UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

PÓS-GRADUAÇÃO EM MICROELETRÔNICA

LUCAS BRUSAMARELLO

Modeling and simulation of device variability and reliability at the electrical level

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

Prof. Dr. Gilson Inácio Wirth
Advisor

Prof. Dr. Roberto da Silva
Coadvisor

Porto Alegre, September 2011

CIP – CATALOGING-IN-PUBLICATION

Brusamarello, Lucas

Modeling and simulation of device variability and reliability at the electrical level / Lucas Brusamarello. – Porto Alegre: PPGC da UFRGS, 2011.

156 f.: il.

Thesis (Ph.D.) – Universidade Federal do Rio Grande do Sul. Pós-Graduação em Microeletrônica, Porto Alegre, BR–RS, 2011. Advisor: Gilson Inácio Wirth; Coadvisor: Roberto da Silva.

1. Microelectronics. 2. Electronic design automation. 3. Yield. 4. Circuit simulation. 5. Monte Carlo method. I. Wirth, Gilson Inácio. II. da Silva, Roberto. III. Título.

ACKNOWLEDGMENTS

This PhD thesis is a collection of works the author have developed on many research fields related reliability modeling. Through these years many people had contributed and helped to improve this final manuscript.

Gilson I. Wirth shared his knowledge on device reliability and noise modeling. Through extensive technical discussions he has actively contributed to every step and idea along this work. Gilson's technical expertise and management skills had essential impact on this work.

Roberto da Silva had a decisive role proposing to use the error propagation technique which we have been implementing since he was my advisor during my Master degree. Roberto made essential contributions to the RTS model and proposed an analytical formula for hold time violations.

Rajeev Murgai, Subodh Reddy, Gustavo Wilke and Daniel Ferrão, from Fujitusu Labs of America, had made the statistical analysis of clock networks possible by developing the clock synthesis tool and preparing the test cases.

Gustavo Neuberger contributed to the statistical analysis of hold time violations. He had proposed an initial model and made measurements of hold time of flip-flops during his PhD.

Philippe Roussel has proposed and implemented (in Mathematica) the Brussel Design of Experiments.

Vinícius and Maurício have worked on NBTI modeling and cell characterization under NBTI. They helped a lot with designing experiments, implementations and discussions.

Rafael Della Giustina has provided support for the case-studies of the RTS trap-detrap scheme.

Digeorgia Natalie da Silva has revised the manuscript and provided feedback for its improvement.

THANKS

To my always present mother Domingas and my father Pedro, who have gone through so much effort in order to give us a better life. Through very hard work they gave us the chances and support they haven't got when they were young.

To my brother Valner, the person who has always incited and motivated me to look for education and thus a better life. His dedication and hard work made him a successful professional whom I have always admired.

To my sister Ivorema, with whom I have spent great moments like trips to beaches which were my greatest fun when a kid, and is a supportive friend.

To the memory of my brother Valner Jose and my sister Jane, who I wish could have had the opportunities I did.

To my sweet Digeorgia for her love, friendship and comprehension along our journey, as well as for her lessons on Physics and Mathematics. She's a trustful and intelligent companion and I'm glad we are achieving our goals together.

To my mother.
My example of kindness and hard work.

CONTENTS

LIST OF ABBREVIATIONS AND ACRONYMS

ASIC	Application-specific integrated circuit
BTI	Bias Temperature Instability
CAD	Computer-aided design
CAS	Computer Algebra Systems
CDF	Cumulative Density Function
CMOS	Complementary metal-oxide-semiconductor
EDA	Electronic Design Automation
ELC	Encounter Library Characterizer
D2D	Die-to-die variations
DFM	Design for manufacturability
DSM	Deep submicron
DOE	Design of Experiments
FF	Flip-flop
HCI	Hot Carrier Injection
ITRS	International Technology Roadmap of Semiconductors
L	Channel length of the transistor
LAR	Least Angle Regression
LF	Low Frequency
LS	Least Squares
LER	Line Edge Roughness
LWR	Line Width Roughness
MC	Monte Carlo
MPU	Microprocessor unit
NGR	Non-Rectangular Gate
Ntr	Number of traps
NBTI	Negative Bias Temperature Instability

OCV	On-chip variations
PDF	Probability Density Function
PTM	Predictive Technology Model
Q-q	Quantile-quantile
RDF	Random Dopant Fluctuations
RSM	Response Surface Methodology
RTL	Register Transfer Level
RTS	Random Telegraph Signal
RNM	Read Noise Margin
SNM	Static Noise Margin
SEM	Scattering Electronic Microscopy
SOI	Silicon on Insulator
SSTA	Statistical static timing analysis
SRAM	Static Random Access Memory
STA	Static timing analysis
W	Channel width of the transistor
WD	Within-die variations
VLSI	Very-large scale integration
Vt	Voltage threshold
Tox	Oxide thickness

LIST OF SYMBOLS

β_p	current gain of transistor p
$\Delta\beta/\beta$	relative fluctuation of current gain
ΔVt	threshold voltage fluctuation of transistor
δvt_i	threshold voltage fluctuation caused by one trap i
$\partial F/\partial x$	sensitivity of function F w.r.t. variable x
ε_{Si}	permittivity of silicon
ε_{ox}	permittivity of oxide
Γ^M	set of simulation points M
λ	mean of a Poisson distribution
μ	mobility
μ_p	mean of p
σ_p	standard deviate of p
$\vec{v}$	vector
φ_B	built-in potential
ρ_{lm}	correlation between variables l and m
σ	standard deviation
$\sigma_{Vt,RDF}$	Voltage threshold standard deviation due to RDF
$\sigma_{Vt,LER}$	Voltage threshold standard deviation due to LER
τ_c	average time of capture
τ_e	average time of emission
$a.u.$	arbitrary units
$BIC(f)$	BIC of model f
BL	left bitline of SRAM cell
BR	right bitline of SRAM cell
C_{ox}	oxide capacitance
C_{oxe}	effective oxide capacitance

E_f	Fermi level energy
Eg	forbidden band-gap (approximately 1.12eV for Si)
E_i	energy of trap i
erf	error function
$E_T^{(i)}$	energy within the band-gap of the $i-th$ trap
$F(k_1,\ldots,k_N)$	function F with inputs variables k_1 to k_N
gm	transconductance of the transistor
I_{ds}	drain-to-source current
Ids_i	drain-to-source current of transistor i
I_{on}	on current
I_{off}	off current
K_B	Boltzmann constant
L_{eff}	effective channel length of the transistor
L_p	gate lenght of transistor p
M	matrix
N	number of simulations
N_A	average channel doping density
N_{doe}	number of simulations required by DOE
N_{IT}	number of interface traps
$N(x,y)$	Normal distribution with mean x and standard deviation y
pc	probability of capturing a electron
pe	probability of emitting a electron
$\Pr(s_i(t)=0)$	probability of trap i to be empty at time t
$\Pr(s_i(t)=1)$	probability of trap i to be occupied at time t
q	elementary charge
s_i	state of trap i (0 =empty or 1 =occupied)
$s_i(t)=x\rightarrow y$	transition of trap i at time t from state x to y
S_i-S_iO_2	interface between silicon and silicon dioxide
T_{AC}	access time of SRAM cell
$t_{CLK\rightarrow Q}$	the clock to Q delay
Td	time delay
t_{hold}	hold time of the FF
T_{WRITE}	time to write of SRAM cell
W_{eff}	effective channel width of the transistor

W_p	gate width of transistor p
Vt	voltage threshold
$Vth0$	BSIM parameter Vth0
V_{ds}	drain-to-source voltage
V_{dseff}	effective drain-to-source voltage
Vfb	flat-band voltage of the transistor
V_{gs}	gate-to-source voltage
Vox	voltage across gate oxide of the transistor
Vt_i	threshold voltage of transistor i
Vtp	voltage threshold of transistor p
xt_i	location of trap i

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

In nanometer scale complementary metal-oxide-semiconductor (CMOS) parameter variations pose a challenge for the design of high yield integrated circuits. This work presents models that were developed to represent physical variations affecting Deep-Submicron (DSM) transistors and computationally efficient methodologies for simulating these devices using Electronic Design Automation (EDA) tools.

An investigation on the state-of-the-art of computer models and methodologies for simulating transistor variability is performed. Modeling of process variability and aging are investigated and a new statistical model for simulation of Random Telegraph Signal (RTS) in digital circuits is proposed.

The work then focuses on methodologies for simulating these models at circuit level. The simulations focus on the impact of variability to three relevant aspects of digital integrated circuits design: library characterization, analysis of hold time violations and Static Random Access Memory (SRAM) cells.

Monte Carlo is regarded as the "golden reference" technique to simulate the impact of process variability at the circuit level. This work employs Monte Carlo for the analysis of hold time and SRAM characterization. However Monte Carlo can be extremely time consuming. In order to speed-up variability analysis this work presents linear sensitivity analysis and Response Surface Methodology (RSM) for substituting Monte Carlo simulations for library characterization.

The techniques are validated using production level circuits, such as the clock network of a commercial chip using 90nm technology node and a cell library using a state-of-the-art 32nm technology node.

Keywords: Microelectronics, electronic design automation, yield, circuit simulation, Monte Carlo method.

RESUMO

O efeito das variações intrínsecas afetando parâmetros elétricos de circuitos fabricados com tecnologia CMOS de escala nanométrica apresenta novos desafios para o yield de circuitos integrados. Este trabalho apresenta modelos para representar variações físicas que afetam transistores projetados em escala sub-micrônica e metodologias computacionalmente eficientes para simular estes dispositivos utilizando ferramentas de Electronic Design Automation (EDA).

O trabalho apresenta uma investigação sobre o estado-da-arte de modelos para variabilidade em nível de simulação de transistor. Modelos de variações no processo de fabricação (RDF, LER, etc) e confiabilidade (NBTI, RTS, etc) são investigados e um novo modelo estatístico para a simulação de Random Telegraph Signal (RTS) e Bias Temperature Instability (BTI) para circuitos digitais é proposta.

A partir desses modelos de dispositivo, o trabalho propõe modelos eficientes para analisar a propagação desses fenômenos para o nível de circuito através de simulação. As simulações focam no impacto de variabilidade em três diferentes aspectos do projeto de circuitos integrados digitais: caracterização de biblioteca de células, análise de violações de tempo de hold e células SRAM.

Monte Carlo é a técnica mais conhecida e mais simples para simular o impacto da variabilidade para o nível elétrico do circuito. Este trabalho emprega Monte Carlo para a análise do skew em redes de distribuição do sinal de relógio e em caracterização de células SRAM considerando RTS. Contudo, simulações Monte Carlo exigem tempo de execução elevado. A fim de acelerar a análise do impacto de variabilidade em biblioteca de células este trabalho apresenta duas alternativas a Monte Carlo: 1) propagação de erros usando aproximação linear de primeira ordem e 2) Metodologia de Superfície de Resposta (RSM).

As técnicas são validados usando circuitos de nível comercial, como a rede de clock de um chip comercial utilizando a tecnologia de 90nm e uma biblioteca de células usando um nó tecnológico de 32nm.

Palavras-chave: Microeletrônica, projeto auxiliado por computador, ruído de baixa freqüência, confiabilidade de circuitos integrados, método Monte Carlo.

1 INTRODUCTION

Previously, advances in very-large scale integration (VLSI) circuit design primarily relied on circuit improvements derived from technology scaling. In those days abstraction relied on enough performance that could be traded for design simplicity.

Synthesis and optimizations in the design flow of digital circuits employed by those technologies were based on corner-based analysis. In this approach delay, power and other design constraints are computed from electrical parameters found to be extreme cases during characterization.

For Deep Sub-Micron (DSM) technologies, variations in the manufacturing process of electronic devices poses major challenges for the industry. Process variability are the fluctuations of the physical and electrical characteristics of the transistors caused by deviations during the manufacturing process. These deviations cause the current-voltage characteristics of the transistors to be different from the nominal specification: they become statistical rather than deterministic. Such process related issues have been posing new challenges to the design of integrated circuits because both Electronic Design Automation (EDA) software and circuit designers need to make use of techniques that correspond to this new paradigm.

Electrical parameters variability may be decomposed into a spatial and a temporal component, as expressed in figure 1.1. The spatial component can be further divided into die-to-die variations (D2D) and within-die variations (WD) (ZUCHOWSKI et al., 2004; ORSHANSKY et al., 2002).

The die-to-die variations affect equally all the elements within the same chip. D2D variations may be originated from equipment asymmetries like asymmetries in chamber gas flows and thermal gradients, as well as imperfections in equipment operation and process flow. These asymmetries and imperfections cause a shift on the average value of a parameter of the wafer or lot of wafers. One example of D2D variation is the thickness of the resist along the wafer, which is constant inside a wafer but might vary from wafer to wafer (BOWMAN; DUVALL; MEINDL, 2002). In technologies older than 180 nm the D2D variations used to be orders of magnitude higher than the WD component, which was safely neglected until recently. EDA industry is familiar with methodologies to deal with D2D components: corner-based analysis. In this technique, the circuit is simulated at different PVT (process, voltage, temperature) extreme conditions in which the circuit is expected to operate. Thus on corner analysis all the transistors are correlated, which is the correct assumption for D2D variability.

Within-die variations cause the electrical characteristics of the transistors to fluctuate non-uniformly across a single chip. It can be further decomposed into a systematic and a random component (BOWMAN; DUVALL; MEINDL, 2002). The systematic component may be originated by optical aberrations causing parameter shifts within a chip.

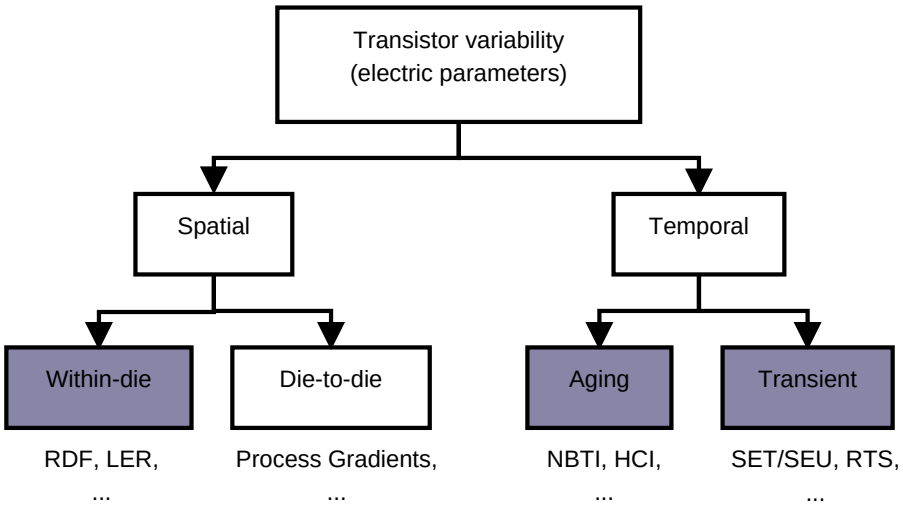


Figure 1.1: Classification of types of variations affecting MOSFET devices (WIRTH, 2010)

These shifts present a pattern across the die and is usually dependent on the device position: for instance the gate length of nearby transistors have a systematic component which cause them to shift accordingly. Random within-die variations are originated from the discreteness of matter and energy, as the number and position of dopant atoms, photo resist molecules, and photons. Random within-die variability is also called intrinsic variability because it cannot be eliminated, being rather a limitation of the materials and the process the transistors are built with. A well known example of WD parameter is threshold voltage (Vt) variability due to the Random Dopant Fluctuations (RDF) (MAHMOODI; MUKHOPADHYAY; ROY, 2005). Due to its intrinsic characteristic which is agravatted with the technology scaling, random WD variations started dominating over the D2D component and have been increasing at each technology node.

Temporal variability can be further divided into aging and transient variations. The main causes of temporal variability are: 1) the capture and emission of electrons by traps in the Si-SiO2 interface and silicon oxide of the devices and 2) spurious radiation particles hitting the device. Aging is the systematic degradation of the transistor characteristics, as for instance the current strength of a transistor decreasing over time due to Bias Temperature Instability (BTI). Transient variability are either instantaneous or intermittent changes in the device current $\times$ voltage curve, which can be caused by radiation (nowadays even at the Earth's surface) or Random Telegraph Signal (RTS).

Random Telegraph Signal (RTS), also known as Low Frequency (LF) noise, is a performance limiting factor for deep sub-micron CMOS devices. This noise is due to succeeding electron capture and emission at the interface and in the bulk of the gate dielectrics. This phenomena causes oscillations in the transistor voltage threshold Vt and drain-to-source current I_{ds}. The propagation delay of a gate depends on the capability of its transistors to drive current. A smaller current driven by the transistor means larger propagation delay, which may lead to timing violations (failures) in a circuit. Hence, the variation in transistor drive current due to RTS may lead to circuit failures in future technology generations, and statistical modeling of random telegraph signal is required.

In order to maximize performance, the reduction of transistor dimensions is not compensated by the corresponding reduction in operating voltage (ASSOCIATION, 2009).

The increased stress causes a significant degradation of electrical parameters of the transistors over time. This phenomenon is called aging. Currently, the dominant factor to limit the lifetime of a PMOS transistor is the so-called Negative Bias Temperature Instability (NBTI) (KACZER et al., 2005). For NMOS transistors Hot Carrier Injection (HCI) is usually the major concern. Over time, this mechanism degrades the transistor threshold voltage Vt, resulting in speed degradation of the logic cells and causing timing violations, which implies circuit malfunction.

Due to its assumptions leading to excessive pessimism, guard-bands or corner-based design styles tend be less realistic at each new technology node. New strategies to model statistical process variations become critical for ensuring high yield in future products using sub-45nm technologies (NASSIF, 2000). Corner analysis can be excessively pessimistic and inaccurate (VISWESWARIAH, 2003) due to the fact that by definition the corners must capture the fastest and slowest possible conditions of the circuit. Because in Application Specific Integrated Circuit (ASIC) timing is the main target to be met, the pessimism regarding the delay constraints increases the need for stronger gates and buffers, thus increasing of area and power consumption of the circuit.

In order to get the most from the technology scaling, *Computer Aided Design* (CAD) tools and the fabrication process must be tied together. Variations and aging, which can be modeled statistically, must be taken into account in the early design phases, and there must be CAD tools capable of predicting the percentage of functional circuits in a wafer. Therefore, along with timing, area and power, yield and the impact of aging must be additional constraints to be taken into account when designing circuits using recent technologies.

Statistical analysis of electrical characteristics of analog and digital circuits is often performed by using Monte Carlo Method (AMAR, 2006), what implies in a large amount of simulations at electric level. Monte Carlo simulations are the standard employed by industry for the analysis of variations at electrical level, and is supported by current versions of electrical simulation tools (SYNOPSYS, 2005).

Statistical Static Timing Analysis (SSTA) gives at logic level a quantitative risk management for the design as a function of the circuit topology, the electrical parameters and the variations (VISWESWARIAH, 2003). In order to apply SSTA methodology, cell libraries are characterized at the electrical level, for which Monte Carlo simulation is nowadays commonly employed. Larger designs may be decomposed into functional blocks and treated at different levels of abstraction. A block may be a simple or complex gate, a sequential block (e.g. flip-flop) or a memory cell. Commercial EDA tools are starting to support statistical characterization at cell level. For instance Cadence's Encounter Library Characterizer and Synopsys NCX can automatically compute linear sensitivity analysis, which can be further employed for statistical characterization.

This thesis studies design automation methodologies and develops models to deal with technology-related issues, such as process variability, noise and aging.

1.1 Motivation

Traditionally, designers of digital ICs relied on levels of abstractions that could hide the effects of process variations on their product. The designer could expect the chip to work within the corners defined by the foundry. Actually the designer expected a *high percentage* of chips to satisfy the design constraints. The concept of *yield* had been implicit for the designer and left to the foundry to take care of. Foundries used to compute

the corners through characterization processes that could be hidden from the designer for simplicity.

Analog designers have already been experiencing the problems of process variations for decades. Analog and mixed-signal circuits often employ channel critical dimensions much larger than minimum CD, attenuating the affects of process variability. As well, matching techniques are usually employed for transistors that must perform ideally identically. These techniques require both the designer and the EDA tool to be aware of technological and physical details of the device process.

It became well known that corner analysis can guarantee sufficient yield after fabrication at the expense of performance and power. Each technology node requires more complex assessment analysis tools, partly due to the increase in circuit complexity and partly due to the increase of physical phenomena that must be taken into account. To complete the chain, computational power itself has been increasing steady-paced, allowing more complex methodologies to be computationally feasible.

The demand for more accurate transistor representations and techniques for accurate circuit analysis has been pushing forward two areas of research and development: 1) modeling of transistor reliability phenomena and 2) methodologies for analysis of integrated circuits considering these phenomena. Table 1.1 presents a piece of ITRS 2009 section on ”Modeling and Simulation”. ITRS points to the need of modeling trap-induced reliability issues such as RTS and NBTI, as well as to the need of circuit models for CMOS devices including reliability. The present work will focus on presenting advances on these two inter-related areas of research.

Thus, back-end designers have to learn how to design digital circuits considering issues that previously affected only analog circuits. This poses great challenges for physical synthesis of ICs as more technology-related issues must be brought to the design flow. In order to obtain more accurate timing and power estimates, abstraction must be sacrificed in such a way that the designer must be more aware of the silicon implementation.

1.2 Contributions of this work

Accurately modeling variability and reliability of transistors is becoming a major challenge for the advance of the semiconductor industry. Moreover, circuit models for propagating nano-scale devices issues to circuit-level simulation must be developed. This thesis focuses on two main topics: 1) modeling of device variability and reliability and 2) methodologies for circuit level simulation of these issues. The context and the contributions of this work in these fields are discussed in the next subsections.

1.2.1 Modeling of transistor reliability

Device modeling is focused on the main sources of variations in today technologies: Random Dopant Fluctuations (RDF), Line Edge Roughness (LER), Negative Bias Temperature Instability (NBTI) and Random Telegraph Signal (RTS). RDF and LER are classified as process-related (spatial) variability while RTS and NBTI are time-dependent (temporal) reliability . This work will discuss in detail a recent model proposed for simulating the trap-detrap phenomena that causes RTS and NBTI. This work presents the following contributions on modeling of reliability issues that affect transistors:

Random Telegraph Signal: In this work variations in drain current over time due to the Random Telegraph Signal (RTS) are modeled as transient changes in transistor

Table 1.1: Section ”Modeling and Simulation” in ITRS 2009 (extracted from (ASSOCIATION, 2009)).

Year of Production	2009	2010	2011	2012	2013	2014	2015	2016	2017
DRAM 1/2 Pitch (nm)	52	45	40	36	32	28	25	22.5	20.0
MPU Physical Gate Length (nm)	27	24	22	20	18	17	15	14.0	12.8
...									
Reliability and noise modeling *	HF, 1/f and RTS noise modeling		Trap generation during operation (HCI, NBTI, PBTI, ...) for conventional and new gate stacks			Degradation mechanisms for novel logic and memory devices			
...									
Active devices	Circuit models for bulk and SOI CMOS devices including reliability, aging and influences of layout, process variability and random fluctuations; ...			Extension to multigate CMOS; standardize SOI and multigate circuit models [7]			Circuits models for nanoscale devices		

threshold voltage Vt. Modeling RTS as a source of Vt variation, on top of other reliability phenomena, is very convenient for simulation of digital blocks and can be easily propagated to circuit level following the circuit-level methodologies described in this work. Two RTS models are proposed: a static model and a dynamic time-dependent model. The static model is further experimented in a SRAM memory as case study to evaluate the impact of RTS in memories. The dynamic model works for both RTS and NBTI.

Negative Bias Temperature Instability: NBTI is related to generation and/or activation of interface traps. Experimentally this is shown to occur when a device is biased in inversion mode of operation, regardless of current flow, and is aggravated by temperature. In this work NBTI is modeled as an effect that shifts V_t of the transistors over time, impacting in speed degradation of the logic cells. A trap-detrap model valid for both RTS and NBTI is presented.

1.2.2 Methodologies for statistical simulation

On top of advanced device variability modeling this work intends to propose computational efficient models for propagating these issues to circuit level simulation. These techniques allow the designer to estimate the circuit performance and yield at early design stages, before silicon. Computer simulations for circuits considering process variations must be accurate but also there must be a compromise on run-time.

In order to address the run-time/accuracy trade-off this thesis proposes to employ linear sensitivity analysis and Response Surface Methodology as alternatives to time-consuming Monte Carlo simulations. The following simulation methodologies have been

implemented to cope with circuit reliability issues on the realm of sub-nanometer transistor era:

Monte Carlo: Monte Carlo method is the most widespread statistical simulation tool and has been applied in many domains since 1950s. Monte Carlo simulations have been implemented for every circuit analyzed in this work, since it is the most accurate method. For statistical cell characterization MC is implemented as a reference method in order to benchmark faster alternatives. The implementations of SRAM characterization under RTS and hold time violation analysis were fully based on Monte Carlo.

Linear Sensitivity Analysis: Linear sensitivity analysis is a simple and efficient alternative for Monte Carlo simulation. Our research group at UFRGS has been studying linear sensitivity analysis since 2005 to cope with characterization of small circuit blocks. In this work linear sensitivity analysis is applied to the statistical characterization of standard cell library and to the analysis of the impact of NBTI in standard cells. Statistical standard cell characterization was validated using Cadence's Encounter Library Characterizer (ELC) support for linear sensitivity analysis.

Response Surface Methodology: A Response Surface Methodology encompasses two steps: Design of Experiments (DoE) and a model fitting. In collaboration with IMEC (Interuniversity Microelectronics Center - Belgium) a novel RSM flow based on a new DoE, a polynomial model selection algorithm, and the subsequent substitution of electrical simulation by the regression function was invented in order to characterize standard cell libraries. RSM was integrated as part of IMEC's statistical cell characterization tool suite (see appendix B). RSM was implemented as a set of scripts interfacing with the existing IMEC framework for statistical cell characterization supporting commercial tools such as Cadence Encounter Library Characterizer (ELC).

1.2.3 Circuit simulation of reliability issues

This work presents strategies for design on the realm of variability on many areas of the design flow: from I-V curves of a transistor to the influence of process variability on hold time violations of logic paths. In order to validate the models and methodologies developed, as well as to show their applicability to relevant design issues, this work has studied some specific problems designers are starting to face today with technology scaling tend to be aggravated in the future. These problems rise from the fact that CAD tools not yet fully support, i.e. automatically support, reliability modeling and statistical design, implying that more research is needed in this field in order to propose suitable methods of risk evaluation of the circuits. The circuit simulation and analysis work focuses on the following problems related to the design of digital integrated circuits:

Analysis of Clock Network of Digital Circuits: The clock signal is the most important global signal in a synchronous circuit. On recent technologies process variations, noise and aging impose challenges for the design of reliable clock networks. They cause changes in the time delay for the clock signal to arrive at the different flip-flops, causing undesirable clock skew. This work analysis the impact of process variations on the delay of the clock signal and the clock skew. Normality tests in measurements of clock skew are performed in order to check the data distribution, and a statistical model for the clock skew is proposed.

Modeling Hold Time Violations of Digital Circuits: hold time violations can be modeled as a random variable which is function of the race immunity of the FF and the clock skew. This work presents Monte Carlo experiments of clock skew and a normal distribution is shown to fit them very well. After coming up with a statistical model for hold time violations due to race conditions, we research methods for fixing those timing violations. We propose a statistical methodology for computing the total amount of delay to be inserted in the data-path to satisfy the yield constraint.

The standard cell design flow needs a set of pre-characterized elements, which are specific to each technology and can be re-used for every design in that technology. This work proposes solutions for the following characterization steps:

Standard cell library characterization: in a typical design flow of ICs the connection between electrical-related parameters and timing characteristics of the circuits is made at cell characterization level. A representative subset of standard cells using a 32nm production level library and statistical device compact model is characterized. Two simulation speedup techniques (RSM and linear sensitivity analysis) are validated and compared to Monte Carlo. The methodologies show good compromise between accuracy and run-time as compared to Monte Carlo. Such statistical library parameters can then be propagated to higher level of the design flow as Static Timing Analysis.

Characterization of SRAM: SRAM cells are designed using small feature sizes and employ state-of-the-art process technology in order to achieve maximum density. Memories are the first circuits to be implemented on new process technologies and are the first to benefit from the scaling, however they always experience the challenges imposed by the devices unreliability and process variations. We investigate sources of failures in memories and use these as models to propagate the effects of RTS and variability to memory cells.

2 IMPACT OF PROCESS VARIATIONS TO THE DESIGN FLOW OF INTEGRATED CIRCUITS

The design flow of ASICs follows a top-down approach (WESTE; HARRIS, 2005), as represented by figure 2.1. The circuit is initially specified by either a high-level behavioral description or a structural description. Each step of the flow generates a lower-level abstraction equivalent to the previous step and closer to the actual implementation.

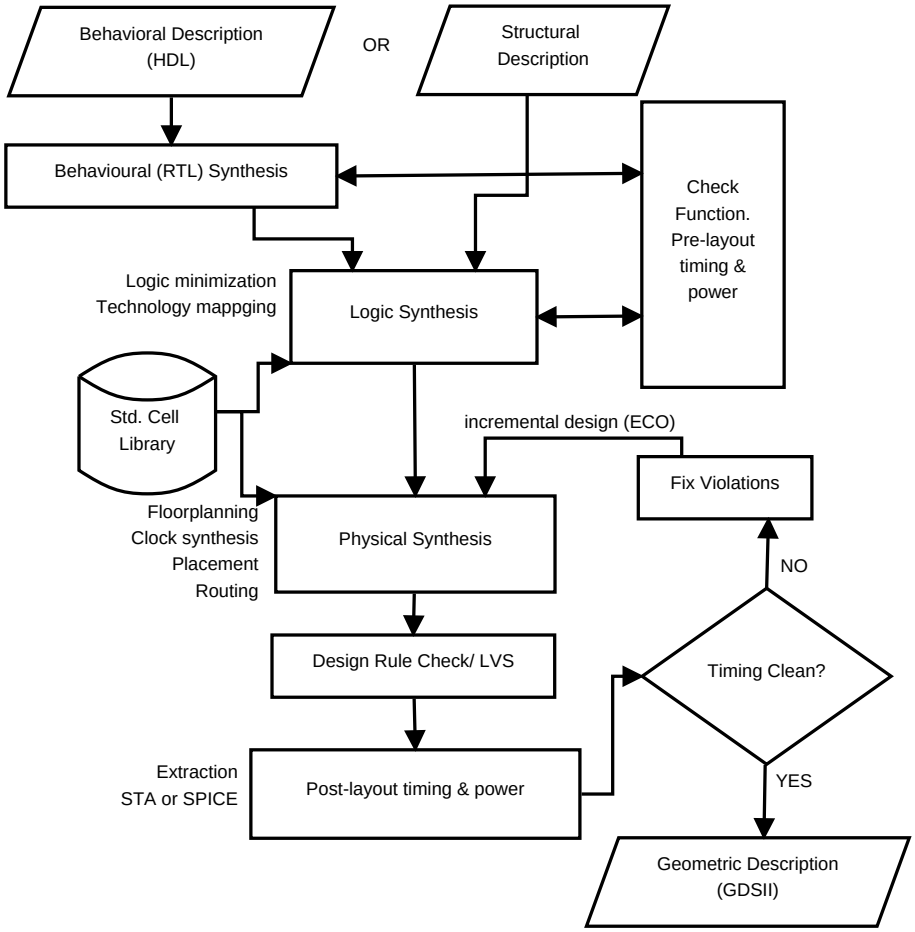


Figure 2.1: Top-down ASIC design flow

The initial behavioral description consists of a system specification at a high level of abstraction, describing how the circuit behaves as a function of its inputs and states. A behavioral description can be an algorithmic description or a data-flow description,

also referred as Register Transfer Level (RTL). The system can be specified through conventional programming languages, such as C, although hardware description languages (HDL) such as VHDL, Verilog and SystemC are preferred since they are specific to circuit description. One of the main steps of behavioral synthesis is the hardware assignment, which allocates the specific combinational and sequential hardware components (adders, multipliers, memories) as well as its quantities. This early step must implement estimates of timing, area and power and support functional verification through test-bench simulation.

Alternatively, some small blocks or special-purpose designs can be directly specified as a structural description, which consists of the circuit being specified in terms of components and interconnections. This is a lower level representation than the behavioral description. It allows more control to the designer over the system implementation and avoids the automated behavioral synthesis.

The input to logic synthesis is a gate-level netlist of the circuit. The gate level is a structural description, being the circuit fully described in terms of components and interconnections. The main goals of logic synthesis are logic minimization and technology mapping. Logic minimization targets at the simplest equivalent circuit optimized for the design constraints. Technology mapping consists of generating an equivalent circuit employing only components existing in the pre-characterized standard cell library.

Physical synthesis takes as input the structural representation of the circuit, a netlist without the physical data like geometries or position, and generates the geometric description of the circuit: the layout. The layout is the lowest level of abstraction in an electronic design automation flow. The circuit ready for tape-out is a description of the masks of the circuit and can be sent to the foundry in Graphic Database System II (GDSII) stream format. The steps of physical synthesis are floor-planning, partitioning, clock synthesis, placement and routing. The main goal of physical synthesis is to minimize interconnects in order to minimize delay, power and area. However variability has been posing new challenges for physical synthesis: maximization of yield and manufacturability nowadays are also constraints to be met. For instance, the router is required not only to find an optimal path between two components, but it is also required to follow manufacturability guidelines when drawing the interconnections in order to ensure proper yield.

Delay, area and power of the digital circuit are estimated at every synthesis level, as well as equivalence checks to verify whether the description is equivalent to the previous one or not. However the closer the design gets to implementation the more information is available making the assessment more accurate. The most accurate timing estimate is computed after the last step of physical synthesis, routing, which draws the interconnections. Thereby, after detailed routing, there is a step of parasitic extraction, which accurately models wires as resistances and capacitances. The data-path delays can be computed by electric-level simulation (spice) or by Static Timing Analysis (STA). By using these wire models for the interconnections, their delays can be computed accurately. In STA, standard cell libraries are pre-characterized and their delays and power are readily computed as a function of the input slew and the load they drive. STA reports the critical paths (paths not meeting the timing constraints) that cause hold and setup time violations. These violations must be fixed, usually by an incremental step of physical synthesis.

Thus, process variation and reliability issues add even more strain over the steps of physical synthesis, specially timing analysis and circuit optimization. This thesis focuses on modeling process variability and device reliability at electric level description of digital ICs and specific circuits, as for instance memories. This chapter presents the typical

design flow and demonstrates how process variability effects, which impact the I-V curve of the transistors can be propagated to higher level metrics of the design. The next sections are presented in a bottom-up fashion: starting from the impact of variability to the transistors I-V curves and then presenting a methodology to propagate these effects to high level metrics such as delay.

2.1 DC Transfer I-V Curve

On-chip-variations induce the devices to present electrical characteristics different from the nominal (average) specification. The I-V transfer characteristic is the identity of a transistor. Variations in the I-V curve directly impact high-level metrics of the digital circuit. A transistor with higher Ion (Ids when Vds and Vgs are maximum) is faster and consumes more energy than a device with lower Ion.

Accurate device modeling has been a necessity of semiconductor industry since its beginning. Transistor models are sets of equations describing the operation of the transistor. These equations are employed by electrical simulators to compute the behavior of the circuit. Since the introduction of Berkeley's SPICE in 1972 and with the always growing dependency on EDA tools, MOSFET models have evolved in complexity to be able to accurately describe the device behavior under the current, voltage and environmental conditions. The first MOSFET model to be implemented by SPICE in 1972 was the level 1 model, or Quadratic I-V Model, which is the simplest MOSFET model to compute Ids as a function of Vgs and Vds. The quadratic model is inaccurate and is not appropriate for modern transistors. The curve is divided into three regions of operation (this discontinuity causes convergence issues) and Ids can be approximated as proposed by Massobrio (1999):

$$I_{ds}(V_{ds},V_{gs}) \approx \begin{cases} 0 & V_{gs} < V_t; \text{cutoff} \\ \beta\left[(V_{gs}-V_t)V_{ds} - \frac{V_{ds}^2}{2}\right] & 0 < V_{ds} < V_{gs} - V_t; \ \text{linear} \\ \frac{\beta}{2}\left[(V_{gs}-V_t)^2\right] & 0 < V_{gs} - V_t < V_{ds}; \ \text{saturation} \end{cases} \tag{2.1}$$

where

$$\beta = \mu C_{ox}\frac{W}{L}\ ; \ \text{and} \ \ C_{ox} = \frac{\varepsilon_{ox}}{t_{ox}}$$

With the advance of transistor technology on the upcoming decades, transistor models had to be constantly enhanced in order to incorporate new physical-related technology characteristics. Some of the most accurate and widely employed by industry Physics-based transistor models are: MOS9 and MOS11 by Philips Semiconductor Research, PSP by Arizona State University (ASU) and NXP (former Philips) (GILDENBLAT et al., 2005), and the Berkeley Short-channel IGFET Model (BSIM) family. Regarding the BSIM family, the most common versions are the BSIM version 3v3, BSIM3, and the BSIM version 4 (HU, 2009), of which the latest version is BSIM4v6.4. PSP is a surface-potential based MOS Model, which leads to a more accurate single-equation formulation valid for the whole operation regime of the transistor than the voltage threshold based

model BSIM (SCHOLTEN et al., 2009). Not only recent physics-based models like BSIM4 and PSP take into account the relevant physical effects of the device, but they also provide enhanced convergence during simulation due to their continuity, i.e. the source-to-drain current is a continuous function over the operation range. As an exercise to illustrate the variability on the DC IV curve, let's consider the single source-to-drain current equation implemented by BSIM4 (HU, 2009):

$$\begin{aligned} I_{ds} &= \frac{I_{ds0}NF}{1+\frac{R_{ds}I_{ds0}}{V_{dseff}}}\left[1+\frac{1}{C_{clm}}ln\left(\frac{V_A}{V_{ASAT}}\right)\right] \\ &\times \left(1+\frac{V_{ds}-V_{dseff}}{V_{ADIBL}}\right)\times\left(1+\frac{V_{ds}-V_{dseff}}{V_{ADITS}}\right)\times\left(1+\frac{V_{ds}-V_{dseff}}{V_{ASCBE}}\right) \end{aligned} \quad (2.2)$$

where I_{ds0} is the source-to-drain current valid from sub-threshold regime to strong inversion regime, NF is the number of finger of the device, R_{ds} is the drain-to-source resistance, V_{ds} is the effective drain-to-source voltage and V_{dseff} is the effective drain-to-source voltage, C_{clm} is parameter modeling channel length modulation. The variables V_{ADIBL}, V_{ADITS} and V_{ASCBE} model the Early Voltages due to Drain-Induced Barrier Lowering (DIBL), Drain-Induced Threshold Shift (DITS) by Pocket Implant and Substrate Current Induced Body Effect (SCBE). For a detailed description on how to compute all these parameters refer to Hu (2009). The intermediate variable V_A is computed as $V_A = V_{ASAT} + A_{ACLM}$, with (HU, 2009):

$$V_{ASAT} = \frac{E_{SAT}L_{eff} + V_{dsat} + 2R_{ds}vsatC_{oxe}W_{eff}V_{gseff}\left[1-\frac{A_{bulk}V_{dsat}}{2\left(V_{gseff}+2v_t\right)}\right]}{R_{ds}vsatC_{oxe}W_{eff}A_{bulk} - 1 + \frac{2}{\lambda}} \quad (2.3)$$

where V_{dsat} is the saturation voltage, A_{bulk} models the bulk charge effect (Vth is not constant along the channel when $V_{ds} \neq 0$ due to non-uniform depletion width), E_{sat} is the critical electrical field at which the carrier velocity becomes saturated, C_{oxe} is the effective oxide capacitance, *vsat* is the saturation velocity , W_{eff} is the effective transistor width, v_t is the thermal voltage, λ models the non-saturation effects in P-type MOSFETS. Again, refer to Hu (2009) for a detailed description of these equations.

Figure 2.2 presents a typical output of DC measurements of a sample of transistors. In this case they come from a Monte Carlo simulation of I-V curves of a 32nm Predictive Technology Model (PTM) device. Variability is modeled by assuming five BSIM4 parameters as random variables: $L_{int}(3\sigma = 3nm), W_{int}(3\sigma = 3nm), Tox_e(3\sigma = 10\%), \mu_o(3\sigma = 5\%), Vth_o(3\sigma = 10\%)$, which are in accordance with the International Technology Roadmap of Semiconductors 2009 (ASSOCIATION, 2009). The linear and saturation regions affect the circuit performance and the sub-threshold region implies in variation of the leakage current. This exercise shows that for this set of parameter variations, I_{on} can vary by up to 40%.

2.2 Compact transistor variability model

Nominally identical devices end up having different I-V curves due to process variability. A compact transistor model aims at modeling the effects of one or many physical sources of variation to the transistor I-V curve.

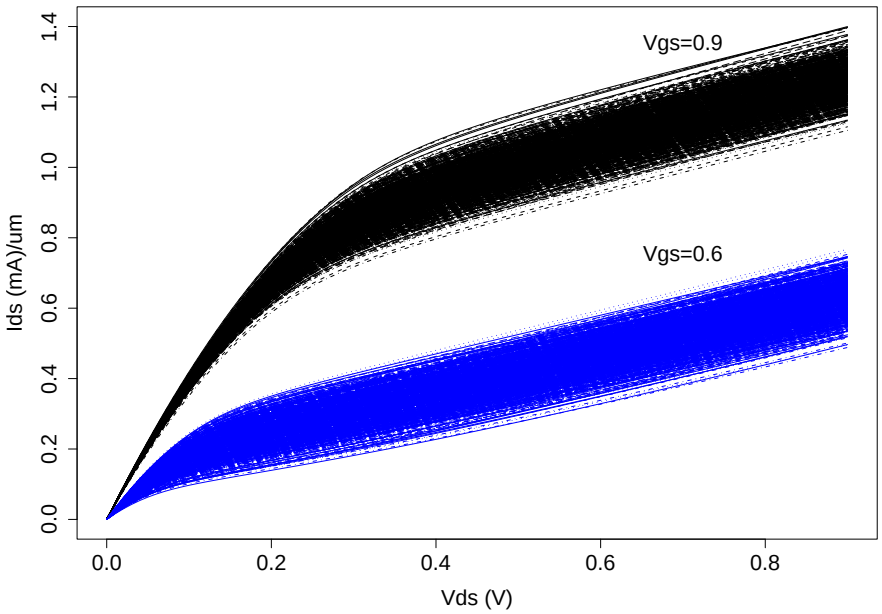


Figure 2.2: Monte Carlo simulation of 32nm transistors I-V characteristic.

Although the quadratic model of equation 2.1 is innacurate to model current technologies, it shows that in a first order the most relevant parameters to express Ids as a function of Vds and Vgs are the threshold voltage Vt and current gain β. This gives the designer an opportunity: for instance by appropriately modifying the transistor Vt in the netlist, the designer can simulate the effect of a source of variability as RDF, RTS, NBTI and even dimension-related issues such as LER (sources of variation are discussed in the next chapters).

Figure 2.3 presents 3 possible methodologies for simulating variability in the I-V curve of transistors. The simplest way is represented by figure 2.3(a): some electrical simulators as HSPICE, SPECTRE and NGSPICE allow the user to specify a Vt shift during instantiation of the transistor, but only for BSIM transistor models. This approach is widely employed because of its simplicity and concordance with measured data (NASSIF et al., 2007).

Another methodology consists of replacing the transistor by an equivalent circuit which includes one voltage source in series with the gate (as figure 2.3(b)) or one voltage source in series with the gate and one current source in parallel with source and drain (as figure 2.3(c)). The voltage and current source model variations in Vt and $\Delta\beta/\beta$, respectively. The advantages of (b) and (c) are 1) independence on the transistor model and 2) independence on the electrical simulator.

Figure 2.4 reports the simulation results of V_{th} and β of a 32nm statistical device model. Comparison of this two-parameter compact variability model shows excellent agreement with the reference Monte Carlo simulations on the foundry variability model (ZUBER et al., 2010).

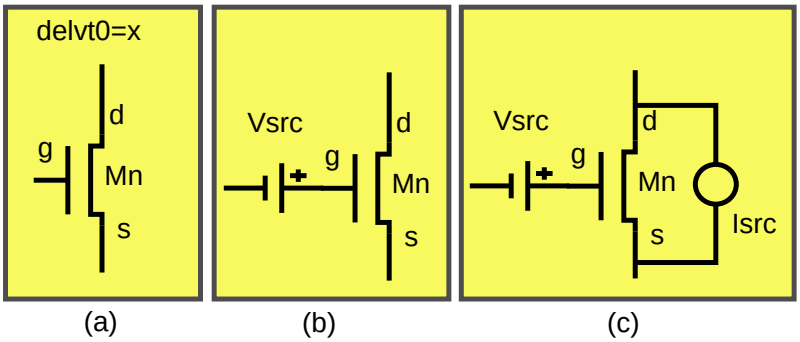


Figure 2.3: Compact variability modeling: (a) using simulator capability on instantiation; (b) voltage source modeling variations in Vt; (c) voltage source modeling variations in Vt and current source modeling variations in the current gain $\Delta\beta/\beta$.

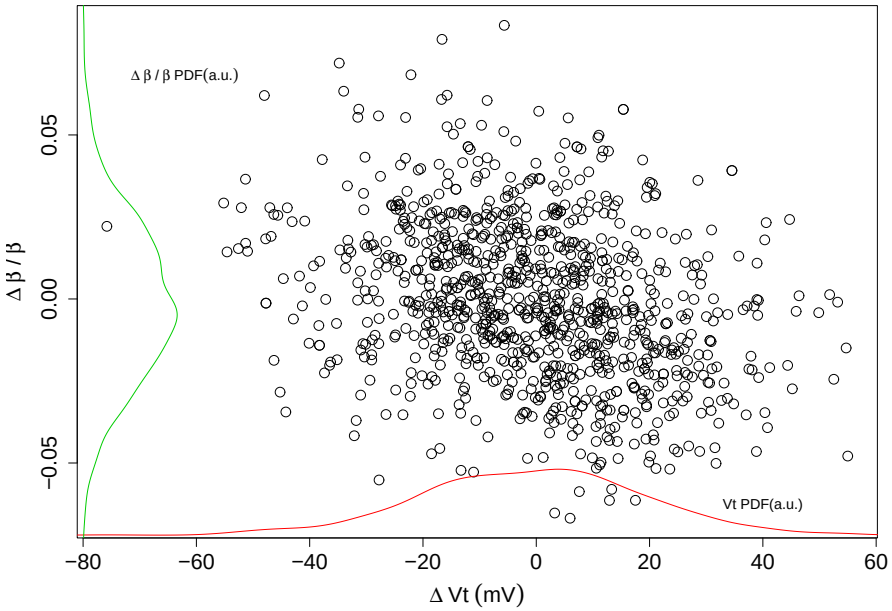


Figure 2.4: Compact variability model: extracted Vt and $\Delta\beta/\beta$ model the variations on the IV curve.

