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

DIGEORGIA NATALIE DA SILVA

An Estimation Method for Gate Delay Variability in Nanometer CMOS Technology

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

Prof. Dr. Renato Perez Ribas
Advisor

Prof. Dr. André Reis
Co-advisor

Porto Alegre, August 2010.

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Da Silva, Digeorgia Natalie

An Estimation Method for Gate Delay Variability in Nanometer CMOS Technology / Digeorgia Natalie da Silva – Porto Alegre: Programa de Pós-Graduação em Microeletrônica, 2010.

153 f.:il.

Tese (doutorado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Microeletrônica. Porto Alegre, BR – RS, 2010. Advisor: Renato Perez Ribas.

1.Tecnologia CMOS. 2.Variabilidade. 3.Atraso. 4. Redes de Transistores. I. Ribas, Renato Perez. II. Título.

ACKNOWLEDGEMENTS

I would like to thank my advisor, Prof. Renato Perez Ribas, who guided me throughout my PhD thesis work and provided me with the opportunity to work on a very important topic in Microelectronics. Also, I thank Professor André Inácio Reis for his precious suggestions and support. I am thankful to Prof. Sachin Sapatnekar for receiving me at the University of Minnesota and teaching me an important background to be used in my research.

I'm grateful to my laboratory colleagues and friends Carlos Klock, Diogo Silva, Felipe Marques, Nívea Schuch, Pedro Paganela, Rafael da Silva, Vinicius Callegaro, Vinicius Dal Bem, and especially Caio Alegretti, Paulo Butzen and Oswaldo Martinello for their company and their help with discussions that enriched my work. Also, I would like to thank Alessandro Goulart for all his assistance and Leomar da Rosa Júnior for his precious advices and talks when I first got into the group.

To my other colleagues and friends at UFRGS, Adriel Zsiemer, Cristina Meinhardt, Edgar Correa, Felipe Pinto, Giovani Pesenti, Glauco Valim, Guilherme Flach, Gustavo Wilke and Tatiana Marcondes, I'd like to say that it would be very hard to go through all this journey without the friendship and the amazing moments you guys provided me with.

I specially thank Lucas Brusamarello, a very important person in my life, for being a loyal friend and the best company ever.

I thank my godmothers, Francisca and Escolástica, and my father, Geraldo Ramos, for their love and support. I thank my brother, Dangelis, for being so helpful, compassionate, loyal and supportive. Finally, I thank my mother, Olita Ferreira, for her endless love and admirable example.

To my brother

TABLE OF CONTENTS

LIST OF ABREVIATIONS **8**
LIST OF FIGURES **9**
LIST OF TABLES **12**
RESUMO **14**
ABSTRACT **15**
1 INTRODUCTION **16**
2 DIGITAL DESIGN AND PARAMETER VARIABILITY IN INTEGRATED CIRCUITS **18**
2.1 Digital Integrated Circuit Design **18**
2.1.1 Logic Styles 20
2.1.2 Logic Gates with Pull-up and Pull-down Networks 20
2.1.3 Cascode Voltage Switch Logic 22
2.1.4 Pass-Transistor Logic 22
2.1.5 Logic Functions and Transistor Networks 22
2.1.6 Standard Cell Design Flow 23
2.2 Variability in Integrated Circuits **24**
2.2.1 Sources of Process Variations 24
2.2.1.1 Photolithography 24
2.2.1.2 Etch 25
2.2.1.3 Line Edge Roughness (LER) 25
2.2.1.4 Chemical Mechanical Polishing (CMP) 25
2.2.1.5 Random Dopant Fluctuations (RDF) 26
2.2.2 Variability of Process Parameters 26
2.2.2.1 Variability of Gate Length 26
2.2.2.2 Variability of Thin Film Thickness 26
2.2.2.3 Variability of Interconnect and Dielectric 26
2.2.3 Variability of Device Characteristics 27
2.2.4 Physical Variability Due to Aging and Wearout 27
2.2.5 Environmental Variations 27
2.2.6 Variations due to Modeling 27
2.2.7 Typical Values for the Parameter Variations 28
2.2.8 Spatial Scales of Variations 28
2.2.8.1 Global Variations 28
2.2.8.2 Local Variations 28
2.2.8.3 Spatially Correlated Variations 29
2.2.8.4 Independent Variations 29
2.2.9 Parametric Yield 29
2.2.10 Considerations 31

2.3 Analysis of the Impact of Parameter Variation on the Threshold Voltage Variation 32
2.3.1 Variations due to Random-Dopant Fluctuations, Channel Length and Oxide Thickness Variations 32
3 TIMING ANALYSIS 36
3.1 Critical Path Method (CPM) 36
3.2 Statistical Concepts 37
3.3 Statistical Static Timing Analysis (SSTA) 39
3.4 SSTA Solution Approaches 39
3.4.1 Numerical Integration Method 39
3.4.2 Monte-Carlo Method 39
3.4.3 Probabilistic Analysis Methods 40
3.5 Delay Modeling 40
3.5.1 Introduction 40
3.5.1.1 Delay 41
3.5.1.2 Transition Time 41
3.5.2 Gate and Interconnect Timing Models 41
3.5.3 Elmore Delay Model 42
3.5.3.1 RC Tree 42
3.5.4 Asymptotic Waveform Evaluation (AWE) 43
3.6 Process Variation Modeling 46
3.6.1 Statistical Delay Models 46
3.6.2 Pelgrom Model 47
4 PROPOSAL AND METHODOLOGY 49
4.1 Introduction 49
4.2 Motivation 49
4.3 Proposal 51
4.4 Methodology 51
5 CMOS LOGIC GATE PERFORMANCE VARIABILITY 54
5.1 Introduction 54
5.2 Variability of Different Transistors Networks 54
5.2.1 Pull-down Network 55
5.2.2 Pull-up Network 56
5.3 CMOS Inverter 56
5.3.1 Analysis 57
5.3.2 Inverter Sizing 57
5.3.3 Output Load 60
5.3.4 Input Transition Time 61
5.3.5 Transistor Network Arrangements 62
5.4 NAND and NOR Gates 65
5.5 NAND: Single Gate Versus Mapped Circuit 68
5.6 And-Or-Inverter (AOI) Logic Gates 70
5.7 Conclusion 72
6 VARIABILITY ESTIMATION METHOD 73
6.1 On-Resistance 75
6.1.1 Response Surface Methodology 76
6.2 MOS Structures Capacitances 77
6.2.1 Gate Capacitance 78
6.2.1.1 Channel Capacitance 78

6.2.1.2 Overlap Capacitance 78
6.2.2 Junction Capacitances 79
6.2.2.1 Bottom-Plate Junction Capacitance 79
6.2.2.2 Side-Wall Junction Capacitance 80
6.2.3 The Inverter 80
6.3 Modeling the Falling-Edge Delay Deviation of a NAND3 81
7 MODELING THE DELAY VARIABILITY OF TRANSISTOR NETWORKS 83
7.1 Calculation of Resistances 83
7.2 Resistance of Parallel Transistors 84
7.3 Resistance of Series Transistors 85
7.4 Estimation of Performance Deviation 86
7.4.1 Series Networks 87
7.4.2 Parallel Networks 89
7.4.3 Delay Variability Estimation Method Considering the Saturated Region of Operation 90
7.4.4 Estimation Method Applied to Different Inverter Topologies 92
7.4.5 The Influence of the Sizing of Transistors on Delay Variability 93
7.4.6 The Influence of the Output Load on Delay Variability 95
7.4.6.1 The Series Transistors Networks 95
7.4.7 Inverter Chains 97
7.5 Conclusion 98
8 EVALUATION OF THE PROPOSED DELAY VARIABILITY MODEL 99
8.1 2-Input XOR Logic Gate 99
8.2 4-Input XOR Logic Gate 106
8.3 Full Adder 112
8.4 Complex Gate 116
8.5 Delay Equation Method 117
8.6 Runtime Analysis 118
8.7 Conclusion 120
9 CONCLUSION 121
REFERENCES 123
APPENDIX A RESISTANCE EQUATIONS SCRIPT 128
APPENDIX B DELAY CALCULATION SCRIPT 130
APPENDIX C DELAY EQUATION SCRIPT 134
APPENDIX D RESUMO DA TESE EM PORTUGUÊS 135

LIST OF ABREVIATIONS

ASIC	Application-Specific Integrated Circuits
AWE	Asymptotic Waveform Evaluation
BDD	Binary Decision Diagram
CAD	Computer-Aided Design
CMOS	Complementary Metal Oxide Semiconductor
CSP	Complementary Series-Parallel
CVSL	Cascode Voltage Switch Logic
DCVSL	Differential Cascode Voltage Switch Logic
DEM	Delay Equation Method
DPTL	Differential Pass Transistor Logic
IC	Integrated Circuit
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
NCSP	Non-Complementary Series-Parallel
PDF	Probability Density Function
PTL	Pass Transistor Logic
PTM	Predictive Technology Model
RSM	Response Surface Methodology
SSTA	Statistical Static Timing Analysis
STA	Static Timing Analysis

LIST OF FIGURES

Figure 2.1: Design abstraction levels in digital circuits. 19
Figure 2.2: Complementary logic gate as a combination of a pull-up and a pull-down network 21
Figure 2.3: Different implementations of a logic function 21
Figure 2.4: Basic principle of the Cascode Voltage Switch Logic style 22
Figure 2.5: Pass-transistor implementation of an AND gate 22
Figure 2.6: Digital circuit design methodology using predefined cell library 24
Figure 2.7: Leakage current and frequency measured in a sample of 1000 chips 29
Figure 2.8: Yield window for frequency constraint of fmin=0.9 fnom and power constraint of $P_{max} = 1.05\ P_{nom}$, for negligible static (leakage) power 30
Figure 2.9: Yield window for frequency constraint of fmin=0.9 fnom and power constraint of $P_{max} = 1.05\ P_{nom}$, for static (leakage) power 31
Figure 2.10: Threshold voltage PDFs of NMOS for process parameter variations 34
Figure 2.11: Threshold voltage PDFs of PMOS for process parameter variations 35
Figure 3.1: An example of a circuit and its timing graph 36
Figure 3.2: Normal distribution of a random variable 38
Figure 3.3: Gate and interconnect delays represented as probability density functions 40
Figure 3.4: The 50% delay and transition time of a waveform 41
Figure 3.5: A step response e(t) and its derivative 42
Figure 3.6: An example of an RC tree 43
Figure 3.7: RC system 44
Figure 3.8: Current-based circuit model for a logic cell 46
Figure 3.9: Representation of transistors that lie on th x-axis 48
Figure 5.1: Some of the CMOS logic gates used for the analysis of falling-edge delay variability in relation to the transistor network arrangement 55
Figure 5.2: Some of the CMOS logic gates used for the analysis of rising-edge delay variability in relation to the transistor network arrangement 56
Figure 5.3: Design for analysis 57
Figure 5.4: Mean subthreshold and maximum currents of an inverter in relation to its area 58
Figure 5.5: Normalized current deviations of an inverter 58
Figure 5.6: Timing metrics of an inverter in relation to its area 59
Figure 5.7: Normalized timing metrics deviation of an inverter 59
Figure 5.8: Timing metrics of an inverter 60
Figure 5.9: Normalized timing metrics deviation of an inverter 61
Figure 5.10: Topology using an auxiliary inverter (X1-X5) connected to the input of the main inverter for changing the input slope 61
Figure 5.11: Timing metrics of an inverter 62

Figure 5.12: Normalized timing metrics deviation of an inverter 62
Figure 5.13: Different topologies of an inverter 63
Figure 5.14: Folding topology of an inverter 64
Figure 5.15: Normalized rise and fall delay deviations in relation to the number of inputs: NAND and NOR gates 66
Figure 5.16: Comparison of N- and PMOS transistor stacking in NAND and NOR gates for different positions of the switching device 67
Figure 5.17: Illustration of a single 3-input NAND gate implemented by using two 2-input NAND gates 68
Figure 5.18: PDF of rise delay for a 3-input NAND gate and a circuit performing the same logic function implemented by using two 2-input NAND gates for the best and the worst delay propagations 69
Figure 5.19: PDF of rise delay for a AOI-21 and AOI-32 gates implemented by using basic CMOS cells and as a single complex gate 71
Figure 6.1: Combinations of *min* and *max* values considered for the threshold voltages (coded variables) of devices in a 3-transistors network 74
Figure 6.2: Delay variability model methodology 74
Figure 6.3: Intrinsic capacitances of a MOS transistor 77
Figure 6.4: CMOS Inverter. 80
Figure 6.5: Switch models of CMOS inverter. 80
Figure 6.6: Pull-down network of a 3-input NAND. 81
Figure 7.1: NMOS transistors stacking 83
Figure 7.2: Probability density functions (PDF) of the resistances that constitute models for the transistors in a NMOS series network 84
Figure 7.3: PMOS transistors stacking 86
Figure 7.4: Simulated and fitted curves for rising-edge delay deviation in relation to the output load of a 4-stacking PMOS. 95
Figure 7.5: Modeled and fitted curves for falling-edge delay deviation in relation to the output load of a 4-stacking NMOS using Elmore Delay model. 96
Figure 7.6: Simulated points for falling-edge delay deviation in relation to the output load of a 4-stacking NMOS. 96
Figure 7.7: Modeled and fitted curves for falling-edge delay deviation in relation to the output load of a 4-stacking NMOS using AWE. 97
Figure 7.8: Measurements structures 97
Figure 8.1: 2-input XOR implemented in different logic styles and topologies 99
Figure 8.2: PDF of the rising-edge delay for the complex gate implementation of a 2-input XOR 102
Figure 8.3: PDF of the rising-edge delay for the implementation of a 2-input XOR with NAND gates 102
Figure 8.4: PDF of the rising-edge delay for the PTL implementation of a 2-input XOR 103
Figure 8.5: PDF of rising-edge delay for different implementations of a 2-input XOR 103
Figure 8.6: PDF of the falling-edge delay for the complex gate implementation of a 2-input XOR 104
Figure 8.7: PDF of the falling-edge delay for the implementation of a 2-input XOR with NAND gates 104
Figure 8.8: PDF of the falling-edge delay for the PTL implementation of a 2-input XOR 105

Figure 8.9: PDF of the falling-edge delay for different implementations of a 2-input XOR 105
Figure 8.10: 4-input XOR implemented in different logic styles and topologies 106
Figure 8.11: PDF of the rising-edge delay for the DCVSL implementation of a 4-input XOR 108
Figure 8.12: PDF of the rising-edge delay for the DPTL implementation of a 4-input XOR 108
Figure 8.13: PDF of rising-edge delay for different implementations of a 4-input XOR 109
Figure 8.14: PDF of the falling-edge delay for the DCVSL implementation of a 4-input XOR 110
Figure 8.15: PDF of the falling-edge delay for the DPTL implementation of a 4-input XOR 110
Figure 8.16: PDF of falling-edge delay for different implementations of a 4-input XOR 111
Figure 8.17: PDF of falling-edge delay for DPTL implementation of a 4-input XOR modeled with AWE technique 112
Figure 8.18: Full Adder implemented in different logic styles 113
Figure 8.19: "SUM" output node PDF of the rising-edge delay for different implementations of a full adder 115
Figure 8.20: "SUM" output node PDF of the falling-edge delay for different implementations of a full adder 115
Figure 8.21: A complex gate implementation 116
Figure 8.22: PDF of the rising-edge delay for the complex gate implementation 117
Figure 8.23: PDF of the rising-edge delay for the complex gate implementation 117

LIST OF TABLES

Table 2.1: 3σ Variation points for the technology parameters of a typical 0.18 μm CMOS process 28
Table 2.2: V_{TH} deviation according to variations in the channel length, oxide thickness and RDF for NMOS and PMOS transistors 35
Table 5.1: Normalized fall delay deviations for different topologies..... 55
Table 5.2: Normalized rise delay deviations for different topologies 56
Table 5.3: Subthreshold leakage current values for the inverter topologies 63
Table 5.4: Rise and fall delays for the inverter topologies..... 63
Table 5.5: Rise and fall transition times for the inverter topologies 64
Table 5.6: Delay deviations for the shortest and the longest paths in Fig. 8.15..... 68
Table 5.7: Delay deviations for AOI_21 and AOI_32 logic gates 70
Table 6.1: Combinations of *min* and *max* values considered for the threshold voltages (coded variables) of devices in a 3-transistors network. 73
Table 7.1: Approximate resistance equations for transistors in a parallel network..... 84
Table 7.2: Approximate resistance equations for NMOS transistors in a series network. 85
Table 7.3: Approximate resistance equations for PMOS transistors in a series network.. 86
Table 7.4: Delay deviations for NMOS transistors in a series network, according to the position of the switching transistor in relation to the output node (1-close...4-far) 87
Table 7.5: Delay deviations for PMOS transistors in a series network, according to the position of the switching transistor in relation to the output node (1-close...4-far) 88
Table 7.6: Delay deviations for NMOS transistors in parallel networks, according to the number of switching transistors..... 89
Table 7.7: Delay deviations for PMOS transistors in parallel networks, according to the number of switching transistors..... 90
Table 7.8: Delay deviation for N- and PMOS transistors in series networks, according to the position of the switching transistor in relation to the output node (1-close...4-far) for the linear and the saturation region of operation 91
Table 7.9: Delay deviation for N- and PMOS transistors in parallel networks, according to the number of switching transistor for the linear and the saturation region of operation. 92
Table 7.10: Rise and fall delay deviations for different inverter topologies. 92
Table 7.11: Rise and fall delay deviations for different inverter topologies with different threshold voltage variations..... 93
Table 7.12: Rise and fall delay deviations for different sizes of inverters 94

Table 7.13: Rise and fall delay deviations for different sizes of inverters with different threshold voltage variations 94
Table 7.14: Rise and fall delay deviations for different chains of inverters 98
Table 8.1: Delay deviations for different implementations of a 2-input XOR provided by statistical simulation and by the proposed method 100
Table 8.2: Delay values and deviations for different implementations of a 2-input XOR 101
Table 8.3: Delay deviations for different implementations of a 4-input XOR provided by statistical simulation and by the proposed method 106
Table 8.4: Delay values and deviations for different implementations of a 4-input XOR 107
Table 8.5: Truth table of a full adder 112
Table 8.6: Delay deviations for different implementations of a full-adder provided by statistical simulation and the proposed method 113
Table 8.7: "SUM" delay deviations for different implementations of a 1-bit full-adder 114
Table 8.8: Delay deviations for a complex gate provided by statistical simulation and by the proposed method 116
Table 8.9: Runtime analysis for different implementations of a XOR2 118
Table 8.10: Delay and runtime analysis of the complex gate topology by using different methods 119
Table 8.11: Delay and runtime analysis of a XOR4 implemented with a DCVSL topology by using different methods 119

RESUMO

No regime em nanoescala da tecnologia VLSI, o desempenho dos circuitos é cada vez mais afetado pelos fenômenos de variabilidade, tais como variações de parâmetros de processo, ruído da fonte de alimentação, ruído de acoplamento e mudanças de temperatura, entre outros. Variações de fabricação podem levar a diferenças significativas entre circuitos integrados concebidos e fabricados. Devido à diminuição das dimensões dos componentes, o impacto das variações de dimensão crítica tende a aumentar a cada nova tecnologia, uma vez que as tolerâncias de processo não sofrem escalonamento na mesma proporção. Muitos estudos sobre a forma como a variabilidade intrínseca dos processos físicos afeta a funcionalidade e confiabilidade dos circuitos têm sido realizados nos últimos anos. Uma vez que as variações de processo se tornam um problema mais significativo devido à agressiva redução da tecnologia, uma mudança da análise determinística para a análise estatística de projetos de circuitos pode reduzir o conservadorismo e o risco que está presente ao se aplicar a técnica tradicional.

O objetivo deste trabalho é propor um método capaz de predizer a variabilidade no atraso de redes de transistores e portas lógicas sem a necessidade da realização de simulações estatísticas consideradas caras em termos computacionais. Este método utiliza o modelo de atraso de Elmore e a técnica de *Asymptotic Waveform Evaluation* (AWE), considerando as resistências dos transistores obtidas em função das variações das tensões de limiar dos transistores no arranjo. Uma pré-caracterização foi realizada em algumas portas lógicas de acordo com a variabilidade de seu desempenho causados por variações da tensão de limiar dos transistores a partir de simulações Monte Carlo. Uma vez que existem vários tipos de arranjos de redes de transistores e esses arranjos apresentam um comportamento diferente em termos de atraso, consumo de energia, área e variabilidade dessas métricas, torna-se muito útil identificar os circuitos nos quais as redes de transistores são menos influenciadas pelas variações em seus parâmetros. O modelamento da variabilidade do atraso é feita através de 2^K simulações DC para a rede "pull-up", 2^N simulações DC para a rede "pull-down" (K e N são os números de transistores de cada rede) e uma simulação transiente para cada porta lógica, o que leva apenas alguns segundos no total. O objetivo de toda a análise é fornecer orientações para a geração de redes lógica ótimas que oferecem baixa sensibilidade às variações de seus parâmetros.

Palavras-Chave: Tecnologia CMOS, variabilidade, atraso, redes de transistores.

ABSTRACT

In the nanoscale regime of VLSI technology, circuit performance is increasingly affected by variational effects such as process variations, power supply noise, coupling noise and temperature changes. Manufacturing variations may lead to significant discrepancies between designed and fabricated integrated circuits. Due to the shrinking of design dimensions, the relative impact of critical dimension variations tends to increase with each new technology generation, since the process tolerances do not scale in the same proportion. Many studies on how the intrinsic variability of physical processes affect the functionality and reliability of the circuits have been done in recent years. Since the process variations become a more significant problem because of the aggressive technology scaling, a shift from deterministic to statistical analysis for circuit designs may reduce the conservatism and risk that is present while applying the traditional technique.

The purpose of the work is to propose a method that accounts for the deviation in the performance of transistors networks and logic gates without the need of performing computationally costly simulations. The estimation method developed uses the Elmore Delay model and the Asymptotic Waveform Evaluation (AWE), by considering the resistances of transistors obtained as functions of threshold voltages variations of the transistors in the arrangement. A pre-characterization was performed in some logic gates according to their performance variability caused by variations in the threshold voltage of the transistors by running Monte Carlo simulations. Since there are several kinds of transistor networks arrangements and they present different behavior in terms of delay, power consumption, area and variability of these metrics, it is very useful to identify circuits with such arrangements of transistors that are less influenced by variations in their parameters. The delay variability modeling relies on (2^K) DC simulations for the pull-up network, (2^N) DC simulations for the pull-down network (K and N are the number of transistors in the pull-up and pull-down network, respectively) and on a single transient simulation for each gate, which take only a few seconds altogether. The goal of the whole analysis is to provide guidelines for the generation of optimal logic networks that present low sensitivity to variations in their parameters.

Keywords: MOS transistor, performance variability, transistor network.

1 INTRODUCTON

The electronics industry transitioned from the era of discrete components to the era of the integrated circuit (IC) in order to achieve higher quality and reliability through less components to handle and assemble, and higher yields through smaller miniature devices (CHIANG, 2007). The process of yield improvement focused on reducing the number of impurities and imperfections which resulted in random effects and lower yields, and on shrinking the design features that resulted in smaller die areas and thus more dies per wafer.

Any manufacturing process presents a certain degree of variability around the nominal value of the product specifications. This phenomenum is accounted for in the description of the product characteristics in the form of a window of acceptable variation for each of its critical parameters. The aggressive shrinking of MOS technology causes the intrinsic variability present in its fabrication to increase. Process tolerances do not scale proportionally with the design dimensions, causing the relative impact of the critical dimension variations to increase with each new technology generation (ARGAWAL, 2007). This scenario demands realistic approaches that are able to predict the impact of parameters variations in the metrics of the circuit.

Since the process variations become a more critical issue, the migration from deterministic to statistical analysis of circuit designs may reduce the conservatism of applying the traditional worst-case approach. The traditional corner-case technique seems as a reasonable way to handle global variations but not local ones. In the case of performance of a circuit, a logic gate may become slower for a certain variation and faster for another, and that might depend on its location on a die. Not only the importance of the intra-die variations has grown but also the number of process parameters that present considerable variations has also increased (SRIVASTAVA, 2005). Such situation requires some changes in the traditional techniques in order to find a better alternative to their deterministic nature.

Increasing levels of processes variations have a major impact on power consumption and performance of a design, resulting in parametric yield loss. This problem is potentialized by the fact that timing has an inverse correlation with leakage power, e.g. a reduction in channel length results in improved performance but also causes an exponential increase in leakage power.

The manufacturing process has been based on optical lithography. The wavelength in recent and future technologies is bigger than the minimum physical drawn dimensions (193 nm), so variability due to lithography has been of increasingly importance. Also, classical modeling of the behavior of dopant materials reached the end of its validity and quantum mechanical behavior of material needs to be taken into account. Deterministic corner-based methodologies were replaced by statistical analysis

in many situations. Furthermore, the random component of variability which was strictly global in nature has a new intra-die component of considerable significance.

Different transistors networks present different electrical characteristics, even if they represent the same logic function (ROSA, 2008). In this sense, the goal of this work is to propose a delay method that takes into account the variability in the threshold voltage of the transistors in a specific network. The first step was to analyze how the variability in the parameters affects the metrics of the gates according to (i) their topology (quantity of series and parallel transistors) and (ii) the position of the transistor with a transient input signal in relation to the output node. Electrical simulations of the gates were performed with HSPICE, a standard circuit simulator that is considered the "golden reference" for electric circuits. An estimation method for the delay variability was proposed and evaluated. We attempted to develop an as-simple-as-possible variability aware timing method for different logic gates (inverter, NAND, NOR, XOR and full adder) and different arrangements of transistors. The transistors were approximated as switches controlled by the input signal applied to the gate terminal and were described by their intrinsic capacitances and their resistances while on conduction (charging or discharging the load capacitance). The resistances and capacitances are functions of the parameters of the transistors. The method includes the use of the Elmore Delay model (ELMORE, 1948) and the Asymptotic Waveform Evaluation (AWE) (PILLAGE, 1990), by considering the resistances of transistors obtained as functions of threshold voltage variations of transistors in the arrangement. The data collected through eletrical simulation is compared to the results generated by the proposed method. The results of the analysis may be used to achieve circuit implementations that present some immunity to variability.

A brief introduction to the context and subject of the work is given in CHAPTER 1. It is followed by a general review on digital circuits, including logic functions and gates topologies, besides the design flow for ASICs as well as a study on the types and sources of variability, in CHAPTER 2. CHAPTER 3 is dedicated to deterministic and statistical timing analysis concepts, and also to some timing models proposed in the literature for transistors and logic gates, along with the basic concepts of the timing analysis that can make use of those models. The proposal of the work and methodology used to develope it are explained in CHAPTER 4. Preliminary results extracted by simulations using the electrical simulator HSPICE are presented and analyzed in CHAPTER 5, including a more complete characterization of an inverter according to the variability on its parametes. The delay variability estimation method proposed in this work in order to predict the variability of a cell delay under variations in its parameters is described in CHAPTER 6. CHAPTER 7 presents the results achieved by using the proposed method to calculate the delay variability for different transistors networks. CHAPTER 8 shows how reliable is the method when applied to calculate the delay variability of different topologies that are used to implement the same logic function. CHAPTER 9 concludes the work.

2 DIGITAL DESIGN AND PARAMETER VARIABILITY IN INTEGRATED CIRCUITS

The rapid scaling of CMOS technology has turned the control of critical device parameters into a very difficult task, and drastic variations of process and design parameters have resulted. Variations in the physical, operational or modeling parameters of a device lead to variations in electrical device characteristics, such as the threshold voltage, drive strengh of transistors, and in the resistance and capacitance of interconnects. Finally, variations in electrical characteristics of the components lead to variations in the performance and power consumption of the circuit.

The first section of this chapter presents some concepts about IC design and different logic families, since the main goal of the work is the modeling and analysis of how the delay variability is influenced by the use of different topologies of logic gates.

The second section briefly describes some of the major fabrication steps in MOS process flow and presents an overview of IC parameter variability. It also presents some of the most important sources of variations, as well as their effects.

That section is also intended to validate the use of non-correlated variations for the threshold voltages (V_{TH}) of MOS transistors, what means that each transistor has a random V_{TH} variation, independently on its position on the die. The impact of channel length (L) and oxide thickness (T_{ox}) variations on threshold voltage were compared to the effect caused on the same electrical parameter by random-dopant fluctuations (RDF).

2.1 Digital Integrated Circuit Design

In the beginning of the last century, electronics circuits used large, expensive and power consuming vacuum tubes. In 1947, the first functioning point contact transistor was built and soon after that the bipolar junction transistor was developed (WESTE, 2005). Early integrated circuits (IC) primarily used bipolar transistors, but the large majority of the current IC's are implemented in the Metal Oxide Semiconductor (MOS) technology. An integrated circuit is an electronic system consisting of a number of miniaturized electronic devices, such as transistors, resistors, capacitors and inductors, fabricated in a monolithic semiconductor substrate (RABAEY, 2005).

The level of integration of chips has been classified as small-scale, medium-scale, large-scale and very-large-scale (VLSI). The million-transistor-per-chip barrier was crossed in the late 1980s and the handcrafted design once implemented has become inappropriate (RABAEY, 2005). Instead of the individualized approach of the earlier designs, a circuit is constructed in a hierarchical way: a system, such as a processor, is a

collection of modules, which are collections of gates that consists in a certain number of transistors.

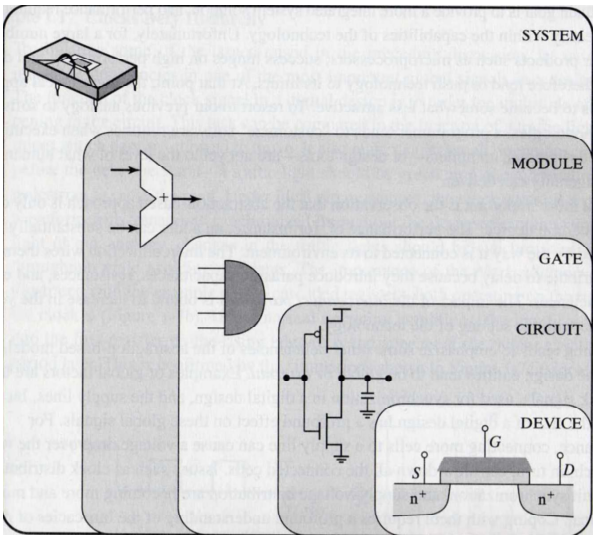


Figure 2.1: Design abstraction levels in digital circuits (RABAEY, 2005).

An integrated circuit can be described in terms of three domains: (i) the *behavioral* domain, (ii) the *structural* domain and (iii) the *physical* domain. The behavioral domain specifies how the system works, its function. The structural domain specifies the interconnection of components required to achieve the behavior that is desired. The physical domain specifies how to arrange the components in order to connect them, which in turn allows the required behavior (WESTE, 2005). **Digital IC design** is used to produce components such as microprocessors, FPGAs (Field-Programmable Gate-Arrays), memories and digital ASICs (Application-Specific Integrated Circuits).

The advance of the technology for constructing ICs demanded the elaboration of design tools able to reduce the complexity, increase productivity and assure the designer a working product. Over the last four decades, **Computer Aided Design** (CAD) tools have gradually been adopted for various tasks of the design process. The adoption of tools has influenced how circuits are designed in various ways. In some cases certain restrictions on what is an acceptable design have to be imposed, so that some tools may be used. In other cases, new tools allow designs to try on several design alternatives, which was not reasonable before since the design was done manually. Design tools include simulation at the various complexity levels, design verification, layout generation and design synthesis.

The viability of an IC design depends on its application and on economic considerations. There is a number of distinct implementation approaches ranging from high-performance, handcrafted to fully programmable, medium-to-low performance designs (RABAEY, 2005). Design is based on a tradeoff to achieve adequate results for performance (speed, power consumption, function, flexibility, robustness), size, time to design, ease of verification, test generation and testability.

When high-performance is desired, handcrafting the circuit topology and physical design seems to be a reasonable option. In **full-custom** design the logic and physical synthesis attain usually the highest performance and smallest size, making use of the most advanced technologies. It is the most technology dependent design approach, since

each switch element present in every cell is manually fine-tuned in order to explore all the performance advantages that a given technology can deliver. The disadvantages of full-custom can include increased manufacturing and design time, and much higher skill requirements on the part of the design team.

In order to avoid the redesign and reverification of frequently used cells, such as basic gates, arithmetic and memory modules, designers most often resort to cell libraries. Cell-based design uses a **standard cell** library as the basic building blocks of a chip. These libraries not only contain the layouts, but also provide complete documentation and characterization of the behavior of the cells. **Cell library** is a finite set of logic cells that implements different Boolean functions with different drive strengths and topologies.

Traditionally, the technology mapping methods rely on static pre-characterized libraries aiming delay, area and power optimizations. Each cell in the library is fully characterized through many simulations, resulting in a set of accurate information about the behavior of the cell. Usually, standard cells have a fixed height with power and ground routed respectively at the top and bottom of the cells. As compared to full-custom design, cell-based design offers much higher productivity since the predesigned cells may be reused many times. The disadvantage is that the constrained nature of the library, especially due to the limited number of cells, reduces the possibility of fine-tuning the design (RABAEY, 2005). The variability on the metrics of a logic cell can also be used as a constraint for the behavior of the cell in a technology mapping.

2.1.1 Logic Styles

Two classes of logic circuits can be identified: a) combinational circuits and b) sequential circuits. Combinational logic circuits present an output that is related to its current input signals by some Boolean expression at any point in time. Sequential logic circuits have their outputs as functions of the current input data and also of previous values of the input signals. A sequential circuit includes a combinational logic portion and a module that holds the state.

Several circuit styles can be used to implement a given logic function. Logic styles are basically classified as being dynamic or static topologies. A static topology connects each gate output to either V_{DD} or GND at every point in time but during the switching transient (RABAEY, 2005). A dynamic topology relies on the temporary storage of signal values on the capacitance of high-impedance circuit nodes.

2.1.2 Logic Gates with Pull-up and Pull-down Networks

Static Complementary MOS is the most widely used logic style, because it presents some important characteristics: low sensitivity to noise (robustness), good performance, low power consumption, availability in standard cell libraries, among others. A static CMOS gate is composed of a pull-up and a pull-down network, also referred to as PUN and PDN, respectively. A PUN connects the output of the circuit to V_{DD} and a PDN connects the output to ground (Fig. 2.2).

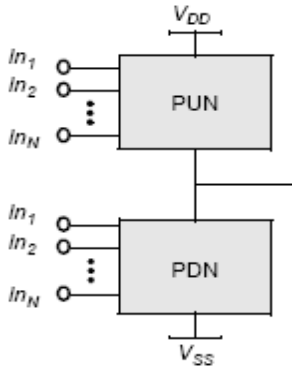


Figure 2.2 Complementary logic gate as a combination of a pull-up and a pull-down network (RABAEY, 2005).

The CMOS structure is naturally an "inverting" gate that is able to implement functions such as NAND and NOR. If a non-inverting Boolean function is desired, it requires an inverter stage. By using inverters, NANDs and NORs it is possible to have any logic equation (function) implemented.

A major problem in using the CMOS style is the large number of transistors that is required to implement a logic function (2N transistors to implement an N-input logic gate). Another characteristic to be considered is the significant load capacitance since each gate drives two devices (a PMOS and an NMOS) per *fan-out* (number of digital inputs that the output of a gate can feed). However, a reduction in the number of transistors can be achieved by using gates that implement more complex functions. These gates are called "Static CMOS Complex Gates" and are obtained by an association of series/parallel transistors.

A minimum number of stacking (series) transistors for the CMOS gate may be achieved by deriving the pull-up and pull-down planes from the best choice of individual networks, not necessarily representing complemented topologies. (ROSA, 2008) shows an algorithm for generating minimum transistor chain networks by equations that represent a given logic function.

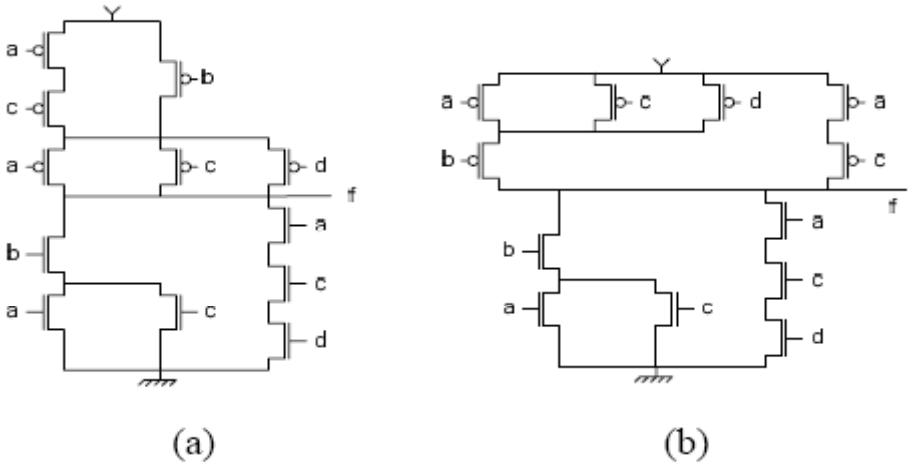


Figure 2.3 Different implementations of a logic function (ROSA, 2008).

2.1.3 Cascode Voltage Switch Logic

Cascode voltage switch logic (CVSL) uses both true and complementary input signals and computes both true and complementary outputs using a pair of NMOS pull-down networks (Fig. 2.4). For any given input pattern, one of the pull-down networks will be ON and the other OFF. The pull-down network that is ON will pull that output low. This low output turns ON the PMOS to pull the opposite output high. When the opposite output rises, the other PMOS turns OFF so no static power dissipation occurs (WESTE, 2005). CVSL increases the speed of the circuit because all of the logic is performed with NMOS transistors, thus reducing the input capacitance.

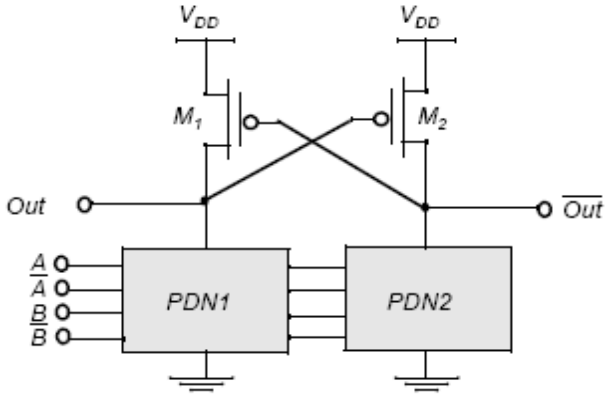


Figure 2.4 Basic principle of the Cascode Voltage Switch Logic style (RABAEY, 2005).

2.1.4 Pass-Transistor Logic

In *pass-transistor logic* (PTL), inputs of the circuit are also applied to the source/drain diffusion terminals in an attempt to reduce the number of transistors required to implement the logic (RABAEY, 2005). These circuits use either NMOS pass transistors or parallel NMOS and PMOS transistors (transmission gates) as switches (Fig. 2.5).

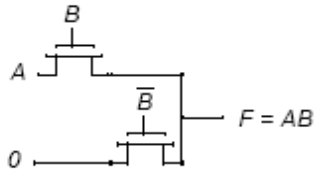


Figure 2.5 Pass-transistor implementation of an AND gate (RABAEY, 2005).

2.1.5 Logic Functions and Transistor Networks

The gates of a circuit are basically constituted by transistors connected in order to perform a logic function. A transistor can be approached by a switch controlled by its gate signal. An NMOS transistor is ON when the controlling signal is high and OFF when the controlling signal is low. A PMOS transistor acts in the opposite way, being ON when the signal is low and OFF when the signal is high.

