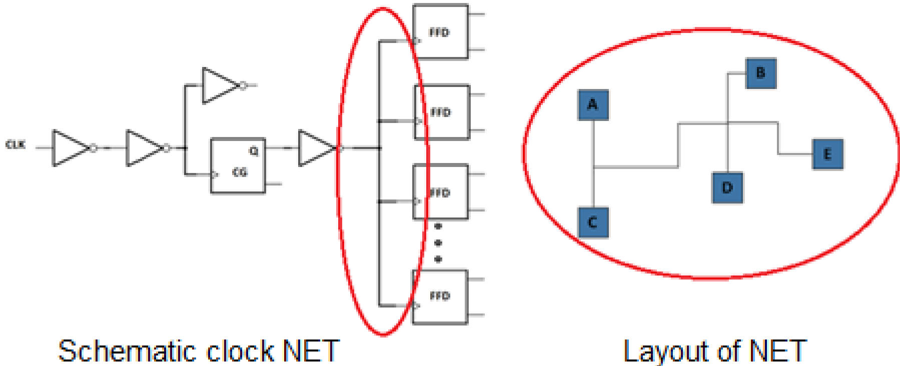


Figure 4.4 Representation of a net

Then, the goal of this algorithm in figure 4.6 is to identify which nets belong to the clock network. Nets described in DEF file allow performing this task. First step is to identify the main node of the clock network, which means what is the net where the clock signals is fed, for example the output of the PLL circuit. Defining the starting point, the algorithm searches the nets relatives to this.

In the net statement of DEF file is declared components that are connected to the same net at specific pin. Figure 4.5 shows a sample of a net in full description inside the DEF file, where the headline is the net name, and the components names in the parenthesis. Also inside the parenthesis is declared the pin of the component. The NEW command denotes the wire connection segment that starts and finishes in the coordinate mentioned in the same line.

We are interested in buffers, clock gates and flip-flops (FF), and we must identify each one. Therefore, we have to search the description of all components, mentioned in red box, inside the LVS file. Identifying the output of these components we can find the next nets that belong to the clock network. For example, if the component is a clock gate or flip-flop we are interested in the net connected to the Q pin, if the component is a buffer we are interested in the net connected to the output. Then when a next net is found, the name of the net is saved in a temporal list until to finish exploring the current net. The net name is saved to verify if the net was explored before and avoid a deadlock.

```
- rc_gclk_11788
 ( wr_1_ack_reg CK ) ( wr_0_ack_reg CK ) ( rd_1_ack_reg CK )
 ( rd_0_ack_reg CK ) ( RC_CG_HIER_INST3/RC_CGIC_INST Q )
 + ROUTED M4 ( 68000 102400 ) ( 74000 * ) Via3_MAR_E
   NEW M3 ( 68000 99200 ) ( * 102400 ) Via3_MAR_W
   NEW M4 ( 74000 102400 ) ( 80800 * ) Via3_MAR_W
   NEW M3 ( 74000 102400 ) ( * 103200 ) Via2
   NEW M2 ( 114000 106400 ) ( 114800 * ) Via1
   NEW M3 ( 80800 100400 ) ( * 102400 )
   NEW M4 ( 80800 102400 ) ( 83200 * ) Via3_MAR_W
   NEW M3 ( 83200 102400 ) ( * 104000 ) Via3_MAR_W
   NEW M4 ( 83200 104000 ) ( 114000 * ) Via3_MAR_W
   NEW M3 ( 114000 104000 ) ( * 106400 ) Via2
   NEW M3 ( 114000 103200 ) ( * 104000 )
   NEW M3 ( 68000 99200 ) Via2
   NEW M2 ( 80800 100400 ) Via1
   NEW M3 ( 80800 100400 ) Via2_MAR_E
   NEW M3 ( 114000 103200 ) Via2_MAR_E
   NEW M2 ( 114000 103200 ) Via1
 + USE CLOCK
 + WEIGHT 20
```

Figure 4.5 Example of a net description in DEF file

All nets found concerning to the clock network are saved with full description in *NETS.txt* and this file it is used as input of another algorithm. The flowchart in Figure 4.6 shows the description of the algorithm mentioned before.

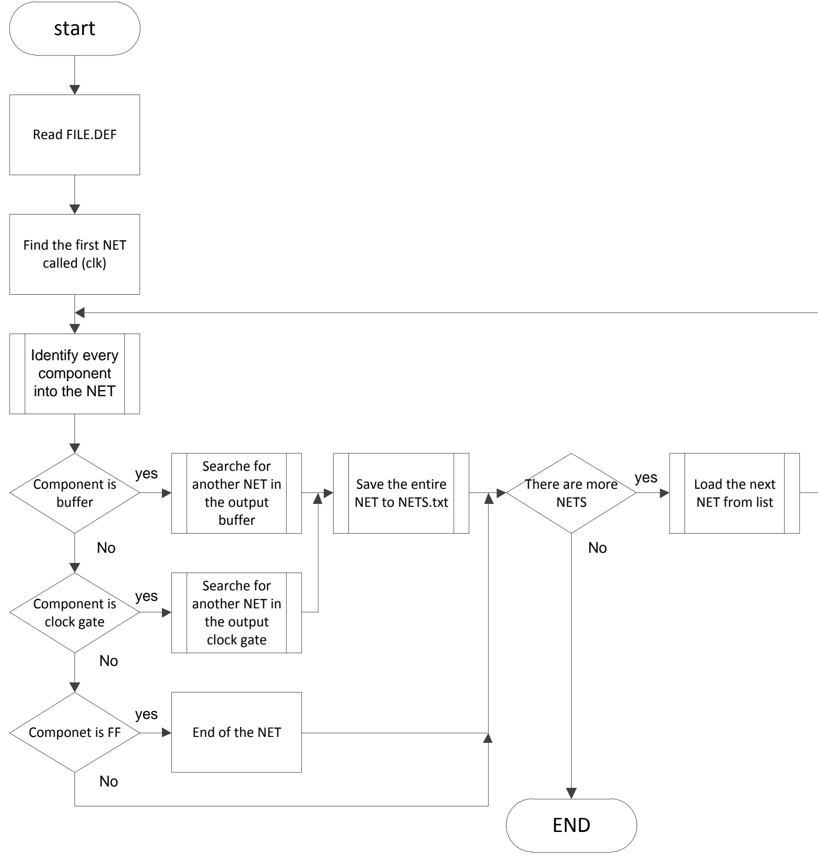


Figure 4.6 Algorithm to separate nets concerning the clock network

After separating the nets that belong to the clock network in *NETS.txt*, we need to replace the asterisk (*) by the correct number because the net description in DEF file has a particular syntax, where (*x* *) indicates that the *y* coordinate is the same value used previously in the wiring specification and (* *y*) indicates that the *x* coordinate is the same value used in previous coordinate. For example, the first line inside the red box in Figure 4.7 shows (68000 102400) (74000 *) where the asterisk replaces the 102400 from the first *y* coordinate. Then, the next algorithm is essential to facilitate the next stage. The algorithm shown in Figure 4.8 is able to do this job.

```
- rc_gclk_11788
 ( wr_1_ack_reg CK ) ( wr_0_ack_reg CK ) ( rd_1_ack_reg CK )
 ( rd_0_ack_reg CK ) ( RC_CG_HIER_INST3/RC_CGIC_INST Q )
 + ROUTED M4 ( 68000 102400 ) ( 74000 * ) Via3_MAR_E
   NEW M3 ( 68000 99200 ) ( * 102400 ) Via3_MAR_W
   NEW M4 ( 74000 102400 ) ( 80800 * ) Via3_MAR_W
   NEW M3 ( 74000 102400 ) ( * 103200 ) Via2
   NEW M2 ( 114000 106400 ) ( 114800 * ) Via1
   NEW M3 ( 80800 100400 ) ( * 102400 )
   NEW M4 ( 80800 102400 ) ( 83200 * ) Via3_MAR_W
   NEW M3 ( 83200 102400 ) ( * 104000 ) Via3_MAR_W
   NEW M4 ( 83200 104000 ) ( 114000 * ) Via3_MAR_W
   NEW M3 ( 114000 104000 ) ( * 106400 ) Via2
   NEW M3 ( 114000 103200 ) ( * 104000 )
   NEW M3 ( 68000 99200 ) Via2
   NEW M2 ( 80800 100400 ) Via1
   NEW M3 ( 80800 100400 ) Via2_MAR_E
   NEW M3 ( 114000 103200 ) Via2_MAR_E
   NEW M2 ( 114000 103200 ) Via1
 + USE CLOCK
 + WEIGHT 20
```

Figure 4.7 Asterisk convention used in net description

The algorithm to replace asterisks takes the output file *NETS.txt* generated previously, and read word by word since the beginning. When a number is found in read processes, this number is stored into the stack. Finally, if an asterisk is found, this asterisk is replaced by the last number stored. It is important to note whether the number is *x* or *y* coordinate. In case of finding an asterisk, it is replaced by the number stored in the stack taking into account the same coordinate position. The same process is repeated until the end of the file.

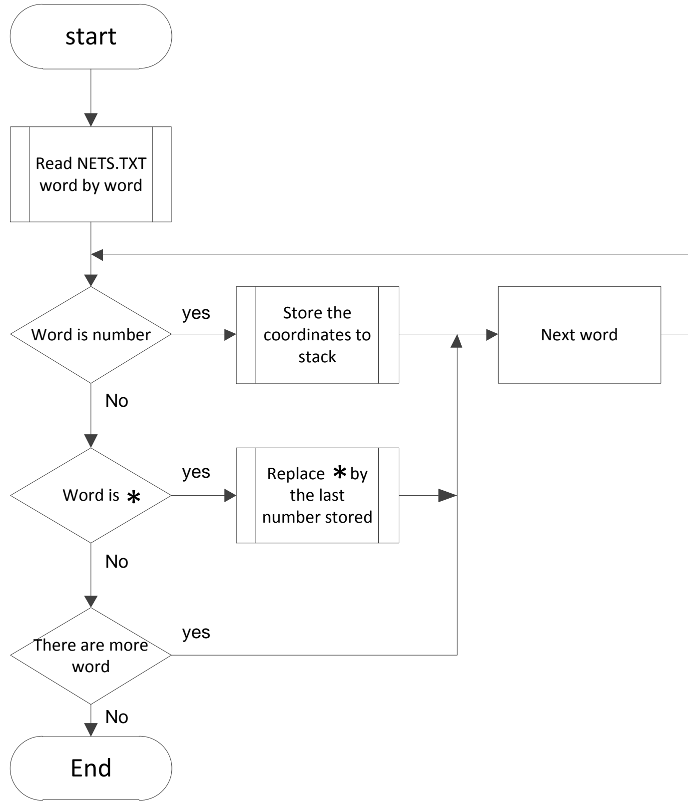


Figure 4.8 Algorithm to replace asterisks by numbers

4.2.3 Net subcircuit headline

Following with Figure 4.2 is time to create the net subcircuit statement. We start reading the *NETS.txt* file (last version without asterisks) and separate net by net to facilitate the task. Then, the net described in DEF file is converted to the subcircuit netlist, as shown in Figure 4.9. The net description in DEF file start with the net name, and the following line describes the component attached to its pin enclosed in round brackets. Every component mentioned is interconnected by the net. The subcircuit headline to be passed to hspice description is relatively simple; the net name will become the subcircuit name and the components together with its pin which will be a node of the hspice netlist. Another node that must be added in the headline is VSS reference.

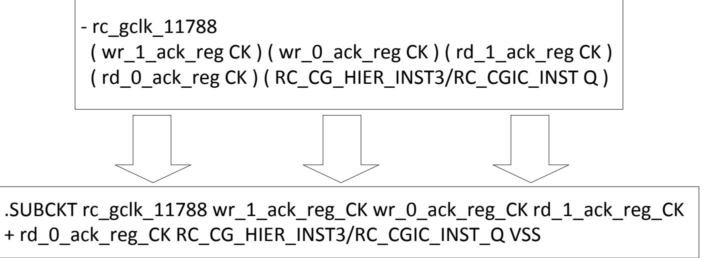


Figure 4.9 Writing the net subcircuit headline.

Therefore we can copy the complete subcircuit headline to *clock_network.sp* and save a copy into *net_headline.txt*, which will be used by other algorithm. The *clock_ network.sp* file is the netlist of the clock network extracted from the layout design.

After finishing the headline statement, we convert the wire connection segment to pi model. CRC or pi is an electrical model of the wire connection as capacitor-resistor-capacitor that allows simulates features of the wire. To calculate the value of resistors and capacitors for each segment of wire, we need information about cut layer and dimensions of the connection. The segment resistance of a wire can be defined in Equation 4.1.

$$\text{segment resistance} = RPERSQ \times wire_length / wire_width \qquad \text{Equation 4.1}$$

Where RPERSQ value specifies the resistance per square of wire, in ohms per square. RPERSQ and *wire_width* are taken from the LEF file and the *wire_length* is calculated from the difference between the coordinates in the net netlist.

To calculate the segment capacitance, the EXT-CLK uses the Equation 4.2. EDGECAPACITANCE is used only if you set layer thickness, or layer height, to zero.

$$\text{segment capacitance} = (CPERSQDIST \times wire_width \times wire_length) + (EDGECAPACITANCE \times 2\,(wire_width + wire_length)) \qquad \text{Equation 4.2}$$

Where CPERSQDIST value specifies the capacitance for each square unit, in picofarads per square micron, this value is used to model wire-to-ground capacitance. And EDGECAPACITANCE value specifies a floating-point value of peripheral capacitance, in picofarads per micron. CPERSQDIST and EDGECAPACITANCE is also found in LEF file. The *wire _width* and *wire _ length* represent the wire width and wire length respectively. Figure 4.10 shows the pi model used in our simulations.

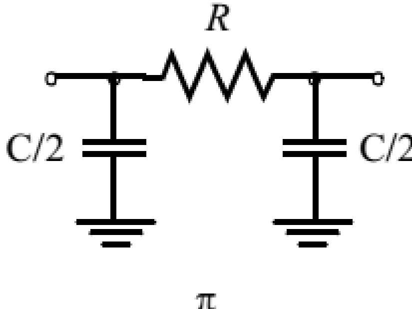


Figure 4.10 Pi model for wire simulation in hspice (Rabaey et al., 1999)

```
- clk__L1_N0
 ( clk__L1_I0 X ) ( clk__L2_I0 A )
 + ROUTED M2 ( 54400 96400 ) ( 55600 96400 ) Via2
   NEW M4 ( 55600 96400 ) ( 84800 96400 ) Via3_MAR_E
   NEW M3 ( 84800 96400 ) ( 84800 106800 ) Via2_MAR_E
   NEW M2 ( 54400 96400 ) Via1HH
   NEW M4 ( 55600 96400 ) Via3_MAR_E
   NEW M2 ( 84800 106800 ) Via1HH
 + USE CLOCK
 + WEIGHT 20
```

```
.SUBCKT clk__L1_N0 clk__L1_I0_X clk__L2_I0_A VSS
c1a clk__L1_I0_X 0 0.000006661800p
r1 clk__L1_I0_X 1 1.5726
c1b 1 0 0.000006661800p
c2a 1 0 0.000160534300p
r2 1 2 38.2666
c2b 2 0 0.000160534300p
c3a 2 0 0.000059662200p
r3 2 clk__L2_I0_A 13.6292
c3b clk__L2_I0_A 0 0.000059662200p
.ENDS clk__L1_N0
```

Figure 4.11 Net description in DEF syntax (left) and Net description in netlist hspice (right).

The *wire_width* and *wire_length* are calculated taking the difference of coordinates in *x* and *y*. Routing is always in 90 degrees. For example, Figure 4.11 (left) shows the first wire segment like (54400 96400) (55600 96400) lies in layer M2, and it will be convert in a pi model equivalent: resistor with a capacitor in each end, as shown in Figure 4.11 (right). Equivalent capacitor value in each end is half that segment capacitance described before. At this point, it is important to note the unit statement in the DEF file, in our case study is 2000, that means, every number as coordinate must be divided by 2000 to calculate the length and width segments.

After creating the netlist of each segment, it is important to recognize the coordinate of each component pin, i.e. we have to identify the coordinate correspond to each component pin, because we know the components and the routing wire but we do not know what end of the net is connected to specific component. Then, we must identify the coordinates reading the length and width of components. This information is available in the LEF file. The idea is to identify whether an end coordinate is placed within the area of the component (logic cell). Of course, several coordinate do not match any component because there are wire segments inside the net that not necessarily ends in a component.

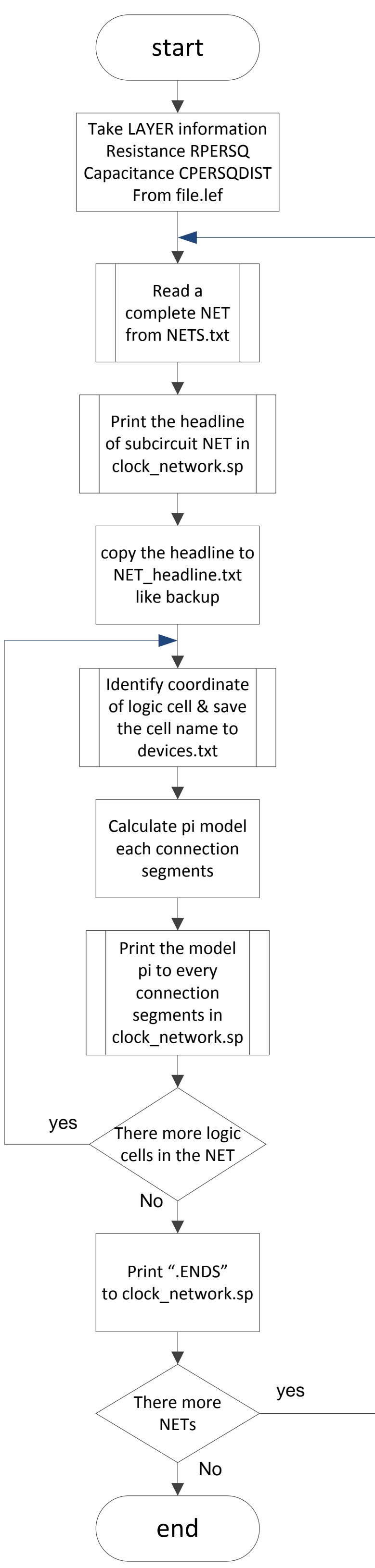


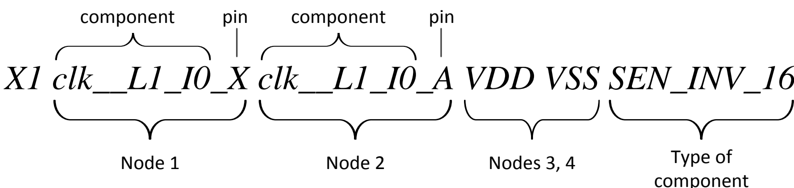
Figure 4.12 Algorithm to create the net subcircuit.

Identified the coordinate of every component, the name of component must be saved in *device.txt* to be used as input for the next step. Coordinates that do not match any

component will be assigned a number to use as a node in hspice netlist. Finally, to finish the conversion of each net to hspice netlist, we need to write the *ENDS* command and read the next net and perform the same task. The workflow of this algorithm is shown in the Figure 4.12.

4.2.4 Call subcircuit of components

The next step consists in describing the clock network using subcircuits. The call of components is split in call of cells and call of nets. In this algorithm (Figure 4.13) we describe how to call the components in the correspondent nodes. To call a subcircuit in hspice netlist we need to declare the component attached to its pin linked to a respective node and declare the type of component, see the example 2:



Example 2. Call subcircuit statement of logic cell

The node 1 represents the component together with its pin *X*, which is the output of the buffer. The node 2 is represented by the component linked with pin *A*, which is the input of the buffer.

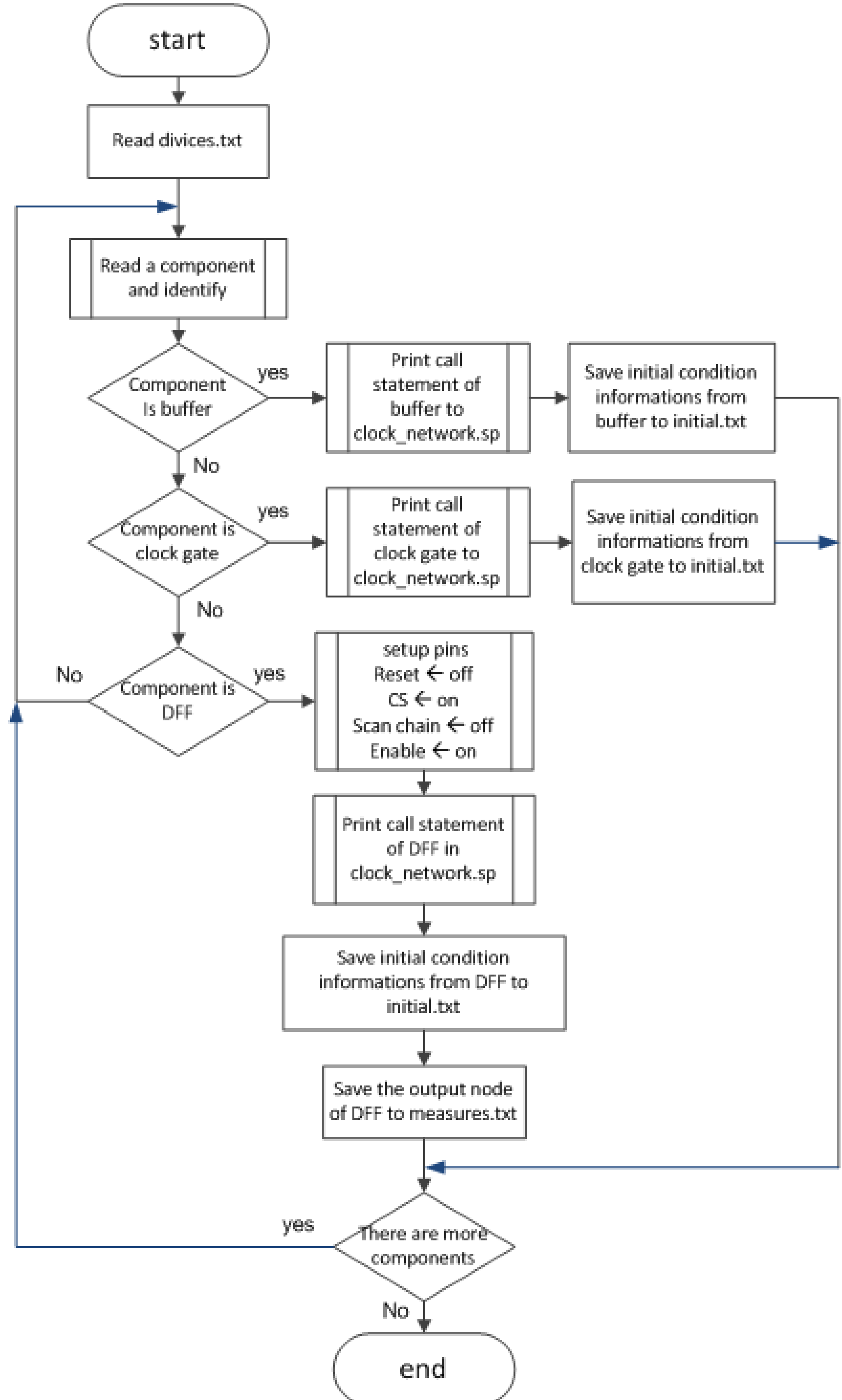


Figure 4.13 Flowchart of the algorithm to call the subcircuit of components.