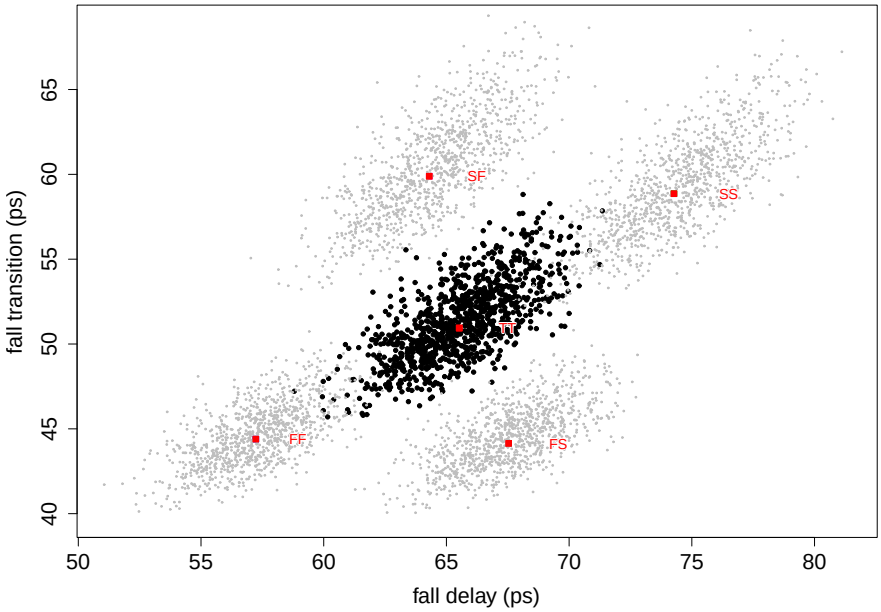


Figure 2.5: WD and D2D variations of fall delay and transition of a XOR2 gate.

2.3 Gate level

The propagation delay of a gate depends on the capability of its transistors to drive current. These changes in the transistor I-V curve cause the circuit metrics such as delay, static and leakage power and noise margin to shift. A smaller current driven by the transistor means larger propagation delay and lower dynamic power.

A variability transistor model is employed to simulate physical phenomena, e.g. variation in the number of dopant atoms, to the circuit-level metrics, e.g. delay and power, through electrical simulation. Electrical simulation is very time consuming and it is a common practice to break the design into small blocks, as for instance cell characterization in a standard cell flow. Special-purpose circuits, as for instance SRAMs and high-frequency clock networks, are also validated by running electric simulations.

Figure 2.5 reports Monte Carlo simulation results of the spread of fall delay and transition time of a commercial 32nm XOR2 gate due to within-die (WD) variations for each process corner (SS, FF, FS, SF). These corners (represented as large square dots) correspond to D2D variations. On top of D2D variations, fall and rise delays are significantly affected by WD variations.

2.4 Circuit (logic path) level

The last step of verification (before possible subsequent re-iterations) of the circuit is the timing analysis. At this level, the circuit is represented as data moving from one sequential element, e.g. FF or latch, to another. In a standard cell design flow the most employed technique is Static Timing Analysis (STA) (BHASKER; CHADHA, 2009).

STA takes as input 1) the extracted netlist containing cells and parasitics and 2) a pre-characterized cell library containing information on delay and power of each cell as a function of its slew and load. STA performs a simulation at logic-level employing graph traversal algorithms such as Critical Path Method (CPM), thus it is much faster than electric-level simulation. It then computes the critical paths and their slacks. Slack is the difference between the longest allowed time of a signal to propagate from the clock

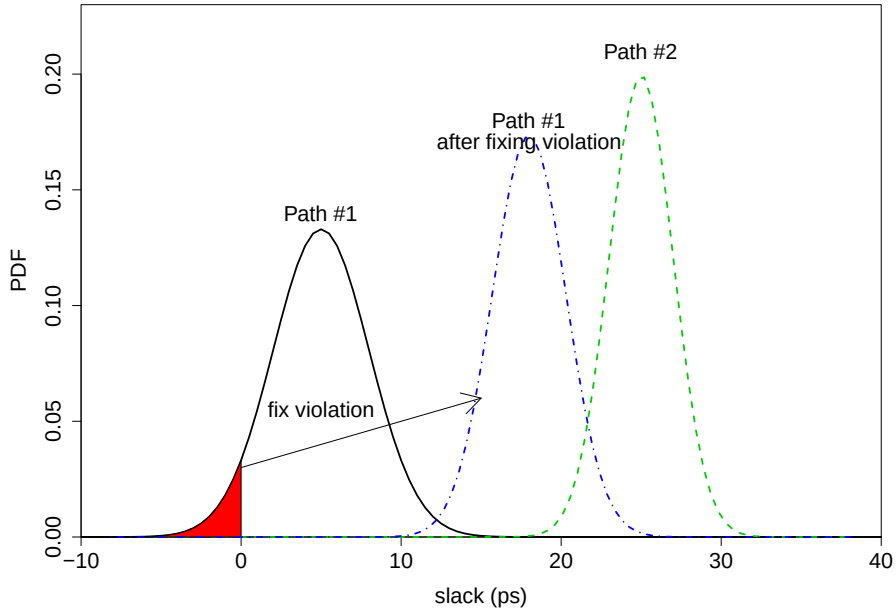


Figure 2.6: PDF of timing slack of two hypothetical paths reported by SSTA. Path #1 has negative slack and must be fixed for violation.

sink to the target FF (required time) and the actual computed delay (arrival time). Ideally all paths would have slack=0. Positive slacks mean that the overall delay of the path can be increased. This is done by re-sizing some gates in that path to smaller/slower ones in order to reduce area and mainly power due to their smaller capacitance. Negative slack results in a timing violation and the path must be made faster, what can be accomplished by using stronger gates and buffers. Ultimately, STA detects two kinds of failures:

hold time violation when the data signal arrives too quickly at the target FF. It happens when the clock signal arrives earlier in the source FF and the logic path is too fast. In this case the target FF can improperly store and propagate the quick data of the source FF in the same clock period, when it should propagate it only in the next one.

setup time violation when the data signal arrives too late at the target FF. It happens when the logic path is too slow and/or when the clock skew is such that the clock signal arrives much earlier in the target FF. In this case the target FF fails to receive the data in time, thus erroneously propagating the previous stored data.

Delay variations due to spatial and temporal variability may induce timing violations (failures), which are not reported by the nominal simulation. Typically STA used to employ the concept of corners to guarantee the circuit to operate even in the presence of process and environmental variations. In this scheme, the circuit is simulated in the fastest, slowest and typical set of paramenters with which it is expected to operate, considering all the transistor variability as being correlated. In sub-65nm technologies, where the WD variability has surpassed D2D variability, corner-based analysis is inaccurate. Thus in the last years much research has been promoted for Statistical Static Timing Analysis (SSTA), which models gate delays as random variables and can compute the statistical distribution of slacks.

Figure 2.6 represents the slack probability density function (PDF) of two hypothetical paths of a circuit. The slack of Path #1 has a high violation probability ($\approx 4.8\%$), imply-

ing as many as 4.8% of the fabricated circuits will present a violation in that path. An optimization step must be run in order to fix the critical paths. This optimization step is run until the circuit becomes timing clean and can be sent to tape-out. In this hyphotetical scenery the tools inserts buffer elements to make the path faster, thus reducing the failure probability of the path.

3 SPATIAL VARIABILITY (PROCESS-INDUCED)

Variability in the manufacturing process, mainly due to doping and lithography, causes variations on transistor threshold voltage, channel length and width. Although process variation incurs shifts in critical dimensions W and L, Tox, mobility and other physical or electrical characteristics of the transistor, these components can be reduced to a compact model where only Vt represents the variations in I-V curves. Thus, the total Vt fluctuation due to process variability is often modeled as a sum of independent sources of variations (CATHIGNOL et al., 2008; YE et al., 2008; LI; YU; CHEN, 2007):

$$\sigma^2_{Vt,total} = \sigma^2_{Vt,RDF} + \sigma^2_{Vt,LER} + ... \tag{3.1}$$

Nowadays the main sources of variation impacting the electric characteristic of transistors are Random Dopant Fluctuations (RDF) and Line Edge Roughness (LER) (CATHIGNOL et al., 2008).

3.1 Random Dopant Fluctuations

Advanced state-of-the-art process fabrication technologies nowadays employ effective channel lengths smaller than 30nm. In these technologies, the number of dopant atoms in the region where the inversion layer is formed is in the order of just tens of atoms (REID et al., 2008). Random Dopant Fluctuations (RDF) are the variations in the crystalline Si structure due to the variations in the number of dopant atoms in the channel, as well as due to their irregular distribution in the channel. RDF is represented in figure 3.1, which was extracted from Hane (2003a).

RDF nowadays represents one of the greatest challenges for the microelectronics industry. Recent works compare different sources of process variability in a 45nm technology node, including RDF, Line Edge Roughness (LER) and poly-gate granularity (PGG), and concluded that RDF is the dominant intrinsic source of statistical variability in MOSFET transistors (CATHIGNOL et al., 2008; YE et al., 2008).

Calibrating the implantation process in such a way to comply with the requirements to keep RDF under control is becoming more difficult with scaling. The challenges imposed by RDF can make it infeasible to keep the – necessary – trend of downscaling MOSFET transistors. The number of dopants are subject to a Poison distribution, and the uncertainty of the number of dopants is in the range of 5-10 % of the total number of dopants for a 50nm MOSFET (BERNSTEIN et al., 2006).

At the circuit level, RDF is modeled as a source of threshold voltage variation which affects each transistor independently of each other. RDF is an uncorrelated source of variability because assuming no systematic source of variation during implantation (which is

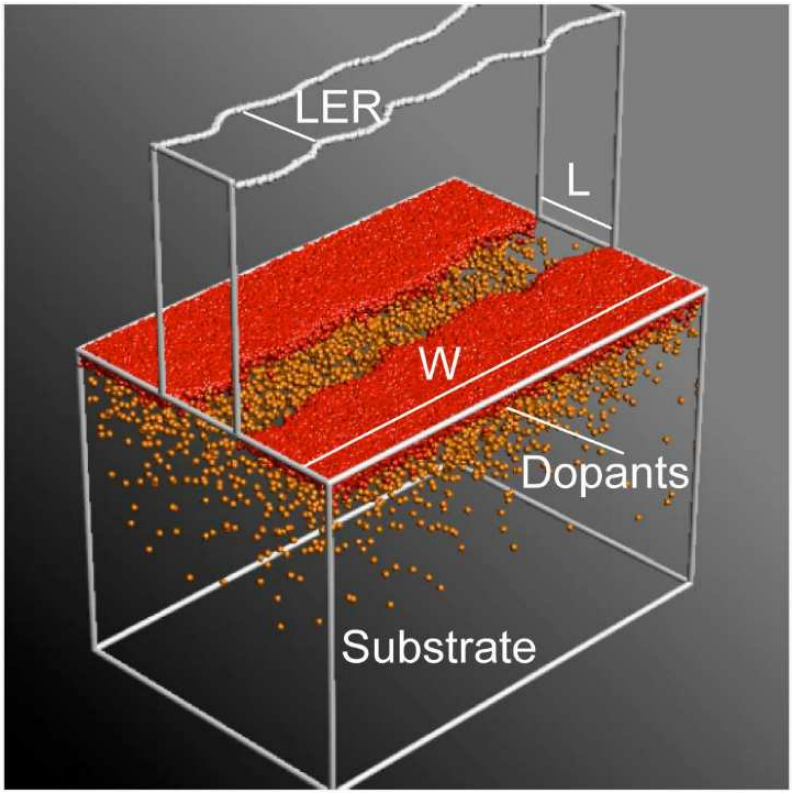


Figure 3.1: Random Dopant Fluctuations (Source: (HANE; IKEZAWA; EZAKI, 2003a))

the case) each transistor has a different number of dopant atoms, leading to a different mobility μ_0 and thus altering the I-V curve independently. The distribution of Vt due to RDF is widely accepted to follow a Normal distribution, and such Normality is demonstrated by Reid (2008). The 5-10% uncertainty in the number of dopants translate to a variation in Vt of $\sigma_{Vt} = 25 - 30mV$, as reported by Bernstein (2006) and Reid (2008).

Vt fluctuations due to RDF have been modeled by Monte Carlo simulations of 3D drift-diffusion (DD) models (STOLK; WIDDERSHOVEN; KLAASSEN, 1998). DD models have presented enough accuracy and agreement with measured data for device dimensions $> 100nm$. However, 3D simulations are computationally expensive and a simple formula to compute σ_{Vt} as a function of technology parameters is essential to the circuit designer. In this sense, 3D simulations are required only at the phase of device characterization, providing a simpler formulation to the designer. Since then, analytical expressions for σ_{Vt} could be derived by fitting the simulations data with formulas with physical support. An expression for σ_{Vt} is proposed by Stolk (1998) as:

$$\sigma_{Vt,RDF} \approx \left(\frac{\sqrt[4]{4q^3 \varepsilon_{Si} \varphi_B}}{\sqrt{3}} \right) \left[\frac{K_B T}{q} \frac{1}{\sqrt{4q\varepsilon_{Si}\varphi_B N_A}} + \frac{T_{ox}}{\varepsilon_{ox}} \right] \left(\frac{N_A^{0.25}}{\sqrt{WL}} \right) \tag{3.2}$$

where $q \approx 1.602 \times 10^{-19} Coulombs$ is the elementary charge, ε_{Si} and ε_{ox} are the permittivity of the silicon and the oxide respectively, N_A is the average channel doping density, φ_B is the built-in potential, T_{ox} is the oxide thickness, K_B is the Boltzmann constant, T is the absolute temperature, and finally W and L are the device dimensions.

Nowadays the most accurate method for modeling the Vt variations due to RDF is by time-consuming 3D atomistic models considering the quantum interaction at sub-atomic level. Asen Asenov's research group at Glasgow University is well-known for their ca-

pability of simulating RDF through atomistic simulation tools running in a cluster of workstations. Their framework is capable of running Monte Carlo simulations with sample sizes of 100,000 (REID et al., 2008). Asenov 2003 proposes a corrected model for Vt uncertainty, which is:

$$\sigma_{vt,RDF} = 3.19 \times 10^{-8} \Big(\frac{t_{ox} N_A^{0.4}}{\sqrt{L_{eff} W_{eff}}} \Big) \tag{3.3}$$

where L_{eff} and W_{eff} are the transistor effective channel length and width respectively, N_A is the average channel doping density and t_{ox} is the oxide thickness. This newer model is similar to the one proposed by equation 3.2, however the most important discovery is the relationship $\sigma_{Vt}^{RDF} \approx N_A^{0.4}$, which implies a much stronger dependence of σ_{Vt} on N_A then the previous model, which had an exponent 0.25 instead. The model has also the advantage of being a very simple expression due to the constant 3.19×10^{-8}, which still holds for recent technology nodes, for instance the 65nm IBM technology node as reported by Bernstein (2006).

3.2 Line Edge Roughness

Line Edge Roughness (LER) is the result of imperfections during the lithographic and etching processes, affecting the shape of the edges of the transistor critical dimensions (CD). LER causes channel width and length (specially and most critically for digital circuits is the channel length, which is usually the minimum dimension allowed by the technology) in such a way that the line edges are no more rectilinear as drawn in the layout. Instead, these lines become rough edged. LER is illustrated in figure 3.2, which presents a 3D transistor. In figure 3.2 the depletion channel is drawn in yellow, source and drain are red and substrate is blue. Notice the non-uniformity of the channel, which ideally should be a straight line.

LER is also referred to as Line Width Roughness (LWR). LWR refers to the width variation from one side of the rectangular shape to the other, while LER refers to the distance between the edges of one side with respect to the nominal line. Currently, ITRS estimates that for 32nm and 22nm half pitch technologies to be feasible, $3\sigma_{LWR} \approx 3-4nm$ must be reached (VAGLIO-PRET; GRONHEID; FOUBERT, 2010). This is still an open problem for the manufacturing industry and must be solved in the next years in order to keep pace with downscaling.

In the last decade many efforts have been made in order to model LER (HANE; IKEZAWA; EZAKI, 2003b; ASENOV; KAYA; BROWN, 2003; HYUN-WOO et al., 2004). LER can be divided into two components: 1) low-frequency LER, also called non-rectangular gate (NGR) and 2) high-frequency LER. Due to its statistical nature, LER causes variability in the I-V characteristic of the transistors. For digital circuits, LER can be modeled as a source of variation that impacts the transistor Vt, which can then model the variations caused in I_{on} and I_{off} (YE et al., 2008). Cao (2008) from Arizona State University employed 3D atomistic simulations to evaluate the impact of LER to σ_{Vt} and proposed an expression of Vt as:

$$\sigma_{Vt,LER}^2 = \frac{C_1}{e^{\frac{2L}{l'}}} \times \frac{W_c}{W} \times \sigma_L^2 \tag{3.4}$$

where C_1 and l' are technology related coefficients, W_c is the correlation length of NGR, W and L are the transistor width and length. The study shows that I_{on} (and thus Vt) has an

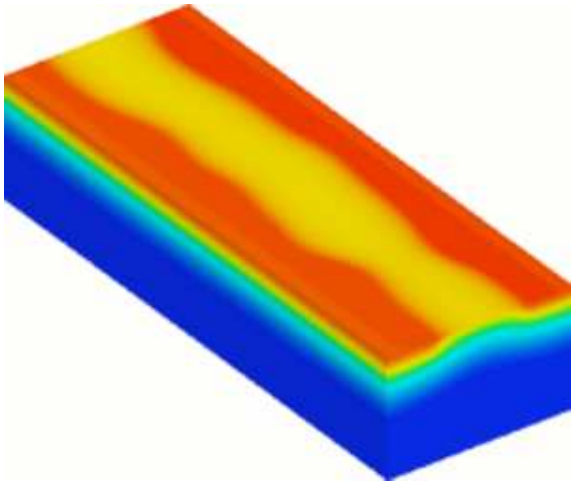


Figure 3.2: 3D Device simulation shows Line Edge Roughness in one transistor (Source: (ASENOV; KAYA; BROWN, 2003))

exponential dependence on gate length, and this is due to Drain-Induced Barrier Lowering (DIBL). This exponential relationship is an important finding because it predicts that LER is going to get much more exacerbated as L shrinks unless the technology constants C_1 and l' improve due to considerable process improvements.

Considering $\sigma_L = 2 - 3nm$, which is in accordance to ITRS for a 32nm technology node, the simulations performed by Ye (2008) of a 30nm device show $\sigma_{Vt,LER} \approx 10 - 20mV$.

3.3 Discussion

This chapter aims at presenting to the reader basic knowledge on sources of random process variations. The main sources of random variation affecting integrated circuits are: Random Dopant Fluctuations and Line Edge Roughness. They have in common the fact that they are not transitory nor time dependent. This characteristic differentiates them from time dependent issues, which is presented in the next chapter.

4 TIME-DEPENDENT VARIABILITY

The lifetime and reliability of digital circuits are being affected by the degradation of the electric characteristics of the transistors over time. The physical characteristics of the transistors suffer from significant degradations, causing changes in the electrical characteristics, specially the voltage threshold (Vt). The random switching between discrete levels of Vt over time is referred as Random Telegraph Signal (RTS). The systematic degradation of an electrical parameter over time is called *aging*. The main factors related to aging are *Negative Bias Temperature Instability* (NBTI) and *Hot Carrier Injection* (HCI).

4.1 Random Telegraph Signal (LF Noise)

Low Frequency (LF) noise is a performance limiting factor for deep sub-micron CMOS devices. In these devices, LF noise is dominated by multiple Random Telegraph Signals (RTS). This noise is due to succeeding electron capture and emission by a number of N_{tr} traps distributed according to a Poisson distribution at the S_i-S_iO_2 interface, as represented in figure 4.2. This phenomena causes oscillations in the transistor current I_{ds}, as represented by figure 4.1. Noise performance may strongly vary between different devices in the same chip, and moreover even between different operation points of a single transistor. Variability in Vt due to RTS has already been reported to be a problem for SRAMs and flash-memory (AGOSTINELLI et al., 2005; TEGA et al., 2006). Memory cells are usually the first ones to be affected by new sources of performance variability, because of their small dimensions, what is needed to achieve high integration density. With scaling, these sources of variability may also affect other circuits.

Until recent years, statistical models for RTS focused on the frequency domain. This is suitable for analog circuits, whose design and analysis are performed in the frequency domain. However, for digital circuits an appropriate time domain statistical analysis is needed, since these circuits are analyzed and designed using time domain metrics. Aiming at addressing this issue the work entitled *An appropriate model for the noise power spectrum produced by traps at the S_i-S_iO_2 interface: a study of the influence of a time-dependent Fermi-level*, by Roberto da Silva and Gilson I. Wirth, presents for the first time a comprehensive model for the RTS in time domain, deriving the relevant statistical parameters. This methodology for modeling RTS as a source of Vt variation is described and extended to consider the density of charges in the channel, as proposed by Gilson Wirth.

The variations in drain current can be modeled as transient changes in threshold voltage Vt. It is already well established that the variation in drain current due to RTS can be modeled as transient changes in gate bias (WIRTH et al., 2004; Wirth; da Silva; Bred-

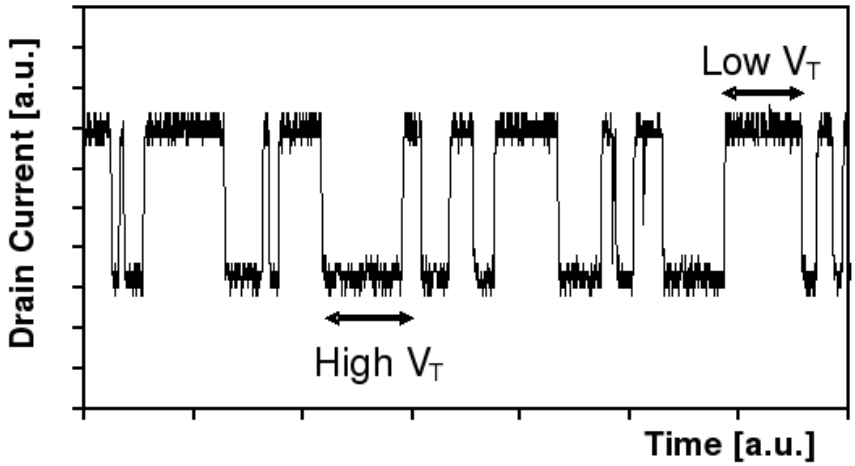


Figure 4.1: Time domain measurements of a stationary random telegraph signal (RTS). Discrete fluctuations are observed in the drain current. The $low-Vt$ state corresponds to the state where the trap is electrically neutral (empty). The $high-Vt$ state corresponds to the state where the trap is electrically charged. (Source: (SILVA; WIRTH; BREDERLOW, 2006))

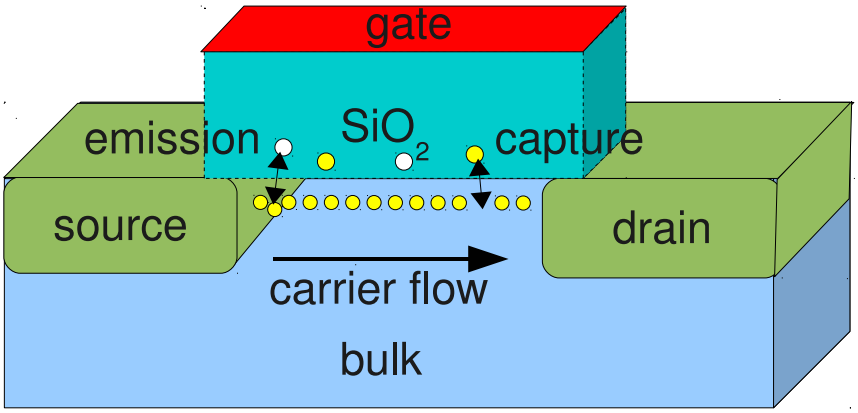


Figure 4.2: Trap-detrap of electrons at the Si-SiO2 interface.

erlow, 2007), induced by electron trapping and emission. The drain current fluctuation caused by the RTS from the $i-th$ trap may be expressed as $\Delta Ids_i = -gm\Delta Vt_i$ where gm is the trans-conductance (SONODA et al., 2007). This approach is adequate to model RTS as a dynamic source of Vt variation. In circuit analysis, this source of variation may be included as one more parameter that can cause circuit performance variability, in addition to the other sources, as for instance the static, time independent Vt variations caused by random dopant fluctuations (HANE; IKEZAWA; EZAKI, 2003b). The proper modeling of this effect becomes of increasing relevance, since it may lead to different results between subsequent measurements (or test) of the same circuit. This poses a challenge not only for the circuit designer, but also for the test engineer.

This section presents a comprehensive statistical study of RTS in time domain, and provides appropriate equations for circuit analysis and electrical simulation. These equations allow quantifying the impact of RTS on the reliability of MOS circuits at higher levels of the design.

The model presented in this section takes into account the position of the trap along the source-drain line of the transistor, as well as the distance of the trap from the inversion layer (position with respect to the Si-SiO_2 interface). Three different charge density models are proposed: constant, linear and exponential charge densities. This work reveals that in the case of the charge density being linear or exponential along the channel, the statistics of RTS noise is very different from the simple constant model. The reason to consider charge densities different from constant is that for large source-drain bias the channel charge density decreases from source to drain (TSIVIDIS, 2004), although for small source-drain bias the charge density is approximately constant along the channel.

Section 4.1.1 shows the methodology for computing the total ΔVt caused by all the traps in the transistor as a function of the impact of one trap: δvt_i. Section 4.1.2 presents the foundations to model the Vt shift due to one single trap, while section 4.1.3 shows a detailed formulation which takes into account charge density varying along channel.

4.1.1 Model derivation

The capture and emission of electrons at the interface trap may be modeled as a two-state fluctuation of the threshold voltage Vt. If the trap is empty we consider the Vt fluctuation to be zero. If an electron is trapped we consider the Vt fluctuation due to the *ith* trap to be equal to δvt_i. In this manuscript we express the threshold voltage fluctuation caused by one single trap i as δvt_i, while the total transistor threshold voltage fluctuation (caused by the combined effect of all traps) is ΔVt.

Trapping and releasing of an electron by a single trap is a Poisson process. The effect of δvt_i due to separate traps is additive (MACHLUP, 1954). In the worst case, at a given time all the traps found in a device may be occupied or empty, leading to large ΔVt. Hence, a statistical treatment of the problem is demanded. The relevant statistical parameters are hereby derived.

In the random telegraph signals, $s_i = 0,1$ denotes the state of the $i-th$ trap (0 =empty or 1 =occupied), the Fermi-Dirac statistics governs the probability of transition:

$$\begin{aligned} \Pr(s_i &= 0 \rightarrow s_i = 1)dt = \frac{dt}{10^{p_i}\left[1+\exp(-q_i)\right]} = \frac{dt}{\tau_c^{(i)}} \\ \Pr(s_i &= 1 \rightarrow s_i = 0)dt = \frac{dt}{10^{p_i}\left[1+\exp(q_i)\right]} = \frac{dt}{\tau_e^{(i)}} \end{aligned}$$

where τ_c and τ_e are the time constants of the Poisson process: the average times of emission and capture of the trap, respectively given by:

$$\tau_c^{(i)} = 10^{p_i}\left[1+\exp(-q_i)\right] \quad (4.1)$$

$$\tau_e^{(i)} = 10^{p_i}\left[1+\exp(q_i)\right] \quad (4.2)$$

The time constants are dependent on the transistor bias, which are expressed as a function of the Fermi-Level of the transistor as in:

$$q_i = \frac{(E_T^{(i)} - E_f)}{K_B T} \quad (4.3)$$

where $E_T^{(i)}$ is the energy within the band-gap of the $i-th$ trap, E_f is the Fermi-level energy, $K_B = 1.3806568 \times 10^{-23} J/K$ the Boltzmann constant and T is temperature.

At this point lies the essential difference between the static model presented in this section and the dynamic model developed later on section 4.3. Dynamic trap-detrap simulation is more complex and takes into account the bias of the transistor at each timestep of the simulation. That causes the time constants, as well as the probabilities of capture and emission, to vary over time. The simpler static model presented in this section, on the other hand, does not compute τ_c and τ_e using the proper equation 4.2. This static model assumes the voltage threshold fluctuation of a transistor as static and voltage-independent through the whole simulation: ΔVt is computed at the beginning of the simulation and is modeled as a static Vt fluctuation by the electrical simulation. The model assumes that $-Q < q_i < Q$ can be considered a uniform random variable and then $\tau_c^{(i)}$, $\tau_e^{(i)}$ are identically distributed, i.e., $\left\langle \tau_c^{(i)} \right\rangle = \langle \tau_c \rangle$ and $\left\langle \tau_e^{(i)} \right\rangle = \langle \tau_e \rangle$ for $i = 1,2,...,N_{tr}$. The input parameter Q corresponds to half of the band-gap width, which is around 2 eV in the case of Si (SILVA; BRUSAMARELLO; WIRTH, 2010).

Here, p_i is also a random uniform variable within an interval $p_{\min} < p_i < p_{\max}$ and in this case in the frequency domain, we can establish an important connection. The power spectrum density corresponding to the noise from the i-th trap is a Lorentzian function $S_i(f_i) = (A_i^2/f_i)\left[1+(f/f_i)^2\right]^{-1}$ where $f_i = 1/\tau_c^{(i)} + 1/\tau_e^{(i)}$ is the corner frequency corresponding to the trap and A_i is its amplitude (MACHLUP, 1954; SILVA; WIRTH; BREDERLOW, 2006; WIRTH et al., 2005; WIRTH; SILVA; BREDERLOW, 2007). It is possible to conclude that $f_i = 10^{-p_i}$ and due to this f_i is uniformly distributed in a $\log_{10}$ scale. That results in a probability distribution $h(f_i) = [\ln 10\ (p_{\max} - p_{\min})\ f_i]^{-1}$ for the corner frequencies (this assumption will be used from now on in this work) (KIRTON; UREN, 1989). From this approach, we can calculate

$$\begin{aligned} \Pr(s_i(t)=0) &= \frac{\tau_e}{\tau_e+\tau_c} \\ \Pr(s_i(t)=1) &= 1-\Pr(s_i(t)=0) = \frac{\tau_c}{\tau_c+\tau_e} \end{aligned}$$

where $Pr(s_i(t)=1)$ is the probability of the *ith* trap being occupied (i.e., the RTS being in the "1" state), and $Pr(s_i(t)=0)$ is the probability of the *ith* trap being empty (i.e., the RTS being in the "0" state).

Thus the threshold voltage fluctuation ΔVt which models the current fluctuation of the transistor at time t due to all the traps is computed by

$$\Delta Vt(t) = \sum_{i=0}^{N_{tr}} \delta vt_i * s_i(t) \tag{4.4}$$

where Ntr is the number of traps and δvt_i for $i = \{1,\ldots,Ntr\}$ is the instantaneous voltage threshold fluctuation when trap i is occupied. The amplitudes δvt_i are random variables and our results will be dependent on its first and second moments, respectively $\langle \delta \rangle$ and $\langle \delta^2 \rangle$. Those δvt_i can be obtained by experimental measurements of $\frac{\delta I}{I_{ds}}$. Although there is a lack in the Literature for accurate modeling of the current fluctuation due to one single trap in deep sub-micron technologies (DSM) technologies, the next section presents a well established model for computing $\frac{\delta I}{I_{ds}}$, which can be used as an approximation.

4.1.2 Computing δvt_i

When one electron is captured by the trap located in the SiO_2 the number of charge carriers in the channel is affected and the total I_{ds} current will decrease because of the loss

of the trapped electron. On the other hand, the subsequent emission of the electron will cause an increase in the I_{ds} current. The fluctuation on the total current flowing through the transistor channel due to one single trap is (SIMOEN et al., 1992):

$$\frac{\delta I_i}{I_{ds}} = \frac{g_m}{I_{ds}} \cdot \frac{q}{W_{eff} \cdot L_{eff} \cdot Cox}$$

where $\frac{\delta I_i}{I_{ds}}$ is unit-less, g_m is the trans-conductance in AV^{-1} , I_{ds} is the source-to-drain current in A, W_{eff} and L_{eff} are the effective transistor dimensions in m, Cox is oxide capacitance in F/m^2, q is the elementary charge given by $1.602 \times 10^{-19} C$. Since $\delta I_i = -g_m \delta V_i$, we arrive at the suitable formulation for the Vt instantaneous fluctuation due to one single trap:

$$\delta vt_i = \frac{q}{W_{eff} \cdot L_{eff} \cdot Cox} \tag{4.5}$$

where δvt_i is in V.

Based on this expression, it is possible obtain the analytical expression of the voltage fluctuation as a function also of the location of the trap in the oxide (GHIBAUDO; BOUTCHACHA, 2002):

$$\delta vt_i = \frac{q}{W_{eff} \cdot L_{eff} \cdot Cox} \cdot \left(1 - \frac{xt_i}{tox}\right) \tag{4.6}$$

where $0 \leq xt_i \leq tox$, xt_i is the location of trap i (how deep it is in the oxide thickness) and tox the oxide thickness. From this expression, we see that traps located closer to the Si-SiO_2 interface affect the threshold voltage more than traps far from the interface. In this work we assume xt_i to be uniformly distributed in the interval $[0,\ tox]$, what is in accordance to Chadwin (2009). However this assumption does not imply any loss of generality to the RTS model proposed in this manuscript if another distribution is experimentally observed.

4.1.3 Non-Uniform charge density

Section 4.1.2 presented a first principle model that may be a good approximation for the statistics of the current fluctuation caused by a trap if the transistor is operated with small source-drain bias. In this case the inversion carrier density is approximately uniform along the channel. If the transistor is operated with large source-drain bias the charge density will not be uniform along the channel. For large source- drain bias the charge density may be a strongly non-linear function of the position along the channel (TSIVIDIS, 2004). In this section we present a more detailed model for dI/Ids and subsequently δvt_i that takes into account the charge density and models it as a function of the location of the trap position along the channel (along the source to drain direction). It is known that if the charge density is not uniform along the channel the amplitude of the current fluctuation caused by the trap depends on position in the channel (ALEXANDER et al., 2005; LEYRIS et al., 2007; VASILESKA; KHAN; AHMED, 2005).

Figure 4.3 shows three assumptions that can be made regarding the charge density varying along the transistor channel of deep sub-micron length: (a) charge density is constantly distributed along the channel length axis, (b) charge density is larger at the source and it decreases linearly along the channel and (c) charge density is larger at the source and decreases exponentially along the channel (as an example of a strongly non-linear dependence of carrier density on channel position).

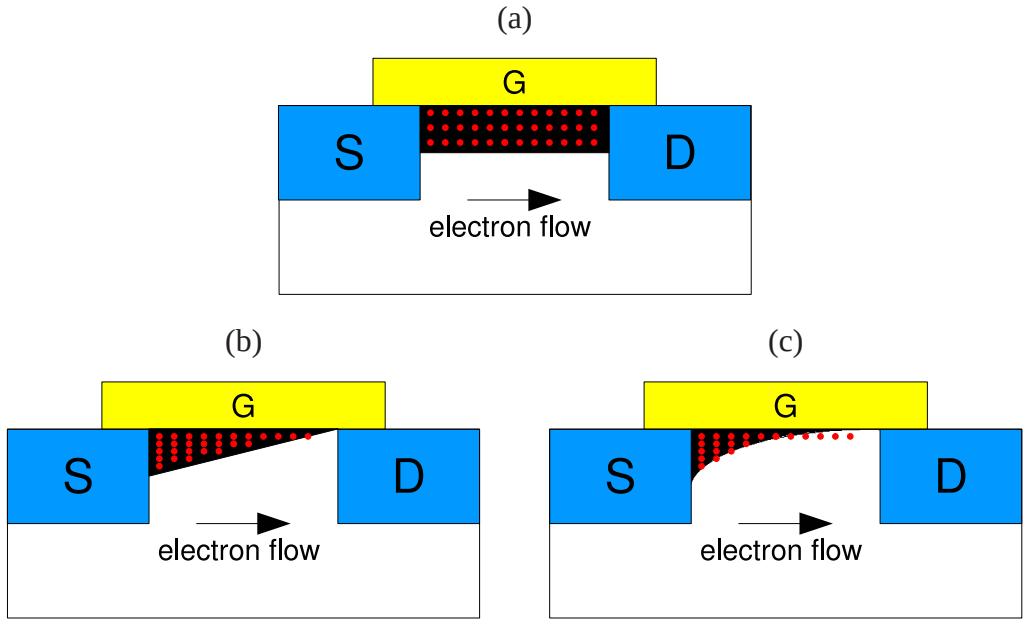


Figure 4.3: (a) Charge density constant along the channel; (b) charge density decreases linearly from source to drain; (c) charge density decreases exponentially from source to drain.

These three scenarios can be modeled by multiplying equation 4.6 by $\alpha(xl_i)$ which is a function of the location xl_i of the trap along the length axis as in:

$$\alpha(xl_i) = \begin{cases} k_c & constant \\ k_l \cdot \frac{xl_i}{L_{eff}} + c_l & linear \\ exp\left(k_e \cdot \frac{xl_i}{L_{eff}}\right) + c_e & exponential \end{cases} \tag{4.7}$$

where $0 \leq xl_i < L_{eff}$ is the location of the trap $i = 1, \ldots, Ntr$ in relation to the channel length, k_c, k_l, k_e, c_l and c_e are fitting constants. The trap location xl_i can be modeled as a random variable following a given distribution which can be determined experimentally. In this work it is modeled as uniformly distributed along the channel length, which is in agreement to the experimental findings of Saks (1990).The threshold voltage fluctuation caused by the occupation of one trap considering the location of the trap in the channel then becomes:

$$\delta vt_i = \alpha(xl_i) \cdot \frac{q}{W_{eff} \cdot L_{eff} \cdot Cox} \cdot \left(1 - \frac{xt_i}{tox}\right) \tag{4.8}$$

In 2010, as a result of a scientific cooperation with Arizona State University, we have been able to obtain atomic-level simulation data of the fluctuation of Vt caused by trapped charges as a function of their position along the transistor channel . These 3D atomistic simulations, described in depth by Camargo (2010), were performed by Nabil Ashraf and Dragica Vasileska. Figure 4.4 shows the average threshold voltage variation in relation to the trap position for 20 devices with different random dopant distributions. The source of the channel is at x = 0. The figure shows a clear trend of the impact of a trap to Vt being inversely proportional to its distance along the channel. No conclusion can be drawn whether the best fit is a linear fit or an exponential fit. Since their $adjusted - R^2$ are respectively 0.61 and 0.56, for this sample the linear fit can be considered slightly better

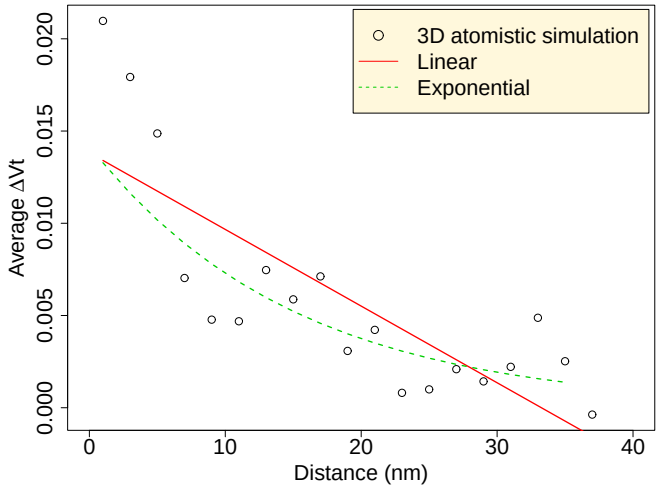


Figure 4.4: Threshold voltage variation due to traps located at the semiconductor/oxide interface and different positions along the middle section of the channel.

than the exponential fit. However the sample size is too small to draw a final conclusion. It is visually shown by Camargo (2010) that a 3rd-order degree polynomial presents a good fit.

The most important fact is that quantitatively these experimental data confirm the theoretical assumption we have proposed in (BRUSAMARELLO; WIRTH; SILVA, 2009): the impact of a trapped charge on Vt depends on the position of the trap along the channel. The model proposed in Brusamarello (2009) was the first RTS model to take this effect into account.

4.1.4 Simulations

We have computed the distributions of ΔVt of the transistors of a SRAM cell using the three proposed dependencies of the trap position along the gate length on current: constant, linear and exponential. Simulations were performed on the PTM 65nm technology node (CAO; MCANDREW, 2007). We assume the number of traps Ntr in the interface to follow a Poisson with rate $\lambda = 3$. This value is extracted from table I of Wirth (2005), properly scaling the experimental data of that work to the PTM 65nm technology node.

Figure 4.5 shows (y-axis is in logarithmic scale) the probability density function (PDF) plots (Kernel Density using a Gaussian and bandwidth=0.5) of 10,000 Monte Carlo (MC) simulations of the ΔVt of one NMOSFET transistor of a SRAM cell by using the proposed model. The parameters values of the transistor are Leff=24.5nm and Weff=80nm, extracted from PTM 65nm technology node and in accordance to Cao (2007). The curve corresponding to no dependence either on xl or xt is using equation 4.5, while the others use equation 4.8 (where the dependence on xl and xt are taken into account). We used constants to fit our results with the MC simulations of the current fluctuation due to one single trap performed by Alexander (2005). In the constant case $k_c = 3$, in the linear $k_l = 6$ and $c_l = 0$, and in the exponential formulation $k_e = 2.198$ and $c_e = 0$.

In all cases there is a peak near 0: nearly 20% of the transistors have $\Delta_{Vt} = 0$. This case means that either all the traps of the transistor are empty or the transistor has no traps at all (the number of traps follow a Poisson distribution).In this case the current fluctuation must be 0. However, the probability of RTS causing a Vt shift greater than 20mV can be more than 20%. The maximum number of traps found in our simulation was 6, while

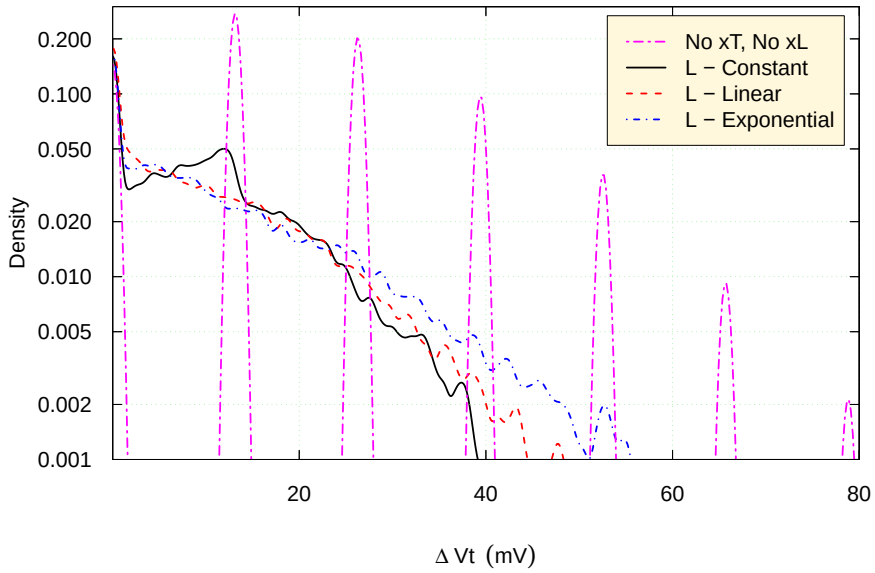


Figure 4.5: Distribution of ΔVt of one transistor caused by RTS, considering the three dependencies on the trap position (y-axis in logarithmic scale).

the minimum was 0 and the average is 3. The worst case occurs when all the traps are occupied and the fluctuation can be up to 50mV (which can happen at a rate of 10^{-4}).

The PDF of shifts in Vt does not follow a Normal distribution. The case where no length and tox dependence are considered, $\overline{\Delta Vt} = 20.4mV$ and $\sigma_{\Delta Vt} = 15.8mV$; constant length dependence has $\overline{\Delta Vt} = 10.2mV$ and $\sigma_{\Delta Vt} = 9.2mV$; linear length dependence presents $\overline{\Delta Vt} = 10.2mV$ and $\sigma_{\Delta Vt} = 10.6mV$; exponential length dependence has $\overline{\Delta Vt} = 12.1mV$ and $\sigma_{\Delta Vt} = 12.9mV$. Refer to da Silva (2011) for a detailed study on distributions of particle retention time phenomena.

Notice that the case which considers no dependence on the trap position is the worst case scenario because δvt_i is always maximum and then ΔVt is dependent only on the number of traps being occupied. The distribution in this case is discrete and follows a Binomial Distribution, which in this limit is a Poisson Distribution.

4.2 Negative Bias Temperature Instability

Bias Temperature Instability (BTI) is a physical phenomenon related to the generation and/or activation of states in the the interface between silicon and silicon oxide, and trapped charges in the oxide. The mechanism is accelerated by temperature and voltage bias, regardless of current flow (GRASSER et al., 2009).

Because of BTI the electrical characteristics of the transistor shift over time. It causes the absolute decrease of the *On* current Ion and transconductance g_m, while causing the increase of absolute values of the Off current I_{off} and threshold voltage Vt. The effect of increased V_T is equivalent as applying a voltage offset to Vg.

The most interesting characteristic of BTI is its dual-stage mechanism: stress and recovery, as represented in the scheme of figure 4.6. The device is under stress when voltage Vgs is applied to the gate of the transistor over a period of time and its Vt increases (degrades). However when the stress voltage is removed the devices goes to the recovery phase: its Vt partially recovers to the level prior to stress.