A transistor network is a set of interconnected devices acting as switches in order to implement arbitrary Boolean functions. Different Boolean equations can represent the same Boolean function. The concern of a "logic synthesis" is to figure out the best equation (s) for a given logic function (s). The optimization criteria in designing a

circuit are related to the implementation of the equation in logic gates and can be aimed at minimizing some characteristics of the circuit, such as area, power consumption or propagation delay.

A circuit may have its characteristics improved by optimizing its transistor networks. Some important characteristics of a circuit are area, propagation delay, power consumption, noise margin and sensitivity to device variations. Network optimizations may be achieved by reorganizing the switch arrangement and placing the switches according to some rules to minimize a specified cost. In the case of trying to reduce propagation delay in a circuit, it may happen that some signals in combinational logic blocks are more critical than others and not all inputs of a gate arrive at the same time. Placing the critical-path transistors closer to the output of the gate can result in a speed-up. Also, manipulating the logic equations can reduce the fan-in requirements and hence decrease the gate delay.

The main concern of this work is on achieving circuit implementations with characteristics that present high immunity to variations in the parameters of the devices, and are able to provide low performance variability.

2.1.6 Standard Cell Design Flow

A **design flow** is a systematic set of procedures that makes it possible to implement a chip according to its specifications in an error-free way. Design of digital ASICs starts at the behavioral level and then proceeds to the structural level (gates and registers), which is called *Register Transfer Level* (RTL), by using a hardware description language (HDL). **Logic Synthesis** tools translate modules described in an HDL language into a *netlist*, what is a description of the standard cells to be used plus the needed electrical connections between them. As part of the logic synthesis step, **technology mapping** is the procedure of expressing a given Boolean network in terms of logic cells or gates. Typically, the objective function aims at the optimal use of all gates in the library to implement a circuit with critical-path delay less than a target value and minimum area. The most common techniques for technology mapping are based on pre-characterized cell libraries. These techniques are also known as **library-based flow**.

In traditional technology mapping, the costs are determined by the worst-case or alternatively mean values of the costs. When the power and delay cost metrics are impacted by variability the algorithms developed so far fail to find an optimal solution, since technology mapping for parametric yield demands probabilistic formulation of the problem. (SINGH, 2005) claims to be the author of the first work that rigorously treats variability in circuit leakage power and delay within logic synthesis. **Variability-aware technology mapping** might be an application of the work to be presented along the next sections.

A **placement** tool processes the gate-level netlist and places the standard cells onto a region representing the final ASIC. The **routing** tool creates the electrical connections between the cells. A circuit **parasitic extractor** generates a model of the chip from the physical layout, including devices sizes, capacitances and resistances of the wires. A **post-layout simulation** and **verification** step (STA and power estimation) verifies the functionality and performance of the chip in the presence of the parasitics.

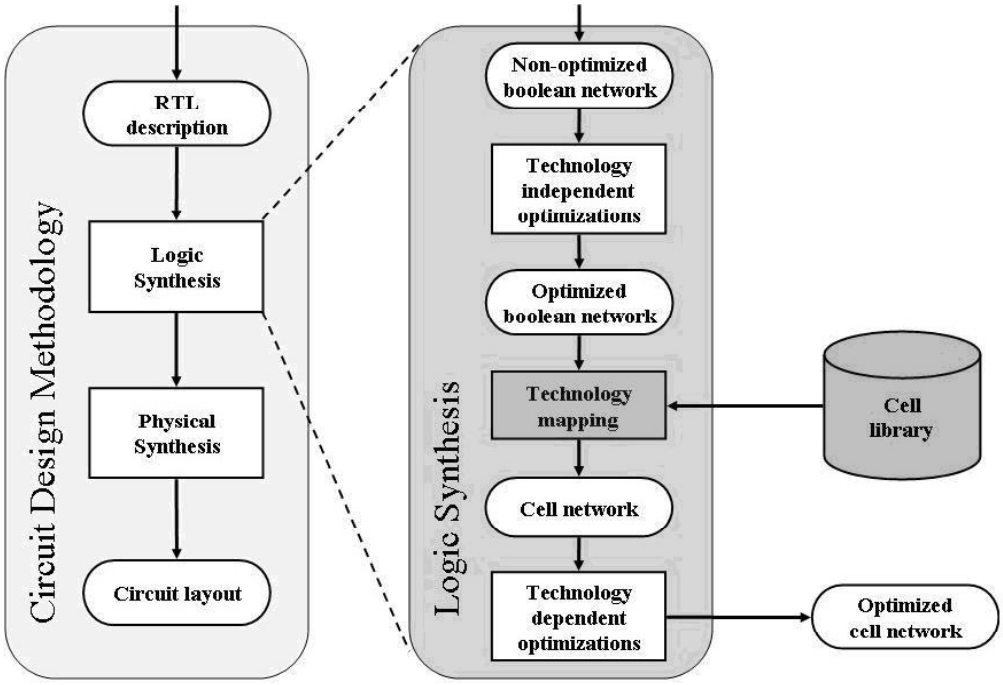


Figure 2.6: Digital circuit design methodology using predefined cell library (ROSA, 2008).

Some approaches for technology mapping propose techniques based on automatic cell generators. These approaches are known as **library-free** (REIS, 1998). Instead of having a predefined static library, they assume that arbitrary cells can be generated on-the-fly through a cell generator, increasing the matching search space. The mapping algorithm defines the set of cells required in the circuit implementation, and this **virtual library** is used as input for a cell generator which provides the logic cell layouts that are further used in the physical synthesis.

2.2 Variability in Integrated Circuits

One way of classifying the variations in a circuit is according to their nature: (i) process, (ii) environmental and (iii) modeling variations (SRIVASTAVA, 2005).

2.2.1 Sources of Process Variations

Process variations are discrepancies in the value of the process parameters observed after fabrication. Process tolerances do not scale proportionally with the design dimensions, what increases the relative impact of the critical dimension variations with each new technology generation (ARGAWAL, 2007).

2.2.1.1 Photolithography

Photolithography constitutes the steps required to transfer a pattern from a mask to the surface of the silicon wafer (JAEGER, 2002). In recent and future technologies, the wavelength employed by the lithography machinery (nowadays 193 nm) is bigger than the dimensions of some patterns produced with it, e.g. the channel length of the transistors. The Rayleigh factor k_1 (process dependent adjustment factor) quantifies the

printability problems by measuring how close a given process is to the resolution limit (ORSHANSKY, 2008):

$$k_1 = CD * (\frac{NA}{\lambda}) \tag{2.1}$$

where *CD* is the critical dimension of the process, λ is the wavelength of the exposure system, and *NA* is the numerical aperture. *NA* is a function of the lens and of the refraction index of the medium between the wafer and the lens of the contact aligner. A low process dependent adjustment factor (Optical Proximity Effect) results in linewidth variation, corner rounding and line-end shortening. Optical Proximity Effect is a distortion due to the diffraction of the light used in lithography, whose wavelength is bigger than the dimensions of the features to be printed. It is related to the dependence of the printed critical dimension (*CD*) on its surrounding. Line shortening refers to the reduction in the length of a rectangular feature. Corner rounding is a type of image distortion that produces a smoothed out pattern.

2.2.1.2 Etch

Chemical etching in liquid or gaseous form is used to remove any material that is not protected by hardened photoresist. Etching non-uniformity manifests as variability of etching bias, which is the difference between the photoresist and etched polysilicon critical dimensions (ORSHANSKY, 2008). The most important component of this variation is a function of layout pattern density, and can be classified in (i) microloading, (ii) macroloading or (iii) aspect-to-ratio dependence.

Micro- and macroloading are related to variation in the layout features, which can increase or decrease the density of the reactant. In microloading, the etching bias depends on the local environment while in macroloading it is determined by the average loading across the wafer. In aspect-ratio-dependent etching, the variation of linewidth is dependent on the distance of nearby features.

2.2.1.3 Line Edge Roughness (LER)

Line Edge Roughness is the local variation of the edge of the polysilicon gate along its width. Some causes of LER are the random variation in the incident photons during exposure, as well as the absorption rate, chemical reactivity and the molecular composition of the resist.

2.2.1.4 Chemical Mechanical Polishing (CMP)

Chemical Mechanical Polishing is used to remove copper and barrier metal sitting outside the trench area. CMP results in deviations of the dimensions due to *dishing* and *erosion*. A wafer is polished in order to remove all excess copper. As copper etches faster than the surrounding dieletric, there is a difference between the final oxide level and the lowest point in the copper wire, a phenomena that is referred to as dishing (ORSHANSKY, 2008). Erosion is the loss in thickness of the surrounding dieletric compared to a cleared surface. The oxide between wires in a dense array tends to be over-polished compared to nearby areas of wider insulators.

2.2.1.5 Random Dopant Fluctuations (RDF)

The fluctuation of random dopants derives mainly from the random nature of ion implantation. In MOSFET transistors, this "atomistic variability" in the channel region can alter the transistor properties, especially threshold voltage. In recent technologies it is a very important issue, since the number of dopant atoms is scaling down with device channel length and it is difficult to control the doping profiles. RDF causes local variations, what means that devices placed close to each other may have different distribution and quantity of implanted ions on their channels.

2.2.2 Variability of Process Parameters

The parameters of transistors that are more susceptible to variations are the number and distribution of dopant atoms, effective transistor gate length, gate width and gate oxide thickness. In the case of interconnects, metal line width and thickness are the parameters that suffer of variations mostly. A brief discussion on the variability of each parameter is presented in the next sub-sections.

2.2.2.1 Variability of Gate Length

The gate length of the MOS transistor is known as "critical dimension" because it defines the minimum feature size of a technology. Its characteristics strongly impact the current drive strength and the speed of the gate. Several fabrication steps influence the *effective channel length* (gate length minus the under-diffusions of the source and drain regions), including the mask, the exposure system, etching, the spacer definition and implantation of source and drain regions (ORSHANSKY, 2008). The channel length variations are mainly caused by lithography induced errors and line edge roughness (LER).

2.2.2.2 Variability of Thin Film Thickness

Silicon dioxide is traditionally used as the dielectric film that isolates the gate from the silicon channel and has great influence on the electrical properties of the transistors. Gate oxide thickness is a function of the temperature and atmosphere of the environment surrounding the wafer on which this oxide is being grown. A change in one or more of these process conditions will certainly affect the thickness of the material (ORSHANSKY, 2008). Also, it is a function of interface roughness, which represents a random variation.

2.2.2.3 Variability of Interconnect and Dieletric

Dishing and erosion effects can result in substantial variation in the thickness of patterned copper features, leading to deviations in metal line resistance. Also, variations in the thickness of dieletric layers can appear due to limitations in Chemical Mechanical Polishing, deposition and plating processes (ORSHANSKY, 2008). Surface topography is a very critical variable in determining metal and dieletric thickness. All these factors play an important role in the variability in the resistance (R) and capacitance (C) of the lines. R and C variations are strongly spatially and cross-correlated. A wider metal will increase C but will reduce R, and no abrupt changes are found in R and C along an individual wire.

2.2.3 Variability of Device Characteristics

The **threshold voltage** is a device parameter determined by the material that implements the gate, the thickness of the silicon dioxide and the concentration and the density profile of the dopant atoms in the channel of the transistor, among other process characteristics. It is mainly affected by the variation in the number and distribution of dopant atoms along the material and variations in oxide thickness. As devices shrink, the number of dopant atoms per transistor may be less than a hundred, what decreases the level of control in the number and uniformity of these atoms along the channel. At this scale, a single dopant atom may change device characteristics, resulting in large variations from device to device (ORSHANSKY, 2008).

Although the thermal oxidation process has been well controlled, some problems arise from the fact that the thickness of the oxide layer has reached the atomic scale of the oxide-silicon interface layer. The interface roughness and the atomic scale discreteness present limitations that make this control increasingly difficult. That leads to variations in the device characteristics such as **mobility** and **threshold voltage**.

2.2.4 Physical Variability Due to Aging and Wearout

The physical parameters of the devices may be affected by time-dependent phenomena that cause variations over time. Some mechanisms of temporal variability are (i) negative-bias temperature instability, (ii) hot carrier effects, and (iii) electromigration (ORSHANSKY, 2008).

Negative-bias temperature instability (NBTI) causes the threshold voltage of the PMOS to increase, reducing its current drive capability and thus affecting the circuit performance. **Hot carrier effect** affects primarily n-channel MOSFETs and it is due to the injection of additional electrons into the gate oxide near the interface with silicon. This leads to the increase of the threshold voltage, lower current drive and also compromises the performance of the device. **Electromigration** is caused by the impact of high current densities on the atomic structure of the wire. This may lead to shorts between wires or to the creation of an open failure in the wire.

2.2.5 Environmental Variations

The environmental variations are related to the surrounding environment of the chip during its operation and may include temperature variations, fluctuations in the power supply and noise coupling among nets (NASSIF, 2000). Environmental variability is largely systematic since it depends predominantly on the details of circuit operation.

2.2.6 Variations Due to Modeling

The delay and power models used to perform design analysis and optimization do not truly describe the device characteristics, resulting in different values for the variables of the circuit from the actual ones (SRIVASTAVA, 2005). Conservative models can make it hard to meet design specifications and aggressive models result in yield loss, so a trade-off is necessary.

Environmental and modeling uncertainties are typically modeled using worst-case margins, whereas process variations are generally treated statistically.

2.2.7 Typical Values for the Parameters Variations

Table 2.1 presents the 3σ variation points for the technology parameters of a typical 0.18 μm CMOS process: (i) the width of the interconnect wire (W), (ii) gate oxide thickness (T_{OX}), (iii) channel length (L), (iv) temperature (T), (v) supply voltage (V_{DD}), and (vi) the threshold voltage (V_{TH}).

Table 2.1: 3σ variation points for the technology parameters of a typical 0.18 μm CMOS process (HASSAN, 2005).

Parameter	*3σ Variation (%)*
W	3
T_{ox}	1.2
L	5
T	10
V_{DD}	10
V_{TH}	5

2.2.8 Spatial Scales of Variations

According to the spatial scales, the types of deviations that affect the characteristics of a circuit can be divided in global or local variations (BERKELAAR, 1997).

2.2.8.1 Global Variations

Deviations that are **global** appear in all the elements of a circuit in a similar way, such as changes in the power supply voltage, global temperatures changes on the chip and variations during the production process. They are also called **inter-die** variations and designate a parameter variation that presents the same value for all devices in a single die, but different values for different dies, wafers or lots (SRIVASTAVA, 2005). These variations are represented by a shift in the mean value of the respective parameter distribution from the nominal value. In this case, variations in a single process parameter can be analyzed by corner models. However, if we are dealing with more process parameters simultaneously it is important to analyze the correlation between them, what would make the corner method prohibitive because of the exponential growth in the number of corners to be simulated.

2.2.8.2 Local Variations

Local deviations are caused by the state of the gate, different input signal rise (or fall) times, local supply noise, local temperature changes, crosstalk on wires, local manufacturing imperfections. They are also called **intra-die** variations and affect the device parameters within a single die in a different way, setting different values for the variations depending on the location of the respective device. They may be spatially correlated or uncorrelated (independent or random component), depending on the source of variations (SRIVASTAVA, 2005).

2.2.8.3 Spatially Correlated Variations

Intra-die variations often exhibit spatial correlations, where devices that are close to each other have a higher probability of being alike than devices that are placed far apart.

2.2.8.4 Independent Variations

The random variations are unpredictable in nature and can happen in the device length, discrete doping fluctuations and oxide thickness variations. The random component presents no correlation across devices.

2.2.9 Parametric Yield

Variability in the characteristics of integrated circuits can lead to significant discrepancies between designed and manufactured products. The different types of variations presented affect the performance and power consumption of devices and interconnects. The analysis of manufactured circuits leads to the definition of **parametric yield**, which is the percentage of chips that meet the performance and power consumption constraints in the presence of variations. **Parametric yield loss** refers to the yield loss resulting from the device's parameters that do not meet the specifications and result in unacceptable functional behavior of the circuit (ORSHANSKY, 2008).

In current designs, the yield of a lot is typically calculated by characterizing the chips according to their operational frequency. However it is observed that a considerable fraction of the fast chips dissipate very large amounts of leakage power and thus are not adequate for commercial usage. Figure 2.7 shows measurements of a chip which indicates that the scattering in the leakage current can be up to 20x, while in the clock frequency can be up to 30%.

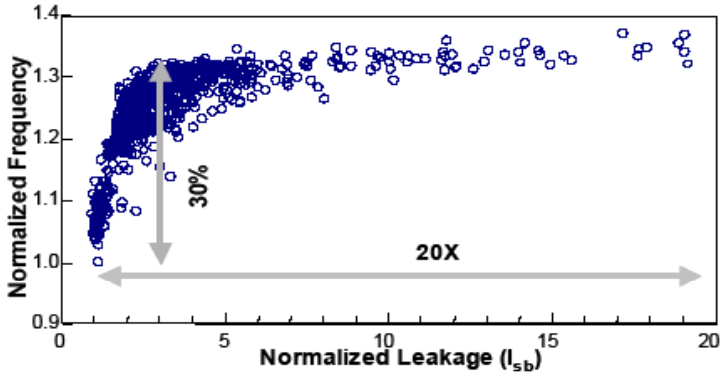


Figure 2.7: Leakage current and frequency measured in a sample of 1000 chips (Source: Intel).

(RAO, 2004) developed a stochastic model for leakage current that includes the effects from multiple sources of variability and captures the dependence of the leakage current distribution on operating frequency. It derives a closed-form expression for the total leakage as a function of all relevant process parameters and also presents an analytical equation to quantify the yield loss when a power limit is imposed.

The exponential dependence of leakage power on two highly-variable parameters (gate channel length and threshold voltage) causes a large spread in leakage current in the presence of variations (SINGH, 2005). Process variation degrades parametric yield not only by impacting power consumption and performance of a design, but also by

doing that in an inverse way. This negative correlation makes many manufactured chips meet timing specifications, but not the power constraint (or vice-versa) (ARGAWAL, 2007). As an example, by increasing the supply voltage of the devices on a chip or by reducing the transistors threshold voltage it is possible to get higher drive currents, what makes the device faster. However, it also increases power consumption because of the higher leakage current in the device.

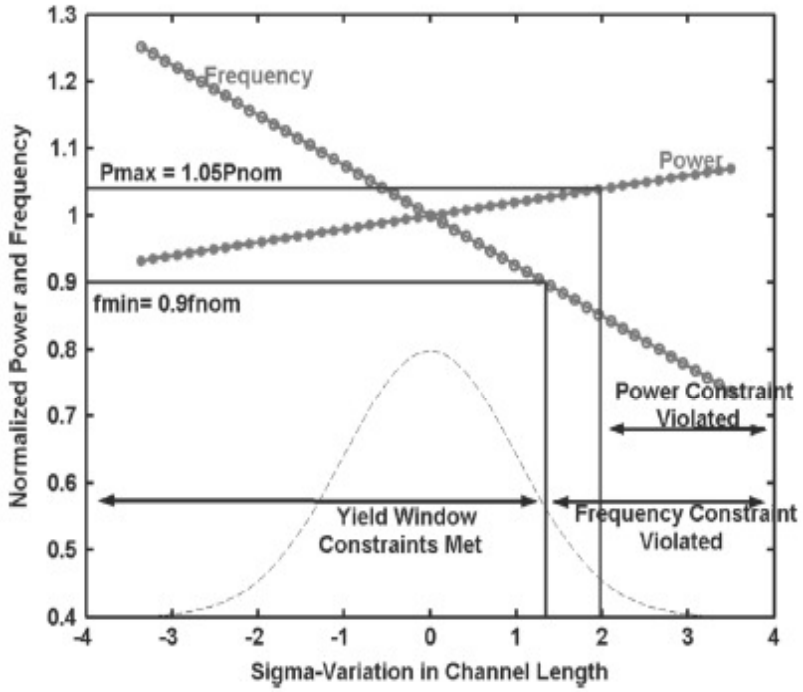


Figure 2.8: Yield window for frequency constraint of fmin = 0.9 fnom and power constraint of Pmax = 1.05 Pnom, for negligible static (leakage) power (ARGAWAL, 2007).

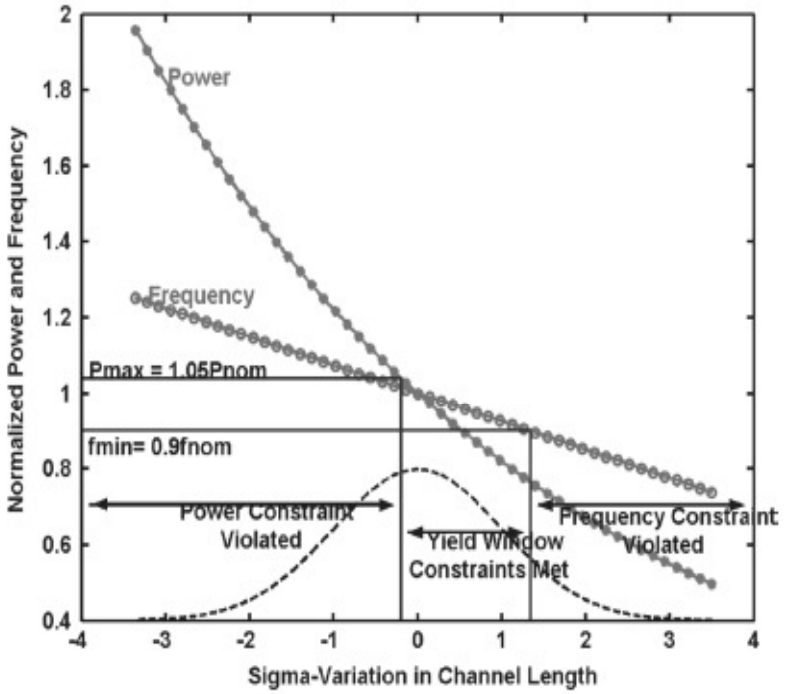


Figure 2.9: Yield window for frequency constraint of fmin = 0.9 fnom and power constraint of Pmax = 1.05 Pnom, with static (leakage) power (ARGAWAL, 2007).

In Figure 2.8 it is shown a one-side-constrained feasible region of operation by not considering the leakage power of the devices. However, if the leakage power is taken into account (Figure 2.9), we have a two-side-constrained region, what represents a significant loss in parametric yield (SRIVASTAVA, 2005). The data are plotted for power and frequency of operation in the presence of variations in the channel length of the devices.

Parametric yield optimization can not be achieved by static timing analysis because the use of corners approaches to verify timing becomes prohibitive with the large number of sources of variations that must be considered, and also the correlation of the parameters.

2.2.10 Considerations

The traditional corner-case technique seems a reasonable way to handle global variations but not the local ones. In the case of the performance of a circuit, a logic gate may become slower for a certain variation and faster for another, and that might depend on its location on a die. Not only the importance of the intra-die variations has grown but the total number of process parameters that present considerable variations has also increased (BLAAUW, 2002). This situation requires some changes in the STA techniques in order to find a better alternative to their deterministic nature.

In deep sub-micron technologies, the minimum feature sizes have approached the limits of photolithography, etch and ion implantation systems. The technology is approaching the regime of randomness in the behavior of silicon structures. Some dimensions in the devices are getting closer to an atomic scale, at which the classical theories must be replaced by quantum physics in order to explain device operation. A

large variability in performance and power consumption among different chips is expected since the device parameters, such as channel length, oxide thickness, threshold voltage and random placement of dopants in channel, may have variations. Since the process variations become a more significant problem because of the aggressive technology scaling, a shift from deterministic to statistical analysis for circuit designs may reduce the conservatism and risk that is present while applying the traditional technique.

The variation in a physical parameter can cause more than one electrical parameters to vary, giving rise to a correlation of these electrical parameters. Correlation of parameters variations must be considered in timing and power analysis, otherwise pessimistic results may result. As an example, gate width variations in a transistor have different (inversely correlated) impacts on the resistance (R) and capacitance (C) of the device. It means that while a variation in gate width decreases the resistance, it increases the capacitance, and worst-case values for R and C would lead to unrealistic RC estimates. The number of random variables one deal with increases rapidly when intra-die variations are considered, and that increases computational cost.

Statistical analysis overcomes the difficulties of the traditional methodology by treating the characteristics of the circuits, such as signal delay and power consumption, as probability density functions (PDF). The process parameters are no longer fixed values but random variables with certain statistical distributions. A statistical approach must be developed in order to predict performance and power consumption more accurately, making it possible not only to analyze the circuit efficiently, but also to optimize it.

2.3 Analysis of the Impact of Parameter Variation on the Threshold Voltage Variation

This section is intended to validate the use of non-correlated variations for the threshold voltages (V_{TH}) of MOS transistors, by considering that each transistor has a random V_{TH} variation, independently on its position on the die. The impact of channel length (L) and oxide thickness (T_{ox}) variations on V_{TH} were compared to the effect caused on the same electrical parameter by random-dopant fluctuations (RDF). RDF cause different and non-correlated variations in V_{TH} of transistors whether they are placed in the vicinity of each other or not. The opposite happens to variations in L and T_{ox} that presents strong correlations across the die area.

2.3.1 Variations due to Random-Dopant Fluctuations, Channel Length and Oxide Thickness Variations

Nanoscaled MOS transistors present statistical variation in their threshold voltages because of random number and placement of dopants in the channel region. The following equation is used to incorporate this effect in the threshold voltage deviation (HOON, 2004):

$$\sigma_{(V_{th})} = \frac{q}{C_{ox}} * \sqrt{\frac{N_a * W_{dm}}{(3 * L * W)}} \tag{2.2}$$

where L and W are the channel length and width of the transistor, respectively. N_a is the substrate doping concentration, q is the electron charge, C_{ox} is the oxide capacitance and W_{dm} is the maximum depletion layer width and it is given by the equation (ROY, 2003):

$$W_{dm} = \sqrt{\frac{(4 * \mu_{si} * K * T * \ln(\frac{N_a}{n_i}))}{(q^2 * N_a)}} \tag{2.3}$$

where n_i is the intrinsic carrier concentration, K is the Boltzmann constant, T is the temperature and ε_{si} is the dielectric permissivity of S_i.

In order to calculate the dependence of threshold voltage on the channel length of short-channel transistors, it is used an equation that quantifies the drain-induced barrier lowering (DIBL) effect (XUEMEI, 2003):

$$\Delta V_{th}(DIBL) = -\Theta_{th}(L_{eff}) * [2 * (V_{bi} - \Phi_s) + V_{ds}] \tag{2.4}$$

where V_{bi} is the built-in voltage of the source/drain junctions and is given by equation (2.5). Φs is the surface potential and is calculated through equation (2.6).

$$V_{bi} = \frac{K * T}{q} * \ln(\frac{N_{DEP} * N_{SD}}{n_i^2}) \tag{2.5}$$

$$\Phi_s = \frac{2 * K * T}{q} * \ln(\frac{N_{ch}}{n_i}) \tag{2.6}$$

θ_{th} is the short-channel effect coefficient (XUEMEI, 2003) and has a strong dependence on the channel length approached by equation (2.7):

$$\Theta_{th}(L_{eff}) = \exp(\frac{-L_{eff}}{(2 * l_t)}) + 2 * \exp(\frac{-L_{eff}}{l_t}) \tag{2.7}$$

l_t stands for the characteristic length and is calculated by using equation (2.8).

$$l_t = \sqrt{\frac{\mu_{si} * t_{ox} * X_{dep}}{\mu_{ox}}} \tag{2.8}$$

where ε_{ox} is the dielectric permissivity of the gate oxide and X_{DEP} is the depletion layer width given by equation (2.9):

$$X_{DEP}=\sqrt{\frac{2*\mu_{si}*(\Phi_s-V_{bs})}{q}*N_{DEP}} \tag{2.9}$$

Table 2.2 shows the results achieved by considering variations of 10% in L and T_{ox} and technology node of 45nm. The equations provided before in this section were used to evaluate the impact of the process parameters variations (L, T_{ox} and RDF) on the electrical parameter V_{TH}.

Table 2.2: V_{TH} deviation resulting from variations in the channel length, oxide thickness and RDF for NMOS and PMOS transistors.

Due to	*NMOS V_{TH} deviation (σ_{vthn})*	*PMOS V_{TH} deviation (σ_{vthp})*
Channel Lengh variation (10%)	0.0042	0.0019
Oxide thickness variation (10%)	0.0249	0.0227
RDF	0.0862	0.0469
Channel Length and Oxide Thickness variations	0.0253	0.0228
All variations	0.0898	0.0521

The probability density functions (PDF) of V_{TH} for RDF and variations in L and T_{ox} are presented in Fig. 2.10 and 2.11.

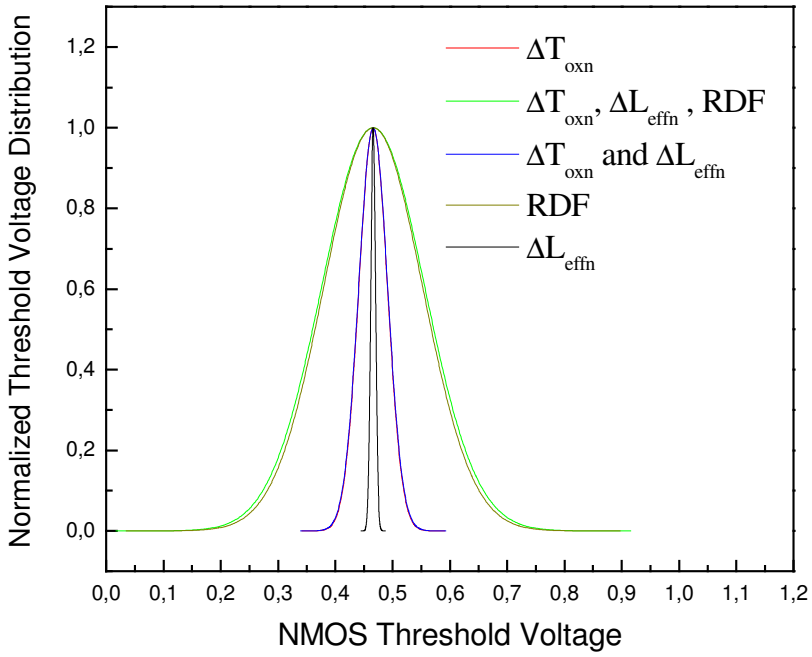


Figure 2.10: Threshold voltage PDFs of NMOS for process parameter variations.

It is observed that variations of L cause little deviation in V_{TH} distribution. A stronger deviation is resulted from variations in T_{ox}, but RDF present the most important influence on V_{TH} PDF. In this sense, it is quite acceptable the approach considered in this work, which makes use of non-correlated V_{TH} variations.

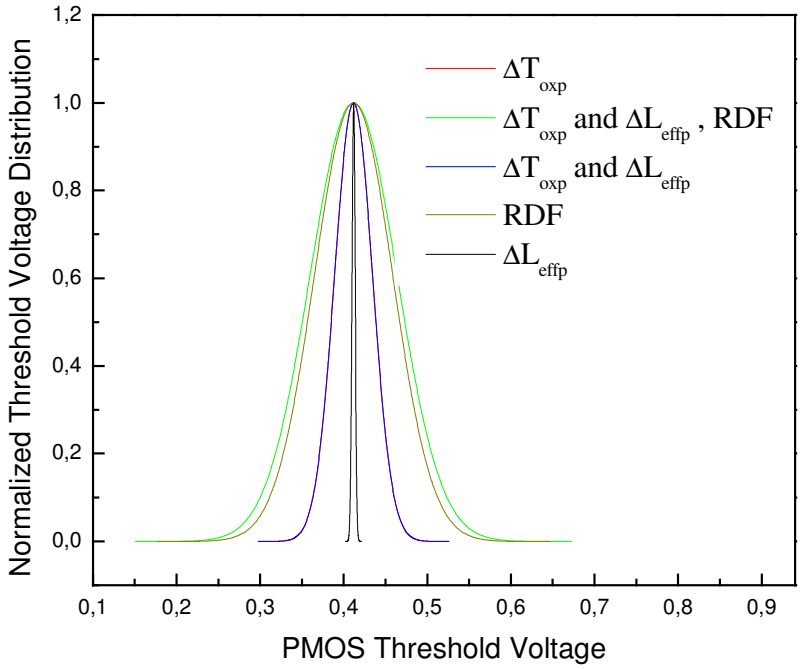


Figure 2.11. Threshold voltage PDFs of PMOS for process parameter variations.

3 TIMING ANALYSIS

This chapter revises the concept of static timing analysis (STA). The use of statistical models in timing analysis (statistical static timing analysis - SSTA) is also presented. These concepts are important to understand the goal of this thesis, which aims at introducing a method to evaluate the statistical characteristics of cell-level networks to be used in circuit-level SSTA. The intention is to provide the context in which the work is inserted and to show some recent efforts to deal with variability in digital design.

The timing of a digital circuit can be verified dynamically or by static timing analysis (STA). Dynamic analysis requires the generation of a set of input vectors, which excite all possible paths in a circuit. It can be performed using circuit simulator SPICE (Simulation Program with Integrated Circuit Emphasis), fast circuit simulators, or gate-level simulators. STA does not rely on input vectors and provides input-independent worst- (maximum delay) or best-case modeling (minimum delay) by using a method that propagates the *arrival signals* through a circuit from the inputs to the outputs (BLAAUW, 2002).

3.1 Critical Path Method (CPM)

Critical Path Method (CPM) – also called Program Evaluation and Review Technique (PERT) – is a technique used for building and evaluating circuit graphs. A circuit graph is a set of (i) internal vertices representing the gate inputs and outputs, (ii) vertices corresponding to primary inputs and outputs and (iii) connections between primary inputs to gates inputs, between gates, and between gates outputs to primary outputs (Fig. 3.1).

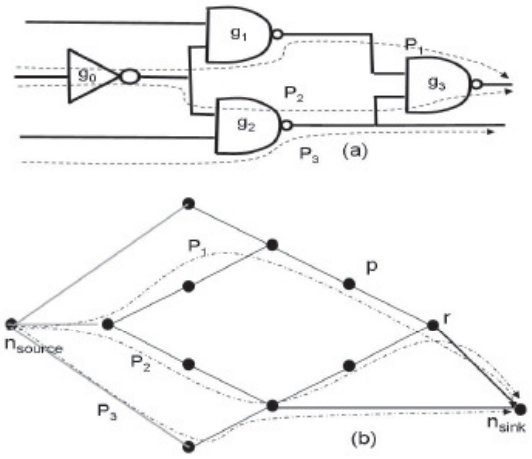


Figure 3.1: An example of a circuit (a) and its timing graph (b) (BLAAUW, 2002).

The procedure consists in propagating the delay of gates and interconnects of a circuit from the source node to the sink node using *sum* or *max* operation in order to find the critical path. The critical path is the one between an input and an output with the maximum delay. The individual gates are pre-characterized and their timing specifications are used to calculate the arrival time at each node. As the arrival times traverse gates, the delay of the gate is added to the arrival time and a maximum value is selected in the case of a multi-fanin node.

Since static timing analysis must provide correctness for any input vector, accurate gate and interconnect timing modeling is demanded. In a standard-cell flow, the gate delay is typically pre-characterized for different capacitive loads and input transition times. The libraries contain analytical expressions for the delay or timing tables that are generated by performing circuit simulation with SPICE after the extraction of the layout parasitics for cells in the library. STA assumes full correlation of process parameters within a die, what means that all devices and interconnect on a chip slow down or speed up in tandem. That would be a reasonable approach if no variations within the same die took place (CHIANG, 2007).

Traditional deterministic computer-aided-design (CAD) tools rely on the use of corner-case models that set worst-case, best-case or nominal values for the process parameters of an integrated circuit while analyzing its performance and power consumption (SRIVASTAVA, 2005). For each process condition the delay of the gates at that process condition is specified. This methodology implies in a great number of simulation runs if the effects of a large amount of sources of variations are to be considered, and can end up being pessimistic (worst-case assumptions) or optimistic (best-case assumptions), and risky at the same time, since not every corner is simulated (VISWESWARIAH, 2006). A deterministic description does not present the variance of a process parameter, but only its mean value. Worst-case values are taken at the analyzed corners, what is a pessimistic assumption, but since it is intractable to cover all possible corners, the missing ones may lead to failures that are not detected before manufacturing the chip. Another drawback of a corner-based STA is the possibility of identifying incorrect critical paths. These paths are obtained by assuming all the devices with the same deviations characteristics, but delay variations can lead to change of critical paths (PAN, 2005).

3.2 Statistical Concepts

The value of a parameter cannot be predicted exactly. This happens because the behavior is (a) unknown, (b) truly random or (c) because it cannot be represented by a model (ORSHANSKY, 2008). As already said before, in a statistical fashion the process parameters are defined as random variables.

A random variable (RV) is any function that assigns a numerical value to each possible outcome of some experiment (ORSHANSKY, 2008). A finite number of outcomes is represented by a discrete RV and an infinite number of outcomes is represented by a continuous RV. A common case of discrete random variables that are encountered in circuits manufacturing is where the RV, let's call it X, can have one of two values. One example is the probability that a circuit has of failing or working correctly. If we get a sample of circuits with the same failure probability to be tested, then the number of working circuits is a Binomial random variable.

Normal, uniform, and log-normal random variables are some of the continuous RV encountered in statistical design applications. Various physical phenomena and parameters follow normal distribution. A normal random variable is described by a probability density function:

$$f(x)=\frac{1}{\sigma\sqrt{2\pi}}e^{(\frac{-1}{2}).(\frac{x-\mu}{\sigma})^2} \tag{3.1}$$

where **μ** is the mean and **σ** is the standard deviation. For a continuous random variable, the probability density function is represented by a curve such that the area under the curve between two numbers is the probability that the random variable will be found in the interval limited by those two numbers (Figure 3.2).

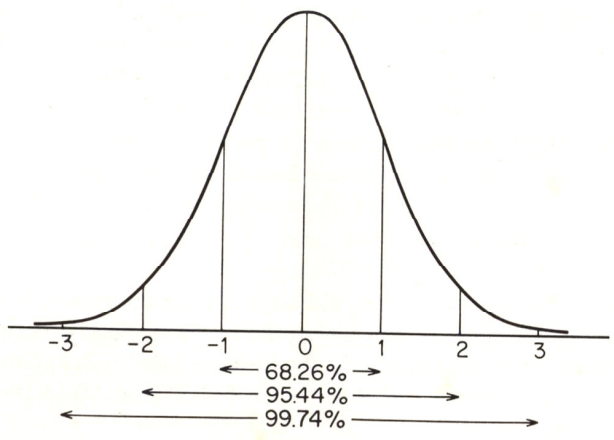


Figure 3.2: Normal distribution of a random variable. The horizontal axis represent the standard deviation (MODE, 1966).

A cumulative distribution function (CDF), which is the integral of the PDF is also used to represent an RV:

$$P(X\leq a)=F(a)=\int_{-\infty}^{a}(\frac{1}{\sigma\sqrt{2\pi}}e^{(\frac{-1}{2}).(\frac{x-\mu}{\sigma})^2}.dx) \tag{3.2}$$

and identifies the probability that the random variable X takes on a value less than or equal to *a*.

A random variable is fully characterized by its PDF and CDF, and partially described in terms of the moments of its probability distribution. The most important moments of a RV are the first moment, the mean (μ), and the second moment that is the variance (σ^2). Usually, the random variables in a statistical analysis of a circuit present some kind of correlation in their behavior. This codependence can be evaluated by using a covariance matrix that defines pair-wise correlations between variables. The covariance measures how much two RV vary together.

A linear combination of normal random variables also follows a normal distribution, but if the function is not linear in the RV, its distribution is not normal. However, an approximate method based on a first-order Taylor-series expansion of the function can be used if it is nearly linear in the small range.

3.3 Statistical Static Timing Analysis (SSTA)

Since cell-level statistical timing analysis was performed in this work, it is important to be aware of the great potential it comes up with for circuit analysis and optimization. Some statisical approaches and techniques are then presented here.