To call the subcircuit of components, the algorithm takes the *devices.txt* file as input, which was created by the algorithm shown in Figure 4.12 and read component by component and identify the type of component. Remember that *devices.txt* file contain the all names of components used in the clock network. Then, we can write the call

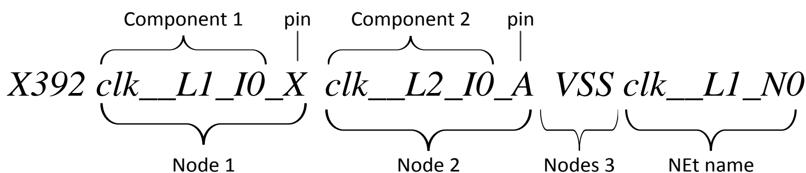
statement in *clock_network.sp* and save the initial condition statement referent to the component. Initial condition statement will be described in the following section.

Taking every component we define the call statement for three different cases, buffer, clock gate and flip-flop. In case of buffer or clock gate the call statement is very similar to the example 2. But in case of flip-flop each pin must be analyzed because there are many types of flip-flop. The idea is setup the flip-flop to allow the data pass through the component to be analyzed, therefore signals like reset, chip select, scan chain and enable must all be disabled in the statement.

The algorithm in Figure 4.13 also defines the measure statement, which allows monitoring the output of every register in clock network. In this work, all result relies on measures taken in the output of the flip-flop. Then, when a flip-flop is found, the algorithm copies the output node to *measures.txt* in order to define the measure statement later.

4.2.5 Call statement of net

Defined the call statement of components, we will link the components using the nets. *Net_headline.txt* file generated by the algorithm in Figure 4.12 is used as input in this section. *Net_headline.txt* contain all headline of nets, used to define the subcircuit statement. In this file there are many nets with several flip-flop, then to avoid a larger headline we break the headline using + character to start a new keyword. If a line start with + character means that the line belong to any nets and it is not necessary generate a call statement. To generate the call statement of net we start with *X$num*, where *num* represents the instance of subcircuit. The example 3 describes a call statement of net.



Example 3. Call statement of net

Figure 4.14 describes the flowchart of the algorithm to generate the call statement of the subcircuit explained before.

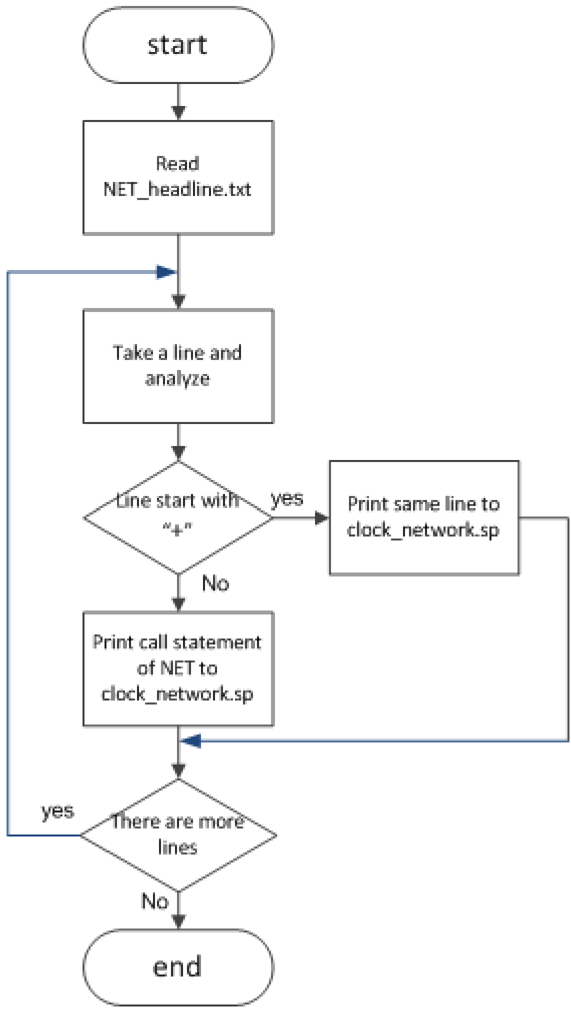


Figure 4.14 Flowchart of the algorithm to call the subcircuit of nets.

4.2.6 Initial condition statement

Initial condition (IC) statement is not required to perform the simulation but this may speedup the simulation in hspice. Initial condition is easy to declare after algorithm in Figure 4.13 create *initial.txt* file. Transient analysis uses the initial condition value as part of solution, for time-point zero. *Initial.txt* file has a list of input buffers nodes and clock gate nodes. These nodes are initialized with a value of voltage (VDD or VSS) to start the transient simulation. The algorithm in Figure 4.15 takes every line of the *initial.txt* file and adds the initial value and writes on clock network netlist.

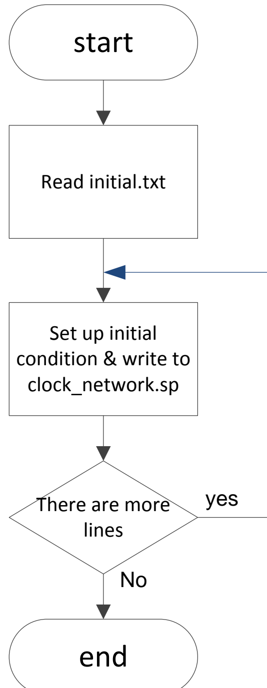


Figure 4.15 Algorithm to define initial condition.

4.2.7 Measure statement

Measure statement is used to define the results of successive simulations. The specifications include: propagation, delay, rise time, fall time, peak to peak voltage, and others user define variables. This command can use several formats, depending on the application. In our case, we use this command to analyze the output of each flip-flop. The flip-flops will be set up to 0 and we are going to take the time when it changes to 1. In eventual strike of heavy ion, if the flip-flop no change the value in a time period, we consider that flip-flop has not been affected. Thus, we use the delay to see if the flip-flop is affected or not.

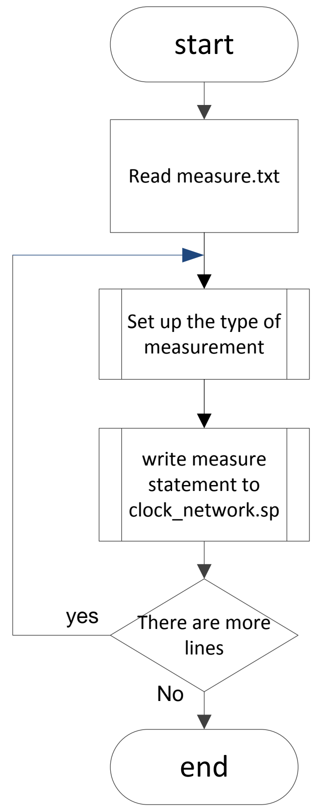


Figure 4.16 Algorithm to measure statement

Using *measure.txt* file, created by the algorithm in Figure 4.13, we define measure statement. *Measure.txt* file contains all output nodes of flip-flops which belong to clock network. Every node is part of measure statement. The example 4 describes a measure statement.

*.MEASURE TRAN TD_REG_Q TRIG AT=5.5N TARG V(REG_Q) VAL='0.5*VDD' FALL=1*

delay

DFF node

Example 4. Measure statement in hspice

In this example, *TRAN* command is a type of simulation (transient), *TD_REG_Q* is an output variable that will take a solution. This measure will start at 5.5 nanoseconds until the *REG_Q* node falls and takes half VDD value. A simple algorithm was made for this task shown in Figure 4.16. Finally when the netlist of the clock network is done, we perform the simulation to evaluate the correct behavior of the circuit. We need to ensure that the signal reach all flip-flops from the clock source with low skew and without attenuation. After verification of the clock network we can proceed with the fault injection simulation.

5 FAULT INJECTION SIMULATIONS

This chapter presents the clock network of the SRAM arbiter modeled in hspice netlist and simulated in electrical level and in logic level in 65 nm CMOS technology. The following simulations are based on the clock network extracted from the layout design. Three simulations regarding to SET injection have been developed in electrical level:

- SET injection in every node
- SET injection in particular path
- SET injection in the clock source.

In every case, the SET pulse was injected at the output of the buffer in the circuit. Others simulation that we performed are:

- Replacing small buffers on clock tree
- Soft error rate on clock tree
- Clock mesh vs clock tree

From the electrical level simulation, only the SET injection in the clock source was taken into account to calculate the soft error rate mentioned before. We are using this simulation because a SET pulse in the clock source may affect all registers at the same time.

At this point, it is important to review a concept that help us to quantify the result, critical charge. Critical charge is a circuit parameter conveying the minimum charge collection which induce upset in a SRAM cell. Typically, the critical charge is computed using circuit simulation, assuming a double exponential ion-induced current pulse at the struck node. The equation which describe the total collected charge is given in the Equation 5.1.

$$Q = \int_0^t I_{drain}(t)dt \qquad \text{Equation 5.1}$$

The charge will be a critical charge when the gate voltage of the struck transistor exceeds the threshold voltage. At this point, the transistor switches from off to on. As mentioned before, the current of the transistor at this situation is modeled assuming a double exponential and is describe by the Equation 5.2.

$$I_{drain}(t) = I_o(e^{-t/\tau_\alpha} - e^{-t/\tau_\beta}) \qquad \text{Equation 5.2}$$

Where I_o is approximately the maximum charge collection current, τ_α is the collection time constant of the junction and τ_β is the time constant for initially establishing the ion track. The maximum charge collection current depends on the ionizing particle linear energy transfer (LET) value and the process technology. When the values I_o, τ_α and τ_β are defined for a given technology, any circuit designed in that technology may be evaluated (Wirth et al., 2007).

In hspice there is an exponential source function that can help us to model the drain current. The general syntax for an exponential source in an independent voltage or current source is described in:

Ixx n+ n- EXP (V1 V2 TD1 TAU1 TD2 TAU2)

Where:

Ixx is an independent current source.

n+ and n- are point of connection of the current source

EXP is the keyword for an exponential time-varying source.

V1 is an initial value of current, in amps.

V2 is a pulse value of current, in amps.

TD1 is a rise delay time, in seconds.

TD2 is the fall delay time, in seconds.

TAU1 is the rise time constant, in seconds.

TAU2 is the fall time constant, in seconds.

In Figure 5.1 is shown an example of exponential source function. The waveform rises exponentially, from -4 V to -1 V, with a time constant of 30 ns. At 60 ns, the waveform start dropping to -4 V again, with a time constant of 40 ns. That description corresponds to:

Idrain 1 0 EXP (-4 -1 2ns 30ns 60ns 40ns)

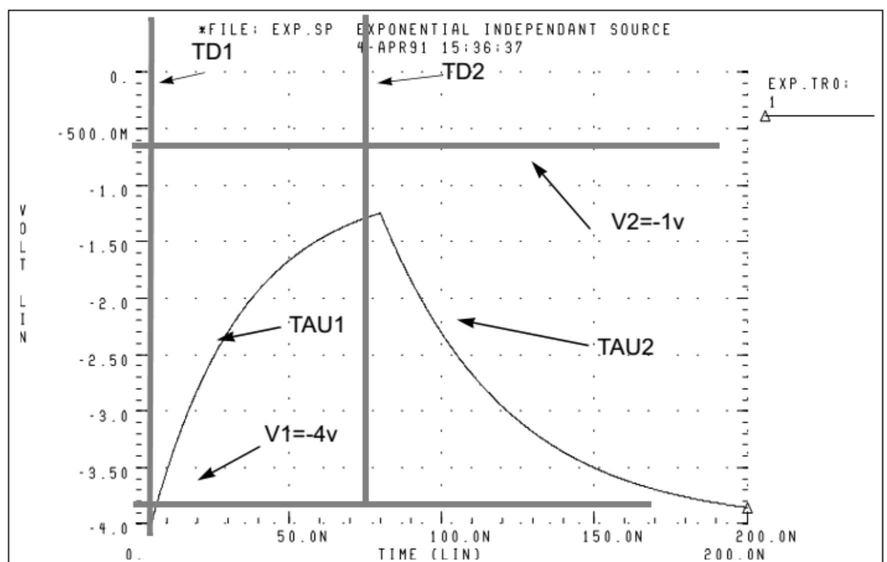


Figure 5.1 Example of exponential source function

With the drain current modeled, we need to evaluate the integral of the current from zero to *t*, where the *t* value is the duration of the transient pulse. Fortunately, hspice has the INTEG function, and mixed with measure statement we can compute the critical charge, like this:

MEASURE TRAN INT INTEG I(VSET) FROM=0NS TO=8NS

Where the measure is transient and the result of this operation will be in the variable INT. The function to be integrated is I(VSET) from 0 ns to 8 ns.

5.1 Case study: SRAM arbiter

SRAM arbiter circuit has been designed in 65 nm process using Encounter tool from Cadence. It has been designed based on minimum skew and minimum delay. The SRAM arbiter is the interface between the logic modules of a Gigabit Ethernet Switch and an external zero-bus turn around (ZBT) SRAM memory. Figure 5.2 shows the clock tree in the SRAM arbiter circuit.

Figure 5.2 Clock Tree Network in SRAM arbiter circuit.

By analyzing the clock tree extracted we found that it has 5 clock gates and 18 buffers, and 367 registers (flip-flips). These buffers can be inverters or non- inverters and each device presents different size and others features as shown the Table 5.1.

Table 5.1 Types of Buffers

Type	Description	Drive option
A	Non-inverter buffer	1,1.5,2,3,4,6,8,10,12,16,20,24
B	Symmetric buffer, anntena diode	0.5,1,1.5,2,3,4,5,6,8,10,12,16
C	Symmetric buffer	1,2,3,4,6,8,12,16
D	Symmetric inverter	0.5,1,1.5,2,3,4,6,8,12,16,32
E	Symmetric inverter, anntena diode	0.5,1,1.5,2,3,4,5,6,8,10,12,16
F	inverter	0.5,0.65,0.8,1,1.25,1.5,2,2.5 3,4,5,6,8,10,12,16,20,24,32
G	Delay buffer	1,2,4,8
CG	Clock gater	1,2,3,4,6,8,12,16

Where non-inverter buffer is composed of two inverters, the first inverter is smaller than the second inverter. Symmetric buffer antenna diode is a buffer with an antenna diode on the input to provide alternative path for discharging static charge and protect the gate. Symmetric Buffer is a synchronous buffer, which has the same rise and fall time design. Also we have found a clock gate (CG). Clock gate is used to control a segment of the clock network. With this control it is possible to minimize the power consumption of the circuit. Some of these inverters and buffers are also used inside the logic circuit and not necessarily in clock network. The SRAM arbiter has 367 registers of one bit like a flip-flop D. The clock tree in SRAM arbiter uses four types of flip-flops as registers. All the flip-flops in the clock tree may be used as scan chain, and some of these have clear pin. In Table 5.1, the drive option means the relative size between cells.

After describing the case study we describe the simulations performed in the clock tree of the SRAM arbiter.

5.2 SET injection in every node of the clock tree

These simulation allows to make a profile of the susceptibility of the clock tree network for a specific circuit design. This profile help us to identify vulnerable nodes in terms of critical charge and get the percentage of fail registers.

The electrical simulations were performed using 65 nm CMOS technology of Predictive Technology Model (NIMO, 2012). The transient pulse is approximately 70 picoseconds and the current amplitude is swept from a few nanoamperes until to produce a fault in the registers. To make the automatic sweeping of current we use the optimization option in hspice. The final current value which produce a fail in the clock tree is computed the charge and to make a profile. Therefore, the results are in terms of Coulomb (C). In this simulation the SET injection is performed at the output of buffers until generate a transient pulse that can generate a bit-flip in one of the 367 registers. We are interested in sensible nodes, then by now we disconsidering which register is flipped. This process is repeated, buffer by buffer, until generating the distribution of minimum charge of the clock tree.

We must to note that, in many paths of the clock tree, there are clock gates which are used to block the clock signal and they can also block the SET propagation. Then, for SET simulation all clock gates are set to allow the propagation of the clock signal from the clock source to all registers. Also, all registers were set up to store 1. Then, in eventual SET effect, the registers change the value store from 1 to 0 and we take this event as a bit-flip.

The automatic SET injection was developed in multithreading level using Intel core quad CPU at 2.66 GHz with 8 GB of memory installed (RAM). As result of this simulation, the Figure 5.3 shows the minimum charges that can provoke an upset in any clock tree registers.

The more robust node found in the clock tree is the output of the buffer *rc_gclk_L2_I0*, because it is necessary to inject 1160 femtocoulomb (fC) to produce an upset in one register. Also, the buffer *rc_gclk_L2_I0* is at level five of the clock tree and this is an inverter of size 16X.

One of the most susceptible node found in the clock tree is the output of the buffer *clk__L5_I0,* which needs 26 fC to produce bit-flip in one of the registers. The buffer *clk__L5_I0* is also at level five and this is an inverter of size 0.8X. A charge of 26 fC in this node can produce an upset and provoke abnormal behavior on the system. In fact, simulations show that first flip-flop to fail is *do_reset_reg*. There are others nodes with high susceptibility that need to be study to see the type of error they may produce.

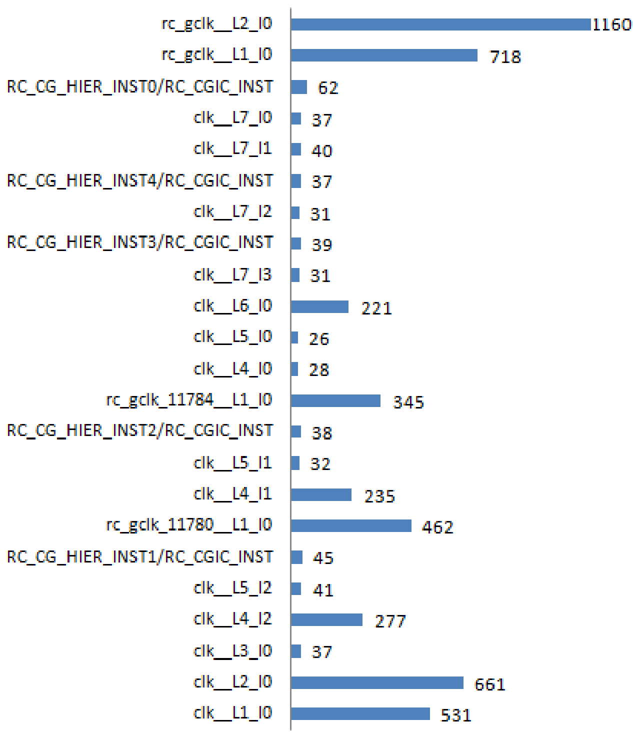


Figure 5.3 Minimum charge injected in the output of buffers to produce a bit-flip in at last one register.

5.3 SET injection in a particular path of the clock tree

This simulation allows to make a profile of a specific path of the clock signal, and shows the relation between the charges needed to provoke fails and size of the cell used in the path. In this particular path, shown in Figure 5.4, there are three types of buffers with different sizes. The type of buffers used in this clock path are F(16), F(0.8), C(8) and clock gate CG(1). The clock gate is a circuit to control the flow of the clock signal. The number inside the parenthesis refers to the drive strength, for example F(16) means that the PMOS and NMOS transistors are sized with effective width of 9.6 µm and 7.68 µm, respectively. While F(0.8) is with 0.48 µm and 0.38 µm to PMOS and NMOS, respectively. At the end of the path, there are tens of flip-flops connected to the buffer C(8). Table 5.1 shows the different types and sizes of buffers using in the design kit.

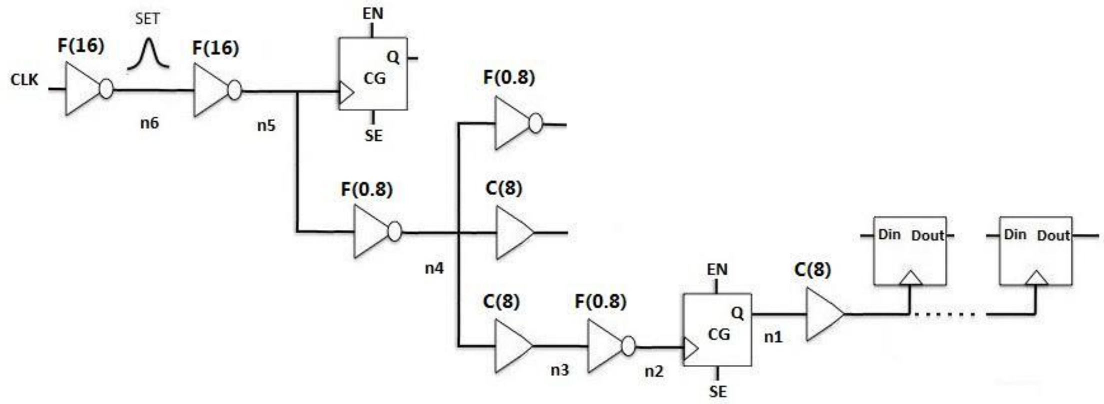


Figure 5.4 Clock path in SRAM arbiter to be analyzed.