In reality, a device is constantly switching between stress and recovery. Because BTI

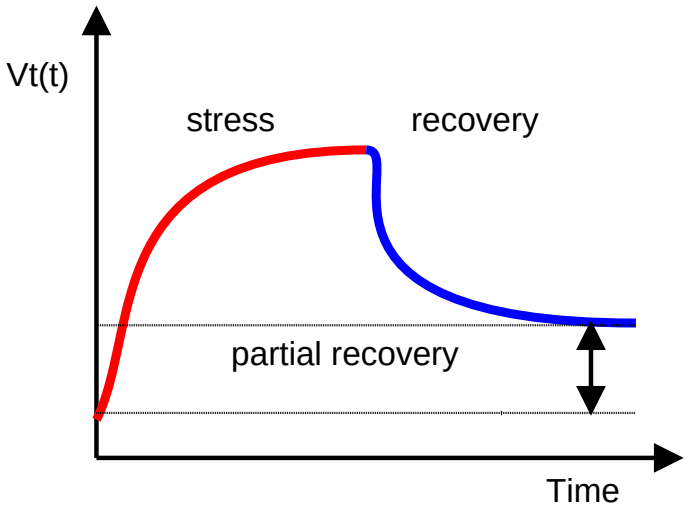


Figure 4.6: The two stages of NBTI: stress when device is biased and recovery. The transistor does not fully recover.

exhibits this complex stress and recovery behavior during dynamic circuit operation, the amount of degradation depends on the stress history of the transistor. This history is represented within the concept of duty cycle, which is the ratio between the device operating in stress and relaxation. Devices in arithmetic and memory circuits tend to present unbalanced duty cycle, while devices on clock circuitry is an example of a duty cycle of 50%.

The BTI effect is observed in both NMOS and PMOS field effect transistor devices, and both are susceptible to Positive Bias Temperature Instability (PBTI) and Negative Bias Temperature Instability (NBTI). Huard (2006) have studied the impact of BTI in four scenarios: 1) NMOSFET biased with Vg>0, 2) NMOSFET biased with Vg < 0, 3) PMOSFET biased with Vg > 0 and 4) PMOSFET biased with Vg < 0. The work clearly demonstrates that PMOSFETs are more susceptible to BTI, regardless of positive or negative bias. The PMOSFET with $Vg < 0$, or NBTI in P-type MOSFETs, is the case that presents the largest Vt shift. This is unfortunate, since in digital circuits PMOSFETs are negatively biased. This is the reason why BTI is often referred to as NBTI and attributed to cause Vt shifts in P-type MOS devices only.

Thus, from the circuit designer perspective, BTI causes the Ids-Vds curve of the transistor to systematically shift over time. As described in chapter 2, these variations in the I-V curve over time can be accurately modeled by a compact model assuming Vt as the only parameter dynamically increasing over time. These degradations can result in speed degradation of the logic cells over time.

In the last decade, accurately modeling NBTI has become a major concern for industry. The systematic degradation of the transistor over time potentially means circuit failures in the field that could not be detected by current test methodologies. There are many theoretical and experimental analysis of NBTI in the Literature, and until very recently the most accepted theory was the reaction-diffusion NBTI model. These models assume the generation of traps in the Si-SiO2 interface when bias is applied at the gate and subsequent annealing of these traps when the stress is removed. These reaction-diffusion models have been successful and widely employed by industry to predict safe guard-bands given by the maximum voltage threshold degradation the transistor could

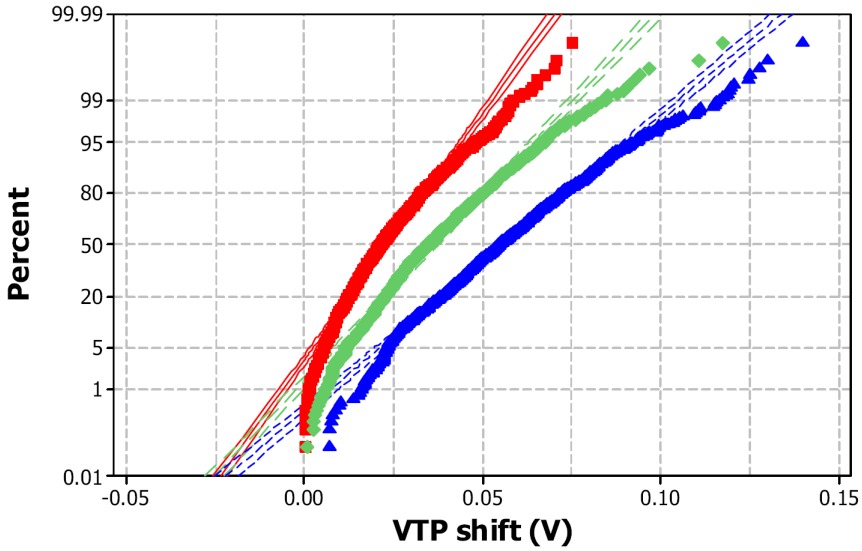


Figure 4.7: Distribution of NBTI V_T shift (colors stand for 3 different levels of NBTI stress) varies over approximately 100 mV in SRAM-sized pFETs (Plot is a courtesy of Ben Kaczer, source (HUARD et al., 2008)).

present after a number of years. When this thesis project started back in 2008, our first works on NBTI consisted of using a reaction-diffusion based formulation to compute the Vt degradation of the transistor, and then simulate the impact of NBTI on small circuits. Thus, the reaction diffusion model is firstly discussed in section 4.2.1.

However, reaction-diffusion models have failed to agree with experimental measurements. When the bias is removed from the device, there are two stages of recovery: a fast recovery component and a slow recovery component. The reaction-diffusion model can only predict the slow recovery but cannot explain or model why there is a fast recover, which occurs right after the stress is removed. The fast recovery component is very important because during normal circuit operation the transistor is often switching between stress and recovery. Recent experimental data from IMEC suggest there might be a relationship between NBTI and RTS (KACZER et al., 2011). Following a cooperation with IMEC, our group at UFRGS has been making significant progress on modeling NBTI as a trap-detrap phenomena, similarly to RTS. Section 4.2.2 presents a recent study on the component of NBTI caused by emission and capture of electrons by traps.

The figure 4.7, extracted from (HUARD et al., 2008), presents the results of measurements of nano-scale devices under different stress conditions (HUARD et al., 2008). This plot shows sample distributions of Vt shifts caused by NBTI as probit plots. The x-axis represents the amount of Vt shift in V, while the y-axis is in probit scale and shows the probability of that value within the sample, i.e. the average of the sample is the point in the x-axis projecting to 50% in the y-axis. In a probit-plot a Normal distribution is a straight line. For all the measured conditions (different stresses), V_T shift of the transistor is not Normal . V_T shift caused by NBTI varies from almost zero in some pFETs to approximately 150 mV in other devices, depending on the stress condition.

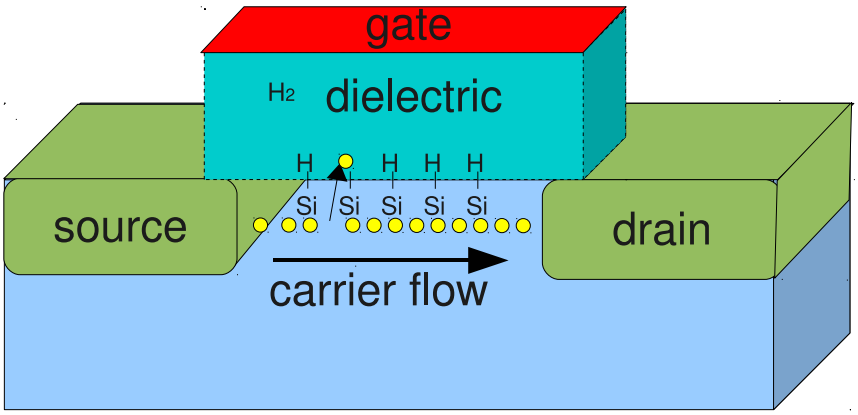


Figure 4.8: The reaction-diffusion NBTI model proposes that during stress holes are trapped in the SiO2-Si interface due to the break of the hydrogen-silicon bonds at the interface.

4.2.1 Reaction-Diffusion Model

The reaction-diffusion NBTI model proposes that during stress holes are trapped at the SiO2-Si interface due to the break of the bonds between hydrogen and silicon atoms at the interface, as represented by figure 4.8 . When the stress voltage is removed from the gate, the Hydrogen slowly returns to the Si-SiO2 interface by diffusion, the bonds are restored and the driving strength of the transistor is recovered (WIRTH; SILVA; KACZER, 2011). The electric field, temperature and concentration of holes influence the process.

Different devices degrade at different speeds. As a result, not only the mean value of the threshold voltage varies over time, but the standard deviation (variability) of V_t between devices also increases. Statistically, this effect can be expressed in terms of an average and standard deviation of threshold voltage depending on time and duty cycle (the stress/recovery ratio). The equation according to the reaction-diffusion model for the Vt degradation as a function of time t is given by Vattikonda (2006):

$$\mu_{\Delta Vt}(t) = \frac{q \times N_{IT}}{C_{ox}} \tag{4.9}$$

where $C_{ox} = \varepsilon_{ox}/t_{ox}$ is the oxide capacitance, ε_{ox} and t_{ox} are the oxide permittivity and thickness respectively. One of the most important parameters to define the fluctuation of Vt due to NBTI in the reaction-diffusion model is the number of interface traps, defined as N_{IT}. The number of interface traps can be defined as a static model or a dynamic model. The static model is simpler and gives an upper bound for NBTI, and can be computed as

$$N_{IT,static}(t) = (K^2 \times t^{\frac{1}{2}} + c^{\frac{1}{2n}})^{2n} \tag{4.10}$$

where t is time in seconds, c is the initial number of interface traps, the constant n is the coefficient of diffusion, which is related to the fabrication process and must be fitted experimentally . According to Mahapatra (15-19 April 2007), if the diffusion species is H_2 then $n \approx 0.16$ However if the diffusion species is H, then $n \approx 0.25$. Also, K is the generation rate of N_{it} and can be computed as:

$$K \approx \sqrt{C_{ox}(V_{gs} - V_{th})} \times e^{\frac{E_{ox}}{E_0}} \times e^{-\frac{E_a}{kT}} \tag{4.11}$$

where C_{ox} is the oxide capacitance, $Eox = \frac{(V_{gs}-V_{th})}{t_{ox}}$ is the electric field at the oxide, k is the Boltzmann constant, E_a and E_o can be obtained by fitting. According to Rakesh (2006), $E_0 \approx 1.9MV/cm$ and $E_a \approx 0.12eV$. Thus, a simplified form of K is proposed by Rakesh (2006) as:

$$K \approx 1 - \frac{V_{ds}}{\alpha(V_{gs} - V_{th})} \tag{4.12}$$

where $\alpha \approx 1.3$ for a $0.25\mu m$ technology node. The static model is suitable for circuit characterization because it does not consider the two states of trap generation: stress and recovery. Thus, it gives an upper bound to the effect of NBTI because it does not fully consider its dynamic mechanism. A more accurate model is dependent on the state of V_{gs}. When $V_{gs} = VDD$ the number of interface traps reduces in such a way that if the transistor stays in that state for some time most of its current strength can be recovered. A more accurate formulation for the number of interface traps is then proposed by Rakesh (2006) as:

$$N_{IT,dynamic}(t) = \begin{cases} \sqrt{K^2(t-t_0)^{1/2} + N_{IT0}^2} + \delta; & V_{gs} \neq V_t; \quad \text{stress} \\ (N_{IT0} - \delta)(1 - \sqrt{\eta \frac{(t-t_0)}{t}}); & V_{gs} \approx V_t; \quad \text{recovery} \end{cases} \tag{4.13}$$

where η and δ are constants of proportionality that must be obtained by fitting experimental measurements. For a technology node with minimal critical dimention of $0.13\mu m$ Rakesh (2006) proposes $\eta = 0.35$. This dynamic formulation is more accurate than the static one, and shows better agreement with the experimental results of Rakesh (2006). However the static approach is simpler to implement and can be more valuable to many simulation applications where the designer does not have fine-grain control on the voltages being applied, as for instance cell characterization. Moreover, the dynamic approach is more accurate than the static one only if the circuit is simulated over a long period of time.

Correct modeling of NBTI is very difficult mainly due to the fact that both the theoretical mechanism of generation and activation of traps and the frameworks for measuring NBTI are not a consensus in the scientific community. In the next few years the correct modeling of NBTI must evolve to a consensus so that its effect to next technology nodes can be modeled. Thus, the Vt fluctuation due to NBTI can be modeled as a random variable to correct for modeling and measurements discrepancies, as well as capture the statistical nature of the phenomena. Kang (2007) proposes to assume Vt shift as a random Normal variable, where its variance as a function of the time t is given by:

$$\sigma^2_{Vt_{NBTI}}(t) = \sigma^2_{NIT} \left(\frac{q}{C_{ox}}\right)^2 = \frac{q \times T_{ox} \times \mu_{\Delta V_t}(t)}{\varepsilon_{ox} \times A_G} \tag{4.14}$$

where A_G is the area of the device.

Then, the Vt of the transistor at a given time can be computed by assuming Vt as a random variable following a Normal distribution with mean given by expression 4.9 and standard deviation given by equation 4.14, as in:

$$Vt(t) = Vt_0 + N(\mu_{\Delta Vt,NBTI}(t), \sigma_{\Delta Vt,NBTI}(t)) \tag{4.15}$$

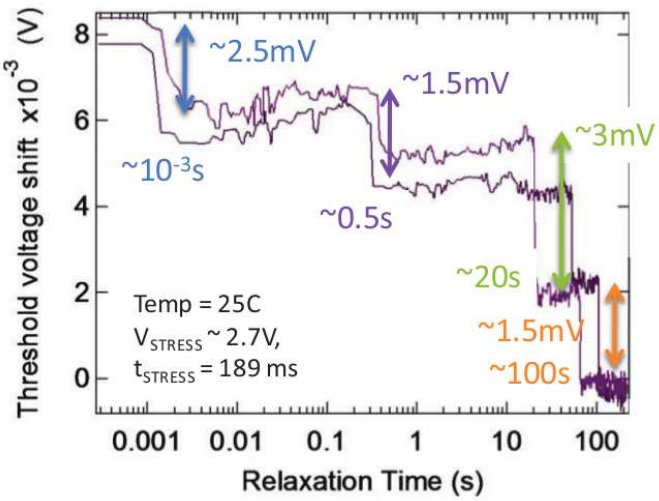


Figure 4.9: Measurements of a $70 \times 90\ nm^2$ NMOS device from (KACZER et al., 2011).

Such simple compact model can then be employed for simulating the impact of NBTI, as well as of another source of variation, in digital circuits. The difference between this variation and spatial sources is its dependence on time and duty cycle.

4.2.2 Charge Trapping Component of BTI

Recently Wirth (2011) presented a theoretical analysis, Monte Carlo simulations and experimental investigation of the charge trapping component of BTI. The model presents a novel analytical model for both stress and recovery phases of BTI. The new charge trap-detrapping BTI model explains BTI as a series of emission and captures of electrons. The theory does not make physical assumptions regarding the generation of new traps, as done by the reaction-diffusion model. There may or may not exist a mechanism of generation of new traps in the interface or oxide. Still, one portion of the traps contributing to BTI are traps that could be as well described as contributing to RTS, but the following aspects should be regarded:

- the traps causing BTI show a difference of many orders of magnitude with respect to their probabilities of capture and emission;
- the traps contributing to RTS noise have probabilities of capture and emission in the same order of magnitude.

Interestingly, recent experimental works based on device measurements suggest this relationship between the fast recovery component of NBTI and RTS (KACZER et al., 2009, 2011). Fig. 4.9 illustrates the fast recovery of NBTI after stress removal. During NBTI stress, traps are occupied. After the removal of the stress, the system relaxes towards a steady-state RTS through a series of individual displacements. For this device, four discrete displacements are clearly visible, meaning four traps have been occupied during stress.

Similarly to RTS, the total Vt fluctuation of a transistor due to the combined effect of all the traps at a time instant t is given by:

$$\Delta Vt(t) = \sum_{i=0}^{N_{tr}} \delta vt_i * s_i(t)$$

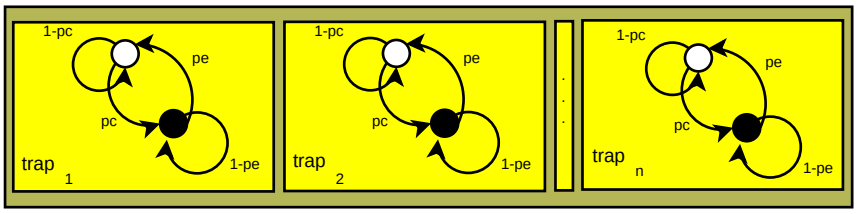


Figure 4.10: Scheme of the Markov Chain process of emission of capture of traps in a transistor

where N_{tr} is the number of traps given by a Poisson distribution with mean λ_{Ntr} as in $N_{tr} = Poisson(\lambda_{Ntr})$. The mean of the Poisson (λ_{Ntr}) is a function of (1) the interface technology, for instance transistors manufactured with high-k materials have more traps than those employing SiO_2, and (2) the gate area, since larger transistors have proportionally more traps.

The figure 4.10 is a representation of the trap-detrap mechanism (SILVA; LAMB; WIRTH, 2011). Each transistor contains a number of traps, which at a given instant of time can be occupied or empty. Each trap is a stochastic process over time known as a Markov Chain, where the probability of a state transition is governed by a given statistics. Depending on its current state, each trap has a probability of capturing or emitting a electron (*pe* and *pc*). These probabilities are computed as (da Silva; Wirth, 2010):

$$\begin{aligned} pc = \Pr(\sigma_i(t) = 0 \rightarrow 1) &= \frac{\Delta t}{\tau_c} \\ pe = \Pr(\sigma_i(t) = 1 \rightarrow 0) &= \frac{\Delta t}{\tau_e} \end{aligned} \tag{4.16}$$

where Δt is the time interval, τ_e and τ_c are the average emission and capture times. As an example, if a trap has $\tau_e = 1ns$ and $\tau_c = 2ns$, that trap will capture one electron every 2ns and emit it in approximately 1ns on average. The average emission and capture times τ_e and τ_c are given by:

$$\begin{aligned} \tau_e &= 10^{-pi}(1 + exp(\frac{E_f - E_i}{K_b T}) \\ \tau_c &= 10^{pi}(1 + exp(\frac{E_i - E_f}{K_b T}) \end{aligned} \tag{4.17}$$

where for a given trap i, K_b is Boltzmann constant, T is the device temperature in Kelvin, E_f is the Fermi-level of the transistor, E_i is the energy level of the trap i, and p_i is the time constant of that trap. The traps have energy levels within the forbidden band-gap and the distribution of their energy follows an U-shape distribution (WIRTH et al., 2009; WIRTH; SILVA; KACZER, 2011), as discussed in section 4.3.2. The Fermi-level depends on the voltage at the transistor terminals and is accurately computed in our model as explained in section 4.3.1.

4.3 Time-dependent trap-detrap simulation

In the time domain, capture and emission of charge carriers by traps in the transistor silicon oxide and in the $Si - Si0_2$ interface cause fluctuations of the current of the transistor

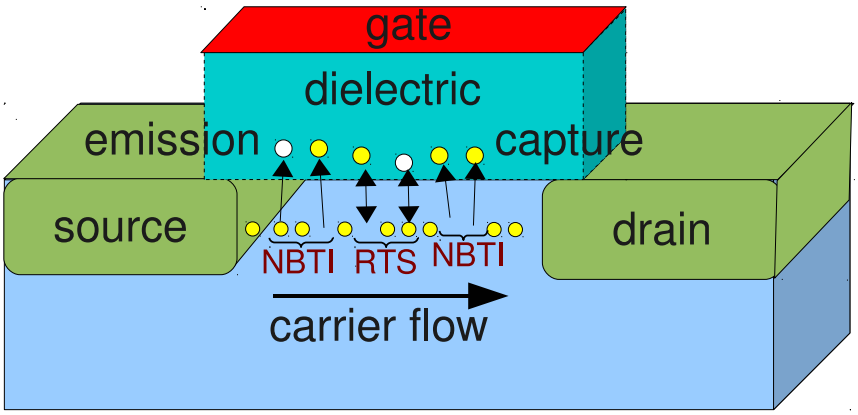


Figure 4.11: Trap-detrap of charges at the Si-SiO2 interface due to RTS and NBTI. Traps contributing to RTS show similar probabilities of capture and emission, while traps contributing to NBTI may have emission and capture times differing by many orders or magnitude.

over time, even when Vgs and Vds are constant over time. The amplitudes of these fluctuations are discrete: when trap i captures a charge carrier, the current Ids decreases by ΔIds_i. The state of all the many traps in the interface add up to the total current fluctuation at a given instant of time.

This section explains the implementation, made possible by modifying the BSIM4 source code, and the results of the time-dependent trap-detrap simulation methodology proposed by Wirth (2011). This new model unifies BTI and RTS as similar sources of Vt fluctuation varying over time. These fluctuations are function of the occupation level of the interface states and oxide traps causing the phenomena.

The scheme of figure 4.11 represents the similarity and difference between RTS and BTI in the model. Both RTS and BTI are caused by traps on the interface and dielectric of the transistor. Traps contributing to RTS emit and capture charges at similar rates. On average the trap is at an occupied state the same amount of time it is at an empty state, and this behavior impacts the standard deviation of Vt, while the average Vt is constant over time. The experimentally relevant average capture and emission times of traps causing RTS is typically in the order of seconds to pico-seconds.

On the other hand, NBTI is caused by traps with very unbalanced average time of capture and emission. When the transistor is not biased (recovery phase), the capture probability of the trap is very small, and thus it is at an empty state. Theoretically, when a bias voltage is applied (stress phase), the Fermi level of the transistor changes and the probability of the trap capturing a charge increases. Eventually when the transistor is biased one charge is captured, Ion current decreases and Vt increases, as the typical NBTI behavior in the stress phase. This highly unbalanced capture/emission rate impacts the average value of Vt over time.

4.3.1 Fermi level approximation

As expressed by equation 4.17, the average times of emission and capture τ_e and τ_c depend on the Fermi Level of the transistor. The Fermi level is indirectly computed inside physics-based transistor models like BSIM. However the Fermi-level is not available at the netlist-level of the simulation. This implies that a netlist-level implementation of a trap-detrap simulation cannot obtain the Fermi-level of the transistor computed inside

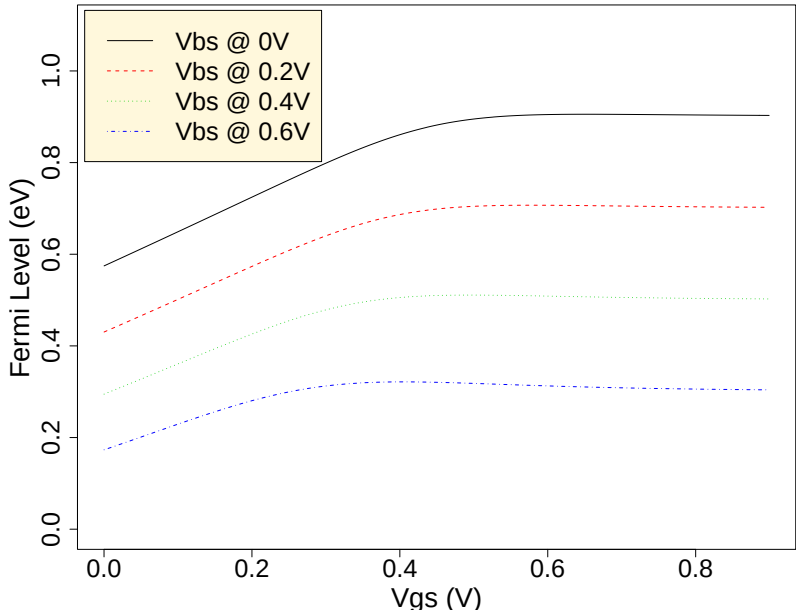


Figure 4.12: Fermi level as a function of Vgs and forward body bias.

the transistor model. The implementation proposed in this work solves this issue by directly modifying the BSIM4 source code, which is openly available and has extensive documentation (HU, 2009).

The Fermi level is a function of the oxide voltage Vox of the transistor. Vox then depends on Vgs and Vbs of the transistor. According to the BSIM4 User Manual (HU, 2009), the oxide voltage Vox is written as $Vox = Voxacc + Voxdepinv$, with:

$$Voxdepinv = k1 * \sqrt{\Phi_{sdepinv}} + Vgsteff \tag{4.18}$$

where, according to Hu (2009), the equation 4.18 is valid and continuous from depletion to inversion modes, which take place when the voltage at the gate is greater than the flat-band voltage ($Vgs > Vfb$). The equation 4.18 is implemented inside BSIM4 as the source code:

```
Voxdepinv = pParam->BSIM4v5k1ox*(T1 - T0)
```

Therefore the Fermi level can be expressed as

$$\Phi_{sdepinv} = (T1 - T0)^2 \tag{4.19}$$

which is an equation valid for depletion and inversion regions of operation. In accumulation, when $Vgs < Vfb$, this equation is not valid and can lead to wrong values if extrapolation is used.

The figure 4.12 shows $\Phi_{sdepinv}$ as an approximation for the Fermi Level for gate voltage between 0V and 0.9V. The transistor dimensions are L=45nm and W=50nm and the technology model-card is the Predictive Technology Model (PTM) 45nm (CAO; MCANDREW, 2007). Forward body bias (FBB) consists of applying a positive bulk-to-source voltage (Vbs) to the transistor. FBB is a technique commonly employed in analog and mixed-signal circuits in order to, among other effects, reduce the impact of 1/f noise. Notice that the bulk-to-source voltage Vbs is inversely proportional to the Fermi Level.

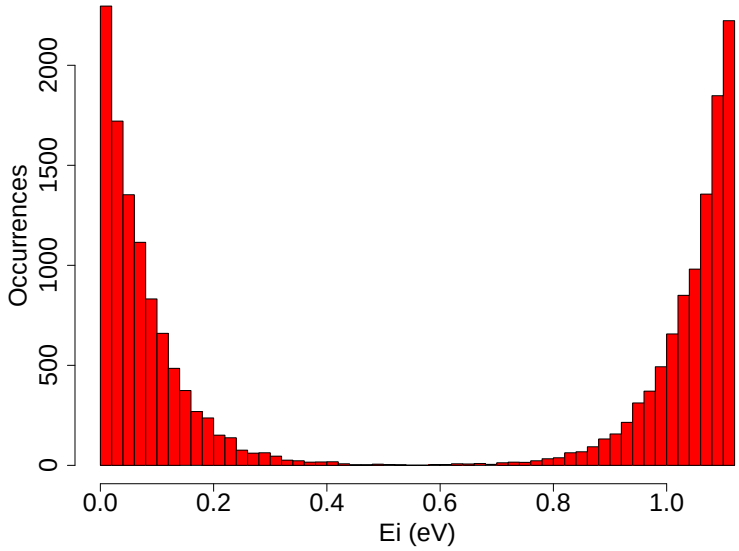


Figure 4.13: Distribution of the energies of the traps.

4.3.2 U-shape distribution of energies of the traps

The traps that contribute to NBTI are discrete energy states that may capture holes. They can be interface states or oxide traps. The energy states of these holes lie within the forbidden band-gap of silicon, which is approximately 1.12eV (SZE; NG, 2006).

Experimental works conclude that the probability of the trap having energy close to the conduction and valence bands are higher than having energy close to the center of the band-gap. In other words, the energies of the traps are distributed according to an U-shape distribution over the forbidden band-gap.

In order to obtain the U-shape distribution for the simulations, we use a random generator for exponential distribution. The proposed U-shaped distribution based on an exponential distribution with rate λ has density:

$$f(x) = \begin{cases} Eg * \lambda e^{-\lambda x} & \textbf{if } 0 < x < 0.5 \\ Eg * (1 - \lambda e^{-\lambda x}) & \textbf{if } 0.5 \leq x \leq 1 \end{cases} \tag{4.20}$$

where Eg is the forbidden band-gap in eV (approximately 1.12eV for silicon). The benefit of the U-shape being expressed as exponential distributions is the possibility to use an random number generator for exponential distribution. Such generator is readily available in Computer Algebra Systems (CAS) and easy to implement. However this approximation for generating an u-shape distribution can be used only with $\lambda \leq 15$, otherwise it generates values smaller than 0 and larger than Eg for large sample sizes . Higher λ leads to less points in the center of the distribution and more points on the edges. Figure 4.13 presents the histogram of an U-shape distribution generated according to equation 4.20 and using a random number generator for exponential distributions, with $\lambda = 15$ and $Eg = 1.12$.

4.3.3 Implementation

The implementation is an attempt to model the trap-detrap phenomena causing RTS and NBTI by modifying the BSIM4 source code (HU, 2009). We made the modifications into BSIM4 in such a way that it became a true statistical transistor model, with Vt varying

over time as a function of occupation and release of charges by the interface and oxide traps.

The traps are implemented as a new structure added to the transistor data structure of BSIM. The electrical simulator calls an initialization routine for each transistor. This routine typically sets the transistor parameters according to its sizing and temperature. The trap-detrap code related to the initialization of the model was added to this routine. This procedure is described in listing 4.1. First, a Poisson number generator gives the number of traps of the transistor. Then, each trap is associated with its δvt, its time constant p_i and its energy in the band-gap E_i.

```
for each transistor of the circuit{
  Ntr = Poisson (λ_Ntr)
  for each trap i<Ntr{
    δvt_i = randomly select from a list
    p_i = Uniform distribution
    E_i = U–Shape distribution
  }
}
```

Listing 4.1: Initialization of the transistors

The Markov Chain trap-detrap probabilistic mechanism described by figure 4.11 is implemented inside the evaluation routine of BSIM4. This routine is executed at every timestep of the transient analysis and computes, among other quantities, the source-to-drain current. Listing 4.2 describes the procedure. Depending on the trap status, its emission or capture probability is computed, based on its average emission or capture time. The random process consists of randomly deciding whether the trap keeps or changes its status in that timestep.

```
for each transistor {
  FermiLevel = Compute based on numerical fitting of Φ_sdepinv
  for each trap i{
    r = uniform random (0,1)
    if s_i(t−1)==0{ // empty trap
      τ_c = 10^pi (1+exp(E_i−E_f / K_b T))
      Pr(s_i(t)=0→1) = Δt/τ_c
      s_i(t) = r > Pr(s_i(t)=0→1) ? 1 : 0
    }else{ // occupied trap
      τ_e = 10^−pi (1+exp(E_f−E_i / K_b T))
      Pr(s_i(t)=1→0) = Δt/τ_e
      s_i(t) = r > Pr(s_i(t)=1→0) ? 0 : 1
    }
    if s_i(t)==1 { ΔVt(t) += δvt_i }
  }
}
```

Listing 4.2: Time-dependent trap-detrap simulation

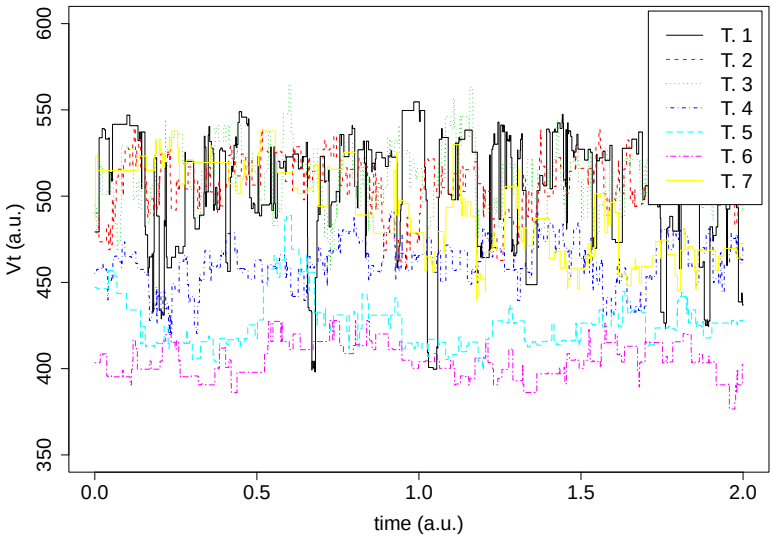


Figure 4.14: Transient simulations of 7 transistors (chosen arbitrarily from a sample of 100) showing the time evolution of the Threshold Voltage.

4.3.4 Simulation results

This section presents the results of the trap-detrap simulation scheme. Computer simulations were run in order to compute the fluctuations of the transistor Vt over time. The methodology was implemented in the open-source electrical simulator NGSPICE. NGSPICE is based on the open-source Spice3 from Berkeley, and it supports the latest transistor model BSIM4 from Berkeley. The modifications were mostly punctual changes on two source files: one related to the transistor initialization and another related to the calculation of the transistor current. The trap-detrap simulation corresponds to a transient analysis of the circuit under test.

The δvt_i, which is the Vt fluctuation due to one single trapped electron are given by 3D atomistic simulations described by Ashraf (2011) and Camargo (2010). These atomistic simulations data are only available for a transistor with dimensions $L = 45nm$ and $W = 50nm$. Thus all the trap-detrap simulation results presented here are restricted to this transistor sizing.

Another simulation parameter, the average number of interface traps (the λ_{Ntr} of the Poisson distribution) can be obtained by device measurements. Table I from Wirth (2005) presents the data from measurements of a 130nm technology node NMOSFET. For small-area transistors we linearly extrapolate the outcome of those measurements and use $\lambda_{Ntr} = 80\frac{W \times L}{45n \times 50n}$, where W and L are the channel width and length of the transistor. Thus, for our device with dimensions $45n \times 50n$, there are 80 traps on average. These traps have their time constant *pi* uniformly distributed in the interval $[-5, -8]$. Note that the number of traps that effectively generate RTS is much smaller, since only traps within a few *kBT* relative to the Fermi level, i.e. have similar emission and capture times, change their occupation state. These two parameters, the average number of interface traps and the time constant interval have not been properly calibrated in this work. Thus the results presented here are not appropriate for quantitative conclusions about the impact of RTS or NBTI on Vt. We present these simulation results with the only intention of introducing the capability of the tool we implemented. Due to the lack of calibration and thus the fact that the data is only shown to illustrate the capability of the simulator, the plots employ

arbitrary units (a.u.) for both time and Vt.

The procedure could correspond to the simulation of traps causing both RTS and NBTI, but in this experiment we intended to isolate the contribution of RTS solely. The reason for considering RTS only is that these experiments are intended to analyze jitter of oscillators. Because in a small simulation time window (such as simulation times in the order of micro-seconds) RTS impacts the standard deviate of Vt while NBTI impacts its average, we assume that RTS is the main cause of jitter. In order to analyze the impact of RTS only, we must detect and discard those traps that do not contribute to this effect. This is accomplished by two pre-characterization stages prior to the start of the trap-detrap simulation: stabilization phase for 1a.u. and activity testing phase for 2a.u. During these phases, which refer to the first 3a.u. of the transient simulation, only the activity is monitored and Vt is kept constant at nominal value.

The first phase, stabilization phase, corresponds to the first 1a.u. of the transient simulation. During the trap initialization, it is randomly assigned with an energy state within the forbidden bandgap and with an initial state (empty or occupied). However, many traps evolve to a more stable state. For instance, traps at low energy level (lower than the Fermi level tend to be), quickly become occupied independently on the initial state and very unlikely will emit a charge. These traps which switch in the initial 1a.u. and very likely will not present activity later are traps contributing to NBTI. During the initial 1a.u. there are more transitions then during the rest of the simulation.

The second phase, activity test, lasts for the next 2a.u. In this phase the activity of each trap is monitored, but no Vt shift is computed yet. The traps presenting less than two transitions during this 2a.u. are traps contributing to NBTI, and are not relevant to the jitter of oscillators due to RTS. These traps without enough activity are then discarded.

The stable trap-detrap simulation runs for the next 2a.u. This is the actual trap-detrap simulation we are interested, in which takes into account only the traps contributing to RTS. The total transient simulation time is 5a.u., but the first 3a.u. accounts for discarding traps which are not related to RTS. Thus, the trap-detrap simulation of RTS runs for 2a.u.

The procedure described by the algorithms 4.1 and 4.2 leads to a stochastic process evolving over time. This stochastic process is a Markov Chain, since the next state of the traps depends only on their current state. This process corresponds to the statistical behavior of one transistor over time. Each transistor has a random number of traps, as well as each trap has a randomly selected δvti and time constant pi.

Thus, one single run of the procedure is a representation of the behavior of one single device over time. In order to study the actual statistical impact of RTS or NBTI on the transistor, we must perform a Monte Carlo simulation of Markov Chains. Thus we run a Monte Carlo simulation using an ensemble of 100 devices, and examine each device behavior over 2a.u.

Figure 4.14 presents the Vt of 7 of these devices (chosen arbitrarily) during the 2a.u. simulation time. Each device presents a different behavior and some devices clearly have lower Vt than others during most of the time. Some devices can show more activity and some traps cause higher fluctuation than others. For instance, transistor T.1 has $\overline{Vth} = 500a.u.$ and $\sigma_{Vt} = 33a.u.$ during the 2a.u. simulation, while transistor T.6 has $\overline{Vth} = 400a.u.$ and $\sigma_{Vt} = 10a.u.$.

Extending this concept of analyzing the moments of the transistors, figure 4.15 reports all the averages and the standard deviations of the 100 Monte Carlo simulations. The average Vt of the 45nm by 50nm device can go from 405a.u. up to 673a.u., while the standard deviation is in the range of 9a.u. to 42a.u. The transistor which presents a

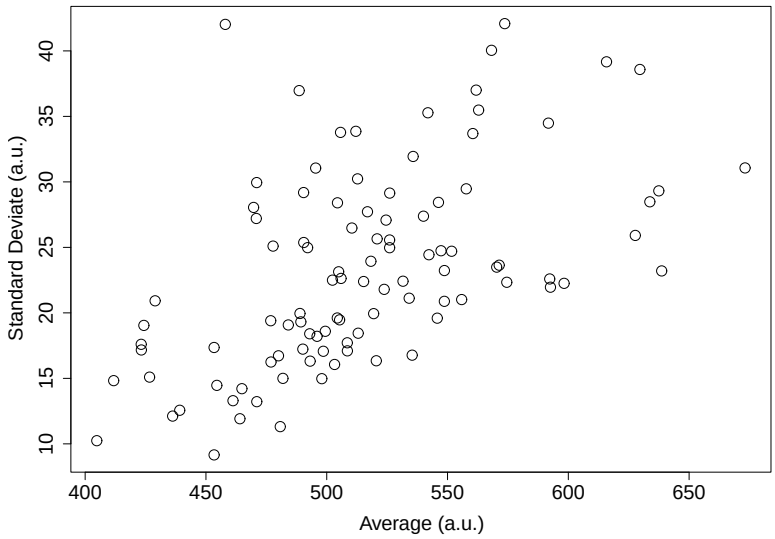


Figure 4.15: Cloud of averages and standard deviates of Vt for the 100 Monte Carlo simulations.

standard deviation of 42a.u. due to RTS causes more jitter (on a ring oscillator) then the one which presents standard deviation of 9a.u. The average of the standard deviations is 23a.u.

The figure 4.16 summarizes the experiment of 100 device simulations during 2a.u. for each run. It presents a Box and Whisker plot of all the simulations (CHAMBERS et al., 1983; VENABLES; RIPLEY, 2002). Each column corresponds to one Monte Carlo run, from 1 to 100, while its projection on the y-axis represents the Vt during the 2a.u. simulation on that run. The Box-Whisker plot is the best way to visualize all the data of the 100 simulations at once. The black line inside the boxes indicates the median of the distribution. The lower and upper corners of the box are the first and third quartiles of the distribution. This plot emphasizes the difference between each one of the 100 device simulations performed for Monte Carlo.

4.4 Discussion

This chapter discusses Random Telegraph Signal and Bias Temperature Instabiltiy. These phenomena affect transistors in a transitory manner, so that the device characteristics vary over time. EDA tools are still lacking accurate and efficient simulation methodologies to enable analysis of these phenomena at early design stages.

This chapter highlights latest research in the field, including recent studies for modeling RTS and BTI in the time domain. These methodologies target at proposing accurate time-domain transient analysis. In this context we developed two simulation methodologies: static and dynamic.

The static methodology is a pre-characterization step to be run prior to the electrical simulation. This step computes derating Vt factors for each device. Then electrical simulation is run considering the impact of noise. The dynamic methodology is a set of modifications in the BSIM4 transistor model. Thus variability is on-the-fly taken into account during simulation. The disadvantage of the dynamic approach is the computational overhead: simulation time increases as a function of the number of traps. The dynamic

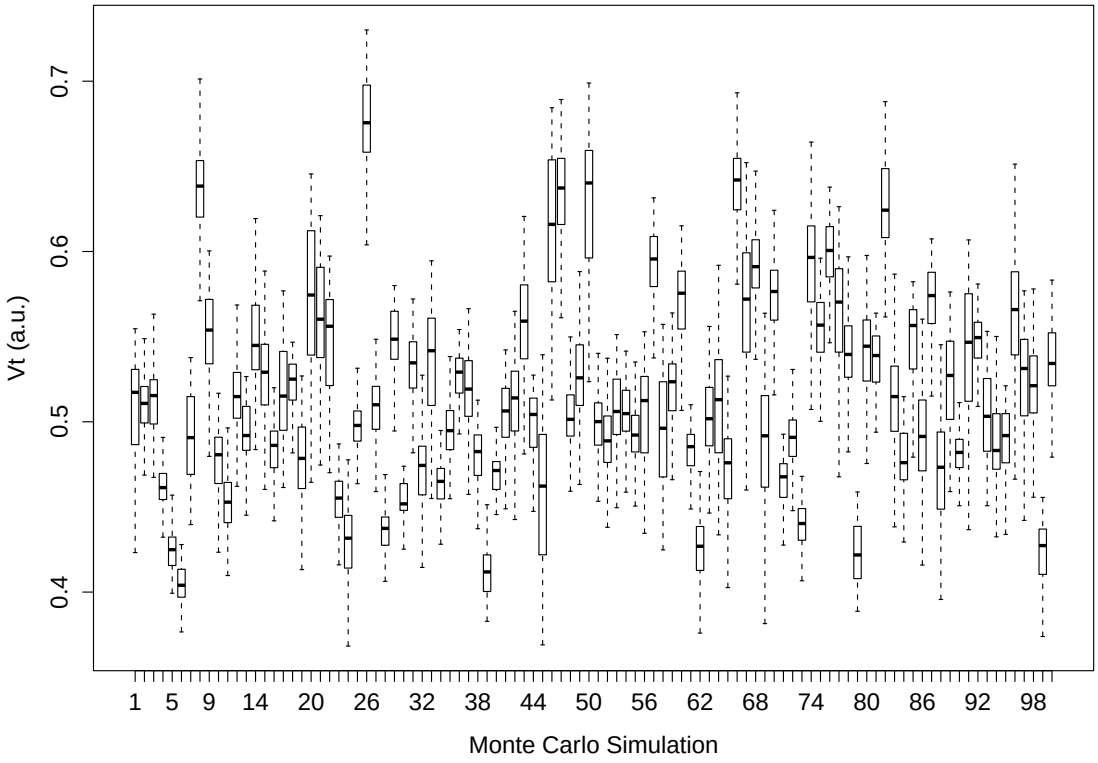


Figure 4.16: Box and Whisker plots of the Monte Carlo simulation, representing *Vt* distribution of an ensemble of 100 devices.

methodology must be fine-tuned with technology data provided by silicon measurements. Neither the static or the dynamic methods have been validated with silicon measurements.

5 LINEAR SENSITIVITY ANALYSIS

This chapter describes a framework to compute circuit variance and its sensitivity to the electrical parameters. The methodology is based on error propagation, which is commonly employed in measurement engineering and instrumentation (PARRAT, 1961). From Error Theory, basically error propagation expresses the error of an output variable as the sum of the squares of the (known) error of the inputs times the sensitivity of the output to that input. This approach is extended to compute the variance of a function which its inputs are known to be random variables. Error propagation requires the sensitivities of each variable with respect to the function being analyzed. These derivatives are computed numerically by electrical simulations.

Most EDA vendors have started to offer commercial tools to support statistical library characterization, e.g., Cadence Encounter Library Characterizer, Magma Silicon Smart, Synopsys NCX, Extreme DA GoldTime. These tools are based on (linear) sensitivity analysis. Assuming that one parameter *drifts* at a time while others are kept at nominal value, the response of the circuit is obtained as a (linear) *drift* of the circuit's output response around its nominal value as function of the amount of parameter change. This approach is based on the assumption that variations of the circuit parameters propagate linearly to circuit responses. For instance, consider a circuit with n transistors and F which is a function of the Vt of these n transistors. Assuming F as a linear function of Vt means that the following approximation is used:

$$F(\overline{Vt_1}+\Delta Vt_1,...,\overline{Vt_n}+\Delta Vt_1) = F(\overline{Vt_1},...,\overline{Vt_n}) + \sum_{i=1}^{n}\left[\frac{\partial F}{\partial Vt_i}\Delta Vt_i\right] \tag{5.1}$$

Figure 5.1 shows the linear sensitivity analysis flow as compared to a Monte Carlo flow. The probability distribution of the random variables must be characterized by measurements performed by the process engineering team of the partner foundry. The random variables are assumed to follow Normal distribution, thus each variable has a mean and a standard deviation. One circuit response is computed for each run and these responses are aggregated in order to compute the sensitivities for each electrical parameter.

Linear sensitivity analysis is becoming the main approach for propagating variability from gate level to circuit level because it is a good compromise between simplicity and accuracy.