The extensive research in timing analysis shows the need for a new method of dealing with the following issues (ARGAWAL, 2003): (i) the ability to control critical device parameters is becoming restricted, since the process geometries continue to shrink, what results in significant variations of these parameters; (ii) the total number of process parameters that exhibit variations has increased, making the number of *corner files* (files that specify the characteristics of the gate for each process condition) increases rapidly; (iii) within-die (intra-die) variations have become a significant component of the total variation; (iv) in addition to device parameters interconnect parameters must be considered.

The deterministic formulation of static timing analysis treats the delay of gates and paths as fixed numbers, thus reducing a probabilistic problem to an arithmetical one. Statistical timing analysis may manipulate explicit parametric functions of delay on process parameters naturally, thus removing the need to identify the worst-case conditions a priori. In SSTA the performance metrics of gates and wires are modeled by stochastic values.

It is important to analyze circuit performance under process variation for yield prediction as well as for circuit optimization. The deterministic-based optimization for the circuit delay tends to concern only about the critical and near critical paths delay since there is no incentive to improve path delays that are not critical. In statistical analysis there is no sense in trying to identify a single path of the circuit as being the critical path, or the path with the maximum delay. Critical paths are then defined as a set of paths with high probability of becoming the slowest path in the circuit (SRIVASTAVA, 2005).

3.4 Statistical Solution Approaches

3.4.1 Numerical Integration Method

This kind of method operates in the space of manufacturing variations (parameter space) or in the space of path delays (performance space) (JESS, 2006). **Performance-space** methods manipulate timing variables such as arrival times and slacks as statistical quantities. The joint probability density function (JPDF) of the delays of all paths is integrated over a cube of side equal to the required delay and of dimensionality equal to the number of paths. **Parameter-space** methods perform manipulations in the space of the sources of variation. The JPDF of the sources of parametric variation are integrated over a complex feasible region in relatively low-dimensional space.

3.4.2 Monte Carlo Method

Monte Carlo simulation is a method for iteratively evaluating a deterministic model using sets of random variables as inputs. This method can sample a system in a number of random configurations and that data can be used to describe the system as a whole. By performing a full-scale transistor-level Monte-Carlo simulation of a circuit, one gets the most accurate way of incorporating the process variation effects into timing

analysis. It generates samples points for a given delay distribution and runs a static timing analyzer at each point. The results are put together to form the delay distribution (SAPATNEKAR, 2004).

3.4.3 Probabilistic Analysis Methods

Probabilistic methods usually propagate arrival-times through the timing graph by performing statistical sum and maximum operations. In these approaches the gate delays and arrival times are computed as random variables whose probability density functions (PDFs) are propagated through the circuit, and hence, the addition and maximum operations becomes a convolution operation and a statistical maximum, respectively. It means that once the delay of gates and interconnects in a circuit is modeled as a Gaussian random variable, the delay of a path is a Gaussian random variable as well (Figure 3.3).

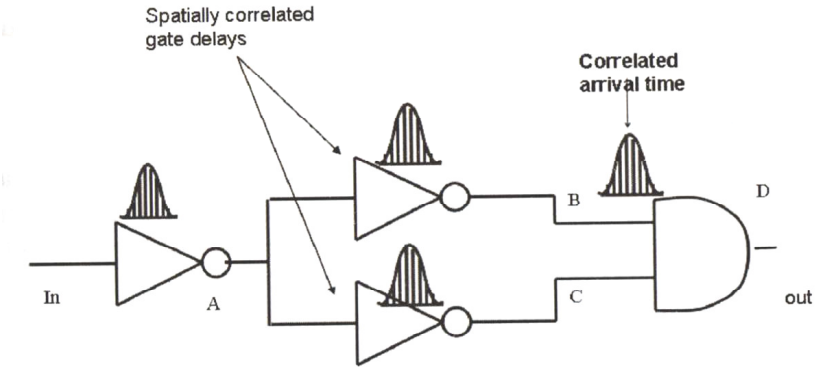


Figure 3.3: Gate and interconnect delays represented as probability density functions (SRIVASTAVA, 2005).

In SSTA the maximum of two random variables is not uniquely defined and a criterion must be used in order to compute the maximum of the variables.

3.5 Delay Modeling

3.5.1 Introduction

A digital circuit is constituted by transistors, usually organized into gates that drive interconnect wires. Typical approaches to timing analysis divide the design into stages, with each stage consisting of a gate output and the interconnect path it drives. Digital systems are often designed at the gate or cell level, making it possible to pre-characterize the gate and cell delay for timing analysis (CELIK, 2002). The cell delays and transitions are generally expressed empirically as a function of load capacitance and input signal transition. A delay calculator expects a certain waveform at the fanout as a function of the waveform at the switching input pin.

At this moment, that would be appropriate to focus on two parameters of a waveform, that are very important for timing analysis: delay and transition time.

3.5.1.1 Delay

Delay is defined as the interval between the time when the input waveform crosses a specified threshold, and when the output waveform crosses a given threshold. These two points are usually set as the points at which the waveforms reach half of their final value (50% point) while in transition (Figure 3.4).

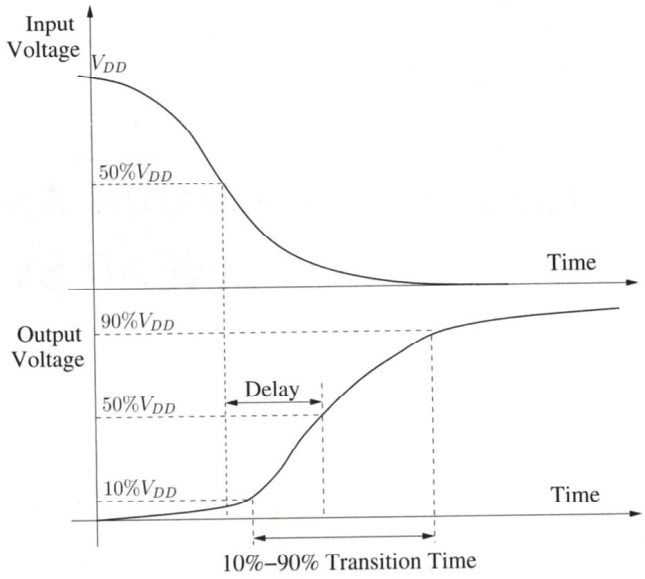


Figure 3.4: The 50% delay and transition time of a waveform.

3.5.1.2 Transition Time

The transition time of a waveform says how long it takes to reach its final value in a transition. Since most such transitions involve exponentials, it is commonly used the interval between the points 10-90% or 20-80% to measure the transition time.

3.5.2 Gate and Interconnect Timing Models

The components in a circuit can be represented by a model, which is an abstraction of the component behavior. By conceiving a model for each component, one can deal with its physical aspects in a more tractable way. A trade-off between accuracy and complexity must be taken into account in order to decide whether a model is reasonable or not. In contrast to designing circuits at the transistor level, gate or cell level design can reduce the design effort by pre-characterizing the gates and cells for timing analysis.

Delay evaluation in the Very Large Scale Integration (VLSI) design is a great concern and it becomes more critical in nowadays deep sub-micron technology. It is necessary to adequately account for nanometer effects during timing analysis, in order to predict the performance of the circuit accurately. Static Timing Analysis is a feasible method for chip-level analysis of the time constants of a circuit without simulation. Depending on the models used for gates and interconnects its accuracy might satisfy the timing constraints of integration-scaled circuits (KUONO, 2005).

The interconnect lengths do not scale in proportion to shrinking chip area, and that results in the dominance of the delay due to interconnect over the delay contributed by the gate. The great shrinking in size presented in recent technologies leads to some

challenges for modeling gate/cell delays: (i) model resistive interconnect effects. As metal widths get narrower, interconnects are becoming more resistive, and sometimes their impedance is much greater than the drive resistance of the driving cell (SYNOPSYS, 2005); (ii) model complex input waveforms; (iii) model delay variation due to cross-capacitance; (iv) model input capacitance. The input capacitance value is not constant and may depend on the falling or rising transition, on the output load and on the transition time of the signal; and (v) process variations.

3.5.3 Elmore Delay Model

The original model proposed by Elmore in 1948 presented an estimation of the 50% delay of a monotonic step response by the mean of its integral, what constitutes an impulse response. It was observed that the impulse responses of circuits with monotonic step responses are functions that can be viewed as probability density functions. The delay (measured at the 50% point) of the step response is equal to the median point of the impulse response (PILEGGI, 1998), and Elmore proposed to approximate the median by the mean, or first moment of the impulse response distribution, providing a dominant constant approximation for monotonic step responses.

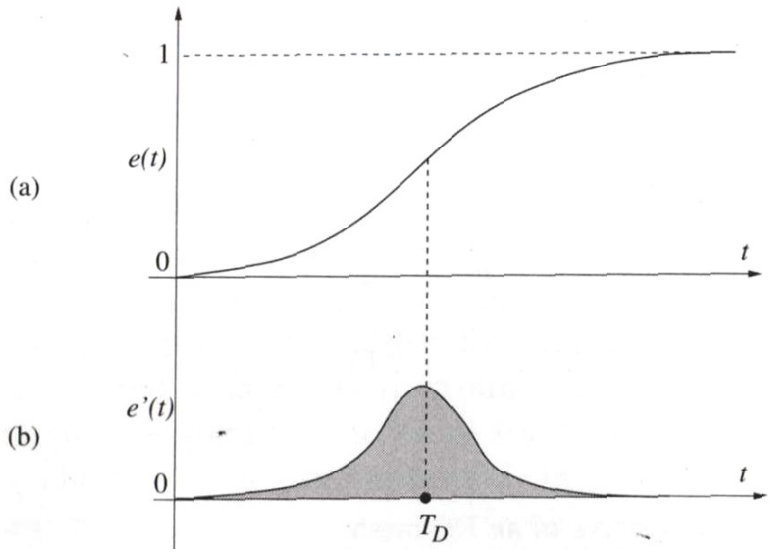


Figure 3.5: A step response *e(t)* (a) and its derivative (b).

The following equation is the first moment of the impulse response:

$$T_D = \int_0^{\infty} t.e'(t).dt \tag{3.3}$$

3.5.3.1 RC Tree

As the sizes of integrated circuits shrink and larger operation speeds are required, some aspects of timing analysis in a circuit are essential as they have never been before. Great attention is paid to the effect of the interconnections on the signal delay, and timing analyzers attempt to capture this effect through simplified models (SRIVASTAVA, 2005). Modeling gates and interconnections with resistors and capacitors have some advantages over more detailed simulation procedures, in spite of the loss in accuracy. For low and mid-frequency operation circuits, in which inductive effects in interconnect lines are supposed to be negligible, digital logic gates and their

associated interconnect paths may be represented as an RC tree. An RC tree is an RC (resistor-capacitor) circuit with capacitors from all nodes to ground, no capacitors between non-ground nodes, and no resistors connected to ground. Interconnects are commonly modeled with topologies that follow a tree like structure (Figure 3.6).

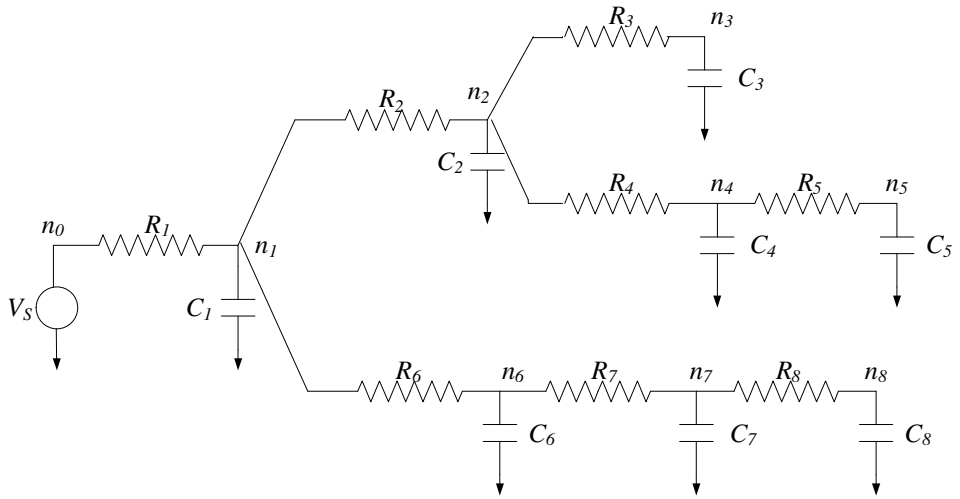


Figure 3.6: An example of an RC tree (SAPATNEKAR, 2004).

Proving that RC tree step responses are monotonic, Penfield and Rubistein (GUPTA, 1997) discovered the use of the Elmore metric for analyzing gate and interconnect delays and turned it into the most often used metric to calculate the signal delay in an RC tree. The widespread usage of Elmore approximation is mainly because of its simplicity since it is expressible as a closed-form expression and is additive. This feature makes it possible to decouple optimization problems into sub-problems, allowing optimal algorithms for buffer insertion and wire sizing.

The Elmore delay to node n_i in the RC tree is given by the expression:

$$T_{Di} = \int_{k=1}^{N} R_{ki} . C_k \tag{3.4}$$

where R_{ki} is the resistance of the portion of the path between the input and node *i*, that is common with the path between the input and node *k*, and C_k is the capacitance at node *k*.

This is the simplest delay metric that captures some of the resistance effects, but not the shielding of some downstream capacitance by wiring resistance. Elmore metric's drawbacks are the uncertainty of its accuracy and its strict validity for step response delay only (ELMORE, 1948).

3.5.4 Asymptotic Waveform Evaluation (AWE)

The Elmore delay is the first moment of an RC network under the impulse response, but higher-order moments can be used for a more accurate delay estimation. Asymptotic Waveform Evaluation provides a generalized approach to linear RLC circuit response approximations. The delay is calculated through the steady-state solutions at the internal

nodes of the network (SAPATNEKAR, 2004). A set of independent state variables must be chosen for a complete characterization of the RC network (as considered in this work) by means of the state equations.

The RC network can be described in terms of the following state equations:

$$x' = A.x + B.u$$
$$y = C.x + D.u \tag{3.5}$$

where $\boldsymbol{x}$ denote the state vector, which in our case is simply the vector of the capacitance voltages. The input vector $\boldsymbol{u}$ is the vector of the input independent voltage and current sources, and the output vector $\boldsymbol{y}$ is the vector of the output node voltages. The matrices A, B, C and D can be obtained in terms of the resistor and capacitor values and are dependent on the topology of the network. Figure 3.7 shows an RC network as an example.

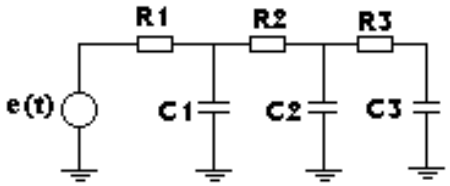


Figure 3.7: RC system.

The matrices G (conductance matrix) and C (capacitance matrix) are taken from the circuit equations:

$$G = \begin{bmatrix} G_1 & -G_1 & 0 & 0 & 1 \\ -G_1 & G_1+G_2 & -G_2 & 0 & 0 \\ 0 & -G_2 & G_2+G_3 & -G_3 & 0 \\ 0 & 0 & -G_3 & G_3 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Calculating the values of the variables:

$$G * X + C * X' = E \tag{3.6}$$

where X is the variables vector

The first moment is calculating by applying a DC voltage to the circuit (capacitors are open):

$$G * m_0 = E_0 \tag{3.7}$$

For *Eo = 1,*

$m_0 = [1\ 1\ 1\ 1\ 1\ 1\ 0]'$

The higher moments are calculated by taking away external sources:

$$G * m_i = -C * m_{(i-1)} \tag{3.8}$$

The calculated moments are matched via Padé approximation to reduced-order function models, which are used to characterize the circuit time and frequency domain responses with high accuracy.

A normalized transfer function for a linear system can be expressed as:

$$H(s)=\frac{(1 + a_{1.}\ s+a_{2.}\ s^2+\ldots +a_{n.}\ s^n)}{(1+b_{1.}\ s+b_{2.}\ s^2+\ldots +b_{m.}\ s^m)} \tag{3.9}$$

where *m>n*. The transfer function *H(s)* can be expanded into a power series with respect to *s*:

$$H(s)=m_0+m_1\,.\,s+m_2\,.\,s^2+\ldots \tag{3.10}$$

where m_i are the moments.

A Padé approximant is a lower order transfer function and it is characterized by its order (SAPATNEKAR, 2004). AWE constructs a q-pole transfer function *H'(s)* to approximate the actual transfer function *H(*s*)*:

$$H'(s)=\sum_{i=1}^{q} \frac{k_i}{(s + p_i)} \tag{3.11}$$

where p_i and k_i are the determined poles and residues. The corresponding time domain impulse response is:

$$h'(t)=\sum_{i=1}^{q} k_i\,.\,\mathrm{e}^{-p_i.t} \tag{3.12}$$

In AWE the delay of any output node ***i*** in a general RC network is defined as the time taken by the asymptotic approximation of the voltage at the node to reach its steady-state value. For the case of an RC tree model, a first-order AWE approximation reduces to the RC tree methods.

3.6 Process Variation Modeling

The analysis of process variation can be considered at two different levels: at the chip-level, which deals with the inter-die variation and at the transistor-level, wich refers to the intra-die variation. The Pelgrom model has been widely used to study the mismatch in devices resulting from random and correlated sources of variations, and it will be discussed in the next section.

3.6.1 Statistical Delay Models

In (FATEMI, 2006) it is proposed a statistical model for logic cell timing analysis in the presence of process variations. It is used a current-based model that has its cell parasitics pre-characterized as a function of the input and the output values. Also, a mathematical method is applied to characterize the sensitivities of the cell elements with respect to the sources of variation.

There are two main components in the model (Fig. 3.8): (i) capacitances representing the parasitical behavior at input and output nodes and the Miller effect between the nodes, and (ii) a current source at the output node to model the nonlinear behavior of the cell.

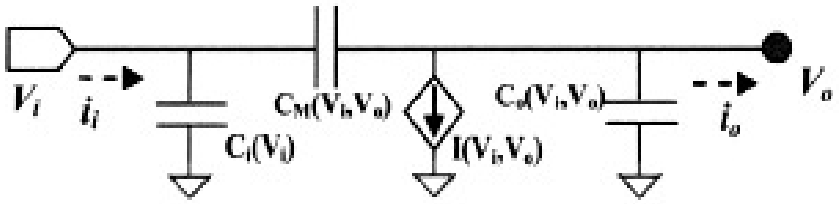


Figure 3.8: Current-based circuit model for a logic cell (FATEMI, 2006).

A 2-D lookup table stores I (V_i,V_o) values that are measured at the cell output while sweeping the DC values of input and output voltages. C_M and C_o values are also characterized through a series of SPICE-based transient simulations.

The physical parameters of interest are L_{int}, V_{tho}, T_{ox} and W_{int}. The logic cell elements, such as the current source and capacitive parasitics are represented as a first-order approximation function of these parameters:

$$E = e_0 + e_1 . \Delta X_1 + \ldots + e_m . \Delta X_m \tag{3.13}$$

where e_0 is the nominal value of the element and e_i is the sensitivity of the element E with respect to the physical parameter X_i. Random variables are applied to the cell elements and the output voltage waveform distribution in the presence of process variations is expressed.

(OKADA, 2003) proposes a model for calculating statistical gate delay variation caused by intra- and inter-chip variability that is based on a response surface methodology. It introduces sensitivity constants to facilitate the calculation of intra-gate variability without assigning variables to every individual transistor.

3.6.2 Pelgrom Model

It is a modeling technique used to capture the mismatch in transistors that suffer from variations in their process parameters. The impact of random and correlated variations is analyzed in the frequency domain (PELGROM, 1989).

Transistors that have been designed to have the same characteristics present mismatch when a parameter *P* varies over the surface of a die. Considering the *x-y* plane, the overal mismatch between two regions (Area1) and (Area2) corresponding to the points (x_1,y_1) and (x_2,y_2), respectively, can be expressed as:

$$\Delta P = \frac{1}{area}\left(\iint_{Area1} P(x,y)\, dx\, dy - \iint_{Area1} P(x,y)\, dx\, dy \right) \tag{3.14}$$

The integral in 3.14 can be viewed as a convolution of the function *P(x,y)* that describes *P* in the *x-y* plane, and a function $f_g(x,y)$ which describes the geometry of the problem. Thus, equation 3.14 can be rewritten as:

$$\Delta P(x,y) = (P * f_g)(x,y) = \int_{-\infty}^{\infty}\iint f_g(x',y').P(x-x',y-y')\, dx'\, dy' \tag{3.15}$$

where '*' is the convolution operator. The equation in the frequency domain then separates the geometry dependent part from the mismatch source:

$$\Delta P(\omega_x,\omega_y) = F(\omega_x,\omega_y)\, P(\omega_x,\omega_y) \tag{3.16}$$

where the operator F represents the two-dimensional Fourier transform.

The variance of parameter ΔP between two rectangular devices, as showed in Fig. 3.9, is given as:

$$\sigma^2(\Delta P) = \frac{A_P^2}{W.L} + S_P^2.D_x^2 \tag{3.17}$$

where A_P is the area proportionality constant for parameter *P*, while S_P describes the variation of parameter *P* with the spacing. It is clear in equation (3.17) that the model predicts the variance of a parameter of the device as inversely proportional to its area. In this sense, larger transistors would present less parameters variations caused by random fluctuations throughout it.

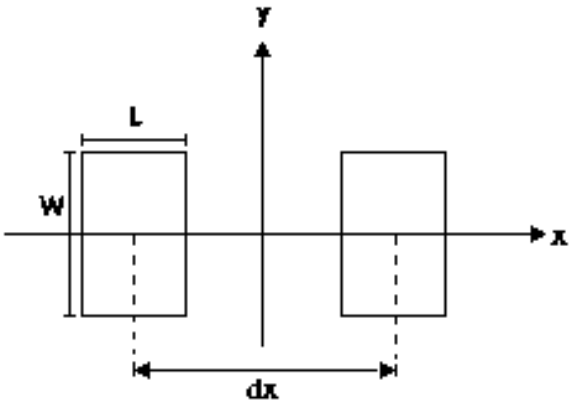


Figure 3.9: Representation of transistors that lie on the x-axis separated by a distance **x**.

4 PROPOSAL AND METHODOLOGY

4.1 Introduction

As already explained in CHAPTER 2, performance and power consumption of an integrated circuit is impacted by the fabrication process variations (channel length and width, gate oxide thickness, doping concentration and distribution etc). Regarding MOS fabrication process, variation effects do not scale proportionally with the design dimensions, causing the relative impact of the critical dimension variations to increase with each new technology. In nanoscale CMOS devices, the reduced average number of dopant atoms in the channel of a transistor causes the effect of random dopant fluctuations on its threshold voltage to increase (MAHMOODI, 2005).

Physical parameters are susceptible to random variations and the statistical nature of process characteristics makes it possible to consider the process parameters and their variations as random variables represented by their probability distribution function (PDF). In (MAHMOODI, 2005) the delay distributions of some logic gates are estimated by considering threshold voltage variations due to random-dopant fluctuations. Regarding the threshold voltage, its standard deviation is modeled as depending on the transistors dimensions (channel length and width, gate oxide thickness etc) and doping concentration.

The general goals of the present work are the analysis and variability estimation of transistors arrangements in order to point at the networks whose performance is as less susceptible as possible to the parameters variations of devices. The first part of the study is based on statistical (Monte Carlo) simulations, and it is intended to shed a light on the behavior of performance variability of the basic transistors structures and logic gates. The second and main focus of this work is on the implementation of a semi-empirical estimation method that describes and predicts this variability according to different transistors arrangements.

Besides the motivation and proposal of the work, this chapter describes in details the methodology applied in the analysis in order to figure out how variability in the threshold voltage of transistors affects a CMOS gate performance in different transistors network arrangements and switching situations. These data are able to indicate topologies that are more affected by variations and how each transistor resistance is affected by changes in the threshold voltages of transistors in the network.

4.2 Motivation

Parametric yield improvement may be achieved by reducing the variability of performance and power consumption of a cell. Figure 2.9 showed a two-side-

constrained region of operation of devices that suffer variations in their channel length. A high sensitivity of a device to variations in its parameters means that the yield window, limited by frequency and power constraints, is narrower than when a device is more immune to variability. A narrow yield window implies in a high quantity of manufactured chips that may not satisfy operational specification, leading to a higher cost of fabrication, since many chips may become useless.

In general, the behavior of signal propagation delays in transistors networks are well-know when it concerns to the increase or decrease of the number of devices in a stack or in a parallel array. Also, one can say that a change in the logic gate delay caused by a change in the position of the switching transistor in the array is at least qualitatively predictable. In many cases, the use of complex gates instead of smaller logic gates can lead to a faster signal propagation (SAKURAI, 1991). However, variability characteristics of the cells are becoming more and more important, since they can affect drastically the performance of the circuit and consequently the parametric yield. Some guidelines applied before in order to guarantee the functionality of a design in especific conditions must be reviewed for the sake of the yield. It is not only a question of reducing the delay of the cell anymore, but also providing a way that its variation do not compromises the operation of the cell.

Detailed electrical simulations of circuits are able to provide precise results for their performance, but they are computationally expensive. Timing models are necessary in timing analysis in order to perform a fast evaluation of the circuit. The models used in the analysis must represent the true behavior of the cell under certain conditions. The authors in (WEBEL, 2004) present a semi-empirical model based on type, geometry of the gates, body effect of transistors, slope of input signals, capacitance loads and threshold logic voltage. It adds a new timing parameter (latency time) which is added to the usual RC time constant.

Static timing analysis is usually used to find the critical points of the circuit that affect the critical path delay. Conventional sizing tools size the gates to optimize area and power consumption while meeting the desired delay constraint (SAPATNEKAR, 1993). The transistor widths are then sized to meet the desired delay constraint while keeping the power consumption and area within a limit. However, due to random process parameter variation, a large number of chips may not meet the required delay.

The traditional design efforts guided only by the best, worst and the nominal case models for the device parameters over- or underestimate the impact of variation. Approaches in the area of statistical static timing analysis have appeared to overcome the issues of corner-based methodology. The authors in (CAO, 2005) developes a physical model for analysis and prediction of circuit performance variability by coupling the Alpha-Power law based model with considerations of short-channel effects such as velocity saturation and threshold voltage dependence on channel length and drain bias (DIBL).

(CHOI, 2004) proposes a statistical design technique considering both inter- and intra-die variation of process parameters. The effects of process variations on the gate delay is pre-characterized and acessed on the fly during statistical timing analysis. The goal is to resize the transistor widths with minimal increase in area and power consumption while improving the confidence that the circuit meets the delay constraint under process variations. They have developed a sizing tool based on Lagragian Relaxation (LR) algorithm for global optimization of transistor widths.

Aiming at improving the parametric yield of chips, more robust cells could be achieved by choosing the adequate logic style and network arrangement for a certain logic function. One way of using the information of which cells are more immune to variability in a circuit design is to attribute a "cost" for each cell in a cell library related to its sensitivity to variations. This cost could be used in technology mapping to guide the use of less sensitive cells in a circuit that demands more robustness to variability. (SINGH, 2005) describes a new technology mapping algorithm that performs library binding to maximize parametric yield limited both by timing and power constraints. It proposes a statistical technology mapping that find a circuit mapping such that the yield at a required power objective is maximized while meeting the required timing constraints at a pre-specified yield level. It is indicated that a statistical technology mapping algorithm can produce mappings with reduced power consumption at the same power and timing yield levels.

4.3 Proposal

The goal of this work is to provide means to parametric yield improvement of ICs through a variability-aware design. The generation of cells that are more robust in relation to variability and the improvement of the parametric yield of a given circuit requires the knowledgement of how the parameters variations impact the performance of the cell. Statistical simulations (Monte Carlo) are able to provide reliable results once the parameters have random variations that follow a Gaussian probability density function (SRIVASTAVA, 2005), but are computationally costly. That is the reason why besides analyzing some transistors networks through simulations, a semi-empirical method that predicts performance variability according to the transistors arrangements is developed and proposed.

4.4 Methodology

The characterization is achieved through statistical (Monte Carlo) simulations of the networks. The analysis starts with simple configurations and later with more complex ones. The results provided by the simulations are then analyzed in order to provide some guidelines that can be used for designing more robust transistors networks. After these primary observations, simulations of some well-known logic cells are performed in order to verify the credibility of those guidelines first provided.

As a first insight, some topologies are analyzed with statistical (Monte Carlo) simulations to estimate the timing variability and compare these results with those achieved by using the semi-empirical method to be proposed. Different numbers of N- or PMOS transistors were considered in a stacking, in a parallel arrangement or in a mixed (series-parallel) network, and also, different positions for the switching device in relation to the output node. The results achieved with some simple transistor arrangements are intended to be the starting point to analyze and predict the variability of logic gates usually present in cells library to be used in technology mapping. In this sense, an important step is to analyze the V_{TH} variations impact on largely used logic gates presenting different topologies and different transistors networks arrangements.

Electrical simulations are proceeded in order to show the variability performance of inverter, NAND and NOR, XOR (in different configurations) and AOI (AND-OR-INVERTER) gates. Also, different topologies of full adders are analyzed. The

simulations are able to provide important information on the behavior of the cells according to their topology. All the variability scenario obtained through statistical simulations is important, but not enough in the study of performance variability. It is fundamental to understand the physics behind the observed tendencies. Besides carefully analyzing the results, the implementation of a method that can predict the gate variability is an important step.

Some delay models for logic gates are evaluated according to the possibility of adapting them to variability delay models. As an example, the authors in (DAGA, 1999) propose an analytical modeling of the speed performance of CMOS gates that is based on the average transfer of charges across the switching nodes under consideration and explicitly use the threshold voltage of the involved transistors in the calculation of delay. Therefore, this model could be used to find an equation that describes how sensitive the delay is in relation to the threshold voltage of devices in the charging/discharging paths. Though this model has some advantages, such as considering the input slope, the input-to-output capacitance coupling and short-circuit current effects, an even simpler but not less efficient model is aimed.

A simple way to analyze signal propagation delay is to replace the transistors by their on-resistances and calculate or extract the intrinsic capacitances of the cell in order to find the RC time constant of the network. There is really a huge amount of work dealing with this type of analysis (RABAEY, 2005) (SRIVASTAVA, 2005), so it is quite reasonable to try to use this method to also analyze delay variability. An RC analysis is a very low-level approach that is able to deal with physical characteristics of the devices, including geometry and intrinsic properties of materials used in the process fabrication.

The variations of many process parameters are translated into variations of the threshold voltages of transistors. That would be much more complicated to study the influences of each physical or electrical parameters, since they can be correlated. As an electrical parameter that is dependent on some process parameters, such as channel length (in submicrometer technologies) and oxide thickness, threshold voltage can be considered the final parameter affected by process variations. It is studied as a random variable represented by its probability density function. The threshold voltage of one device is assumed to be independent of the threshold voltage of another, what implies in non-correlated variations even for transistors placed close to each other in a die.

Before calculating the delay, it was found of great value to analyze the impact of V_{TH} variability on transistors resistances. Then, the delay of signal propagation on the network is estimated as an RC time constant by the use of Elmore Delay (ELMORE, 1948) and also calculated by using Asymptotic Waveform Evaluation – AWE (SAPATNEKAR, 2004) in order to overcome the limitations of Elmore technique and get a beter approximation. The statistical analysis provided by the method becomes possible after some steps that include linear regression and sum of probability density functions considered for the threshold voltages.

In this work, the intrinsic capacitances are calculated for a specific technology node. Also, a load capacitance is represented by the gate capacitance of an inverter with five times the drive strength (for most of the analysis) of the logic gate used as its driver. The transistor is considered as an open-switch when it is in the cut-off region and as an resistance when it is conducting. Two types of resistances were calculated to represent each device (i) when it is conducting with steady input signal applied to its gate and (ii)

when it is conducting, but with a transient input. The resistances were taken as an average of the resistances calculated for four different drain-source voltages. The reliability and limitations of the proposed method are discussed along with the possible causes of the results achieved.

CHAPTER 5 is dedicated to the observations provided by statistical simulations of the performance of different networks and logic cells. CHAPTER 6 explains the conception of the method, and CHAPTERS 7 and 8 state the results achieved by using the proposed method. The method is also validated by comparing its results with those provided by the simulations.

5 CMOS LOGIC GATE PERFORMANCE VARIABILITY

5.1 Introduction

The analyses presented in this CHAPTER are pre-characterizations of some logic cells and comprehended the first step of this work. The characterizations provided an idea of how is the behavior of the transistors network under the effect of threshold voltage variations. This information might also be useful in the development of design guidelines for parametric yield improvement. It is evaluated the impact of transistor threshold voltage variations on delay behavior of CMOS logic gates, according to (i) network topology (transistor arrangement) and (ii) the relative position of the switching transistor in relation to the power supply and output terminals.

This part of the work aimed at analyzing the delay variability of different transistors networks in relation to variations in the threshold voltage of devices. Electrical simulations were proceeded in order to show the variability performance of a CMOS inverter, NAND and NOR, and AOI (AND-OR-INVERTER) gates. The threshold voltages (V_{TH}) of transistors were varied and timing measurements (delay) were taken for all the configurations. The mean delay and standard deviation of the logic gates were compared and the relation of these values to the transistor network arrangements is emphasized. Rise and fall delays of the gates – inverter, 2-, 3- and 4-input NAND, 2-, 3- and 4-input NOR, 2-input XOR, AOI-21 and AOI-32 configurations – were verified with statistical (Monte Carlo) simulations. Ten thousand simulations were run for each experiment. The measurements were taken for a 3σ deviation of 10% of the nominal threshold voltage of transistors. The *normalized standard deviation* (σ/μ) of the metrics were compared for different transistor network arrangements. The *normalized* standard deviation makes it possible to compare the variability of arrangements with different mean delays. The technology node used in this work is 45 nm and the model file is a Predictive Technology Model (PTM) (ZHAO, 2007) based on BSIM4. No correlation between different types of transistors were taken into account, what means that a PMOS placed in the vicinity of a NMOS may come up with different variations in its parameter. Simulations were carried out by using HSPICE tool.

5.2 Variability of Different Transistors Networks

This first analysis is performed by changing the topology and number of transistors in the network to be studied – pull-up or pull-down network. In this case, variations in

the threshold voltage are considered only for devices of the respective branch. The simulations were run for configurations with the same drive strengths $\frac{W_p}{W_N} = 2$.

5.2.1 Pull-down network

Some of the topologies used for analyzing fall delay variability are presented in Fig. 5.1. The same signal was applied to all the inputs of the test structure. By running statistical simulations (Monte Carlo) for the networks in Fig. 5.1 (a) and (b) with different number of devices, it was observed that for the series network arrangement, fall delay deviation slightly increases as more NMOS are placed in the gate. For the parallel network, fall delay deviation decreases for higher number of transistors. The values of delay variability are presented in Table 5.1 for different number of transistors in series and parallel arrays. Configurations with series and parallel devices such as that presented in Fig. 5.1 (c) showed that if the parallel branch is close to the output node, the delay variability is lower than for a parallel branch far from the output. In the case of a series/parallel topology, a higher number of NMOS transistors causes a slight decrease in the variability when the same switching condition (position and number of switching devices) is kept.

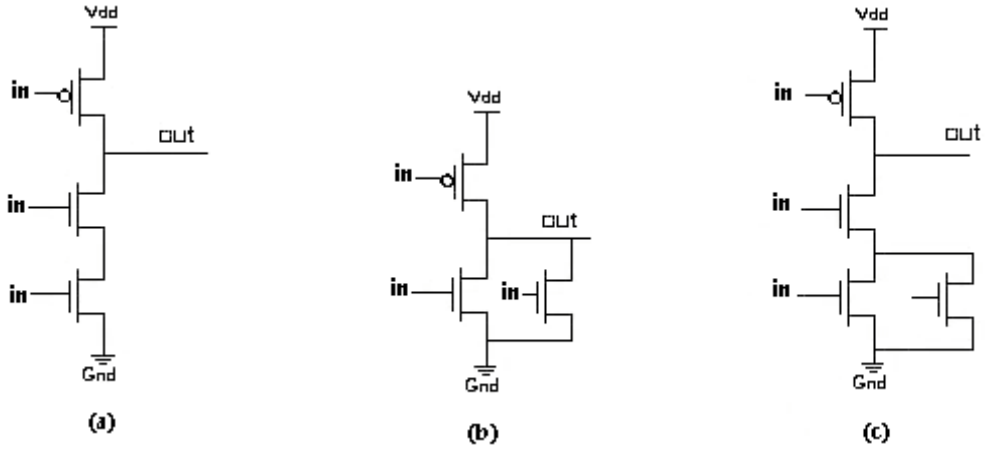


Figure 5.1: Some of the CMOS logic structures used for the analysis of fall delay variability in relation to the NMOS transistor network arrangement.

Table 5.1 – Normalized fall delay deviations for different topologies.

Number of Transistors	*Fall Delay Deviation*	
	Series	*Parallel*
2	0.0347	0.0403
3	0.0336	0.0291
4	0.0364	0.0262

5.2.2 Pull-up network

Rise delay variability is studied by using topologies similar to those presented in Fig. 5.2. The simulations for the topology presented in Fig. 5.2 (a) with different number of series PMOS reveals that larger number of transistors results in lower variability, as shown in Table 5.2. Parallel networks, as in Fig. 5.2 (b), presented higher variability as the number of PMOS increases. The values of delay variability are presented in Table 5.2 for different number of transistors in series and parallel arrays. Results for configurations with series and parallel branches such as that presented in Fig. 5.2 (c) showed that higher numbers of PMOS transistors do not cause considerable changes in the variability when the same switching condition (position and number of switching devices) is kept. The position of the parallel branch, far or close to the output node, does not affect the delay variability.

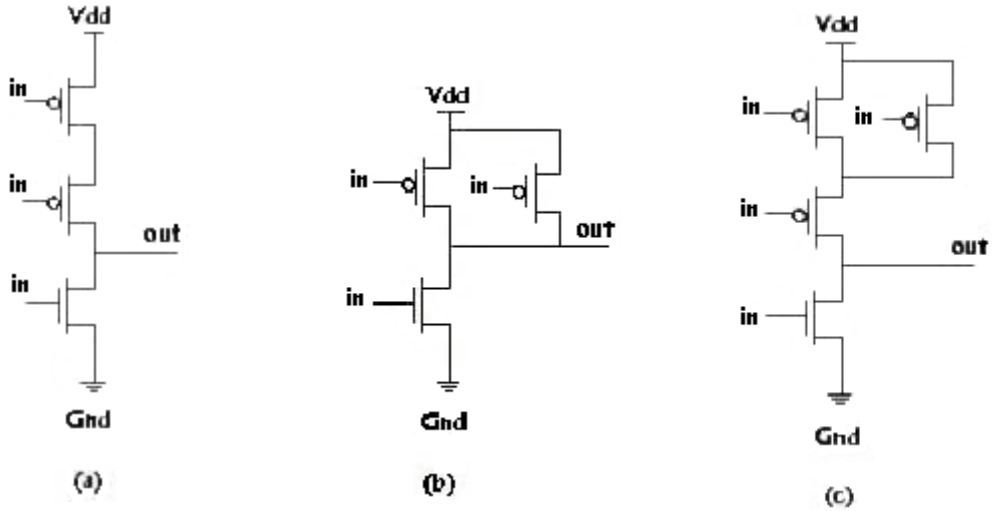


Figure 5.2: Some of the CMOS logic structures used for the analysis of rise delay variability in relation to the PMOS transistor network arrangement.

Table 5.2 – Normalized rise delay deviations for different topologies.