In this simulation we inject transient pulse in each output buffer of the clock path selected. The result shown in Figure 5.5 represents the distribution of minimum charge to provoke the first bit-flip in any register.

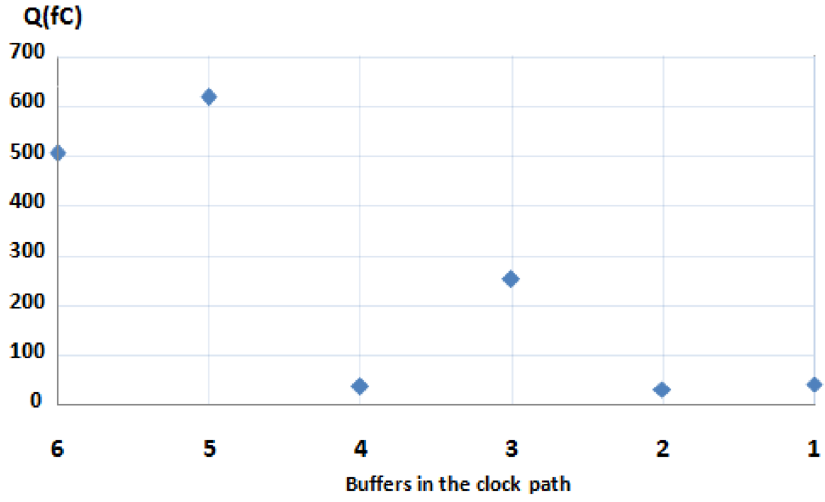


Figure 5.5 Critical charge results in the SRAM arbiter clock path

Thus, we found a distribution of charge from 38 fC to 617 fC. The most vulnerable nodes are *n4, n2* and *n1*, in contrast the less vulnerable nodes are *n6* and *n5*. The most vulnerable are the buffers with smaller drive strengths. Nodes *n4* and *n2* are placed at the output of buffer F(0.8) with 0.48 µm and 0.38 µm to PMOS and NMOS respectively. The first conclusion of this simulation is that there is a direct relationship between low capacitance node and sensitive nodes. The size of the transistors give a reference of capacitance.

5.4 Replacing small devices on clock tree

In this simulation we take the risk of change small devices by larger on the clock tree just to see the difference of use large devices and see what charge is needed to provoke a fault. Probably this action may increase the clock skew from the original design or increase the delay of some path. But is a necessary risk to see the direct relationship between the size transistor and sensitive nodes. As final result, we show the comparison between before replace and after replace the devices in terms of charge.

The process of replacing is made manually. The smallest devices is replaced by one of 10 times higher approximately. For example, the device number 1 in Table 5.2 is an inverter of 0.8X and is replaced by one of 10X. But the device number 3 is a buffer of 8X and in this case is replaced by one of 16X because it is enough large to increase by 10 times. This criteria is apply to device number 3, 4 and 8 all of them buffers.

For example, before changing the device number 1, the output of this inverter needed a charge of 41.7 fC to produce a fault at 105 registers. After replacing this device by one 10 times higher this node need 370 fC at 3.4 mA to generate faults in 29 registers. This means that the node was significantly improved, Table 5.2 shows more details of this simulation.

Table 5.2 Details of SET simulation replacing the small devices

					Before replacing logic cells		After replacing logic cells		
No	Device	Logic cell	Description	SET pulse	Charge @ current	Reg. Fails	Logic cell	Charge @ current	Reg. Fails
1	clk__L3_I0_X	SEN_INV_0P8	inverter	from 1 to 0	41.7fC @390uA	105	SEN_INV_10	370fC @3.4mA	29
2	clk__L4_I0_X	SEN_INV_0P8	inverter	from 0 to 1	48fC @450uA	14	SEN_INV_10	622fC @5.7mA	10
3	clk__L4_I1_X	SEN_BUF_S_8	buffer	from 1 to 0	316fC @2.9mA	19	SEN_BUF_S_16	578fC @5.3mA	19
4	clk__L4_I2_X	SEN_BUF_S_8	buffer	from 1 to 0	300fC @2.8mA	58	SEN_BUF_S_16	577fC @5.3mA	2
5	clk__L5_I0_X	SEN_INV_0P8	inverter	from 1 to 0	31fC @290uA	3	SEN_INV_10	375fC @3.5mA	10
6	clk__L5_I1_X	SEN_INV_0P8	inverter	from 0 to 1	45fC @420uA	19	SEN_INV_10	578fC @5.3mA	19
7	clk__L5_I2_X	SEN_INV_0P8	inverter	from 0 to 1	59fC @550uA	72	SEN_INV_10	599fC @5.5mA	35
8	clk__L6_I0_X	SEN_BUF_D_8	buffer	from 1 to 0	283fC @2.6mA	7	SEN_BUF_D_16	556fC @5.1mA	7
9	clk__L7_I0_X	SEN_INV_0P8	inverter	from 0 to 1	47fC @440uA	3	SEN_INV_10	535fC @4.9mA	2
10	clk__L7_I1_X	SEN_INV_0P8	inverter	from 0 to 1	48fC @450uA	4	SEN_INV_10	546fC @5 mA	4
11	clk__L7_I2_X	SEN_INV_0P8	inverter	from 0 to 1	45fC @420uA	3	SEN_INV_10	579fC @5.3mA	3
12	clk__L7_I3_X	SEN_INV_0P8	inverter	from 0 to 1	45fC @420uA	4	SEN_INV_10	579fC @5.3mA	4

There are some devices that we obtain no significant result in comparison to others, as shows the Figure 5.6. As the devices number 3, 4 and 8 (all of them buffers) was replaced by cells two times higher, then the charge needed to generate faults is almost duplicated. For example, device 3 (buffer of 8X) needs 316 fC of charge to generate faults, and after replacing by one two times higher this output need 578 fC to generate the same faults. This result is similar to device number 4 and 8.

But finally in all cases the result of replace small devices improved the nodes against SET effects as we expected. The main purpose is compare the quantity of charge needed to generate a fault before replacing and after replacing the logic cell.

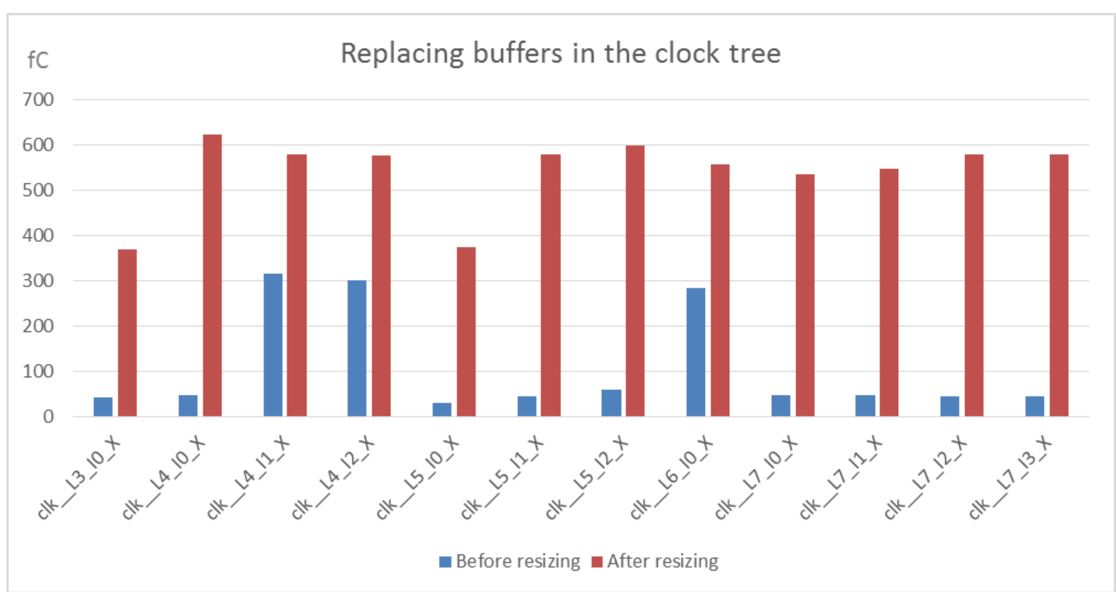


Figure 5.6 Result of the SET simulation replacing small devices

5.5 SET injection in the clock source

Through this simulation we know the susceptibility of each register of the circuit. In this simulation, the transient pulse is injected over the root of the network and also we monitored the outputs of all registers. The result shows the minimum value of charge to get a fault. The result allows to know the percentage of fail to each register. The percentage of fail from each register will be used to know the soft error rate of the circuit together with simulations in logic level, in the next section.

As previous simulation, the SET pulse has a duration of 70 ps and modeling as exponential source function. Analyzing the result, we can see the SRAM arbiter circuit starts to fail with charge higher than 435 fC at 4 mA up to 100% of fails. Figure 5.7 shows the percentage of registers upset. The result shows that 489 fC provoke 2% of bit-flips in the entire clock tree, which means 7 of 367 registers failed. And this reaction progressively increases until reach 100% of registers fails with 1,109 fC at 10.2 mA. The big step of fail from 870 fC to 979 fC is because a new branch of the clock tree is reached and probably that branch has several sinks. Just for make a comparison, a regular register on the same technology needs approximately 26 fC to be flipped. Then 1,109 fC is considered too much charge for 65 nm CMOS technology.

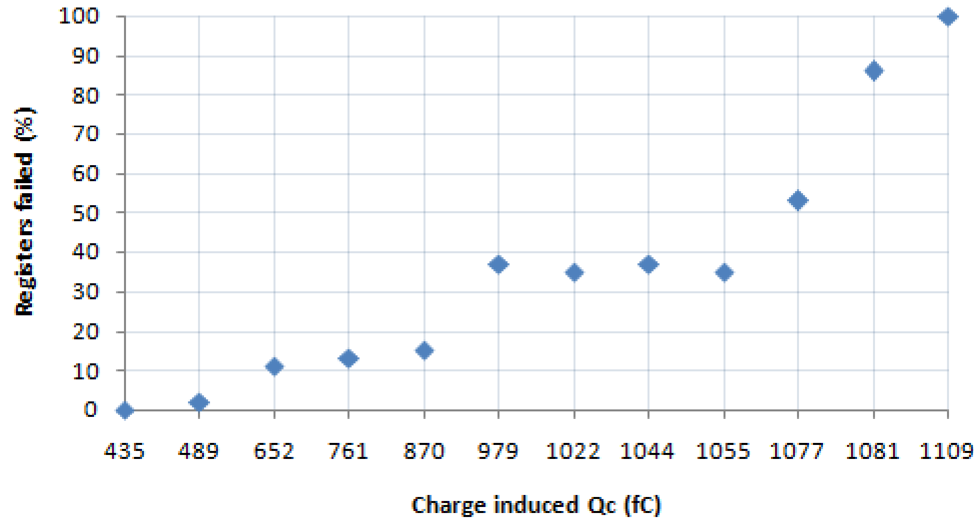


Figure 5.7 Percentage of registers failed due to induced charge injected in the clock source.

We performed 24 simulation from other point of view, this time to get the error rate of all registers. This simulation allows to know the most sensitive and the least sensitive registers. Figure 5.8 shows the results of the bit-flips due to SET injection in the clock source, for better visualization, we renamed the 367 registers from register_1 to register_367 but we can highlight some registers with additional information. For example, the register_5 is *do_reset_reg* which is the one with more sensitive, because it has 95% of error rate. The same behavior are found in register_[0,1,2] which are the first three of register *counter_reg[3]*. The same behavior is observed from the register_87 to register_104 that represent 18 registers of *sram_addr_reg[18]*. The average of error rate at all registers is 18%.

All these information we take account the next section to get the soft error rate.

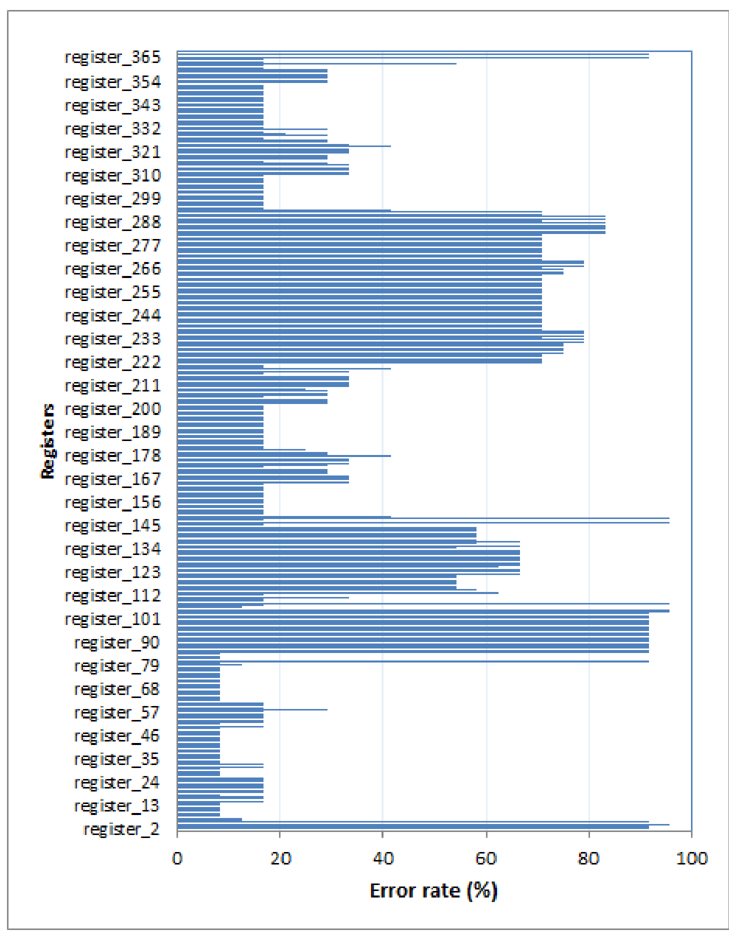


Figure 5.8 Bit-flip rate in registers due to SET injection by electrical simulation.

5.6 Soft Error Rate methodology

This simulation combines electrical level and logic level simulation. The combination of these results is the soft error rate of each register of the clock tree. Logic level simulation is done to see the masking effects when the circuit is working. As mentioned in section 2.3.2 the masking effects or latching-window masking can block the SET pulse to reach the registers.

In order to analyze the soft error rate due to SET in the clock tree network, it is important to consider two types of errors: the result of SET in the clock tree to provoke a bit-flip in one or more of the register and the result of a bit-flip in one register to provoke a failure in the circuit outputs.

Logic-level simulation is done to analyze the registers that cause failure in the Ethernet Switch when an application is running. The procedure used in this chapter is to modify the values stored in registers a period of time and verify correct function of Ethernet switch. The way to simulate the SEU injection was using a testbench for the Ethernet Switch. One of the functions of circuit is to find the output port for each frame received. The circuit must to learn the address of the frames and in this process it is necessary to access the memory. The testbench also generates random data to simulate frames arrived for processing.

SEU injection was performed at logic level using the ModelSim from Mentor Graphics environment. Only one fault is injected in one register per testbench execution. The circuit has 367 signals that represent the output of each register. At one point the testbench uses the RAM memory to store data, and then we compare the data stored in two situations, when a fault was injected and a normal execution. A gold execution is named when a testbench is running without fault injection (normal execution) and this result is stored to future comparison. In every fault injection a signal is changed at any time by 100 ps, if occurs any difference in the comparison, an error is reported. This process is done for each one of 367 registers at 175 simulations.

The Equation 5.3 defines the SER of certain SET pulse to generate an error in the circuit outputs:

$$SER_{SET\text{-}clock\text{-}tree} = ErrorRate(_{SET \rightarrow bit\text{-}flip}) \times ErrorRate(_{bit\text{-}flip \rightarrow circuit\text{-}failure}) \quad \text{Equation 5.3}$$

The equation explains the soft error rate of transient pulse in clock tree network generates a bit-flip in one or more registers, and these bit-flips (no filtered) provoke a failure in the circuit output. Each part of the equation is:

$ErrorRate(_{SET \rightarrow bit\text{-}flip})$ is the result of a SET in the clock tree reaches the clock input of the flip-flop and generate a bit-flip. This result is shown in the section 5.5 and is used to obtain the $SER_{SET\text{-}clock\text{-}tree}$.

$ErrorRate(_{bit\text{-}flip \rightarrow circuit\text{-}failure})$ is the result of a bit-flip to generate an fault in the circuit output while a testbench is in execution at ModelSim environment. According to the application and the time that the bit-flip occurs, this effect can be observed in the output or it can be masked by the logic gates.

Figure 5.9 shows the result of SEU injection by logic simulation. A set of 17 registers are shown to be more vulnerable. This means that 4.6% of 367 registers of the circuit have more chance to generate a failure in the application in eventual SEU. One of the most susceptible register is the register_[5], which represents the *do_reset*. This register provoked failures in the memory in 96% of the simulations. This means that the *do_reset* register provoked 168 errors in 175 simulations, which can be considered a main concern. A failure in *do_reset* register means that SRAM arbiter can be reset and loses data that is being computed in that moment.

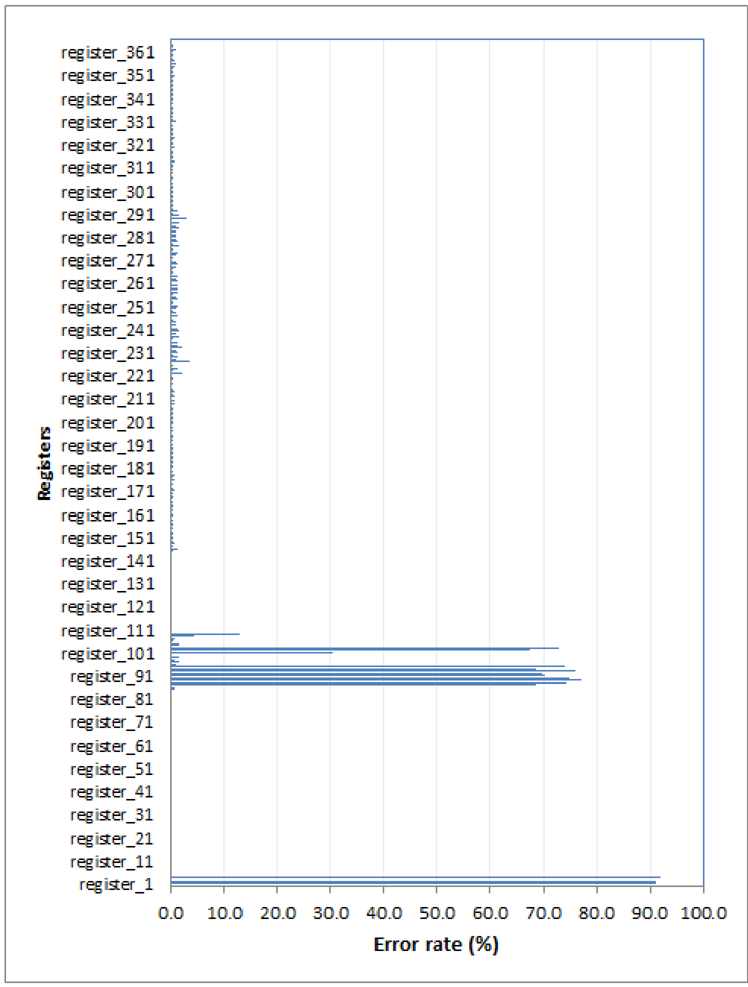


Figure 5.9 Error rate the circuit outputs due to SEU injections in the registers by logic simulation.

Registers with the same behavior under SEU injection are register_[0,1,2] that represent the three first register of *counter[3]*. *Counter[3]* register has 4 bits and these control a state machine of 16 states, 12 states for reading and 4 states for writing. Then, if the *counter[0]* register fails due to SEU, it is likely that the state machine will fail provoking an abnormal reading behavior. *Sram_addr[n]* is another register with high incidence of failure because this is represented by register_89 to register_100. *Sram_addr[18]* has 19 bits and 12 of them have approximately the same incidence of failure.

5.6.1 Soft error rate due to SET in clock tree networks

Considering concluded simulations in electrical level and logic level, we computed the soft error rate due to SET propagation in clock tree network of the SRAM arbiter circuit. Then, taking the results shown in Figure 5.8 multiply by the Figure 5.9, in accordance with Equation 5.3, we obtain the soft error rate due to SET in the clock tree network. For better visualization, we filtered the registers with less than 4% of soft error

rate. Therefore, Figure 5.10 shows the registers with higher SER as result of the product. This means that 4.6% of 367 flip-flops in the circuit are very susceptible to SET propagation in the clock tree network.