5.1 Error propagation

Given the statistical nature of fabrication process, device characteristics such as Vt, W, L and $\Delta\beta/\beta$ of the transistors can be modeled as random variables. The circuit met-

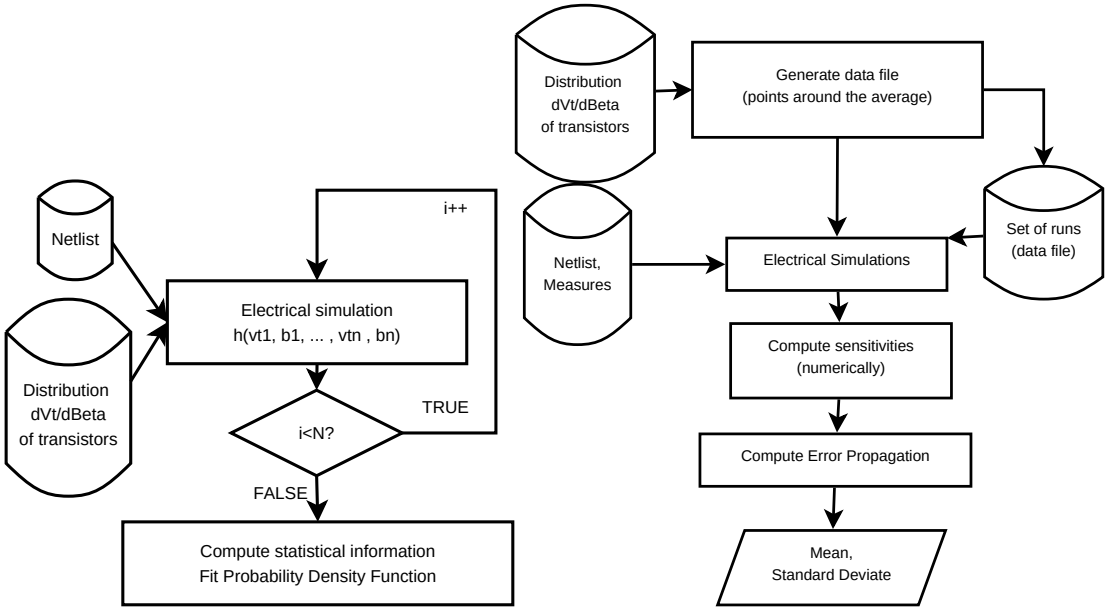


Figure 5.1: (a) Traditional Monte Carlo based flow and (b) sensitivity-analysis based flow.

rics (the output of a simulation) such as performance are also random variables and can be modeled using the classic propagation of uncertainties approach from Error Theory (PARRAT, 1961). In order to use that approach, two assumptions must hold:

1. the inputs follow a Normal distribution;

2. the propagation function can be approximated by a linear function in the region of interest.

Under these assumptions, the output p of the simulation (for instance rise or fall delay, transition time, dynamic or leakage power) can be approximated as a Normal random variable, and its moments can be computed analytically. Without loss of generality, consider two random variables in our compact variability model: Vt and β, and the number of transistors in the circuit under test is n. The inputs of a tool performing sensitivity analysis are the standard deviations of the inputs (σ_{Vti} and $\sigma_{\beta i}$ of each transistor i). These data come from process technology characterization. The standard deviation of the circuit response p can be approximated by:

$$\begin{aligned}\sigma_p^2 \approx & \sum_{i=1}^{n}\left[\left(\frac{\partial p}{\partial Vt_i}\sigma_{Vt_i}\right)^2+\left(\frac{\partial p}{\partial \beta_i}\sigma_{\beta_i}\right)^2\right]+2\sum_{i=1}^{n}\sum_{j=i}^{n}\left(\frac{\partial p}{\partial Vt_i}\frac{\partial p}{\partial \beta_i}\rho_{Vt_i,\beta_j}\right)\\ & + 2\sum_{i=1}^{n}\sum_{j=i}^{n}\left(\frac{\partial p}{\partial Vt_i}\frac{\partial p}{\partial Vt_i}\rho_{Vt_i,Vt_j}\right)+2\sum_{i=1}^{n}\sum_{j=i}^{n}\left(\frac{\partial p}{\partial \beta_i}\frac{\partial p}{\partial \beta_i}\rho_{\beta_i,\beta_j}\right)\end{aligned} \tag{5.2}$$

where σ_{Vti} and σ_{β_i} are the standard deviations of voltage threshold and current variation of transistor i, respectively, $\frac{\partial p}{\partial Vt_i}$ and $\frac{\partial p}{\partial \beta_i}$ are the sensitivities of Vt and β w.r.t parameter p, and finally $\rho_{x,y}$ are the correlation coefficients between random variables x and y. In absence of correlation between the random variables, error propagation simplifies to:

$$\sigma_p^2 \approx \sum_{i=1}^{n} \left[\left(\frac{\partial p}{\partial Vt_i} \sigma_{Vt_i} \right)^2 + \left(\frac{\partial p}{\partial \beta_i} \sigma_{\beta_i} \right)^2 \right] \tag{5.3}$$

The reader should notice that formulations 5.2 and 5.3 require the same number of electrical simulations, because both require the computation of exactly the same amount of partial derivatives. This means that correlations between electrical parameters can be taken into account without overhead in the number of simulations.

Considering that the response is approximately linear to the inputs in the region of interest and that there is no correlation between the input random variables, the circuit response p can be assumed to follow a Normal distribution with mean and variance given by (BRUSAMARELLO, 2006; BRUSAMARELLO et al., 2008):

$$\begin{cases} \mu_p \approx \overline{p} \\ \sigma_p^2 \approx \sum_{i=1}^{n} \left[\left(\frac{\partial p}{\partial Vt_i} \sigma_{Vt_i} \right)^2 + \left(\frac{\partial p}{\partial \beta_i} \sigma_{\beta_i} \right)^2 \right] \end{cases} \tag{5.4}$$

The non biased sampling estimator to standard deviation computed from a sample of n_{sample} experimental measures of S, denoted as S_1, S_2, ..., $S_{nsample}$, calculated by expression

$$\delta_S = \sqrt{\frac{1}{(n_{sample}-1)} \sum_{i=0}^{n_{sample}} (S_i - \langle S_i \rangle)^2}$$

must be numerically equal to σ_S for a n_{sample} sufficiently large, i.e.,

$$\delta_S \approx \sigma_S$$

Monte Carlo simulation (AMAR, 2006) is often employed in order to obtain the probability density function (PDF) of some circuit output (delay, power consumption, leakage current, ...). Usually, a sample size n_{sample} is generated, aiming the convergence of the standard deviation. However, the error in a Monte Carlo simulation is hardly reduced, once it is $O(1/\sqrt{n_{sample}})$. Figure 5.2 presents the convergence of σ as a function of the number of Monte Carlo simulations compared to the standard deviate computed analytically using error propagation (using 1st order and 2nd order approximation for the sensitivity, which will be discussed in the next section).

So, partial derivatives of the circuit response for the random parameters, standard deviation of random parameters and correlation between random parameters are inputs for the error propagation formula. Standard deviations and correlation coefficients of the input random variables are technology dependent and can be extracted. According to what will be shown in section 5.2, as $F(k_1, \ldots, k_N)$ is an arbitrary function that can be computed by electrical simulation, the numerical estimates for derivatives $\frac{\partial F}{\partial k_i}$ can also be computed by electrical simulation. From these derivatives, the variability of the output can be computed.

5.2 Numerical estimate of sensitivities

Numerical approximations of sensitivities are applied in order to present a generic methodology independent of circuit topology. First order and second order linear approximations, using respectively 1 and 2 points around the nominal values, are exploited

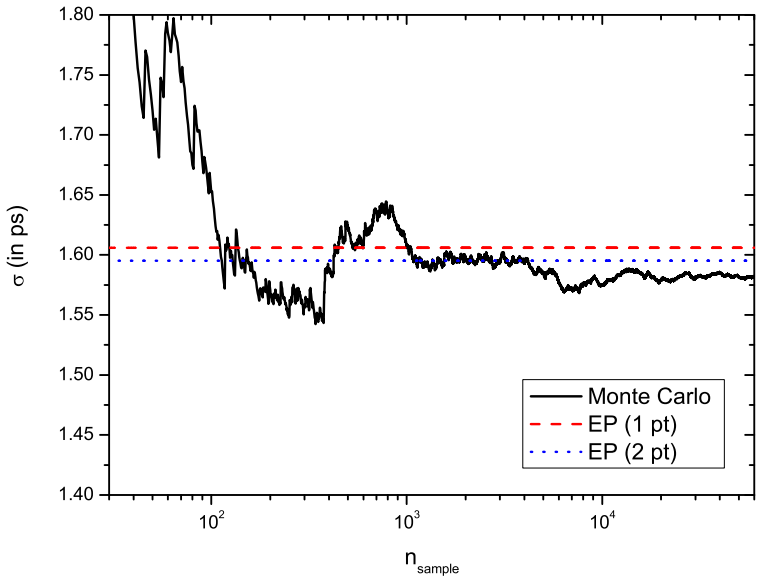


Figure 5.2: Convergence of σ as a function of the number of Monte Carlo simulations.

aiming to obtain sensitivity of circuit response for the variables of interest. The difference between these formulas is the accuracy in the numerical estimates and the number of electric simulations: higher order approximations require more simulations and are more accurate.

Problem Formulation: *Consider a general function of n variables* $f = f(x_1, x_2, \ldots, x_n)$*, such that numerical values for the variables are* $x_1 = \overline{x_1}, \ldots, x_n = \overline{x_n}$*. By error propagation we have* $\sigma_f^2 = (\partial f/\partial x_1)^2_{x_1=\overline{x}_1}\sigma^2_{x_1} + \ldots + (\partial f/\partial x_n)^2_{x_n=\overline{x}_n}\sigma^2_{x_n}$*. Find numerical approximation for* $\partial f/\partial x_i$ $(i = 1, \ldots, n)$*.*

5.2.1 1st Order Approximation

Expanding the n-dimensional Taylor series of order 2 around the point $f(\overline{x_1}, \ldots, \overline{x_i}, \ldots, \overline{x_n})$ yields:

$$\begin{aligned} f(\overline{x_1}, \ldots, \overline{x_i} + \Delta, \ldots, \overline{x_n}) &= f(\overline{x_1}, \ldots, \overline{x_i}, \ldots, \overline{x_n}) + \frac{\partial f(\overline{x_1}, \ldots, \overline{x_i}, \ldots, \overline{x_n})}{\partial x_i}(\overline{x_i} + \Delta - \overline{x_i}) + O(\Delta^2) \\ &= f(\overline{x_1}, \ldots, \overline{x_i}, \ldots, \overline{x_n}) + \Delta\frac{\partial f(\overline{x_1}, \ldots, \overline{x_i}, \ldots, \overline{x_n})}{\partial x_i} + O(\Delta^2) \end{aligned} \tag{5.5}$$

Numerical value for $f(x_1, \ldots, x_n)$ is given by electrical simulator. Thus, one can calculate the sensitivity of point $f(\overline{x_1}, \ldots, \overline{x_i} + \Delta, \ldots, \overline{x_n})$, rewriting 5.5 for $\Delta \ll 1$ as follows

$$\frac{\partial f}{\partial x_i}(\overline{x_1}, \ldots, \overline{x_i}, \ldots, \overline{x_n}) = \frac{f(\overline{x_1}, \ldots, \overline{x_i} + \Delta, \ldots, \overline{x_n}) - f(\overline{x_1}, \ldots, \overline{x_i}, \ldots, \overline{x_n})}{\Delta} + O(\Delta) \tag{5.6}$$

Thus, to compute the sensitivities of a parameter p of the circuit w.r.t. the variables Vt_i and β_i of the circuit three simulations are required: the nominal simulation $\overline{f}$ and the two simulations computing p drifting one of the variables at a time. Figure 5.3 illustrates

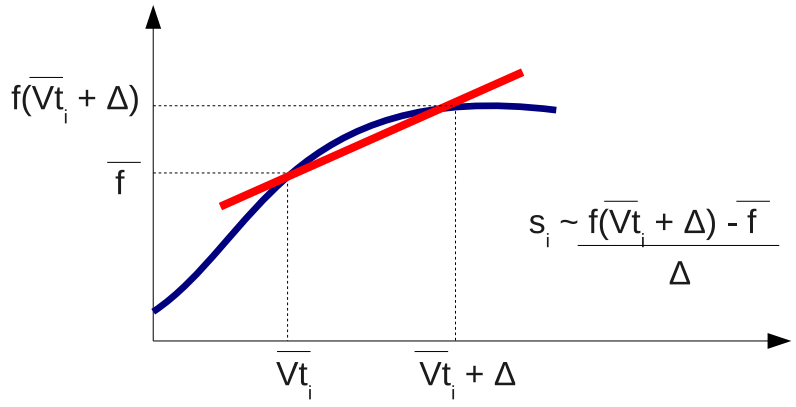


Figure 5.3: Numerical 1st order linear approximation of sensitivity.

the linear approximation as opposed to the response function, as well as the approximated sensitivity. The sensitivities are computed as:

$$\begin{aligned} \frac{\partial p}{\partial Vt_i} &= \frac{f(\overline{Vt_1},\ldots,\overline{Vt_i}+\Delta Vt,\ldots,\overline{Vt_n},\overline{\beta_1},\ldots,\overline{\beta_n})-\overline{f}}{\Delta Vt} \\ \frac{\partial p}{\partial \beta_i} &= \frac{f(\overline{Vt_1},\ldots,\overline{Vt_n},\overline{\beta_1},\ldots,\overline{\beta_i}+\Delta\beta,\ldots,\overline{\beta_n})-\overline{f}}{\Delta\beta} \end{aligned} \tag{5.7}$$

where $f(\overline{Vt_1},\ldots,\overline{Vt_n},\overline{\beta_1},\ldots,\overline{\beta_n})$ corresponds to a circuit response and can be computed by simulation (such as rise and fall delays, transition times, power, hold and setup times, etc). The response is a function of the variations in Vt and β of the n transistors. The simulation $\overline{f}$ corresponds to the nominal circuit response. The functions $f(\overline{Vt_1},\ldots,\overline{Vt_i}+\Delta Vt,\ldots,\overline{Vt_n},\overline{\beta_1},\ldots,\overline{\beta_n})$ and $f(\overline{Vt_1},\ldots,\overline{Vt_n},\overline{\beta_1},\ldots,\overline{\beta_i}+\Delta\beta,\ldots,\overline{\beta_n})$ correspond to two electrical simulations, where respectively Vt_i and β_i drift by a given Δ.

Complexity for 1st order approximation: For this case it is required 2 electrical simulations to compute each partial derivative, one is required to compute $f(\overline{x_1},..,\overline{x_i}+\Delta,...,\overline{x_n})$ and another run for $f(\overline{x_1},..,\overline{x_i},...\overline{x_n})$. But, as $f(\overline{x_1},..,\overline{x_i},...\overline{x_n})$ is constant for all partial derivatives, it can be computed only once. Thus, computation of all partial derivatives using first order approximation requires $n+1$ runs.

5.2.2 2nd Order Approximation

In order to obtain a more precise approximation, algebraic manipulations over Taylor expansion results in a formula with accuracy $O(\Delta^2)$. Consider for now a Taylor expansion about point $f(\overline{x_1},\ldots,\overline{x_i}-\Delta,\ldots,\overline{x_n})$ as follows:

$$\begin{aligned} f(\overline{x_1},\ldots,\overline{x_i}-\Delta,\ldots,\overline{x_n}) &= f(\overline{x_1},\ldots,\overline{x_i},\ldots,\overline{x_n})+\frac{\partial f(\overline{x_1},\ldots,\overline{x_i},\ldots,\overline{x_n})}{\partial x_i}(\overline{x_i}-\Delta-\overline{x_i})+O(\Delta^2) \\ &= f(\overline{x_1},\ldots,\overline{x_i},\ldots,\overline{x_n})-\Delta\frac{\partial f(\overline{x_1},\ldots,\overline{x_i},\ldots,\overline{x_n})}{\partial x_i}+O(\Delta^2) \end{aligned} \tag{5.8}$$

Then, using results 5.5 and 5.8, a better approximation for $\frac{\partial}{\partial x_i} f(\overline{x_1},\ldots,\overline{x_i},\ldots,\overline{x_n})$ can be computed according to:

$$\frac{\partial}{\partial x_i} f(\overline{x_1},\ldots,\overline{x_i},\ldots,\overline{x_n}) = \frac{f(\overline{x_1},\ldots,\overline{x_i}+\Delta,\ldots,\overline{x_n}) - f(\overline{x_1},\ldots,\overline{x_i}-\Delta,\ldots,\overline{x_n})}{2\Delta} + O(\Delta^2) \tag{5.9}$$

Complexity for 2nd order approximation: this formulation requires two electrical simulations for each variable of interest: one run for $f(\overline{x_1},\ldots,\overline{x_i}+\Delta,\ldots,\overline{x_n})$ and another one for $f(\overline{x_1},\ldots,\overline{x_i}-\Delta,\ldots,\overline{x_n})$. Thus, to calculate n partial derivatives to all the variables, 2nd order approximation requires $2n$ runs. As one nominal simulation is required for the approximation of the mean, $2n+1$ runs are required.

5.2.3 Complexity discussion

In order to apply error propagation technique to the analysis of variability in integrated circuits, means to compute the partial derivatives of the circuit – the sensitivity of the circuit response w.r.t the random variables are needed. The formula using two points around the mean gives a more accurate error order, at the cost of an increase in the running time.

The number of electrical simulations is a function of the approximation formula, number of transistors and number of random variables. For n transistors and j electrical parameters considered as random variables, the number of variables in the function is $m = n \times j$. Thus, 1*st* order approximation formula requires $m+1$ simulations, while 2*nd* order approximations require $2m+1$ runs.

The main goal is to accomplish variability simulation of small electrical blocks such as static and dynamic logic gates and memories. These kind of circuits often present small number of transistors, and often only a few number of electrical parameters are assumed to present variations due to process variability. Thus, the numerical estimates of derivatives can avoid thousand of simulations in comparison to traditional Monte Carlo – widely employed in electrical simulation tools –, for which a reasonable number of runs must be performed to obtain a suitable estimate of variance in the measures of interest.

6 RESPONSE SURFACE METHODOLOGY[1]

The CPU time expensive Monte Carlo method can be employed for the characterization of standard cell libraries (AMAR, 2006). Such approach allows variability-aware analysis to be implemented with minor changes on top of existing characterization tools but requires a large number of runs N. Indeed, thousands of simulations are needed for accurately capturing the tails of the distribution of the affected metrics, typically at a 3σ distance from the average value. The uncertainty of an estimator in Monte Carlo is $\approx 1/\sqrt{N}$: in order to obtain a result 2x more accurate, the number of simulations have to be increased by a factor of 4.

To achieve near MC accuracy with a speedup improvement of orders of magnitude, this chapter presents the use of Response Surface Methodology in conjunction with a new Design of Experiments (Brussel DOE) which performs the selection of design points and guarantees true statistical relevance of these points. It is combined with a model selection algorithm capable of building the most suitable non-linear regression model to represent the circuit response.

This chapter presents a novel time-efficient and accurate variability aware standard cell characterization approach. The approach is accurate because of twofold: (1) the use of a new DoE capable of capturing the true statistical nature of the underlying process parameters and (2) the use of a model selection algorithm capable of building the most suitable non-linear regression model to represent the circuit responses. On the other hand, the approach is time-efficient because the number of electrical simulations is reduced by several orders of magnitude comparing to conventional MC and the simulation effort linear with the number of devices of the gate. Finally the DoE and model selection algorithm described in this manuscript are generic enough to be added on top of conventional Non-Linear Delay and Power Model (CROIX; WONG, 1997) as well as recent cell models described by Synopsys (TRIHY, 2008) and Intel (MENEZES; KASHYAP; AMIN, 2008).

Section 6.1 summarizes the existing state-of-the-art. Section 6.2 presents the described statistical RSM flow and how it compares to the traditional statistical cell characterization flow. Section 6.3 explains the details of the Brussel design, a novel DoE approach for selecting few points that represent the original random MC points. Section 6.4 presents the automatic model selection flow.

[1]This chapter describes an invention that has been filed as patent in the European Union and in the United States of America: (MIRANDA; ROUSSEL; BRUSAMARELLO, 2010) and (MIRANDA; ROUSSEL; BRUSAMARELLO, 2011).

6.1 Background

6.1.1 Design of Experiments

Accurate gate level modeling for delay and power response estimation has become a major challenge for nano-metric technologies (KELLER; TAM; KARIAT, 2008). The use of RSM techniques in VLSI design for standard cell characterization is not new and its use was originally proposed in the late 19880's (ALVAREZ et al., 1988; HOCEVAR; COX; YANG, 1988). Recently, the use of these techniques raised interest again as an effective technique to cope with the explosion on the required process corners to capture the combined impact of local and global process variations in the gate response. Basu (2007) proposes an analytical function for gate delay is described which is built using RSM after transforming the correlated components of the output response into uncorrelated ones. More recently, Kim (2007) proposes a fast methodology based on sensitivity analysis for characterizing transistor level circuit descriptions has been also proposed. In the context of interconnect modeling, the use of RSM techniques has reported good speedup and accuracy when accounting for the impact of process variations in interconnect timing (WANG; GHANTA; VRUDHULA, 2004). In the context of mixed signal design Mcconaghy (2009) has proposed a regression model for quick evaluation of the impact of circuit parameter changes in the desired circuit response by providing only relative accuracy to guide the optimizer, hence not guaranteeing absolute accuracy.

Maricau (2010) improves on the earlier work using a regression model aiming estimating absolute accuracy. Li (2009) employs Least Angle Regression (LAR) substituting Least Squares (LS) with model selection. While that work proposes a method for very high dimensionality (thousands of variables), we propose to use a compact model to reduce the dimensionality by orders of magnitude. Also, by using $2n+1$ Brussel Design points we guarantee a non-linear model with cross-terms which is better than the random selection of points used for both LAR and LS in that paper.

Many of these works suggest the use of conventional DoE methods like Central-Composite-Design, full factorial and/or Box-Behenken Design (MYERS; MONTGOMERY, 2002), which do not consider the statistical nature of the process variations parameters for their design point selection criteria and yet require many more simulations than the Brussel Design.

6.1.2 Deterministic Propagation Function

Propagation functions of standard cells have a sparse vector of coefficients (LI, 2009) because usually only few parameters affect the output response, i.e. Vt of the PMOS transistor barely affects fall delay of an inverter. Algorithms for automatic model selection have been studied by the statistical community in order to find correct propagation function describing such propagation functions with many variables but very sparse matrix (AKAIKE, 1973). Regression output like residuals (sum of squares), t-statistics, r^2, F-statistic represent the goodness of fit and can be used to drive a search for the best model describing the underlying function (VENABLES; RIPLEY, 2002). Using these criteria tend to produce over-fitted models without physical support, thus Akaike (1973; 1974) and Schwarz (1978) proposed model selection algorithms driven by Akaike Information Criteria (AIC) and Bayesian Information Criteria (BIC). Both are based on the model Likelihood, but with an penalization term which is a function of the number of coefficients of the model. This penalization prevents over-fit because models with smaller number of coefficients have priority over large number of coefficients.

The regression models used in previous model selection algorithms for VLSI design are usually based on previous knowledge about the function being modeled such as assuming a template knowledge of the target function (for instance assuming the response to follow a particular polynomial) (ALVAREZ et al., 1988; HOCEVAR; COX; YANG, 1988; BASU; VEMURI, 2007; KIM; JONES; HOROWITZ, 2007; MARICAU; GIELEN, 2010). The recent work of Li (2009) presents the most advanced method, based on LAR, but even that method cannot handle non-linearity without drastically increasing the sample size. This is not required by our method as our approach is based on a generic and efficient algorithm for model selection.

6.1.3 Advantages

A standard cell library that captures statistical information about the gates, i.e. probability distributions, is at the core of Statistical Static Timing Analysis (SSTA), from block-based SSTA (VISWESWARIAH et al., 2004; HELOUE; ONAISSI; NAJM, 2008) to non-parametric MC-based approaches(IMAI et al., 2008). Thus, accurate and time-efficient modeling of delay and power of transistor level circuit descriptions of nano-metric technologies has become a major challenge (KELLER; TAM; KARIAT, 2008). As compared to previous works, the proposed novel approach differentiates in the following aspects:

- **Variability Aware**: Unlike conventional DoE approaches (e.g., full factorial design) the Brussel DoE selects only design points that are statistically relevant [NOTAR explicar melhor, exemplo] to the parameter domain distribution .
- It considers **Input Correlations**: The Brussel DoE properly captures the existing correlation between input parameters, hence being able to expose cross-term dependencies between the input process parameter domain and output gate response.
- The best propagation function is found **on-the-fly** : the polynomial for fitting the circuit response is selected on-the-fly and is not limited to a pre-defined template function. The propagation function can have linear, quadratic and cross-terms.
- The approach works under **Non-Normality** assumption, e.g. not limited to assumptions of any nature (e.g., Gaussian) underlying the statistical distribution of the process parameters.

6.2 High-level description of the flow

Figure 6.1.a shows the traditional cell characterization flow based on Monte Carlo simulations at electric level. The accuracy of the estimators obtained using this flow is limited by the number of electrical simulations N. The alternative flow we propose is presented in figure 6.1.b. The Brussel design of experiments is used as a pre-processing step to determine a small set of N_{doe} artificially generated points that represent the original sample of N_{doe} random ΔVt and $\Delta\beta$. The tremendous speedup of the flow relies on the fact that $N_{doe} \ll N$, so the number of electrical simulations is much smaller. After the N_{doe} selected electrical simulations, a model selection algorithm searches for an optimal non-linear regression model relating inputs to outputs hence representing the outcome of electrical simulations.

After this, a large sample of MC experiments can be run quickly using the surrogate model instead of spice, because computing each instance of the regression function is very fast.

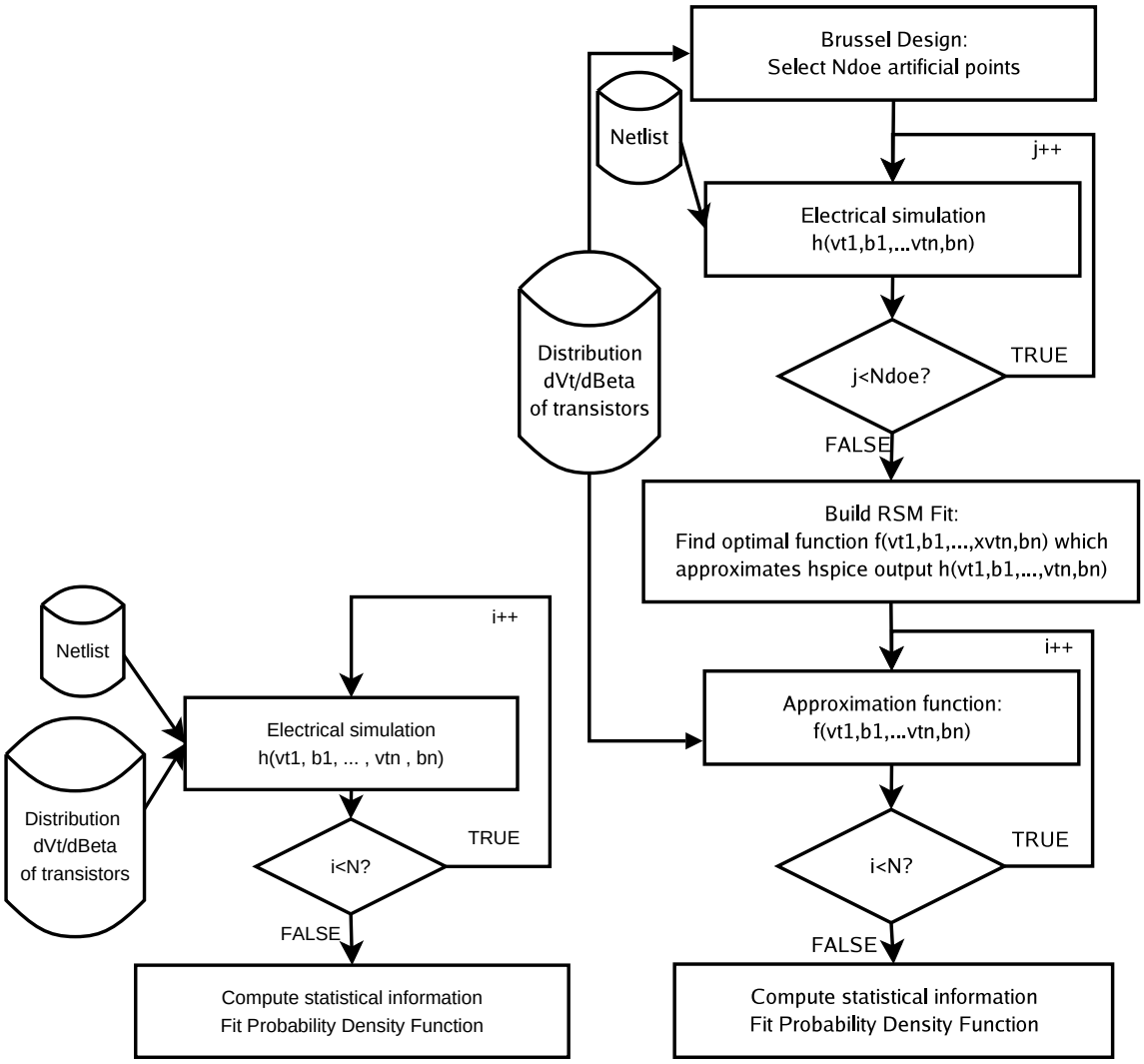


Figure 6.1: (a) Traditional Monte Carlo flow for cell characterization and (b) proposed flow based on DoE and RSM.

6.3 Design of Experiments: Brussel Design

The first step in order to achieve a good response surface fit is to perform Design of Experiments (MYERS; MONTGOMERY, 2002). The goal of this stage is to find N_{doe} points which are representative for the n-dimensional input space of random variables. At this stage there is no previous knowledge regarding the propagation function to be modeled. The points need to be selected in such a way that they cover as much as possible the domain of the output distribution.

Problem definition: Let a Monte Carlo ensemble Γ^M of size N of the n-dimensional function be given by

$$\Gamma^M = \{\{Vt_1, \beta_1, \ldots, Vt_n, \beta_n\}_1, \ldots, \{Vt_1, \beta_1, \ldots, Vt_n, \beta_n\}_N\}.$$

Find an alternative ensemble Γ^B of size $Ndoe$ given by

$$\Gamma^B = \{\{Vt_1, \beta_1, \ldots, Vt_n, \beta_n\}_1, \ldots, \{Vt_1, \beta_1, \ldots, Vt_n, \beta_n\}_{Ndoe}\}$$

which covers the same sub-domain as the original sample.

The DoE proposed in this work, the Brussel Design, is a DoE technique which exploits the existing knowledge about the statistical input variable domain to be sampled. This DoE allows fitting a linear response surface as well as higher order approximations (2^{nd} and 3^{rd} order) and offers a proper balance between accuracy and input variable validity range. Philippe Roussel, from Reliability Group at IMEC, proposed the Brussel DOE presented in this section.

To select appropriate points according to the Brussel design there are two steps: 1) build an n-dimensional Probability Density Function (PDF) representing the multivariate statistic and 2) proper selection of $2n+1$ DoE points based on the cloud spread. These steps are detailed respectively in subsections 6.3.0.1 and 6.3.1.

6.3.0.1 Weighted Multicomponent Multivariate Normal PDF representation

This section describes a generic formulation for describing Probability Density Functions (PDF): the Weighted Multicomponent Multivariate Normal. It approximates multidimensional input statistical domains and can reliably handle any shape of PDF, such as Normal, Bimodal, Exponential and so on. It consists of dividing the dataset into clusters, fitting a multivariate Normal distribution to each separate cluster and finally aggregating (and weighting) the Normal distributions. However, the reader should notice that this methodology is exceptionally needed in practice, i.e. when the distribution deviates significantly from normality. As the input variability data (Vt and β) shown in this paper follow normality, they are modelled using a single component multivariate normal PDF. In this case assume the number of clusters k is 1.

The first step to build the multinormal n-dimensional PDF is to partition the dataset into k cluster components. Clustering algorithms separate the N elements of the dataset into k groups, so that the elements of each cluster are similar to each other according to a robust distance criterion. The most common clustering algorithms can be divided into (GAN; MA; WU, 2007):

- hierarchical clustering: agglomerative (bottom-up) and divisive (top-down) clustering methods;
- center-based clustering: k-means algorithm;
- search-based clustering: clustering using genetic algorithms and tabu search;
- model-based clustering: EM algorithm;
- fuzzy clustering: fuzzy k-means.

These and more clustering algorithms are described in detail by Gan (2007). Some algorithms, as for instance k-means, require the user to specify the number of clusters. In our experience k is usually in the range 1..3. The clustering algorithm also depends on the distance metric, as for instance Euclidean distance, Manhattan distance or Mahalanobis distance. However when dimensions mix different units it is important to consider a normalized distance. Our partitioning is based upon a unit free, rescaled Euclidean distance criterion, which is a robust version of a Mahalanobis distance and guarantees good partitioning in cases where dimensions have different units. Methods for automatically deciding the number of clusters and distance metrics for clustering have been much discussed in the Literature, and Gan (2007) presents a detailed discussion on these topics.

Once the dataset has been divided into optimal clusters, a multivariate Normal distribution is fitted to each cluster component $\{Vt_1, \beta_1, ...\}_i$. The PDF of a multivariate Normal distribution for a single component cluster i is described as:

$$f_i\left(\vec{t}, \vec{\mu}_i, S_i\right) = \frac{e^{-\frac{1}{2}(t-\mu_i).S_i^{-1}.(t-\mu_i)}}{(2\pi)^{\frac{n}{2}}\sqrt{|S_i|}} \tag{6.1}$$

where $\bar{t}$ is the multivariate stochastic, $\vec{\mu}_i$ is the vector of central value of the variables and S_i is the covariance matrix between the variables given by:

$$\boldsymbol{S} = \begin{pmatrix} \sigma_1^2 & \rho_{12}\sigma_1\sigma_2 & \cdots & \rho_{1n}\sigma_1\sigma_n \\ \rho_{12}\sigma_1\sigma_2 & \sigma_2^2 & \cdots & \rho_{2n}\sigma_2\sigma_n \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{1n}\sigma_1\sigma_n & \rho_{2n}\sigma_2\sigma_n & \cdots & \sigma_n^2 \end{pmatrix} \tag{6.2}$$

here ρ_{lm} is the correlation between variables l and m. Then the multiple PDF's are accumulated into a proportional sum weighted by cluster component size:

$$f\left(\vec{t}\right) = \frac{\sum_{i=1}^{k} w_i f_i\left(\vec{t}, \vec{\mu}_i, S_i\right)}{\sum_{i=1}^{k} w_i} \tag{6.3}$$

where $\vec{\mu}_i$ and S_i are $\vec{\mu}$ and S of the variables of the cluster component i, w_i is its size. Each data cluster generates a different covariance matrix S.

6.3.1 Selection of DoE points

Each covariance matrix S, given by eq. (6.2), is decomposed using the diagonal matrix of σ values for each variable:

$$S = \sigma \cdot \rho \cdot \sigma, \text{with} \tag{6.4}$$

$$\sigma = \begin{pmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_n \end{pmatrix} \tag{6.5}$$

where σ is extracted as the square root of the matrix diagonal, so that ρ becomes the corresponding correlation matrix:

$$\rho = \begin{pmatrix} 1 & \rho_{12} & \cdots & \rho_{1n} \\ \rho_{21} & 1 & \cdots & \rho_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{n1} & \rho_{n2} & \cdots & 1 \end{pmatrix} \tag{6.6}$$

In effect, this standardizes the variables into unit free ones:

$$f\left(\vec{t}, \vec{\mu}, S\right) = \frac{e^{-\frac{1}{2}\left(\frac{t-\mu}{\sigma}\right)\cdot\rho^{-1}\cdot\left(\frac{t-\mu}{\sigma}\right)^{\mathsf{T}}}}{(2\pi)^{\frac{n}{2}}\sqrt{|S|}} \tag{6.7}$$

Next, a principal value decomposition of the correlation matrix is performed:

$$\rho = R^T \cdot E \cdot R \tag{6.8}$$

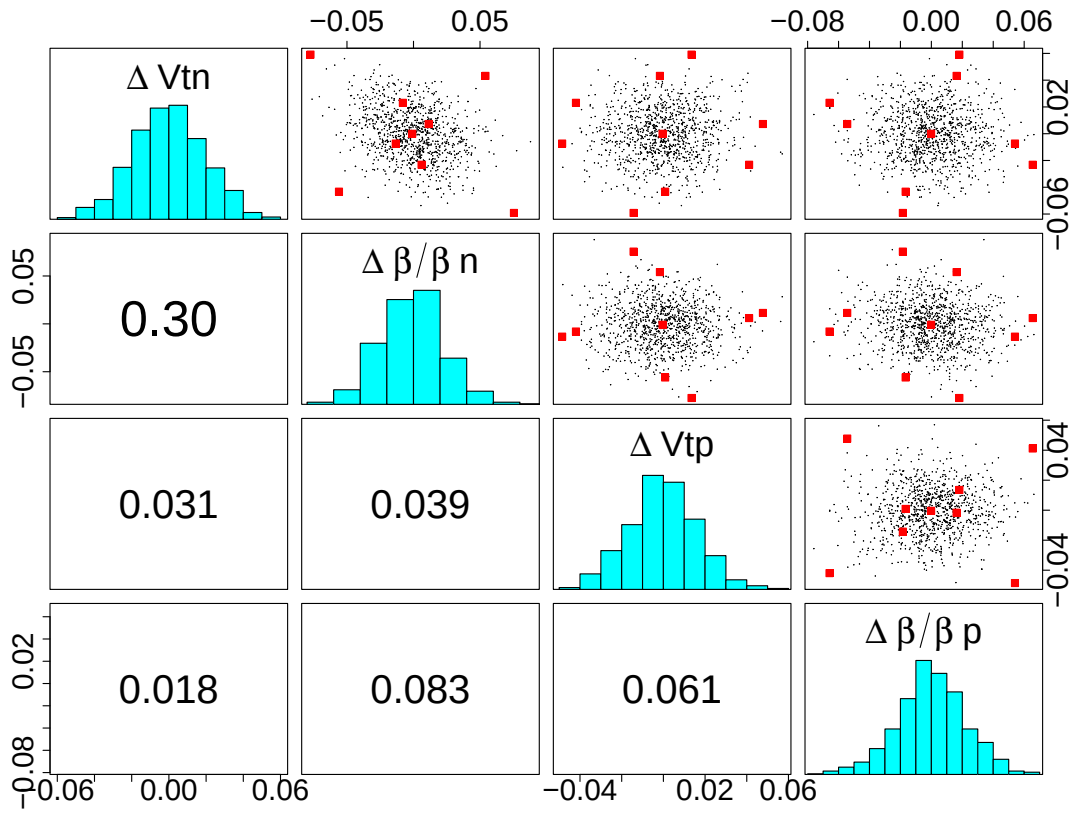


Figure 6.2: Upper-right diagonal: pairwise 2-D distributions and histograms of Vt and β of an inverter. Brussel DoE points are the large squares. Diagonal: histograms of Vt and β. Lower diagonal: correlation coefficients.

where R is the rotation matrix and E is the diagonal matrix of Eigenvalues as in:

$$E = \begin{pmatrix} e_1 & 0 & \cdots & 0 \\ 0 & e_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & e_n \end{pmatrix} \tag{6.9}$$

and the vector of eigenvalues is given by $\vec{e} = [e_1, e_2, \ldots, e_n]_n$. Thus, by substituting eq. (6.8) in eq. (6.4) the covariance matrix S of each cluster component can be decomposed as:

$$S = \sigma \cdot R^T \cdot E \cdot R \cdot \sigma \tag{6.10}$$

This decomposition describes a rotation of the variables into an equivalent set of *independent* Studentized variables t_p:

$$\vec{t_p} = \frac{R \cdot \frac{\vec{t} - \vec{\mu}}{\vec{\sigma}}}{\sqrt{\vec{E}}} \tag{6.11}$$

In the PDF contour plot, the orientation of the rotated standardized axis system corresponds with the principal axes of the ellipsoid contours of the multivariate PDF description.

$$f(\vec{t}, \vec{\mu}, S) = \frac{e^{-\frac{1}{2} t_p \cdot t_p^{\intercal}}}{(2\pi)^{\frac{n}{2}} \sqrt{|E||\sigma|}} \tag{6.12}$$

where $\sqrt{|S|} = |\sigma|\sqrt{|E|} = \prod_{j=1}^{n} \sigma_j \sqrt{\prod_{j=1}^{n} e_j}$.

In a uni-variate context, the $3{\cdot}\sigma$ limits of a standard Normal distribution encompass 99.73% of the total Cumulative Density Function(CDF). An extension of this concept is to find the contour of the underlying PDF encompassing a given percentage, as for instance 99.73% or higher as specified by the designer, of the total CDF. The χ^2 distribution with ν degrees of freedom gives the distribution of sums of squares of ν values sampled from a normal distribution, so its CDF can be used to find the total probability covered by a hyper-sphere with a given radius. Thus, the ellipsoid describing the contour encompassing a specified percentage of the total distribution is defined by back-transforming the hyper-sphere with the radius defined by the inverse CDF of the χ^2 distribution:

$$q\chi^2 = \sqrt{2 f_\Gamma^{-1}\left(\frac{n}{2}, 0, p\sigma\right)} \tag{6.13}$$

where f_Γ is the Regularized Gamma Distribution, $p\sigma = \int_0^{l^2} \chi^2(t,\nu)dt$ with $\nu = 1$ (one dimension) and l refers to how many σ from the center the designer wants to be confident on the outcome, i.e. $l \cdot \sigma$. If $l = 3$ then $p\sigma \approx 0.9973$. Therefore, in terms of total PDF content, this value is the generalization of the 3σ limits in the uni-variate case.

Next, the corresponding ellipsoid contour in the rotated parameter space is defined by :

$$Ellipsoid \begin{cases} \vec{c} = & [0]_n \\ \vec{r} = & q\chi^2(q,n)\sqrt{\vec{e}} \\ D = & R \end{cases} \tag{6.14}$$

which represents a n-dimensional ellipsoid centered at the origin with semi-axis radii $q\chi^2(q,n)\sqrt{\vec{e}}$ aligned with the direction R, where $q\chi^2(q,n)$ is a the Quantile Chi-Square function for a distance q from the center of the distribution and n degrees of freedom.

The Brussel DOE points are then positioned at the intersects of the ellipsoid principal axes and that PDF contour plus an extra point at the component center. A response surface can be fitted to those points. This approach offers a proper balance between sufficient accuracy and validity over the input variable range required for MC sampling, while still requiring a limited amount of terms in the generic propagation function to be fitted. Figure 6.2 presents the position of the Brussel Design of Experiments methodology for selecting the relevant DOE points according to the statistical inputs of an inverter. In this case, the dataset was approximated using a single-component multivariate Normal distribution because the sources of variation do not deviate from Normality.

6.4 Fitting the Response Surface

The goal of the previous section was to select an ensemble of $Ndoe$ points to run electrical simulations on. Using those runs, we compute an appropriate regression model to relate the statistical inputs to the simulated circuit responses: delay, power, transition times, etc.

Problem definition Let $Y_i = H(\Gamma_i^B)$, for $1 \leq i \leq Ndoe$ be the set of circuit responses corresponding to the $Ndoe$ Brussel Design points. Find the optimal regression model such as

$$F(x_1,\ldots,x_p) \approx H(x_1,\ldots,x_p)$$

where $p = 2n$ so that $x_1 = Vt_1, x_2 = \beta_1, \ldots, x_{p-1} = Vt_n, x_p = \beta_n$, and the function F is a

nonlinear function such as

$$\begin{aligned} F(x_1,\ldots,x_p) &= \alpha_{11}x_1 + \alpha_{12}x_1^2 + \ldots + \alpha_{1z}x_1^z + \ldots + \alpha_{pz}x_p^z \\ &+ \zeta_{1_1 2_1} x_1^1 x_2^1 + \zeta_{1_1 3_1} x_1^1 x_3^1 + \zeta_{1_1 p_1} x_1^1 x_p^1 + \ldots + \zeta_{p_1 p-1_1} x_p^1 x_{p-1}^1 \\ &+ \zeta_{123} x_1 x_2 x_3 + \ldots + \zeta_{pp-1p-2} x_p x_{p-1} x_{p-2} \end{aligned} \tag{6.15}$$

where z is the polynomial degree of the approximation function, α_{ij} is the coefficient multiplying variable x_i^j, and $\zeta_{i_j k_l}$ is the coefficient multiplying the interaction $x_i^j \times x_k^l$. These coefficients are determined by a fitting procedure such as Least Squares Fit.

Both the **true** function H and the best approximation function F are unknown. The approximation function F will be later employed to *predict* the outputs for all MC combinations of Vt's and β's. For this purpose, using the full form of equation (6.15) as an approximation function would lead to poor fitting and predictions. In cell characterization problems, the vector of coefficients of equation 6.15 is very sparse, which means that few parameters are relevant to the fit.

This section presents an algorithm for searching the space of possible approximations and, without manual intervention or any previous knowledge about the circuit response (delay, power, etc) , provide the best possible non-linear function to approximate that response. The algorithm is divided into the following steps:

1. **Initial Fit:** fit a full linear model to the data;
2. **Variable Screening:** remove negligible terms;
3. **Model Improvement:** interactively add non-linear terms and cross-terms.

The next sections are devoted to going into the details of these steps.

6.4.1 Assessing Model Quality

The model selection algorithm is driven by the optimization of a metric representing the model quality. The residuals (sum of squares), t-statistic, F-statistic and Likelihood of the regression can be used, however these metrics tend to suggest a better model is achieved as the number of parameters increase, what increases the risk of over-fit. To overcome this issue, Schwarz (1978) proposed the Bayesian Information Criterion (BIC). BIC uses the model Likelihood but adds a penalty factor to compensate for the number of coefficients as in:

$$BIC = log(N_{doe})k - 2ln(L(\theta)) \tag{6.16}$$

where k is the number of unknowns, $L(\theta)$ is the Maximum Likelihood of the model θ and N_{doe} is the number of simulations. The Maximum Likelihood Estimation (MLE) method, implemented in most computer algebra systems, can be computed as presented in Venables (2002). Thus, by adding a penalty to the number of coefficients, BIC prioritizes a model with the minimum number of variables so that the regression is meaningful, reducing the risk of over-fitting.

6.4.2 Initial Fit

The first step to search for the best surrogate model is to fit the simplest regression model, which is a linear function with all the terms and no cross-terms, as in:

$$H_i = \alpha_{1_1} x_{1_i} + \alpha_{2_1} x_{2_i} + \ldots + \alpha_{p_1} x_{p_i} + \varepsilon_i \tag{6.17}$$

where H_i is the output of the i^{th}, $1 \leq i \leq N_{doe}$, spice run which has the vector x of inputs. The LS method aims at minimizing the sum of errors given by $\sum_{i=0}^{Ndoe} \varepsilon_i^2$.

6.4.3 Variable screening

Not every variable has an influence on the circuit response. For instance, the rise delay of an inverter is weakly related to Vt and β fluctuations of the NMOS transistor, and thus excluding those terms from the approximation function leads to a better regression model. Models without physical support should be avoided, and thus we propose a method to remove unimportant variables.