Number of Transistors	*Rise Delay Deviation*	
	Series	*Parallel*
2	0.0294	0.0303
3	0.0240	0.0246
4	0.0219	0.0211

5.3 CMOS Inverter

This section presents some DC characteristics and timing analysis performed on different designs of an inverter. The purpose is to analyze how the variability in the threshold voltage of transistors affects the DC characteristics and timing metrics of an

inverter according to (i) its drive strength, (ii) the cell fanout, (iii) its input transition time and (iv) its transistor network arrangement. Though this work deals with timing analysis, the following data are able to indicate which metrics are more affected by variations and which design characteristics make the gate more sensitive to variability, besides analyzing the delay variations.

5.3.1 Analysis

The threshold voltages (V_{TH}) of the transistors in an inverter are varied while DC characteristics (subthreshold and maximum currents) and timing measurements (delay and output transition time) are taken. The mean values and standard deviation (σ/μ) of the metrics are compared and the relation between these values and area, input transition time, fanout and transistor network arrangements is emphasized. The measurements were taken for a 3σ deviation of 10% of the nominal threshold voltage. The topologies are analyzed with statistical (Monte Carlo) simulations to estimate the metrics distributions.

5.3.2 Inverter Sizing

The first set of simulations were run for inverters with different drive strengths while keeping a constant ratio of widths (W_P/W_N). Several measurements established that W_N = 0.045 μm and W_P = 0.170 μm were the dimensions for a minimum-sized inverter (which one called X1) that has the threshold operation point ($V_{in} = V_{out}$) as close as possible to $V_{DD}/2$. It is shown in Fig. 5.3 the test structure used in this section and in the next. The first two inverters were placed between the ideal voltage source and the gate under analysis (X) in order to provide a more realistic input slope.

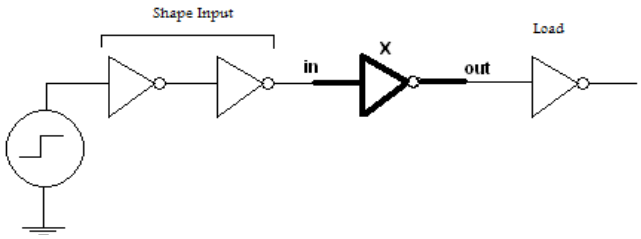


Figure 5.3: Design for analysis.

As seen in Fig. 5.4 to 5.7, the increase of the drive strength (X1, X2,…,X5) results in different behavior of the metrics and their variability. High and low noise margins are just a little affected by variations in the threshold voltages of the transistors and these data are not presented here. The subthreshold currents of NMOS and PMOS and the short-circuit current are proportional to the area (Fig. 5.4) but their variability, except that of NMOS subthreshold current, remains constant.

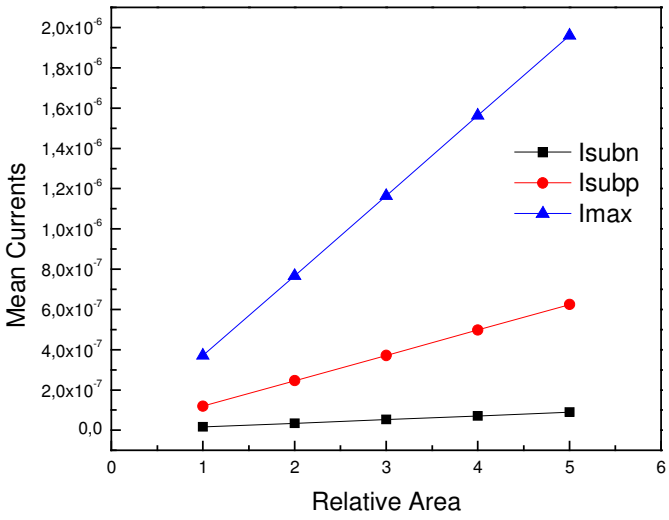


Figure 5.4: Mean subthreshold and maximum currents of an inverter in relation to its area.

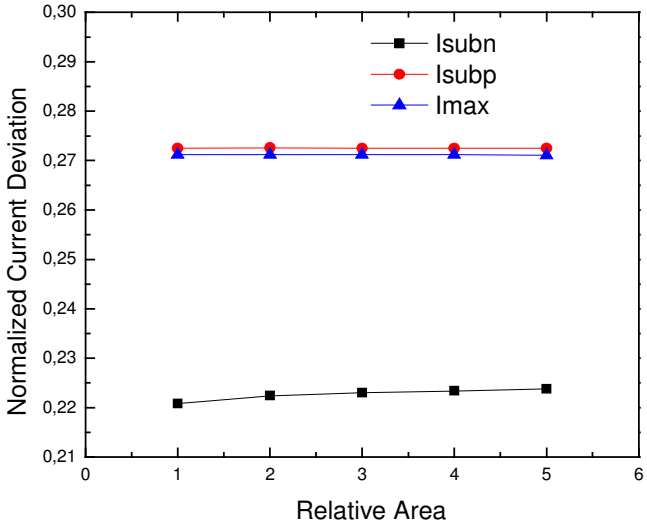


Figure 5.5: Normalized current deviations of an inverter.

The timing metrics of the gate - rise and fall delay, and fall output transition time - increase when the area increases. On the other hand, rise output transition time is just slightly affected over the sizing range (Fig. 5.6).

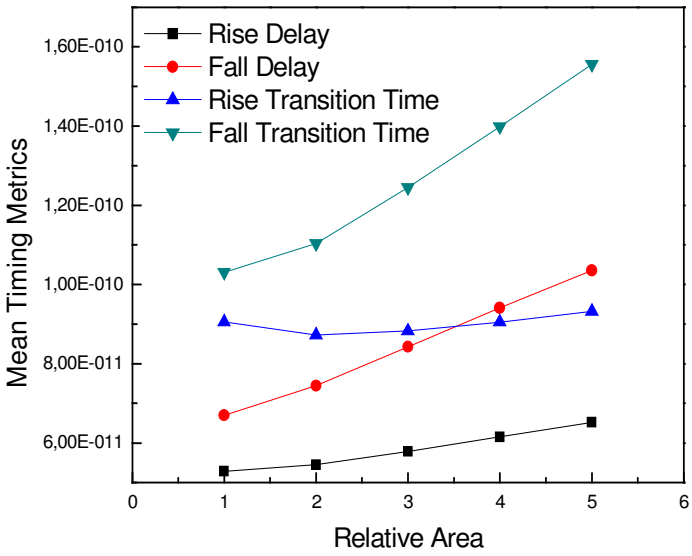


Figure 5.6: Timing metrics of an inverter in relation to its area.

It can be observed in Fig. 5.6 that the timing metrics directly related to the PMOS transistor – rise delay and rise output transition time – are less impacted by variations in the area of the transistors than the metrics depending directly on the NMOS – fall delay and fall output transition time. Regarding timing metrics deviation, their behavior is different from that of the mean values (Fig. 5.7). The larger the size of the inverter, the smaller the rise and fall output slopes and the rise delay. Only the fall delay presents larger deviation when the area increases. It means that falling transitions at the output node of inverters which are placed on the critical paths of the circuits are more critical for parametric yield and timing stability. Such information is quite useful for buffer insertion tasks (JIANG, 1998), for instance.

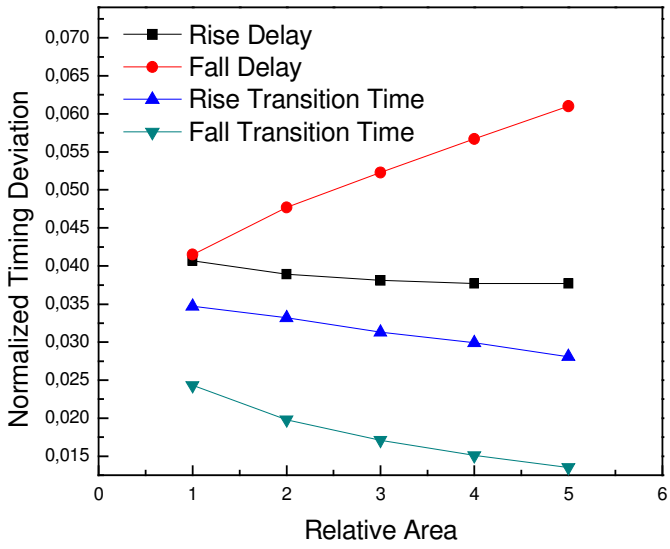


Figure 5.7: Normalized timing metrics deviation of an inverter in relation to the transistors areas.

The methodology used in the work considered the same threshold voltage variation (± 10%) independently on the size of the gate. Therefore, our purpose was to analyze how size (drive strength) affects currents and timing variability once a specific variation takes place.

5.3.3 Output Load

The influence of the fanout on the variability of the metrics is also studied. The fanout was represented by an inverter with a range of area from one to eight times the size of the driver inverter. The subthreshold and short-circuit currents, the noise margins values, and also their variability, are practically not affected by the different fanouts used in the simulations. However, as the fanout increases, the mean values of all of the timing measurements also increases, as in Fig. 5.8.

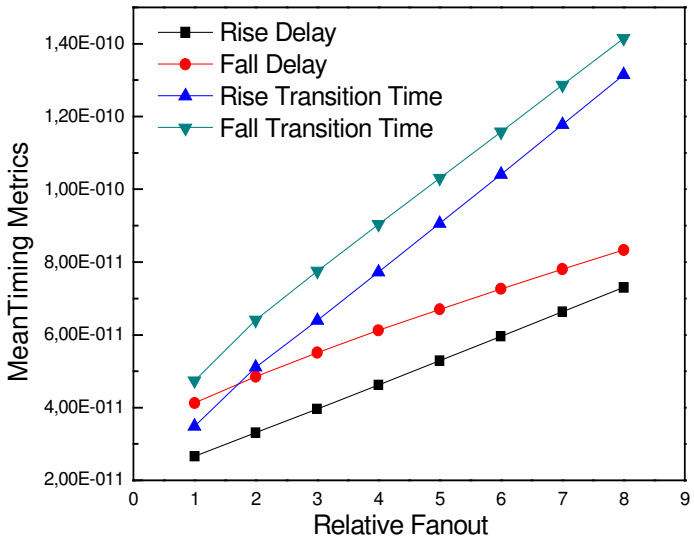


Figure 5.8: Timing metrics of inverter in relation to the output load.

The rise and fall transition times at the output node are more affected by changes in the fanout than the rise and fall delays. The normalized fall delay deviation is kind of a hyperbolic function of the fanout – as it increases, the deviation decreases (Fig. 5.9). The deviations of the other metrics increases in a quasi-linear rate, but the measurements for a falling-edge at the output are more sensitive to changes in the fanout of the gate. For a fanout cell with area between 5 and 6 times the minimum-sized inverter there is a point of intersection in the curves of delay deviations, that could be used as information to design gates with reasonable values for both delays.

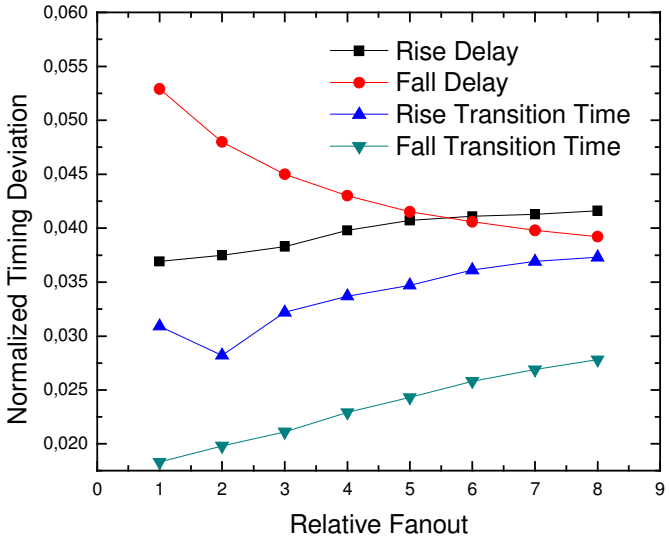


Figure 5.9: Normalized timing metrics deviation of an inverter in relation to the output load.

5.3.4 Input Transition Time

Inverters with different areas were placed at the input of inverter on which the measurements are being performed in order to change the transition time of the input signal. In Fig. 5.10, the input gate that contributes with the input capacitance is called X1 to X5, depending on its size relatively to the size of the main gate, called X1 (W_N = 0.045 µm and W_P = 0.170 µm).

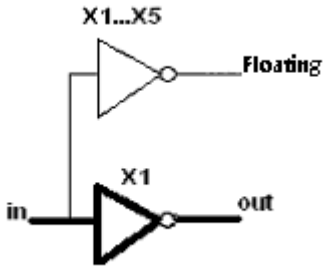


Figure 5.10: Topology using an auxiliary inverter (X1…X5) connected to the input of the main inverter for changing the input slope.

Fig. 5.11 and 5.12 present the results for the mean timing metrics and their deviations. The mean delays and output transition times are proportional to the size of the auxiliary inverter, what means that these metrics increase as the input slope increases. Regarding deviations, only the fall delay presents higher deviations for larger capacitances connected to the input. Rise delay deviation is practically constant and rise output transition time deviation decreases. The fall output transition time deviation has an interesting behavior, being a convex function of the size of the auxiliary inverter with a minimum value when it is sized as X3 (W_N = 0.135 µm and W_P = 0.510 µm).

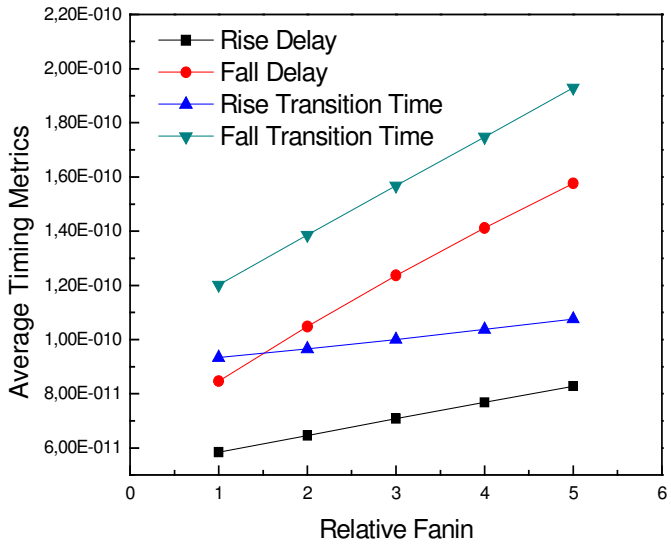


Figure 5.11: Timing metrics of an inverter in relation to the input capacitance.

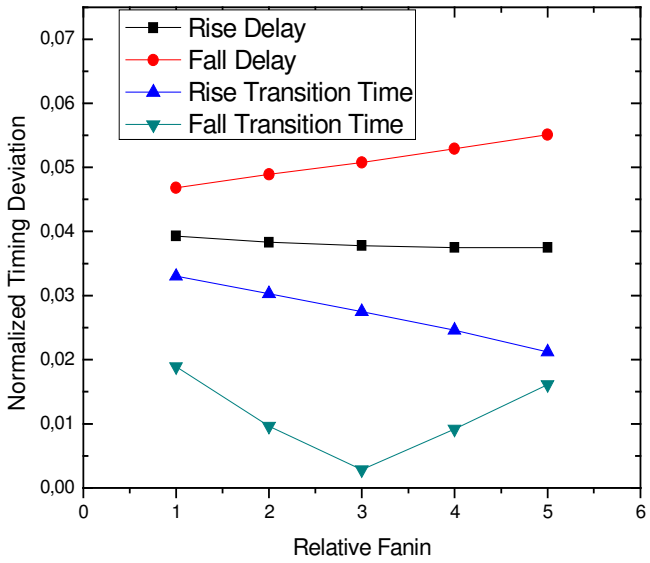


Figure 5.12: Normalized timing metrics deviation of an inverter in relation to the input capacitance.

5.3.5 Transistor Network Arrangements

Fig. 5.13 shows different topologies for an inverter keeping constant the ratio (W_P/W_N) and the drive strength of the pull-up and pull-down networks. The topology in Fig. 5.13(b), that represents the split of transistors in larger series ones, is commonly used for subthreshold leakage current saving. Fig. 5.13(c) represents a layout optimization technique called folding, that splits a transistor in smaller parallel ones. The comparison among the topologies showed that a series configuration of transistors results in lower delay variability for the currents analyzed, the delays and the output transition times.

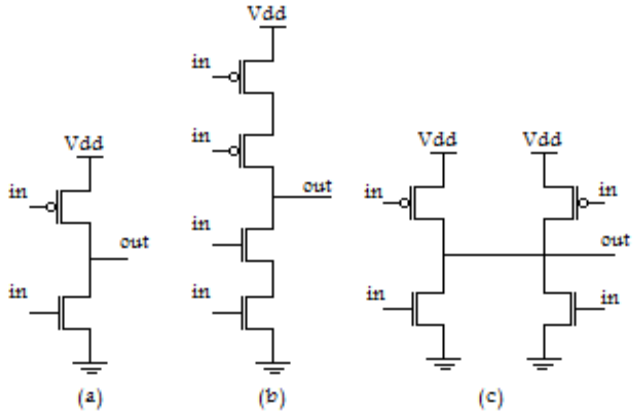


Figure 5.13: Different topologies of an inverter.

Tables 5.3 to 5.5 present subthreshold leakage current and timing measurements for the topologies in Fig. 5.13.

Table 5.3 - Subthreshold leakage current values for the inverter topologies.

Inverter Topology	*Subthreshold Current*			
	PMOS		*NMOS*	
	Mean (A)	*deviation*	*Mean (A)*	*deviation*
(a)	118.878e-9	0.2725	16.002e-9	0.2208
(b)	72.338e-9	0.2122	10.327e-9	0.1733
(c)	237.747e-9	0.2725	31.965e-9	0.2208

A stack of N- or PMOS transistors as in Fig. 5.13(b) present less subthreshold leakage current than the other configurations, since an arrangement with series devices in cut-off state leads to higher equivalent resistance than a single transistor or a parallel network. Also, the series arrangement presented lower delay deviation under threshold voltage variations for each transistors in the network.

Table 5.4 – Rise and fall delays for the inverter topologies.

Inverter Topology	*Rise Delay*		*Fall Delay*	
	Mean (s)	*deviation*	*Mean (s)*	*deviation*
(a)	58.896e-12	0.0407	67.025e-12	0.0415
(b)	73.875e-12	0.0283	104.617e-12	0.0355
(c)	34.489e-12	0.0318	63.374e-12	0.0522

Rise and fall delay variability is also lower for the series transistors configuration, but their mean values are the highest among the topologies presented. The same situation takes place for the rise and fall transition times, as presented in Table 5.5.

Table 5.5 - Rise and fall transition times for the inverter topologies.

Inverter Topology	*Timing*			
	Rise Transition		*Fall Transition*	
	Mean (s)	*deviation*	*Mean (s)*	*deviation*
(a)	90.603e-12	0.0347	103.084e-12	0.0243
(b)	114.074e-12	0.0250	123.923e-12	0.0138
(c)	93.183e-12	0.0344	118.415e-12	0.0218

Fig. 5.14 presents another folding design in which the inverter is divided into larger number of parallel transistors. This topology has the lowest delay variability among all presented so far.

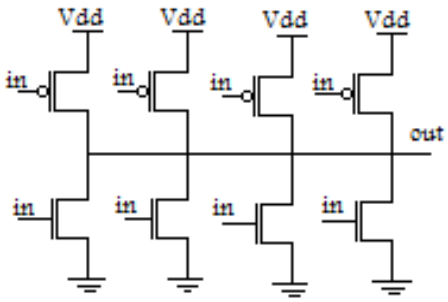
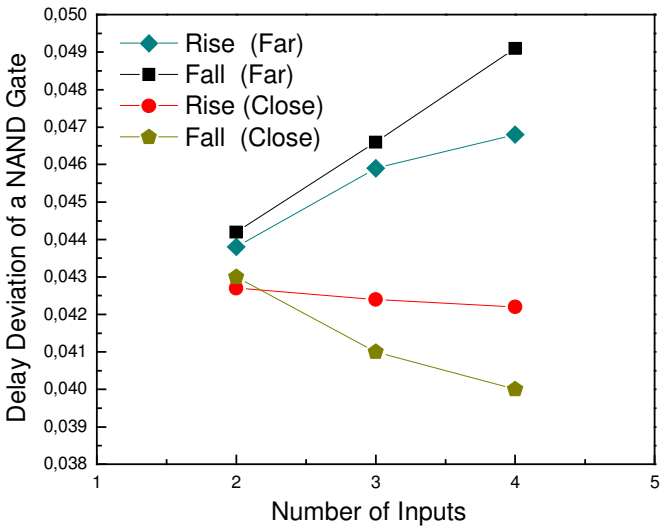


Figure 5.14: Folding topology of an inverter.

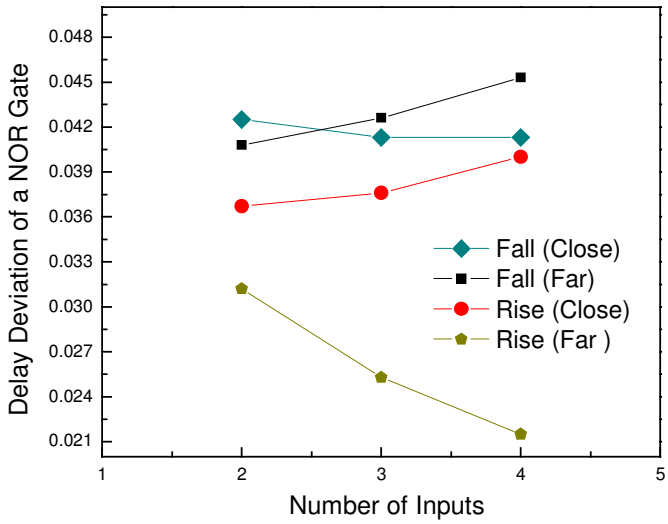
Successive folding seems a good technique when the goal is to control delay variability. However, the threshold voltage variations of the transistors were keept constant in the methodology here applied, independently on the new sizes transistors acquire in the folding technique. This point is a limitation of the analysis, since the Pelgrom model describes the threshold voltage as an inverse function of the are of transistors. So, larger devices have lower V_{TH} variations and that can compensate some of the advantage presented by the use of a higher number of smaller devices. Further analysis on this subject will be performed on CHAPTER 7.

5.4 NAND and NOR Gates

NAND and NOR static CMOS logic gates were also considered for such an investigation since they allow the evaluation of series transistors impact, for pull-up PMOS and pull-down NMOS transistor stacks in NOR and NAND cells, respectively. Usually, timing arcs are taken into account for each input signal transition. Fig. 5.15(a) shows rise and fall delay deviations according to the position of switching device in relation to the output node of NAND gates with different number of inputs. Two extreme situations can be identified: (i) when the switching transistor is connected to the cell output terminal ('close' switching) and (ii) when it is connected to the power supply terminal (V_{DD} or ground) in a stack arrangement ('far' switching). Transitions close to the logic gate output node result in lower mean rise delay and its deviation than transitions far from such node. In this case, the rise delay deviations obtained are similar for different numbers of inputs. For a signal applied close to the output, fall delay deviation decreases as the number of inputs of the NAND gate increases. The fall and rise delay deviations increase as the number of inputs increases for a transient signal applied far from the output node. Regarding the mean value of delay, there is an increase with the number of inputs, especially when the transient signal is applied far from the output.



(a)



(b)

Figure 5.15: Normalized rise and fall delay deviations in relation to the number of inputs: (a) NAND and (b) NOR gates.

In the particular case of NAND gates, lower delay values and delay deviations may be achieved when transient input signals are applied close to the output node. In a transistor stack there are differences in the potential of similar areas of devices, resulting in different gate-to-source (V_{GS}) and drain-to-source (V_{DS}) voltages. Therefore, variations in the threshold voltage may lead to different impact on the drive strength of devices. In NAND gates, the amount of charge that needs to go through a switching transistor far from the output is larger than when it is close to the output, considering other devices in 'on-state'. It helps to explain the dependence of performance variation of the logic gate on the position of the switching transistor.

Fig. 5.15(b) shows rise and fall delay deviations for transitions far and close to the output node of a NOR gate. In the case of a switching transistor close to the output node, NOR presents rise delay deviations that slightly increase with the number of inputs. The opposite happens for a switching transistor far from the output node, where the deviation decreases as the number of inputs increases. Rise delay is, in general, less affected by variations in the threshold voltage of transistors than fall delay, as observed in CMOS inverter.

When a series PMOS close to the output is switched the situation is similar to that one where a NMOS far from the output is applied a transient signal, in the sense that other intrinsic capacitances in the arrangement are already or still charged.

The analysis of series transistor configuration in NAND and NOR arrangements showed that the position of the switching transistor in relation to the output node influences the sensitivity of the gate to performance variations. In NOR gates the best situation (higher robustness) happens when the switching transistor is as far as possible from the output and less robustness is observed when the closest-to-the-output transistors are switched. In the case of NAND gate, in turn, higher robustness is

achieved by applying the transient signal close to the output node. The results for variations of delay are not the same as the results for the absolute delay value. It is well known that a better timing (lower delay) is achieved when a critical path signal is crossing through the switching device closer to the logic gate output node. A trade-off is required since it is not interesting to have the timing of the cell with a high mean value even though it presents low variability.

Rising- and falling-edge output signals go through essentially different paths in NAND and NOR gates. In the former, series transistors are in pull-down NMOS network and they are responsible for a falling-edge output signal. On the other hand, in the latter, series transistors are in pull-up PMOS network and are responsible for a rising-edge output. The comparison between the influences of the variations in the parameters on NOR and NAND delays are physically more appropriate by considering equivalent array of transistors: series-to-series or parallel-to-parallel. In this case, the fall delay variations of NAND gate may be compared to the rise delay variations of NOR gate, and vice-versa. Fig. 5.16 shows delay deviations for transitions far and close to the output node for NAND and NOR gates in stacked transistors. For both situations, NAND gates are more sensitive to variations in transistor threshold voltage than NOR gates.

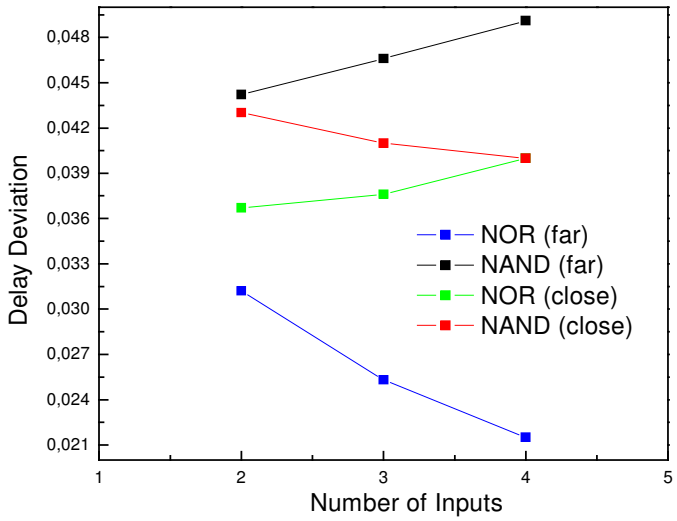


Figure 5.16: Comparison of NMOS and PMOS transistor stacking in NAND and NOR gates, respectively, for different positions of the switching device ('close' to and 'far' from output node).

By analyzing the sensitivity of basic gates to V_{TH} variations, some tendencies were observed in the deviations of their delay due to transistor network structure and the position of the switching transistor in relation to the output node. Such analysis cannot conclude that NAND and NOR gates with fewer inputs would be the best or the worst choice, once opposite behavior of delay deviation is observed according to the position of the switching device in the network. On the other hand, in critical paths optimization, by switching transistors closer to the gate output node tends to provide better performance in terms of absolute delay as well as parametric yield improvement.

5.5 NAND: Single Gate Versus Mapped Circuit

While evaluating topologies with different number of inputs a question arose: would it be better, in terms of variability, to replace a single complex gate with large number of inputs by a circuit mapped to basic gates with fewer inputs, in order to implement the same logic function? Fig. 5.17 illustrates how it could be done in the case of a 3-input NAND gate. Table 5.6 presents the results obtained for a single 3-input NAND gate ('NAND3') and a version composed with two 2-input NAND gates ('2xNAND2'). This case was investigated considering only one transistor switching at a time, and the fastest and the slowest paths were identified.

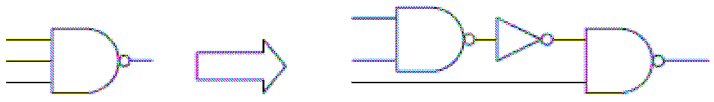


Figure 5.17: Illustration of single 3-input NAND gate implemented by using two 2-input NAND gates ('2xNAND2').

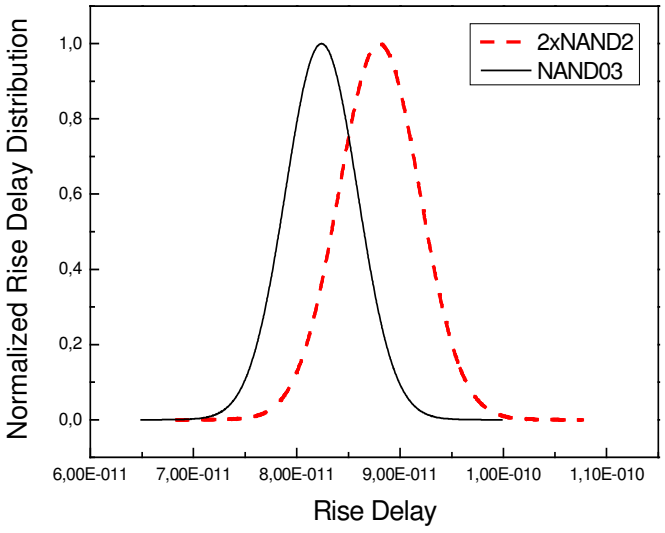
Table 5.6: Delay deviation of the shortest and the longest paths in Fig. 5.17.

	Best-case delay		*Worst-case delay*	
	2xNAND2	*NAND3*	*2xNAND2*	*NAND3*
Mean Rise Delay (ps)	87.99	82.39	153.93	141.30
Norm. Rise Delay Deviation	0.0446	0.0424	0.0293	0.0457
Mean Fall Delay (ps)	83.42	48.75	163.18	67.85
Norm. Fall Delay Deviation	0.0339	0.0466	0.0243	0.0410

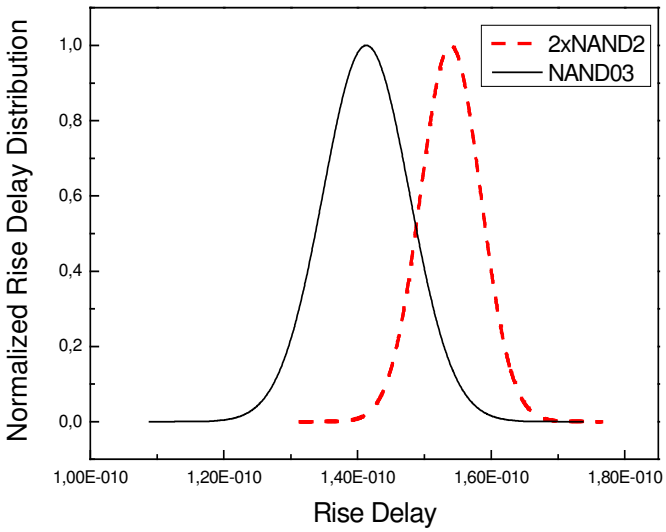
The single 3-input NAND gate is more sensitive to variations of transistor threshold voltage than the version composed by two 2-input NAND gates for the slowest signal propagation (worst-case), since variations in the threshold voltage of a 3-input NAND resulted in higher delay deviation than in the case when two 2-input NAND gates (with additional inverter) were used. For the fastest propagation (best-case) it is not completely so. Though fall delay deviation is higher for NAND3, rise delay deviation is almost the same for both configurations. Also, the NAND3 is much faster than the implementation with 2-input NAND gates for a falling-edge at the output node. The results shown in Table 5.6 agree well with Fig. 5.15a for rise delay deviation, once it is not really affected by the number of input signals in the logic gate.

A more complete analysis is possible by the probability density functions (PDF) of delay for both topologies, as presented in Fig. 5.18. Though NAND03 is more sensitive to variations in V_{TH}, the PDF of rise and fall delays for the longest and the shortest paths show that this gate guarantees faster signal propagation for almost every variation in V_{TH}.

It could be concluded that performing the technology mapping task using preferentially small (basic) logic gates instead of complex ones leads to a significant parametric yield improvement. It is probably true for the worst-case rise delay in Table 5.6, whose mean delay is similar for both approaches. In the case of the fall delay values shown in the same table, such analysis must be continued by considering circuit sizing optimization, once the mean fall delays are quite different.



(a)



(b)

Figure 5.18: PDF of rise delay for a 3-input NAND gate and a circuit performing the same logic function implemented by using two 2-input NAND gates for the best (a) and the worst (b) delay propagations.

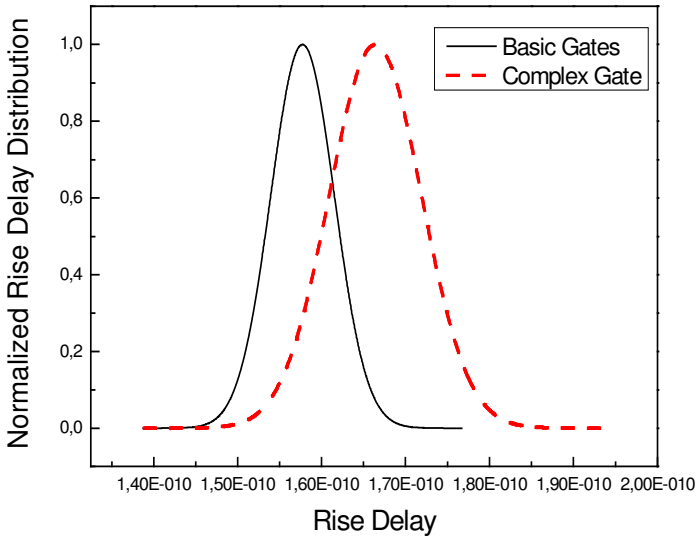
5.6 And-Or-Inverter (AOI) Logic Gates

Previous analysis, taking into account separately pull-down NMOS and pull-up PMOS logic networks, has demonstrated that the lower device count is present in a transistor arrangement, the less sensitive it is to threshold voltage variations. And-Or-Inverter configurations (AOI_21 and AOI_32) were implemented in two versions: (i) as a single CMOS complex gate and (ii) by using basic cells (2-input NAND and NOR gates). These topologies provide mixed arrangements of series and parallel transistors in the pull-up PMOS and pull-down NMOS networks. The goal is to evaluate if such implementation becomes more susceptible to variations than the same logic function mapped with basic gates.

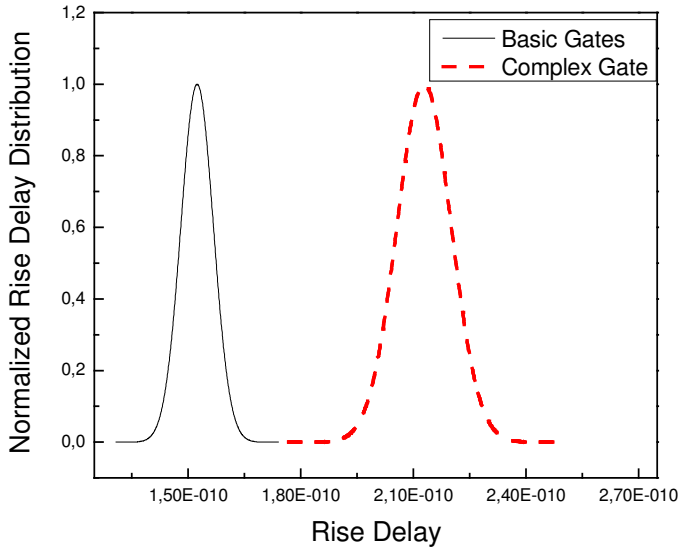
The implementation of AOI_21 by considering basic gates presented lower delay deviations, but higher mean fall delay in comparison to the single complex gate approach. Similar situation has been observed for AOI_32 implementation. The results are summarized in Table 5.7. Fig. 5.19 illustrates the rise delay distributions for both topologies.

Table 5.7: Delay deviation for AOI_21 and AOI_32 logic gates.

Metrics	*AOI-21*		*AOI-32*	
	Complex gate	*Basic gates*	*Complex gate*	*Basic gates*
Mean Rise Delay (ps)	166.39	157.73	212.84	152.38
Norm. Rise Delay Deviation	0.0331	0.0240	0.0339	0.0284
Mean Fall Delay (ps)	52.64	126.97	103.78	172.45
Norm. Fall Delay Deviation	0.0465	0.0188	0.0395	0.0303



(a)



(b)

Figure 5.19: PDF of rise delay for AOI_21 (a) and AOI_32 (b) gates implemented by using basic CMOS cells and as a single complex gate.

The implementation with basic gates was able to reduce the overall delay of an AOI_32 configuration and guaranteed more reliability for changes in transistors threshold voltages. It suggests that complex implementations presenting a larger number of series and parallel transistors in the cell topology may reduce the mean delay value at expense of increasing the performance variability. Circuit sizing was not considered for performance optimization, being all gates sized for similar drive strength.

In these last experiments by considering AOI logic gates, the results and analysis are similar to the ones discussed in the previous section. The mapped circuits, based on small cells, provide better performance in terms of delay variability. The values

presented in Table 5.5 suggest lower normalized delay deviations for ‘basic gates’ approach in both cases when the mean delay does not present the same tendency, as observed in the AOI_32 results. It reinforces the design guideline that suggests the use of small (basic) gates as preferential choice in the technology mapping task when parametric yield improvement is targeted.

5.7 Conclusion

Results obtained in this work so far about performance variability in CMOS logic gates submitted to transistor threshold voltage variation demonstrated the strong dependency it has on the gate topology, the number of stacked transistors, and the relative position of switching device in transistor network arrangements. Such analysis suggests the preferential use of basic CMOS gates instead of complex ones (AOI, for instance) in the technology mapping task of combinational circuits. Moreover, in terms of critical delay paths optimization, switching transistors placed close to the gate outputs are preferable for absolute delay propagation. Also, the results presented for the inverter gate showed that not only its performance, but also its DC characteristics are affected by variations in the threshold voltage of transistors. In a gate-level analysis it is also essential to study the condition of the input signal (input slope) and the load cell(s) since they also affects variability. Some changes in the transistor network can be done to reduce variability, like folding. The analysis performed in this work can be extended to other logic gates.

6 DELAY VARIABILITY ESTIMATION METHOD

Fluctuations of devices characteristics are very pronounced in deep submicron technology. As a consequence, it is necessary to take transistors characteristics variability into account in order to properly calculate the delay of a logic cell.

The present work proposes the implementation of a method that can predict the delay variability of a logic gate. It is performed here by assigning a unique variable – the threshold voltage variation – to each transistor in the analyzed logic gate. Whenever possible, the gate is divided into pull-up and pull-down networks, what reduces the number of devices and variables one must deal with.

Factorial designs (MYERS, 2002) are used to identify the main effects of the factors (ΔV_{TH}) on the primarily targeted variable, the transistors 'on-resistance'. Since each variable of interest is given by two levels (the coded values –1 and +1), each variant of such a design has 2^k experimental runs being called a 2^k factorial design, where 'k' is the number of transistors in the gate under test (i.e.,'k' threshold voltage values). As an example, Table 6.1 shows factorial values when a network with three transistors is considered. The values correspond to those used in the runs with HSPICE in order to find the resistances for each corner, with the difference that the threshold voltages in the netlist files have their natural (non-coded) values for performing the simulations.

Table 6.1: Combinations of *min* and *max* values considered for the threshold voltages variations (coded variables) of devices in a 3-transistors network .