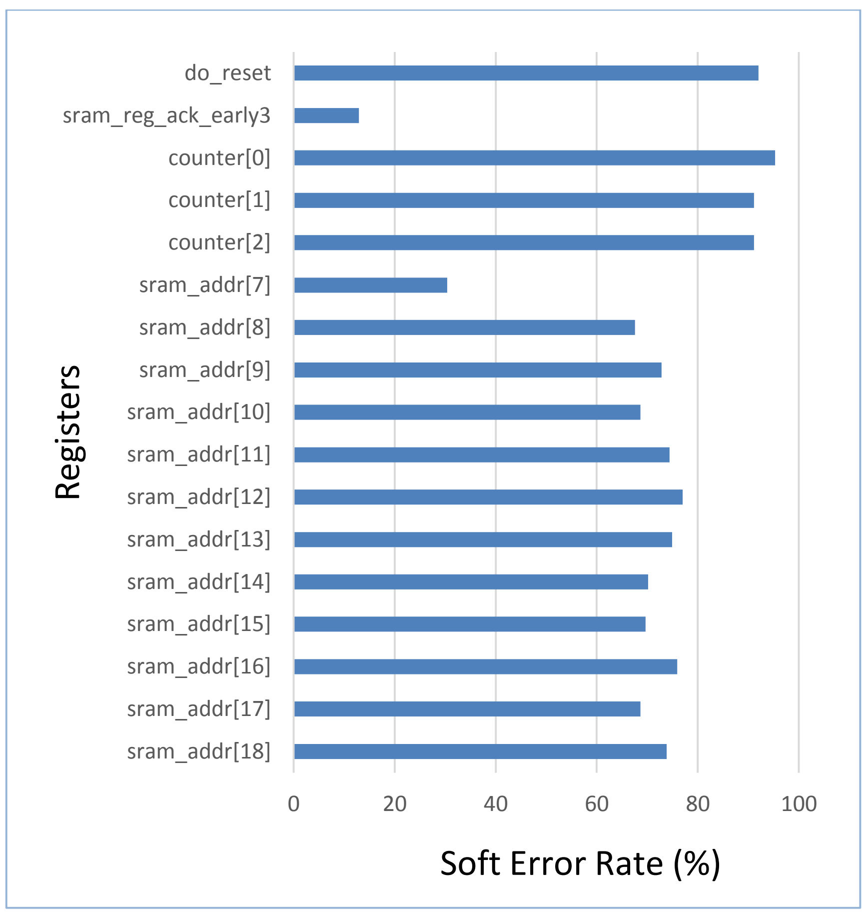


Figure 5.10 Soft error rate due to SET in the clock tree network.

5.7 Clock Mesh vs Clock Tree

Buffered clock tree, H-trees, clock mesh are commonly used topologies, each one with individual feature like latency, power dissipation, skew. But until this point, no one has studied these topologies in radiation environments. In this section we analyzed two types of clock distribution network using the same circuit. The goal is to know which one is more sensitive to radiation threats. Using a same case study we compare clock tree with clock mesh.

After using the EXT-CLK for extraction at both cases, we obtain the profile of the clock tree network. We perform the simulation of section 5.5 again (SET injection in clock source) to extend the result and make a good comparison. This time we make 140 interactions, where the width of pulse is approximately 70 ps and the current amplitude is swept from 1mA up to 15 mA increasing by 100 µA. The SET injection is performed in the output of the first buffer from the clock source All registers were set to store 1, this means, in eventual SET effect, the register can change the value stored from 1 to 0 and

we take this event as a bit-flip. In each interaction, we take the percentage of failed registers to create the profile of the clock tree.

Figure 5.11 shows the layout design of the clock tree. Figure 5.12 shows the new profile of the SRAM arbiter using the clock tree where the SET injection charge is from 100 fC to 1200 fC.

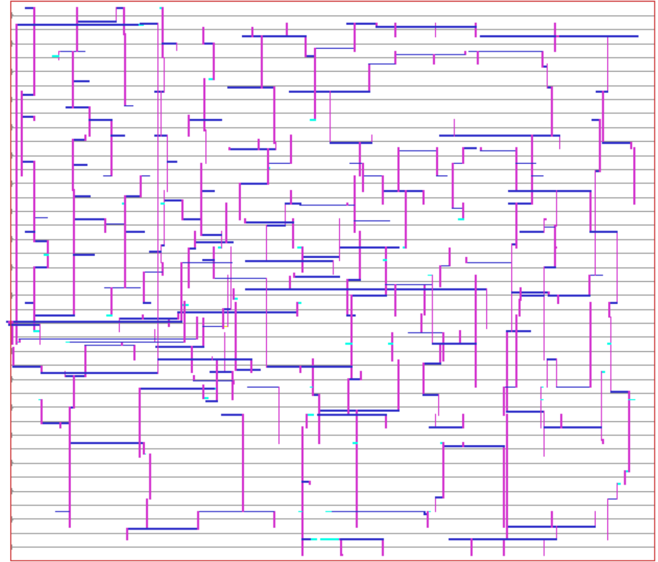


Figure 5.11 SRAM arbiter layout using clock tree topology at 65nm.

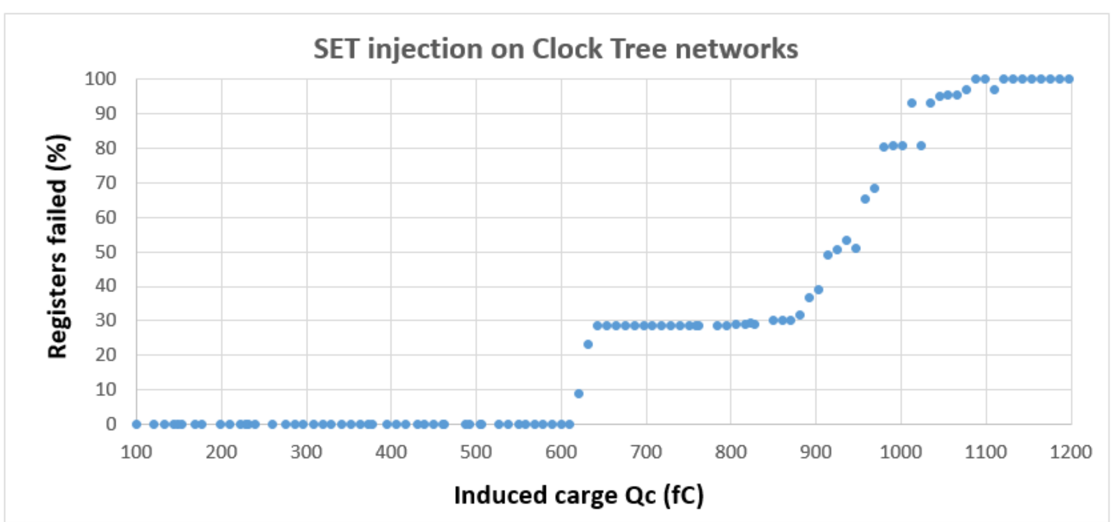


Figure 5.12 Profile SRAM arbiter using clock tree topology.

The mesh topology is build using horizontal branches and vertical trunks in two adjacent layers. An additional algorithm was developed to identify the mesh (grid) inside the DEF file. Then, the algorithm search the mesh described by a set of coordinates. With these information we can obtain the pi model of the entire mesh, but it is not enough because we need to connect others paths related to local tree. The root initiation of small local trees is not explicitly connected to the mesh (reading the DEF file) but we know the

coordinate of each root. Then the algorithm take each coordinate and calculate if that coordinate is within the rectangle of some branch or trunk of the mesh. This operation is necessary because we have to split the branch or trunk to connect the small local clock trees. Knowing where root are connected, we can make the electrical model of each path together with the mesh.

Figure 5.13 represents the electric model of the mesh (without capacitors for better illustration at right side) where CG and FF are clock gates and flip-flop respectively.

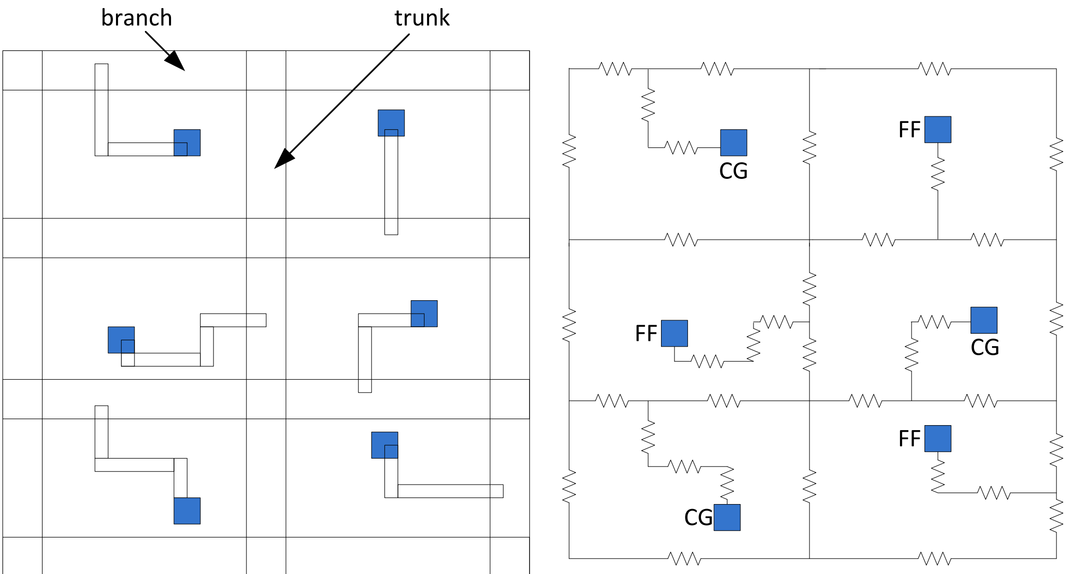


Figure 5.13 Electric model of mesh.

The clock mesh design in SRAM arbiter has a top-level chain with two pre-driver buffers, a global mesh with 4 pre-drive buffers, 16 final-driver buffers and 4 trunks by 4 branches with 22 µm of pitch. The local trees connected on the grid have 49 clock gates to drive 367 flip-flops. After the extraction of the clock mesh using EXT-CLK, we generate a profile performing 151 interactions from 1.258 mA up to 1.273 mA increasing by 0.1 µA. The electrical simulation was developed taking the same consideration of the clock tree, *i.e.,* the SET injection is performed in the output of the first buffer from the clock source. Figure 5.14 shows the clock mesh layout in Encounter environment and, the Figure 5.15 shows the profile of clock mesh from 199 fC to 202 fC.

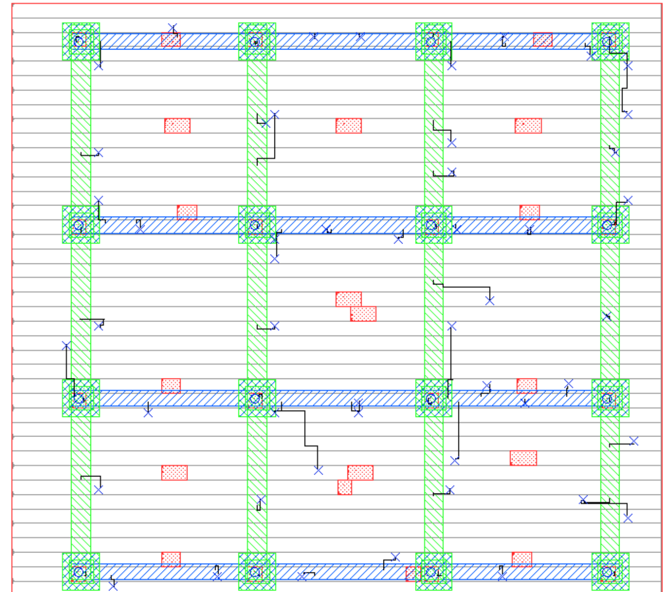


Figure 5.14 SRAM arbiter layout using clock mesh.

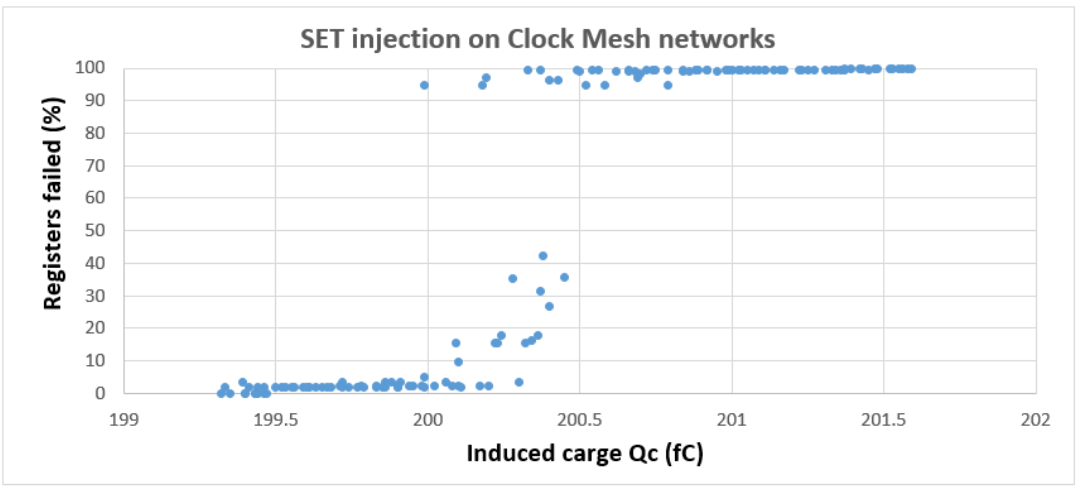


Figure 5.15 Profile SRAM arbiter using clock mesh.

After simulate of the SRAM arbiter using clock mesh, we found interesting result. The circuit reaches the 100% of fails at just 200.5 fC in almost a step. In small window of charge the 367 registers of the circuit are affected. To see a good comparison with a clock tree, we placed the two profile at the same scale and we show in the Figure 5.16.

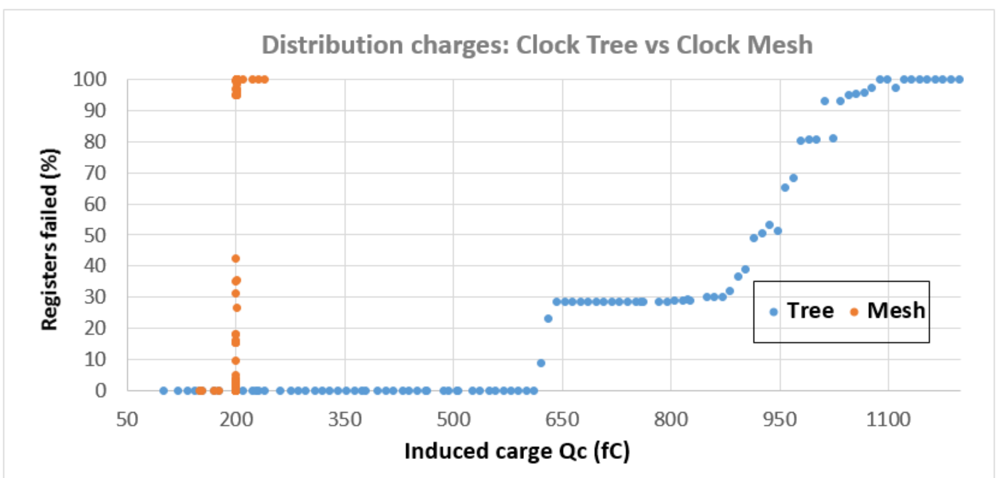


Figure 5.16 Comparison between clock tree and clock mesh

Clock tree has a completely different profile, clock tree starts fails at 620 fC with 9% of fail registers and increase step by step up to 100% of fails with 1,100 fC. This means that in an eventual impact of heavy ion, in these circuit, over the first buffer of the clock network, the circuit with clock mesh will fail before the circuit with clock tree. The immediate question is why the clock mesh fail first? Looking inside, the structure of the mesh has a little clock tree (top level chain and global mesh without the grid) that ends at 16 final-driver buffers with the same properties in every cross corner of the grid at Figure 5.14. Every output of these buffers joint at the mesh as part of clock mesh distribution. Then, the clock signal is radially distributed from the center of the chip die to the periphery of the IC. From the mesh, the signal is distributed to all registers. The mesh, which was usually fairly uniform over the chip, contributes to a grid wire capacitance that is uniform over the chip, but the clock density from the local clock blocks is often larger, and is remarkable non-uniform. On the other hand, clock trees have low latency, low power, minimal wiring track usage, and the potential for very low skew.

The mesh has the property to reduce local skew by connecting nearby points directly. Even with large process variations that cause significant skew across the chip, the resulting local skew is relatively small. Then, low interconnect resistance and high capacitance associated with the mesh wire structure allows a perfect propagation of the clock signal but at the same time these features can be turned a great disadvantage when a transient pulse takes this way, as shown in our results.

For aerospace application, clock tree topology result more appropriated in front clock mesh topology. Even if both of them have hardened registers. Clock mesh presents more power consumption and the layout circuit will be density by the redundancy of wires. Considering hardened register such as mentioned in chapter 3.5 to be replaced by the most vulnerable registers can improve the fault tolerance of the circuit. Probably to futures works we can quantify the improvement of the tolerance with numbers.

6 CONCLUSION

With the technology scaling, circuit designs become more complex that previous version. The need for synchronization of each block or stage is priority, clock distribution network takes a relevance when the clock signal must to be delivered. Especially when integrated circuit works in aerospace, because radiation may generate single event effects over the clock network. The effect may be translated in a clock pulse and reaches register elements storing incorrect data. There are techniques to analyze and improve the design against radiation but there is a lack of techniques to analyze the susceptibility of SET focused in clock distribution networks.

This work has investigated the SET effects in clock distribution network developing a methodology to evaluate any circuit before being manufactured. Using our method, it is possible to extract the clock network from any ASIC layout design and calculate the sensitivity of the circuit in terms of charge at different points and different ways. To extract the clock network, we developed the EXT-CLK tool in Bash command line script. A case study to validate a new methodology was the SRAM arbiter as part of Gigabit Ethernet circuit. In chapter 5, we show a variety of analysis over the clock network that help us to identify the more sensitive nodes or components of the circuit. One of these simulations explores the possibility of changing smaller buffers by higher improving the circuit against SET effects.

In section 5.6 we investigate the soft error rate of the SRAM arbiter due to SET in the clock tree. The result reveals that 4.6% of the registers of the circuit are very susceptible to generate an error in functional behavior. We have found that 17 registers are the most susceptible in the SRAM arbiter.

EXT-CLK tool also extracts the clock mesh designs, which allows to make a comparison between clock tree and clock mesh to see which one is more sensitive to SET effects. Section 5.7 describes the comparison of these types of clock distribution and explains some issues of the result and shows that clock mesh is more susceptible than clock tree to SET effects.

Designers clearly require accurate estimate of circuit error rates to make appropriate cost/reliability trade-offs. This work can help for generating these estimates and finally choose the best mitigation technique as hardware redundancy or use hardness-by-design technique over sensitive nodes before the manufacturing process and avoid spend of money.

REFERENCES

ACTEL. One Chip is All You Need. Single-Event Effects in FPGAs. [S.l.], 2007, Disponível em: <http://www.actel.com/documents/FirmErrorPIB.pdf >. Acesso em: jul. 2010.

AUTRAN, J. L.; CHABRERIE, C.; PAILLET, P.; FLAMENT, O.; LERAY, J.L. BOUDENOT, J.C. Radiation-induced interface traps in hardened MOS transistors: an improved charge-pumping study. **Nuclear Science, IEEE Transactions on**, [S.l.], set. 1996, v. 43, n. 6, p. 2547-2557.

BALOCH, S.; ARSLAN, T.; STOICA, A. Design of a single event upset (SEU) mitigation technique for programmable devices. In: QUALITY ELECTRONIC DESIGN, 7TH INTERNATIONAL SYMPOSIUM ON, **Proceedings…** San Jose, CA. [s.n.], mar. 2006. p. 342-345.

BARNABY, H.J.; MCLAIN, M.L.; ESQUEDA, I.S.; CHEN, X.J. Modeling ionizing radiation effects in solid state materials and CMOS devices. In: CUSTOM INTEGRATED CIRCUITS CONFERENCE. **Proceedings...** [S.l.:s.n.], 2008, p. 273-280.

BATTEZZATI, N.; DECUZZI, F.; VIOLANTE, M.; BRIET, M. Application-oriented SEU sensitiveness analysis of Atmel rad-hard FPGAs. In: ON-LINE TESTING SYMPOSIUM, 2009. IOLTS 2009. **Proceedings...** [S.l.:s.n.], 2009, p.89-94.

BOUDENOT, J. C. Radiation Space Environment. In: VELAZCO, R.; FOUILLAT, P.; REIS R. **Radiation Effects on Embedded Systems**. 1 ed. The Netherlands: Springer, 2007, p. 269.

CADENCE. LEF/DEF 5.4 Language Reference. [S.l.], 2003. Disponível em: < ftp://ftp.sitsemi.ru/pub/Cadence/lefdefref.pdf >. Acesso em: 28 mar. 2014.

CHELLAPPA, S.; CLARK, L. T.; HOLBERT, K. E. A 90-nm radiation hardened clock spine. In: RADIATION AND ITS EFFECTS ON COMPONENTS AND SYSTEMS (RADECS), 12TH EUROPEAN CONFERENCE ON, **Proceedings...** Sevilla, [s.n.], 19-23 sep. 2011, p. 479-484.

DASH, R.; GARG, R.; KHATRI, S.P.; GWAN CHOI. SEU hardened clock regeneration circuits. In: QUALITY OF ELECTRONIC DESIGN. ISQED 2009. QUALITY

ELECTRONIC DESIGN, **Proceedings...** San Jose, CA. [s.n], 16-18 mar. 2009. p. 806-813.

DODD, P. E.; MASSENGILL, L. W. Basic mechanisms and modeling of single-event upset in digital microelectronics. **Nuclear Science, IEEE Transactions on**, [S.l.], jun. 2003, v. 50, n. 3, p. 583-602.

DODD, P.E.; SHANEYFELT, M.R.; FELIX, J.A.; SCHWANK, J.R. Production and propagation of single-event transients in high-speed digital logic ICs. **Nuclear Science, IEEE Transactions on**, [S.l.], 2004, v. 51, n. 6, p. 3278-3284.

FERLET-CAVROIS, V.; PAILLET, P.; MCMORROW, D.; FEL, N.; BAGGIO, J.; GIRARD, S.; DUHAMEL, O.; MELINGER, J.S.; GAILLARDIN, M.; SCHWANK, J.R.; DODD, P.E.; SHANEYFELT, M.R.; FELIX, J.A. New Insights Into Single Event Transient Propagation in Chains of Inverters-Evidence for Propagation-Induced Pulse Broadening. **Nuclear Science, IEEE Transactions on,** [S.l.], dic. 2007, v. 54, n. 6, p. 2338-2346.