The variable screening step is very important for the non-linear model selection algorithm. Non-relevant variables of the model must be removed before non-linearities and cross-terms are inserted into the model mainly because goodness of fit need minimum degrees of freedom in order to be relevant. Degrees of freedom is defined as:

$$df = N_{doe} - N_{coeff} - 1$$

where N_{coeff} is the number of coefficients of the regression. This means that the number of coefficients of the regression model must be smaller than the number of simulations. We estimate that the maximum number of coefficients allowed for the regression must be around $0.5 \times N_{doe}$ and $0.6 \times N_{doe}$.

The variable screening is accomplished by detecting and removing linear terms that have negligible contribution to the circuit response. The listing of algorithm 1 presents the procedure to remove negligible linear terms. The method consists of iteratively checking the model BIC supposing one variable is removed, and then remove the variable for which removal leads to the best BIC. This iteration is performed until the model cannot be improved by removing a variable.

Algorithm 1 Variable screening

```
repeat
  for all variables x_i of function f do
    f_o ← remove term x_i of function f
    if BIC(f_o) < BIC(f) then
      store f_o in list L sorted by BIC(f_o)
    end if
  end for
  f ← pick model from list L with lowest BIC
until model does not improve
```

After executing this procedure, we obtain a linear model with better BIC than the full linear model. This reduced model F is at the same time less complex and a better approximation for H, and thus more suitable for prediction.

6.4.4 Model Improvement

A first order representation of the circuit response is not sufficient for predicting the circuit characteristics with accuracy. Delay, transition time and power of a standard cell have non-linearities and cross-term.

Algorithm 2 lists the procedure for finding the best non-linear model for the circuit response. It takes as inputs the electrical simulations and the reduced linear model. At each

step, three operations are tried: (1) insert a z-order term (linear,quadratic, cubic,...,z^{th}), (2) insert cross-term between two existing terms and (3) remove an existing term. In our experience, $z \leq 4$ gives good results. For each operation, the resulting model is stored in a list ranked by the model BIC. At each step, the operation that leads to the best local BIC is chosen.

As mentioned earlier, the regression model must be the as simple as possible and contain the minimum number of coefficients. Thus, the algorithm must check the number of coefficients of the model, and stop adding variables when the number of coefficients is around $0.6 \times N_{doe}$. Thus the iterative process stops both when no operation leads to model improvement or when the number of coefficients gets to the maximum threshold.

Algorithm 2 Model improvement

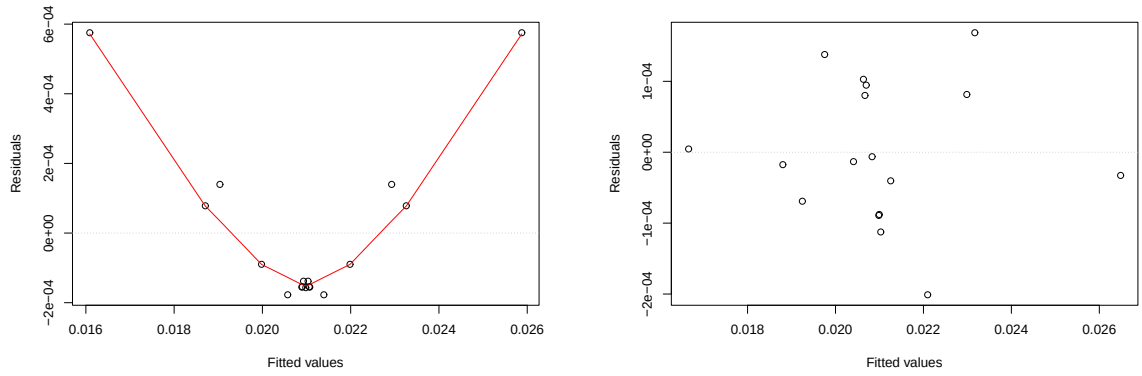
repeat
  for all variables x_i of function f **do**
    for $k = 1..z$ **do**
      $f_{add} \leftarrow$ add term x_i^k
      store f_{add} in list L sorted by $BIC(f_{add})$
    end for
    $f_{remove} \leftarrow$ remove term x_i
    store f_{remove} in list L sorted by $BIC(f_{remove})$
    for all variables x_j of function f **do**
      $f_{cross-term} \leftarrow$ add term $x_i \times x_j$
      store $f_{cross-term}$ in list L sorted by $BIC(f_{cross-term})$
    end for
  end for
  if best BIC stored in L $<$ BIC(f) **then**
    $f \leftarrow$ pick model from list L with lowest BIC
    $N_{coeff} \leftarrow$ number of coefficients of f
  end if
until model does not improve OR $N_{coeff} > 0.6N_{doe}$

Such a greedy algorithm is feasible for a search of a model with a very small number of variables, but it is the simplest possible optimization approach and can be improved further. The bottleneck is that for each iteration, one new regression model must be fitted to the data and the BIC must be calculated. Being N the number of variables, this algorithm requires $O(N^2)$ iterations when allowing quadratic order terms and cross-terms, impplying $O(N^2)$ runs of the Least Squares algorithm.

Figure 6.3 presents the comparison between the initial full linear model and the best model found using the optimization loop, in the case of the delay of a logic gate. The residuals of the linear model present a U-shape curve, which means a mismatch in the tails and is an indication of using the wrong regression model.

The non-linear model presents a satisfactory fitting: it is constant near 0 over the output domain with few outliers in the middle of the domain. Also, the maximum residual of the non-linear model is smaller than the linear model (1.5×10^{-5} instead of 6×10^{-4}) and especially the tails have a better fit.

Figure 6.4 shows the distributions of the residuals of the full linear model and the non-linear model. The residuals of the non-linear model follow a Normal distribution and those of the linear one does not. The non-linear model found presents two advantages



(a) (b)

Figure 6.3: Fitted values and residuals of (a) full linear model and (b) nonlinear model proposed by the optimization algorithm .

over the first one: (1) the maximum residual is 1 order of magnitude smaller and (2) the residuals occur in both directions.

The final step of the methodology consists of running the full Monte Carlo simulation interpolating over the function approximating the electrical simulations, i.e. compute the statistics of $F(x), \forall x \in \Gamma^M$. The complexity of applying one input vector to function F is O(1) and is many orders of magnitudes faster than running one electrical simulation.

6.5 Discussion

This chapter introduces a Response Surface Methodology which is suitable for substituting statistical simulation through Monte Carlo at electric level. The flow is composed by these phases:

Design of Experiments selects artificial points which are representative of the Monte Carlo simulation;

Run electrical simulations on the pre-selected points

Model improvement iterates over possible non-linear regression; models to represent the cell characteristic as a function of its random variables;

Use the surrogate model instead of electrical simulation in order to perform Monte Carlo simulation.

The most computer intensive tasks of our RSM are the electrical simulations and the model improvement algorithm. As compared to Monte Carlo, the number of electrical simulations required by RSM can be reduced by orders of magnitude.

The model improvement algorithm, however, can potentionally jeopardize the speedup. In order to prevent long runtime of the model improvement algorithm, an accuracy switch has been added to the RSM script, as discussed in Appendix B. Lower numbers for this option stops the algorithm to use cross-terms and high order terms, further speeding up the model search. A good tradeoff between accuracy and speed is to set accuracy_fit as 2 or 3. Option 2 searches for a reduced linear model, in other words it runs the variable

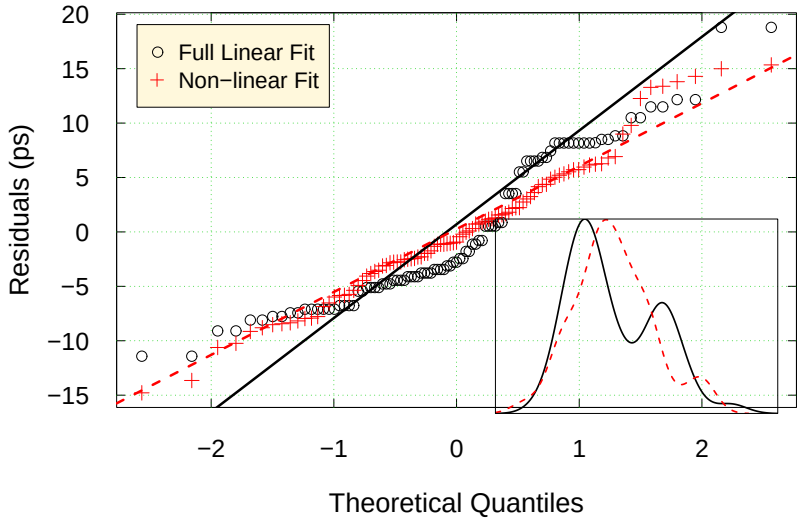


Figure 6.4: Distribution of residuals of the full linear fit and the non-linear fit with quadratic and cross-terms of the FF. The residuals of the non-linear model are smaller and the distribution is closer to a Normal distribution.

screening algorithm (algorithm 6.4.3) only. Option 3 runs algorithm 6.4.3 and the model improvement algorithm (listing 6.4.4), allowing quadratic terms and linear cross-terms.

7 STATISTICAL CELL LIBRARY

This section exhibits results of the statistical characterization of a subset of cells from a 32nm standard cell library, using a statistical compact model library. The selected subset of cells is shown in table 7.1. The library generated is compatible with Synopsys Liberty Format (.LIB), also supported by Cadence tools. For performing the most comprehensive benchmark we performed statistical library characterization by the 3 means:

1. *Monte Carlo:* reference method with a sample size of 1000;
2. *Error propagation using linear sensitivity analysis:* enabled by the commercial statistical library characterization tool which requires $n+1$ runs and performs sensitivity analysis;
3. *Non-Linear RSM:* the proposed regression-based methodology using the Brussel DoE and model improvement algorithms, which requires $2n+1$ runs.

Both the sensitivity analysis and RSM have a number of runs dependent on the number of transistors of the circuit. The number of Monte Carlo runs, on the other hand, is not dependent on the number of random variables but on the target accuracy. Roughly the error of a Monte Carlo simulation scales with $O(\sqrt{1/N})$, meaning an accuracy improvement of ≈ 3 is achieved when the number of simulations is increased by a factor of 10.

Thus, the number of simulations required by Monte Carlo will be kept constant, as a reference, through the benchmark. On the other hand, the number of electrical simulations of linear sensitivity analysis and RSM are a function of the number of variables (referred as n) of the cells. Table 7.1 shows the number of transistors and the respective number of variables of the cells in the cell library. The number of variables is twice the number of transistors because for each transistor we consider two random variables: Vt and $\Delta\beta/\beta$.

We set up an experimental framework to allow fair comparison between Brussel DoE and linear sensitivity analysis. Both responses given by RSM and the error propagation

Cell	Number of transistors	Number of variables
INVERTER	2	4
NAND	4	8
NOR	4	8
XOR	10	20
Flip-Flop type D	24	48

Table 7.1: Number of transistors of the benchmark cell library.

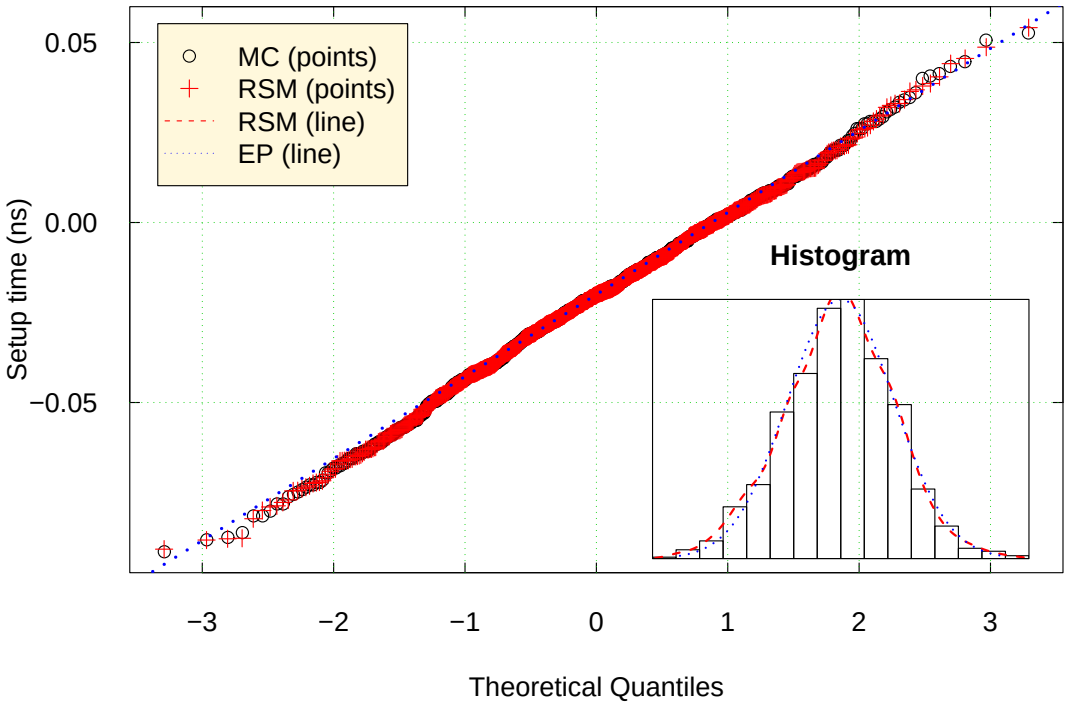


Figure 7.1: Quantile-quantile plot (Normal distribution is a straight line) of the distribution of FF setup time comparing true response computed using MC with linear sensitivity analysis (EP) and RSM. Inset shows histogram and PDFs.

are compared against MC simulation using a sample size of 1000. Thus, we categorize the benchmark into two groups. In a first moment we benchmark the response distributions, arriving at the conclusion that our method is on average $5-10\times$ more accurate for estimating the 4 moments of the distributions than linear sensitivity performed by a state-of-the-art commercial tool.

7.1 Analysis of circuit response PDF and CDF

In this section we compare the response distributions obtained using the proposed methodology with the one given by the commercial tool and the one obtained by MC. Figures 7.1, 7.2 and 7.3 present, respectively, the distributions of setup time, hold time and clock-to-q delay of the flip-flop cell (FF). In the insets, these figures show the histogram of MC (reference), as well as curves representing the PDFs obtained using MC analysis and the non-linear RSM. Linear sensitivity analysis and the RSM present good agreement with MC. Nevertheless, combined PDF or histogram graphs on a linear scale lacks visual information about the accuracy on the tails. For analysis of the distribution with sufficient accuracy this work uses Quantile-Quantile (q-q) Plot, very widespread amongst the Statistic community. These figures show q-q plots of the hold time, setup time and clk-to-q delay distributions. Using this technique allows us to verify that the non-linear method has a perfect agreement with MC simulations on the whole domain of the distribution: both in the center and the tails. On the other hand, the error propagation using linear sensitivity analysis is less than 1% off on the center, but it becomes more inaccurate on the tails of the distribution.

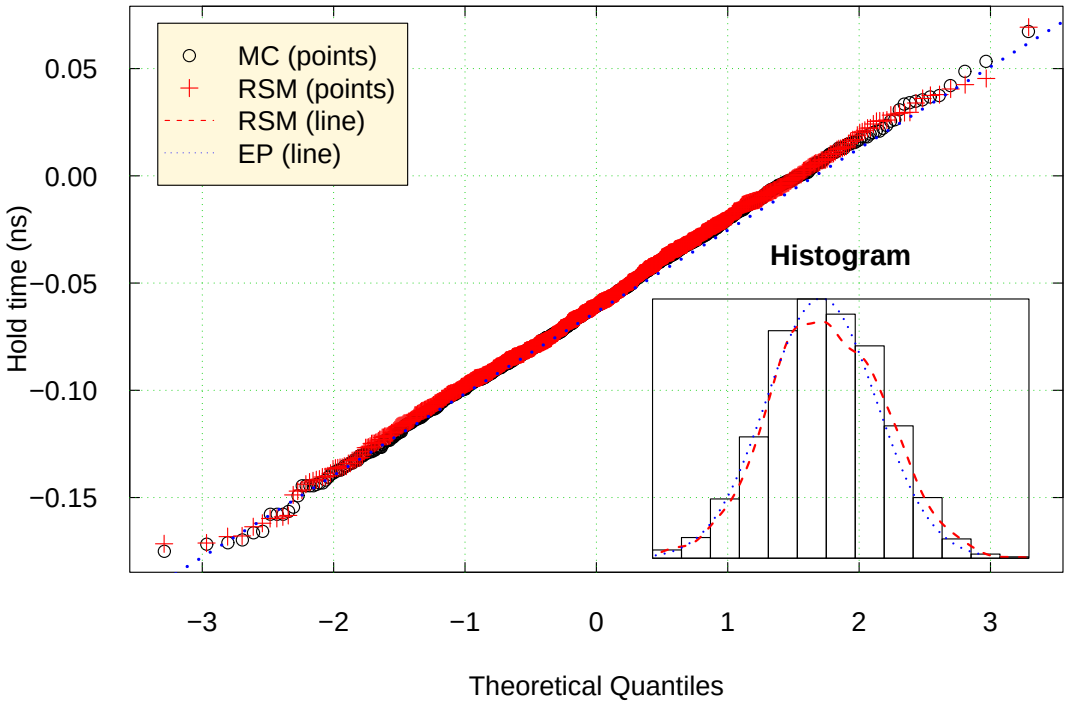


Figure 7.2: Q-q plot of the FF hold time comparing true response computed using MC with linear sensitivity analysis (EP) and RSM. Inset shows histogram and PDFs.

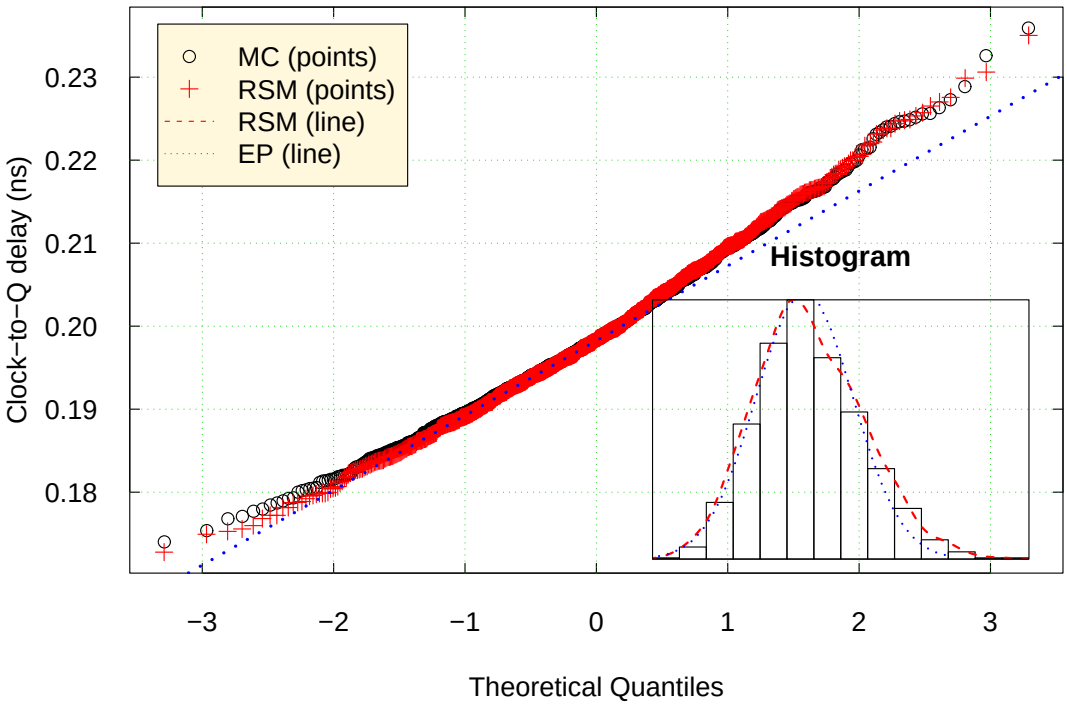


Figure 7.3: Q-q plot of the FF clock-to-q delay comparing true response computed using MC with linear sensitivity analysis (EP) and RSM. Inset shows histogram and PDFs.

Table 7.2: Benchmark of std cell library generated using our methodology as compared to Monte Carlo.

Gate	Param	Response surface methodology						Error Propagation					
		μ_{err} (%)	σ_{err} (%)	S_{err} (%)	K_{err} (%)	$3\sigma_{err}$ (%)	runs	μ_{err} (%)	σ_{err} (%)	S_{err} (%)	K_{err} (%)	$3\sigma_{err}$ (%)	runs
INV	delay	0	0.5	-13.3	0.7	0	9	-0.1	-2.5	-100	2.2	-0.9	5
	transition	0	1.7	-27.4	6.2	-0.2		-0.2	-10.8	-100	-17.9	-2	
	power	-1.6	-0.7	-7.5	0	-1.4		NA	NA	NA	NA	NA	
NAND2	delay	0	0.9	-11.9	-4.5	0	17	-0.7	-17.9	-100	2.2	-3	9
	transition	0	1.5	10.8	4.1	0.5		-0.1	8.8	-100	12.2	0.3	
	power	0.4	-7.5	-1.1	-5	5.4		NA	NA	NA	NA	NA	
NOR2	delay	-0.1	-2.4	-25.8	3.9	1	17	0	-4.6	-100	-6.8	3.7	9
	transition	0.1	3.1	-26.5	-3.1	0		0	-4.3	-100	-3.4	-1.2	
	power	-0.6	3.8	22.8	-0.9	0		NA	NA	NA	NA	NA	
XOR2	delay	0.1	0	16.8	0.7	0.1	41	0.2	-6.4	-100	5.4	-0.5	21
	transition	0	3.4	-28.3	1.5	0.1		-0.2	-8.6	-100	8.4	1.5	
	power	0	-5.5	-43.5	4.1	1.7		NA	NA	NA	NA	NA	
DDFQ	hold	-0.8	1.1	-10.7	2.2	3.9	97	5.5	-3.1	-100	4.1	11.3	49
	setup	0.4	0	-33.1	0.2	1.1		-3.9	-1.6	-100	-1.9	2.3	
	CLK-Q	0	1.4	-8.6	-1	0.1		-0.5	-7.5	-100	-1.5	2.7	
	power	0	2.5	-22.4	-5	0		NA	NA	NA	NA	NA	
||Average|| (timing)		0.14	1.45	19.38	2.55	0.64		1.04	6.92	100	6	2.67	

The best way to compare the responses of the cells of the case study library characterized using RSM and linear sensitivity analysis methodologies as compared to those of the reference library characterized using Monte Carlo is by comparing the moments of the distributions generated using the approximation approaches with those of the reference MC, as given in Table 7.2. For each parameter, the table shows the relative error between the four first central moments of the distributions: mean (μ_{err}), standard deviation (σ_{err}), skewness (S_{err}) and kurtosis (K_{err}), which indicate the degree of asymmetry and the tail weight of the resulting PDF's respectively. In addition, the table presents $3\sigma_{err}$, which is the error of the approximations at 99.97% of the distribution. It shows the quality of the approximation at the tail of the distributions.

The last line of Table 7.2 presents the average of the errors for delay and transition times. Notice that the table presents the absolute errors. Also notice that power is not taken into account for computing the averages because there is no power information for the error propagation. The errors of **standard deviation** and **mean** are less than 2% for the Response Surface Methodology, as compared to errors of 7% when using linear sensitivity analysis. Notice that the linear sensitivity analysis is also limited to Gaussian distributions, and so by definition its output is always limited to $S = 3$ and $K = 0$. Also notice that the commercial tool does not compute statistical estimates of power. Thus, although the error of kurtosis and skewness can be high even for RSM, it performs better than linear sensitivity analysis.

The column "runs" is the number of electrical simulations required by RSM and linear sensitivity analysis. It is important to notice that for these methods the number of simulations is linearly dependent on the number of gates of the device. The electrical simulations are the most time consuming step of Monte Carlo, linear sensitivity and RSM. Also, one simulation takes exactly the same amount of time for each of these methodologies. Hence, the number of electrical simulations is the most representative metric of performance. Roughly, 10 times less runs implies 10 times less run-time.

The number of simulations required by linear sensitivity and RSM scale linearly with the number of variables: $n+1$ and $2n+1$, respectively, being n the number of variables. However, the number of Monte Carlo simulations is independent on the number of inputs. Our Monte Carlo of reference uses sample size of 1000, but this number can be increased for higher accuracy.

The run-time of linear sensitivity and RSM increase with the number of transistors, in such a way that the speedup is inversely proportional to the gate complexity. This limits the applicability of RSM for circuits having less than N/4 transistors, where N is the number of Monte Carlo simulations. Linear sensitivity analysis, on the other hand, can be used to circuits with up to N/2 transistors.

7.2 Analysis of errors of the predicted values

While section 7.1 focuses on the differences between the statistics of the output distributions, it does not show information about the individual errors of the values predicted by the models as compared to the reference electrical simulation. On the other hand this section presents a study about the accuracy of the models when predicting individual values. Figure 7.4 presents the distribution of the relative errors (computed as $model/reference-1$) produced by linear sensitivity and Brussel Design as compared to MC. The linear regression model has discrepancies up to 3% compared to MC, and the mean error is ≈ -0.5, what causes the distribution of errors to be off to the left side of the reference, as shown in the inset of the plot, indicating systematic error (caused by the simplistic linear model). The distribution computed using the non-linear RSM model has a maximum error of 2% and the average error is 0. Moreover, the errors of the RSM follow a Normal distribution centered at 0, which means there are no systematic causes of discrepancies. The conclusion from this figure is that RSM can achieve better accuracy to predict both the central moment of the distribution as well as its tails.

7.3 Runtime analysis

The most time consuming step of the flows is the electrical simulation needed to characterize the standard cells. The runtime of Monte Carlo depends on the target error margin and does not depend on the number of devices. Designers usually employ 10^2 – 10^4 runs.

Unlike Monte Carlo, both the error propagation and the RSM have linear dependency on the number of random variables. RSM requires $2n+1$ runs, while first order sensitivity analysis requires $n+1$ simulations. Thus, when comparing runtime of the electrical simulation only, RSM has a penalty of $2\times$ as compared to linear sensitivity analysis.

Each characterization of the cell library takes approximately 3 minutes running on a server using 10 processors in parallel. The most timing consuming cell is the flip-flop, which takes about 90% of the characterization time of this subset. Thus, characterizing the subset of the standard cell library using the MC approach with 1000 runs (reference) takes three minutes multiplied by 1000, totaling 49 hours. Using the same parallel environment, the characterization takes only a fraction of that time: 2 hours for the linear sensitivity analysis and 4 hours for the non-linear RSM. Notice that for RSM the total runtime is not taken into account: since only the number of electrical simulations is reported, the overhead of RSM, e.g. selection of points and the model selection, is not accounted for.

For characterization of standard cell libraries, which timing and power characteristics has nearly linear relationship with Vt, linear sensitivity analysis offers better tradeoff

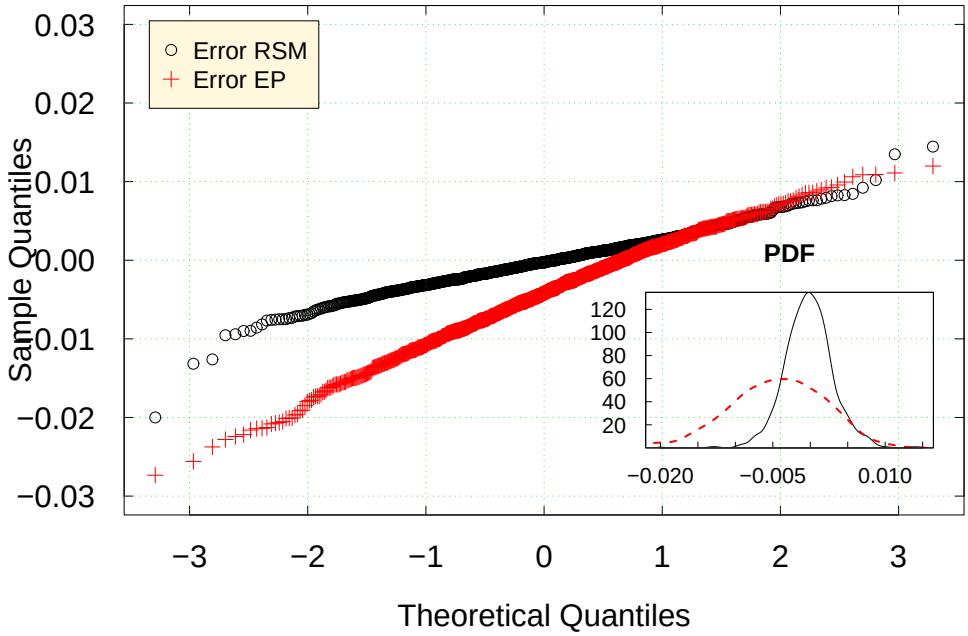


Figure 7.4: Error of linear sensitivity and RSM approaches as compared to Monte Carlo using 1000 runs.

between accuracy and runtime than Monte Carlo and Response Surface Methodology. Linear sensitivity analysis tends to be the best approach when linearity can be assumed. RSM can be a good approach to model non-linear behavior with a small number of variables, e.g. analog circuits. Monte Carlo is a good approach to model non-linear behaviour with large number of variables.

7.4 Impact of aging on the delay of standard cells

This section presents the analysis of the impact of aging due to NBTI on the delay of an inverter. The inverter is simulated using the 32nm Predictive Technology Model (CAO; MCANDREW, 2007).

The time delay of the inverter can be written as a function of the electrical characteristics of the transistors as $Td(Vt_p, Vt_n, ...)$ where Vt_p and Vt_n are the voltage threshold PMOS and NMOS transistors and follow Normal distributions. While the NMOS transistor is affected only by process variation and not affected by NBTI, thus $Vt_n = Vtn_{process}$, the PMOS transistor is affected both by process variations and NBTI, thus $Vt_p = Vtp_{process} + Vtp_{NBTI}$. The standard deviation of the inverter delay can be computed through error propagation (PARRAT, 1961):

$$\sigma^2_{Td}(t) = \left(\frac{\partial Td}{\partial Vtp}\right)^2 \sigma^2_{Vtp_{process}} + \left(\frac{\partial Td}{\partial Vtn}\right)^2 \sigma^2_{Vtn_{process}} + \left(\frac{\partial Td}{\partial Vtp}\right)^2 \sigma^2_{Vt_{NBTI}}(t) \qquad (7.1)$$

This means that both the transistors are affected by process variation, while only the PMOS is affected by the NBTI component. These 2 components are considered to be independent: it is assumed to exist no correlation between process variation (RDF, LER,

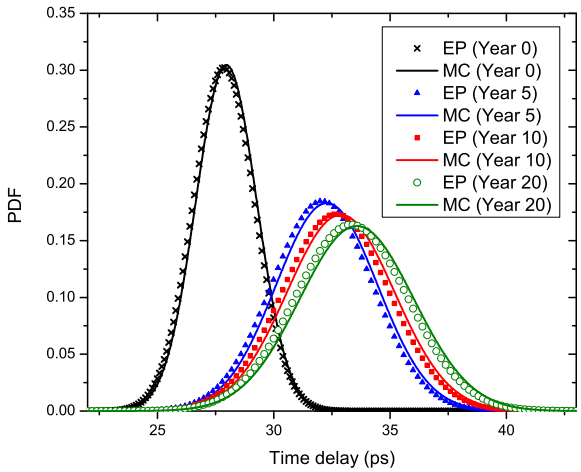


Figure 7.5: Delay PDF as function of time, for both Monte Carlo and Error propagation. Solid lines refer to Monte Carlo results, while the symbols refer to error propagation results.

...) and NBTI.

Figure 7.5 presents the Probability Density Functions (PDF) generated using the mean and standard deviate evaluated by EP, compared to the ones obtained by MC simulations. The solid lines refer to Monte Carlo runs while the symbols refer to the error propagation. The figure shows the Normal PDFs for year 0, 5, 10 and 20. At each year both the mean and the standard deviate of the delay increase due to NBTI. The dimensions of the transistors are Ln=Lp=32nm and Wn=Wp=48nm.

Figure 7.6 shows the histogram obtained from MC simulation compared to the PDF plotted using σ_{Td} and μ_{Td} computed using the proposed approach at the 5th year of operation of the circuit. The EP curve fits very well to the MC histogram.

The table 7.3 presents the results for year 0, i.e. right after circuit fabrication, where the Vt variations are only due to process variability, until year 20, showing the evolution of the delay degradation due to NBTI. The σ_{Td} computed using the methodology proposed in this paper (EP) is compared to σ_{Td} computed by Monte Carlo. The columns Err refer to the difference between EP and MC results (in percent). For each year, we run MC with 1000 simulations. For computing σ_{Td} error propagation requires only 4 electrical simulations: 2 for computing $\partial Td/\partial Vt_{PMOS}$ and 2 for computing $\partial Td/\partial Vt_{NMOS}$. The maximum absolute error of the linear sensitivity analysis approach as compared to Monte Carlo reported in these simulations is 1%. The mean of MC is compared to the simulation using the nominal values of Vt for each year (for which only 1 electrical simulation). The maximum error using this approach for approximating the mean value is 0.7%. Then, EP methodology requires only 5 simulations for computing the mean and standard deviate. Hence a speedup of 200 times is achieved as compared to MC.

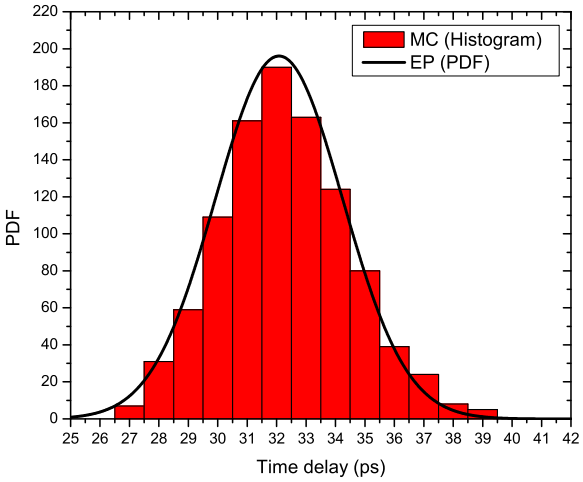


Figure 7.6: Histogram computed by Monte Carlo compared to PDF computed by Error Propagation.

Table 7.3: The σ_{Td} and μ_{Td} computed using MC compared to the methodology proposed

Year	σ(EP) (ps)	σ(MC) (ps)	Err (%)	μ (EP) (ps)	μ(MC) (ps)	Err (%)
0	1.32	1.31	0.1	27.86	27.96	-0.4
1	1.95	1.96	-0.7	31.01	31.18	-0.6
2	2.03	2.05	-0.6	31.43	31.61	-0.6
3	2.09	2.10	-0.8	31.70	31.89	-0.6
4	2.13	2.15	-0.9	31.91	32.11	-0.6
5	2.16	2.18	-1	32.08	32.28	-0.6
10	2.30	2.31	-0.4	32.66	32.88	-0.7
15	2.37	2.39	-0.6	33.04	33.27	-0.7
20	2.43	2.45	-0.8	33.33	33.57	-0.7

8 STATISTICAL ANALYSIS OF HOLD TIME VIOLATIONS

One of the major goals of a design flow is to satisfy timing constraints without sacrificing area and power. One of the most important tasks of design optimization is to identify and remove setup and hold time violations.

Two logically adjacent FFs (namely FF1 and FF2) controlled by CLK1 and CLK2 with no logic or with a fast data path between them may be affected by clock skew. If the clock skew is large enough–i.e. CLK2 arrives after CLK1 and exceeds the internal race immunity of the FF–a hold time violation is produced and detected if the output of both FFs are of the same value at the same time (Q1(t)=Q2(t)) (SHI et al., 2008). The internal race immunity of a FF is given by

$$R = t_{CLK \rightarrow Q} - t_{hold}$$

where $t_{CLK \rightarrow Q}$ is the clock to Q delay and t_{hold} is the hold time of the FF.

Let the clock skew S be given by $S = t_{CLK2} - t_{CLK1}$, which is the delay difference between the two clock signals, and t_d is the delay of the data signal from output Q of $FF1$ to input D of $FF2$.The following definition describes the timing conditions for a hold time violation:

$$\text{A hold time violation occurs} \iff R - S + t_d < 0 \tag{8.1}$$

Thus hold time violations are dependent of FF race immunity (that is inherent to the FF type and its transistor sizing) and the clock skew of the circuit. Both race immunity and clock skew are susceptible to process variations (MEHROTRA; BONING, 2001; ZARKESH-HA; MULE; MEINDL, 1999; CHEN et al., 2005). Historically they have been modeled as worst-case scenarios, thus leading to excessive pessimism (VISWESWARIAH, 2003). Since short paths are increasingly becoming dominant issues of ASIC design, this work addresses a methodology for the analysis and repair of hold time violations. This work extends the work developed by Neuberger (2007). The contributions of this work are:

- analysis of Monte Carlo simulations presenting the distribution of clock skew of a commercial ASIC design of a 90nm technology node, concluding that under process variability clock skew follows a Normal distribution;

- improvement over the methodology for computing the delay to be inserted in order to fix hold time violations. These analytical equation have been proposed by Roberto da Silva.

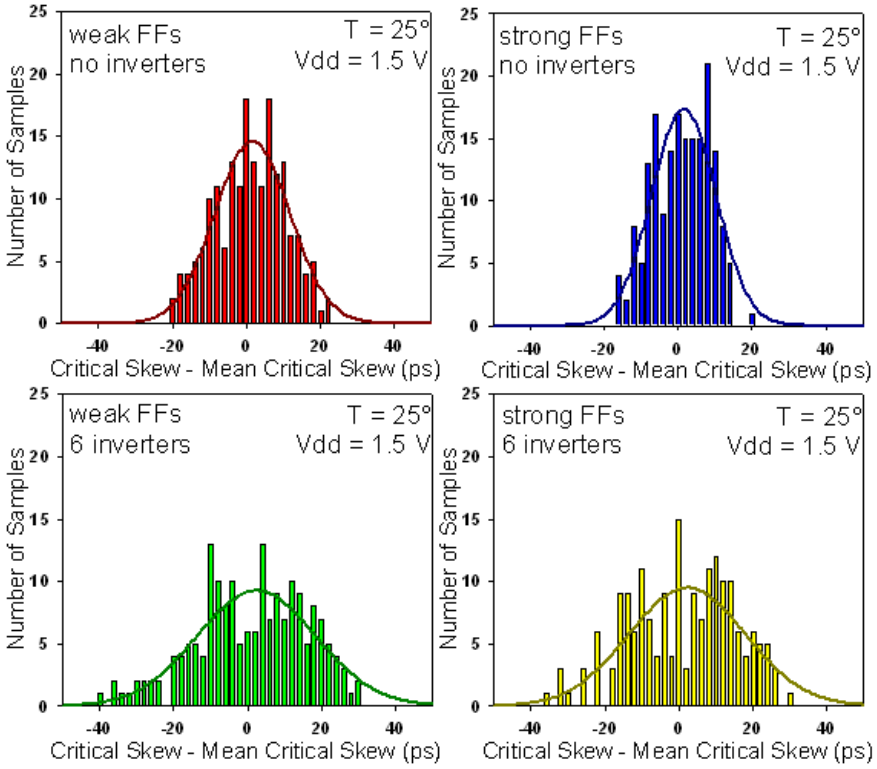


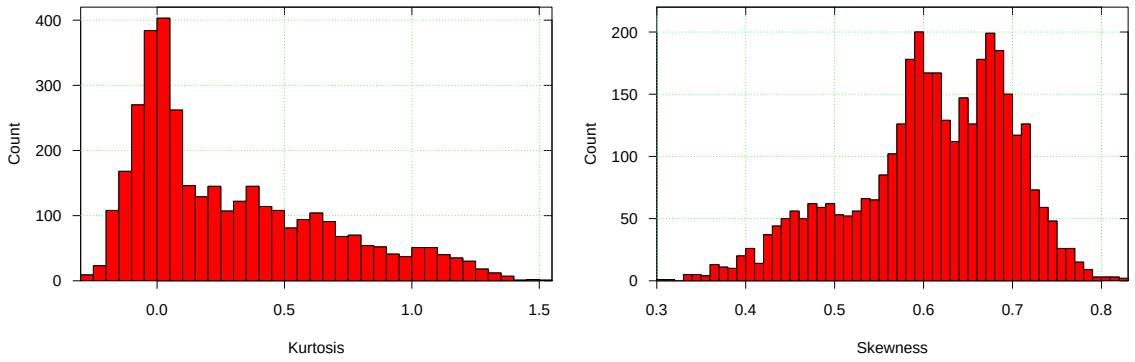
Figure 8.1: Measured distribution of the critical clock skews (race immunity) for rising transitions. The nominal case (mean critical skew) is set to 0ps. Courtesy of Gustavo Neuberger, as appears in Neuberger (2007).

Section 8.1 presents statistical characterization of race immunity made by Neuberger et. al and shows that it can also be modeled as a Gaussian random variable. Section 8.2.2 presents statistical analysis of the clock skew data measured from a commercial standard cell design, showing that clock skew can be modeled as Normal distribution. Section 8.3 presents comparison of three statistical models for hold time violations: i) considering both race immunity and clock skew as worst-case; ii) considering clock skew as worst case and modeling race immunity as a random variable; and iii) modeling both race immunity and clock skew as random variables. Section 8.4 presents an extension of the work developed by Neuberger (2007), statistical method for computing the exact delay that is required to be inserted in the data-path to fix hold time violations.

8.1 Race Immunity: probabilistic approach

Neuberger et al. performed on-chip measurement of race immunity of flip-flops subject to process variations and presented the results in Neuberger (2007). A programmable delay line was developed with resolution of approximately $1ps$. Many experiments were performed on the fabricated circuits to measure race immunity on many dies. The experimental results show that the race immunity can be assumed to follow a Gaussian distribution, with 3σ values of up to 15%, as shown in figure 8.1.

Based on these measurements, the next step would be to estimate the probability of hold time violations taking both race immunity and clock skew into account.



(a) (b)

Figure 8.2: Histogram of the (a) Kurtosis and (b) Skewness of the FFs delays

8.2 Statistical analysis of clock skew

Synchronous digital circuits rely on the distribution of the clock signal from the clock source to the sequential elements of the circuit. Automated clock synthesis tools play a major role in the design of high-performance designs and ASICs. Historically clock synthesis target low area and low dynamic power consumption, power reduction through clock gates, small latency and small deviation from the desired skew (which is usually zero, but when using useful skew the delays at certain ffs is required to be smaller (CHINNERY; KEUTZER, 2002)). But since process variability is playing a major role in skew, synthesizing clock networks insensitive to process variation have become as relevant as other issues(LAM; KOH, 2005).

Zero-skew clock network synthesis tools target at building a clock network which the delays at every sink is the same. Many reasons cause the desired ideal zero-skew to be impossible. The routing of the clock wires impose a challenge: as ffs are distributed over the chip, rare is the case where two wires at least from the bottom-most buffer to the sink can have the same delay. Clock schemes such as H-trees and meshes try to compensate the routing issues, but perfect match is never reached.

The synthesis of the clock network and the analysis of the clock signal delays and skew could be evaluated using corner-based analysis. In this approach the delays are computed using extreme cases of the electrical parameters (transistors voltage threshold, wires capacitances and resistances). Also, corner-based Static Timing Analysis assumes all the devices are completely correlated. Because of these reasons, corner-based analysis is excessively pessimistic (VISWESWARIAH, 2003). This pessimism translates into the latency and skew of the clock network being overestimated, thus the frequency end up being unnecessarily smaller than it could be if the estimates were more accurate.

On DSM technologies WD and D2D variations impose the biggest challenge for the design of reliable clock networks because the variations of the delays of the logic gates and the wires of the clock network have been increasing. These manufacturing variations, together with noise and NBTI, cause the delays d_i to be a random variable, for instance like a Normal distribution, $d_i = N(\mu_d i, \sigma_d i)$. Since the clock skew and the clock latency dictate the maximum frequency of the circuit, the correct estimate of these parameters is essential to verify if the design satisfies its timing constraints and to estimate the yield of

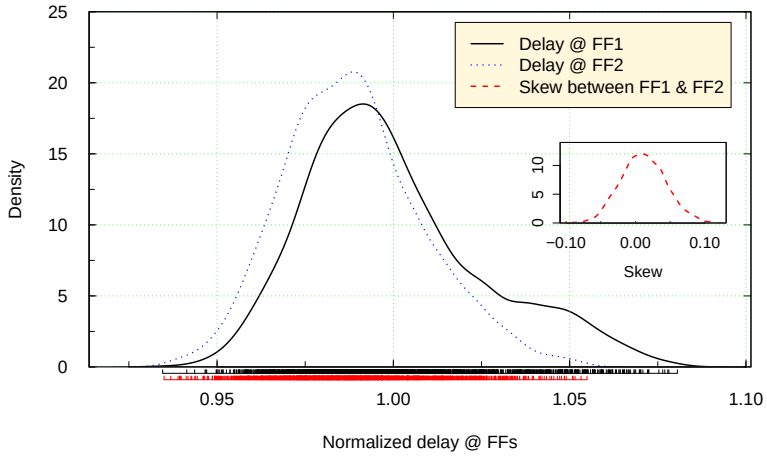


Figure 8.3: Kernel Density of FFs delay (main plot) and skew between them (small plot)

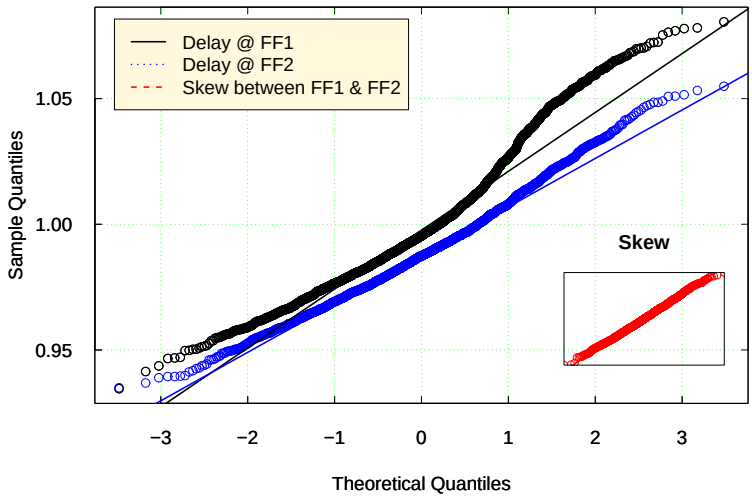


Figure 8.4: Quantile-quantile plot of the delays at the FFs (main) and the Q-Q plot of the skew

the circuit.