Corners	ΔV_{th3}	ΔV_{th2}	ΔV_{th1}
1	-1	-1	-1
2	-1	-1	1
3	-1	1	-1
4	-1	1	1
5	1	-1	-1
6	1	-1	1
7	1	1	-1
8	1	1	1

Fig. 6.1 presents graphically the combinations of min and max coded values considered for the threshold voltages variations of devices in a 3-transistors network. Since in this example there are 3 transistors, the number of runs is 2^3.

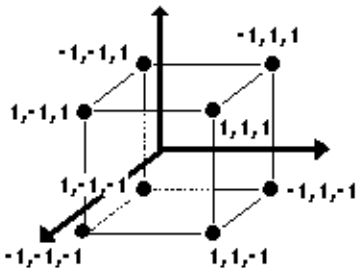


Figure 6.1: Combinations of minimum and maximum values considered for the threshold voltages variations (coded variables) of devices in a 3-transistors network .

Fig. 6.2 presents an overview of the delay variability methodology and the next section explain in details how the variables are incorporated into the model that provides an statistical analysis of the delay.

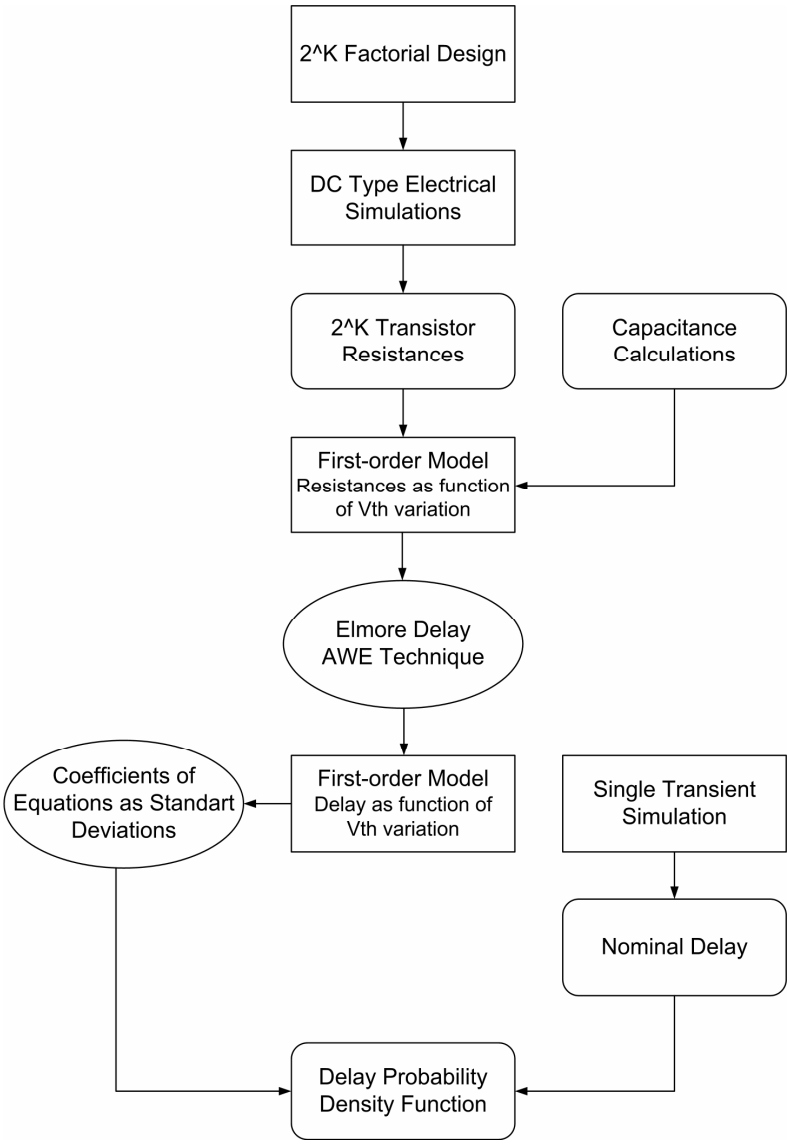


Figure 6.2: Gate delay variability estimation method flow .

Initially, DC type electrical simulations extract the 'on-resistances' of transistors for different threshold voltages (V_{TH}) in a corner-based principle. The V_{TH} of transistors are varied ±10% of their nominal values and transformed into coded (normalized) variables with values –1 and +1, which represent the minimum and maximum values assumed by V_{TH}, respectively.

The effect estimated in a 2^K factorial design is then converted into a regression model (first-order function for the resistance of the device) that can be used to analyze the response at any point in the space spanned by the factors (coded ΔV_{TH}) in the design. For the example presented in Table 6.1, the resistances are provided by 2^3 (three transistors) DC simulation with HSPICE, by dividing the voltage drop in each transistor by the current through it.

Linear regression technique is next used to develop approximate models for the resistances of transistors in different states as function of ΔV_{TH} of all the devices in the network. The dynamic response of a MOS is a function of the time. It takes into account the charging and discharging of parasitic capacitances that are intrinsic to the device, and the capacitances introduced by the interconnect lines and the load. The intrinsic capacitances originate from three sources: the basic MOS structure, the channel charge, and the depletion regions of the reversed-biased pn-junction of transistor drain and source regions (RABAEY, 2005). The intrinsic capacitances are calculated for specific technology nodes and are considered as constants in the proposed method. The delay is calculated through the RC constant of the network by using again factorial design. A first-order regression model represents the delay as a function of the coded ΔV_{TH}.

Two different methods have been used to calculate the delay variation for the transistors networks, the Elmore Delay model (ELMORE, 1948) and the Asymptotic Waveform Evaluation technique (AWE) (PILLAGE, 1990). The resistance equations obtained in the first step are used to perform the following analysis. The threshold voltages variations of the transistors are considered as random variables represented by probability density functions (PDF) with mean values equal to zero and normalized standard deviations [N(0,1)]. Since the V_{TH} variations are treated as coded variables, the coeficients of the final delay equation is considered as the standard deviation (σ) resulting from each V_{TH} variation. It is then performed a sum of normal distributions, and the result is a PDF that represents the delay of the network with the two aspects considered: mean and standard deviation (square root of variance). The normalized standard deviation (standard deviation divided by the mean) of the delay is the focus of the analysis of different networks, because it makes possible to compare the variability of arrangements with different mean delays.

6.1 On-Resistance

Qualitatively, a MOS transistor can be modeled as a switch with a finite on-resistance R_{ON} (RABAEY, 2005). When the voltage between the gate and the source (V_{GS}) is lower than the threshold voltage (V_{TH}), the switch is considered open. When V_{GS} is higher than V_{TH}, the transistor behaves as a finite resistance. This resistance is not constant and changes according to the region of operation of the transistors, which also depends on the difference of potential between drain and source (V_{DS}). The on-resistance of an MOS transistor depends upon its operation point and varies during the switching transient. Though the use of a linear constant resistance may introduce some error in the performance calculations, the use of a fixed R_{ON} might be a reasonable

approach, since the model is aiming at the delay variability and not at this absolut value. A transistor that presents a transient input signal operates in different regions as the signal reaches different values. Then, R_{ON} can be approximated to the average value of the resistance in different points over the operation region of interest.

The signal propagation delay can be analyzed by calculating the RC time constant of the network. The transistors are replaced by their on-resistances (R) and the fanout and intrinsic capacitances are calculated or extracted (C). Though this approach has been used to predict the delay of a logic gate, it is not know whether it shows good results for variability or not. The intention here is to extend this delay calculation technique to a delay variability calculation technique.

In this work, the resistances are first calculated in points mainly situated in the linear region of transistors operation. Later, an analysis considering points mainly in the saturated region was performed and the results were compared and discussed. The R_{ON} calculations in different regions and their further application in the method is useful in the sense that it shows whether the choice of the points for calculating the transistor resistance is crucial or not for the reliability of the proposed method. As already mentioned in CHAPTER 4, the resistance of each transistor in a network is calculated as a function of the threshold voltages variations of all transistors. Changes is the threshold voltage of a device can affect more or less the resistance of another device, depending on its position in the arrangement.

Electrical timing simulations performed with HSPICE furnished the resistances of transistors for different threshold voltages in a case-based methodology. The threshold voltages (V_{TH}) of the transistors were varied to –10% and 10% of their nominal values and transformed into coded variables with values of –1 and +1, which are assigned to the minimum and maximum values V_{th} can assume, respectively. The coded value for a nominal threshold voltage is 0. In each case - combinations of –1 and +1 for the V_{TH} of the devices in the network - the average of resistances for different drain-source voltages (V_{DS}) was taken. A lower gate voltage was applied to one or more transistors in series or parallel networks in order to simulate switching transistors. Linear regression technique was used to develop approximate models for the resistances of transistors in different topologies as functions of threshold voltages variations of all the devices in the network. The capacitances were approximated as to be independent on variations of threshold voltages and only the resistances were calculated as a function of these variations.

The procedure of fitting regression models to the responses of the simulation model evaluated at several points is known as Response Surface Methodology (RSM). This technique also includes optimization of the resulting regression function (MYERS, 2002).

6.1.1 Response Surface Methodology

Monte Carlo circuit simulations present valuable results but it is a computationally prohibitive methodology. As an alternative solution, Response Surface Methodology (RSM) expands the circuit performance around nominal process values providing response surface models (CAO, 2005).

RSM explores the relation between the independent (input) variables and one or more response variables, represented, in this work, by the performance metrics: rise and fall delays of the logic gates. The form of the true response function **f** is unknown and sometimes very complicated, so it is necessary to approximate it (MYERS, 2002). The relation between the response **y** and the independent variables $\xi_1, \xi_2,..., \xi_k$ is:

$$y = f(\xi_1, \xi_2,..., \xi_k) + \varepsilon \tag{6.1}$$

where ε is the term that represents sources of variability not accounted for in the function **f**, such as measurement errors, background noise etc. The variables $\xi_1, \xi_2,..., \xi_k$ are called the "natural variables" because they are expressed in the natural units of measurements, such as degrees Celsius, grams per liter etc. That can be convenient to transform the natural variables into "coded variables" $x_1, x_2,..., x_k$, which are usually defined to be dimensionless with mean zero and the same standard deviation. By using coded variables, the response function will be written as:

$$\eta = f(x_1, x_2,..., x_k) \tag{6.2}$$

The order of the model that will represent the response depends on the region of the independent variable space over which the true response surface is approximated. For the case of three independent variables (there is no interaction between them), as presented before, the first-order model in terms of the coded variable is:

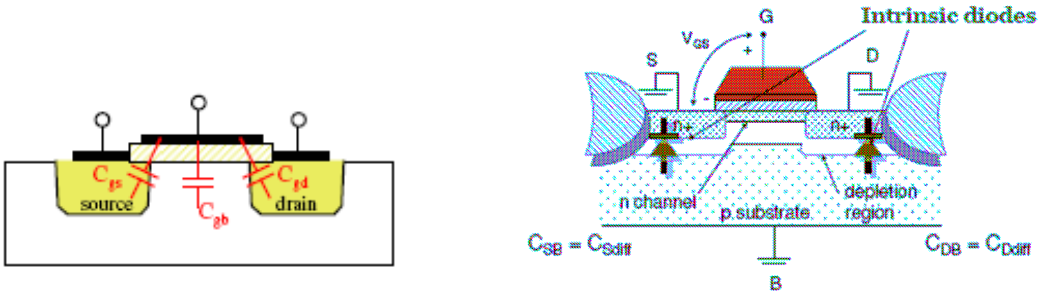
$$\eta = \beta_0 + \beta_1.x_1 + \beta_2.x_2 + \beta_3.x_3 \tag{6.3}$$

where the β's are the coefficients resulting from the fitting. This first-order model is sometimes called a main effects model, since it includes only the main effects of the variables.

(OKADA, 2003) uses RSM to propose a model that characterizes a statistical gate delay variation. Coefficients of RSM are derived from several SPICE simulations by using a least square method.

6.2 MOS Structures Capacitances

The dynamic response of a MOS is a function of the time it takes to charge and discharge the parasitic capacitances that are intrinsic to the device, and the extra capacitances introduced by the interconnect lines and the load. The intrinsic capacitances originate from three sources: the basic MOS structure, the channel charge, and the depletion regions of the reversed-biased *pn*-junctions of drain and source (RABAEY, 2005), represented by the intrinsic diodes in Fig. 6.3b.



(a) (b)

Figure 6.3: Intrinsic capacitances of a MOS transistor.

6.2.1 Gate Capacitance

6.2.1.1 Channel Capacitance

The gate of the MOS transistor is isolated from the conducting channel by the gate oxide and its capacitance per unit area is equal to:

$$C_{ox} = \frac{\varepsilon_{ox}}{t_{oxn}} \tag{6.4}$$

The total channel capacitance is calculated by multiplying the value above and the gate area:

$$C_{gn} = \left(\frac{\varepsilon_{ox}}{t_{oxn}}\right) * W_n * L_n \tag{6.5}$$

in which W_n is the width and L_n is the length of the channel for a NMOS transistor.

$$C_{gp} = \left(\frac{\varepsilon_{ox}}{t_{oxp}}\right) * W_p * L_p \tag{6.6}$$

in which W_p is the width and L_p is the length of the channel for a PMOS transistor.

The gate-to-channel capacitance (C_{GC}) depends upon the operation region and terminal voltages of the transistor. It is divided into three components: gate-to-source (C_{GCS}), gate-to-drain (C_{GCD}) and gate-to-body (C_{GCB}) capacitances. In the cut-off region, no channel is formed and the total gate-to-channel capacitance appears between gate and body:

$$C_{GCB} = \left(\frac{\varepsilon_{ox}}{t_{oxn}}\right) * W_n * L_n \tag{6.7}$$

In the resistive (linear) region, the gate capacitance distributes evenly between source and drain, as the body electrode is shielded from the gate by the channel:

$$C_{GCS} = C_{GCD} = \left(\frac{1}{2}\right) * \left(\frac{\varepsilon_{ox}}{t_{oxn}}\right) * W_n * L_n \tag{6.8}$$

In the saturation mode, the capacitance between gate and drain and the gate-body capacitance are approximately zero. All the capacitance is therefore between gate and source:

$$C_{GCS} = (\frac{2}{3}) * (\frac{\varepsilon_{ox}}{t_{oxn}}) * W_n * L_n \tag{6.9}$$

6.2.1.2 Overlap Capacitance

In reality, source and drain diffusion are not entirely performed along only one direction inside the semiconductor and tend to extend a little below the oxide by an amount x_d, what characterizes a lateral diffusion of the ions implanted. It gives rise to a parasitic capacitance between gate and source (drain) that is called overlap capacitance and it has a fixed value (per unit area). These capacitances are given by:

$$C_{gdn} = C_{gdo} * W_n \quad \text{for the NMOS transistor;} \tag{6.10}$$

$$C_{gdp} = C_{gdo} * W_p \quad \text{for the PMOS transistor;} \tag{6.11}$$

in which C_{gdo} is the gate-drain capacitance per unit area. Fig. 6.3 shows these capacitances as C_{gs} and C_{gd}.

The overlap capacitances between gate and source are calculated in the same way as above, changing C_{gdo} by C_{gso}. In the proposed method, the source diffusion capacitances of the transistors connected to the power lines are not included in the calculations, since they are considered permanently charged or discharged.

6.2.2 Junction Capacitances

A capacitive component is contributed by the reverse-biased source-bulk and drain-bulk *pn*-junctions.

6.2.2.1 Bottom-Plate Junction Capacitance

This capacitance appears because of the depletion layer between the drain region (with doping N_D for NMOS and N_A for PMOS) and the substrate with doping N_A (NMOS) or N_D (PMOS). The total depletion region capacitance for this component equals:

$$C_{dbn} = C_{jn} * W_n * L_{dn} \quad \text{for the NMOS transistor;} \tag{6.12}$$

where C_{jn} is the junction bottom capacitance per unit area for the NMOS transistor and L_{dn} is the drain region length.

$$C_{jn} = \frac{C_{jo}}{(1 - (\frac{Vbs_n}{\varphi_j}))^{MJ}} \tag{6.13}$$

where C_{jo} represents the zero-bias junction capacitance per unit area and MJ is a grading coefficient. φ_j represents the difference in the internal chemical potentials between the *n* and *p* sides of the junction.

$$\varphi_j = \frac{K*T}{q} * \ln\left(\frac{N_A * N_D}{n_i^2}\right) \tag{6.14}$$

where n_i represents the intrinsic carrier concentration.

$$C_{dbp} = C_{jp} * W_p * L_{dp} \qquad \text{for the PMOS transistor;} \tag{6.15}$$

where C_{jp} is the junction bottom capacitance per unit area for the PMOS transistor

$$C_{jp} = \frac{C_{jo}}{\left(1+\left(\frac{Vbs_p}{\varphi_j}\right)\right)^{MJ}} \tag{6.16}$$

6.2.2.2 Side-Wall Junction Capacitance

This capacitance is counted for the sides of drain and source region, but not for the side where the conductive channel is placed.

$$C_{swdn} = C_{sjwn} * (2 * L_{dn} + W_n) \tag{6.17}$$

$$C_{swdp} = C_{sjwp} * (2 * L_{dp} + W_p) \tag{6.18}$$

6.2.3 The Inverter

The inverter is the nucleus of digital designs and once its operation and properties are understood, it gets simpler to design more intricate structures. Fig. 6.4 presents the eletrical diagram of a inverter and Fig. 6.5 shows the diagram of a CMOS inverter modeled as an RC circuit.

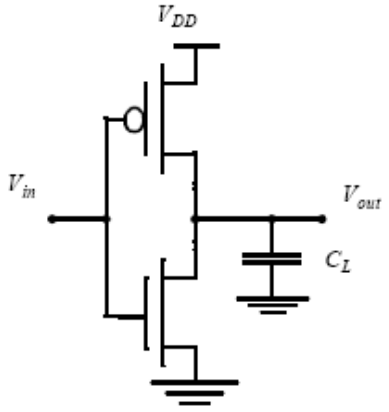


Figure 6.4: CMOS Inverter (RABAEY, 2005).

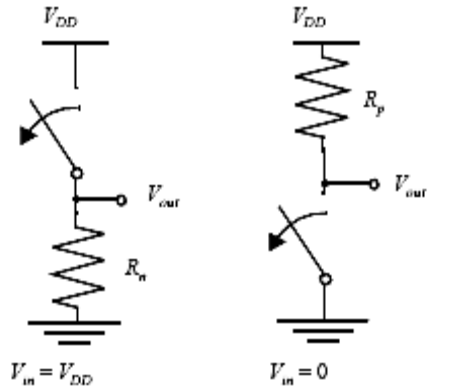


Figure 6.5: Switch models of CMOS inverter (RABAEY, 2005).

When the voltage applied to the gates of the transistors (the same voltage to both gates at the same time) is high, the NMOS transistor is on, while the PMOS is off, and a current path exits between V_{out} (Fig. 6.5) and the ground node, resulting in a steady-state value of 0 V. When the input (gate) voltage is low, the NMOS transistor is off and the PMOS transistor is on. A path exists between the V_{DD} source and the output node, yielding a high output voltage.

The inverter model used to calculate the propagation delay assumes that all capacitances are lumped together in a so called load capacitor C_L, placed between the output node and ground. So, the total capacitance considered for each case of conduction is:

$$C_n = C_{gn} + C_{gp} + C_{gdn} + C_{gdp} + C_{dbn} + C_{dbp} + C_{swdn} + C_{swdp} + C_{gcdn} \quad (6.19)$$

$$C_p = C_{gp} + C_{gn} + C_{gdn} + C_{gdp} + C_{dbn} + C_{dbp} + C_{swdn} + C_{swdp} + C_{gcdp} \quad (6.20)$$

The gate capacitances C_{gn} and C_{gp} considered in the calculations are related to a fanout load represented by a inverter with a drive strentgh five-times higher than the driver, and C_{gcdn} and C_{gcdp} refers to the gate-to-channel capacitance of the transistor that is conducting.

For the case of this simple design, the delay values of the device by using its resistances and capacitances are:

$$Delay_{rise} = R_P * C_p \quad (6.21)$$

$$Delay_{fall} = R_N * C_N \quad (6.22)$$

6.3 Modeling the Falling-Edge Delay Deviation of a NAND3

The calculation of the falling-edge delay deviation of a 3-input NAND is presented in this section for a better understanding of the proposed method. In this case, only the pull-down network of the logic gate is considered and can be represented as in Fig. 6.6.

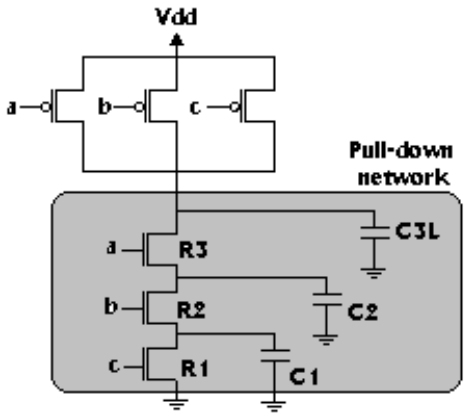


Figure 6.6: Pull-down network of a 3-input NAND.

The resistances are provided by 2^3 (3 transistors) DC simulation with HSPICE, by dividing the voltage drop in each transistor by the current through it. This is done for 2^3 combinations of natural (non-coded) values of threshold voltages of the transistors. Each transistor is given 8 resistances values which are used in a linear regression to provide the resistance equations (functions of the threshold voltages variations), as seen in Appendix B. The capacitances are explained and the delay calculations are shown in the script of Appendix C. This script is written to be run in MATLAB® and provides a combination of 8 delay values, that are then used to furnish the delay as a function of the threshold voltages variations (Appendix D). Since the V_{TH} variations are treated as coded variables (-1,1), the coeficients of the final delay equation is considered the standard deviation (σ) resulting from each V_{TH} variation. The final delay equation has the following form:

$$T(x, y, z) = \mu + a*x + b*y + c*z \tag{6.23}$$

where *x*, *y* and *z* represents the standard deviation of the threshold voltages of transistors and *a*, *b* and *c* are the coefficients of the fitted equations.

The final standard delay deviation (σ_D) is then:

$$\sigma_D = \sqrt{a^2 + b^2 + c^2} \tag{6.24}$$

The normalized standard delay deviation (σ_D/μ_D) makes it possible a fair comparison between logic gates that present different mean delay values. The mean value of the modeled delay PDF is provided by a single transient simulation with nominal values of threshold voltages.

7 MODELING THE DELAY VARIABILITY OF TRANSISTOR NETWORKS

This chapter presents the results achieved by using the proposed semi-empirical method to calculate the delay deviation of different transistors networks. DC electrical simulations were performed with HSPICE. The other calculations were performed with the MATLAB® (Matrix Laboratory) program.

7.1 Calculation of Resistances

In CHAPTER 6 it has been already explained how the resistances of transistors were calculated. This section presents an analysis that is able to justify why each transistor in a network must be modeled with different PDFs for its resistances, unless the network is purely parallel. Fig. 7.1 shows a NMOS stacking and Fig. 7.2 presents the PDFs for each transistor in the series arrangement.

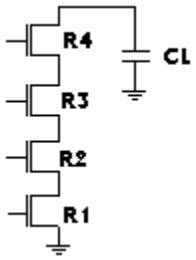


Figure 7.1: NMOS transistors stacking.

It is quite clear that each transistor has a different resistance mean value and standard deviation. The variability presented by the resistance of the device that is close to the output node (far from the ground source) is much higher than the variability of those which are respectively closer to the ground source. In this sense, the modeling of one single transistor as a resistance and the use of this model for the other transistors in the network implies in a source of error for the delay deviation calculus.

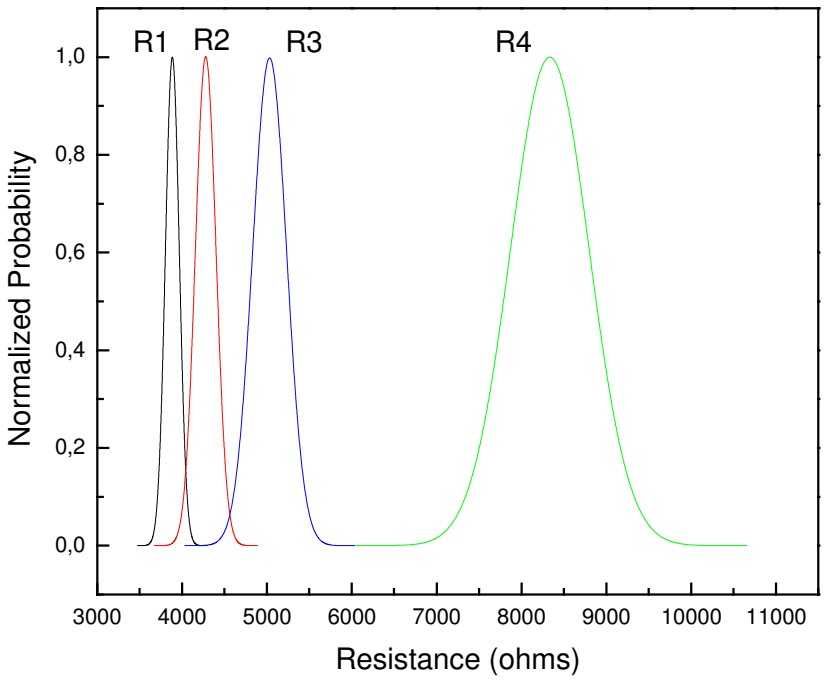


Figure 7.2: Probability density functions (PDF) of the resistances that constitute models for the transistors in a NMOS series network.

7.2 Resistance of Parallel Transistors

The transistors in a parallel network present equal independent terms and equal coeficients in the resistance equations, since they are submited to the same drain-source and gate-source voltages. Table 7.1 presents resistance equations for N- and PMOS parallel transistors with channel width equals to 90nm. Variables x_1, x_2, x_3 and x_4 represent the threshold voltage variations of the respective transistors and are coded with values of -1 or $+1$.

Table 7.1: Approximate resistance equations for transistors in a parallel network.

Number of Transistors	*Approximate Resistance Equations*	
	NMOS	*PMOS*
04	$\mathbf{R_1} = 6568.5 + 672.4\,\mathbf{x_1}$	$\mathbf{R_1} = 24222 + 1882\,\mathbf{x_1}$
	$\mathbf{R_2} = 6568.5 + 672.4\,\mathbf{x_2}$	$\mathbf{R_2} = 24222 + 1882\,\mathbf{x_2}$
	$\mathbf{R_3} = 6568.5 + 672.4\,\mathbf{x_3}$	$\mathbf{R_3} = 24222 + 1882\,\mathbf{x_3}$
	$\mathbf{R_4} = 6568.5 + 672.4\,\mathbf{x_4}$	$\mathbf{R_4} = 24222 + 1882\,\mathbf{x_4}$

The resistance (R) of a transistor is not dependent on threshold voltages of other transistors in the parallel network and the equations are the same for topologies with

different number of devices. The method uses the voltage drop over the transistor and divide it by the flowing current to find out the resistance. Once this voltage drop does not depend on the voltage in the other transistors nodes, the resistance is independent on the threshold voltage variations of these devices.

7.3 Resistance of Series Transistors

On the other hand, the resistances of transistors in series networks also depend on the threshold voltage of others transistors in the stacking, especially on the V_{th} of the transistor that is closer to the output node. Variations in V_{TH} of this transistor might impact the performance of the network a little bit more than variations in other transistors. Table 7.2 shows how each transistor resistance is a function of the threshold voltages of all transistors in the stacking. Figure 7.1 shows a 4-NMOS stacking configuration with each device labeled as resistances (R_1...R_4).

Table 7.2 shows the resistances (R) of devices are numbered from 1 to 4, as in Fig. 7.1. The higher the number, the closer it is to the output terminal. Variables x_1, x_2, x_3 and x_4 represent the threshold voltage of the transistors, but in an inverse order: x_1 is the coded threshold voltage variation of R_4, x_2 corresponds to the variation in R_3 and so on.

Table 7.2: Approximate resistance equations for NMOS transistors in series network.

Number of Transistors	*Approximate Resistance Equations*
	NMOS Series Network
04	$R_1 = 3885.2 - 20.3\mathbf{x_1} - 10.7\mathbf{x_2} - 7.6\mathbf{x_3} + 263.4\mathbf{x_4}$
	$R_2 = 4278.5 - 60.0\mathbf{x_1} - 31.5\mathbf{x_2} + 357.3\mathbf{x_3} + 16.8\mathbf{x_4}$
	$R_3 = 5032.4 - 198.7\mathbf{x_1} + 564.3\mathbf{x_2} + 27.0\mathbf{x_3} + 21.2\mathbf{x_4}$
	$R_4 = 8333.7 + 1391.0\mathbf{x_1} + 61.2\mathbf{x_2} + 42.4\mathbf{x_3} + 33.1\mathbf{x_4}$

In a NMOS stacking, the transistor closer to the output node has higher resistance than the others, when all devices are considered “ON”. Though the input gate voltage is at the same level for all the devices, the gate-source voltage is not. When the stacking first turns on, the devices have their sources at $V_{DD} - V_{TH}$, except for the device in the bottom of the stacking, that has its source grounded. The source terminals of the top-of-stacking devices spend more time near $V_{DD} - V_{TH}$ and less time conducting strongly than the other devices. That leads to a higher effective V_{TH} due to the body effect what results in higher effective resistance. That might also be related to the reason why the dependence of R on the respective threshold voltage increases as the transistor gets closer to the output.

Also for a PMOS stacking, the transistor close the output has higher resistance than the others, when all devices are considered “ON”. Also, the resistance of the devices are

very dependent on the threshold voltage of the transistor that is far from the output node as in Table 7.3.

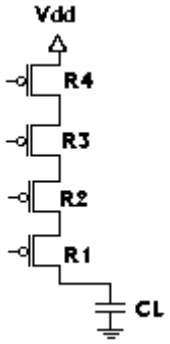


Figure 7.3: PMOS transistors stacking.

Variable x_1 is the coded threshold voltage variation of R_4, x_2 corresponds to the variation in R_3 and so on.

Table 7.3: Approximate resistance equations for PMOS transistors in series network.

Number of Transistors	*Approximate Resistance Equations*
	PMOS Series Network
04	$\mathbf{R_1} = 28434.75 + 40.75\mathbf{x_1} + 45\mathbf{x_2} + 50.875\mathbf{x_3} + 2701.75\mathbf{x_4}$
	$\mathbf{R_2} = 24132 + 75.75\mathbf{x_1} + 83.5\mathbf{x_2} + 2069\mathbf{x_3} - 212.875\mathbf{x_4}$
	$\mathbf{R_3} = 21537.625 + 85.5\mathbf{x_1} + 1687.25\mathbf{x_2} - 75.125\mathbf{x_3} - 88.625\mathbf{x_4}$
	$\mathbf{R_4} = 19748.125 + 1431.125\mathbf{x_1} - 19.875\mathbf{x_2} - 22.5\mathbf{x_3} - 26.375\mathbf{x_4}$

Although PMOS transistors can strongly pass “1”, the source of devices far from the V_{DD} source spend more time in a voltage that has an absolute value lower than V_{DD}, since transistors are not ideal switches and present some voltage drop due to their resistances. Due to the lower absolute gate-source voltage, these devices spend less time conducting strongly than the other devices, what results in higher effective resistance.

7.4 Estimation of Performance Deviation

Elmore Delay model (ELMORE, 1948) and Asymptotic Waveform Evaluation (AWE) (SAPATNEKAR, 2004) were used in the proposed model to calculate the delay variation for the transistors networks. The threshold voltages of the transistors that appear in the resistances equations (x_1, x_2, x_3 and x_4) were considered as random variables represented by probability density functions (PDF) with mean values equal to zero and normalized standard deviations [N(0,1)].

7.4.1 Series Networks

The use of Elmore Delay model (ELMORE, 1948) allows us to perform a straightfoward sum operation of PDFs in order to calculate the equivalent resistances of the transistors networks, and then the time constant RC. The Asymptotic Waveform Evaluation method requires some other steps before providing a final delay PDF. As explained in SECTION 3.5.4, AWE requires the calculations of the conductance and capacitances matrices by using the equations of the considered circuit. Then the moments are found and matched via Padé approximation resulting in reduced-order function models. In both methods, it is necessary to calculate the delay for the corner values of x_1, x_2, x_3 and x_4, and fit an equation.

The equations obtained by the estimation method, as those in Table 7.2 and 7.3, are used to perform the analysis described. The procedure is applied only for the resistances of transistors, since no variations are considered for the capacitances in this work. The result is a PDF that represents the delay of the network with the two moments considered: mean and standard deviation (square root of variance). The normalized standard deviation (standard deviation divided by the mean) of the delay is the object of analysis of different networks, because it makes possible to compare the variability of arrangements with different mean delays.

Table 7.4 presents the normalized delay deviation for NMOS transistors stackings with 2, 3 and 4 devices (stack-02, stack-03 and stack-04 respectively), according to the position of the switching transistor in relation to the output node. Position 1 is the closest to the output terminal and position 4 is the farthest one.

Table 7.4: Delay deviation for NMOS transistors in series network, according to the position of the switching transistor in relation to the output node (1-close...4-far).

Position of Switching Transistor	*Stack-02*			*Stack-03*			*Stack-04*		
	Model Elmore	*Model (AWE)*	*Simulation*	*Model Elmore*	*Model (AWE)*	*Simulation*	*Model Elmore*	*Model (AWE)*	*Simulation*
1	**0.0520**	0.0224	0.0472	**0.0472**	0.0258	0.0434	**0.0438**	0.0293	0.0423
2	**0.0391**	0.0366	0.0451	**0.0386**	0.0277	0.0381	**0.0377**	0.0246	0.0346
3				0.0272	**0.0318**	0.0361	0.0268	**0.0273**	0.0311
4							0.0217	**0.0284**	0.0299

The methodology used to find the empirical equations shows that the presence of a switching transistor in a network causes it to be more sensitive to threshold voltages variations. The method proposed by using AWE provides results with an average error of 25.3%, maximum error equals to 52.5% and minimum error of 5.0% when compared to the simulated values. It can predict better delay deviations for transitions in devices far from the output than for transitions in devices close to output. The number of transistors in the stacking does not influence much the delay deviation of the network,

but it can be noticed a little reduction in the metric when more transistors are present. That is similar for the modeled and the simulated situation, except for the "close switching".

Though the results furnished by the method are reasonable, they do not present the same tendency as the Monte Carlo simulations when the position of the switching device changes for the AWE technique. According to the Monte Carlo simulations performed, a "close switching " causes higher delay deviation independently on the number of transistors in the stack. The opposite is presented by the estimation method, according to which a "close switching " causes lower delay deviation independently on the number of transistors in the stack.

The proposed model by using Elmore Delay technique provides results with an average error of 12.4%, maximum error equals to 27.4% and minimum error of 1.3%. Differently from the results provided by the use of the AWE technique, it presents the same tendency as the Monte Carlo simulations, in which a "close switching " causes higher delay deviation independently on the number of transistors in the stack. In agreement with the Monte Carlo results, as the number of transistors in the stack decreases, the delay deviation of the network increases.

Table 7.5 presents the normalized delay deviation for PMOS transistors stackings with 2, 3 and 4 devices (stack-02, stack-03 and stack 04 respectively), according to the position of the switching transistor in relation to the output node.

Table 7.5: Delay deviation for PMOS transistors in series network, according to the position of the switching transistor in relation to the output node (1-close...4-far).

Position of Switching Transistor	*Stack-02*			*Stack-03*			*Stack-04*		
	Model Elmore	*Model (AWE)*	*Simulation*	*Model Elmore*	*Model (AWE)*	*Simulation*	*Model Elmore*	*Model (AWE)*	*Simulation*
1	**0.0264**	0.0210	0.0382	**0.0218**	0.0216	0.0379	0.0188	**0.0210**	0.0403
2	**0.0238**	0.0218	0.0334	0.0197	**0.0198**	0.0301	**0.0172**	0.0155	0.0297
3				**0.0188**	0.0176	0.0281	**0.0163**	0.0159	0.0261
4							0.0160	**0.0173**	0.0249

The model proposed by using AWE provides results with an average error of 39.9%, maximum error equals to 47.9% and minimum error of 30.5%. The number of transistors in the stacking influences the delay deviation of the network a little bit more than when NMOS are used. It can also be observed a little reduction in the normalized deviation when more transistors are present for simulated and modeled results, but still not for the case of a "close switching". According to statistical simulations performed, a "close switching " causes higher delay deviation independently on the number of transistors in the stacking. The same situation is presented by the estimation method,

according to which a “close switching ” causes higher delay deviation for the cases with 3 and 4 transistors in the stacking.

The model proposed by using Elmore Delay technique provides results with an average error of 38.3%, maximum error equals to 53.4% and minimum error of 30.9%. The delay deviation is practically constant for a certain number of transistors in the network independently on the position of the switching transistor, but it decreases as the number of transistors increases.

7.4.2 Parallel Networks

Table 7.6 presents the normalized delay deviation for NMOS transistors parallel networks with 2, 3 and 4 devices (parallel-02, parallel-03 and parallel-04 respectively). The metrics are presented considering 1, 2 or 3 transistors switching and the other(s) turned off. The equivalent resistance of the parallel arrangement is applied to get the RC time constant.

Table 7.6: Delay deviation for NMOS transistors in parallel networks, according to the number of switching transistors.

Number of Switching Transistors	*Parallel-02*		*Parallel -03*		*Parallel -04*	
	Method	*Simulation*	*Method*	*Simulation*	*Method*	*Simulation*
1	0.0560	0.0545	0.0560	0.0519	0.0560	0.0505
2	0.0431	0.0575	0.0431	0.0536	0.0431	0.0503
3			0.0351	0.0619	0.0351	0.0578

Statistical Monte Carlo simulations presented higher delay deviation for a parallel transistor network then for a series network. The results provided by the proposed method also determined so. Transistors that are not conducting have little influence on the delay deviation of the arrangement and the model actually presents no influence of these devices. The method provides results with an average error of 20.4%, maximum error equals to 39.3% and minimum error of 2.8%.

The higher the number of switching devices, the higher the delay deviation, according to the simulations. However, the model does not handle the impact of multiple switchings on delay deviation, since it presents lower delay deviation as the number of switching transistors increases.

On the other hand, the model shows that when all the transistors are conducting, the higher the number of devices in the network, the lower the delay deviation. That comes in agreement with what was observed in the simulations results: for the same number of switching transistors, the fewer the number of devices, the higher the deviation.

Table 7.7 presents the normalized delay deviation for PMOS transistors parallel networks with 2, 3 and 4 devices (parallel-02, parallel-03 and parallel-04 respectively).

Table 7.7: Delay deviation for PMOS transistors in parallel networks, according to the number of switching transistors.

Number of Switching Transistors	*Parallel-02*		*Parallel -03*		*Parallel -04*	
	Method	*Simulation*	*Method*	*Simulation*	*Method*	*Simulation*
1	0.0361	0.0426	0.0361	0.0429	0.0361	0.0430
2	0.0257	0.0279	0.0257	0.0278	0.0257	0.0276
3			0.0209	0.0218	0.0209	0.0213

In the case of PMOS transistors networks, the model agrees with the simulation results when it comes to how the number of switching devices impacts the delay deviation: the higher the number of switching transistors, the lower the delay deviation. Also, it can be said that both results point to no influence of the turned-off devices on the this metric. The proposed method provides results with an average error of 7.9%, maximum error equals to 15.9% and minimum error of 1.9%.

7.4.3 Delay Variability Modeled by Considering the Saturated Region of Operation

As discussed in chapter 6, the on-resistance of an MOS transistor depends upon its operation point and varies during the switching transient. The results achieved so far with the model were based in on-resistances calculated in the linear region of operation, once the DC voltage applied to the devices was set to small values, and so the drain-source voltage of the transistors. This section presents the delay deviations calculated with on-resistances of transistors operating in the saturation region with the proposal of evaluating whether a change in the on-resistances compromises the reliability of the model or not. If so, it is important to determine the best way of calculating the resistances to be applied in the performance analysis.