FRIEDMAN, E. G. Clock distribution networks in synchronous digital integrated circuits. In: **Proceedings of the IEEE**, [S.l.], mai. 2001, v. 89, n. 5, p. 665-692.

GARBER, S. Explorer-I and Jupiter-C. NASA History page, [S.l.], oct. 2007. Disponível em: < http://history.nasa.gov/sputnik/expinfo.html >. Acesso em: 5 feb. 2013.

GARG, R.; KHATRI, S. P. A novel, highly SEU tolerant digital circuit design approach. In: COMPUTER DESIGN, 2008. ICCD 2008. IEEE INTERNATIONAL CONFERENCE ON, **Proceedings...** [S.l.:s.n.], 12-15 oct. 2008, p. 14-20.

HAJ ALI ASGARI, F.; SACHDEV, M. A low-power reduced swing global clocking methodology. **Very Large Scale Integration (VLSI) Systems, IEEE Transactions on**, [S.l.], mai. 2004, v. 12, n. 5, p. 538-545.

HIROTSUGU, K.; SATOSHI, T.; KATSURO, S. Half-Swing Clocking Scheme for 75% Power Saving in Clocking Circuitry. In: VLSI CIRCUITS, 1994. DIGEST OF TECHNICAL PAPERS. 1994 SYMPOSIUM ON, **Proceedings...** [S.l.:s.n.], 9-11 jun. 1994, p. 23-24.

JUHNKE, T.; KLAR, H. Calculation of the soft error rate of submicron CMOS logic circuits. **Solid-State Circuits, IEEE Journal of**, [S.l.], jul. 1995, v. 30, n. 7, p. 830-834.

KOBAYASHI, H.; SHIRAISHI, K.; TSUCHIYA, H.; MOTOYOSHI, MAKOTO ; USUKI, H.; NAGAI, Y.; TAKAHISA, KEIJI ; YOSHIIE, TOSHIMASA ; SAKURAI, Y.; ISHIZAKI, T. Soft errors in SRAM devices induced by high energy neutrons, thermal neutrons and alpha particles. In: ELECTRON DEVICES MEETING, 2002. IEDM '02. INTERNATIONAL, **Proceedings...** [S.l.:s.n.], 8-11 Dec. 2002. p. 337-340.

LACOE, R. C. Improving Integrated Circuit Performance Through the Application of Hardness-by-Design Methodology. **Nuclear Science, IEEE Transactions on**, [S.l.], 2008, v. 55, n. 4, p. 1903-1925.

LOVELESS, T.D.; KAUPPILA, J.S.; JAGANNATHAN, S.; BALL, D.R.; ROWE, J.D.; GASPARD, N.J.; ATKINSON, N.M.; BLAINE, R.W.; REECE, T.R.; AHLBIN, J.R.; HAEFFNER, T.D.; ALLES, M.L.; HOLMAN, W.T.; BHUVA, B.L.; MASSENGILL, L.W. On-Chip Measurement of Single-Event Transients in a 45 nm Silicon-on-Insulator Technology. **Nuclear Science, IEEE Transactions on,** [S.l.], dic. 2012, v. 59, n. 6, p. 2748-2755.

MALLAJOSYULA, A.; ZARKESH-HA, P. A Robust Single Event Upset Hardened Clock Distribution Network. In: INTEGRATED RELIABILITY WORKSHOP FINAL REPORT, 2008. IRW 2008. IEEE INTERNATIONAL, **Proceedings...** [S.l.:s.n.], 12-16 oct. 2008, p. 121-124.

MING, Z.; SHANBHAG, N. R. A CMOS design style for logic circuit hardening. In: RELIABILITY PHYSICS SYMPOSIUM, 2005. 43RD ANNUAL. 2005 IEEE INTERNATIONAL, **Proceedings...** [S.l.:s.n.], 17-21 abr. 2005, p. 223-229.

MINSIK, C.; PAN, D. Z.; PURI, R. Novel binary linear programming for high performance clock mesh synthesis. In: COMPUTER-AIDED DESIGN (ICCAD), 2010 IEEE/ACM INTERNATIONAL CONFERENCE ON, **Proceedings...** [S.l.:s.n.], 7-11 nov. 2010, p. 438-443.

MOGOLLON, J.M.; PALOMO, F.R.; AGUIRRE, M.A.; NAPOLES, J.; GUZMAN-MIRANDA, H.; GARCIA-SANCHEZ, E. TCAD simulations on CMOS propagation induced pulse broadening effect: Dependence analysis on the NMOS VT. In: RADIATION AND ITS EFFECTS ON COMPONENTS AND SYSTEMS (RADECS), 2009 EUROPEAN CONFERENCE ON, **Proceedings...** Bruges, [s.n.], 14-18 set. 2009, p. 574-579.

MUNTEANU, D.; AUTRAN, J. L. Modeling and Simulation of Single-Event Effects in Digital Devices and ICs. **Nuclear Science, IEEE Transactions on**, [S.l.], 2008, v. 55, n. 4, p. 1854-1878.

NARASIMHAM, B.; GADLAGE, M.J.; BHUVA, B.L.; SCHRIMPF, R.D.; MASSENGILL, L.W.; HOLMAN, W.T.; WITULSKI, A.F.; GALLOWAY, K.F. Test Circuit for Measuring Pulse Widths of Single-Event Transients Causing Soft Errors. **Semiconductor Manufacturing, IEEE Transactions on**, [S.l.], 2009, v. 22, n. 1, p. 119-125.

NICOLAIDIS, M. A low-cost single-event latchup mitigation scheme. In: ON-LINE TESTING SYMPOSIUM, 2006. IOLTS 2006. 12TH IEEE INTERNATIONAL, **Proceedings...** Lake Como, [s.n.], 2006.

NIMO. Nanoscale Integration and Modeling (NIMO) Group. Predictive Technology Model 2012. ARISONA STATE UNIVERSITY. Disponível em: < http://ptm.asu.edu/ >. Acesso em: 5 set. 2014.

RABAEY, J. M.; CHANDRAKASAN, A.; NIKOLIC, B. **DIGITAL INTEGRATED CIRCUITS**, A DESIGN PERSPECTIVE. 2nd ed. [S.l.], Prentice Hall, 2003.

RAJARAM, A.; PAN, D. Z. MeshWorks: A Comprehensive Framework for Optimized Clock Mesh Network Synthesis. **Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on**, [S.l.], dic. 2010, v. 29, n. 12, p. 1945-1958.

RAMANATHAN, P.; DUPONT, A. J.; SHIN, K. G. Clock distribution in general VLSI circuits. **Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on**, [S.l.], mai. 1994, v. 41, n. 5, p. 395-404.

RESTLE, P.J.; MCNAMARA, T.G.; WEBBER, D.A.; CAMPORESE, P.J.; ENG, K.F.; JENKINS, K.A.; ALLEN, D.H.; ROHN, M.J.; QUARANTA, M.P.; BOERSTLER, D.W.; ALPERT, C.J.; CARTER, C.A.; BAILEY, R.N.; PETROVIC, J.G.; KRAUTER, B.L.; MCCREDIE, B.D. A clock distribution network for microprocessors. In: VLSI CIRCUITS, 2000. DIGEST OF TECHNICAL PAPERS. 2000 SYMPOSIUM ON, **Proceedings…** Honolulu, [s.n.], 2000. p. 184-187.

SCHRIMPF, R. D. Radiation Space Environment. In: VELAZCO, R.; FOUILLAT, P.; REIS R. 1 ed. **Radiation Effects on Embedded Systems**. The Netherlands: Springer, 2007. p. 12.

SCHRIMPF, R. D.; FLEETWOOD, D. M. Radiation effects and soft error in integrated circuits and electronic devices. **Selected topics in electronics and systems**. Tuck Link, World scientific Publishing Co. 2004, p. 348.

SEIFERT, N.; SHIPLEY, P.; PANT, M.D.; AMBROSE, V.; GILL, B. Radiation-induced clock jitter and race. In: RELIABILITY PHYSICS SYMPOSIUM. 43RD ANNUAL. 2005 IEEE INTERNATIONAL, **Proceedings**... [S.l.:s.n.], 17-21 abr. 2005. p. 215-222.

SHIVAKUMAR, P.; KISTLER, M.; KECKLER, S.W.; BURGER, D.; ALVISI, L. Modeling the effect of technology trends on the soft error rate of combinational logic. In: DEPENDABLE SYSTEMS AND NETWORKS. DSN 2002. INTERNATIONAL CONFERENCE ON, **Proceedings...** [S.l.:s.n.], 2002. p. 389-398.

STERPONE, L.; BATTEZZATI, N.; KASTENSMIDT, F.L.; CHIPANA, R. An Analytical Model of the Propagation Induced Pulse Broadening (PIPB) Effects on Single Event Transient in Flash-Based FPGAs. **Nuclear Science, IEEE Transactions on**, [S.l], 2011, v. 58, n. 5, p. 2333-2340.

TONFAT, J. Projeto, **Verificação Funcional e Síntese de Módulos Funcionais para um Comutador Gigabit Ethernet**. 2011. 128 f. Dissertação (Mestrado em Microeletrônica) – Instituto de Informática, Instituto de Informática, Universidade Federal do Rio Grande do Sul. Porto Alegre, 2011.

VARGAS, F.; NICOLAIDIS, M. SEU-tolerant SRAM design based on current monitoring. In: FAULT-TOLERANT COMPUTING, 1994. FTCS-24. DIGEST OF PAPERS, 24TH INTERNATIONAL SYMPOSIUM ON, **Proceedings...** [S.l.:s.n.], 15-17 jun. 1994, p. 106-115.

WILKE, G. R. **Analysis and optimization of mesh-based Clock Distribution Architectures**. 2008. 155 f. Tese (Doutorado em Ciencia da Computação) - Unniversidade Federal do Rio Grande do Sul, Instituto de Informatica, Programa de Pos-Graduacao em Computacao, Porto Alegre, 2008.

WIRTH, G.; KASTENSMIDT, F. L.; RIBEIRO, I. Single Event Transients in Logic Circuits-Load and Propagation Induced Pulse Broadening. **Nuclear Science, IEEE Transactions on**, [S.l], 2008, v. 55, n. 6, p. 2928-2935.

WIRTH, G. I.; VIEIRA, M. G.; KASTENSMIDT, F. G. L. Accurate and computer efficient modelling of single event transients in CMOS circuits. Circuits, **Devices & Systems, IET**, [S.l.], 2007, v. 1, n. 2, p. 137-142.

WISSEL, L.; HEIDEL, D.F.; GORDON, M.S.; RODBELL, K.P.; STAWIASZ, K.; CANNON, E.H. Flip-Flop Upsets From Single-Event-Transients in 65 nm Clock Circuits. **Nuclear Science, IEEE Transactions on**, [S.l.], 2009, v. 56, n. 6, p. 3145-3151.

APÊNDICE A:

ANÁLISE DE SINGLE EVENT EFFECT EM REDES DE RELÓGIO

A.1 Introdução

Podemos dizer que a indústria dos semicondutores chegou rapidamente a ser a maior indústria do mundo nos últimos tempos. Isso é porque os circuitos integrados (IC) formam parte de quase todos os equipamentos eletrônicos e estes formam parte da sociedade moderna.

No início os IC's tinham várias portas lógicas, mas com o tempo estas foram incrementando, chegando a desenhos complexos. A redução dos tamanhos dos transistores e desenhos cada vez mais complexos tem permitido a criação de System –on-chip (SoC) e Applications Specific Integrated Circuit (ASIC) que trabalham em altas frequências e em alguns casos até com frequências diferentes.

Com circuitos maiores e complexos surge a necessidade de sincronizar os blocos multiplex do circuito. As redes de relógio nos circuitos integrados tem como objetivo distribuir o sinal de relógio de maneira integra e com o mínimo atraso possível, a todos os dispositivos que o requeiram. A rede de relógio dá suporte ao sinal usado para o fluxo de dados e de controle. Se alguma falha acontece na rede de relógio, esta falha poderia se traduzir em um erro do circuito. Um erro no circuito seria fatal especialmente em circuitos dedicados ao trabalho espacial como um satélite ou outro tipo de equipamento espacial.

Prótons, nêutrons, elétrons e íons pesados provenientes da radiação espacial são responsáveis pela interferência e mal funcionamento dos transistores em satélites e outros dispositivos espaciais. Os efeitos foram classificados como Single Event Effects (SEE) (Barnaby *et al.*, 2008). Se o efeito ocorre dentro de uma célula de memória, é chamado de Single Event Upset (SEU), se ocorre num circuito combinacional então é chamado de Single Event Transient (SET). A rede de relógio também é vulnerável a efeitos de radiação desde que um SET ocorra na saída de um dos buffers, no transistor em off (Chellappa *et al.*, 2011). O pulso transiente pode-se propagar através da rede de relógio causando comportamentos indesejados nos registros.

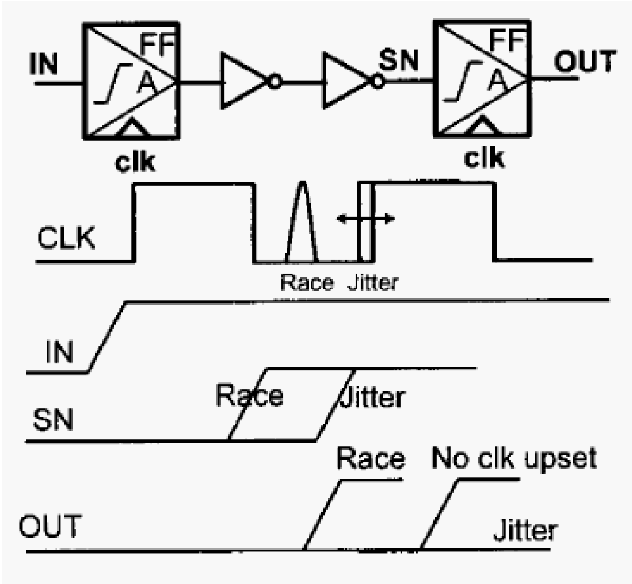


Figura A.1 Pulso transiente e jitter na rede de relógio (Seifert et al., 2005).

A.1.1 Motivação

A necessidade de sincronizar os blocos de circuitos cada vez mais complexos e a redução da tecnologia dos semicondutores, faz com que as redes de relógio tomem relevância, especialmente quando os circuitos são dedicados ao trabalho aeroespacial. Na literatura encontramos técnicas para analisar a sensibilidade dos circuitos em geral, mas não existe uma técnica focada somente nas redes de relógio.

Com o intuito de desenvolver uma técnica contra efeitos da radiação, é necessário primeiro desenvolver uma análise eficiente e exata de sensibilidade de SET nas redes de relógio.

Mas se torna difícil a realização de uma análise de sensibilidade de SET na rede de relógio quando existem muitos circuitos diferentes com redes diferentes. Então nasce a proposta de elaborar uma ferramenta capaz de extrair a rede de relógio do circuito a ser analisado.

As principais contribuições deste trabalho são:

- Desenvolvimento da ferramenta EXT-CLK, o qual extrai a rede de relógio dos arquivos de leiaute do circuito e gera como saída o modelo da rede em hspice netlist. A ferramenta extrai três tipos de topologia; clock tree, clock mesh e topologia H. A extração da rede de relógio permite realizar diferentes tipos de simulações de sensibilidade SET nos circuitos.
- Simulação de injeção de SET em cada nó da rede de relógio. Nesta simulação podemos identificar os nós mais sensíveis em termos de carga crítica (Q_c).
- Simulação de injeção de SET num caminho específico desde a raiz até algum registro. Esta simulação permite gerar um perfil do caminho e mostra a relação entre a carga necessária para gerar a falha e o tamanho da célula usada no caminho.

- Simulação de injeção de SET na raiz da rede de relógio. Com esta simulação podemos saber a sensibilidade de cada registro em função da carga injetada.
- Experimentar uma maneira simples e arriscada de melhorar a sensibilidade da rede de relógio, trocando os buffers mais pequenos por alguns maiores.
- Calcular o soft error rate da rede de relógio. Combinando dois tipos de simulações, a nível elétrico e a nível logico, obtendo o soft error rate de cada registro da rede de relógio.
- Realizar a comparação de sensibilidade entre duas topologias, rede de arvore e rede de tipo malha.

A.2 Efeitos da radiação em componenes semicondutores

A radioatividade no espaço representa uma ameaça para os componentes semicondutores. A fonte da radioatividade pode ser classificada em quatro categorias: Cinturões de radiação, explosões solares, ventos solares e raios cósmicos. A grande quantidade de energia que experimentam os dispositivos eletrônicos em órbita requerem que tais componentes tenham circuitos redundantes para proteção contra radiação.

Nos anos 70s, partículas alfa emitidas pelo decaimento natural de urânio, tório e isótopos secundários presentes como impurezas no empacotamento dos chips se revelaram como a maior causa de soft error rate em memórias DRAMs (Dodd e Massengill, 2003). Durante o mesmo período foi demostrado que a ionização criada pela iteração de nêutrons cósmicos de alta energia com material dos dispositivos poderiam causar soft error, e em meados dos anos 90, foi estabelecido que radiação cósmica de alta energia foi o principal responsável dos erros nas memórias DRAMs.

As anomalias produzidas nos semicondutores no espaço são principalmente divididas em dois efeitos: Total Ionizing Dose (TID) e Single Event Effect (SEE).

A.2.1 Total Ionizing Dose

O fenômeno impacta principalmente na camada de isolante, onde pode-se capturar as cargas e finalmente degradar o comportamento elétrico. Em tecnologias antigas o efeito foi descrito como o acúmulo de energia depositada uniformemente. Esta afirmação é baseada no tamanho relativamente maior dos componentes e na energia depositada por partículas individuais ou fótons. Mas para tecnologias menores de 130 nm esta afirmação não é mais válida (Schrimpf, 2007). Como resultado do TID nos transistores, encontramos o deslocamento de voltagem threshold e correntes de fuga.

A Figura A.2 mostra uma forma simples de explicar os efeitos de TID nos semicondutores.

Criação de pares eletrons-lacuna → Alteração do V_{th} e da corrente de fuga → O desempenho eléctrico é degradado

Figura A.2 Efeito de ionização: efeito acumulativo

A.2.2 Mecanismo de TID sobre o gate

O mecanismo do TID pode ser descrito em quatro passos. No primeiro passo, a exposição à radiação cria pares de elétron-lacuna. Logo depois uma fração dos pares eléctron-lacuna se recombinam. O tempo que dura o processo de recombinação é muito curto, limitado pelo tempo que toma o elétron para ser removido do óxido da porta (tipicamente menor a 0.1 ps). A recombinação acontece devido ao campo elétrico da porta, inclusive sem ter voltagem aplicada na porta. O efeito do campo elétrico no processo de recombinação é separar os elétrons e as lacunas, o que resulta em menos recombinação.

Neste ponto do mecanismo, a única carga que resta no oxido de silício (SiO_2) são as lacunas, como é mostrado na Figura A.3 (c). O segundo passo é o deslocamento das lacunas remanescentes para a interface SiO_2/Si. O processo foi estudado e foi determinado que o transporte das lacunas ocorre principalmente pulando dentro do SiO_2. As lacunas transportadas são rapidamente dispersadas, durando algumas décadas depois da radiação do pulso.

No terceiro passo, uma fração de lacunas são capturadas na interface SiO_2/Si. As cargas capturadas no óxido de silício são positivamente carregados, e podem ser neutralizadas por tunneling dos elétrons desde o silício ou por emissão térmica de um elétron da banda de valência do óxido.

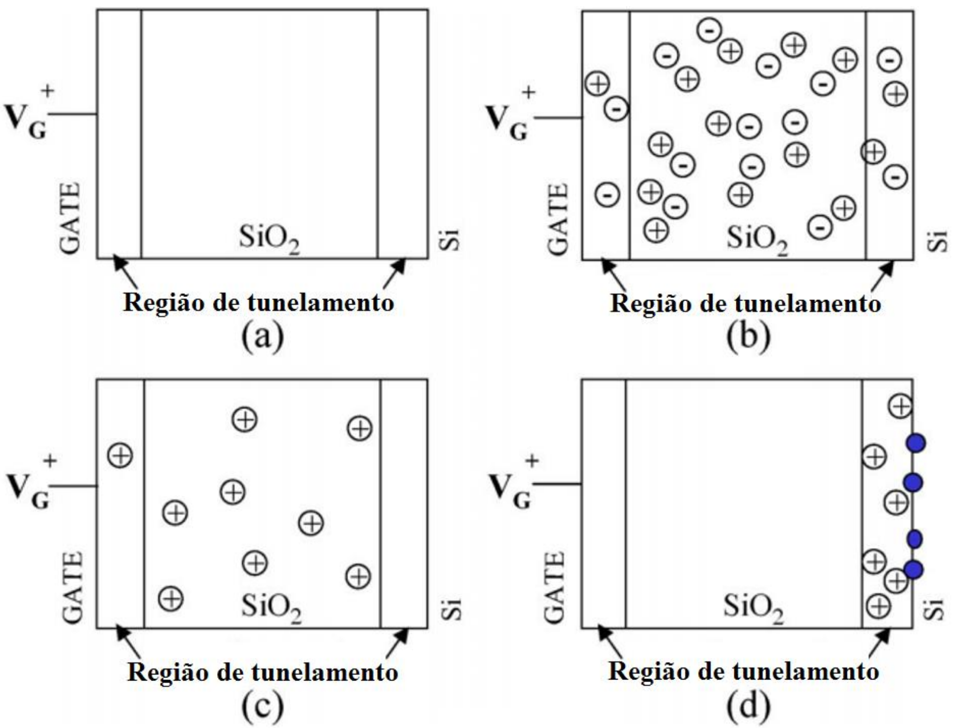


Figura A.3 Distribuição das cargas na porta do transistor depois da radiação.