Monte Carlo simulation (AMAR, 2006) is appropriate to model the statistical behavior of the circuit imposed by process variations. By performing Monte Carlo analysis instead of corner-based, the WD variations of the devices, which are uncorrelated in nature, lead to paths with uncorrelated delays. This chapter presents in-depth analysis of valuable information given by Monte Carlo simulations of a clock network. Section 8.2.1 presents the analysis of the delays from the clock source up to the FFs, as well as the skew between pair of FFs. Section 8.2.2 reports the clock skew modeled as a random variable.

8.2.1 Delay distribution

Consider the flip-flops ff_i for $i = \{1, \ldots, n\}$, then d_i is the delay from the clock source to ff_i. The clock latency is $latency = max(d_1, ..., d_n)$ and the clock skew between ff_k and ff_j is $skew_{kj} = d_k - d_j$.

Monte Carlo simulations of the clock network of a commercial Fujitsu circuit on the 90nm technology node were run. The clock network was designed using a proprietary

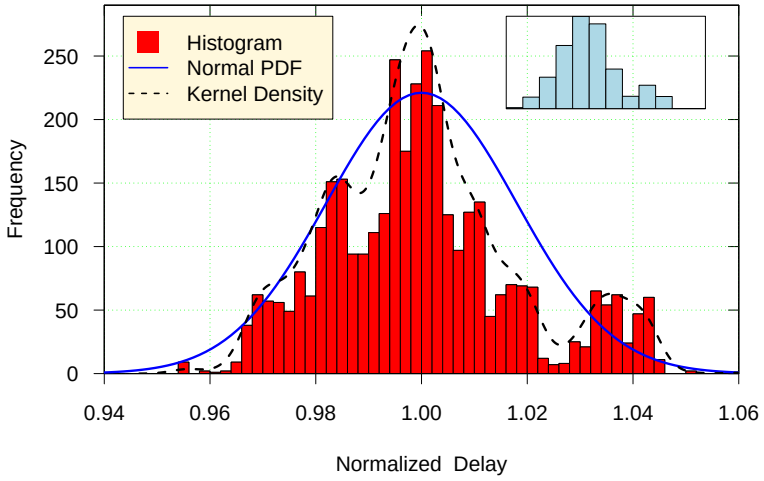


Figure 8.5: Histogram of the delay distribution in one Monte Carlo iteration

Fujitsu clock synthesis tool. The sample size is 2000, and the voltage threshold (Vt) of the transistors is modeled as a Gaussian random variable with μ and σ characterized for the process. The environmental parameters VDD and temperature, and the wire capacitance and resistance are modeled using de-rating factors to represent a slow corner.

First we explore the distribution of the delays from the clock source up to the sinks in respect to normality. Figures 8.2.a and 8.2.b show respectively the kurtosis (PRESS et al., 1992) (normalized to 0 in relation to Normal distribution) and the skewness (PRESS et al., 1992) histograms of the delays at the FFs, considering 2000 MC runs. Nearly 37% of the FFs delays kurtosis are between -0.1 and 0.1, which means those have a tail weight close to a Gaussian distribution. The others have a heavier tail than Gaussian distribution should have, up to 1.5, what is too large for Gaussian. The skewness of the delays vary from 0.3 up to 0.8, which means that the delays of every FF presents an asymmetric distribution which is always positive: skewed to the right tail. Most of them is in the range 0.55-0.75.

Let's then explore in detail the delay of two representative FFs. FF1 has kurtosis=0.08 and skewness=0.69, while FF2 has kurtosis 0.08 and skewness=0.38. The kernel densities of the distribution of (normalized) delays at two FFs is shown in figure 8.3. The plot shows that the delays of these FFs follow a Normal distribution in the left tail and center, but fails to follow it in the right tail. The small portion of the figure highlights the difference between FF1 and FF2, which fits well a Gaussian distribution, even in the tails.

Figure 8.4 shows the Quantile-Quantile plot (q-q plot) of the distribution of delays at the FFs in the main plot and shows the q-q plot of the skew between them in the smaller portion. These plots corroborate to the visual information given by the histogram: the delay at the FFs have a non-symmetry issue in the right tail. But still the difference between them is symmetrical and fits perfectly with a Gaussian Distribution.

Figure 8.5 reports how the delays of the FFs are distributed. It corresponds to one single Monte Carlo iteration, and shows that the delays at most of FFs are very similar, and thus the skew at most of the FFs is very small. The issue is the difference of delays from the maximum delay to the minimum delay, corresponding to the tails of the distribution. This difference imposes the maximum clock skew between two FFs in this iteration, i.e. for a given set of random variables of the Monte Carlo iteration. In this case, the maximum skew of the iteration is $1 - 0.88 = 0.12$.

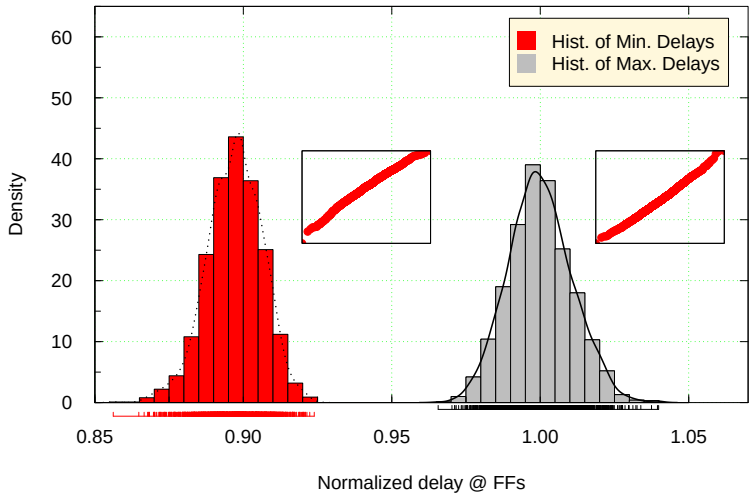


Figure 8.6: Histogram and quantile-quantile plots of the distribution of the minimum and maximum delays

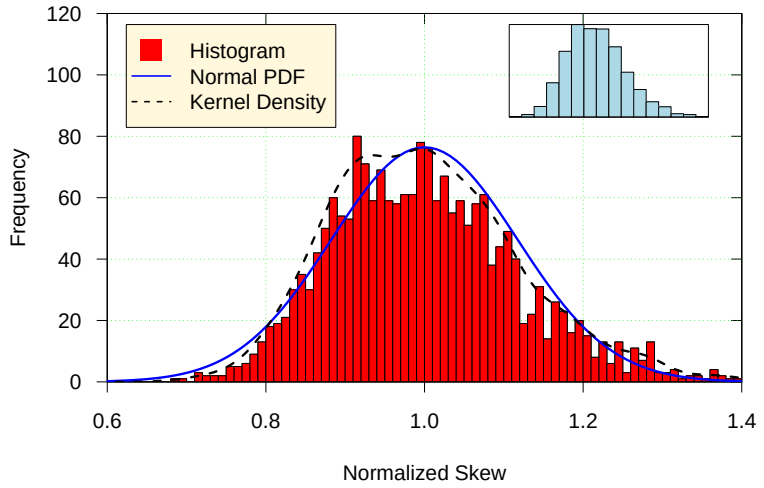


Figure 8.7: Histogram of the clock skew

8.2.2 Maximum clock skew

The clock skew of the design is given by $skew_{max} = max((d_s - d_f)_1, ..., (d_s - d_f)_m)$, for each of the m pairs (ff_s, ff_f) of cascaded flip-flops. The maximum difference between two FFs cascaded (when one is the start-point of the data-path and another as an endpoint) is the most important constraint for the clock synthesis because it is closely related to the maximum frequency of the circuit. Also, smaller clock skew leads to smaller setup and hold time violations, cutting down design time and reducing time-to-market.

Due to process variations, the pair with higher skew and the skew itself is different from one circuit to another, i.e. histogram of figure 8.5 is different for each die. By running Monte Carlo simulation one can simulate the distribution of the minimum delays and the distribution of maximum delays from the source to the FFs. Figure 8.6 presents the distribution of the minimum and maximum delays for MC simulation with a sample size of 2000. Although the delay distribution does not follow a Gaussian distribution, the maximum and minimum of the delays fit very well with a Normal distribution. Figure 8.6

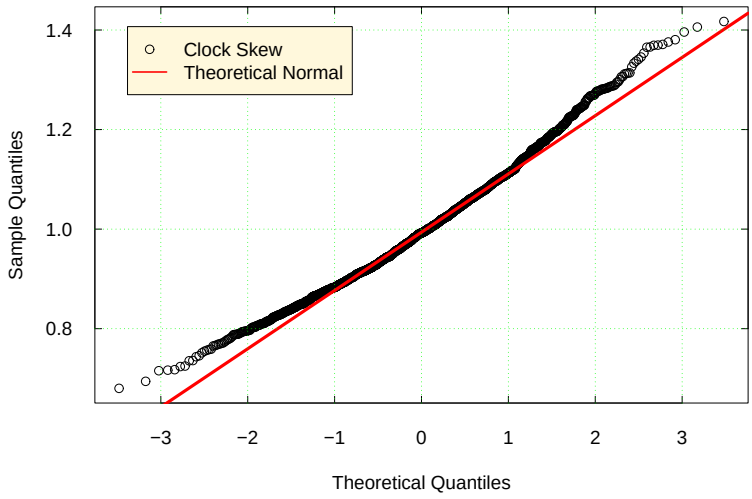


Figure 8.8: Quantile-quantile plot of the clock skew

also presents the quantile-quantile plots for both the distribution of the minimum and the maximum delay. Both the histogram and the quantile-quantile plot visually indicate Gaussianity. We performed a Shapiro-Wilk test for testing the hipothesis of non-Gaussianity of the data. For the maximum, Shapiro Wilk test indicates $W = 0.9978$ and $p = 0.0793$, but for the minimum $W = 0.9953$ and $p = 6.37 \times 10^{-6}$. The tests then indicate that the data of the minimum delays does not seem to come from a Gaussian distribution, while there is no evidences to state that the maximum does not follow a Gaussian distribution.

More important then the distribution of the maximum and minimum delays is the distribution of the skew, which is the difference between them. The distribution of the clock skew of the circuit under analysis is shown in figure 8.7. The figure presents the kernel density, as well with a fit with a Gaussian distribution. In the small portion of the graphic, it is shown an histogram with greater bin sizes, to confirm the good symmetry of the data in relation to a Gaussian distribution. Figure 8.8 shows the quantile-quantile plot of the clock skew. It shows that the data follows a Gaussian distribution in the center of the distribution, but fails to fit in the tails, demonstrating a skewed behavior.

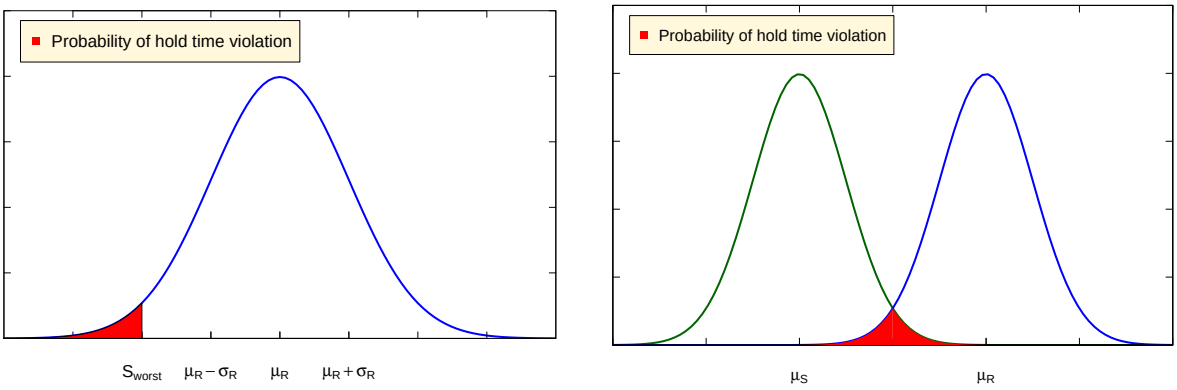
8.3 Models for Hold Time Violations

8.3.1 Hold Time Violation: worst-case approach

A hold time violation occurs when the clock skew is higher than the race immunity. Assuming both race immunity and clock skew are worst-case values, denoted respectively by R_{worst} and S_{worst} –which is the scenario usually found in the literature and supported by EDA tools– with the definition 8.1 for hold time violation, and assuming $t_d = 0$, the probability of a hold time violation is as follows:

$$P_{hold}(S_{worst}, R_{worst}) = \begin{cases} 1 & \textbf{if } S_{worst} < R_{worst} \\ 0 & \textbf{if } S_{worst} > R_{worst} \end{cases} \tag{8.2}$$

In this case S_{worst} and R_{worst} are the worst case clock skew between the two FFs and the worst case race immunity of the FF respectively.



(a) (b)

Figure 8.9: (a) Calculation of hold time violation probability (cumulative distribution function). (b) Hold time violation probability considering clock skew as a random variable following a Normal distribution.

8.3.2 Hold Time Violation: race immunity as random variable and clock skew as worst-case value

If the clock skew is assumed to be a deterministic (fixed) value and the race immunity is assumed to follow a Normal distribution according to Neuberger (2007), the probability of a hold time violation of a short path is the probability of the race immunity, which is less than the clock skew. It is illustrated as the red area in Figure 8.9.a. From the definition 8.1 of hold time violation and assuming $t_d = 0$ the probability of the race immunity (which is characterized by average μ_R and standard deviation σ_R) to be smaller than the clock skew (here denoted by S_{worst}) is given by:

$$P_{hold}(S_{worst}, \mu_R, \sigma_R) = \frac{1}{2}\left(1 + erf\left(\frac{S_{worst} - \mu_R}{\sigma_R\sqrt{2}}\right)\right) \tag{8.3}$$

where $erf(x) = \frac{2}{\sqrt{(\pi)}}\int_0^x e^{-t^2}dt$, S_{worst} is the worst case of the clock skew, μ_R and σ_R are the average and standard deviation of race immunity.

Figure 8.10 shows the probability of failure (z-axis) as a function of the race immunity, where race immunity was modeled as a random variable with a Normal distribution and the clock skew is modeled deterministically (worst case). Since clock skew is considered as the worst case, even if the clock skew is very small–let's say nearly 70ps–a path with an FF with race immunity of 100ps has nearly 100% chance of causing a hold time violation. As we will see in the next section, this approach is pessimistic and if clock skew is also modeled as random the failure probability computed is smaller.

8.3.3 Hold Time Violation: probabilistic approach

Let the race immunity and clock skew be random variables which can be approximated by Gaussian distributions, which is a good approach as shown in the previous sections. The probability of hold time violation is the probability that the clock skew is higher than the race immunity. In this case, we must evaluate the probabilities for the race immunity value (normally distributed) being smaller than the clock skew (also a normally

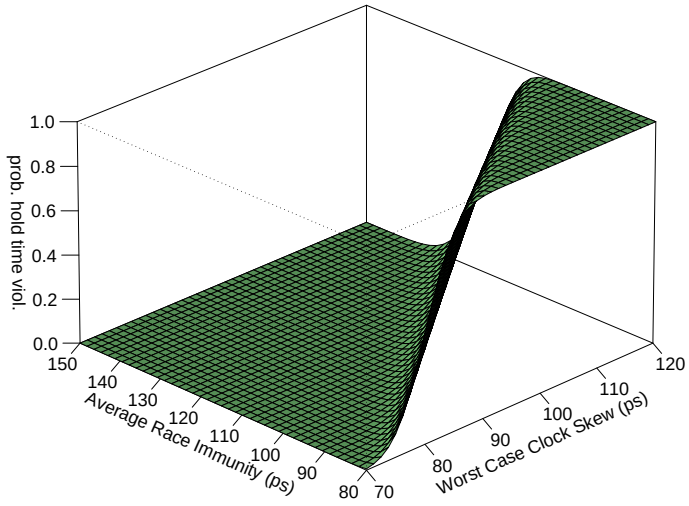


Figure 8.10: Probability of hold time violations (z-axis) as a function of clock skew (x-axis represents the worst case) and race immunity (y-axis represents the average of the Normal distribution)

distributed random variable). This is the convolution of the two Gaussians, also discussed in Neuberger (2007) and is graphically represented in figure 8.9(b).

To evaluate the probability, we will assume that the race immunity and clock skew follow independent normal random variables. This is a valid assumption since the variations we are dealing with come from RDF, which makes the transistors have different electrical characteristics (like V_t) with no correlation. Using this assumption, with the definition of hold time violation given in 8.1 and assuming the $t_d = 0$, the following equation can be used to calculate the probability of hold time violation:

$$P_{hold}(\mu_R, \sigma_R, \mu_S, \sigma_S) = \frac{1}{2} \cdot \left(1 + erf\left(\frac{\mu_S - \mu_R}{\sqrt{\sigma_S^2 + \sigma_R^2}\sqrt{2}}\right)\right) \tag{8.4}$$

where μ_R and σ_R are the average and standard deviation of race immunity, μ_S and σ_S are the average and standard deviation of clock skew, and erf is the error function.

Figure 8.11 reports the probability of failure (z-axis) as a function of race immunity and clock skew modeled as random variables following Normal distribution. The probabilistic model leads to less paths being reported as presenting violations, although circuit reliability and performance constraints are satisfied.

8.4 Fixing hold time violations with probabilistic delay insertion

Increasing the data-path delay t_d by $k$ps for removing hold time violations has the same effect as reducing the average clock skew of $k$ps or increasing the race immunity by $k$ps. The probability of hold time violation as a function of clock skew with race immunity being a random variable with Normal Distribution and with data-path delay t_d as a fixed value is given by:

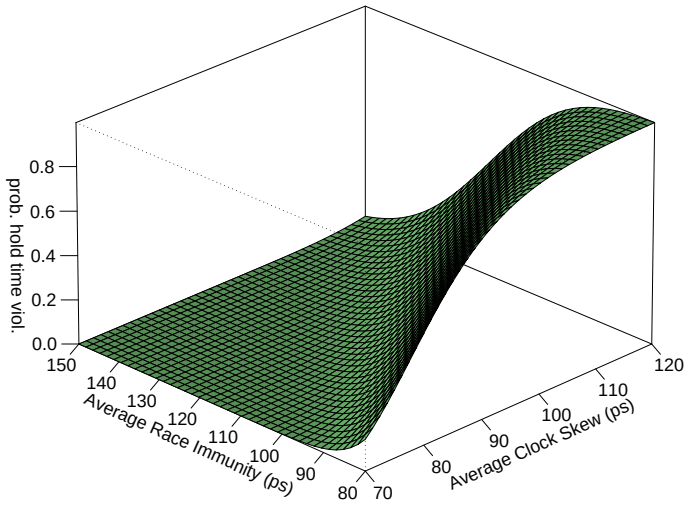


Figure 8.11: Probability of hold time violations (z-axis) as a function of clock skew (x-axis represents the average of the Normal distribution) and race immunity (x-axis represents the average of the Normal distribution).

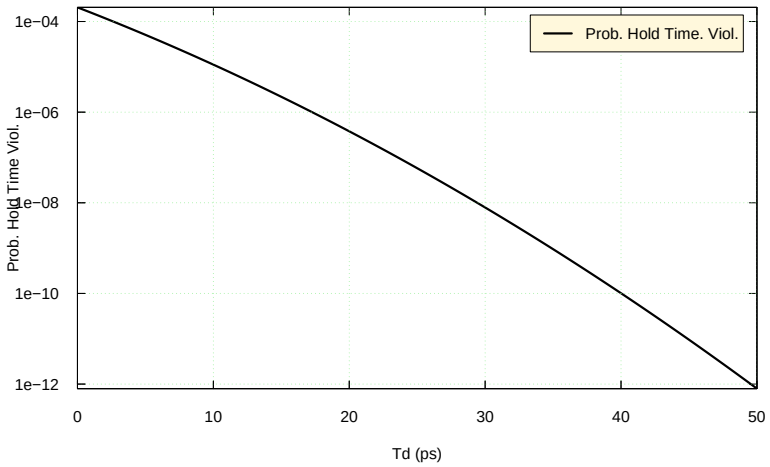


Figure 8.12: Probability of violation as a function of the data-path delay.

$$P_{hold}(\mu_R, \sigma_R, \mu_S, \sigma_S, t_d) = \frac{1}{2} \cdot \left(1 + erf\left(\frac{\mu_S - \mu_R - t_d}{\sqrt{\sigma_S^2 + \sigma_R^2}\sqrt{2}}\right)\right) \tag{8.5}$$

Figure 8.12 shows the dependence of the probability of hold time violation on the time delay of the data-path. This figure represents a path where the clock skew average is 100ps and the FF race immunity average is 150ps. The hold time probability decreases exponentially with the data-path delay. Also, figure 8.12 shows that the probability of hold time violations strongly decrease as delay in the data-path from FF1 to FF2 increases. In a back-end design flow, the timing analysis tool evaluates this delay (referred as negative slack), and if it is not enough–leading to a hold time violation–then an optimization algorithm can insert the appropriate amount of extra delay to avoid that timing violation. This method is known as padding.

Padding is the placement of extra delay in the fast logic paths to increase the race im-

munity and to prevent hold time violations. This extra delay can be achieved by inserting inverters, buffers, or wire jogs. Padding is the most effective way to prevent digital circuits against hold time violations. Padding was presented as a technique to prevent against hold time violations in short paths by Shenoy (1993) and is employed by commercial tools to fix hold time violations. The problem with hold time fixing tools as it is today is that the amount of delay to be inserted in the path is computed with a corner-based approach. In that fixed-values scenario, the amount of delay t_d that padding must insert in a given path can be calculated as:

$$t_{d_{worst}} = S_{worst} - R_{worst} \tag{8.6}$$

where S_{worst} is the worst-case clock skew and R_{worst} is the FF race immunity.

We now aim at finding an expression to compute the total delay t_d to be inserted into the data-path so that the probability $p = P_{hold}(\mu_R, \sigma_R, \mu_S, \sigma_S, t_d)$ is less than a given threshold. Also, assume that we wish the probability of hold time violation to be very small and thus $p \ll 0.1$. In order to accomplish that, we have to isolate the variable t_d in equation 8.5. Roberto da Silva proposed an analytical manipulation to isolate t_d, aiming at a closed-form solution (BRUSAMARELLO et al., 2010). For this purpose, consider the handy numerical approximation for the error function (erf) presented by Winitzki (2003):

$$erf\,(y) = \left[1 - \exp\left(-y^2 \frac{\frac{4}{\pi} + ay^2}{1 + ay^2}\right)\right]^{1/2}$$

where in our case $y = \frac{\mu_S - \mu_R - t_d}{\sqrt{2(\sigma_S^2 + \sigma_R^2)}}$. The best approximation related to the above equation is obtained when we set $a = 0.147$.

Then from equation 8.5 we have:

$$1 - 2p = \left[1 - \exp\left(-y^2 \frac{\frac{4}{\pi} + ay^2}{1 + ay^2}\right)\right]^{1/2}$$

From this we obtain the fourth degree equation:

$$0.147y^4 + \left(\frac{4}{\pi} + 0.147 \ln\left[4p(1-p)\right]\right) y^2 + \ln\left[4p(1-p)\right] = 0$$

where making the substitution $y = x^2$ becomes a quadratic equation:

$$0.147x^2 + \left(\frac{4}{\pi} + 0.147 \ln\left[4p(1-p)\right]\right) x + \ln\left[4p(1-p)\right] = 0$$

Solving this equation in x and returning to $y = x^2$ we find four candidates for the solution. Eliminating the solutions that are not valid in the range of y and p, we verify that the solution for y is:

$$y = -\frac{1}{4\pi}\sqrt{\Delta_1 + \Delta_2}$$

where

$$\begin{aligned}
\Delta_0 &= \ln\left(4p - 4p^2\right) \\
\Delta_1 &= -217.69\pi - 8.0\pi^2\Delta_0 \\
\Delta_2 &= 54.42\pi\sqrt{1.18\pi\Delta_0 - 0.59\pi^2\Delta_0 + 0.02\pi^2\Delta_0^2 + 16}
\end{aligned}$$

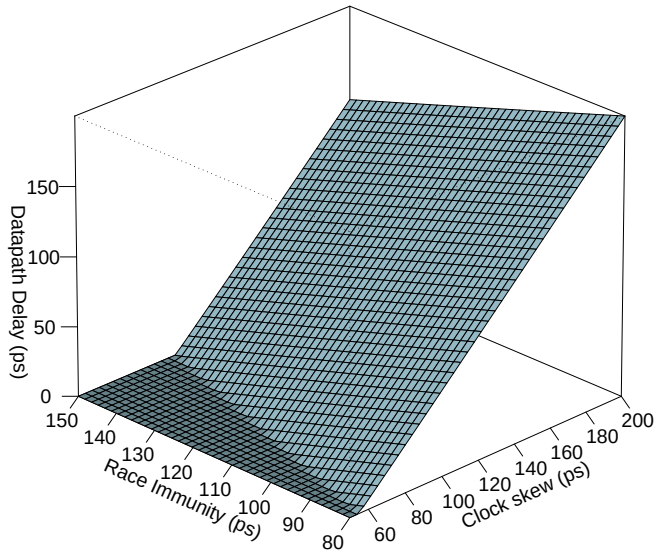


Figure 8.13: Data-path delay required to satisfy the yield constraint due to hold time violations using the probabilistic approach.

And making the substitution $y = \frac{\mu_S - \mu_R - t_d}{\sqrt{2(\sigma_S^2 + \sigma_R^2)}}$. we find the suitable formulation for td:

$$t_d = \mu_S - \mu_R + \frac{\sqrt{2\left(\sigma_S^2 + \sigma_R^2\right)}}{4\pi}\sqrt{\Delta_1 + \Delta_2}$$

In a design containing n short paths, the probability of one path to present hold time violation p is related to the design yield loss due to hold time violations as in $yield = (1-p)^n$. Thus, the probability of hold time violation can be computed from the yield goal as in

$$P = 1 - yield^{\frac{1}{n}}$$

where $yield$ is the circuit yield and n is the number of short paths.

Figure 8.13 shows the computation of td required to be inserted in the data-path using the corner-based analysis, the proposed probabilistic methodology, and the method where skew is considered worst case while race immunity is probabilistic. On both probabilistic scenarios the yield is set to 95% with n=100 paths, which results in $P = 5.13 \times 10^{-4}$. Using worst-case scenario, as opposed to the proposed probabilistic methodology, would not allow for prediction of the circuit yield.

9 STATIC RANDOM ACCESS MEMORY (SRAM)

Nowadays most of ASICs and dedicated high-speed circuits such as microprocessors present a considerable area devoted to Static Random Access Memories (SRAM).

Part of SRAM success is because it can be fully integrated to the logic part of the ASIC: it consists of two inverters and two pass transistors which are processed together with the rest of the chip. For instance this is not the case for DRAM memories, which need a special manufacturing process, mainly due to the load storing the bit. Thus DRAMs must be built in a separate die, not so close to the logic, implying in communication overhead because the data must pass through a bus.

Dynamic RAM also has the side effect of its load discharging to ground after a given amount of time. This would result in bit-flips, and thus a special circuitry, a refresh circuit, must be build in order to refresh the bits of a DRAM. Due to the feedback loop of its two inverters, another advantage of SRAM over DRAM is the lack of need a circuit to refresh the bits. This implies even more speed advantage.

Mainly due to its implementation being so close to logic, SRAM is the fastest high-density storage element existing in today's technology. Although latches and FFs can be actually faster, their area is orders of magnitudes more than SRAM, thus they are not suitable for high-density.

The first level of cache of microprocessors, which requires maximum speed, is always an SRAM memory. The area of a state-of-the-art microprocessor. Nowadays can be up to 80% SRAM memory. Depending on the application, ASICs also can present very high density of SRAM memories. Recent FPGAs also ship with embedded SRAM arrays in order to offer a high-speed memory.

Figure 9.1 illustrates a typical SRAM memory architecture as discussed in Haraszti (2000). This scheme of memory is composed of

- memory cell array composed of N_{COL} columns and N_{ROW} rows of SRAM cells, and N_R redundant columns;
- internal timing circuit to generate the control clocks;
- data-in/data-out buffer circuitry;
- register and decoder blocks for the **bit address**, commonly referred as **row address**;
- register and decoder blocks for the **word address**, commonly referred as **column address**.

SRAM memories present a regular architecture in which most of the chip area is dedicated to regularly disposed SRAM cells. Consider the memory array designed with N_{COL}

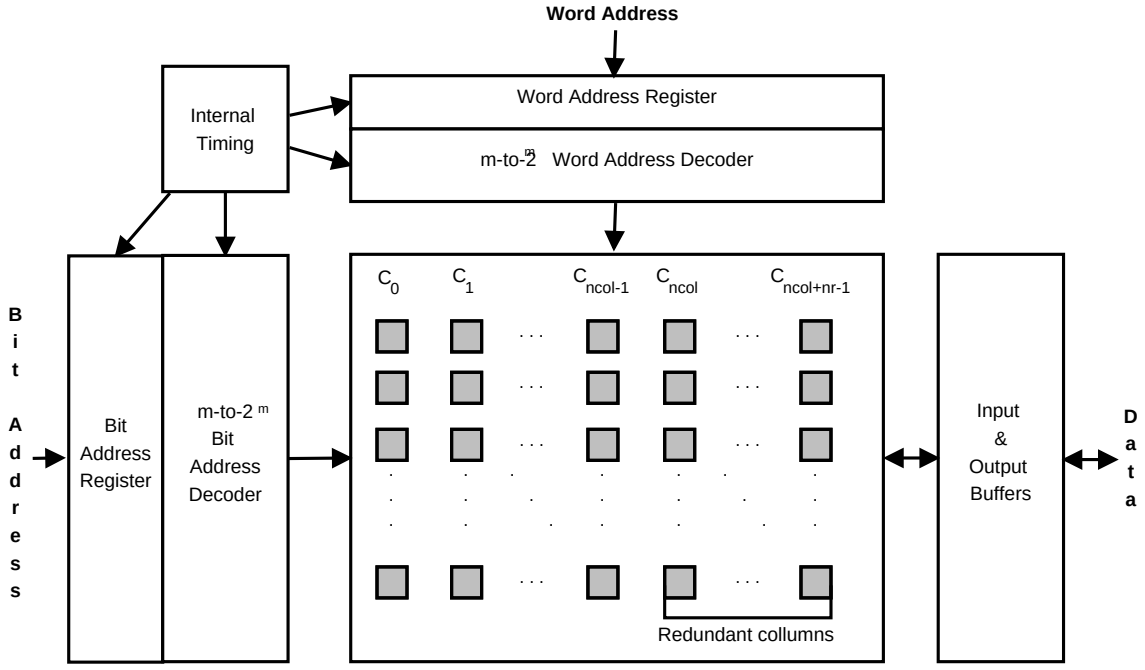


Figure 9.1: Scheme of a SRAM memory

columns and N_{ROW} rows of SRAM cells, and N_R redundant columns. If process fabrication variability causes at least one memory cell to fail in a column, that column is replaced, during circuit test phase, by one of the redundant columns. Some applications employing memories sometimes apply binning techniques, as for instance when more than N_R columns fail, the circuit is re-wired to utilize a reduced amount of memory. Although ASIC designs usually do not have that flexibility and need the full memory to work, relying solely on redundancy. In any case, faulty SRAM cells lead to reduced yield and reduce profit margins.

Denoting p as the probability of the SRAM cell to work properly in the presence of process variability, $P_{COL} = (p)^{N_{ROW}}$ gives the probability of no cell failure per column. In addition, the probability of column to be faulty is given by $Q^i_{COL} = 1-(p)^{N_{ROW}}$. Next, we are interested in the probability of successfully manufacture N_{COL} working columns, in a total of $N_{COL}+N_R$ designed columns. Then, the yield (percentage of working chips) of a SRAM memory design is given by a binomial distribution (MAHMOODI; MUKHOPADHYAY; ROY, 2005):

$$P_{MEM} = \sum_{i=N_{COL}}^{N_{COL}+N_R} \binom{N_{COL}+N_R}{i} (P_{COL})^i (1-P_{COL})^{N_{COL}+N_R-i} \tag{9.1}$$

In order to offer maximum density, SRAM memory cells are usually designed using the smallest feature sizes allowed by the technology. Nowadays SRAM is the component of a digital design that benefits most from technology scaling. For SRAM, technology scaling still guarantees higher density at each new technology node. However, in the sub-100nm regime SRAM design must consider variability and reliability aspects in order to guarantee the reliability of the circuit. The schematic of the most typical design of SRAM, a 6-T SRAM cell, is shown in figure 9.2.

Computer simulation methodologies for analysis of SRAM yield due to process variations have been the topic of much research in the last years. Analysis of yield of SRAM

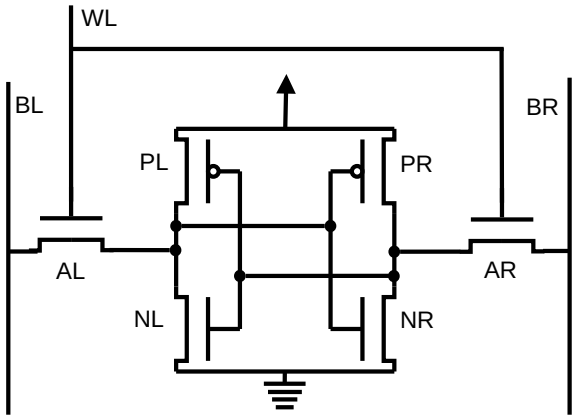


Figure 9.2: 6-transistors SRAM cell

memories using Monte Carlo has been studied by Agarwal (2005). Linear sensitivity analysis at electrical level for yield analysis of SRAM memory has been explored in Mukhopadhyay (2004) and Mukhopadhyay (2004). These works propose statistical models for failures in SRAM cells (access time failure, read failure, write failure and hold failure). The yield of SRAM array can then be computed as a function of SRAM cell yield.

9.1 Failures in a SRAM Cell

Failure probability in a SRAM memory array is given in function of the number of columns, number of rows, number of redundant columns, and the probability of a SRAM cell to work properly in the presence of variability. Failures in SRAM cells are due to:

access time violation when reading the value stored in a cell, *bit* and $\overline{bit}$ are set to VDD, and when *wl* is set to VDD, one of them discharges (through AL-NL or PR-NR). The access time (T_{AC}) is defined as the time required to discharge a secure margin of the bit-line. As the maximum access time (T_{MAX}) is a design input related to chip frequency, violation occurs when T_{AC} of the cell is greater than T_{MAX}. The access time is a non-linear function of Vt, but its inverse can be considered linear (AGARWAL; NASSIF, 2006a).

read failure when reading the content of a cell, *bit* or $\overline{bit}$ discharges (through AL-NL or PR-NR). This causes input of one of the inverters (PL-NL or PR-NR) to be charged to voltage V_{READ}. If maximum V_{READ} is greater then trip point V_{TRIP} of the inverter, read operation will cause the stored bit to erroneously flip. The read failures of a cell can be evaluated by the Read Noise Margin (RNM). Read failure can be modeled as a linear function of Vt (AGARWAL; NASSIF, 2006a).

write failure to write a value to the cell, *bit* and $\overline{bit}$ are set to the proper values, and then *wl* is set to VDD for a time $[T_{WL}$ in order to the signals to be stored in the cell. Consider that signal takes T_{WRITE} to be written to the cell, then it must apply $T_{WRITE} < T_{WL}$, otherwise the signal will not be successfully stored. Although write stability is not linear with Vt, its inverse is so (AGARWAL; NASSIF, 2006a).

hold failure although write and read operations are the most critical moments to the SRAM cell, it can also happen that the cell cannot hold its contents in a stable

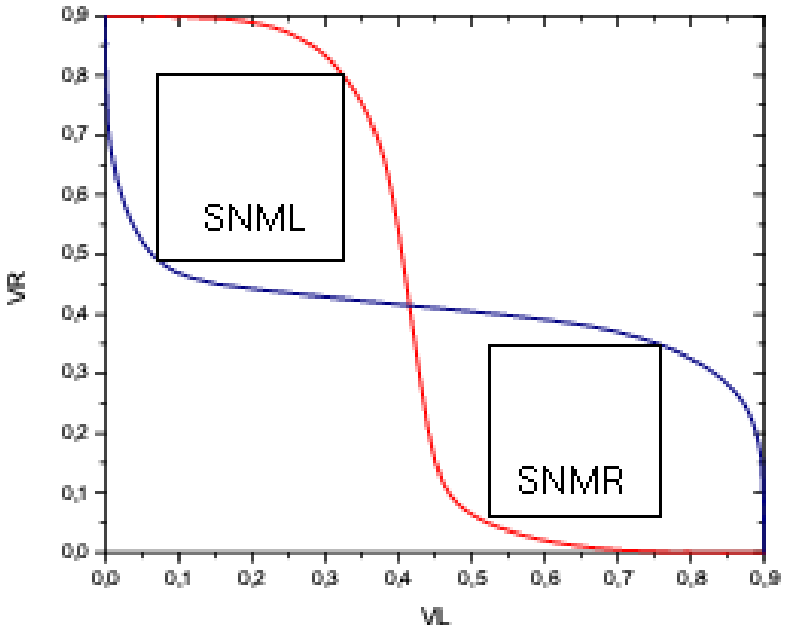


Figure 9.3: Butterfly Curves

manner. The Static Noise Margin (SNM) of the cell can evaluate the cell capability of holding a stable value . SNM can be considered as being a linear function of Vt (AGARWAL; NASSIF, 2006a).

9.2 DC Static Noise Margin (SNM) and Read Noise Margin (RNM)

Noise margin (NM) is the minimum DC voltage that flips the content of the cell once applied to the input of one of the inverters (PL-NL or PR-NR) of the cell. The noise margin is a DC metric, which can be computed by applying a DC voltage to the input of the inverter and analyzing the inverter response. The NM is dependent of the cell operation mode, thus there are two important noise margins:

Static Noise Margin (SNM) : during normal static operation (simply storing the bit), when neither a read or write operation is being performed;

Read Noise Margin (RNM) : read margin during read operation, which is the moment the cell is more vulnerable to failures. Usually RNM is the limiting NM of the cell.

Static Noise Margin (SNM) is computed from the analysis of the butterfly curve of the SRAM cell (BHAVNAGARWALA; TANG; MEINDL, 2001). Figure 9.3 presents the butterfly curve for a SRAM cell designed using 65nm PTM model card with the following sizing: $W_{NL} = W_{NL} = 100nm$, $W_{PL} = W_{PR} = 100nm$, $W_{AL} = W_{AR} = 140nm$. The SNM of this cell is approximately the bottom line of the largest square that can be fit inside the curves, as drawn in the picture. In this case, the SNM is approximately 0.3V.

The design of a SRAM cell is a compromise between read, write and hold stability's and read, write and hold speeds. The sizing of SRAM cell is expressed as the beta-ratio (the ratio between width of transistors N over width of A).

Table 9.1: Output of the electrical simulator, stored in a table as VL and VR.

VR or VL	VR=f(VL)	VL=g(VR)
0	899.5610m	899.5610m
50.00000m	899.2891m	899.2891m
100.00000m	898.2426m	898.2426m
150.00000m	894.7680m	894.7680m
200.00000m	884.5981m	884.5981m
250.00000m	858.6611m	858.6611m
300.00000m	801.0821m	801.0821m
350.00000m	673.6515m	673.6515m
400.00000m	319.3079m	319.3079m
450.00000m	76.7541m	76.7541m
500.00000m	40.7186m	40.7186m
550.00000m	24.2140m	24.2140m
600.00000m	14.0134m	14.0134m
650.00000m	7.3248m	7.3248m
700.00000m	3.2791m	3.2791m
750.00000m	1.2485m	1.2485m
800.00000m	435.0082u	435.0082u
850.00000m	165.4791u	165.4791u
900.00000m	90.9049u	90.9049u

Although the cell of figure 9.3 exhibits a reasonable SNM of approximately 0.3, its low low beta-ratio (0.71) causes the cell to be prone to write failures. If width of the pull-up transistors of the inverter were smaller than the pull-down ones, and the pull-down ones were similar to the size of the pass-transistor and a better write capability would be achieved. Write failures can be analyzed by transient analysis instead of DC.

Process-induced variability such as RDF and LER, and temporal variability due to RTS and NBTI causes the butterfly curve of SRAM cell to shift towards a smaller square, meaning a degradation of SNM and RNM (BHAVNAGARWALA; TANG; MEINDL, 2001). But computing NM is not as straightforward as computing transient parameters of a logic gate, as for instance rise/fall delays. In order to investigate the impact of variability to the cell noise margin, the first problem is defining an automated methodology for the computation of NM. This methodology must be fully automated because, in order to allow variability analysis, it must be inserted inside a MC, thus repeated many times.

Agarwal (2006a) proposes an accurate and efficient computer methodology for computing SNM and RNM. This method is simple to implement and very automated, thus it is the most appropriate solution for computing NM in a Monte Carlo loop. The method is numerical, in the sense that it employs electrical simulations to evaluate the circuit and then uses a post-processing to compute the NM. The first step is to simulate the Butterfly curve of the SRAM cell in an electrical simulator and then store the curve in a database, as Table 9.1. Notice that in this case f(VL) and g(VR) are the same because this is a nominal simulation, in a Monte Carlo simulation they would differ. The first row can be interpreted as VR (when computing VL) or VL (when computing VR). Functions f and g employed from now on are approximated from these numerical simulations stored in the table.

Then, the loop gain of each side of the cell can be computed. The loop gain of VL (the equation is similar for VR) is given by (AGARWAL; NASSIF, 2006a):

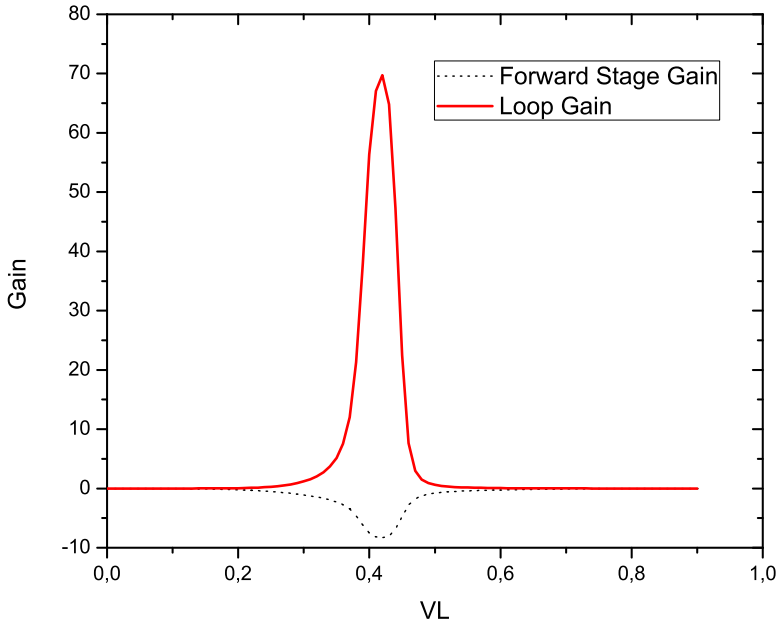


Figure 9.4: Loop gain of a 6T SRAM

$$LoopGain(VL) = \frac{\partial f}{\partial VL}\frac{\partial g}{\partial VR}\bigg|_{VR=f(VL)} \tag{9.2}$$

where the partial derivatives are computed numerically from the values stored in the database (output of the electrical simulation), simply as $\frac{\partial f_i}{\partial VL} = \frac{VR_{i+1}-VR_i}{VL_{i+1}-VL_i}$. Then, the minimum DC value in the input of inverter PR-NR that flips the content of the cell is defined as

$$VL_{flip} = (\text{VL that causes } LoopGain(VL) = 1) \tag{9.3}$$

Figure 9.4 shows the analysis of the loop gain of a SRAM cell in hold (stand-by) mode of operation. The sizing is $W_{NL} = W_{NL} = 100nm$, $W_{PL} = W_{PR} = 100nm$, $W_{AL} = W_{AR} = 140nm$. In the example, $VL_{flip} \approx 0.3$. The noise margin of side L is then defined as (AGARWAL; NASSIF, 2006a):

$$NML = VL_{flip} - g(f(VL_{flip})) \tag{9.4}$$

The noise margin of side R, NMR, is computed similarly, just substituting R and L in the previous expressions. Then, the noise margin of the cell is given by the minimum noise margin between the two sides as in (AGARWAL; NASSIF, 2006a):

$$NM = Min(NML, NMR) \tag{9.5}$$

The methodology for computing SNM and RNM was implemented as a PERL script interfacing with electrical simulations performed by HSPICE. These scripts are parameterized and are easily adaptable to model the impact of different sources of variation to the noise margin of SRAM cells. Due to its power, parameterization capabilities and easy

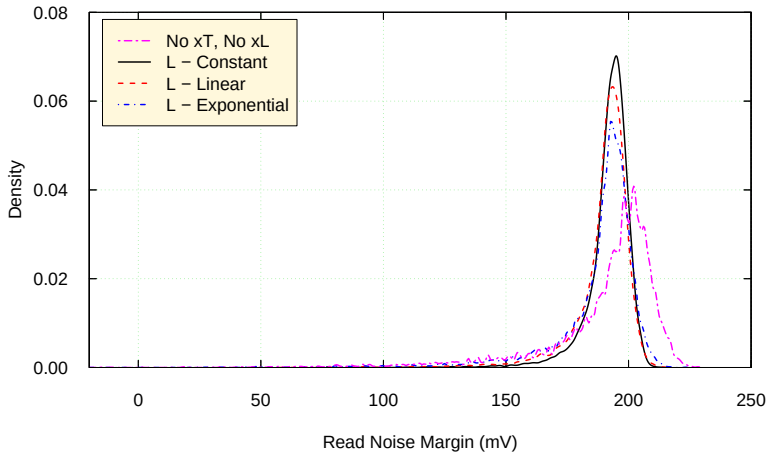


Figure 9.5: Effect of RTS on the read noise margin of the 65nm 6T-SRAM cell.

of use, these scripts have been largely employed in the group for modeling the impact of RTS and radiation to SRAM cells.

9.3 Statistical analysis of SRAM cell stability under RTS

The yield of a SRAM cache can be computed as a function of the number of cells, number of redundant cells and the probability of a SRAM cell to fail. Failures in SRAM cell can be due to: read failure, write failure or access time violation. This section presents the impact of RTS on the probability of read and write failures of a SRAM cell.