Table 7.8 presents the normalized delay deviation for NMOS and PMOS transistors stackings with 4 devices (stack-04), according to the position of the switching transistor in relation to the output node. The results are provided by modeling the on-resistances in the linear and in the saturation regions.

Table 7.8: Delay deviation for N- and PMOS transistors in series networks, according to the position of the switching transistor in relation to the output node (1-close...4-far) for the linear and the saturation regions of operation.

Position of Switching Transistor	*NMOS*					*PMOS*				
	Simulation	*Linear Region*		*Saturation Region*		*Simulation*	*Linear Region*		*Saturation Region*	
		Model (AWE)	*Model Elmore*	*Model (AWE)*	*Model Elmore*		*Model (AWE)*	*Model Elmore*	*Model (AWE)*	*Model Elmore*
1	0.0423	0.0293	**0.0438**	0.0171	0.0467	0.0403	**0.0210**	0.0188	0.0155	0.0141
2	0.0346	0.0246	**0.0377**	0.0261	0.0451	0.0297	0.0155	**0.0172**	0.0124	0.0123
3	0.0311	0.0273	0.0268	0.0295	**0.0312**	0.0261	0.0159	**0.0163**	0.0128	0.0116
4	0.0299	**0.0284**	0.0217	0.0278	0.0250	0.0249	**0.0173**	0.0160	0.0133	0.0113

The bold-faced values represent the best results achieved for the delay deviation by using AWE or Elmore Delay technique for each type of transistors (N- or PMOS). By "best results" one understand those between the methods using two techniques which get closer to the results provided by simulations. In the case of a NMOS stacking, the method applied by using resistances calculated in the linear region of the transistor operation provides results that are mostly closer to the simulated values. In the case of a PMOS stacking, the estimated results are better when the resistances are extracted in the linear region for all the positions considered for the switching device. In order to get a more reliable conclusion on which region of operation should be used the calculus of the resistances, it is interesting to investigate the delay deviation of another type of transistor network in relation to the region of operation used to calculate the on-resistances.

Table 7.9 presents the normalized delay deviation for N- and PMOS parallel transistors networks with 4 devices (Parallel-04), according to the amount of switching transistors. The results are provided by modeling the on-resistances in the linear and in the saturation regions.

Table 7.9: Delay deviation for N- and PMOS transistors in parallel networks, according to the number of switching transistor for the linear and the saturation region of operation.

Position of Switching Transistor	*NMOS*			*PMOS*		
	Simulation	*Method*		*Simulation*	*Method*	
		Linear Region	*Saturation Region*		*Linear Region*	*Saturation Region*
1	0.0505	**0.0560**	0.0605	0.0430	**0.0414**	0.0251
2	0.0503	0.0431*	**0.0435**	0.0276	**0.0257**	0.0178
3	0.0578	0.0351*	**0.0354**	0.0213	**0.0209**	0.0146

The results achieved by using the resistances calculated with points in the saturation region are worse than those achieved with points in the linear region. Actually, this region present much worst results for the PMOS transistors networks. By analyzing the delay deviation values provided, it was found a better choice to stick to the linear region in order to calculate the on-resistances of the transistors.

7.4.4 Estimation Method Applied to Different Inverter Topologies

As already discussed, a semi-empirical method was developed in order to analyze delay variability of different transistor networks. This method was applied to the inverter topologies showed in Fig. 5.13 and the results are presented in Table 7.10. The total area for the N- or PMOS network is kept constant for the configurations presented ($W_{N\ Total}$ = 180nm and $W_{P\ Total}$ = 680 nm) and the fanout (output load) consists of an inverter with W_N = 225nm and W_P = 850 nm.

Table 7.10: Rise and fall delay deviations for different inverter topologies.

Inverter Topology	*Timing*			
	Rise Delay Deviation		*Fall Delay Deviation*	
	Simulation	*Method*	*Simulation*	*Method*
(a)	0.0415	0.0359	0.0852	0.0595
(b)	0.0319	0.0260	0.0956	0.0448
(c)	0.0283	0.0255	0.0355	0.0522

The method presented delay variability values in agreement to the simulated results when it concerns to the the topology less affected by V_{TH} variations for the rising-edge

delay. The falling-edge delay deviation achieved with the method implies in less variability for the series topology, while the simulated results points at the folded (parallel) topology as the most immune to variations in V_{TH}. By considering both delays, the method establishes the series configuration for the inverter as the most robust, while the simulation shows that the folded topology is the best choice.

However, the measurements and calculations that provided the results in Table 7.10 do not take into account that, according to Pelgrom model, the larger the size of the transistor, the lower its threshold voltage deviation. The same deviation ($3\sigma = 10\%$) was applied to the three sizes of inverters analyzed. In order to investigate the influence of the area on the standard deviation of V_{TH} and consequently on the delay of the logic gate, different variations were considered for different sizes by applying the relation between the parameter variation and the area:

$$\sigma^2(\Delta P) \quad \alpha \quad \frac{1}{W.L}$$

as already demonstrated by equation (3.17).

Table 7.11 shows delay deviation results for the same inverter topologies presented in Table 7.10), but with different V_{TH} standard deviations. The inverter named *(a)* was given a deviation as before ($3\sigma = 10\%$). The inverter named *(b)* was given a deviation of $3\sigma = 7.07\%$ and the inverter *(c)* a deviation of $3\sigma = 14.1\%$ with the fanout consists of an inverter with $W_N = 225$nm and $W_P = 450$ nm.

.

Table 7.11: Rise and fall delay deviations for different inverter topologies with different threshold voltage variations.

Inverter Topology	*Timing*			
	Rise Delay Deviation		*Fall Delay Deviation*	
	Method	*Simulation*	*Method*	*Simulation*
(a)	0.0359	0.0445	0.0595	0.0941
(b)	0.0201	0.0223	0.0321	0.0656
(c)	0.0363	0.0431	0.0618	0.0834

7.4.5 The Influence of the Sizing of Transistors on Delay Variability

The second section of CHAPTER 5 has already discussed the influence of the dimensions of the transistors on the delay variability of an inverter. Table 7.12 shows the results provided by the proposed method for different drive strengths while keeping a constant ratio of widths (W_P/W_N). The transistors widths used for the inverters were: *(1)* W_N=0.045μ and W_P=0.170μ, *(2)* W_N=0.090μ and W_P=0.340μ and *(3)* W_N=0.180μ and W_P=0.680μ. The threshold voltage variation range is the same for all the measurements ($3\sigma = 10\%$ variation).

Table 7.12: Rise and fall delay deviations for different sizes of inverters.

Inverter Sizes	*Rise Delay Deviation*		*Fall Delay Deviation*	
	Method (Elmore)	*Simulation*	*Method (Elmore)*	*Simulation*
(1)	0.0360	0.0407	0.0350	0.0415
(2)	0.0359	0.0372	0.0600	0.0556
(3)	0.0359	0.0415	0.0595	0.0852

The results provided by the simulations furnished similar rise delay deviations for differently sized inverters. This is also so for the method, which presented near equal results for the three sizes analyzed. The simulated and modeled results showed an increase in the fall delay deviation as the sizes of devices increases. The variability in the performance of the logic gate is more sensitive to the resizing of the NMOS than of the PMOS transistor.

Table 7.13 shows delay deviation results for the differently sized devices (the same as is Table 7.12), but with different V_{TH} standard deviation, as predict by Pelgrom model. The inverter named *(1)* was given a deviation as before ($3\sigma = 10\%$). The inverter named *(2)* was given a deviation of $3\sigma = 7.07\%$ and the inverter *(3)* a deviation of $3\sigma = 5\%$.

Table 7.13: Rise and fall delay deviations for different sizes of inverters with different threshold voltage variations.

Inverter Sizes	*Rise Delay Deviation*		*Fall Delay Deviation*	
	Method (Elmore)	*Simulation*	*Method (Elmore)*	*Simulation*
(1)	0.0360	0.0407	0.0350	0.0415
(2)	0.0254	0.0271	0.0425	0.0427
(3)	0.0180	0.0223	0.0298	0.0468

The method predicts lower delay variability as the size of the inverter increases and it is in accordance with the model when the rising-edge delay deviation is considered. However, the statistical simulations present higher falling-edge delay deviation when larger devices are used. It is the same tendency as showed in Table 7.12, though the increase in the delay deviation is not so pronounced when the V_{TH} deviation is lowered. The method does not agree with that, once it presents lower falling-edge delay deviation when larger devices are used.

7.4.6 The Influence of the Output Load on Delay Variability

7.4.6.1 The Series Transistors Networks

In the case of the evaluated N- and PMOS 4- transistor stacking, the method can handle different fanouts when the Elmore Delay technique is applied, but not if the AWE method is used. In the former, the rising-edge delay deviation decreases when the output load driven by the network increases, in accordance with the Monte Carlo simulations. In the later, the delay deviation increases as the fanout increases. Fig. 7.4 and 7.5 present the relation between different fanouts and rise delay variability for the stacking transistors analyzed.

Fig. 7.6 and 7.7 present the relation between different fanouts and fall delay variability for the stacking transistors analyzed. The simulated results for falling-edge delay variability are not as easily able to fit as the former results. Fig. 7.6 reveals the tendency of a decreased variability for a certain interval in which the output load increases, but is is not so for all the fanouts studied. The use of the AWE method for different output loads is the closest tendency one could get to the simulated values tendency, as shown in Fig. 7.7.

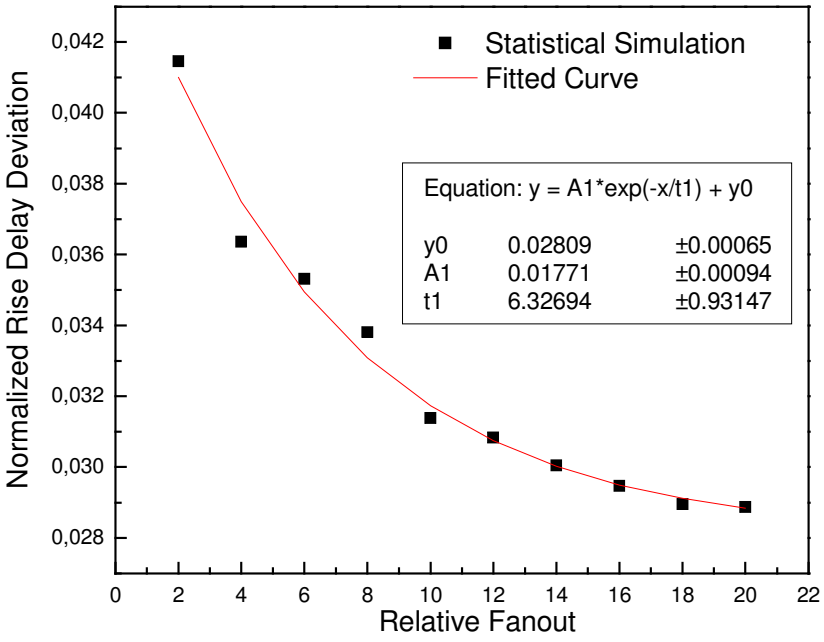


Figure 7.4: Simulated and fitted curves for rising-edge delay deviation in relation to the output load of a 4-stacking PMOS network.

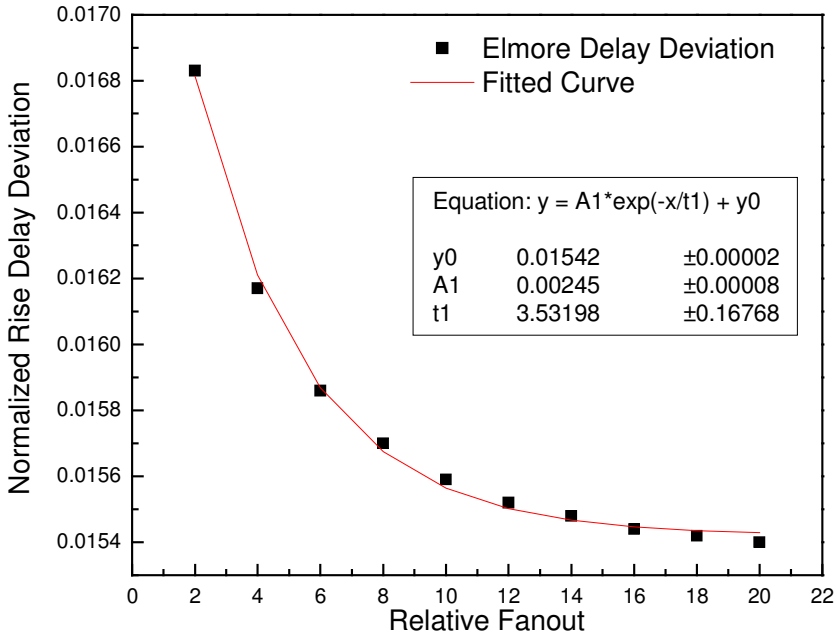


Figure 7.5: Modeled and fitted curves for rising-edge delay deviation in relation to the output load of a 4-stacking PMOS network using Elmore Delay model.

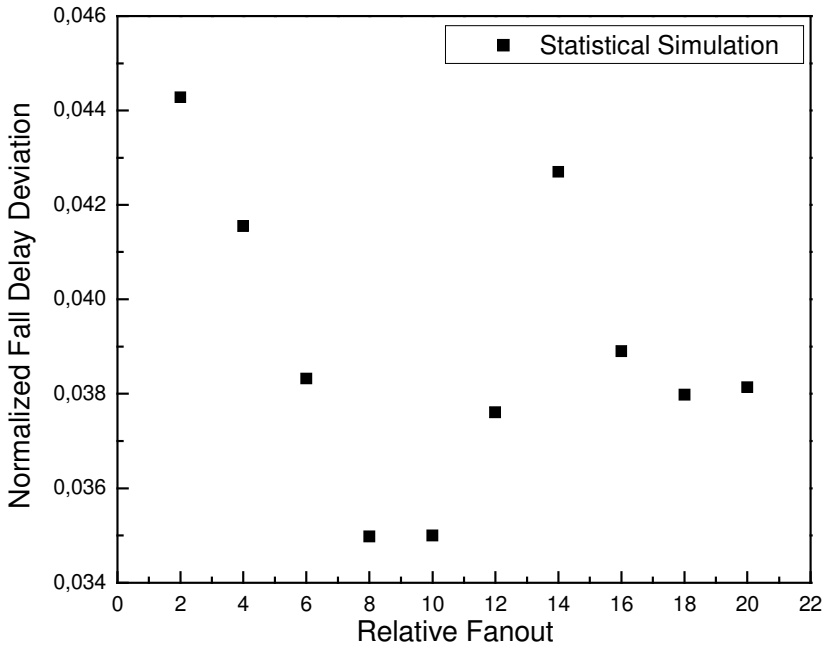


Figure 7.6: Simulated points for falling-edge delay deviation in relation to the output load of a 4-stacking NMOS network.

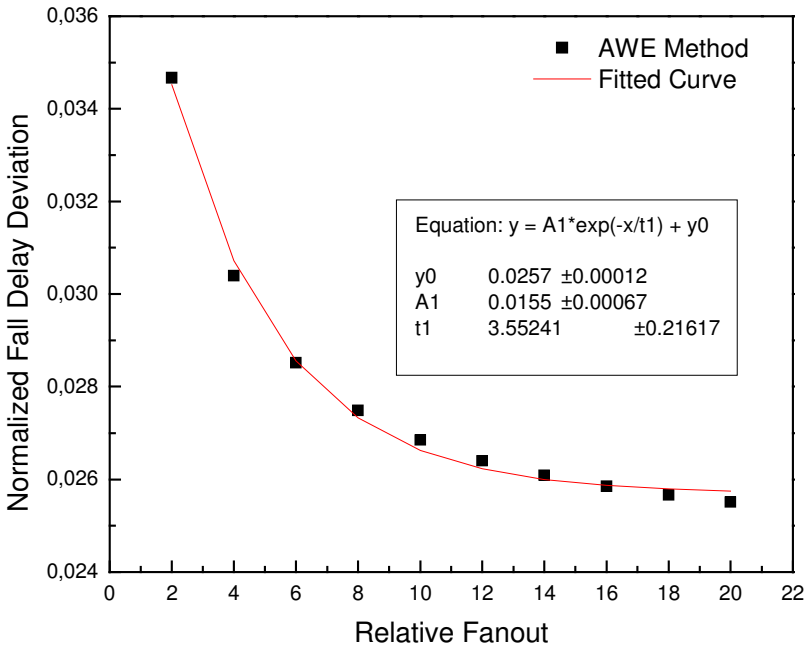


Figure 7.7: Modeled and fitted curves for falling-edge delay deviation in relation to the output load of a 4-stacking NMOS network using AWE.

7.4.7 Inverter Chains

The delay deviations of chains with different number of inverters are measured and the results can be seen in Table 7.14. It is shown in Fig. 7.8 two of the measurement structures used in this section. In each one, the first two inverters were placed between the ideal voltage source and the gate under analysis (X) in order to provide a more realistic input slope. From one to five inverters X were placed in the chain and timing deviations were taken. It also presents results provided by the proposed model. The dimensions are the same as used before for the case of analyzing inverters: W_N = 0.045µ and W_P=0.170 µ.

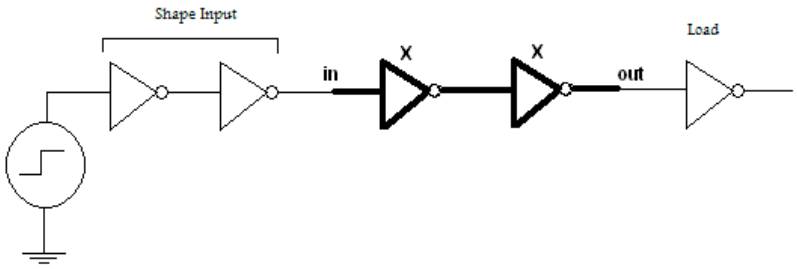


Figure 7.8: Measurement structures.

Table 7.14. Rise and fall delay deviations for different chains of inverters.

Number of Inverters	*Rise Delay Deviation*		*Fall Delay Deviation*	
	Method	*Simulation*	*Method*	*Simulation*
1	0.0360	0.0407	0.0350	0.0415
2	0.0252	0.0293	0.0252	0.0332
3	0.0206	0.0266	0.0205	0.0279
4	0.0178	0.0224	0.0178	0.0253
5	0.0160	0.0209	0.0159	0.0222

In both cases - simulated and modeled – chains with the same numbers of inverters present similar rise and fall delay deviations. The results provided by the Monte-Carlo simulations and the semi-empirical method agree in an important point: as the number of inverters in the chain increases, the normalized deviations of rise and fall delays decrease. The proposed model provides results with an average error of 18.4%, maximum error equals to 23.4% and minimum error of 11.6% for the rise delay deviation. The results for the fall delay deviation present an average error of 24.9%, maximum error equals to 29.6% and minimum error equals to 15.7%.

7.5 Conclusion

Previous simulations with series NMOS transistors revealed lower delay deviation for a switching transistor close to the output node than for a transition in a device far from the output node. For a stack with PMOS transistors higher immunity to variation is achieved when the switching transistor is far from the output terminal. The semi-empirical method achieved for the resistances and used to calculate timing deviation in the networks has also proved the same tendency for the PMOS network but that was not exactly so for the NMOS network.

The results revealed that, in general, the position of switching transistor impacts more strongly N- and PMOS stacks with fewer transistors. The model proposed for the resistances of transistors in a network allows one to investigate how variations in threshold voltage of each device influence the resistance of all devices in the arrangement. By using the resistance functions it is also possible to analyze the performance variability according to the state and position of the devices in the network. Though Elmore delay technique present some limitations, it is plausible to be used to perform the analysis.

8 EVALUATION OF THE PROPOSED DELAY VARIABILITY ESTIMATION METHOD

In this chapter, the proposed estimation method was used to evaluate the delay deviation of logic functions implemented in different topologies and logic styles. The propagation signal path was simplified in order to apply the modeled resistances to the Elmore Delay Model (ELMORE, 1948) in the calculation of the delay variability. This was performed by replacing parallel resistances with their equivalent ones. The AWE technique (PILLAGE, 1990) was also used in some cases, as specified along the text. The logic cells comprehend a 2-input XOR (XOR2), a 4-input XOR (XOR4) and a full adder, which are common structures used in cell libraries for technology mapping.

8.1 2-Input XOR Logic Gate

A 2-input XOR was implemented in different topologies, as shown in Fig. 8.1. The sizing of the devices was performed as to balance the drive strength of the networks by using PMOS devices twice wider than the NMOS devices. The semi-empirical method is used to provide delay variation values (normalized delay deviation). Measurements of rising- and falling-edge delays are also taken with statistical simulations. The purpose is to investigate how reliable the model is to predict which topology should be chosen (or disregarded) when a more robust configuration (with less delay variability) is desired. The configurations considered are: (i) a single complex CMOS gate, (ii) an implementation with 4 NAND gates and (iii) an implementation using pass-transistor logic.

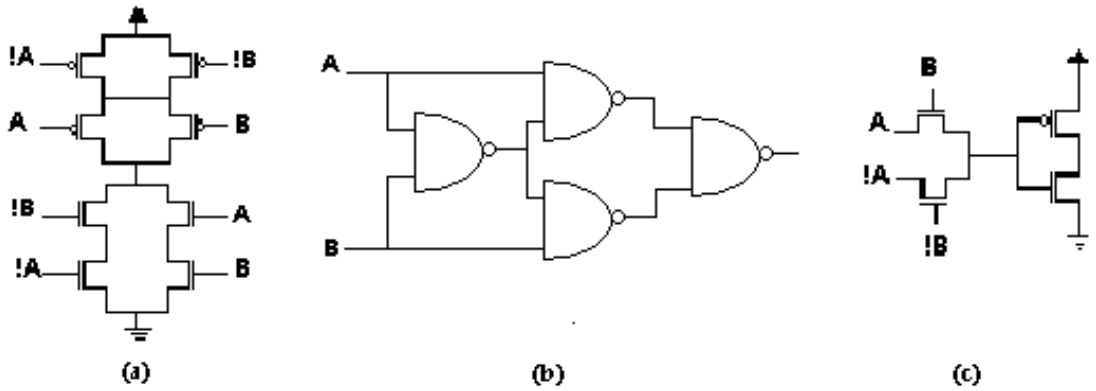


Figure 8.1: A 2-input XOR implemented in different logic styles and topologies: (a) complex CMOS gate; (b) basic CMOS gates and (c) pass-transistor logic (PTL).

The configurations presented in Fig. 8.1 were modeled in a transistor-level as explained in CHAPTERS 6 and 7. For each cell, 2^k DC simulations for each network (pull-up and pull-down) and one transient simulation for the overall logic gate were run in order to model each gate. Though some results achieved with the model present large errors when compared to the simulated ones, they are able to indicate the topology that is more susceptible to threshold voltage variations among those analyzed for both rising- and falling-edge delay (Table 8.1).

Table 8.1: Delay deviations for different implementations of a 2-input XOR provided by the proposed method and by statistical simulation.

Topology or Logic Style	*Rise Delay Deviation*			*Fall Delay Deviation*		
	Method	*Simulation*	*Error (%)*	*Method*	*Simulation*	*Error (%)*
Complex CMOS Gate	0.0258	0.0411	37.2	0.0480	0.0226	112.4
Basic Gates (NAND)	0.0339	0.0287	18.1	0.0306	0.0277	10.1
PTL	0.0252	0.0350	28.0	0.1537	0.1330	15.6

The method was able to predict the large falling-edge delay deviation presented by the PTL style. It overestimated the falling-edge delay deviation for the complex CMOS gate and the PTL configurations analyzed. Results provided by the proposed method and the statistical simulations points at the configuration with NAND gates as the most immune to variations in V_{TH} when both rising- and falling-edge delay deviations are considered.

The highest error presented by the model was for the falling-edge delay deviation of the CMOS complex gate by considering the modeled resistances applied to the Elmore Delay Model. However, the AWE technique was also tested for this case and a better result (0.0388) with an error of 71.7 % was achieved.

Complete statistical simulation results are presented in Table 8.2. Regarding the mean delay values, the complex gate is the better choice among the configurations. Though it presents the worst rising-edge delay deviation (simulated), it also presents the lower falling-edge delay deviation. The pass-transistor logic revealed itself a bad choice regarding timing analysis, since it presents the higher overall mean delay and the much worst overall delay variation.

Table 8.2: Delay values and deviations for different implementations of a 2-input XOR.

Topology or Logic Style	*Rise Delay*			*Fall Delay*		
	Mean (ps)	*Absolute Deviation (ps)*	*Normalized Deviation*	*Mean (ps)*	*Absolute Deviation (ps)*	*Normalized Deviation*
Complex CMOS Gate	80.003	3.289	0.0411	91.403	2.062	0.0226
Basic Gates (NAND)	190.683	5.464	0.0287	148.191	4.112	0.0277
PTL	90.310	3.163	0.0350	244.602	32.521	0.1330

Once the method presented so far proposes a solution for the delay variability, but not for its mean value, the analysis was completed by running a single transient simulation with nominal values of threshold voltages. These nominal delays were then used, along with the normalized standard deviation provided by the model, to provide the delay probability density functions (normal distributions). Figures 8.2-9 show the delay PDFs provided by statistical simulations and the PDFs provided by the estimation method for each topology. Figures 8.3 and 8.7 show that regarding both rising- and falling-edge delay, a 2-input XOR implemented with NAND logic gates has its delay PDF very well predict by the proposed method. For the rising-edge delay, the method is optimistic for the complex gate and the PTL topologies analyzed, once it presented lower deviation values than those provided by the simulations.

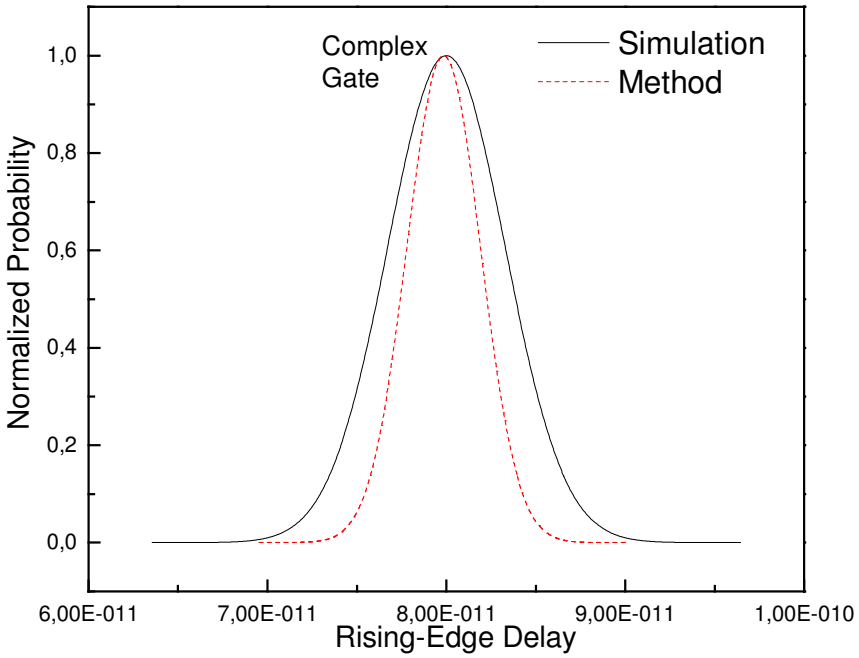


Figure 8.2: PDF of the rising-edge delay for the complex gate implementation of a 2-input XOR.

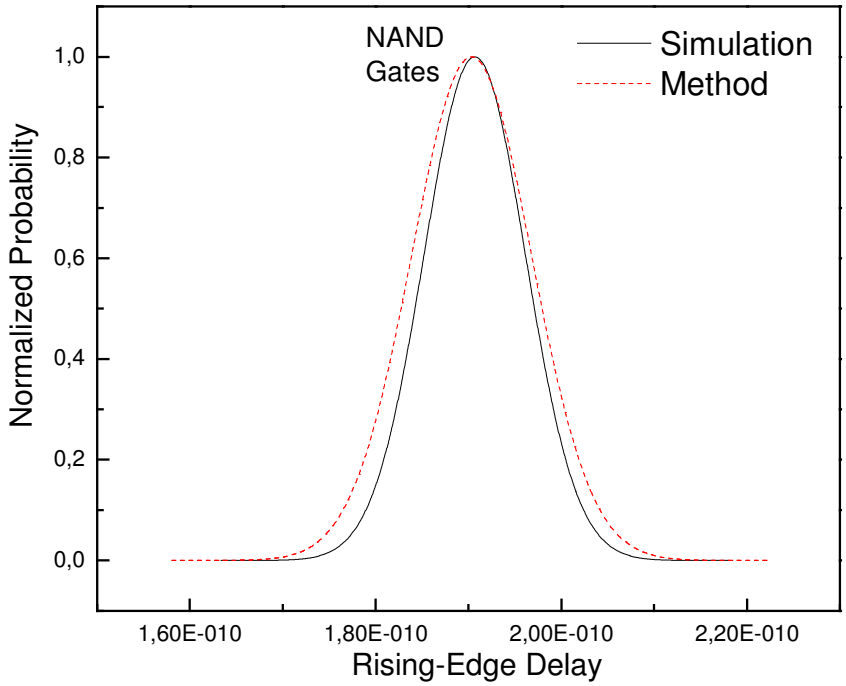


Figure 8.3: PDF of the rising-edge delay for the implementation of a 2-input XOR with NAND gates.

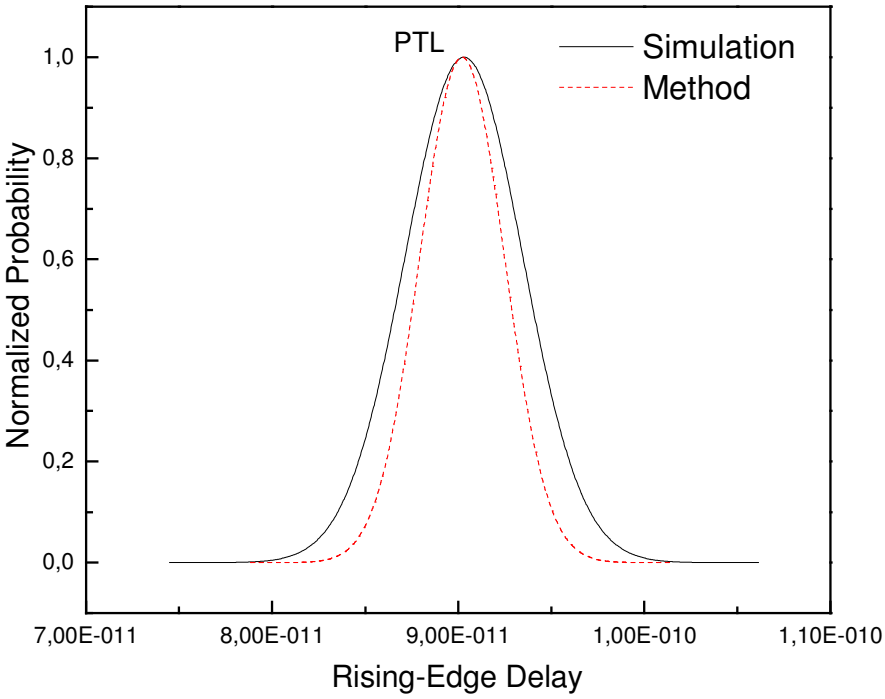


Figure 8.4: PDF of the rising-edge delay for the PTL implementation of a 2-input XOR.

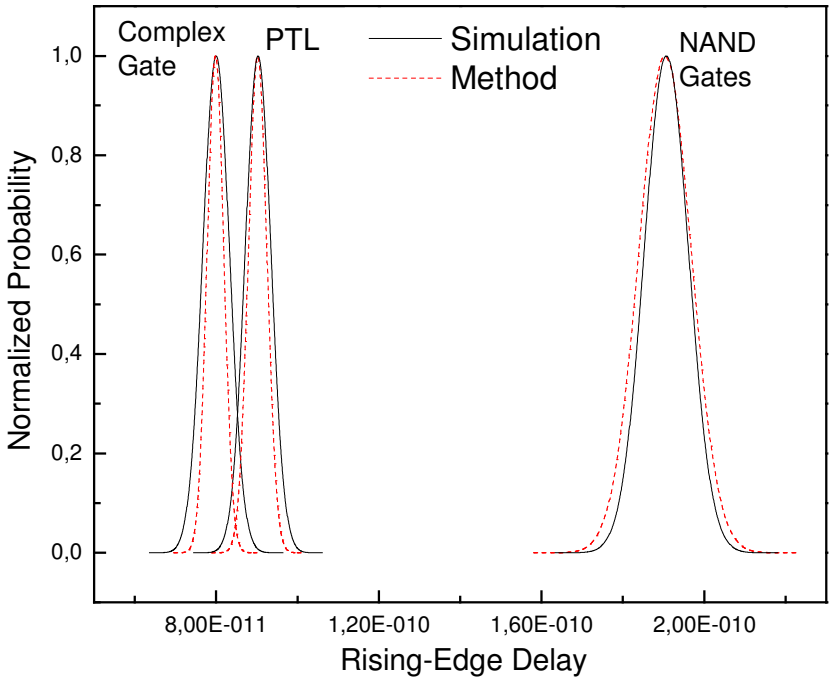


Figure 8.5: PDF of rising-edge delay for different implementations of a 2-input XOR.

Fig. 8.6 shows that the results provided by the method for the complex gate configuration is very pessimistic when falling-edge delay deviation is considered. The method was able to predict the large variability presented by the DPTL logic style, what

may cause the use of this configuration to be prohibitive in certain designs, for the sake of parametric yield improvement.

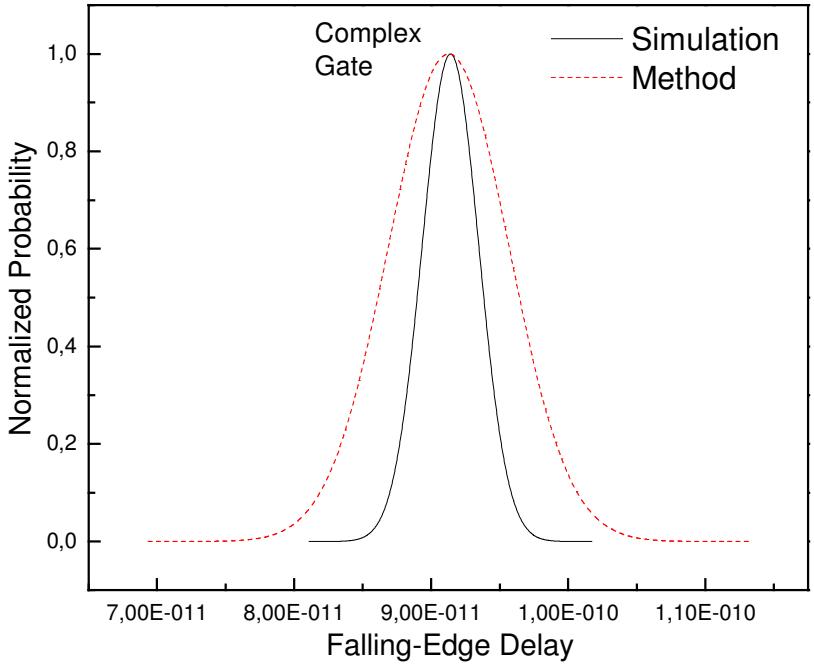


Figure 8.6: PDF of the falling-edge delay for the complex gate implementation of a 2-input XOR.

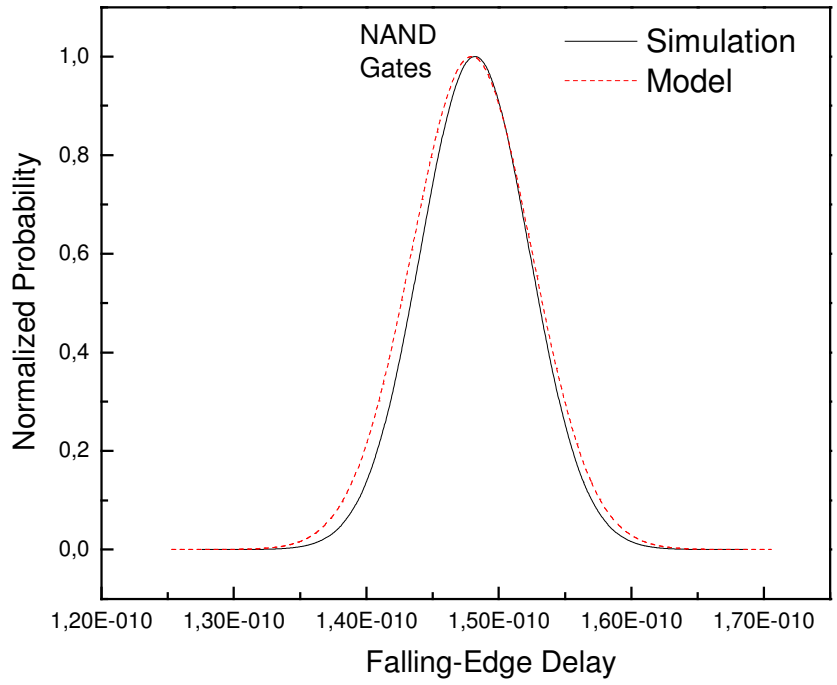


Figure 8.7: PDF of the falling-edge delay for the implementation of a 2-input XOR with NAND gates.

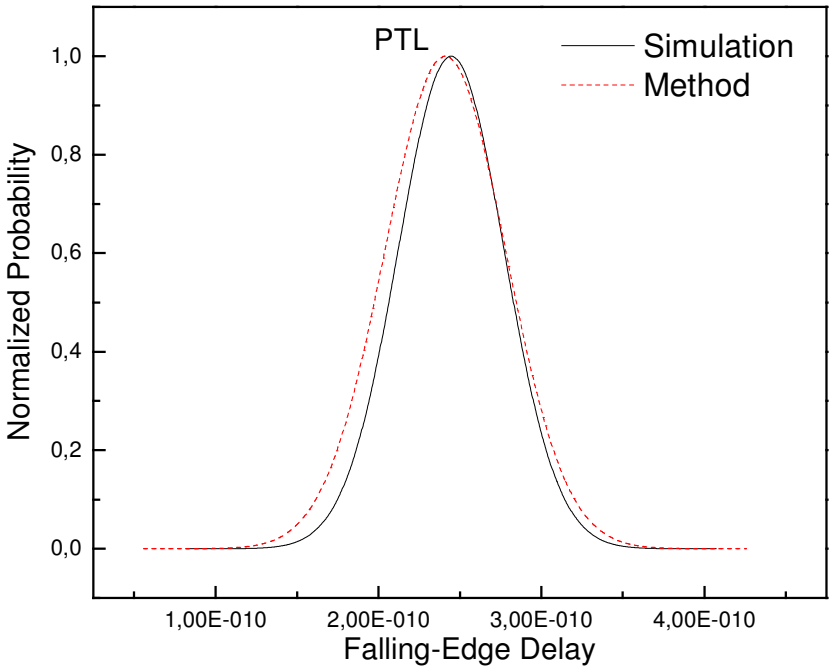


Figure 8.8: PDF of the falling-edge delay for the PTL implementation of a 2-input XOR.