No quarto passo do mecanismo depois da exposição de radiação, é possivel ocorrer a formação de “interstate traps”. O estado da interface é associado com ligações livres entre Si e SiO_2. Existem estados de interface dentro das bandas proibidas do silício na interface com SiO_2. Para transistores NMOS, o estado da interface pode atuar como cargas negativas no óxido, ou como cargas positivas para o transistor PMOS. Os quatro passos são mostrados na Figura A.3.

A.2.3 Deslocamento da voltagem umbral

O efeito de introduzir cargas no óxido de silício e/ou na interface SiO_2/Si é o deslocamento da voltagem umbral (ΔV_T) do transistor CMOS. A quantidade de carga gerada no óxido pela radiação é proporcional a largura de óxido. Adicionalmente, o efeito no ΔV_T é proporcional à distância da carga ao eletrodo da porta. O efeito pode ser descrito na Equação A.1.

$$\Delta V_T = -\frac{Q_{ot}}{C_{ox}} \propto x_{ox}^2 \quad \text{Equation A.1}$$

Então, para larguras de óxido pequenas, o efeito da ionização pela radiação é menos significativo que para antigas tecnologias com larguras maiores. O óxido nas portas com larguras finas, são inclusive menos sensíveis do que se estima, porque muitas das cargas dentro de 5 nm de interface são rapidamente deslocadas por tunneling (Schrimpf, 2007).

A.2.4 Corrente de fuga nas bordas

A borda onde acontece a corrente de fuga nos transistores comuns é localizada na interface do óxido na porta com óxido de campo espesso “thick field oxide (FOX)”. Os transistores são tipicamente isolados um ao outro pelo FOX. A Figura A.4 mostra as bordas onde acontece a fuga de corrente.

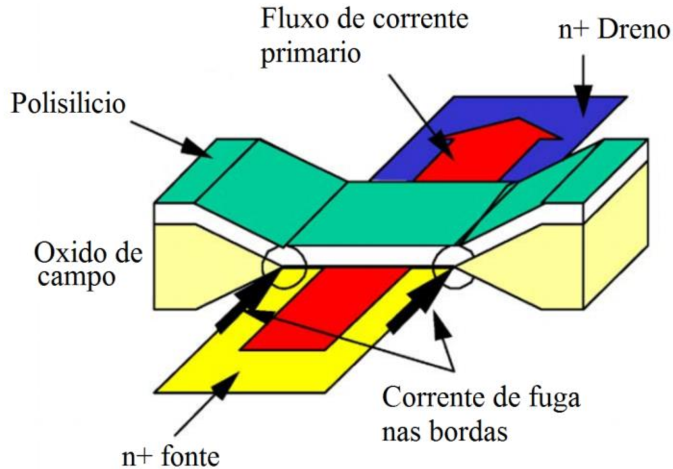


Figura A.4 Corrente de fuga nas bordas

Comparando duas tecnologias, em 0.35 µm, 1 nA de corrente de fuga é causado por 100krad, mas para 0.18 µm de tecnologia, 1 nA de corrente de fuga é causado por 350 krad. Como foi mencionado anteriormente, o dose de radiação afeta muito mais os semicondutores de tecnologias antigas do que as mais novas devido a área de cargas depositada.

A.2.5 Single event effects (SEE)

Single event effects é um fenômeno desfavorável, o que resulta da interação entre uma partícula de energia com a região sensível do circuito microeletrônico. A partícula pode ser próton, nêutron ou íon de alta energia (Munteanu e Autran, 2008). O fenômeno pode ser classificado em hard error e soft error. Hard error são aqueles que não podem ser recuperados (erro irreversível), mas o soft error pode ser restabelecido com um reset, comutando a fonte de energia ou reescrevendo a informação. Os fenômenos que nos focaremos são single event upset, single event transiente e single event latchup.

A.2.5.1 Single Event Upset (SEU)

Single Event Upset (SEU) é definido pela NASA como “um erro produzido pela radiação em circuitos microeletrônicos causado quando partículas carregadas perdem energia ionizando o meio por onde passa, deixando um rastro de pares de elétron-lacuna” (Baloch *et al.*, 2006). SEU é um erro não permanente, reescrevendo o dado ou um simples reset o dispositivo volta a funcionar normalmente. SEU pode acontecer em circuitos digitais quando o impacto de uma partícula provoca uma mudança de estado do dado armazenado em dispositivos como flip-flop, latch ou memórias SRAM. Ionização direta é o principal mecanismo de SEU causado por íons pesados com número atômico $Z \geq 2$. Partículas não pesadas como prótons e nêutrons, geralmente não tem carga suficiente para

ionizar o meio e gerar um SEU no circuito. A Figura A.5 mostra a diferença entre os íons ao produzir SEUs.

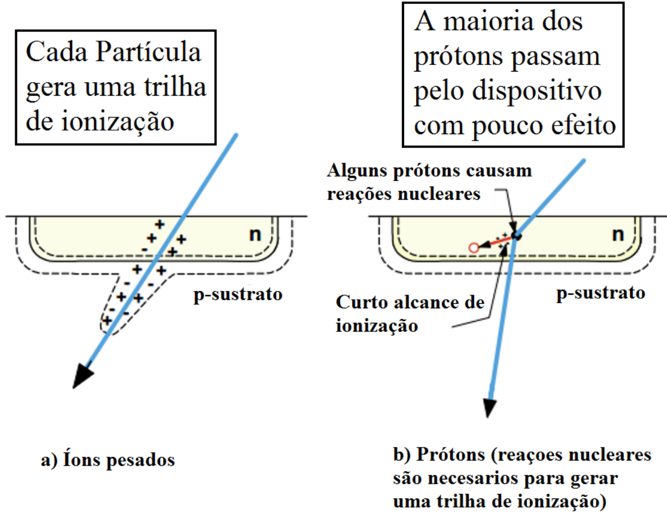


Figura A.5 Mecanismo de íons e prótons no semicondutor

A.2.5.2 Single Event Transient (SET)

Single Event Transient (SET) é um distúrbio na voltagem de algum sinal causado pelo impacto de uma partícula. O SET acontece em circuitos combinacionais, sinais de controle ou em redes de relógio, mas não acontece em circuitos de armazenamento (Loveless *et al.*, 2012). Circuitos combinacionais podem interpretar o SET como um estado lógico alto por um curto período e produzir um comportamento indesejado no circuito.

Com o avanço da tecnologia os circuitos combinacionais são mais sensíveis ao SET com menor largura de pulso. O pulso transiente não afetará o circuito se não é capturado por algum dispositivo de armazenamento. O pulso transiente no circuito pode não ser capturado na memória porque pode ser mascarado por algum dos seguintes fenômenos.

- **Mascaramento lógico**, acontece quando o SET é produzido em um sinal mas não é tomado em conta por ser irrelevante no cálculo seguinte. A figura A.6 mostra um exemplo do mascaramento lógico.

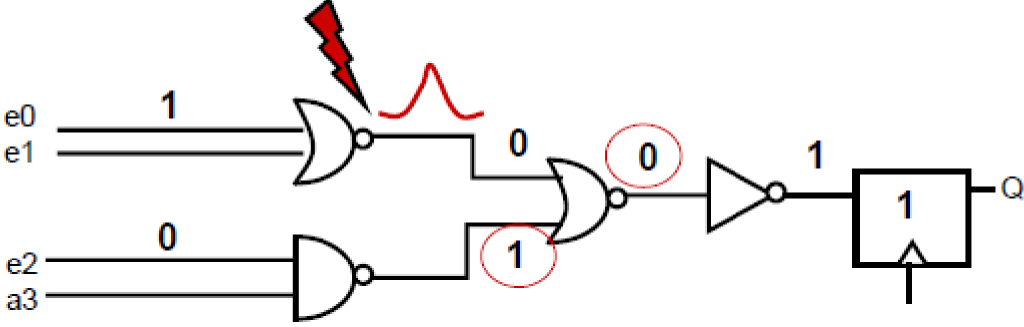


Figura A.6 Mascaramento logico

- **Mascaramento elétrico**, Acontece quando o pulso transiente é atenuado pelas caraterísticas elétricas das células, até o ponto que este não afeta o resultado do circuito. A Figura A.7 mostra um exemplo deste caso.

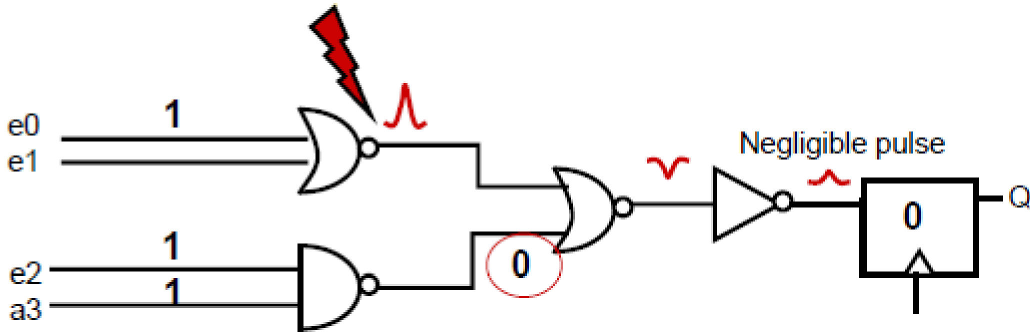


Figura A.7 Mascaramento elétrico

- **Mascaramento por Latchig-window**, se presenta quando o pulso gerado pelo impacto do da partícula alcança a célula mas não gera distúrbio porque o pulso do relógio não é o apropriado. Um exemplo deste caso é mostrado na Figura A.8.

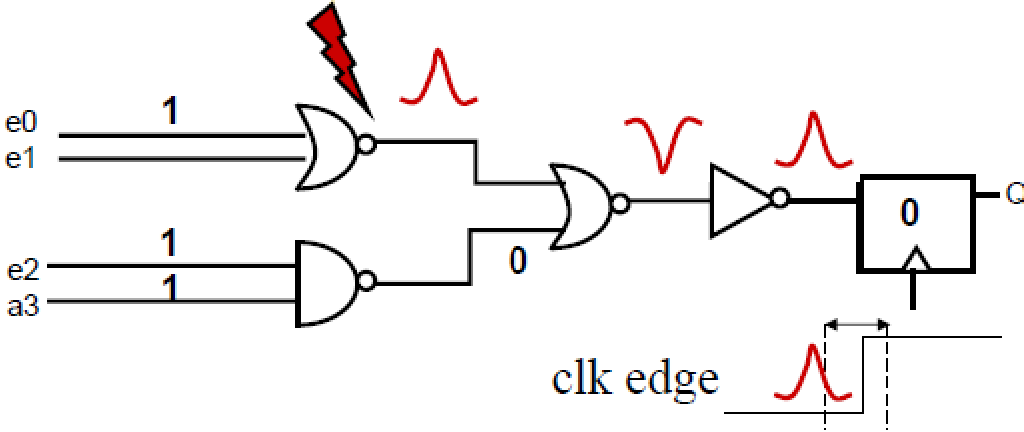


Figura A.8 Mascaramento por Latchig-window

A.2.5.3 Single event latchup (SEL)

Single event Latchup é um estado anormal de alto consumo de corrente nos dispositivos, o que resulta na perda do dispositivo. O fenômeno é causado pela interação de partícula energética através das regiões sensíveis. As regiões adjacentes n-type e p-type no circuito CMOS podem criar um tiristor parasita formado por um par de transistores bipolares parasitas. O SEL acontece quando a estrutura PNPN comuta de alta impedância a baixa impedância. A corrente espúria em um dos transistores pode ser amplificada pela realimentação positiva do tiristor, cuasando um curto circuito virtual entre V_{dd} e V_{ss} (Nicolaidis, 2006). A Figura A.9 representa a estrutura do transistor PNPN no circuito CMOS.

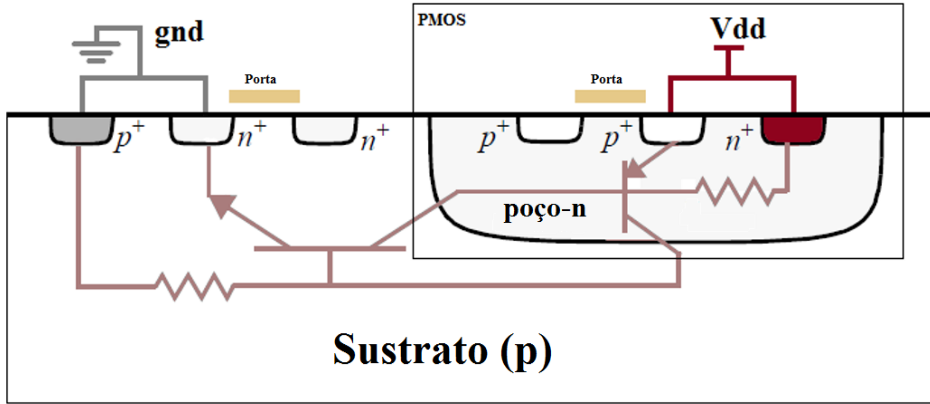


Figura A.9 Estrutura PNPN dentro do circuito CMOS

A.3 Redes de relógio em semicondutores

Idealmente, o sinal de relógio deveria chegar em todos os elementos no mesmo tempo, porém, o sinal chega com diferente atraso a cada dispositivo. A máxima diferença no atraso, chamado de “skew”, é porque o sinal passa por interconexões e outros dispositivos próprios da distribuição de relógio como buffers e clock gates que contribuem no atraso. Se o skew é maior do que o padrão certo, isso poderia causar erro no circuito (Ramanathan *et al.*, 1994).

Para lidar com esse problema, diferentes estruturas ou redes de relógio foram implementadas, como o caso das redes de árvore, redes de relógio H e a rede de malha (mesh). A seguir faremos um resumo de cada uma dessas redes de relógio.

A.3.1 Árvore de relógio

A distribuição em árvore é a estrutura mais usada como rede de relógio nos circuitos. A estratégia é baseada em inserir buffers em todos os caminhos da árvore, onde finalmente as folhas da árvore são os registros. Em muitos casos a distribuição em arvore é misturada com outros tipos de redes, dependendo das exigências dos desenhos. A figura A.10 mostra um conjunto de redes derivadas de uma única raiz.

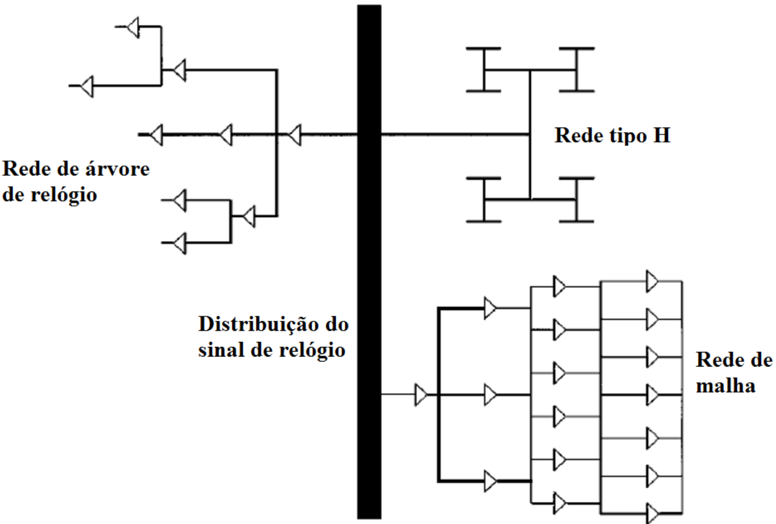
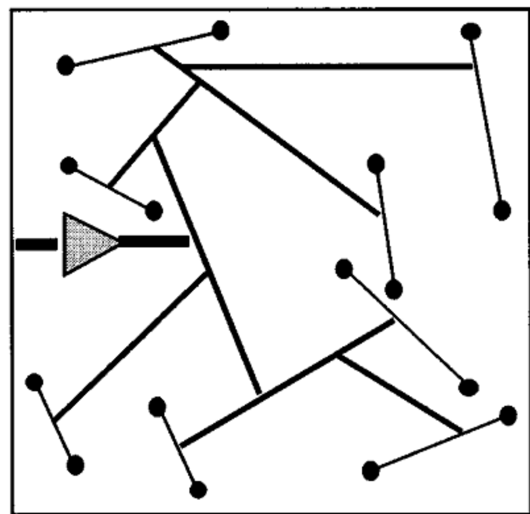


Figura A.10 Distribuição do sinal de relógio complexa (Friedman, 2001).

A área da distribuição é desenvolvida por um algoritmo que controla cuidadosamente as variações de atraso do sinal para minimizar o skew. A ideia é construir uma estrutura de árvore com os pinos de relógio dos registros. A estrutura é criada usando um algoritmo recursivo de baixo pra acima. Em cada pino de relógio é definida uma net, o ponto onde dois skew são zero e simétricos, e é selecionada como uma nova net. O processo continua até gerar o árvore de relógio, assim cada novo caminho é selecionado com skew zero. A Figura A.11 mostra a geometria do algoritmo.



A.11 Algoritmo de árvore de relógio para alcançar o skew zero

A.3.2 Rede de distribuição H

A topologia de distribuição H é uma arquitetura simétrica que assegura skew zero por preservar a distribuição uniforme. Geralmente os buffers usados são idênticos em toda a rede. Nesta distribuição o primeiro buffer é conectado no centro da estrutura H e o sinal é distribuído nos quatro quadrantes do H. Os quatro sinais idênticos são as entradas do seguinte nível da estrutura H. O processo de distribuição continua através de muitos níveis progressivamente mais pequenos. A Figura A.12 mostra a distribuição H. Finalmente a distribuição H termina nos registros ou são amplificados por buffers locais para conduzir outros registros. Com a rede H, cada caminho da estrutura tem praticamente o mesmo atraso. Porém, a principal causa do atraso neste tipo de rede é a variabilidade de processo. A quantidade de skew dentro da estrutura H é fortemente dependente do tamanho físico e como os registros são distribuídos na rede.

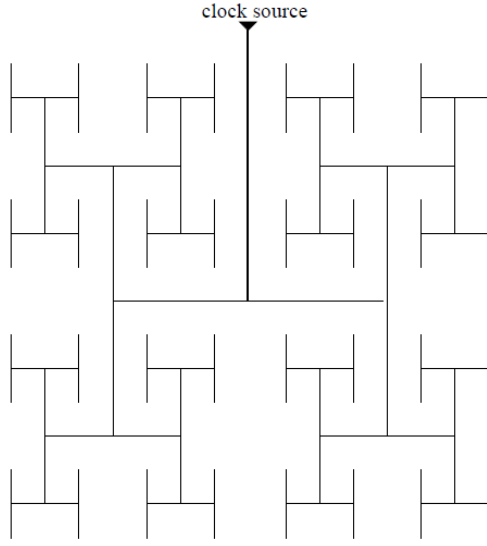


Figura A.12 Rede de distribuição H

A.3.3 Rede de relógio em malha (mesh)

A distribuição em malha permite skew zero no circuito. A estrutura tem uma distribuição de malha e os registros podem ser conectados diretamente ou redistribuir o sinal do relógio a uma rede local. A malha é usada tipicamente no estágio final da rede de relógio e é muito utilizada no desenho de processadores. A vantagem desta rede é a acessibilidade em vários pontos do leiaute para obter o sinal de relógio caso o desenho seja modificado. Usando o Encounter podemos criar a rede em malha, e basicamente podemos ter o top level, a malha, e a rede de árvore se for necessário. A Figura A.13 mostra uma possível rede em malha. Em comparação com as anteriores redes, a estrutura de malha consome muita potência devido a redundância de conexões (Minsik *et al.*, 2010).

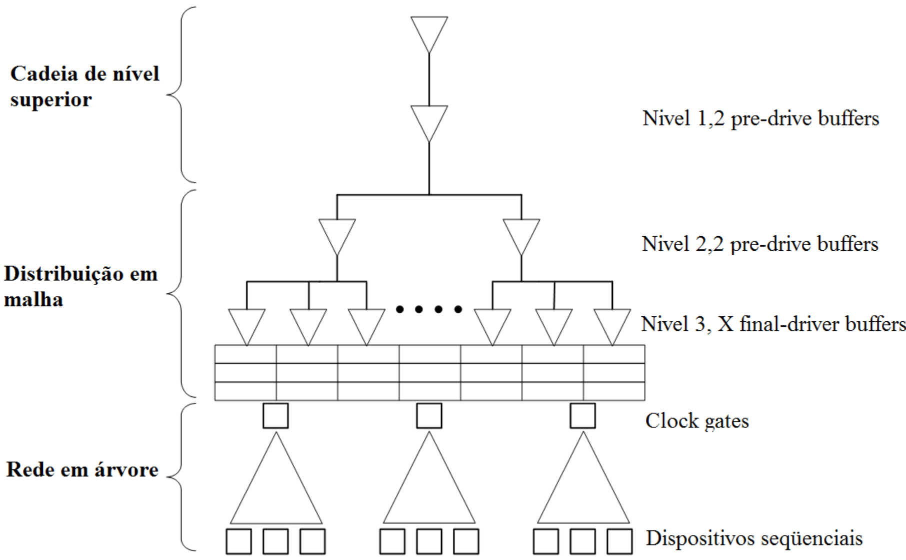


Figura A.13 Distribuição de malha

A.4 Metodologia proposta para a análise do SET em redes de relógio

Como foi mencionado nos capítulos anteriores, as redes de relógio também são afetadas pela radiação sem importar a fonte. Nesta seção apresentamos a metodologia para analisar a sensibilidade das redes de relógio em presença do SET. O objetivo é identificar os nós e os componentes mais sensíveis do circuito testado.

A.4.1 Metodologia proposta

Com o intuito de desenvolver uma técnica de mitigação contra os efeitos da radiação nas redes de relógio, é necessário desenvolver uma análise eficiente de sensibilidade nas redes de relógio. Torna-se difícil analisar as redes de relógio quando os circuitos são diversos e com diferente distribuição de relógio. A proposta analisa o circuito extraindo a rede de relógio dos arquivos de leiaute, usando EXT-CLK, sem importar o tipo de rede que tenha o circuito.