The instantaneous current fluctuations (modeled as threshold voltage shifts) caused by RTS are responsible for performance variability, which may cause read and write failures in SRAM cells. The three different models for the dependence of the amplitude of threshold voltage shifts on trap position along the channel are investigated.

For these simulations the transistor sizing is aligned to the transistor sizings of a conventional SRAM cell disclosed by Ohbayashi (2006): $Wp = 130nm, Wn = 90nm, Wa = 90nm$. Moreover, the transistor length $L = 65nm$ and $Leff = 24.5nm$ according to the minimum transistor length allowed by the technology node we use (CAO; MCANDREW, 2007). The number of traps of the transistors are computed following a Poisson law where λ_{Ntr} is in accordance to the transistor area.

9.3.1 Read failures

A read failure can happen when reading the value stored in the cell. At this time *BL* or *BR* (depending on the value stored) discharges through NL-AL or NR-AR , and this causes Node L or Node R to be charged to voltage V_{READ}. If V_{READ} becomes greater than the trip point V_{TRIP} of inverter NL-PR or NR-PR, read operation will cause the stored bit to erroneously flip.

Figure 9.5 shows the probability density (kernel density using bandwidth=1) of 10,000 MC simulations of Read Failures Probability (P_R) caused by RTS. P_R is modeled according to Agarwal (2006b), which is an appropriate approach for modeling P_R under process variations and RTS: P_R is computed by considering the DC noise margin of the cell during a read operation. From integrating the probability density functions, the probability of failures can be computed for the 4 different approaches of computing Vt shifts. Considering no xL and no xT dependence $P_R = 0.029\%$, considering constant dependence

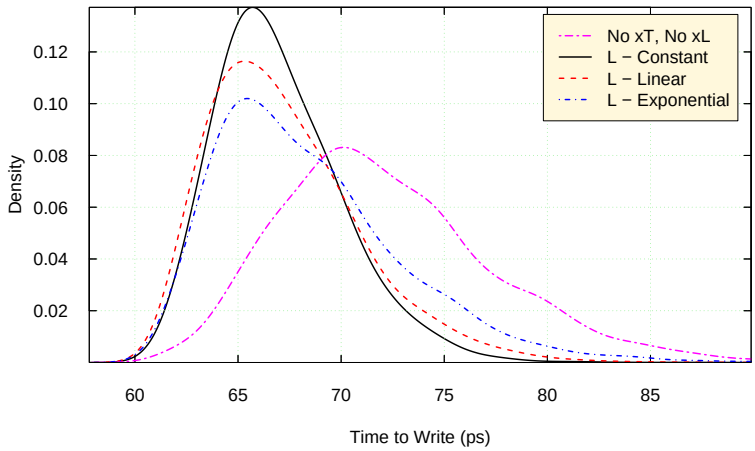


Figure 9.6: Effect of RTS on the write time of the 65nm 6T-SRAM cell.

$P_R = 4 \times 10^{-14}\%$, linear dependence $P_R = 4 \times 10^{-14}\%$ and using exponential dependence $P_R = 0.01\%$.

Please notice that the cache failure probability may be quite high even tough probability of a single bit to fail (represented here by P_R) seems very small. This is because cache memories nowadays usually contain millions of bits.

The different P_R values obtained for the different approaches of computing Vt shifts clearly shows the importance of detailed statistical analysis of RTS amplitude dependence on trap position in the oxide and along the channel. The values here used are based on Alexander (2005), where no statistical analysis was performed, i.e., it corresponds to the case here called "no xL and no xT dependence". The statistical analysis of dependence on trap position in the oxide and along the channel was introduced through equations 4.8 and 4.7, respectively.

9.3.2 Write failures

To write a value to the cell, *BL* and *BR* are set to the proper values, and then *wl* is set to VDD for a time T_{WRITE} in order to the signals to be stored in the cell. The period T_{WRITE} is related to the memory clock, in such a way that the writing operation must be performed in less time than T_{WRITE} to be successful. One cell requires the time T_W for a value to be properly stored. This T_W, due to process variations, is different from one cell to another, and can be described as a random variable. Therefore for a given cell to be able to correctly write values, $T_W < T_{WRITE}$ must hold, otherwise the value will not be successfully stored. The situation in which $T_W > T_{WRITE}$ is referred to as write failure.

The probability of the cell to fail because of a write failure is described as P_W. P_W is modeled according to Agarwal (2006b), as a transient simulation, and the probability of a write failure to occur is given by $P_W = P(T_{WRITE} > T_W)$, where T_{WRITE} is the time in which the signal WL stays high.

Figure 9.6 shows the density plots (using bandwidth=1) of 10,000 MC simulations of the time to write (T_W) variations in 6-T SRAM cell caused by RTS. The mean and standard deviates of the time to write are as follows. In the case where no length and tox dependence are considered, $\overline{T_W} = 72.2ps$ and $\sigma_{T_W} = 5.3ps$; constant length dependence has $\overline{T_W} = 67ps$ and $\sigma_{T_W} = 3ps$; linear length dependence presents $\overline{T_W} = 67.2ps$ and $\sigma_{T_W} = 3.6ps$; exponential length dependence has $\overline{T_W} = 68.4ps$ and $\sigma_{T_W} = 4.5ps$. Indeed, the probability of a write failure is dependent of the distribution of T_W and also dependent of

the time constraint T_{WRITE}.

Thus, the read and write failures probabilities (P_R and P_W) clearly depend on the shape of the function modeling Vt shifts as a function of trap position. This is an important finding because points to the fact that the charge density above channel must be taken into account when analyzing the impact of interface traps to circuit variation.

10 CONCLUSIONS

This manuscript presents a comprehensive study on statistical analysis of integrated circuits. Compact models for circuit simulation of process variability and aging are proposed, and three methodologies are employed for analyzing the impact of these variations to three different classes of circuits (standard cell, memory and clocking circuitry).

The manuscript describes some well-known models of process variability issues such as Random Dopant Fluctuations (RDF) and Line Edge Roughness (LER). These issues, as well as other process-related variability issues, are properly taken into account in the cell characterization and the clock analysis.

For the statistical cell characterization, partner foundry of IMEC provided variability measurements data of a 32nm technology node. These measurements were post-processed to the compact variability model as shifts in Vt and β.

For the clock network analysis, the statistical data was extracted from the process corners defined in the documentation provided by Fujitsu's 90nm technology node. By using a statistical simulation instead of corner-based approach, the simulation gives less pessimistic and more accurate results. Based on the assumptions that both the clock skew and the race immunity of flip-flops can be modelled by a Normal distribution, we proposed a simple and efficient method for computing the probability of hold time violation.

Initially, we proposed a static methodology for simulation of Random Telegraph Signal (RTS). The model is so called static because in this approach the Vt is constant during the transient simulation, in contrast to the dynamic trap-detrap simulation. The static RTS model gives as output a distribution of Vt, which is suitable for representing RTS as yet another source of Vt variation, on top of other issues such as RDF and LER. This approach allows the methodology to be supported by a running statistical flow with minor modifications. Later, as discussed below, a dynamic model was proposed.

As a case study for the static RTS model, an automated framework for the statistical analysis of SRAM, based on a state-of-the-art SRAM analysis methodology, was implemented. Thus we could study the impact of RTS on the Static Noise Margin (SNM), Read Noise Margin (RNM) and write time of the SRAM cell. Our experiments conclude that RTS alone, causing maximum Vt fluctuations around 60mV, can cause variation of nearly 100mV in the read noise margin of the SRAM cell. It is important to notice that RTS adds up to other variability phenomena and, if not taken properly into account, read failures may occur.

Random Telegraph Signal and its relationship to NBTI have emerged recently as a topic of great interest. This is because the classical NBTI reaction-diffusion model cannot explain certain behaviors found on experimental measurements. A cooperation between our group at UFRGS and the Reliability Group at IMEC aims to investigate this relationship between RTS and NBTI. In this work we have developed a simulation scheme

modeling both RTS and NBTI as a trap-detrap phenomena. The first results of the methodology are presented in this manuscript and the implementation will be important to our group in order to study the impact of RTS and NBTI on electrical circuits in the future.

An ongoing project, in cooperation with Texas Instruments, intends to employ the trap-detrap simulation to analyze jitter of oscillators. The methodology was implemented in NGSPICE, an open-source electrical simulator. We have some preliminary results of the impact of RTS on jitter of ring oscillators, although they are not present in this manuscript. Our group at UFRGS will continue studying trap-detrap phenome in the near future and the methodology present in this work is the key technology allowing the simulation of this phenomena.

The trap-detrap simulation also benefits from a research in cooperation with Arizona State University (ASU). The group of Prof. Dragica Vasileska at ASU performed atomistic simulations to study the impact of interface traps in a 45nm by 50nm transistor, taking into account the interaction between RDF and RTS. Their result is the most accurate simulation data exists nowadays of the impact of one interface trap to the Vt of the transistor. These state-of-the-art data contributed to the accuracy of our simulations.

This work has employed three simulation methodologies for propagating variability and reliability models to the circuit level. We vastly employ Monte Carlo method, which is the most common statistical simulation method. However error propagation and response surface methodology were developed in order to perform variability simulation with speedup of many orders of magnitude, while accuracy comparable to Monte Carlo is achieved.

We were among the pioneers to employ the classical error propagation formulation using linear sensitivity analysis for variability simulation of special purpose circuits, such as SRAM, in 2005. By performing n+1 simulation, being n the number of random variables, error propagation using sensitivity analysis gives an estimate of the standard deviation of the circuit response. It is the simplest and most efficient method to perform statistical analysis of circuit blocks. Error propagation is so successful that nowadays commercial EDA suites provide support for variability analysis through linear sensitivity analysis out of the box.

This work describes a novel methodology based on Response Surface Methodology for statistical characterization of circuit blocks. A new design of experiments, the Brussel design, is paired with a model selection algorithm, allowing accurate representation of the non-linear relationship between the input variability, such as Vt variation, to the circuit response, such as delay and power. The methodology, of which the PhD proponent is co-inventor, is protected under patents in the Europeans Union and United States of America, with title ”Response Characterization of an electronic system under variability effects”.

Error propagation using linear sensitivity analysis and RSM show average errors of less than 2% compared to MC for statistical characterization of a production level 32nm standard cell library. Unlike MC, the number of simulations required by RSM and sensitivity analysis is a function of the number of devices of the circuit. Being n the number of random variables, RSM requires 2n+1 electrical simulations and error propagation requires n+1. Thus the speedup of RSM and linear sensitivity over Monte Carlo is inversely proportional to the number of transistor of the circuit. Roughly, RSM and error propagation are recommended for circuits with up to one hundred (100) devices. For larger circuits Monte Carlo is the best all-around generic solution, although ad-hoc solutions usually present better accuracy-runtime tradeoff when available, e.g Statistical Static Timing Analysis.

Support for statistical analysis of integrated circuits has improved tremendously in the last decade, and our group has been in the right track proposing methodologies to speed up the time consuming Monte Carlo. Moreover, new issues negatively impacting the reliability of devices have been imposing new challenges for the design of integrated circuits. In this work we propose new models to deal with some of these issues. Reliability modeling and statistical analysis of digital circuits still requires manual intervention, ad-hoc methods and expertise from the designer, since it is far from a push-of-a-button process. We hope this work can contribute to the advance of the microelectronics industry and to the scientific community with small but important steps.

REFERENCES

AGARWAL, A. et al. Process variation in embedded memories: failure analysis and variation aware architecture. **IEEE Journal of Solid-State Circuits**, [S.l.], n.9, p.1804– 1814, September 2005.

AGARWAL, K.; NASSIF, S. Statistical analysis of SRAM cell stability. In: DESIGN AUTOMATION CONFERENCE, 2006 43RD ACM/IEEE, 2006. **Anais...** [S.l.: s.n.], 2006. p.57–62.

AGARWAL, K.; NASSIF, S. Statistical analysis of SRAM cell stability. **Design Automation Conference**, [S.l.], p.57–62, 0-0 2006.

AGOSTINELLI, M. et al. Erratic fluctuations of sram cache vmin at the 90nm process technology node. **Electron Devices Meeting, 2005. IEDM Technical Digest. IEEE International**, [S.l.], p.655–658, Dec. 2005.

AKAIKE, H. Information theory and an extension of the maximum likelihood principle. In: INT. SYMP. ON INFORMATION THEORY, 2., 1973. **Proceedings...** [S.l.: s.n.], 1973. p.267–281.

AKAIKE, H. A new look at the statistical model identification. **Automatic Control, IEEE Transactions on**, [S.l.], v.19, n.6, p.716–723, Dec. 1974.

ALEXANDER, C. et al. Impact of single charge trapping in nano-MOSFETs-electrostatics versus transport effects. **IEEE Trans on Nanotechnology**, [S.l.], v.4, n.3, p.339–344, May 2005.

ALVAREZ, A. et al. Application of statistical design and response surface methods to computer-aided VLSI device design. **Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on**, [S.l.], v.7, n.2, p.272–288, Feb. 1988.

AMAR, J. G. The Monte Carlo Method in Science and Engineering. **Computing in Science and Engineering**, Los Alamitos, CA, USA, v.8, n.2, p.9–19, 2006.

ASENOV, A. et al. Simulation of intrinsic parameter fluctuations in decananometer and nanometer-scale MOSFETs. **Electron Devices, IEEE Transactions on**, [S.l.], v.50, n.9, p.1837 – 1852, Sep. 2003.

ASENOV, A.; KAYA, S.; BROWN, A. R. Intrinsic parameter fluctuations in decananometer MOSFETs introduced by gate line edge roughness. **IEEE Transactions on Electron Devices**, [S.l.], v.50, n.5, p.1254–1260, 2003.

ASHRAF, N. et al. Accurate Model for the Threshold Voltage Fluctuation Estimation in 45-nm Channel Length MOSFET Devices in the Presence of Random Traps and Random Dopants. **Electron Device Letters, IEEE**, [S.l.], v.32, n.8, p.1044 –1046, aug. 2011.

ASSOCIATION, S. I. **International Technology Roadmap for Semiconductors**. [S.l.: s.n.], 2009. Available at <http://www.itrs.net>. Visited on Oct. 2009.

BASU, S.; VEMURI, R. Process Variation and NBTI Tolerant Standard Cells to Improve Parametric Yield and Lifetime of ICs. In: VLSI, 2007. ISVLSI '07. IEEE COMPUTER SOCIETY ANNUAL SYMPOSIUM ON, 2007. **Anais...** [S.l.: s.n.], 2007. p.291–298.

BERNSTEIN, K. et al. High-performance CMOS variability in the 65-nm regime and beyond. **IBM J. Res. Dev.**, Riverton, NJ, USA, v.50, p.433–449, July 2006.

BHASKER, J.; CHADHA, R. **Configuring the STA Environment**. [S.l.]: Springer US, 2009. 179-225p.

BHAVNAGARWALA, A.; TANG, X.; MEINDL, J. The impact of intrinsic device fluctuations on CMOS SRAM cell stability. **Solid-State Circuits, IEEE Journal of**, [S.l.], v.36, n.4, p.658 –665, Apr. 2001.

BOWMAN, K. A.; DUVALL, S. G.; MEINDL, J. D. Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration. **Solid-State Circuits, IEEE Journal of**, [S.l.], v.37, n.2, p.183–190, 2002.

BRUSAMARELLO, L. **Statistical Yield Analysis at Electrical Level Using Error Propagation and Numerical Derivatives**. 2006. Dissertação (Mestrado em Ciência da Computação) — Programa de Pós-Graduação em Ciência da Computação - Universidade Federal do Rio Grande do Sul, Porto Alegre, RS, Brasil.

BRUSAMARELLO, L. et al. Probabilistic Approach for Yield Analysis of Dynamic Logic Circuits. **Circuits and Systems I: Regular Papers, IEEE Transactions on**, [S.l.], v.55, n.8, p.2238–2248, Sept. 2008.

BRUSAMARELLO, L. et al. Statistical analysis of hold time violations. **Journal of Computational Electronics**, [S.l.], p.1–8, October 2010.

BRUSAMARELLO, L. et al. Fast and accurate statistical characterization of standard cell libraries. **Microelectronics Reliability**, [S.l.], v.In Press, Corrected Proof, p.–, 2011.

BRUSAMARELLO, L.; WIRTH, G. I.; SILVA, R. da. Statistical RTS model for digital circuits. **Microelectronics Reliability**, [S.l.], v.49, n.9-11, p.1064 – 1069, 2009.

CAMARGO, V. et al. Impact of RDF and RTS on the performance of SRAM cells. **Journal of Computational Electronics**, [S.l.], v.9, p.122–127, 2010. 10.1007/s10825-010-0340-9.

CAO, Y.; MCANDREW, C. MOSFET modeling for 45nm and beyond. In: PROC IEEE/ACM INTL CONFERENCE ON COMPUTER-AIDED DESIGN, 2007, Piscataway, NJ, USA. **Anais...** IEEE Press, 2007. p.638–643.

CATHIGNOL, A. et al. Quantitative Evaluation of Statistical Variability Sources in a 45-nm Technological Node LP N-MOSFET. **Electron Device Letters, IEEE**, [S.l.], v.29, n.6, p.609 –611, jun. 2008.

CHADWIN, D. Pulsed Id/Vg Methodology and Its Application to Electron-Trapping Characterization and Defect Density Profiling. **IEEE Trans Electron Dev**, [S.l.], 2009.

CHAMBERS, J. M. et al. **Graphical Methods for Data Analysis**. New York: Chapman and Hall, 1983.

CHEN, H. et al. A sliding window scheme for accurate clock mesh analysis. In: COMPUTER-AIDED DESIGN, 2005. ICCAD-2005. IEEE/ACM INTERNATIONAL CONFERENCE ON, 2005. **Anais...** [S.l.: s.n.], 2005. p.939–946.

CHINNERY, D.; KEUTZER, K. **Closing the Gap between ASIC and Custom**: tools and techniques for high-performance asic design. [S.l.]: Kluwer Academic Publishers (now Springer), 2002.

CROIX, J. F.; WONG, D. F. A fast and accurate technique to optimize characterization tables for logic synthesis. In: DAC '97: PROCEEDINGS OF THE 34TH ANNUAL DESIGN AUTOMATION CONFERENCE, 1997, New York, NY, USA. **Anais...** ACM, 1997. p.337–340.

da Silva, R.; Wirth, G. I. Logarithmic behavior of the degradation dynamics of metal-oxide-semiconductor devices. **Journal of Statistical Mechanics: Theory and Experiment**, [S.l.], v.4, p.25–+, Apr. 2010.

GAN, G.; MA, C.; WU, J. **Data Clustering**: theory, algorithms, and applications (asa-siam series on statistics and applied probability). illustrated edition.ed. [S.l.]: SIAM, Society for Industrial and Applied Mathematics, 2007.

GHIBAUDO, G.; BOUTCHACHA, T. Electrical noise and RTS fluctuations in advanced CMOS devices. **Microelectronics Reliability**, [S.l.], v.42, n.4-5, p.573 – 582, 2002.

GILDENBLAT, G. et al. Introduction to PSP MOSFET Model. In: TECHNICAL PROCEEDINGS OF THE 2005 WORKSHOP ON COMPACT MODELING, 2005. **Anais...** [S.l.: s.n.], 2005. p.19–24.

GRASSER, T. et al. Modelling of negative bias temperature instability under dynamic stress and recovery conditions. **Microelectronic Engineering**, [S.l.], n.7-9, p.1876–1882, 2009.

HANE, M.; IKEZAWA, T.; EZAKI, T. Coupled atomistic 3D process/device simulation considering both line-edge roughness and random-discrete-dopant effects. In: PROCEEDING OS SISPAD 2003. INTERNATIONAL CONFERENCE ON SIMULATION OF SEMICONDUCTOR PROCESSES AND DEVICES, 2003, 2003. **Anais...** [S.l.: s.n.], 2003. p.99– 102.

HANE, M.; IKEZAWA, T.; EZAKI, T. Coupled atomistic 3D process/device simulation considering both line-edge roughness and random-discrete-dopant effects. In: INTL CONF ON SIMULATION OF SEMICONDUCTOR PROCESSES AND DEV, 2003. **Anais...** [S.l.: s.n.], 2003. p.99–102.

HARASZTI, T. **CMOS memory circuits**. [S.l.]: Kluwer Academic, 2000.

HELOUE, K.; ONAISSI, S.; NAJM, F. Efficient block-based parameterized timing analysis covering all potentially critical paths. In: COMPUTER-AIDED DESIGN, 2008. ICCAD 2008. IEEE/ACM INTERNATIONAL CONFERENCE ON, 2008. **Anais...** [S.l.: s.n.], 2008. p.173–180.

HOCEVAR, D.; COX, P.; YANG, P. Parametric yield optimization for MOS circuit blocks. **Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on**, [S.l.], v.7, n.6, p.645–658, June 1988.

HU, T. H. M. W. M. Y. M. V. D. X. J. X. J. H. L. K. M. C. X. J. J. J. O. M. C. A. M. N. C. **BSIM 4.6.4 MOSFET Model Users' Manual**. [S.l.]: EECS Department, University of California, Berkeley, 2009.

HUARD, V.; DENAIS, M.; PARTHASARATHY, C. NBTI degradation: from physical mechanisms to modelling. **Microelectronics and Reliability**, [S.l.], v.46, n.1, p.1–23, 2006.

HUARD, V. et al. NBTI degradation: from transistor to sram arrays. In: RELIABILITY PHYSICS SYMPOSIUM, 2008. IRPS 2008. IEEE INTERNATIONAL, 2008. **Anais...** [S.l.: s.n.], 2008. p.289–300.

HYUN-WOO, K. et al. Experimental investigation of the impact of LWR on sub-100-nm device performance. **Electron Devices, IEEE Transactions on**, [S.l.], v.51, n.12, p.1984– 1988, Dec. 2004.

IMAI, M. et al. Non-parametric statistical static timing analysis: an ssta framework for arbitrary distribution. In: DAC '08: PROCEEDINGS OF THE 45TH ANNUAL DESIGN AUTOMATION CONFERENCE, 2008, New York, NY, USA. **Anais...** ACM, 2008. p.698–701.

KACZER, B. et al. Temperature dependence of the negative bias temperature instability in the framework of dispersive transport. **Applied Physics Letters**, [S.l.], v.86, n.14, p.143506, 2005.

KACZER, B. et al. NBTI from the perspective of defect states with widely distributed time scales. In: RELIABILITY PHYSICS SYMPOSIUM, 2009 IEEE INTERNATIONAL, 2009. **Anais...** [S.l.: s.n.], 2009. p.55–60.

KACZER, B. et al. Recent trends in bias temperature instability. **Journal of Vacuum Science and Technology B**, [S.l.], v.29, n.1, p.01AB01, 2011.

KANG, K. et al. Estimation of Statistical Variation in Temporal NBTI Degradation and its Impact on Lifetime Circuit performance. In: ICCAD, 2007. **Anais...** [S.l.: s.n.], 2007.

KELLER, I.; TAM, K. H.; KARIAT, V. Challenges in gate level modeling for delay and SI at 65nm and below. In: DAC '08: PROCEEDINGS OF THE 45TH ANNUAL DESIGN AUTOMATION CONFERENCE, 2008, New York, NY, USA. **Anais...** ACM, 2008. p.468–473.

KIM, J.; JONES, K. D.; HOROWITZ, M. A. Fast, non-Monte-Carlo estimation of transient performance variation due to device mismatch. In: DESIGN AUTOMATION CONFERENCE, 2007, New York, NY, USA. **Proceedings...** ACM, 2007. p.440–443.

KIRTON, M.; UREN, M. Noise in solid-state microstructures: a new perspective on individual defects, interface states and low-frequency (1/f) noise. **Advances in Physics**, [S.l.], v.38, p.367–468, 1989.

LAM, W.-C.; KOH, C.-K. Process variation robust clock tree routing. **Design Automation Conference, 2005. Proceedings of the ASP-DAC 2005. Asia and South Pacific**, [S.l.], v.1, p.606–611 Vol. 1, Jan. 2005.

LEYRIS, C. et al. N-MOSFET oxide trap characterization induced by nitridation process using RTS noise analysis. **Microelectronics Reliability**, [S.l.], v.47, n.1, p.41–45, 2007.

LI, X. Finding deterministic solution from underdetermined equation: large-scale performance modeling by least angle regression. In: DAC '09: PROCEEDINGS OF THE 46TH ANNUAL DESIGN AUTOMATION CONFERENCE, 2009, New York, NY, USA. **Anais...** ACM, 2009. p.364–369.

LI, Y.; YU, S.-M.; CHEN, H.-M. Process-variation- and random-dopants-induced threshold voltage fluctuations in nanoscale CMOS and SOI devices. **Microelectronic Engineering**, [S.l.], v.84, n.9-10, p.2117 – 2120, 2007. INFOS 2007.

MACHLUP, S. Noise in Semiconductors: spectrum of a two-parameter random signal. **Journal of Applied Physics**, [S.l.], v.25, p.341–343, Mar. 1954.

MAHAPATRA, S. et al. On the Physical Mechanism of NBTI in Silicon Oxynitride p-MOSFETs: can differences in insulator processing conditions resolve the interface trap generation versus hole trapping controversy? **Reliability physics symposium, 2007. proceedings. 45th annual. ieee international**, [S.l.], p.1–9, 15-19 April 2007.

MAHMOODI, H.; MUKHOPADHYAY, S.; ROY, K. Estimation of delay variations due to random-dopant fluctuations in nanoscale CMOS circuits. **Solid-State Circuits, IEEE Journal of**, [S.l.], v.40, n.9, p.1787–1796, 2005.

MARICAU, E.; GIELEN, G. Variability-Aware Reliability Simulation of Mixed-Signal ICs with Quasi-Linear Complexity. In: DESIGN, AUTOMATION AND TEST IN EUROPE 2010 (DATE), 2010. **Proceedings...** [S.l.: s.n.], 2010.

MASSOBRIO, G.; ANTOGNETTI, P. **Semiconductor Device Modeling with Spice**. [S.l.]: McGraw-Hill, 1999.

MCCONAGHY, T.; GIELEN, G. Globally Reliable Variation-Aware Sizing of Analog Integrated Circuits via Response Surfaces and Structural Homotopy. **Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on**, [S.l.], v.28, n.11, p.1627–1640, Nov. 2009.

MEHROTRA, V.; BONING, D. Technology scaling impact of variation on clock skew and interconnect delay. **Interconnect Technology Conference, 2001. Proceedings of the IEEE 2001 International**, [S.l.], p.122–124, 2001.

MENEZES, N.; KASHYAP, C.; AMIN, C. A "true" electrical cell model for timing, noise, and power grid verification. In: DAC '08: PROCEEDINGS OF THE 45TH ANNUAL DESIGN AUTOMATION CONFERENCE, 2008, New York, NY, USA. **Anais...** ACM, 2008. p.462–467.

MIRANDA, M.; ROUSSEL, P.; BRUSAMARELLO, L. **Response Characterization Of An Electronic System Under Variability Effects**. European Patent Application No. 10189434.3, Unpublished (filing date Oct. 29, 2010).

MIRANDA, M.; ROUSSEL, P.; BRUSAMARELLO, L. **Response Characterization Of An Electronic System Under Variability Effects**. US Patent Application No. 13/005,487, Unpublished (filing date Jan. 12, 2011).

MUKHOPADHYAY, S.; MAHMOODI, H.; ROY, K. Statistical design and optimization of SRAM cell for yield enhancement. In: ICCAD-2004. IEEE/ACM INTERNATIONAL CONFERENCE ON COMPUTER AIDED DESIGN, 2004., 2004. **Anais...** [S.l.: s.n.], 2004. p.10– 13.

MUKHOPADHYAY, S.; MAHMOODI-MEIMAND, H.; ROY, K. Modeling and estimation of failure probability due to parameter variations in nano-scale SRAMs for yield enhancement. In: SYMPOSIUM ON VLSI CIRCUITS DIGEST OF TECHNICAL PAPERS, 2004., 2004. **Anais...** [S.l.: s.n.], 2004. p.64–67.

MYERS, R. H.; MONTGOMERY, D. C. **Response Surface Methodology**: process and product in optimization using designed experiments. second.ed. New York, NY, USA: John Wiley & Sons, Inc., 2002.

NASSIF, S. Design for variability in DSM technologies [deep submicron technologies]. **Quality Electronic Design, 2000. ISQED 2000. Proceedings. IEEE 2000 First International Symposium on**, [S.l.], p.451–454, 2000.

NASSIF, S. et al. High Performance CMOS Variability in the 65nm Regime and Beyond. In: ELECTRON DEVICES MEETING, 2007. IEDM 2007. IEEE INTERNATIONAL, 2007. **Anais...** [S.l.: s.n.], 2007. p.569 –571.

NEUBERGER, G. **Protecting Digital Circuits against Hold Time Violations Due to Process Variations**. 2007. Tese (Doutorado em Ciência da Computação) — Programa de Pós-Graduação em Ciência da Computação - Universidade Federal do Rio Grande do Sul, Porto Alegre, RS, Brasil.

NEUBERGER, G. et al. Statistical analysis of systematic and random variability of flip-flop race immunity in 130nm and 90nm CMOS technologies. **Very Large Scale Integration, 2007. VLSI - SoC 2007. IFIP International Conference on**, [S.l.], p.78–83, Oct. 2007.

OHBAYASHI, S. et al. A 65 nm SoC Embedded 6T-SRAM Design for Manufacturing with Read and Write Cell Stabilizing Circuits. **Symp on VLSI Circuits**, [S.l.], p.17–18, 0-0 2006.

ORSHANSKY, M. et al. Impact of spatial intrachip gate length variability on the performance of high-speed digital circuits. **Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on**, [S.l.], v.21, n.5, p.544–553, 2002.

PARRAT, L. G. **Probability and Experimental Errors on Science**. New York, NY, USA: John Wiley and Sons Inc, 1961.

PRESS, W. H. et al. Moments of a Distribution: mean, variance, skewness, and so forth. In: **Numerical recipes in FORTRAN**: the art of scientific computing. New York, NY, USA: Cambridge University Press, 1992. p.604–609.

REID, D. et al. Prediction of random dopant induced threshold voltage fluctuations in NanoCMOS transistors. In: SIMULATION OF SEMICONDUCTOR PROCESSES AND DEVICES, 2008. SISPAD 2008. INTERNATIONAL CONFERENCE ON, 2008. **Anais...** [S.l.: s.n.], 2008. p.21 –24.

SAKS, N.; ANCONA, M. Spatial Uniformity of Interface Trap Distribution in MOSFETs. **IEEE Trans Electron Dev**, [S.l.], 1990.

SCHOLTEN, A. et al. The New CMC Standard Compact MOS Model PSP: advantages for rf applications. **Solid-State Circuits, IEEE Journal of**, [S.l.], v.44, n.5, p.1415 – 1424, may 2009.

SCHWARZ, G. Estimating the dimension of a model. **The annals of statistics**, [S.l.], v.6, n.2, p.461–464, 1978.

SHENOY, N. V.; BRAYTON, R. K.; SANGIOVANNI-VINCENTELLI, A. L. Minimum padding to satisfy short path constraints. In: ICCAD '93: PROCEEDINGS OF THE 1993 IEEE/ACM INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN, 1993, Los Alamitos, CA, USA. **Anais...** IEEE Computer Society Press, 1993. p.156–161.

SHI, S. X. et al. Latch Modeling for Statistical Timing Analysis. In: DATE, 2008. **Anais...** [S.l.: s.n.], 2008. p.1136–1141.

SILVA, R. da; BRUSAMARELLO, L.; WIRTH, G. I. Statistical fluctuations for the noise current from random telegraph signals in semiconductor devices: monte carlo computer simulations and best fits. **Physica A: Statistical Mechanics and its Applications**, [S.l.], v.389, n.14, p.2687 – 2699, 2010.

SILVA, R. da et al. A simple combinatorial method to describe particle retention time in random media with applications in chromatography. **Physica A: Statistical Mechanics and its Applications**, [S.l.], v.In Press, Accepted Manuscript, p.–, 2011.

SILVA, R. da; LAMB, L. C.; WIRTH, G. I. Collective Poisson process with periodic rates: applications in physics from micro-to nanodevices. **Royal Society of London Philosophical Transactions Series A**, [S.l.], v.369, p.307–321, Jan. 2011.

SILVA, R. da; WIRTH, G.; BREDERLOW, R. Novel analytical and numerical approach to modeling low-frequency noise in semiconductor devices. **Physica A: Statistical Mechanics and its Applications**, [S.l.], v.362, n.2, p.277 – 288, 2006.

SIMOEN, E. et al. Explaining the amplitude of RTS noise in submicrometer MOSFETs. **IEEE Trans on Electron Dev**, [S.l.], v.39, n.2, p.422–429, Feb 1992.

SONODA, K. et al. Discrete Dopant Effects on Statistical Variation of Random Telegraph Signal Magnitude. **Electron Devices, IEEE Transactions on**, [S.l.], v.54, n.8, p.1918–1925, Aug. 2007.

STOLK, P.; WIDDERSHOVEN, F.; KLAASSEN, D. Modeling statistical dopant fluctuations in MOS transistors. **Electron Devices, IEEE Transactions on**, [S.l.], v.45, n.9, p.1960 –1971, Sep. 1998.

SYNOPSYS. **HSPICE Simulation and Analysis User Guide**. [S.l.: s.n.], 2005.

SZE, S.; NG, K. K. **Physics of Semiconductor Devices**. [S.l.]: John Wiley & Sons, Inc., 2006.

TEGA, N. et al. Anomalously Large Threshold Voltage Fluctuation by Complex Random Telegraph Signal in Floating Gate Flash Memory. **Electron Devices Meeting, 2006. IEDM '06. International**, [S.l.], p.1–4, Dec. 2006.

TRIHY, R. Addressing library creation challenges from recent Liberty extensions. In: DAC '08: PROCEEDINGS OF THE 45TH ANNUAL DESIGN AUTOMATION CONFERENCE, 2008, New York, NY, USA. **Anais...** ACM, 2008. p.474–479.

TSIVIDIS, Y. **Operation and Modeling of the Mos Transistor (The Oxford Series in Electrical and Computer Engineering)**. [S.l.]: Oxford University Press, 2004.

VAGLIO-PRET, A.; GRONHEID, R.; FOUBERT, P. Roughness characterization in the frequency domain and LWR mitigation with post-litho processes. **Advances in Resist Materials and Processing Technology XXVII**, [S.l.], v.7639, n.1, p.763930, 2010.

VASILESKA, D.; KHAN, H.; AHMED, S. Quantum and Coulomb Effects in Nanodevices. **Intl Journal of Nanoscience**, [S.l.], v.4, n.3, p.305–361, 2005.

VATTIKONDA, R.; WANG, W.; CAO, Y. Modeling and minimization of PMOS NBTI effect for robust nanometer design. In: DAC '06: PROCEEDINGS OF THE 43RD ANNUAL CONFERENCE ON DESIGN AUTOMATION, 2006, New York, NY, USA. **Anais...** ACM, 2006. p.1047–1052.

VENABLES, W. N.; RIPLEY, B. D. **Modern Applied Statistics with S**. 4.ed. New York: Springer, 2002.

VISWESWARIAH, C. Death, taxes and failing chips. In: DESIGN AND AUTOMATION CONFERENCE, DAC, 40., 2003, Anaheim, CA, USA. **Proceedings...** New York: ACM Press, 2003. p.343–347.

VISWESWARIAH, C. et al. First-order incremental block-based statistical timing analysis. In: DAC '04: PROCEEDINGS OF THE 41ST ANNUAL DESIGN AUTOMATION CONFERENCE, 2004, New York, NY, USA. **Anais...** ACM, 2004. p.331–336.

WANG, J.; GHANTA, P.; VRUDHULA, S. Stochastic analysis of interconnect performance in the presence of process variations. In: COMPUTER AIDED DESIGN, 2004. ICCAD-2004. IEEE/ACM INTERNATIONAL CONFERENCE ON, 2004. **Anais...** [S.l.: s.n.], 2004. p.880–886.

WESTE, N.; HARRIS, D. **CMOS VLSI design**: a circuits and systems perspective. [S.l.]: Pearson/Addison-Wesley, 2005.

WINITZKI, S. **Computational Science and Its Applications**. [S.l.]: Springer–Verlag, 2003. p.962.

WIRTH, G. et al. Modeling of statistical low-frequency noise of deep-submicrometer MOSFETs. **IEEE Trans on Electron Dev**, [S.l.], v.52, n.7, p.1576–1588, July 2005.

WIRTH, G. et al. Statistical model for MOSFET low-frequency noise under cyclo-stationary conditions. In: ELECTRON DEVICES MEETING (IEDM), 2009 IEEE INTERNATIONAL, 2009. **Anais...** [S.l.: s.n.], 2009. p.1 –4.

WIRTH, G. I. **Reliability and Yield of MOS devices and Circuits**. 2010.

Wirth, G. I.; da Silva, R.; Brederlow, R. Statistical Model for the Circuit Bandwidth Dependence of Low-Frequency Noise in Deep-Submicrometer MOSFETs. **IEEE Transactions on Electron Devices**, [S.l.], v.54, p.340–345, Feb. 2007.

WIRTH, G. I. et al. Modelling of Statistical Low-Frequency Noise of Deep-Submicron MOSFETs. **IEEE Transactions on Electron Devices**, [S.l.], p.1576–1588, 2004.

WIRTH, G. I.; SILVA, R. da; KACZER, B. Statistical Model for MOSFET Bias Temperature Instability Component Due to Charge Trapping. **Electron Devices, IEEE Transactions on**, [S.l.], v.PP, n.99, p.1 –9, 2011.

WIRTH, G.; SILVA, R. da; BREDERLOW, R. Statistical Model for the Circuit Bandwidth Dependence of Low-Frequency Noise in Deep-Submicrometer MOSFETs. **IEEE Trans on Electron Dev**, [S.l.], v.54, n.2, p.340–345, Feb. 2007.

YE, Y. et al. Statistical modeling and simulation of threshold variation under dopant fluctuations and line-edge roughness. In: DESIGN AUTOMATION CONFERENCE, 2008. DAC 2008. 45TH ACM/IEEE, 2008. **Anais...** [S.l.: s.n.], 2008. p.900 –905.

ZARKESH-HA, P.; MULE, T.; MEINDL, J. Characterization and modeling of clock skew with process variations. **Custom Integrated Circuits, 1999. Proceedings of the IEEE 1999**, [S.l.], p.441–444, 1999.

ZUBER, P. et al. Exponent Monte Carlo for Quick Statistical Circuit Simulation. In: PATMOS, 2010. **Anais...** Springer, 2010. p.36–45. (Lecture Notes in Computer Science, v.5953).

ZUCHOWSKI, P. S. et al. Process and environmental variation impacts on ASIC timing. In: COMPUTER AIDED DESIGN, 2004. ICCAD-2004. IEEE/ACM INTERNATIONAL CONFERENCE ON, 2004. **Anais...** [S.l.: s.n.], 2004. p.336–342.

APÊNDICE A MODELAGEM DE CONFIABILIDADE E VARIABILIDADE DE TRANSISTORES EM NÍVEL ELÉTRICO

Tradicionalmente, projetistas de circuitos integrados (ICs) digitais contavam com níveis de abstração onde variabilidade no processo de fabricação era intrinsecamente levada em consideração, contudo ficava escondida do projetista a fim de tornar o fluxo de projeto mais simples. Uma vez que o projetista desenhasse o circuito em conformidade com as regras de projeto, os casos extremos de comportamento do circuito poderiam ser simulados com os modelos fornecidos. O designer poderia esperar que o chip funcionasse dentro das especificações definidas pela *foundry*. Na verdade, o projetista esperava que uma *alta porcentagem* de chips atendesse aos requisitos, enquanto o conceito de *rendimento* era implícito ao projetista e era de responsabilidade da foundry.

Entretanto, considerando-se que as dimensões dos transistores atuais está na escala de dezenas de nano-metros, pequenos desvios das características do dispositivo em relação ao caso nominal podem levar a falhas no circuito. Em tecnologias nano-métricas, estes desvios podem acontecer não somente devido a defeitos durante a produção, mas também acontecem devido a impossibilidade do controle exato de características dos equipamentos (por exemplo a profundidade exata na etapa de implantação iônica), e cada vez mais a variabilidade intrínseca devido à discretude da matéria torna-se predominante. Esses três fatores (defeitos, variabilidade dos equipamentos e variabilidade intrínseca) fazem com que as características elétricas dos transistores devam ser tratadas como variáveis aleatórias. Essa mudança de paradigma, onde o comportamento elétrico do circuito não é determinístico mas sim estatístico, impõe novos desafios para o projeto de circuitos analógicos e digitais.

Conforme ilustra a figura 1.1, a variabilidade dos parâmetros elétricos dos transistores pode ser decomposta em duas componentes: espacial e temporal.

A variabilidade espacial pode ser ainda decomposta em parâmetros que apresentam variações entre pastilhas (D2D, do inglês *die-to-die*) e parâmetros que apresentam variabilidade dentro da pastilha (WD, do inglês *within-die*)(ZUCHOWSKI et al., 2004) (ORSHANSKY et al., 2002). Variabilidade D2D pode acontecer devido a assimetria nos equipamentos (como assimetria na distribuição do gás dentro de uma câmara e gradientes de temperatura em um forno) ou imperfeições na operação de equipamentos e no fluxo de processo. Essas assimetrias afetam a média de um parâmetro entre pastilhas, *wafer* ou lote.

Variabilidade nos parâmetros WD pode ainda ser decoomposta em duas componentes: variabilidade sistemática e variabilidade aleatória (ou intrínseca). Variações WD aleatórias são originárias de inúmeras fontes relacionadas às características quânticas dos materiais, tais como a discretude da matéria e energia (átomos de dopante, fótons, etc).

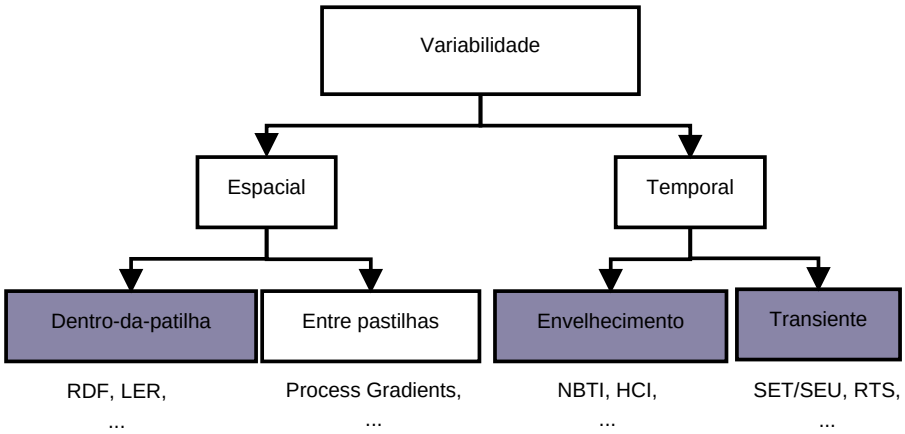


Figura A.1: Classificação de variabilidade em temporal e espacial segundo Wirth (2010).

Em tecnologias atuais a variabilidade intrínseca aleatória já domina as outras fontes de variabilidade e devido a tendência de miniaturização esta deve se tornar cada vez mais importante.

A componente temporal pode ser dividida em envelhecimento e variações transitórias. As principais razões para a variabilidade temporal são: 1) captura e emissão de elétrons por armadilhas no silício e na interface entre silício e óxido de silício dos dispositivos e 2) partículas de radiação atingindo o dispositivo. O envelhecimento é a degradação sistemática das características do transistor, como por exemplo a corrente máxima de um transistor PMOS ficando mais fraca ao longo do tempo devido a instabilidade acelerada por temperatura e tensão (NBTI). Variabilidade transitória são mudanças instantâneas ou intermitentes na corrente do dispositivo, que pode ser causada por radiação ou sinais aleatórios telegráficos (RTS).

A demanda da indústria para projetar circuitos em tecnologias nano-métricas exige pesquisa em duas áreas relacionadas a integração da tecnologia às ferramentas de CAD: 1) modelagem de confiabilidade de transistores e 2) metodologias para análise de circuitos integrados considerando confiabilidade. A seção de ”Modelagem e Simulação” do Roteiro Internacional dos Semicondutores 2009 (ASSOCIATION, 2009) aponta para a necessidade de modelos para fenômenos de captura e emissão de elétrons por armadilhas como RTS e NBTI. O ITRS também aposta a necessidade necessidade de metodologias de simulação de confiabilidade de circuitos CMOS.

Neste trabalho de doutorado foram estudados e propostos avanços nestas duas áreas inter-relacionadas de pesquisa: modelagem de confiabilidade e metodologias de simulação de variabilidade. Esta tese apresenta novos modelos estatísticos de RTS e NBTI. Estes modelos estatísticos são aplicados a circuitos para estudo de caso. Além disso, essa tese propõe metodologias eficientes de simulação estatística. Propagação de incertezas utilizando derivadas numéricas e metodologia de superfície de resposta são implementadas e suas eficiências são avaliadas em relação a simulações Monte Carlo.

A.1 Variabilidade temporal causada por emissão e captura por armadilhas de interface

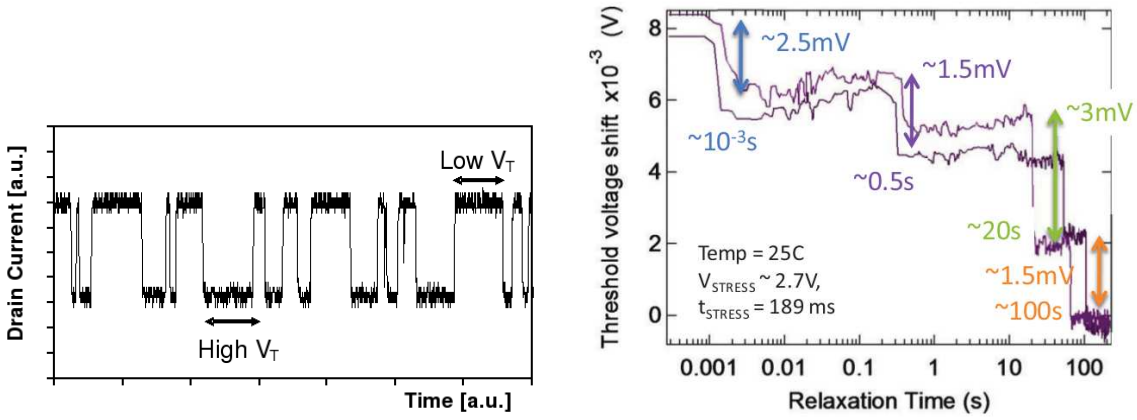
No domínio tempo, capturas e emissões de elétrons por armadilhas (cargas positivas) no silício e na interface entre silício e o óxido de silício do transistor causam flutuações na sua corrente ao longo do tempo, mesmo mantendo Vgs e Vds constantes. Estas flutuações são discretas: quando a armadilha i captura um elétron, a corrente Ids diminui em ΔIds_i. O estado de todas as armadilhas na interface somam-se para formar a flutuação total da corrente em um dado instante de tempo.