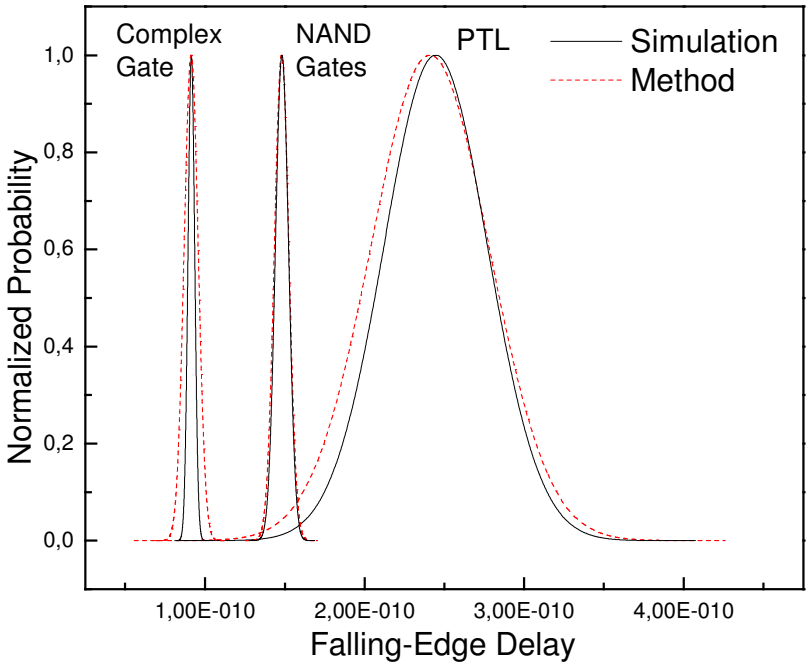


Figure 8.9: PDF of the falling-edge delay for different implementations of a 2-input XOR.

8.2 4-Input XOR Logic Gate

Fig. 8.10 presents a 4-input XOR implemented in different logic styles and topologies. The modeled resistances were first used in the Elmore Delay Model (ELMORE, 1948), and those analyses that presented major errors were also performed by using AWE technique (PILLAGE, 1990).

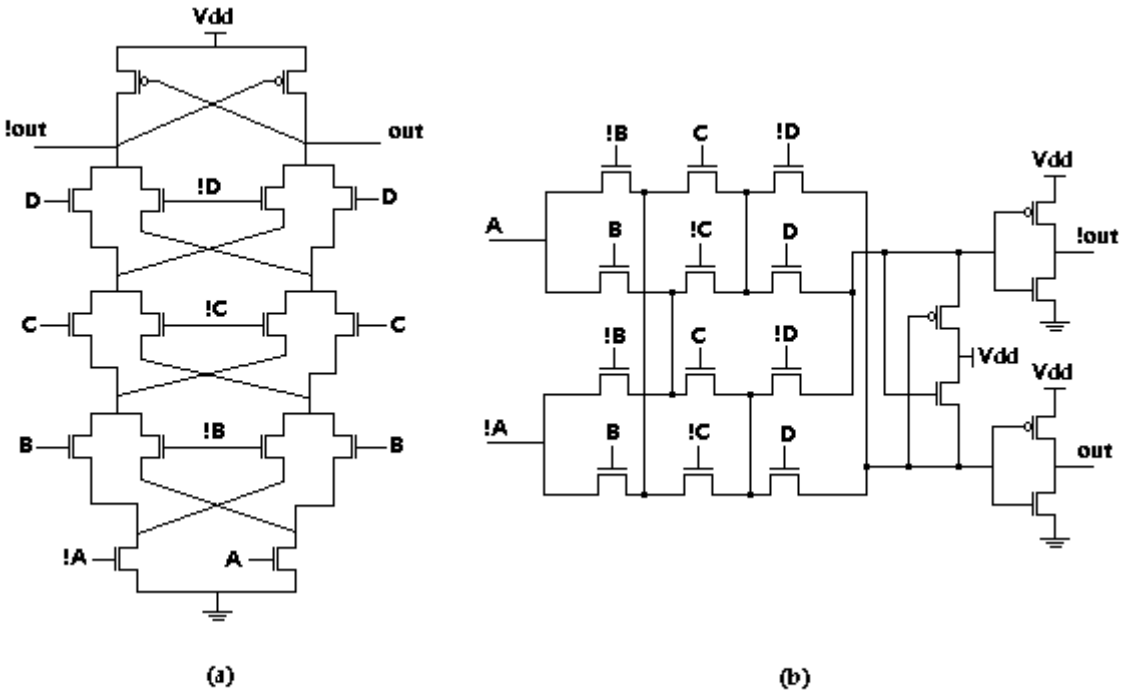


Figure 8.10: 4-input XOR implemented in different logic styles: (a) differential cascode voltage logic (DCVSL) and (b) pass-transistor logic (PTL).

Table 8.3 compares the delay deviations provided by the proposed method with those achieved through statistical simulations.

Table 8.3: Delay deviations for different implementations of a 4-input XOR provided by statistical simulation and by the proposed method.

Topology or Logic Style	*Rise Delay Deviation*			*Fall Delay Deviation*		
	Method	*Simulation*	*Error (%)*	*Method*	*Simulation*	*Error (%)*
DCVSL	0.0361	0.0426	15.3	0.0282	0.0261	7.9
DPTL	0.0351	0.0341	2.9	0.0717	0.1464	51.0

The best results (the most similar to the simulated values) presented by the method considering both rising- and falling-edge delay variations were for the DCVSL style.

The method was able to predict the least robust configuration among those used to implement the 4-input XOR gate. The highest error presented by the model was for the falling-edge delay deviation of the DPTL style by considering the modeled resistances applied to the Elmore Delay Model. However, the AWE technique was tested for this case and a better result (0.1314), with an error of 10.3 % was achieved.

Rise and fall delay mean values as well as their deviations provided by statistical simulations are found in Table 8.4. The complex gate implementation presented good overall variability results and the lowest rise delay deviation among the topologies studied.

Table 8.4. Delay values and deviations for different implementations of a XOR4.

Logic Style	*Rise Delay*			*Fall Delay*		
	Mean (ps)	*Absolute Deviation (ps)*	*Normalized Deviation*	*Mean (ps)*	*Absolute Deviation (ps)*	*Normalized Deviation*
DCVSL	341.803	14.554	0.0426	67.737	1.769	0.0261
DPTL	315.539	10.749	0.0341	310.429	45.456	0.1464

Definitely, the pass-transistor logic style was the configuration that presented worst deviation results, especially for the falling-edge delay, where it also presented the worst mean delay value. However, for the rising-edge delay it was the best topology among those presented.

Fig. 8.11 and 8.12 show the probability density functions of rise and fall delays for each configuration. The method presented a reliable rising-edge delay deviation for the topologies studied, since it provided PDFs that are very close to the statistically simulated ones.

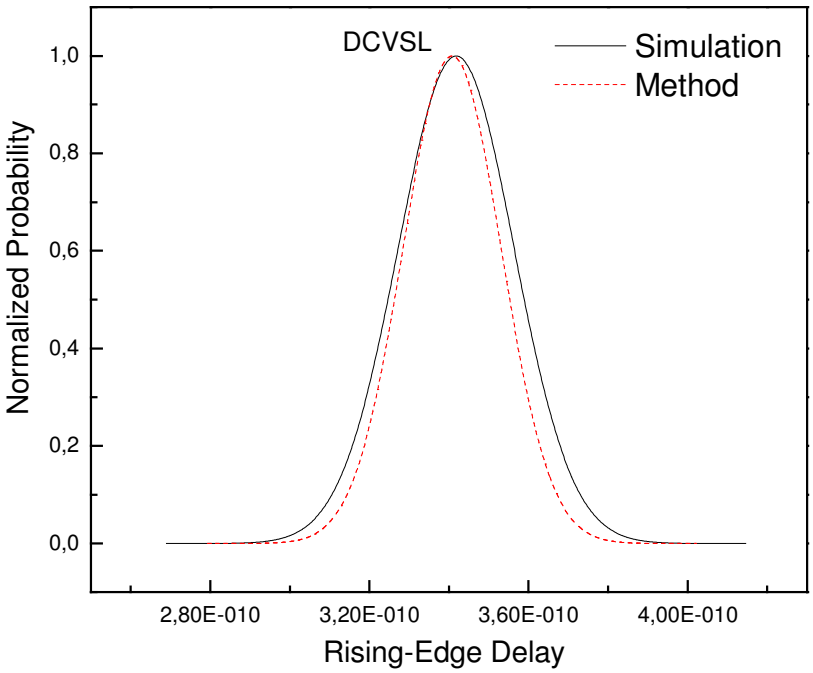


Figure 8.11: PDF of the rising-edge delay for the DCVSL implementation of a 4-input XOR.

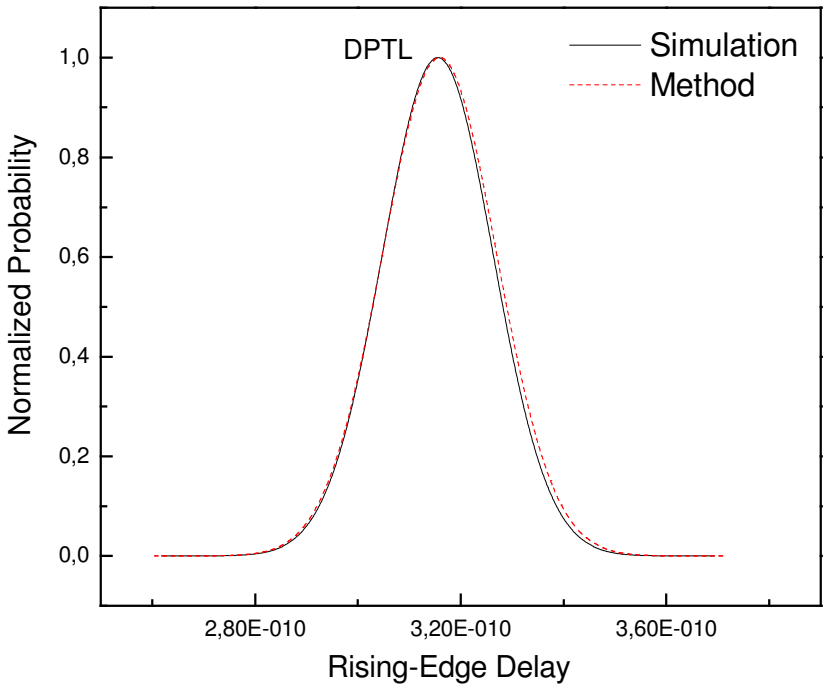


Figure 8.12: PDF of the rising-edge delay for the DPTL implementation of a 4-input XOR.

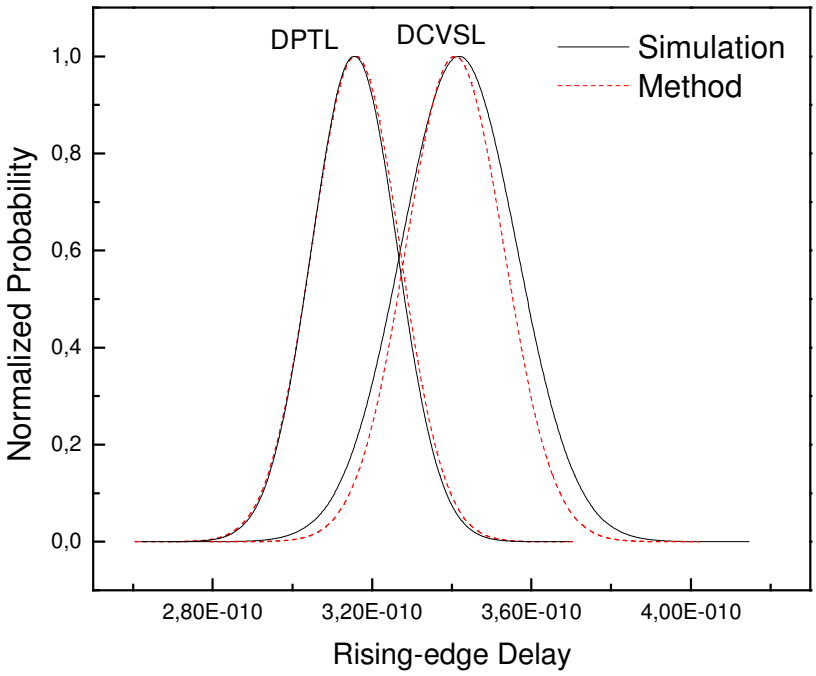


Figure 8.13: PDF of rising-edge delay for different implementations of a 4-input XOR.

In Fig. 8.16 it is remarkable the high falling-edge delay deviation presented by the pass-transistor logic style. The method was very optimistic for the DPTL style, providing lower delay deviation for this topology when Elmore Delay model was applied to calculate the deviation. Despite of its limitation regarding the DPTL style, the method is able to predict the configuration with the highest variability among those evaluated. The method presented good reliability for the DCVSL topology.

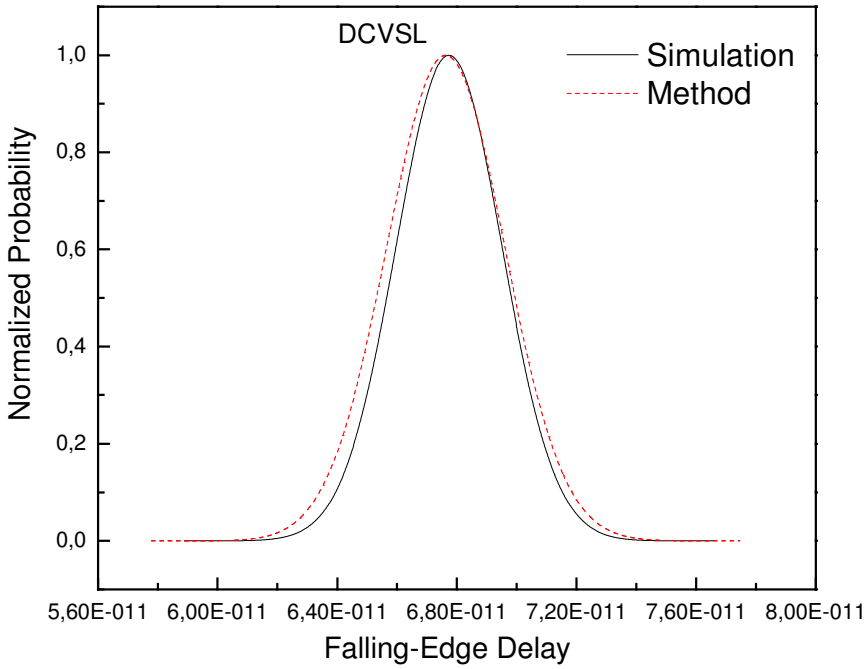


Figure 8.14: PDF of the falling-edge delay for the DCVSL implementation of a 4-input XOR.

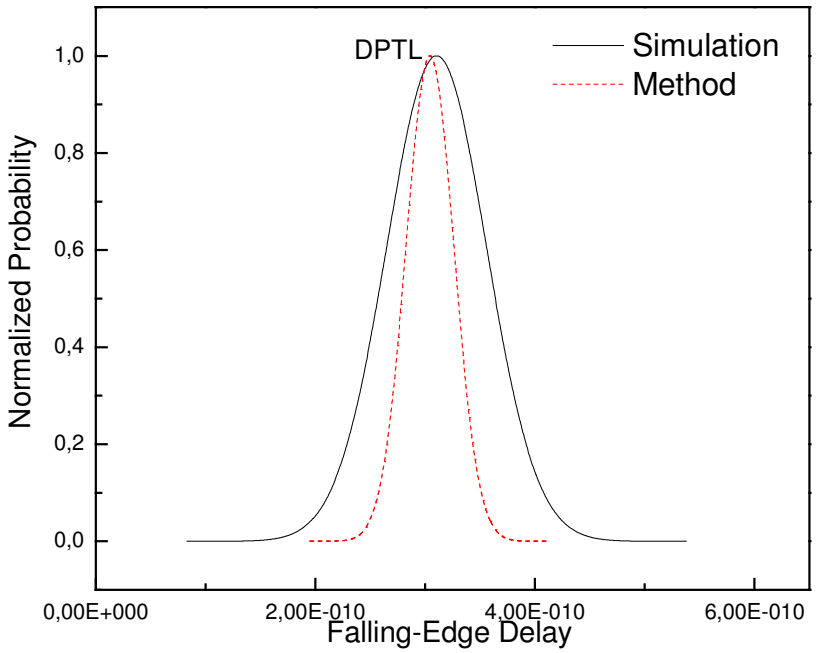


Figure 8.15: PDF of falling-edge delay for the DPTL implementation of a 4-input XOR.

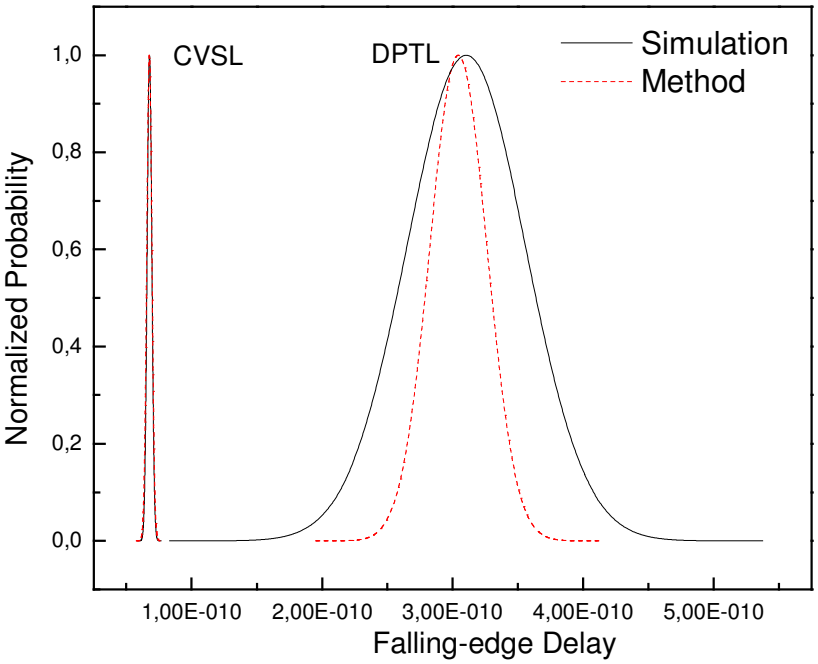


Figure 8.16: PDF of falling-edge delay for different implementations of a 4-input XOR.

Fig. 8.17 shows the probability density functions of falling-edge delay for the DPTL implementation by considering the deviation provided by the method when AWE technique is used. The modeled PDF is much more similar to the simulated one than in Fig. 8.15.

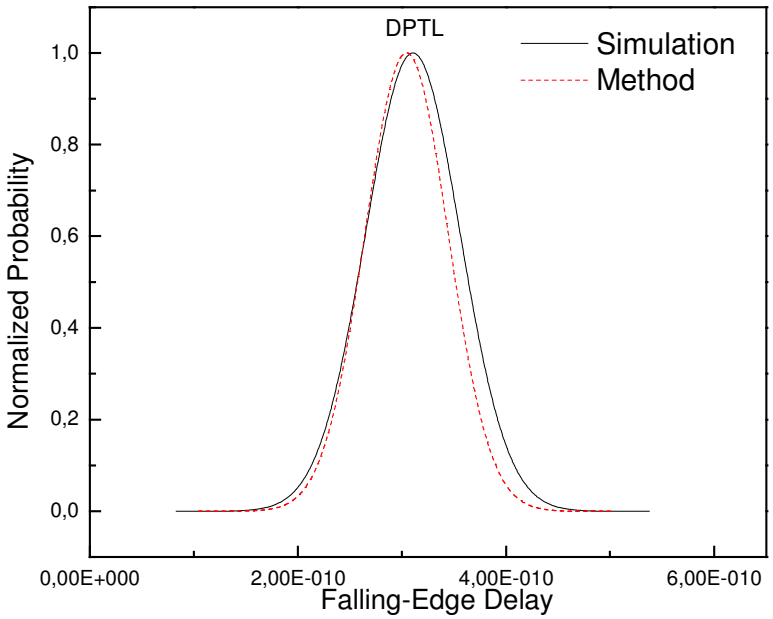


Figure 8.17: PDF of falling-edge delay for DPTL implementation of a 4-input XOR modeled with AWE technique.

8.3 Full Adder

The full adder is a logic cell with three inputs and two outputs (WESTE, 2005) and it presents the logic functions summarized in the truth table showed in Table 8.5:

Table 8.5. Truth table of a full adder.

Inputs			*Outputs*	
C_{IN}	*B*	*A*	C_{OUT}	*SUM*
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

The relation between the inputs and the “SUM” output can be represented as the following logic functions:

The presence of a carry-in input and a carry-out output makes the full adder highly scalable and it is found in many cascade circuit implementations. A full adder was implemented in two different logic styles: (i) a complex CMOS gate and (ii) in the

differential pass-transistor logic (DPTL). The topologies shown in Fig. 8.18 provide the "SUM" output of the logic cell.

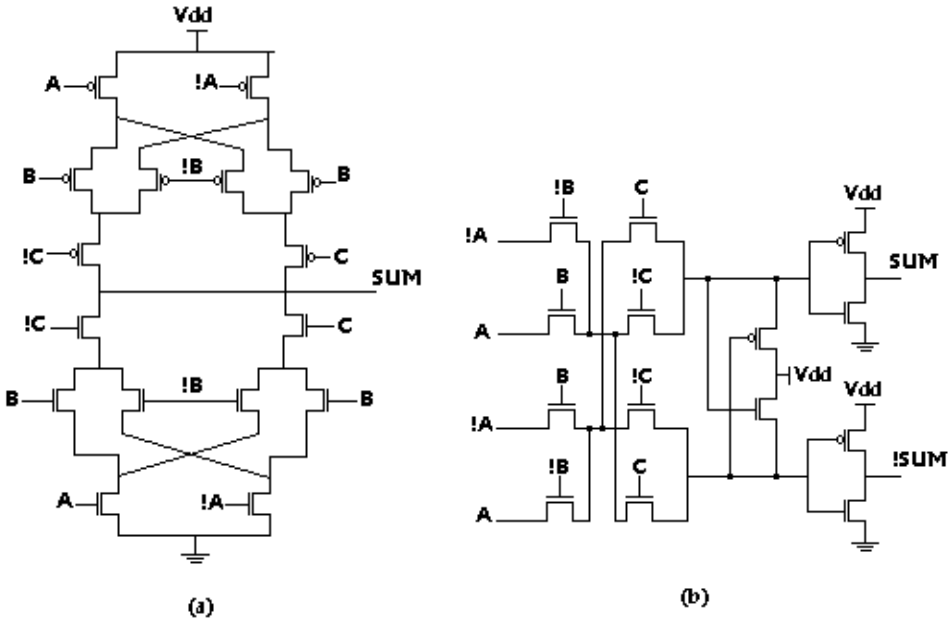


Figure 8.18: Full adder implemented in different logic styles and topologies: (a) complex CMOS gate; (b) differential pass-transistor logic (DPTL).

Table 8.6 compares the delay deviations provided by the proposed method with those achieved through statistical simulations.

Table 8.6: Delay deviations for different implementations of a full adder provided by statistical simulation and by the proposed method.

Topology or Logic Style	*Rise Delay Deviation*			*Fall Delay Deviation*		
	Method	*Simulation*	*Error (%)*	*Method*	*Simulation*	*Error (%)*
Complex Gate	0.0162	0.0234	30.8	0.0376	0.0314	19.7
DPTL	0.0316	0.0207	52.7	0.1194	0.0867	37.7

The complex gate presented in Fig. 8.7a has the highest rise delay deviation and the logic cell in Fig. 8.7b has the highest fall delay deviation according to the statistical simulations. The NMOS network of the complex CMOS is more susceptible to variations than the PMOS network, as seen in CHAPTER 7. The method agrees with the simulation results when it regards the falling-edge delay deviation, showing that the DPTL topology is the least robust.

The error presented by the method for the falling-edge delay deviation of the DPTL style by considering the modeled resistances applied to the Elmore Delay Model was reduced to 17.2 % when the resistances were used in the AWE technique. In this case,

the deviation was 0.1016. No better result was achieved when AWE was used in the model to calculate the rising-edge delay deviation of the DPTL style and the error remained around 50%.

The results of rise and fall delay measurements for the "SUM" output node are shown in Table 8.7.

Table 8.7. "SUM" delay deviation for different implementations of a full adder.

Topology / Logic Style	*Rise Delay*			*Fall Delay*		
	Mean (ps)	*Absolute Deviation (ps)*	*Normalized Deviation*	*Mean (ps)*	*Absolute Deviation (ps)*	*Normalized Deviation*
Complex Gate	267.387	6.256	0.0234	81.626	2.560	0.0314
DPTL	293.651	6.077	0.0207	221.442	19.198	0.0867

As can be seen in Fig. 8.9 and 8.10, the method is reliable for the cell in the DPTL style if one consider that the worst delay mean value of this logic style is the rising-edge delay, whose variability was overestimated by the method, providing a safe margin.

The normal distributions of simulated and modeled delay deviations presented the complex CMOS gate as the most apropriate topology to be used in this implementation. The method was more precise when it was applied to the CMOS complex gate (especially for the falling-edge delay), but it was also reliable when used for the DPTL style, since it was a little bit pessimistic.

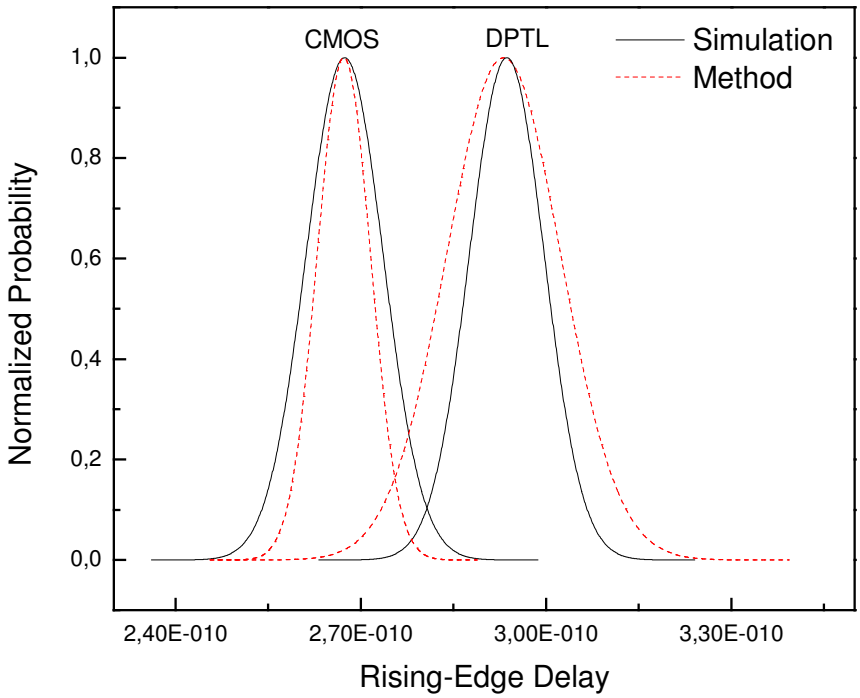


Figure 8.19: "SUM" output node PDF of the rising-edge delay for different implementations of a full adder.

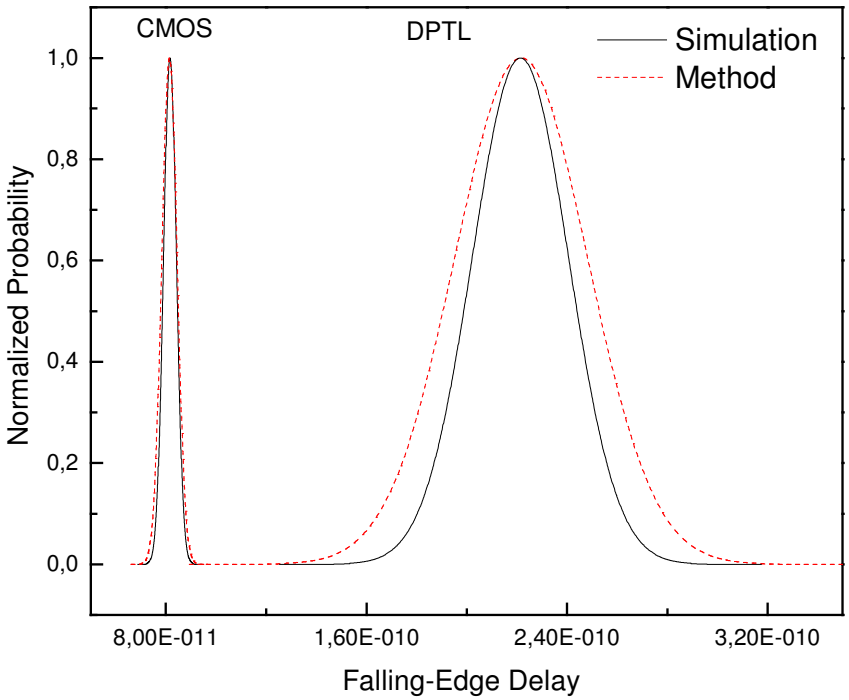


Figure 8.20: "SUM" output node PDF of the falling-edge delay for different implementations of a full adder.

8.4 Complex Gate

The method was also used to evaluate the delay variability of the complex gate configuration showed in Fig. 8.21.

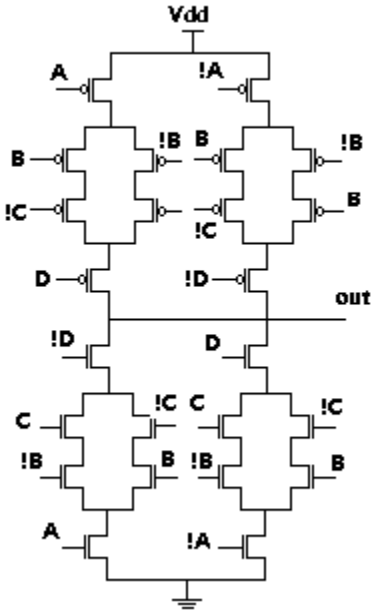


Figure 8.21: A complex gate implementation;

Table 8.8: Delay deviations for a complex gate provided by statistical simulation and by the proposed method.

Topology or Logic Style	*Rise Delay Deviation*			*Fall Delay Deviation*		
	Method	*Simulation*	*Error (%)*	*Method*	*Simulation*	*Error (%)*
Complex Gate	0.0190	0.0224	15.2	0.0291	0.0311	6.4

The method was very reliable in the evaluation of both rising- and falling-edge delay deviations of the complex gate.

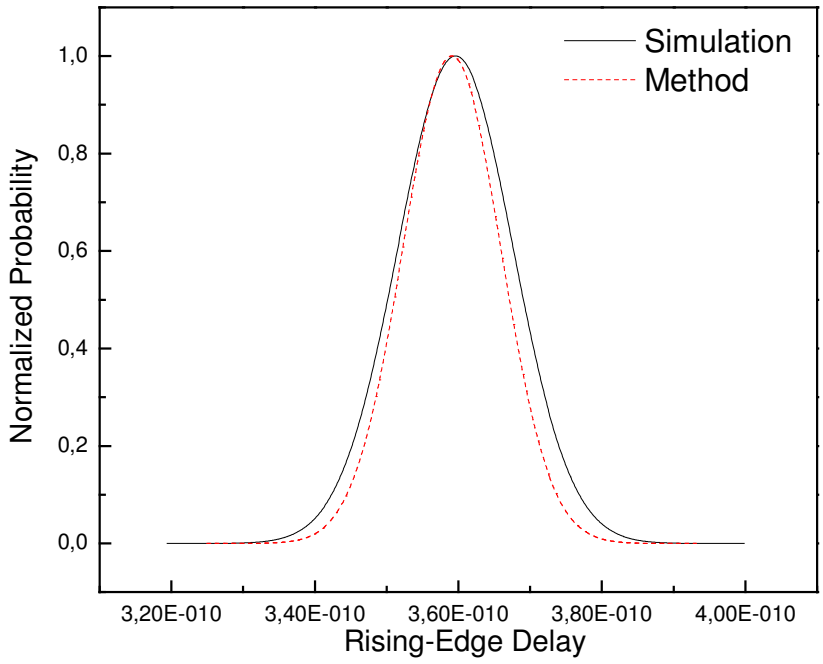


Figure 8.22: PDF of the rising-edge delay for the complex gate implementation.

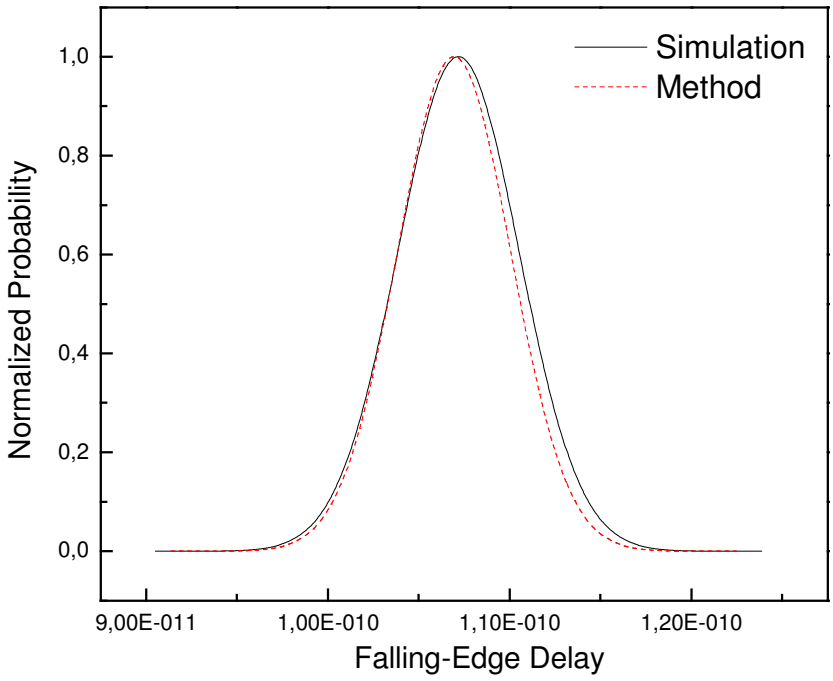


Figure 8.23: PDF of the falling-edge delay for the complex gate implementation.

8.5 Delay Equation Method

As an alternative to the computationally expensive Monte Carlo simulations, the here so-called *Delay Equation Method (DEM)* also uses a least square method in order

to derive the coefficients of the first-order model (delay equation) as a function of the transistors threshold voltage variations by using SPICE simulations. The difference of this method from the estimation method proposed in this work is basically that it requires transient simulations in spite of the DC type ones. In this sense, DEM does not use the RC constant of the network in order to provide the delay of the logic cell. The delay is the metric directly provided by the transient simulations.

Variations of this method are already available in the literature (OKADA 2003) and that is why the analysis provided by the proposed estimation method is compared to the *DEM* in the next sub-section, regarding the runtime. Since this approach can be used to provide the PDF of a logic cell delay without the need of performing Monte Carlo simulations, it is fair that the proposed method is also compared with *DEM*.

8.6 Runtime Analysis

The characterization of the logic gates by using Monte Carlo approach is computationally expensive and takes a long time to be completed (see Table 8.9 for 10.000 runs), what makes it prohibitive in some cases. The approach presented in this work to characterize the cells requires simulations that takes only a few seconds, since only one of them is a transient run and all the others are DC. An impressive speedup can be observed for all the gates analyzed.

The approaches were applied to the 2-input XOR (complex gate topology), to the 4-input XOR (DCVSL logic style) and to the complex gate topology in Fig. 8.21. Table 8.9-11 shows the delay (also deviation) results and the runtime analysis of these structures for the three methods presented: (i) the proposed estimation method, (ii) the delay equation method (*DEM*) and (iii) the statistical simulation (Monte Carlo).

Table 8.9: Delay and runtime analysis for different implementations of a XOR2.

COMPLEX GATE	*Rise Delay*			*Fall Delay*			*Runtime Analysis*
	Mean Delay (ps)	*Standard Deviation (ps)*	*Normalized Deviation*	*Mean Delay (ps)*	*Standard Deviation (ps)*	*Normalized Deviation*	
Estimation Method	79.8	2.1	0.0258	91.3	4.4	0.0480	**3s**
Delay Equation Method	80.0	3.3	0.0409	92.3	2.0	0.0219	37s
Statistical Simulation (Monte Carlo)	80.0	3.3	0.0411	91.4	2.1	0.0226	6.1 h

Table 8.10: Delay and runtime analysis of the complex gate topology by using different methods.

COMPLEX GATE	*Rise Delay*			*Fall Delay*			*Runtime Analysis*
	Mean Delay (ps)	*Standard Deviation (ps)*	*Normalized Deviation*	*Mean Delay (ps)*	*Standard Deviation (ps)*	*Normalized Deviation*	
Estimation Method	359.1	6.8	0.0190	106.9	3.1	0.0291	**11s**
Delay Equation Method	362.5	6.9	0.0191	108.1	2.8	0.0263	2385s ~ 40min
Statistical Simulation (Monte Carlo)	359.6	8.0	0.0224	107.2	3.3	0.0311	13.7 h

As already seen in SECTION 8.4, the results provided by the estimation method were quite close to the statistically simulated ones. Table 8.11 shows the delay and its deviation results and the runtime analysis of the 4-input XOR implemented with the DCVSL topology.

Table 8.11: Delay and runtime analysis of the XOR4 implemented with a DCVSL topology by using different methods.

XOR4 DCVSL	*Rise Delay*			*Fall Delay*			*Runtime Analysis*
	Mean Delay (ps)	*Standard Deviation (ps)*	*Normalized Deviation*	*Mean Delay (ps)*	*Standard Deviation (ps)*	*Normalized Deviation*	
Estimation Method	340.8	12.3	0.0361	67.6	2.0	0.0291	**10s**
Delay Equation Method	342.2	4.4	0.0128	68.3	1.5	0.0218	1474s ~ 25min
Statistical Simulation (Monte Carlo)	341.8	14.6	0.0426	67.7	1.8	0.0261	16.6 h

Tables 8.10 and 8.11 show that the *DEM* also presented good results, but at the expense of a higher simulation runtime than the proposed method.

8.7 Conclusion

In most of the cases, the proposed estimation method turned to be a good approximation for calculating the delay probability density function of the analyzed logic gates, providing safe margins of delay deviation. The method was able to provide approximate delay PDFs for the 2- and 4-input XOR, and for the full adder. Also, the method could predict the delay variability of the complex gate with a very good approximation.

9 CONCLUSION

Different topologies of inverters present different delay deviations when the threshold voltages of transistors vary. Preliminary analysis based on Monte Carlo simulations showed that, for a certain logic function, the choice of the topology and/or logic style, the number of transistors in the network and the position of a switching transistor in relation to the output node influence the delay variability of the logic cell. When only a set of transistors is analyzed, as the pull-up or the pull-down network in a CMOS configuration for example, better results (less delay deviation) were achieved for a higher number of transistors in the arrangement.

The variability estimation method proposed was mainly applied to complementary MOS (CMOS) logic style, and also presented valuable results for DCVSL, PTL and DPTL styles. For composing the method, the resistance of each transistor was calculated as a function of the threshold voltages variations of all the devices in the network. Each topology required a 2.2^k DC electrical simulations, which took less than 10 seconds even for logic gates with more than 10 transistors. The use of two different regions of operation – linear and saturation – in order to achieve the resistances values showed that the linear region provides more reliable resistance model functions.

The calculus of the resistances was done for each type of transistor (pull-up or pull-down configuration) at a time. In this sense, the on-resistance of NMOS transistors do not depend on the characteristics of PMOS, and vice-versa. This point may be important to explain the differences between the delay variability provided by NAND and NOR gates (CHAPTER 5) and the structures analyzed in CHAPTER 7. In this chapter, the configurations were simulated with Monte Carlo by considering only one type of network (only pull-up or pull-down configurations) and not a complete logic gate.

The analysis of networks with series and parallel transistors revealed considerable differences whether N- or PMOS devices compose the arrangement. NMOS networks have delay deviations that are more sensitive to variations in the threshold voltages of transistors. In general, a higher number of MOS in a stacking topology represents a more robust arrangement. According to the position of a switching transistor, it is seen that N- and PMOS transistors switching far from the output node results in lower delay deviations. In a pure parallel network, variations in the threshold voltages of transistors that are not conducting have no influence on delay variability.