Com a metodologia proposta podemos fazer diferentes tipos de simulações. Simulações a nível lógico e a nível elétrico. As simulações a nível lógico permitem obter o *soft error rate* nas redes de árvore e com as simulações elétricas podemos fazer diversos tipos de simulações de injeções de SET no circuito extraído em ambiente hspice. A figura A.13 mostra a metodologia proposta.

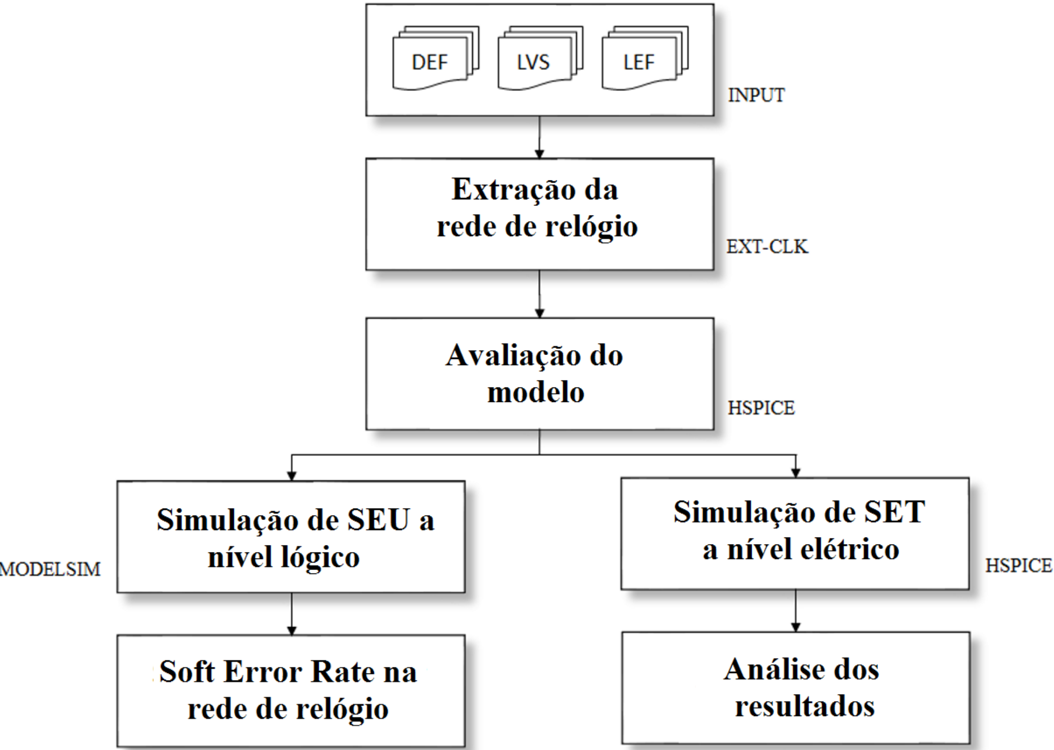


Figura A.13 Metodologia proposta

A metodologia é baseada na extração da rede de relógio dos arquivos de leiaute DEF, LEF e LVS. Para a extração da rede de relógio, foi desenvolvido para esta tese uma ferramenta chamada EXT-CLK que entrega a netlist da rede de relógio em modelo hspice.

A.4.2 Ferramenta EXT-CLK

A ferramenta EXT-CLK extrai as redes de tipo árvore, rede H e rede de tipo malha. A ferramenta foi desenvolvida em linha de comando Bash no ambiente Linux. A ferramenta não necessita nenhuma biblioteca adicional e também roda em ambiente OS MAC. EXT-CLK toma como entrada arquivos de leiaute do circuito (DEF, LEF, LVS) para extrair o netlist em hspice. A ferramenta inicia na saída do PLL ou onde inicia a distribuição do sinal de relógio, identificando os caminhos que pertencem a rede de relógio. Com os caminhos identificados, é mais fácil identificar os buffers (inversores e não inversores), clock gates e registros da rede. Conforme os elementos das redes são identificados e salvos, os caminhos do roteamento são modelados eletricamente usando o modelo pi mostrado na Figura A.14. Os cálculos de resistência e capacitância são realizados usando RPERSQ (resistência por área quadrada), CPERSQDIST (capacitância por área quadrada por distância) e EDGECPACITANCE (capacitância nos bordes) de cada leiaute de metal. Os valores são extraídos dos arquivos LEF do desenho.

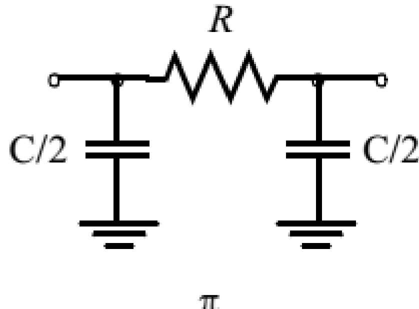


Figura A.14 Modelo pi das conexões de roteamento (Rabaey *et al.*, 1999)

Depois de extrair a rede de relógio para o modelo hspice, a ferramenta entrega um netlist com a descrição de cada net e células usadas. Adicionalmente, a ferramenta insere condições iniciais para melhorar o rendimento da simulação e pontos de medição para obter os resultados organizados e de fácil acesso. A Figura A.15 mostra o modelo do netlist de saída do EXT-CLK. Para mais detalhes sobre a ferramenta, ver capitulo 4.

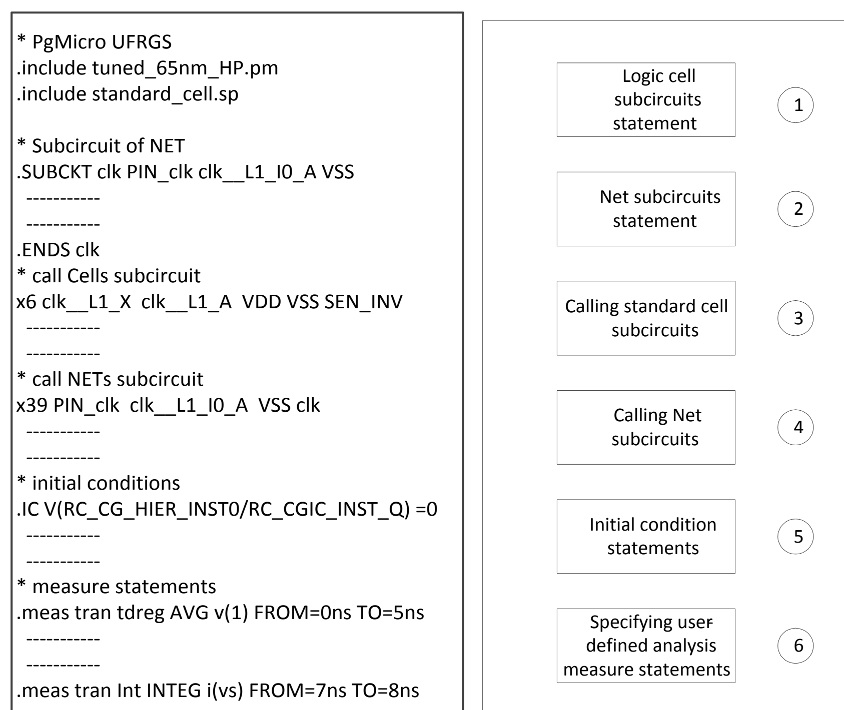


Figura A.15 Netlist de saída do EXT-CLK

A.5 Simulação de injeção de falhas

Considerando o circuito SRAM arbiter como estudo de caso, nós extraímos a rede de relógio de estrutura de árvore e fizemos a análise de sensibilidade de SET. Três simulações em nível elétrico foram realizadas inicialmente.

- Injeção de SET em cada nó
- Injeção de SET em um caminho
- Injeção de SET no início da rede

Outras simulações foram realizadas com distintos propósitos como:

- Substituição de buffers pequenos por maiores
- *Soft error rate* na árvore de relógio
- Comparação entre rede de árvore e rede de malha

Das simulações a nível elétrico só a injeção de SET na raiz é utilizada para o cálculo do *soft error rate*.

Todos os resultados das simulações são em termos de carga com unidades de Coulomb (C). O cálculo das cargas resulta da integral da corrente em relação ao tempo. Como na Equação A.1.

$$Q = \int_0^t I_{drain}(t)dt \qquad \text{Equação A.1}$$

A corrente usada nas simulações é modelada com o pulso de duplo exponencial, como é descrito em (Wirth *et al.*, 2007).

O estudo de caso é o circuito *SRAM arbiter* que foi desenhado em 65 nm usando o design kit da IBM no Encounter. O *SRAM arbiter* é a interface entre módulos lógicos do *Switch Gigabit Ethernet* com o *SRAM* externo *Zero-Bus Turn (ZBT)*, e foi desenhado com mínimo skew e mínimo atraso. A Figura A.16 mostra o leiaute do circuito.

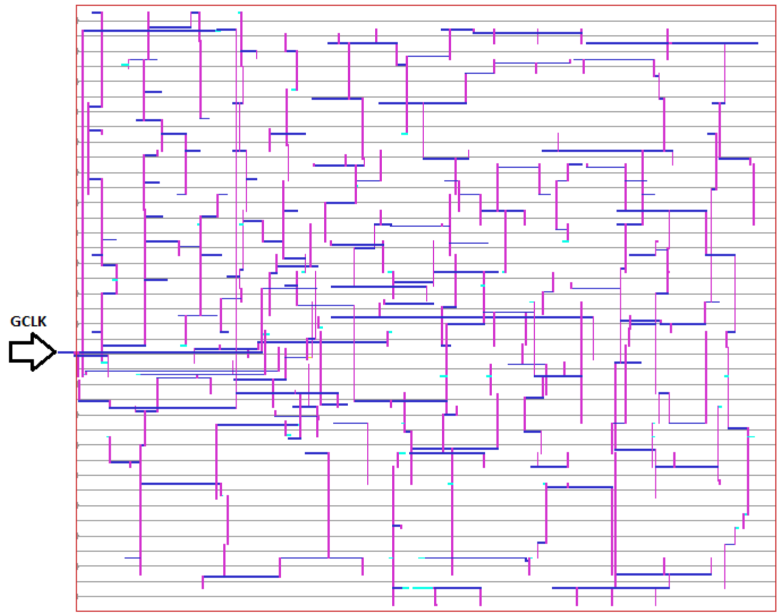


Figura A.16 Árvore de relógio do *SRAM arbiter*

A.5.1 Injeção de SET em cada nó

A simulação permite gerar um perfil do circuito em relação a sensibilidade dos nós da rede de relógio. O resultado da simulação na Figura A.17 mostra todos os nós da rede de relógio com a quantidade de carga necessária para gerar uma falha em um dos registros. A rede de relógio tem 367 registros separados em grupos. Em todas as simulações com hspice foi utilizado o modelo 65 nm do PTM. O pulso SET tem duração aproximada de 70 psec (o valor foi obtido experimentalmente em varias simulações para 65 nm). A amplitude da corrente é incrementada em um valor de poucos nano amperes até produzir falhas nos registros.

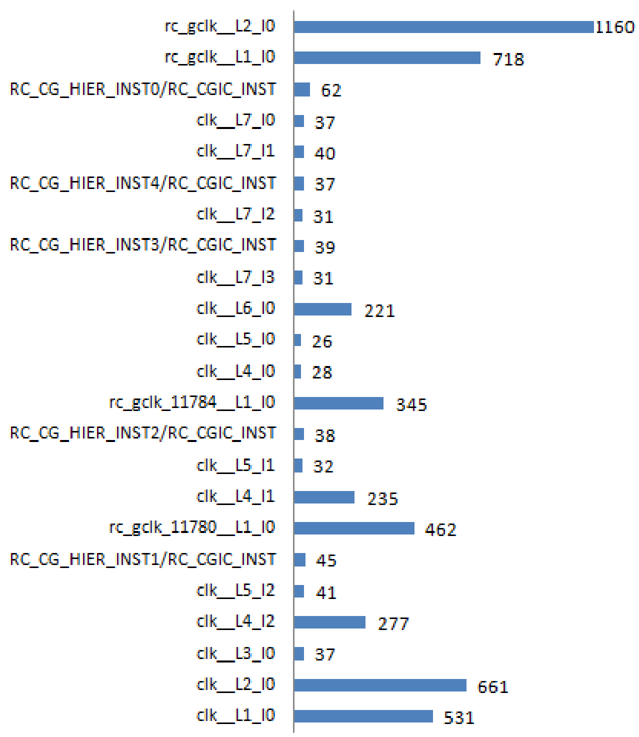


Figura A.17 Mínima carga (pC) injetada nos nós

A.5.2 Injeção de SET num caminho da rede

Nesta simulação podemos gerar o perfil de um caminho em particular e ver a relação entre a carga necessária para provocar uma falha e o tamanho da célula usada na rede de relógio. A figura A.18 mostra o caminho escolhido para simulação.

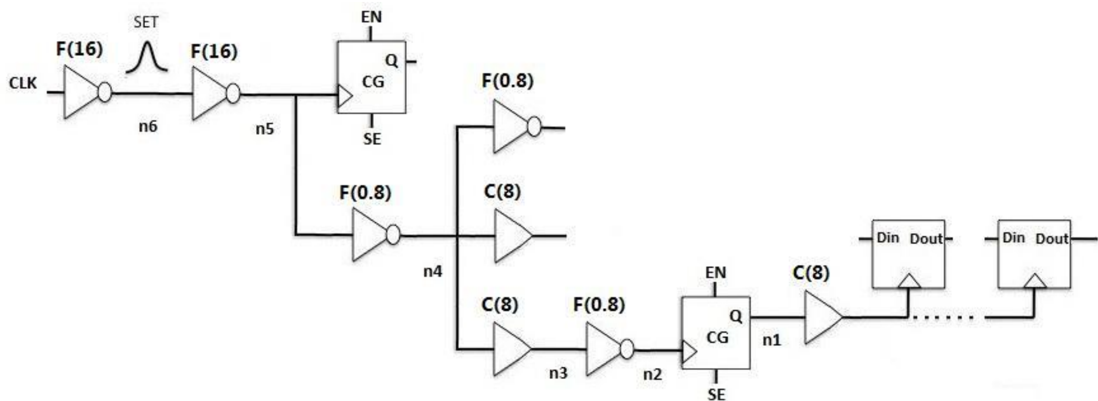


Figura A.18 Caminho da rede a ser analisado

Na simulação o pulso SET foi injetado na saída de cada buffer. O caminho selecionado tem três tipos de buffers, F(16), F(0.8), C(8) e o clock gate CG(1). O número dentro dos parêntesis representa o ganho do buffer. A Figura A.19 representa o perfil do caminho em termos de distribuição de carga.

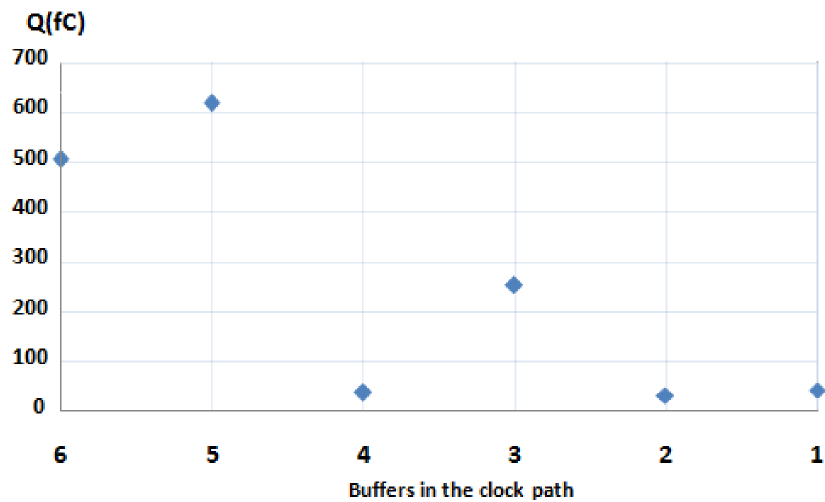


Figura A.19 Perfil da sensibilidade do caminho em termos de carga mínima

A.5.3 Substituição de buffers pequenos por maiores

Nesta simulação ousamos trocar os buffers menores por outros maiores para ver a diferença em relação a quantidade de carga necessária para produzir falhas. É muito provável que a troca de buffers incremente o skew do circuito e/ou aumente o atraso de algum caminho, mas é um risco para verificar a relação entre o tamanho dos buffers com a sensibilidade dos nós. A Figura A.20 mostra detalhes dos buffers antes e depois de serem trocados.

					Antes de sustituir a célula		Depois de sustituir a célula		
No	Componente	Células lógicas	Descrição	Pulso SET	Carga @ corrente	Reg. Fails	Células lógicas	Carga @ corrente	Reg. Fails
1	clk__L3_I0_X	SEN_INV_0P8	inverter	1 --> 0	41.7fC @390uA	105	SEN_INV_10	370fC @3.4mA	29
2	clk__L4_I0_X	SEN_INV_0P8	inverter	0 --> 1	48fC @450uA	14	SEN_INV_10	622fC @5.7mA	10
3	clk__L4_I1_X	SEN_BUF_S_8	buffer	1 --> 0	316fC @2.9mA	19	SEN_BUF_S_16	578fC @5.3mA	19
4	clk__L4_I2_X	SEN_BUF_S_8	buffer	1 --> 0	300fC @2.8mA	58	SEN_BUF_S_16	577fC @5.3mA	2
5	clk__L5_I0_X	SEN_INV_0P8	inverter	1 --> 0	31fC @290uA	3	SEN_INV_10	375fC @3.5mA	10
6	clk__L5_I1_X	SEN_INV_0P8	inverter	0 --> 1	45fC @420uA	19	SEN_INV_10	578fC @5.3mA	19
7	clk__L5_I2_X	SEN_INV_0P8	inverter	0 --> 1	59fC @550uA	72	SEN_INV_10	599fC @5.5mA	35
8	clk__L6_I0_X	SEN_BUF_D_8	buffer	1 --> 0	283fC @2.6mA	7	SEN_BUF_D_16	556fC @5.1mA	7
9	clk__L7_I0_X	SEN_INV_0P8	inverter	0 --> 1	47fC @440uA	3	SEN_INV_10	535fC @4.9mA	2
10	clk__L7_I1_X	SEN_INV_0P8	inverter	0 --> 1	48fC @450uA	4	SEN_INV_10	546fC @5 mA	4
11	clk__L7_I2_X	SEN_INV_0P8	inverter	0 --> 1	45fC @420uA	3	SEN_INV_10	579fC @5.3mA	3
12	clk__L7_I3_X	SEN_INV_0P8	inverter	0 --> 1	45fC @420uA	4	SEN_INV_10	579fC @5.3mA	4

Figura A.20 Detalhe dos buffers substituídos

Finalmente, em todos os casos dos buffers substituídos o resultado foi favorável. Na seguinte Figura A.21 podemos ver o resultado antes e depois de fazer a troca dos buffers pequenos por maiores. Por exemplo, antes de mudar o componente numero 1 na saida dele era necessário 41.7 fC para produzir uma falha em 105 registros. Depois de substituir o componente 1 por outro 10 vezes maior o nó ficou menos sensível e requer de 370 fC para produzir uma falha em 29 registros.

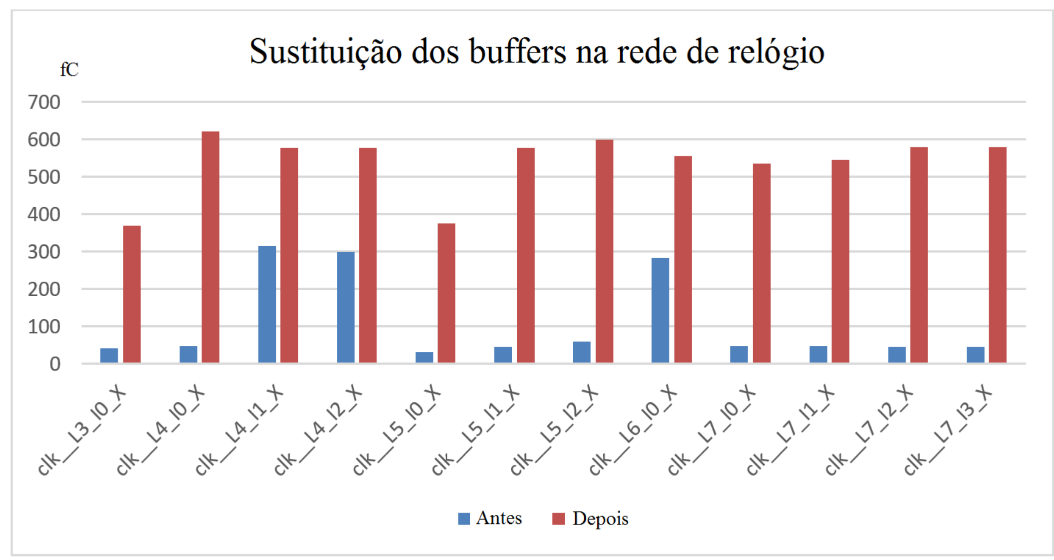


Figura A.21 Resultado da substituição dos buffers pequenos

Alguns componentes não tiveron melhoria significativa em relação a os outros. Os componentes 3, 4, e 8 foram reemplazados por outro 2 vezes maior, o que resulta em casi duas vezes a carga. Por exemplo, o componente 3 antes de ser substituido requer 316 fC para gerar falha em 19 registros. Depois de subsitutir o componente ele requer 578 fC para gerar o mesmo número de falhas. O mesmo resulta para os componentes 4 e 8.

A.5.4 Injeção de SET no início da rede

Através desta simulação sabemos a sensibilidade de cada registro em termos de porcentagem de falhas e também podemos criar o perfil do circuito em termos de cargas mínimas para produzir falhas. Lembremos que o circuito tem 367 registros, então para melhor visualização renomeamos os registros de registro_1 até registro_367. A Figura A.22 mostra o resultado da simulação de injeção de SET no início da rede em árvore.