A figura A.2(a) mostra a currente do transistor variando devido aos sinais aleatórios telegáficos (RTS, do inglês Random Telegraph Signals). RTS faz com que a corrente do transistor, e portanto os parâmetros elétricos relacionados a corrente como por exemplo tensão de limiar, oscilem em níveis discretos intermitentemente ao longo do tempo. Instabilidade acelerada por temperatura e tensão (NBTI, do inglês Negative Bias Temperature Instability) contudo, trata-se do aumento sistemático do Vt, ou seja, diminuição da corrente do transistor, ao longo do tempo. Essa degradação é acelerada pela temperatura e, especialmente, pela tensão aplicada no gate do transistor. O mecanismo de NBTI é dito ter duas fases: stress, quando tensão é aplicada no gate, e recuperação, com tensão nula. Na fase de stress percebe-se que o Vt do transistor aumenta, enquanto na fase de recuperação o Vt diminui parcialmente. A figura A.2(b) mostra que NBTI apresenta uma componente semelhante a RTS, segundo Kaczer (2011). Até então, o modelo mais aceito para NBTI tem sido o modelo de reação-difusão, o qual explica NBTI como sendo causado pela quebra das ligações entre hidrogênio e silício na interface entre o silício e óxido de silício. O modelo de reação-difusão, apesar de amplamente adotado, tem problemas ao explicar a rápida recuperação que acontece assim que o stress é removido, como nas medidas da figura A.2(b).

Wirth (2011) apresenta análises teóricas e simulações Monte Carlo do componente de captura e emissão responsável por BTI. O trabalho apresenta um modelo analítico válido para as fases de stress e de recuperação. A teoria assume que pode ou não existir um mecanismo de geração de novas armadilhas na interface ao londo do tempo. Pode haver geração de novas cargas positivas devido a quebra das ligações de hidrogênio, conforme assumia a teoria clássica de NBTI, modelo de reação-difusão. Contudo, existe possibilidade de que muitos traps que causam NBTI sejam traps com tempo médio de captura e emissão muito longos. Assim, uma parte das armadilhas contribuindo para NBTI são armadilhas com comportamento semelhante a armadilhas RTS, mas com uma diferença importante:

- as armadilhas causando NBTI têm diferenças de várias ordens de magnitude com relação a suas probabilidades de captura e emissão;
- enquanto as armadilhas que contribuem para o ruído RTS têm probabilidades de captura e emissão de mesma ordem de magnitude.

Essa seção mostra a metodologia de simulação proposta em Wirth (2011), a qual é válida para simular RTS e NBTI. Cada transistor contém uma série de armadilhas, que em um dado instante de tempo podem estar ocupadas ou vazias. Dependendo do seu estado atual, cada armadilha tem uma probabilidade de capturar ou emitir um elétron no



(a) (b)

Figura A.2: (a) Representação do impacto de sinais aleatórios telegáficos na corrente do transistor e como pode ser modelado com flutuações em Vt e (b) medidas de instabilidade acelerada por tensão e temperatura em um dispositivo de 70 $\times$ 90 nm^2 realizadas por Kaczer (2011).

estado seguinte (*pe* e *pc*) dadas por:

$$\begin{aligned} pc = \Pr(\sigma_i(t) = 0 \rightarrow 1) &= \frac{\Delta t}{\tau_c} \\ pe = \Pr(\sigma_i(t) = 1 \rightarrow 0) &= \frac{\Delta t}{\tau_e} \end{aligned} \tag{A.1}$$

onde Δt é o intervalo de tempo, τ_e e τ_c são os tempos médios de emissão e captura, que por sua vez são calculados como:

$$\begin{aligned} \tau_e &= 10^{-pi}(1 + exp(\frac{E_f - E_i}{K_b T}) \\ \tau_c &= 10^{pi}(1 + exp(\frac{E_i - E_f}{K_b T}) \end{aligned} \tag{A.2}$$

onde para a carga positiva i, K_b é a constante de Boltzman , T é a temperatura do dispositivo em Kelvin, E_f é o nível de Fermi do transistor, e_i é o nível de energia da armadilha i e p_i é a constante de tempo da armadilha. As armadilhas têm níveis de energia dentro do bandgap proibido e a distribuição de sua energia segue uma distribuição em foma de U (WIRTH et al., 2009; WIRTH; SILVA; KACZER, 2011). O nível de Fermi depende da tensão nos terminais de transistor e é precisamente calculado em nosso modelo através de ajuste de função.

A figura A.3 mostra 100 simulações Monte Carlo referentes a execuções do modelo dinâmico (Cadeias de Markov) de RTS ao longo de 2us. Cada caixa corresponde a uma rodada da Cadeia de Markov (durante 2us) inicializada com uma semente aleatória diferente. É importante ressaltar a necessidade de se rodar uma simulação Monte Carlo de Cadeias de Markov: uma rodada apenas não seria representativa do comportamento do transistor, pois cada rodada trata-se de um transistor com número de armadilhas, constantes de tempo e δvti distintos. Assim, a figura mostra que, para os parâmetros utilizados, RTS pode causar variações de mais de 250mV.

A simulação se refere a um modelo de tecnologia de 45nm (PTM) com o dimensionamento L = 45nm e W= 50nm , e tensão entre bulk e source Vbs = 0. A média é de

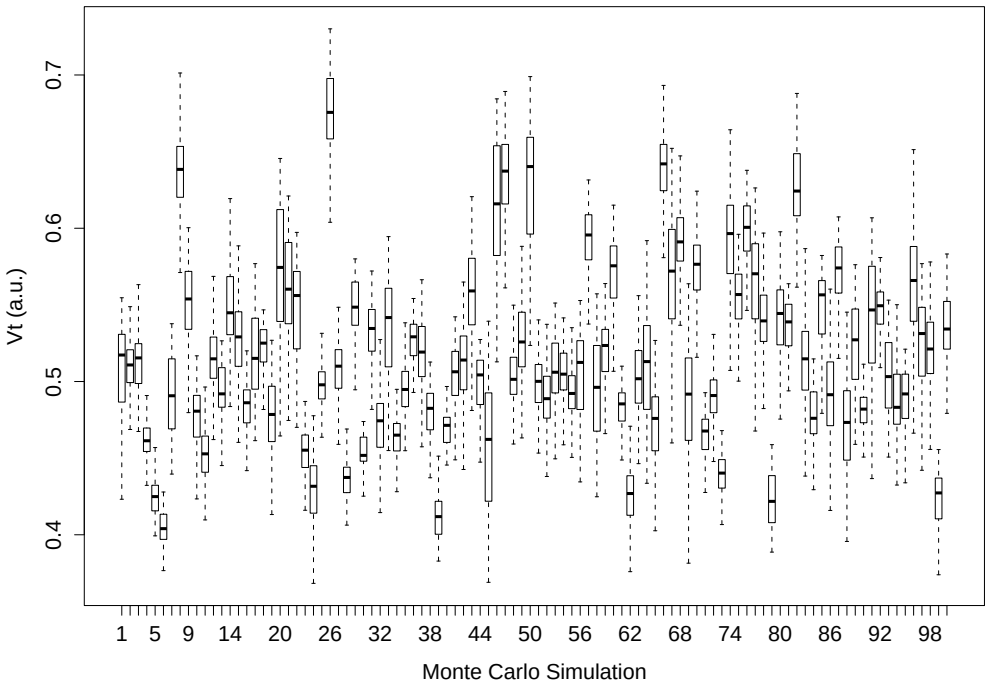


Figura A.3: Distribuições do Vth do transistor ao longo de 100 simulações Monte Carlo das Cadeias de Markov.

$overline{V} = 515mV$, e seu desvio padrão é de $\sigma_{Vt} = 59mV$. Essa simulação refere-se RTS e não a NBTI porque 1) não há mecanismo de geração de traps ao longo do tempo e 2) as constantes de tempo das armadilhas *pi* estão no intervalo [-5, -8]. Para simular NBTI, valores menores de pi deveriam ser permitidos.

Nessa simulação o número médio de armadilhas interface é 80 ($lambda_{Ntr} = 80$), o qual está de acordo com dados de Wirth (2005), linearmente re-escalados para as dimensões do dispositivo aqui simulado. A distribuição das flutuações causadas por uma única armadilha δvt_i foram obtidas através de simulações atomísticas descritas por Camargo (2010).

A.2 Modelos de Simulação

Dada a natureza estatística do processo de fabricação, características elétricas como Vt (tensão de limiar) e $\Delta\beta/\beta$ (variação na corrente) dos transistores podem ser modeladas como variáveis aleatórias. Este trabalho visa modelar o impacto das flutuações estatísticas de Vt e $\Delta\beta/\beta$ em circuitos elétricos. Para isso, foram utilizadas três metodologias para propagar as incertezas das entradas e avaliar seu impacto no circuito elétrico: Monte Carlo, propagação de erros (EP) e metodologia de superfície de resposta (RSM).

Monte Carlo é a metodologia mais comumente empregada para simulação de variabilidade em circuitos elétricos. Simulação Monte Carlo (SYNOPSYS, 2005) é comumente empregada para calcular a função de densidade de probabilidades (PDF) de alguma resposta do circuito (atraso, potência, corrente de fuga, ...). Mas para isso é necessário um grande número de simulações elétricas, pois o erro em simulações Monte Carlo é $O(1/\sqrt{n_{sample}})$. É o método mais simples de implementar e mais preciso, contudo requer

maior tempo de simulação. Métodos alternativos a Monte Carlo são de grande interesse para a indústria, a fim de reduzir os tempos de simulação. Propagação de erros usando derivadas lineares está começando a ser empregado na indústria por ferramentas de CAD comerciais. A Metodologia de Superfície de Resposta (RSM) apresentada a seguir foi proposta e desenvolvida no âmbito deste doutorado.

A.2.1 Propagação de erros

As respostas do circuito (as saídas da simulação), tais como desempenho e potência, são variáveis aleateatórias e podem ser modeladas utilizando o método da propagação de incertezas (PARRAT, 1961). A fim de usar essa abordagem, duas suposições devem ser feitas:

1. as variáveis aleatórias de entrada seguem uma distribuição Normal;

2. a função de propagação pode ser aproximada por uma função linear na região de interesse.

A partir desses pressupostos então a saída da simulação (atraso por exemplo) pode ser aproximada como uma variável aleatória Normal, e seus momentos podem ser calculados analiticamente. A partir dos desvios padrões σ_{Vt_i} e σ_{β_i} de cada transistor i, calcula-se 1) a simulação nominal *overlines* e 2) as derivadas parciais de cada resposta para cada parâmetro de entrada, ou seja s_{Vti}, $s_{\beta i}$, que são calculadas numericamente como em:

$$\begin{aligned} s_{Vti} &= \frac{f(\overline{Vt_1},\ldots,\overline{Vt_i}+\varepsilon,\ldots,\overline{Vt_n},\overline{\beta_1},\ldots,\overline{\beta_n})-\overline{s}}{\varepsilon} \\ s_{\beta i} &= \frac{f(\overline{Vt_1},\ldots,\overline{Vt_n},\overline{\beta_1},\ldots,\overline{\beta_i}+\varepsilon,\ldots,\overline{\beta_n})-\overline{s}}{\varepsilon} \end{aligned} \tag{A.3}$$

onde $f(\overline{Vt_1},\ldots,\overline{Vt_n},\overline{\beta_1},\ldots,\overline{\beta_n})$ são as características da célula (tais como atrasos de subida e descida, tempos de transição, potência, etc), que são funções das variações em Vt e β dos n transistores. A partir da simulação nominal e das derivadas parciais, a função de propagação pode ser aproximada por uma função linear:

$$\widehat{y_i} = \overline{s} + \sum_{i=1}^{n} \left[s_{Vti}\Delta Vt_i + s_{\beta i}\Delta\beta_i \right] \tag{A.4}$$

Sendo assim a resposta do circuito p pode ser considerada uma distribuição normal com média e variância dadas por (BRUSAMARELLO, 2006; BRUSAMARELLO et al., 2008):

$$\begin{cases} \mu_p \approx \overline{s} \\ \sigma_p^2 \approx \sum_{i=1}^{n} \left[(s_{Vti}\sigma_{Vti})^2 + (s_{\beta i}\sigma_{\beta i})^2 \right] \end{cases} \tag{A.5}$$

A.2.2 Metodologia de Superfície de Resposta[1]

[1] A invenção descrita neste capítulo está protegida por patentes na União Européia (MIRANDA; ROUSSEL; BRUSAMARELLO, 2010) e Estados Unidos da América (MIRANDA; ROUSSEL; BRUSAMARELLO, 2011).

A fim de obter precisão semelhante a Monte Carlo com ganho de desempenho de ordens de magnitude, este capítulo apresenta o uso de Metodologia de Superfície de Resposta. A metodologia é dividida em duas etapas. O primeiro passo consiste em um novo Projeto de Experimentos (Brussel) que realiza a seleção dos pontos do espaço de entradas e garante a relevância estatística desses pontos. O projeto de experimentos Brussel é combinado com um algoritmo de seleção de modelo. Esse algoritmo encontra a função não-linear de regressão mais adequada para representar a resposta do circuito em função das variáveis aleatórias.

O primeiro passo da metodologia de superfície de resposta é realizar projeto de experimentos (MYERS; MONTGOMERY, 2002). O objetivo desta etapa é encontrar N_{doe} pontos que são representativos para o espaço n-dimensional de variáveis aleatórias. Nesta fase não há nenhum conhecimento prévio sobre a função de propagação a ser modelada. Os pontos precisam ser selecionados de tal maneira que cubram tanto quanto possível o domínio da distribuição da entrada. A seguir é apresentado o procedimento proposto por Philippe Roussel para a seleção dos pontos.

Seja um ensamble Monte Carlo Γ^M com uma amostra de tamanho N da função de n-dimensões representado pela matriz $N \times n$:

$$\Gamma = \begin{pmatrix} Vt_1^1 & \beta_1^1 & Vt_2^1 & \beta_2^1 & \dots & Vt_{n/2}^1 & \beta_{n/2}^1 \\ Vt_1^2 & \beta_1^2 & Vt_2^2 & \beta_2^2 & \dots & Vt_{n/2}^2 & \beta_{n/2}^2 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ Vt_1^N & \beta_1^N & Vt_2^N & \beta_2^N & \dots & Vt_{n/2}^N & \beta_{n/2}^N \end{pmatrix} \tag{A.6}$$

onde γ_{ij} corresponde a i-ésima rodada Monte Carlo e j corresponde à j-ésima variável aleatória. A seguir será explicado o procedimento de seleção dos pontos sem fazer a "cobertura" do domínio de entrada por algum método de suavização da PDF, como por exemplo o método das Múltiplas Gaussianas que é utilizado no método descrito por Brusamarello (2011), do domínio de entrada.

Inicialmente calcula-se um vetor das médias das colunas da matriz de entrada, o qual é dado por $\vec{\mu} = \{\mu_1, \mu_2, \dots, \mu_n\}$. Similarmente calcula-se uma matriz diagonal dos desvios padrões das variáveis (cada coluna da matriz Γ) de entrada:

$$\sigma = \begin{pmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_n \end{pmatrix} \tag{A.7}$$

Então pode-se facilmente calcular também a matriz de correlação como em:

$$\rho = \begin{pmatrix} 1 & \rho_{12} & \cdots & \rho_{1n} \\ \rho_{21} & 1 & \cdots & \rho_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{n1} & \rho_{n2} & \cdots & 1 \end{pmatrix} \tag{A.8}$$

onde ρ_{lm} é a correlação entre as variáveis l e m, por exemplo Vt de um dado transistor com Vt do outro transistor. A seguir calcula-se a matriz de autovetores:

$$E = \begin{pmatrix} e_1 & 0 & \cdots & 0 \\ 0 & e_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & e_n \end{pmatrix} \tag{A.9}$$

e o vetor de autovalores $\vec{e} = [e_1, e_2, \ldots, e_n]_n$. Pode-se então contruir uma estrutura de dados Ellipsoid:

$$Ellipsoid \begin{cases} \vec{c} = & [0]_n \\ \vec{r} = & q\chi^2(q,n)\sqrt{\vec{e}} \\ A = & E \end{cases} \tag{A.10}$$

o qual representa o elipsóide de n dimensões, com centro na origem, raio do eixo dado por $q\chi^2(q,n)\sqrt{\vec{e}}$ sendo $q\chi^2(q,n)$ a função quantil Qui-Quadrado para uma distância q do centro da distribuição e n graus de liberdade, alinhado com os ângulos A. Por exemplo $q = 0.997$ é o equivalente a uma distância de 3σ da média no caso de $n = 1$. A seguir constrói-se uma matriz de dimensões $2n+1 \times n$ que será usada posteriormente:

$$M = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ r_1 & 0 & \cdots & 0 \\ 0 & r_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & r_n \\ -r_1 & 0 & \cdots & 0 \\ 0 & -r_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -r_n \end{pmatrix} \tag{A.11}$$

E finalmente calcula-se a matriz B de dimensões de n colunas (variáveis) e $2n+1$ linhas, sendo que cada linha representa uma simulação:

$$B = (MA\vec{\sigma} + \vec{\mu})^T \tag{A.12}$$

Assim, a matriz B, com $2n+1$ linhas é uma seleção de pontos artificialmente criados para representar a matriz da simulação Monte Carlo Γ^M. O ganho de performance obtido pela metodologia está no fato de que Γ^M tem N linhas, e para um número pequeno de variáveis (como é o caso em circuitos específicos como células de uma biblioteca, célula de memória SRAM, etc), $N >> 2n+1$. A figura A.4 mostra os pontos selecionados pela metodologia para um inversor.

O próximo passo da metodologia consiste em rodar $2n+1$ simulações elétricas. Após a execução das simulações, faz-se um ajuste de função a fim de encontrar uma função que relacione as variáveis aleatórias com o resultado das simulações (atraso, potência, etc). O ajuste e seleção de modelo não-linear é dividido em três etapas:

1. **Ajuste inicial:** fazer ajuste linear aos dados;
2. **Redução de variáveis:** remover termos insignificantes;
3. **Melhoria do modelo:** interativamente adicionar termos não lineares e termos cruzados.

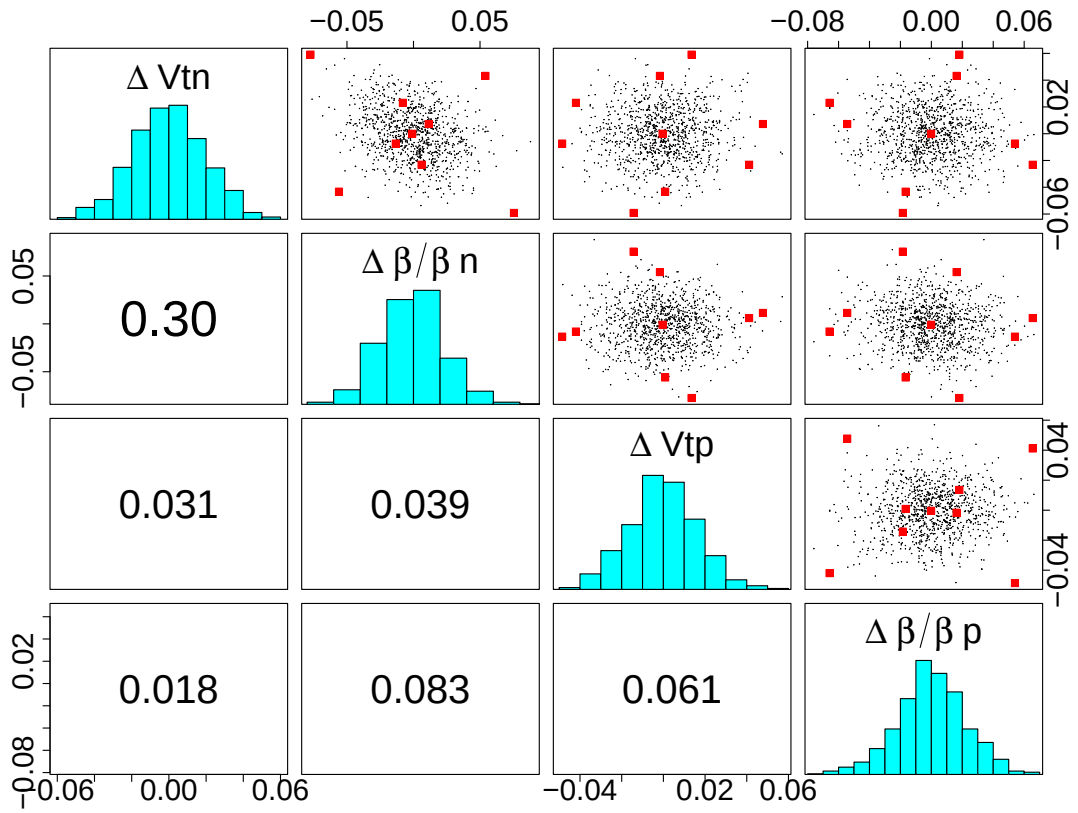


Figura A.4: Diagonal superior: pares de distribuições de Vt e β de um inversor. Os pontos maiores são os pontos propostos pelo Brussel DoE, enquanto os pontos pequenos correspondem aos 1,000 pontos de Monte Carlo. Diagonal inferior: coeficientes de correlação. Diagonal: histogramas.

Inicialmente é feito um ajuste linear de forma que:

$$H_i = \alpha_{1_1} x_{1_i} + \alpha_{2_1} x_{2_i} + \ldots + \alpha_{n_1} x_{n_i} + \varepsilon_i \tag{A.13}$$

onde H_i é a saída da i-ésima simulação elétrica sendo $1 \leq i \leq 2n+1$, sendo x_j a variável onde $1 \leq j \leq n$. O Método dos Mínimos Quadrados busca minimizar a soma dos quadrados dos resíduos conforme $\sum_{i=0}^{2n+1} \varepsilon_i^2$.

Foi desenvolvido um algoritmo interativo que remove variáveis que não tornam o fit melhor. Por exemplo, o atraso de subida de um inversor não tem relação (ou pode-se dizer que tem uma relação tão baixa que é desprezível) com o Vt do transistor NMOS. O algoritmo é baseado no Critério de Informação Bayesiano (BIC), proposto por Schwarz (1978). A listagem A.1 apresenta o algoritmo de redução de variáveis.

```
repetir {
  para cada variável x_i da função f {
    f_o ← remove termo x_i da função f
    Se  BIC(f_o) < BIC(f) {
      armazena f_o na lista L ordenada por BIC(f_o)
    }
  }
  f ← seleciona modelo da lista L com menor BIC
}até modelo não melhorar
```

Listing A.1: Redução de variáveis

O resultado da execução do algoritmo A.1 é uma função de ajuste linear que contém menos termos do que o modelo linear completo descrito pela eq. A.13. Além de mais compacto, sendo o algoritmo guloso, a função apresenta um fit melhor do que o fit inicial, segundo o critério de informação bayesiano.

O passo seguinte da metodologia é uma continuação do passo anterior no sentido que se busca uma função de ajuste com BIC ainda melhor. O BIC pode ser melhorado ainda mais através da inserção de termos quadráticos e de mais alta ordem, assim como termos cruzados (um termo linear multiplicando outro termo linear, assim como termos quadráticos multiplicando termos lineares e assim por diante). A listagem A.2 mostra o método utilizado para melhoria da função de ajuste. O método guloso pára de inserir termos quando não é possível obter um fit com BIC melhor ou quando o número de termos ultrapassa um limite pré-determinado: o número de termos da função de ajuste deve ser menor do que o tamanho da amostra (2n+1).

```
repetir{
  para cada variável x_i da função f{
  para k=1..z {
    f_add ← adiciona termo x_i^k
    guarda f_add na lista L ordenado por BIC(f_add)
  }
   f_remove ← remove termo x_i
   guarda f_remove na lista L ordenado por BIC(f_remove)
   para cada variável x_j da função f{
     f_cross-term ← adiciona termo x_i × x_j
     guarda f_cross-term na lista L ordenado por BIC(f_cross-term)
   }
 }
 se ( melhor BIC guardado em L < BIC(f)){
   f ← escolhe modelo de L com menor BIC
   N_coeff ← número de coeficientes de f
 }
} até modelo não melhorar OU N_coeff > 0.6N_doe
```

Listing A.2: Melhoria do modelo

A.3 Caracterização de Biblioteca de Células

Esta seção apresenta resultados da caracterização de um subconjunto de células de uma biblioteca para um nó tecnológico de 32nm. As células selecionadas são apresentados na tabela A.1. As bibliotecas geradas pelas ferramentas são compatíveis com o formato Liberty da Synopsys (.LIB), o qual as ferramentas Cadence também dão suporte. A caracterização da biblioteca estatística foi realizada usando as 3 metodologias de análise:

1. *Monte Carlo:* método de referência, sendo 1000 o tamanho da amostra;

2. *propagação de erro usando análise de derivada linear:* exige n +1 simulações e calcula o desvio padrão através de propagação de erro, derivadas são calculadas numericamente;

Tabela A.1: Comparação de Propagação de Erros e Metodologia de Superfície de Resposta com Monte Carlo.

Porta	**Param**	**Metodologia de Superficie de Resposta**						**Propagação de Erros**					
		μ_{err} (%)	σ_{err} (%)	S_{err} (%)	K_{err} (%)	$3\sigma_{err}$ (%)	r.	μ_{err} (%)	σ_{err} (%)	S_{err} (%)	K_{err} (%)	$3\sigma_{err}$ (%)	r.
INV	delay	0	0.5	-13.3	0.7	0	9	-0.1	-2.5	-100	2.2	-0.9	5
	transition	0	1.7	-27.4	6.2	-0.2		-0.2	-10.8	-100	-17.9	-2	
	power	-1.6	-0.7	-7.5	0	-1.4		NA	NA	NA	NA	NA	
NAND2	delay	0	0.9	-11.9	-4.5	0	17	-0.7	-17.9	-100	2.2	-3	9
	transition	0	1.5	10.8	4.1	0.5		-0.1	8.8	-100	12.2	0.3	
	power	0.4	-7.5	-1.1	-5	5.4		NA	NA	NA	NA	NA	
NOR2	delay	-0.1	-2.4	-25.8	3.9	1	17	0	-4.6	-100	-6.8	3.7	9
	transition	0.1	3.1	-26.5	-3.1	0		0	-4.3	-100	-3.4	-1.2	
	power	-0.6	3.8	22.8	-0.9	0		NA	NA	NA	NA	NA	
XOR2	delay	0.1	0	16.8	0.7	0.1	41	0.2	-6.4	-100	5.4	-0.5	21
	transition	0	3.4	-28.3	1.5	0.1		-0.2	-8.6	-100	8.4	1.5	
	power	0	-5.5	-43.5	4.1	1.7		NA	NA	NA	NA	NA	
DDFQ	hold	-0.8	1.1	-10.7	2.2	3.9	97	5.5	-3.1	-100	4.1	11.3	49
	setup	0.4	0	-33.1	0.2	1.1		-3.9	-1.6	-100	-1.9	2.3	
	CLK-Q	0	1.4	-8.6	-1	0.1		-0.5	-7.5	-100	-1.5	2.7	
	power	0	2.5	-22.4	-5	0		NA	NA	NA	NA	NA	
||Média|| (tempo)		0.14	1.45	19.38	2.55	0.64		1.04	6.92	100	6	2.67	

3. *Non-Linear RSM:* metodologia de superfície de resposta, usando Brussel DoE e método para busca de fit não linear.

Cada uma das iterações de caracterização da biblioteca de células leva aproximadamente três minutos em um servidor com 10 processadores. A célula que consome mais tempo é o flip-flop, o qual leva cerca de 90% do tempo da caracterização do grupo. Assim, o tempo total de caracterização das 1000 iterações de Monte Carlo é de 49 horas. Usando o mesmo ambiente paralelo, a caracterização leva apenas uma fração desse tempo para as alternativas: 2 horas para a propagação de erros usando derivadas numéricas e 4 horas para o RSM não-linear.

A tabela A.1 mostra a comparação entre os momentos das distribuições geradas usando os 3 diferentes métodos de caracterização citados acima. Para cada parâmetro, a tabela mostra o erro relativo entre os quatro momentos das distribuições: média (μ_{err}), desvio-padrão (σ_{err}), assimetria (S_{err}) e curtose (K_{err}). O terceiro e quarto momentos indicam respectivamente o grau de assimetria e o peso da cauda da PDF resultante. Além disso, a tabela apresenta 3 σ_{err} , que é o erro de as aproximações a uma distância de $99,97\%$ da média da distribuição. Esse dado mostra o qualidade da aproximação na cauda das distribuições.

A última linha da Tabela 7.2 apresenta a média dos erros absolutos de tempos de atraso e de transição. A potência não é levada em conta para cálculo das médias, porque esta informação não é disponível para a propagação de erros. Os erros de **desvio padrão** e **média** são abaixo de 2% para a Metodologia Superfície de Resposta (RSM), em comparação com os erros de 7% quando se utiliza propagação de erro (EP). Observe que EP limita a distribuição a ser tratada como Normal, e assim, por definição, a sua saída é sempre limitada a $S = 3$ e $K = 0$.

A coluna “r” é o número de simulações elétricas necessárias por RSM e EP. É importante notar que para ambos os métodos o número de simulações elétricas é linearmente dependente do número de transistores do dispositivo, enquanto o número de simulação Monte Carlo é arbitrário. As simulações elétricas são o passo mais demorado de Monte Carlo, EP e RSM. Uma simulação leva exatamente a mesma quantidade de tempo para cada uma dessas metodologias. Assim, o número de simulações elétricas é a métrica mais representativas de desempenho. O número de simulações necessárias por EP e RSM é $n+1$ e $2n+1$, respectivamente, sendo US n o número de variáveis. Embora, o número de rodadas Monte Carlo é independente do número de entradas. Nossa referência usa tamanho da amostra de 1000, mas este número poderia ser aumentado para maior precisão.

Sendo o tempo de execução de EP e RSM linear número de transistores, a aceleração destes sobre Monte Carlo é inversamente proporcional à complexidade da porta. Considerando 2 variáveis por transistor (Vt e β), isso limita a aplicabilidade de RSM para circuitos com menos de $N/4transistores$, onde N é o número de simulações de Monte Carlo. Contudo propagação de erros apresenta ganho de desempenho sobre Monte Carlo para circuitos com até $N/2$ transistores.

A figura A.5 apresenta a distribuição de tempo de espera (hold time) do flip-flop (FF). No inset do gráfico é mostrado o histograma de Monte Carlo (referência), bem como as curvas que representam os PDFs obtidas usando propagação de erros e RSM não-linear. Como PDF e histograma em escala linear não têm informação suficiente a respeito da cauda da distribuição, o gráfico principal mostra o Quantil-quantil plot (q-q plot), uma ferramenta muito difundida entre a comunidade de Estatística. O eixo x mostra o quantil da distribuição, i.e. distância em desvios padrões da média, e o eixo y mostra o tempo de espera (hold time). Usando esta técnica permite-nos verificar que o RSM não-linear tem concordância perfeita com as simulações de Monte Carlo de referência em todo o domínio da distribuição: no centro e as caudas. Por outro lado, a propagação de erros usando análise de derivadas linear nesse caso apresentou um erro de aproximadamente -1 % na média da distribuição (quantil zero), mas torna-se mais imprecisa nas caudas da distribuição.

A.4 Conclusões

Este trabalho de doutorado apresenta um estudo sobre análise estatística de circuitos integrados. Metodologias para simulação de variabilidade do processo, ruído e envelhecimento são propostos e testados em circuitos estudo de caso.

Diferentes metodologias de simulação são empregadas para analisar o impacto das variações a diferentes classes de circuitos (células de uma biblioteca, caminhos lógicos, memória e árvore de clock). Contudo, este resumo em portugês apresenta somente dois dos tópicos abordados no doutorado:

- modelagem dinâmica em tempo de simulação de fenômenos de captura e emissão de elétrons por armadilhas de interface e
- metodologias de caracterização estatística de biblioteca de células considerando variabilidade no processo de fabricação.

A possibilidade de relação entre sinais aleatórios telegráficos (RTS) e instabilidade acelerada por temperatura e tensão (NBTI) surgiram recentemente a partir da possibilidade de se explicar o comportamento de NBTI com modelos já estabelecidos de RTS. Isso

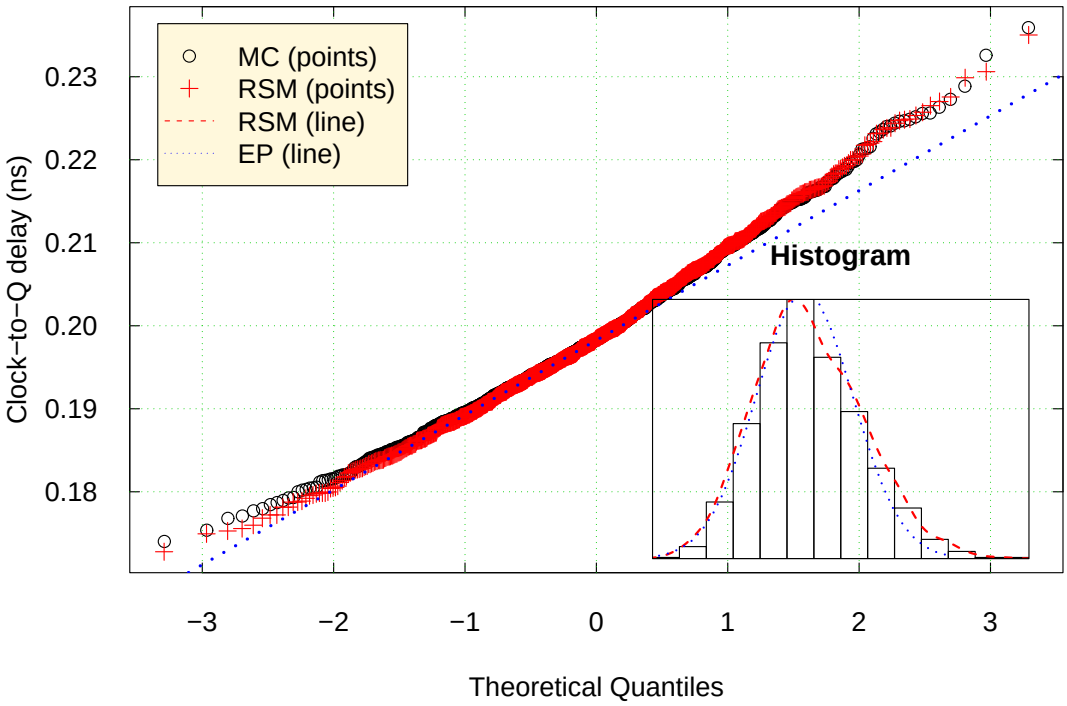


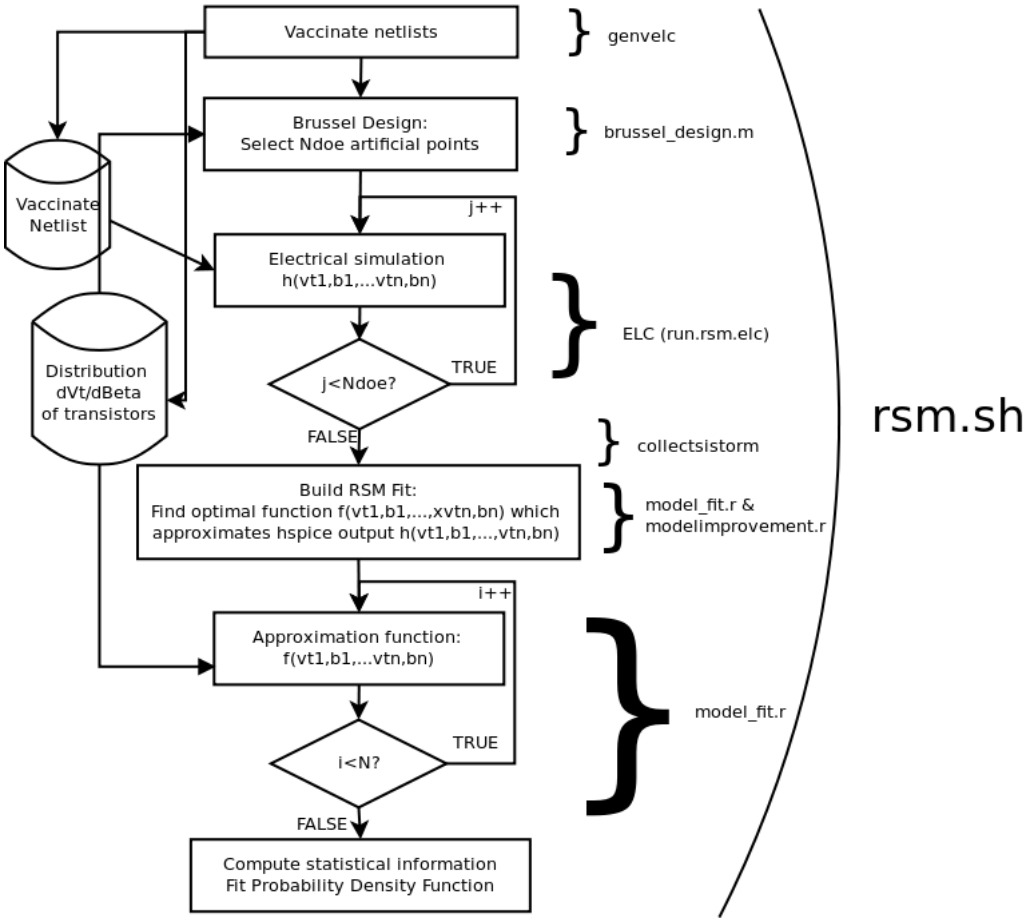
Figura A.5: Gráfico quantil-quantil e histograma do tempo de espera (hold time) do flip-flop.

se deve ao fato de que os modelos de NBTI existentes, baseados no modelo de reação-difusão, não são capazes de explicar medidas experimentais. Num trabalho de cooperação entre o nosso grupo de pesquisa e o grupo de confiabilidade no IMEC começou-se a investigar essa relação e propôs-se um modelo capaz de corresponder melhor aos dados experimentais.

A implementação de RSM foi integrada às ferramentas de análise estatística do IMEC. O fluxo de RSM para a caracterização do circuito é protegido sob patente nos Estados Unidos da América e na União Européia, das quais o proponente desta tese de doutorado é co-inventor. Propagação de erros utilizando derivadas numéricas e RSM apresentaram erros médios abaixo de 2% em relação a Monte Carlo para a caracterização de uma biblioteca de tecnologia de 32nm, com um ganho de performance na ordem de $10\times$.

APPENDIX B INTEGRATION OF RESPONSE SURFACE METHODOLOGY FLOW INTO CELL CHARACTERIZATION USING VAM (DOCUMENTATION AND USER GUIDE)

B.1 RSM Flow for cell library characterization



B.2 RSM files

B.2.1 Top-level files

run_rsm.sh: wrapper script kept in the same directory of the vaccinate's config.m; setups variables and calls rsm.sh.

rsm.sh: top-level flow bash script. It takes input setup parameters and automatically calls all the required steps.

gen_brussel_patch.sh: patch mode for regenerating Brussel Design points part 1.

fit_patch.sh: patch mode part, for fitting regression and sampling from propagation function.

B.2.2 Main algorithms

brussel_design.m: Mathematica code for generating Brussel DoE points.

model_fit.r: R code for reading brussel design inputs and outputs, Monte Carlo inputs, reading configuration for model fit, and generates the MC output (based on the propagation function computed). Iterates over all the circuit metrics. Full Linear and simplified linear models are generated by this script.

modelimprovement.r: R code that takes a full linear model and a simplified linear model and, by iterativelly adding quadratic, adding cross-terms, removing terms, finds the best surrogate model to the circuit metrics.

B.2.3 Auxiliary files

convert_brussel_csv_lib.pl: Perl code for converting csv files into VAM's parameters .lib.

parse_lib.pl: Perl code to to convert vaccinated dVt and dBeta in .lib to .csv.

B.3 Running the RSM flow for cell characterization

B.3.1 Required files in the vaccinate directory

Create a new directory, as would create for running conventional VAM characterization flow. Copy the required files: clean, sourceme, elccfg, micron.setup (setup file), part2.elc, preamble.elc, compactmodel.xml, run1.csh, run_rsm.sh (top-level for rsm flow).

B.3.2 Edit run_rsm.sh

Manually edit the configuration variables:

rsm_dir: directory containing the RSM files mentioned in section "RSM files".

vam_dir: directory containing VAM code.

base_dir: vaccinate top-level folder (created in "Required files in the vaccinate directory").

geos: geometries to apply RSM (matched, c2c). Enter list of geometries between parenthesis separated by space; e.g. (matched c2c) or (matched)

configm: VAM configuration file.

sigma_distance_brussel [float 0-inf :] Controls the distance of the DoE points from the average. Lower number gives a better approximation in the center of the distribution, higher gives a better approximation in the tails. Recommended: 3.

accuracy_brussel_doe[1-2]: an integer either 1 or 2 specifying the number of DOE points generated in relation to the number of variable n:

1. n+1 points, allows sufficient accuracy only for full linear propagation model;
2. 2n+1 points, allows sufficient accuracy for quadratic model including cross-terms.

accuracy_fit [1-5 :] an integer between 1 and 5 to control accuracy of the propagation function, 1 is fastest and 5 is most accurate:

1. Full linear;
2. Reduced linear model;
3. Quadratic terms;
4. Quadratic terms and allows insertion of linear terms while searching quadratic ones;
5. Quadratic and cross-terms.

ncpu: number of cpus to be used for the model selection algorithm. The algorithm is massively parallel and speed scales linearly with number of cpus. Recommended: use all cpus available in the machine.

B.3.3 Patch options (re-run)

Patch[0-1]: 0 is normal mode and 1 means patch Brussel design points mode. To be used if ELC failed to characterize some runs of a circuit. It happens because for a given combination of Vt and Beta ELC cannot compute some metric, and a liberty file is not generated for that Brussel run.

patch_ckt: circuit to be patched.

patch_runs: runs that failed, to be patched.

patch_g: geometry to be patched.

patch_sigma_distance_brussel_doe [float 0-sigma_distance_brussel_doe :] Controls the distance of the DoE points from the average for the patch run. Lower number gives a better approximation in the center of the distribution, higher gives a better approximation in the tails. As a simulation has already failed using the previous sigma_distance_brussel_doe use a smaller value, recommended: 2 or 1.

APPENDIX C LIST OF PUBLICATIONS (2008-2011)

C.1 Journals

1. Brusamarello, Lucas ; Wirth, Gilson I. ; Roussel, Philippe ; Miranda, Miguel . Fast and accurate statistical characterization of standard cell libraries. Microelectronics and Reliability, In press, 2011.

2. Camargo, Vinícius; Ashraf, Nabil; Brusamarello, Lucas; Vasileska, Dragica; Wirth, Gilson. Impact of RDF and RTS on the performance of SRAM cells. Journal of Computational Electronics, online November 2010.

3. Brusamarello, Lucas; Neuberger, Gustavo; Wirth, Gilson; da Silva, Roberto; Reis, Ricardo; Murgai, Rajeev; Reddy, Subodh; Walker, William. Statistical analysis of hold time violations. Journal of Computational Electronics, online October 2010.

4. da SILVA, Roberto ; BRUSAMARELLO, Lucas ; WIRTH, Gilson . Statistical fluctuations for the noise current from random telegraph signals in semiconductor devices: Monte Carlo computer simulations and best fits. Physica. A (Print), v. 389, p. 2687-2699, 2010.

5. Brusamarello, Lucas; Wirth, Gilson I.; da Silva, Roberto. Statistical RTS model for digital circuits. Microelectronics and Reliability. , v.49, p.1064 - 1069, 2009.

6. da Silva, Roberto, Wirth, Gilson Inacio, Brusamarello, Lucas. An appropriate model for the noise power spectrum produced by traps at the Si-SiO interface: a study of the influence of a time-dependent Fermi level. Journal of Statistical Mechanics. Theory and Experiment. 2008.

7. Brusamarello, Lucas; da Silva, Roberto; Wirth, Gilson I.; Reis, Ricardo A. L. Probabilistic Approach for Yield Analysis of Dynamic Logic Circuits. IEEE Transactions on Circuits and Systems. I, Regular Papers. , v.55, p.2238 - 2248, 2008.

C.2 Conferences

1. Miranda, Miguel ; Roussel, Philippe ; Brusamarello, Lucas ; Wirth, Gilson Inacio . Statistical Characterization of Standard Cells Using Design of Experiments with Response Surface Modeling. In: Design Automation Conference, 2011, San Diego. Design Automation Conference 2011, 2011.

2. da Silva, Mauricio B ; Camargo, Vinicius V. A. ; BRUSAMARELLO, Lucas ; Wirth, Gilson ; da Silva, Roberto . NBTI-aware technique for transistor sizing of high-performance CMOS gates. In: LATW '09. Buzios, RJ. Proceedings of the 10th Latin American Test Workshop. Piscataway, USA : IEEE, 2009. p. 13-18.

3. BRUSAMARELLO, Lucas ; Camargo, Vinicius V. A. ; da Silva, Mauricio B ; WIRTH, G. I. ; SILVA, Roberto da ; GLOESEKOETTER, P. . Numerical Method for Modeling Process Variations and NBTI. In: IFIP/IEEE VLSI-SOC 2008 Conference on Very Large Scale Integration System-on-Chip, 2008, Rhodes Island. FIP/IEEE VLSI-SOC 2008. Piscataway, USA : IEEE, 2008. p. 514-518.

C.3 Patents

1. MIRANDA, M.; ROUSSEL, P.; BRUSAMARELLO, L. Response Characterization Of An Electronic System Under Variability Effects. US Patent Application No. US 2011/0178789 A1 , (publication date Jul. 21, 2011).

2. MIRANDA, M.; ROUSSEL, P.; BRUSAMARELLO, L. Response Characterization Of An Electronic System Under Variability Effects. European Patent Application No. 10189434.3, Unpublished (filing date Oct. 29, 2010).