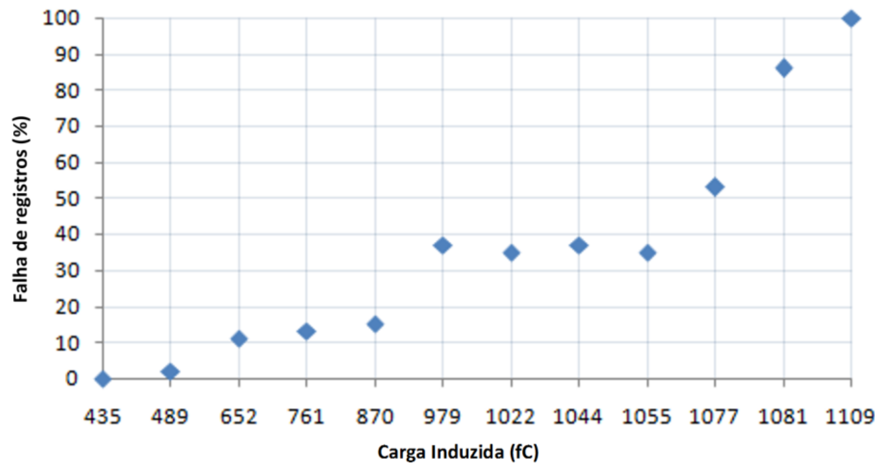


Figura A.22 Porcentagem de falha de registros devido ao SET no início da rede de relógio

A distribuição de carga na figura A.22 mostra alguns degraus. Por exemplo, de 870 fC para 979 fC o degrau revela que um novo segmento de rede de relógio foi atingido e provavelmente tem maior numero de registors que o segmento anterior.

Em outro ponto de vista, podemos ver a proporção de falha dos registros em 24 simulações feitas a nível elétrico. A Figura A.23 mostra o resultado desta simulação onde os register_[0,1,2] tem uma frequência de falhas significativa em 95% das simulações. Os register_[0,1,2] são os três primeiros registros de *counter_reg[3]*. O mesmo comportamento ocorre com o register_5, cujo representa o registro *do_reset_reg*. Os resultados desta simulação serão usados para calcular o soft error rate na seguinte seção.

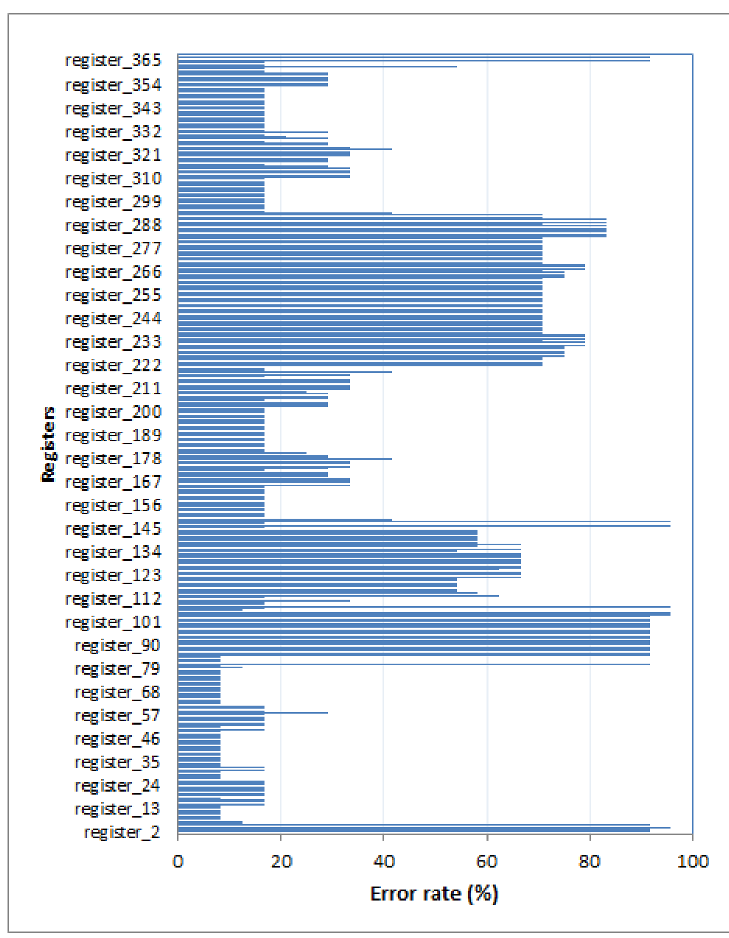


Figura A.23 Porcentagem de falhas de registros a nível elétrico

A.5.5 Soft error rate na árvore de relógio

Com o intuito de saber o *Soft Error Rate (SER)* devido ao SET na rede de relógio, é importante considerar dois tipos de falhas: a) *bit-flip* provocado por um SET na rede e b) falha na saída do circuito provocado por um *bit-flip*. Para o primer tipo de simulação usamos hspice (nível elétrico) e para o segundo tipo usamos o ModelSim (nível lógico). O cálculo de SER corresponde à Equação A.2.

$$SER_{SET\text{-}clock\text{-}tree} = ErrorRate(_{SET \rightarrow bit\text{-}flip}) \times ErrorRate(_{bit\text{-}flip \rightarrow circuit\text{-}falha}) \quad \text{Equação A.2}$$

Onde o ErrorRate($_{SET \rightarrow bit\text{-}flip}$) é um erro causado por SET na rede de relógio que pode gerar um *bit-flip* nos registros. A medida foi calculada na seção A.5.4.

O ErrorRate($_{bit\text{-}flip \rightarrow circuit\text{-}falha}$) é uma falha na saida do circuito como resultado de um *bit-flip*. Quando uma aplicação está rodando no circuito e ocorre um *bit-flip*, o efeito pode ser observado na saída como uma falha ou pode ser mascarado sem causar efeitos.

Na simulação do ErrorRate($_{bit\text{-}flip \rightarrow circuit\text{-}falha}$) só um bit-flip foi inserido por execução do testbench. Durante a execução o testbench, utiliza-se a memória RAM para armazenar dados. Então, realiza-se uma comparação entre duas situações, em operação normal e quando a injeção de falha é feita. A operação normal é chamada de "gold execution" e o resultado é armazenado para futuramente ser comparado. No caso da injeção de falhas, para simular o *bit-flip*, o valor armazenado no registro é alterado em qualquer momento por um período de 100 ps. Após cada simulação de injeção de falhas, o resultado é comparado com o resultado da "gold execution". Se alguma diferença ocorre, um erro é reportado. A comparação é feita para os 367 registros em 175 simulações.

A Figura A.24 mostra o resultado da simulação do bit-flip em termos de porcentagem de erro de cada registro.

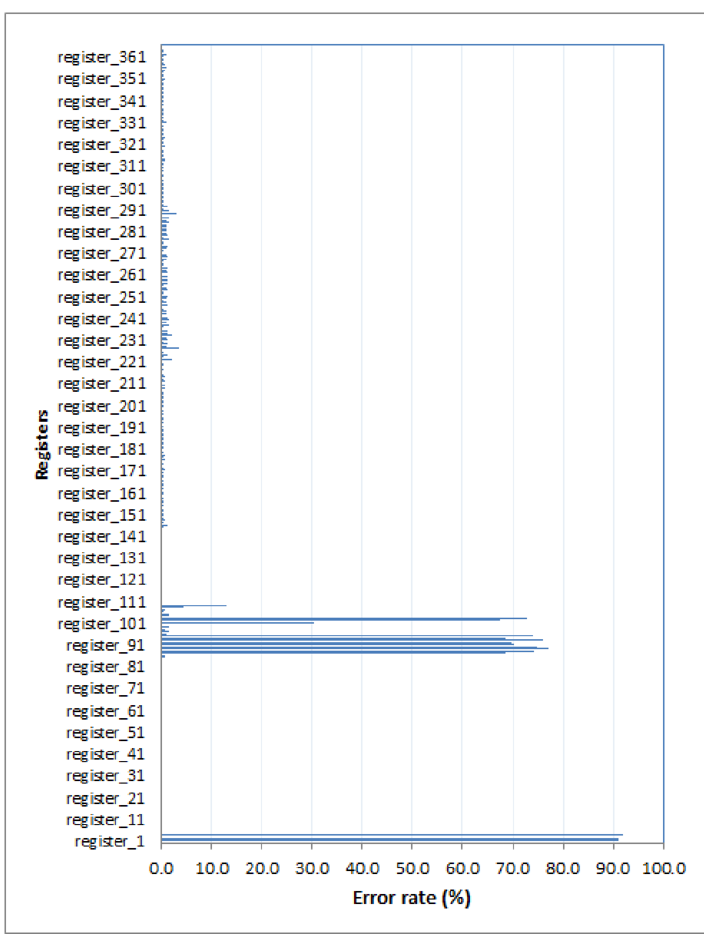


Figura A.24 Porcentagem de erro dos registros, devido a injeção de bit-flip.

Após obter o ErrorRate(bit-flip → circuit-error) calculamos o SER descrito na equação A.2. Para melhor visualização mostramos os registros com mais SER e retiramos os resultados menores de 4%. Como resultado a Figura A.25 mostra 17 registros com alto SER.

Figura A.25 Soft error rate devido ao SET na rede de relógio

Praticamente o resultado do soft error rate é igual ao resultado obtido em ErrorRate(bit-flip → circuit-error). Aparentemente, o SET injetado no início da rede de relógio gera mais falhas (dependendo da quantidade de carga), mas muitas destas falhas não são usadas pela testbench, tendo como resultado 17 registros com maior incidência em ambos casos.

A.5.6 Comparação entre rede de árvore e rede de malha

O objetivo desta simulação é investigar qual das redes de relógio é mais sensível ao SET. Usando o mesmo caso de estudo, criamos o circuito SRAM arbiter usando rede de malha (clock mesh). Logo, extraímos a rede de relógio de tipo malha com o EXT-CLK. Nas simulações, a injeção de SET foi no início da rede. Para gerar o perfil de sensibilidade que possa ser comparado entre as duas redes, refizemos a simulação da rede de árvore, 140 interações foram feitas para a rede em árvore. A figura A.26 mostra o leiaute da árvore de relógio e a Figura A.27 mostra o resultado da simulação refeita.

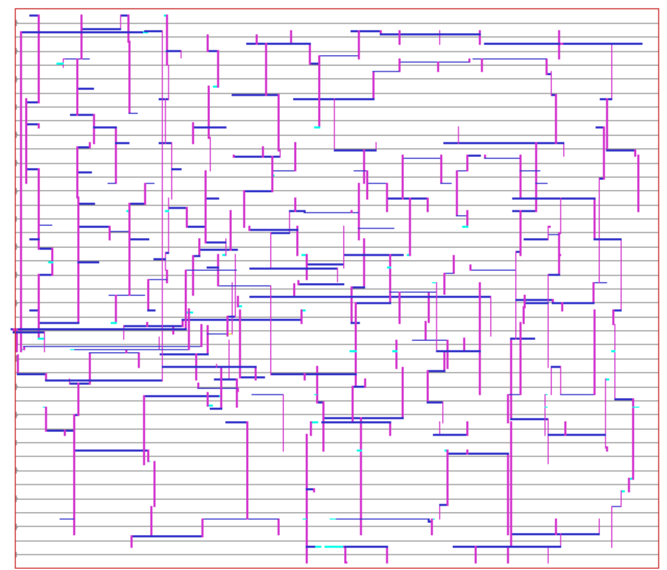


Figura A.26 Leiaute da árvore de relógio SRAM arbiter

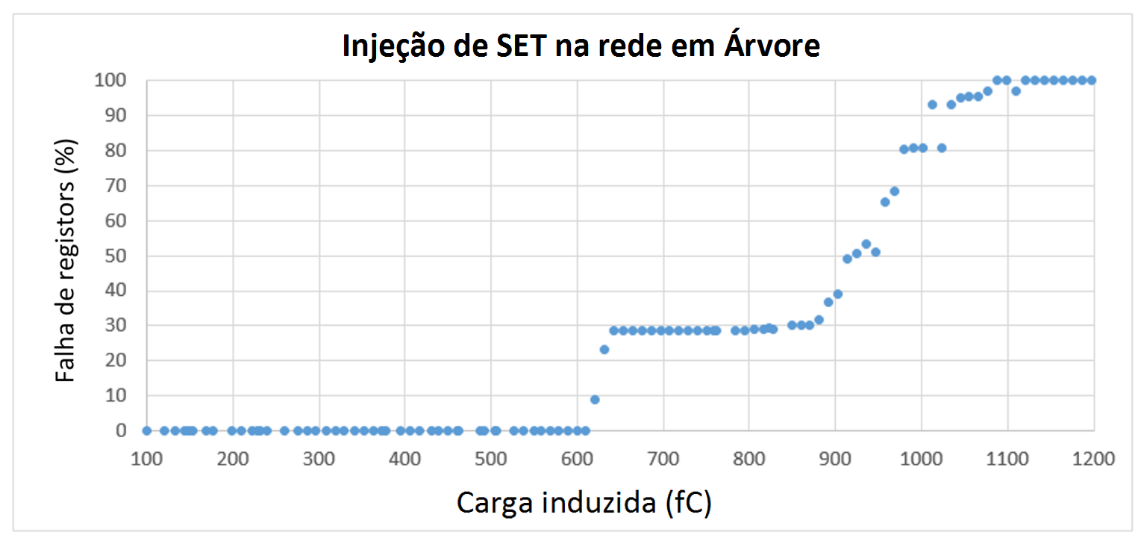


Figura A.27 Perfil de sensibilidade da árvore de relógio

Após criar a rede de malha, realizamos a extração e injetamos falhas no início da rede. Para a rede de malha, foram feitas 151 iterações com corrente de 1.258 mA até 1.273 mA, com incremento de 0.1 uA. A Figura A.28 mostra o leiaute da rede em malha e a Figura A.29 mostra o resultado da simulação.

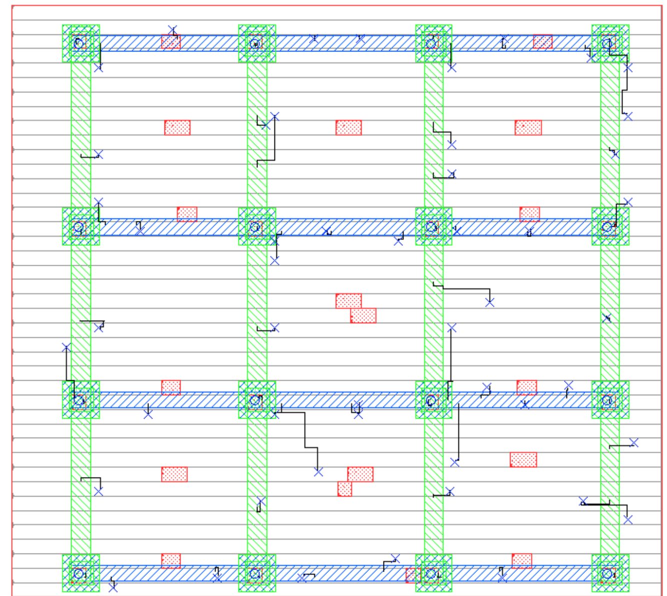


Figura A.28 Leiaute da rede em malha

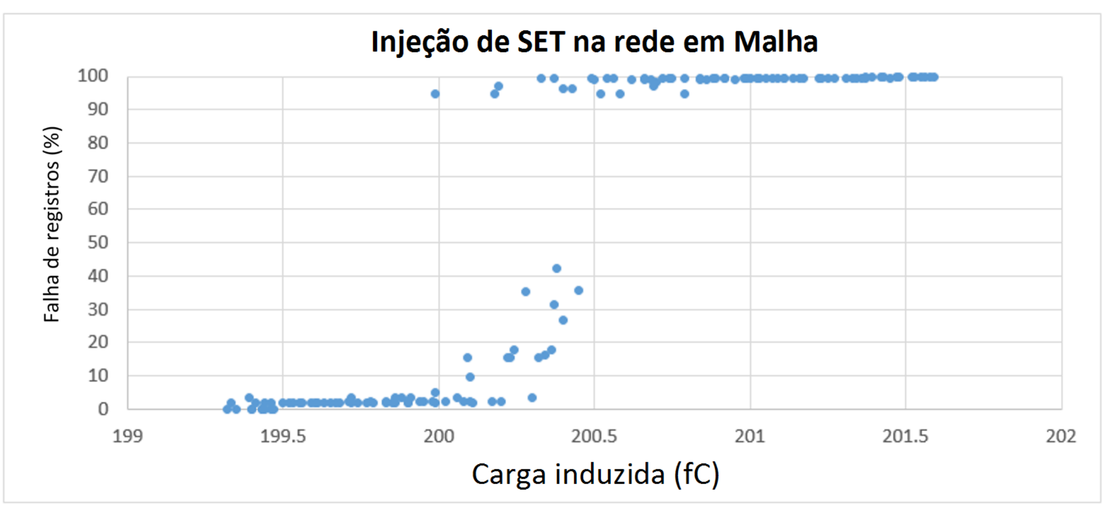


Figura A.29 Perfil de sensibilidade da rede em malha

Após a simulação, podemos observar que a rede em malha é mais sensível em comparação com a rede de árvore. A rede de árvore tem uma sensibilidade progressiva mas a rede em malha muda a sensibilidade conforme a carga supera um umbral, neste caso o umbral é de 200.5 fC. Superado o umbral, a rede em malha passa de 0% de falha a 100% de falha. Para comparar as duas redes a Figura A.30 mostra ambas simulações na mesma escala.

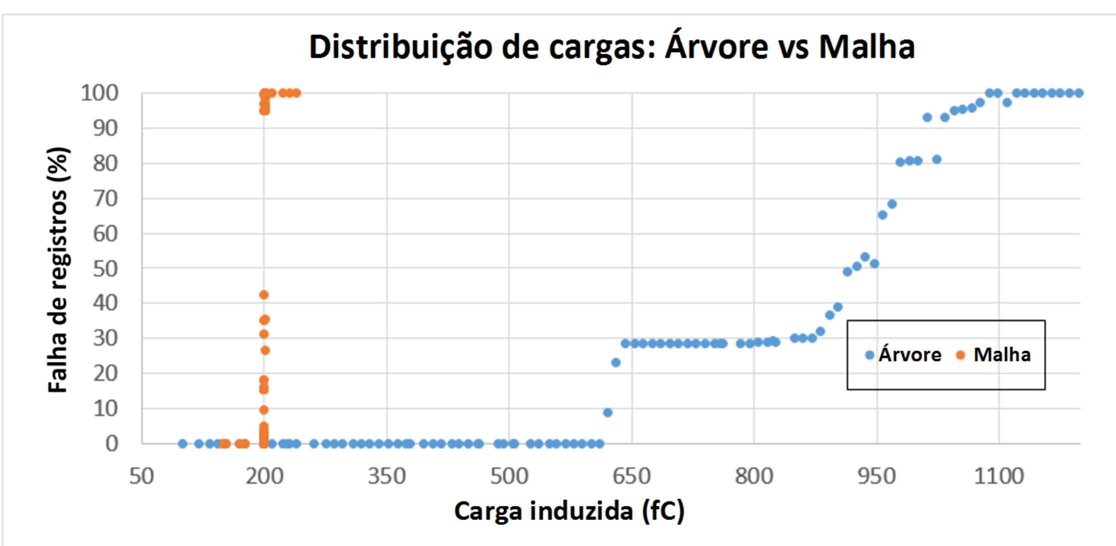


Figura A.30 Comparação entre rede de árvore e rede em malha

No caso de uma partícula atingir a saída do buffer mais próximo do início da rede de relógio, o circuito com rede em malha (mesh) será o primeiro em apresentar falhas. A sensibilidade da rede de malha ocorre devido as suas propriedades de distribuição do sinal. A rede em malha permite uma distribuição uniforme com mínimo atraso e quase zero skew, a redundância de conexões físicas e a capacitância uniforme em toda a rede permite pouca impedância e uma distribuição perfeita do sinal de relógio, mas a mesma propriedade pode ser uma desvantagem quando um SET ingressa na rede. A rede de relógio tem capacitancia não uniforme e nao utiliza redundância de conexões. Neste caso, é recomendável usar dispositivos "hardened" para melhorar a tolerância de falhas nas redes tipo malha.

A.6 Conclusões

Hoje em dia com o avanço da tecnologia os circuitos estão cada vez mais complexos. A necessidade de sincronizar cada etapa do circuito é uma prioridade, a rede de relógio se torna importante quando o sinal de relógio deve ser integrado com os dispositivos. A situação é relevante quando o circuito forma parte de um sistema aeroespacial, devido à possibilidade da radiação gerar *single event effects* na rede de relógio.

O trabalho apresentado investigou os efeitos do SET na rede de relógio, desenvolvendo uma metodologia para avaliar o circuito antes de ser fabricado. Através desta metodologia, é possível extrair a rede de relógio dos circuitos e criar o modelo que permite simular os efeitos do SET. O estudo de caso foi usado para validar a metodologia, no capitulo 5 vimos uma variedade de análises na rede de relógio que nos ajuda a identificar os nós mais sensíveis e componentes mais vulneráveis no circuito. Uma das simulações explora a possibilidade de trocar os buffers pequenos por maiores para melhorar a tolerância do SET e o resultado foi bastante favorável.

Na seção 5.6 investigamos o SER do SRAM arbiter devido ao SET na rede de árvore. O resultado da simulação mostra que 17 dos registros do circuito são altamente

susceptíveis a gerar um erro no circuito. 17 registros representam 4.6% do total de registros do SRAM arbiter.

O extrator de rede EXT-CLK permite extrair também redes em malha, o que permitiu comparar a sensibilidade das duas redes: rede em malha e rede de árvore. A secção 5.7 descreve a comparação, explica alguns pontos do resultado, e mostra que a rede em malha é mais susceptível aos efeitos do SET.

Projetistas de circuitos claramente precisam conhecer o *soft error rate* do circuito para fazer a estimativa de costo/beneficio do projeto. As informações adquiridas nos resultados podem ajudar na seleção de alguma técnica de mitigação como redundância de *hardware* ou mudar alguns dispositivo a *hardness* para nós sensíveis antes do processo de fabricação, evitando gasto financeiro.