In the case of inverters designed with different topologies when the V_{TH} deviation is changed according to the dimensions of the transistors (Pelgrom model) it is clear that larger MOS devices provide lower delay variability.

Results provided by both simulations and modeling also showed that for a chain of inverters, the higher the number of inverters in the chain, the lower the rising- and

falling-edge delay deviations. Finally, the comparison of different topologies and logic styles used to implement the same logic function – a 2-input XOR – showed that the delay of the configuration with NAND gates suffers less influence of the threshold voltage variations of transistors. The pass-transistor logic presented the highest delay deviation among the topologies analyzed. The logic gate(s) was(were) also analyzed with the proposed method, which provided compatible results with the statistical simulations. In the case of a 4-input XOR, the gate implemented in the DCVSL style appears as the most reliable configuration among those analyzed. For the full adder logic cell, the complex CMOS gate is preferible over the DPTL style, when the delay of the "SUM" output node is analyzed.

The discrepancies between the simulation results and the delay deviations provided by the estimation method can be partly explained by the limitations of the later. The method does not take into account the effect of threshold voltage variations on the gate capacitances of devices. Also, the logic cell has its rising- and falling-edge delay paths analyzed separately, then the model does not regard the influence of devices of the pull-up network on the pull-down network, and vice-versa. Other limitation is the modeling of the devices, once the calculation of the resistances of transistors is only an approximation of their actual behavior. However, in order to provide the delay probability density function of a certain topology, the proposed method demands a runtime that is around 3.000x smaller than that demanded by the Monte Carlo method (10.000 runs) . The runtime is also much smaller than that of the so-called *DEM*.

REFERENCES

AMIN, C. et al. A Multi-Port Current Source Model for Multiple-Input Switching Effects in CMOS Library Cells. In: DESIGN AUTOMATION CONFERENCE, 43rd, San Francisco, USA. **Proceedings...** ACM Press, p. 247-252, July 2006.

ARGAWAL, A. et al. Path-Based Statistical Timing Analysis Considering Inter and Intra-Die Correlations. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, p. 621-625, 2003.

ARGAWAL, K. et al. Parametric Yield Analysis and Optimization in Leakage Dominated Technologies. **IEEE Transactions on VLSI Systems**, New York, USA, v. 15, n. 6, p. 613-623, June 2007.

ARUNACHALAM, R., DARTU, F., PILEGGI, L. CMOS Gate Delay Models for General RLC Loading. **IEEE International Conference on Computer Design**. pp. 224-229, October 1997.

BERKELAAR, M. Statistical Delay Calculation, A Linear Method, TAU, p. 15-24, 1997.

BLAAUW, D., ZOLOTOV, V., SUNDARESWARAN, S. Slope Propagation in Static Timing Analysis. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, v. 21, n. 10, October 2002.

CAO, Y., CLARK, L. T. Mapping Statistical Process Variations Toward Circuit Performance Variability: An Analytical Modeling Approach. In: DESIGN AUTOMATION CONFERENCE, 42nd, Anaheim, USA. **Proceedings...** ACM Press, p. 13-17, June 2005.

CELIK, M., PILEGGI, L., ODABASIOGLU, A. **IC Interconnect Analysis**. Springer, 2002.

CHIANG, C., KAWA, J. **Design for Manufacturability and Yield for Nano-Scale CMOS**. Springer Netherlands, 2007.

CHOI, S., PAUL, B., ROY, K. Novel Sizing Algorithm for Yield Improvement under Process Variation in Nanometer Technology. In: DESIGN AUTOMATION CONFERENCE, 41st, San Diego, USA. **Proceedings...** ACM Press, p. 454-459, June 2004.

CROIX, J., WONG, D. Blade and Razor: Cell and Interconnect Delay Analysis Using Current-Based Models. In: DESIGN AUTOMATION CONFERENCE, 40th, Anaheim, USA. **Proceedings...** ACM Press, p. 386-389, June 2003.

DAGA, J., AUVERGNE, D. A Comprehensive Delay Macro Modeling for Submicrometer CMOS Logics. **IEEE Journal of Solid-State Circuits**, v. 34, n. 1, January 1999.

DARTU, F., MENEZES, N., PILEGGI, L. Performance Computation for Precharacterized CMOS Gates with RC Loads. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, v. 15, n. 5, May 1996.

DARTU, F., PILEGGI, L. Calculating Worst-Case Gate Delays due to Dominant Capacitance Coupling. In: DESIGN AUTOMATION CONFERENCE, 34th, Anaheim, USA. **Proceedings...** ACM Press, p. 46-51, June 1997.

ELMORE, W. C. The Transient Analysis of Damped Linear Networks with Particular Regard to Wideband Amplifiers. **J. Applied Physics**, v. 19, n. 1, p. 55-63, 1948.

FATEMI, H., NAZARIAN, S., PEDRAM, M. Statistical Logic Cell Delay Analysis Using a Current-based Model. In: DESIGN AUTOMATION CONFERENCE, 43rd, San Francisco, USA. **Proceedings...** ACM Press, p. 253-256, July 2006.

GUPTA, R., TUTUIANU, B., PILEGGI, L. The Elmore Delay as a Bound for RC Trees with Generalized Input Signals. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, v. 16, n. 1, p. 95-104, January 1997.

HOON CHOI, S., PAUL, B., ROY, K. Novel Sizing Algorithm for Yield Improvement under Process Variation in Nanometer Technology. In: DESIGN AUTOMATION CONFERENCE, 41st, San Diego, USA. **Proceedings...** ACM Press, p. 454-459, June 2004.

JAEGER, R. **Introduction to Microelectronic Fabrication**. Upper Saddle River: Prentice Hall, 2002.

JESS, J., KALAFALA, K., NAIDU, S., OTTEN, R., VISWESWARIAH, C. Statistical Timing for Parametric Yield Prediction of Digital Integrated Circuits. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, v. 25, n. 11, November 2006.

JIANG, Y., SAPATNEKAR, S., BAMJI, C., KIM, J. Combined Transistor Sizing with Buffer Insertion for Timing Optimization. In: CUSTOM INTEGRATED CIRCUITS CONFERENCE, Santa Clara, USA. **Proceedings...** IEEE, p. 605-608, 1998.

KOUNO, T., HASHIMOTO, M., ONODERA, H. Input Capacitance Modeling of Logic Gates for Accurate Static Timing Analysis. **Asian Solid-State Circuits Conference**, p. 453-456, November 2005.

KORSHAK, A., LEE, J. An Effective Current Source Cell Model for VDSM Delay Calculation, 2001 **International Symposium on Quality Electronic Design**, p.296-300, 2001.

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

MAURINE et al. Transition Time Modeling in Deep Submicron CMOS. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**. v. 21, p. 1352-1363, November 2001.

MYERS, R. H., MONTGOMERY, D. C. **Response Surface Methodology**: Process and Product Optimization Using Designed Experiments. 2 Ed. John Wiley & Sons, Inc, 2002.

MODE, E. **Elements of Probability and Statistics**. Englewood Cliffs: Prentice-Hall, c1966.

NASSIF, S. Design for Variability in DSM Technologies. **IEEE 2000 First International Symposium on Quality Electronic Design**, p. 451-454, March 2000.

OKADA, K., YAMAOKA, K., ONODERA, H. Statistical Modeling of Gate Delay Variation with Consideration of Intra-Gate Variability. **IEEE Transactions on Computer-Aided Design**, v. 5, p. 513-516, May 2003.

OLIVIERI M., SCOTTI G., TRIFILETTI A. A novel yield optimization technique for digital CMOS circuits design by means of process parameters run-time estimation and body bias active control. **IEEE Transactions Very Large Scale Integration (VLSI) Systems,** v. 13, p. 630-638, May 2005.

ORSHANSKY, M., NASSIF, S., BONING, D. **Design for Manufacturability and Statistical Design**: A Constructive Approach. Springer, 2008.

PAN, M., CHU, C., ZHOU, H. Timing Yield Estimation Using Statistical Static Timing Analysis. **IEEE International Symposium on Circuits and Systems**, v. 3, p. 2461-2464, May 2005.

PELGROM, M., DUINMAIJER, A., WELBERS, A. Matching Properties of MOS Transistors. **IEEE Journal of Solid-State Circuits**, v. 24, n.5, October 1989.

PILEGGI, L. Timing Metrics for Physical Design of Deep Sub-Micron Technologies. In: INTERNATIONAL SYMPOSIUM ON PHYSICAL DESIGN, Monterey, USA. **Proceedings...** ACM Press, p. 28-33, April 1998.

PILLAGE, L., ROHRER, R. Asymptotic Waveform Evaluation for Timing Analysis. **IEEE Transactions on Computer-Aided Design**, v. 9, n. 4, pp. 352-366, April 1990.

QIAN, J., PULLELA, S., PILLAGE, L. Modeling the effective capacitance for the RC Interconnect of CMOS gates. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, v. 13, n. 12. December, 1994.

RABAEY, J.M.; CHANDRAKASAN. A.; NIKOLIC, B. **Digital Integrated Circuits**: A Design Perspective. 2nd ed. Upper Saddle River: Prentice Hall, 2005.

RAO, R., DEVGAN, A., BLAAUW, D., SYLVESTER, D. Parametric Yield Estimation Considering Leakage Variability. In: DESIGN AUTOMATION CONFERENCE, 41st, San Diego, USA. **Proceedings...** ACM Press, p. 442-447, June 2004.

RATZLAF, C., PULLELA, S., PILLAGE, L. Modeling the RC-interconnect effects in a hierarchical timing analysis. In: CUSTOM INTEGRATED CIRCUITS CONFERENCE, Boston, USA. **Proceedings...**, IEEE, p. 1561-1564. May, 1992.

REIS, A.I. **Assignation Technologique sur Bibliotheques Virtuelles de Portes Complexes CMOS**. 1998. 123 f. These (Doctorate m Electronique, Optronique et Systemes) - Université de Montpellier, Montpellier, France.

ROSA JR., L. **Automatic Generation and Evaluation of Transistor Networks in Different Logic Styles**. 2008. PhD. Thesis, Universidade Federal do Rio Grande do Sul.

ROY, K., MUKHOPADHYAY, S., MAHMOODI-MEIMAND, H. Leakage Current Mechanisms and Leakage Reduction Techniques in Deep-Submicrometer CMOS Circuits. **Proceedings of the IEEE...**, v. 91, n. 2, February 2003.

SAKURAI, T., NEWTON, R. Alpha-Power Law MOSFET Model and its Applications to CMOS Inverter Delay and Other Formulas. **IEEE Journal of Solid-State Circuits**, v. 25, No 2, April 1990.

SAKURAI, T., NEWTON, R. Delay Analysis of Series-Connected MOSFET Circuits. **IEEE Journal of Solid-State Circuits**, v. 26, No 2, February 1991.

SAPATNEKAR, S. **Timing**. Kluwer Academic Publishers, Boston, MA, 2004.

SAPATNEKAR, S., KANG, S. **Design Automation for Timing-Driven Layout Synthesis**. Kluwer Academic Publishers, 1993.

SASAO, T. **Switching Theory for Logic Synthesis**. Boston: Kluwer Academic, 2000.

SHAO, M., WONG, M., CAO, H., GAO, Y., YUAN, L., HUANG, L., LEE, S. Explicit Gate Delay Model for Timing Evaluation. In: INTERNATIONAL SYMPOSIUM ON PHYSICAL DESIGN, Monterey, USA. **Proceedings...** ACM Press p. 32-38. April, 2003.

SINGH, A., MANI, M., ORSHANSKY, M. Statistical Technology Mapping for Parametric Yield. In: INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN, San Jose, USA. **Proceedings...** ACM Press, p. 511-518, November 2005.

SRIVASTAVA, A, SYLVESTER, D., BLAAUW, D. **Statistical Analysis and Optimization for VLSI: Timing and Power**. New York, NY, USA: Springer, 2005.

SYNOPSYS. **CCS Timing: Technical White Paper**. Synopsys Inc., 2005. Available at: http://www.opensourceliberty.org/ccspaper/ccs_timing_wp.pdf

TAUR, Y., NING, T. **Fundamentals of Modern VLSI Devices**. New York: Cambridge Univ. Press, 1998.

VISWESWARIAH, C., RAVINDRAN, K., KALAFALA, K., WALKER, S., NARAYAN, S. First-Order Incremental Block-Based Statistical Timing Analysis. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, v. 25, No. 10, p. 2170-2180, October 2006.

XUEMEI, X. et al. **BSIM4.3.0 MOSFET Model: User's Manual**. The Regents of the University of California, 2003.

WAGNER, F., REIS, A., RIBAS, R. **Fundamentos de Circuitos Digitais**. Porto Alegre, RS: Sagra Luzzatto, 2006.

WEBEL, L. F., BAMPI, S. A Timing Model for VLSI CMOS Circuits Verification and Optimization. **IEEE International Symposium on Circuits and Systems**, v. 1, p. 439-442, June 2004.

WESTE, N.H.E.; HARRIS, D. **CMOS VLSI Design**: A Circuits and Systems Perspective. 3rd ed. Boston: Pearson/Addison-Wesley, 2005.

ZHAO W., CAO Y. New generation of Predictive Technology Model for sub-45nm early design exploration. In: INTERNATIONAL SYMPOSIUM ON QUALITY ELECTRONIC DESIGN, 7th, San Jose, USA. **Proceedings...** IEEE, p. 585-590, 2006.

ZHAO, W. Predictive Technology Model, 2007. Available at: http://www.eas.asu.edu/~ptm/.

APPENDIX A RESISTANCE EQUATIONS SCRIPT

```
%Results for o Transistor 1 - R1
%Linear Regression

fid = fopen('rsm_nmos_nand_3_best_case_linear.txt','w');

x11 =  [-1  1 -1  1 -1  1 -1  1]';
x21 =  [-1 -1  1  1 -1 -1  1  1]';
x31 =  [-1 -1 -1 -1  1  1  1  1]';

y1=[2.3785E+03
2.3283E+03
2.3688E+03
2.3227E+03
2.7198E+03
2.6461E+03
2.7057E+03
2.6382E+03];

X1=[ones(size(x11)) x11 x21 x31];

a1=pinv(X1)*y1
Y1=X1*a1;
MaxErr1=max(abs(Y1-y1))

%Resultados para o Transistor 2 - R2

x12 =  [-1  1 -1  1 -1  1 -1  1]';
x22 =  [-1 -1  1  1 -1 -1  1  1]';
x32 =  [-1 -1 -1 -1  1  1  1  1]';

y2=[2.5696E+03
2.4499E+03
3.0256E+03
2.8374E+03
2.5825E+03
2.4580E+03
3.0484E+03
2.8513E+03];

X2=[ones(size(x12)) x12 x22 x32];

a2=pinv(X2)*y2
Y2=X2*a2;
MaxErr2=max(abs(Y2-y2))

%Resultados para o Transistor 3 - R3
```

```
x13 =  [-1  1 -1  1 -1  1 -1  1]';
x23 =  [-1 -1  1  1 -1 -1  1  1]';
x33 =  [-1 -1 -1 -1  1  1  1  1]';


y3=[7.9810E+03
1.2422E+04
8.1542E+03
1.2707E+04
8.1162E+03
1.2664E+04
8.2892E+03
1.2947E+04];

X3=[ones(size(x13)) x13 x23 x33];

a3=pinv(X3)*y3
Y3=X3*a3;
MaxErr3=max(abs(Y3-y3))


fprintf(fid,'\n %3.5f \n', a1,a2,a3,MaxErr1,MaxErr2,MaxErr3)


fclose(fid)
```

APPENDIX B DELAY CALCULATION SCRIPT

```
%Modeling the transistor using ON resistance
%NMOS transistors – Simulation NAND3
```

%Parameters provided by the technology files

```
%Impurities concentrations
Na = 2e20;
Nd = 2e20;

%%%%%TOXE: electrical gate equivalent oxide thickness

TOXEn = 1.75e-9;
TOXEp = 1.85e-9;

%Temperature (K)
T = 300;

%electron charge
q = 1.6e-19;

%%%%k: Boltzmann constant

k = 8.6173e-5;
kb = 1.38e-23;

%%%%NDEP: Channel doping concentration at depletion edge for zero body bias

NDEPn = 3.24e18;
NDEPp = 2.44e18;

%%%%Nsub: Nsubstrate

Nsub = 6e16;

%%%%Energy band gap of Si

Eg = 1.16-((7.02e-4)*(T^2)/(T+1108));

%%%%ni: intrinsec carrier concentration

ni = 1.45e10*(T/300.15)*(sqrt(T/300.15))*exp(21.5565981-((q*Eg)/(2*kb*T)));

%%%%EPSROX: gate dielectric constant relative to vacuum

EPSROX = 3.9;
```

```
eo = 8.85e-12;

%%%%%Coxe: effective oxide capacitance (used to calculate vth)

Coxen = EPSROX*eo/TOXEn;
Coxep = EPSROX*eo/TOXEp;

Esi = 11.9*eo;

%%%%Source-bulk effective potential

Vbseffp = 0;
Vbseffn = 0;

%%%%%Gate-source effective potential (just a trial)

%Effective channel length

Leffp = 37e-9;
Leffn = 37e-9;

fid = fopen('nmos_nand_3_stack3a_delay_elmore.txt','w');

%Drain length

Ldp = (Leffp)/2;
Ldn = (Leffn)/2;

%Source length

Lsp = (Leffp)/2;
Lsn = (Leffn)/2;

%Channel Width

Wpload = 5*(90e-9);
Wn = 135e-9;
Wnload = 5*(45e-9);
Wp = 90e-9;

%************************************************************
```

%CAPACITANCES

```
%Gate capacitances for a fan-out inverter

Cgn = Coxen*Wnload*Leffn;
Cgp = Coxep*Wpload*Leffp;

%Cgs and Csb series capacitance (Cgbs)
%%%Drain and source area under gate (xd)

%Overlap capacitance
%Cjo: junction capacitance at zero bias
%MJ: bottom junction capacitance grading coefficient
%PB: built-in potential

Cjo = 5e-4;
```

```
PB = (kb*T/q)*log(Nsub*Nd/(ni^2));

MJ = 0.5;

Cjn = Cjo/((1+(Vbseffn/PB))^MJ);

Cjp = Cjo/((1+(Vbseffp/PB))^MJ);

Cgdo = 1.1e-10;
Cgso = 1.1e-10;

%Cgd and Cdb series capacitance (Cgbd)

%Cgbd = Coxen*Cj*Wn*xd/(Coxen+Cj)

Cgdn = Cgdo*Wn;
Cgdp = Cgdo*Wp;
Cgsn = Cgso*Wn;
Cgsp = Cgso*Wp;

%Cdb drain-bulk capacitance (non-overlaped region)

Cdbn = Cjn*Wn*(Ldn);
Cdbp = Cjp*Wp*(Ldp);

%Csb source-bulk capacitance (non-overlaped region)

Csbn = Cjn*Wn*(Lsn);
Csbp = Cjp*Wp*(Lsp);

%Side-wall junction

Csjwo = 5e-10;
MJSW = 0.33;

Csjwn = Csjwo/((1+(Vbseffn/PB))^MJSW);
Csjwp = Csjwo/((1+(Vbseffp/PB))^MJSW);

Cswdn = Csjwn*(2*Ldn+Wn);
Cswdp = Csjwp*(2*Ldp+Wp);

Cswsn = Csjwn*(2*Lsn+Wn);
Cswsp = Csjwp*(2*Lsp+Wp);

%Channel capacitance for linear region

Cgchnn = (Coxen*Wn*Leffn)*1/2;
Cgchnp = (Coxep*Wp*Leffp)*1/2;
```

%Total capacitance

```
%Considering Cgn and Cgp the gate capacitances of a fanout inverter
%linear region

CL = Cgn + Cgp
C3 = Cgchnn  + Cgdn + Cdbn + Cswdn + 3*(Cgdp + Cdbp + Cswdp)
C2 = 2*Cgchnn + Cgsn + Cgdn + Cdbn + Csbn + Cswdn + Cswsn
C1 = 2*Cgchnn + Cgsn + Cgdn + Cdbn + Csbn + Cswdn + Cswsn
C3L = C3 + CL
```

```
%threshold voltage variations as coded variables (-1,1)

for x3 = -1:2:1
      for x2 = -1:2:1
                for x1 = -1:2:1

%Resistances Equations

R1 = 2513.51250 - (29.68750*x1) - (4.66250*x2) + (163.93750*x3);
R2 = 2727.83750 - (78.68750*x1) + (212.83750*x2) + (7.21250*x3);
R3 = 10410.07500 + (2274.92500*x1) + (114.27500*x2) + (94.02500*x3);

%Elmore Delay
T = (R1*C1) + (R1+R2)*C2 + (R3+R2+R1)*C3L

fprintf(fid,'\n %3.20e \n', T)

end
      end
                end

fclose(fid)
%************************************************
```

APPENDIX C DELAY EQUATION SCRIPT

```
%Delay as a function of Vth variations of the transistors in the network
%Delay Results
%Linear Regression

fid = fopen('rsm_nmos_nand_3_delay_best_case_elmore.txt','w');

x11 =  [-1  1 -1  1 -1  1 -1  1]';
x21 =  [-1 -1  1  1 -1 -1  1  1]';
x31 =  [-1 -1 -1 -1  1  1  1  1]';


y1=[1.20561912955632150000e-011
 1.54555906234382130000e-011
 1.26683484373539300000e-011
 1.60677477652289310000e-011
 1.26391275425000020000e-011
 1.60385268703750010000e-011
 1.32512846842907170000e-011
 1.66506840121657160000e-011];

X1=[ones(size(x11)) x11 x21 x31];

a1=pinv(X1)*y1
Y1=X1*a1;
MaxErr1=max(abs(Y1-y1))


fprintf(fid,'\n %3.20e \n', a1,MaxErr1)


fclose(fid)
```

APPENDIX D RESUMO DA TESE EM PORTUGUÊS

Método de Estimativa da Variabilidade do Atraso de Portas Lógicas em Tecnologia CMOS

INTRODUÇÃO

Qualquer processo de produção apresenta um certo grau de variabilidade em torno do valor nominal das especificações do produto. Este fenômeno se traduz na descrição das características do produto na forma de uma faixa de variação aceitável para cada um dos seus parâmetros críticos. A redução das dimensões da tecnologia MOS faz com que a variabilidade intrínseca em sua fabricação aumente. As tolerâncias de processo não se escalam proporcionalmente com as dimensões de concepção, fazendo com que o impacto relativo das variações de dimensão crítica aumente a cada geração de novas tecnologias (ARGAWAL, 2007). Este cenário exige abordagens realistas que são capazes de prever o impacto das variações dos parâmetros nas métricas do circuito.

Uma vez que as variações de processo tornam-se uma questão mais crítica, a transição de uma análise determinística para uma análise estatística de projetos de circuitos pode reduzir o conservadorismo da abordagem tradicional. A técnica tradicional de análise estática de atraso (STA) é uma forma razoável de se lidar com as variações globais, mas não as locais. No caso do desempenho de um circuito, uma porta lógica pode tornar-se mais lenta para uma certa variação de parâmetros e mais rápida para outra variação, o que pode depender de sua localização em um circuito. Não apenas a importância das variações *intra-die* tem crescido, mas também o número de parâmetros de processo que apresentam variações consideráveis também aumentou (Srivastava, 2005).

Diferentes redes de transistores apresentam diferentes características elétricas, ainda que representem a mesma função lógica (ROSA, 2008). Nesse sentido, o objetivo deste trabalho é propor um método de estimativa de atraso que leva em conta a variabilidade da tensão de limiar de transistores em uma rede específica. O primeiro passo foi analisar como a variabilidade dos parâmetros afeta as métricas das portas lógicas de acordo com (i) a topologia (a quantidade de transistores em série e em paralelo) e (ii) a posição do transistor com um sinal de entrada transitória em relação ao nó de saída. Simulações elétricas das portas foram realizadas com HSPICE, um simulador de circuito que é considerado referência para os circuitos elétricos. Um método de estimativa para a variação de atraso foi proposto e avaliado. Procurou-se desenvolver um método simples de se estimar a variabilidade de atraso, levando-se em consideração diferentes portas lógicas e diferentes arranjos de transistores. O método inclui a utilização do modelo de atraso de Elmore (ELMORE, 1948) e a Avaliação de Forma de Onda Assintótica (AWE) (PILLAGE, 1990), considerando-se as resistências dos transistores obtidos em função de variações de tensão de limiar de transistores no arranjo. As resistências e capacitâncias são funções dos parâmetros dos transistores. Os dados coletados através de simulação elétrica são comparados aos resultados gerados pelo método proposto. Os

resultados da análise podem ser utilizados para se alcançar implementações do circuito que são relativamente imunes à variabilidade.

PROJETO DIGITAL E VARIABILIDADE DE PARÂMETROS EM CIRCUITOS INTEGRADOS

As portas lógicas de um circuito são constituídas basicamente por transistores conectados a fim de se realizar uma função lógica. De uma forma bastante simplicada, um transistor pode ser concebido como um interruptor controlado pelo sinal aplicado ao terminal de porta. Um transistor NMOS está "ligado" quando o sinal de controle é elevado e "desligado" quando o sinal de controle é baixo. Um transistor PMOS age no sentido oposto, sendo "ligado" quando o sinal é baixo e "desligado" quando o sinal é alto.

Uma rede de transistor é um conjunto de dispositivos interligados atuando como chaves para implementar determinadas funções booleanas. Diferentes equações booleanas podem representar a mesma função booleana. A questão da síntese lógica é descobrir a melhor equação para uma função lógica dada. Os critérios de otimização do projeto de um circuito estão relacionados à aplicação da equação de portas lógicas e podem ser destinados a minimizar algumas características do circuito, como área, consumo de energia ou atraso de propagação.

Um circuito pode ter suas características melhoradas através da otimização das suas redes de transistor. A otimização da rede pode ser alcançada através da reorganização dos dispositivos de acordo com algumas regras para minimizar um determinado custo. No caso de se tentar reduzir o atraso de propagação em um circuito, pode acontecer que alguns sinais em blocos de lógica combinacional são mais críticos do que outros e nem todas as entradas de uma porta transicionam ao mesmo tempo. Colocar os transistores que estão no caminho crítico mais próximos da saída da porta pode resultar em uma redução do tempo de propagação do sinal.

A preocupação principal deste trabalho é realizar implementações que resultem em atraso de propagação de sinal com elevada imunidade às variações nos parâmetros dos dispositivos.

Fluxo de Projeto de Células-Padrão

Um fluxo de projeto é um conjunto sistemático de procedimentos que possibilita a implementação de um chip de acordo com as especificações de uma forma que se evite erros. Concepção de ASIC digital começa a nível comportamental e, em seguida, passa para o nível estrutural (portas e registradores), que é chamado de *Register Transfer Level* (RTL), usando uma linguagem de descrição de hardware (HDL). Ferramentas de síntese lógica traduzem módulos descritos em uma linguagem HDL gerando assim um

netlist, que é uma descrição das células-padrão a ser usadas mais as conexões elétricas necessárias entre elas. Como parte da etapa de síntese lógica, o mapeamento tecnológico é o processo de se implementar uma determinada rede *booleana* em termos de células lógicas ou portas. Normalmente, visa-se à utilização otimizada das portas lógicas de uma biblioteca para se implementar um circuito com menor atraso em seu caminho e menor área possível. As técnicas mais comuns para o mapeamento tecnológico baseiam-se em bibliotecas de células pré-caracterizadas.

Variabilidade em Circuitos Integrados

Uma maneira de se classificar as variações em um circuito é de acordo com a natureza da variação (Srivastava, 2005): (i) variações de processo, (ii) variações ambientais e (iii) variações de modelamento.

Variabilidade de Parâmetros de Processo

Os parâmetros de transistores que são mais suscetíveis à variações são o número e a distribuição dos átomos dopantes, o comprimento efetivo do canal do transistor, a largura e a espessura do óxido de porta. No caso de interconexões, a largura e a espessura da linha de metal são os parâmetros que sofrem maiores variações.

Variabilidade das Características dos Dispositivos

A tensão de limiar do dispositivo (V_{TH}) é um parâmetro determinado pelo material que implementa a porta (*gate*), pela espessura da camada de dióxido de silício, pela concentração e pelo perfil da densidade dos átomos dopantes no canal do transistor, entre outras características do processo. V_{TH} é principalmente afetada pela variação no número e na distribuição dos átomos dopantes ao longo do material e as variações na espessura do óxido. Conforme os dispositivos diminuem drasticamente de tamanho, o número de átomos dopantes por transistor pode ser inferior a cem, o que diminui o nível de controle do número e da uniformidade desses átomos ao longo do canal. Nesta escala, um único átomo dopante pode alterar as características do dispositivo, resultando em grandes variações de dispositivo para dispositivo (ORSHANSKY, 2008).

ANÁLISE DE ATRASO

Este capítulo revisa o conceito de análise de atraso estática (STA). O uso de modelos estatísticos de análise de atraso (análise de atraso estática e estatística - SSTA) também é apresentada. Esses conceitos são importantes para se compreender o objetivo desta tese, que visa à concepção de um método para avaliar as características estatísticas de redes a nível de células lógicas para ser usado em nível de circuito. A intenção deste capítulo é fornecer o contexto no qual o trabalho está inserido e para mostrar alguns dos recentes esforços para lidar com variabilidade em projeto digital.

Statistiscal Static Timing Analysis (SSTA)

A extensa pesquisa sobre análise de atraso em circuitos lógicos mostra a necessidade de um novo método que seja capaz de lidar com as seguintes questões (ARGAWAL, 2003): (i) a capacidade de controlar os parâmetros críticos do dispositivo tornam-se limitado, uma vez que as dimensões de processo continuam a encolher, o que resulta em variações significativas destes parâmetros, (ii) o número total de parâmetros de processo que apresentam variações aumentou, fazendo com que o número de "*corners*" (valores que especificam as características da porta para cada condição de processo) aumentará rapidamente, (iii) variações *intra-die* tornaram-se uma componente significativa da variação total, e (iv) além dos parâmetros do dispositivo, os parâmetros de interconexão devem ser considerados.

A formulação determinística da análise de tempo trata o atraso das portas e caminhos como números fixos, reduzindo assim um problema probabilístico a um problema aritmético. Em SSTA as métricas de desempenho das portas lógicas e das interconexões são modelados a partir de valores estocásticos, resultando em funções de densidade de probabilidade (PDF). Na análise estatística não há sentido em se tentar identificar um único caminho do circuito como sendo o caminho crítico, ou o caminho com o máximo de atraso. Caminhos críticos são, então, definidos como um conjunto de caminhos com alta probabilidade de se tornarem o caminho mais lento no circuito (SRIVASTAVA, 2005).

PROPOSTA E METODOLOGIA

Introdução

O desempenho e o consumo de energia de um circuito integrado é afetado pelas variações do processo de fabricação (comprimento e largura do canal, espessura do óxido de porta, concentração e distribuição de átomos dopantes etc). Em relação ao processo de fabricação MOS, os efeitos das variações não escalam proporcionalmente com as dimensões de concepção, fazendo com que o impacto relativo das variações de dimensão crítica aumentem a cada nova tecnologia (MAHMOODI, 2005).

Os parâmetros físicos são suscetíveis a variações aleatórias e a natureza estatística das características do processo faz com que seja possível considerar os parâmetros do processo e suas variações como variáveis aleatórias representadas pela suas funções de densidade de probabilidade (PDF). Em (MAHMOODI, 2005) as PDFs dos atrasos de portas lógicas são estimadas considerando-se as variações de tensão de limiar devido às flutuações aleatórias dos átomos dopantes. O desvio padrão da tensão de limiar é modelado como dependente das dimensões dos transistores (comprimento do canal e largura, espessura do óxido de porta etc) e da concentração de dopantes.

Os objetivos gerais do presente trabalho são: (i) a análise da variabilidade do atraso de diferentes redes de transistores e (ii) a concepção de um método capaz de estimar a variabilidade de determinadas topologias sem a necessidade de se fazer uso de simulações computacionalmente caras. A primeira parte do estudo se baseia em simulações estatísticas (Monte Carlo), e destina-se a lançar uma luz sobre o comportamento da variabilidade do desempenho de estruturas básicas de transistores e de portas lógicas. A segunda parte, principal foco deste trabalho, é a implementação de um método semi-empírico que descreve e prediz a variabilidade de acordo com diferentes arranjos de transistores.

VARIABILIDADE NA PERFORMANCE DE PORTAS LÓGICAS CMOS

Introdução

As análises apresentadas neste capítulo são resultados da pré-caracterização de algumas células lógicas e compreendem a primeira etapa deste trabalho. As caracterizações proporcionaram um conhecimento inicial sobre o comportamento das redes sob o efeito de variações de tensão de limiar de seus transistores. Esta informação pode ser útil no desenvolvimento de diretrizes de projeto para a melhoria de rendimento paramétrico. Foi avaliado o impacto das variações na tensão de limiar do transistor no comportamento do atraso de portas lógicas CMOS, de acordo com (i) a topologia da rede (arranjos de transistores) e (ii) a posição relativa do transistor que possui um sinal transiente em sua entrada em relação à fonte de alimentação e aos terminais de saída.

Simulações elétricas foram realizadas a fim de mostrar a variabilidade no atraso de um inversor CMOS, e portas lógicas NAND, NOR e AOI (AND-OR-INVERSOR). As tensões de limiar (V_{TH}) dos transistores foram variadas e as medidas de atraso foram realizadas para todas as configurações. O atraso médio e desvio padrão das portas lógicas foram comparados e a relação desses valores com a rede de transistor foi enfatizada. Os atrasos de subida e descida das portas - inversor, NAND e NOR de 2, 3 e 4 entradas, e configurações AOI-21 e AOI-32 - foram verificadas a partir de simulações estatísticas (Monte Carlo). Dez mil simulações foram realizadas para cada experimento. As medidas foram feitas para um desvio 3σ de 10% da tensão de limiar nominal dos transistores. Os valores de desvio padrão normalizado (σ/μ) das métricas das diferentes configurações foram comparados e analizados. O desvio padrão normalizado torna possível comparar a variabilidade de arranjos com diferentes valores médios de atraso. O nó tecnológico utilizado neste trabalho é de 45 nm e o arquivo de modelo inserido no netlist das simulações foi fornecido pelo *Predictive Tecnology Model* (PTM) (ZHAO, 2007), com base no BSIM4. Nenhuma correlação entre os diferentes tipos de transistores foi levada em consideração, o que significa que um PMOS colocado na proximidade de um NMOS pode apresentar variações diferentes de um mesmo parâmetro. As simulações foram realizadas utilizando-se a ferramenta HSPICE.

MÉTODO DE ESTIMATIVA DA VARIABILIDADE DE ATRASO

O presente trabalho propõe a implementação de um método capaz de prever a variabilidade no atraso de uma porta lógica. Uma única variável - a variação da tensão limiar – é atribuída a cada transistor da porta lógica analisada. Sempre que possível, a porta é dividida em redes *pull-up* e *pull-down*, o que reduz o número de dispositivos e variáveis com os quais se é preciso lidar.

O método de fatoriais (MYERS, 2002) é utilizado para se identificar os principais efeitos dos fatores (ΔV_{TH}) sobre as primeiras variáveis-alvo, ou seja, sobre as resistências dos transistores. Uma vez que a cada variável de interesse são atribuídos dois níveis (valores codificados -1 e +1), cada variante de tal projeto tem 2^k simulações DC a serem executadas, o que é chamado de projeto fatorial 2^k, onde 'k' é o número de transistores na porta lógica a ser analisada ('k' limiares de tensão).

Inicialmente, simulações elétricas do tipo DC são realizadas para extrair as 'resistências-ON" de transistores para diferentes tensões de limiar (ΔV_{TH}) de acordo com a metodologia de 2^k casos. O V_{TH} de transistores são variados em ± 10% de seus valores nominais e transformados em variáveis codificadas (normalizadas) com valores -1 e +1, que representam os valores mínimo e máximo assumidos pela V_{TH}, respectivamente. Para o exemplo apresentado na Figura 6.1, as resistências são fornecidas por 2^3 (sendo três transistores) simulações DC, dividindo-se a queda de tensão em cada transistor pela corrente que passa através dele.

O efeito estimado em um método fatorial com 2^k casos é então convertido em um modelo de regressão (função de primeira ordem para a resistência do dispositivo), que pode ser usado para analisar a resposta (resistência) em qualquer ponto do espaço gerado pelos fatores codificados (ΔV_{TH}) no projeto.

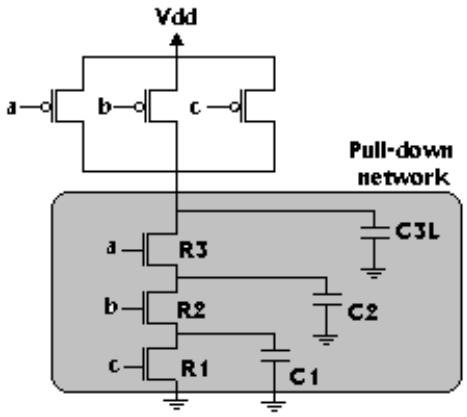


Figura 6.6: Rede *pull-down* de uma porta lógica NAND com 3 entradas.

A técnica de regressão linear é o próximo passo no desenvolvimento dos modelos aproximados para as resistências dos transistores em função do ΔV_{TH} de todos os dispositivos na rede. A resposta dinâmica do transistor MOS é uma função do tempo. Leva-se em conta a carga e a descarga das capacitâncias parasitas que são intrínsecas ao dispositivo, e das capacitâncias introduzidas pela interconexões e pela carga. As capacitâncias intrínsecas são originárias de três fontes: (i) da estrutura básica MOS, (ii) da carga do canal e (iii) das regiões de depleção das junções *pn* das regiões de fonte e dreno (Rabaey, 2005). As capacitâncias intrínsecas são calculadas para uma tecnologia específica e são consideradas como constantes no método proposto. O atraso é calculado através da constante RC da rede usando o método fatorial. Um modelo de regressão de primeira ordem representa o atraso do arranjo de transistores como uma função das variáveis ΔV_{TH} codificadas.

Dois métodos diferentes foram utilizados para se calcular a variação do atraso para as redes de transistores: (i) o modelo de atraso de Elmore (ELMORE, 1948) e (ii) a avaliação de forma de onda assintótica (AWE) (PILLAGE, 1990). As equações lineares que representam as resistências são utilizadas para realizar a análise. As variações da tensão limiar dos transistores são consideradas como variáveis aleatórias representadas por funções de densidade de probabilidade (PDF) com valores médios iguais a zero e desvio padrão normalizado [N (0,1)]. Uma vez que as variações de V_{TH} são tratadas como variáveis codificadas, o coeficiente da equação de atraso final é considerado como o desvio padrão (σ) resultante de cada ΔV_{TH}. Em seguida, é realizada uma soma de distribuições normais, e o resultado é uma PDF que representa o atraso da rede com os dois aspectos considerados: média e desvio padrão (raiz quadrada da variância). O desvio padrão normalizado (desvio padrão dividido pela média) do atraso é o foco da análise de redes diferentes, pois torna possível comparar a variabilidade de diferentes arranjos com diferentes atrasos médios.

MODELAMENTO DA VARIABILIDADE DE ATRASO DE REDES DE TRANSISTORES

Este capítulo apresenta os resultados obtidos utilizando-se o método de estimativa proposto para o cálculo do desvio do atraso das diferentes redes de transistores. Simulações elétricas do tipo DC foram realizadas com HSPICE. Os outros cálculos foram realizados com a ferramenta MATLAB ® (Matrix Laboratory).

Em uma topologia com transistores NMOS em série, o transistor mais próximo do nó de saída tem uma resistência maior do que os outros quando todos os dispositivos são considerados como *ligados* (conduzindo corrente). Embora a tensão da porta de entrada esteja no mesmo nível para todos os dispositivos, o mesmo não acontece com a tensão porta-fonte. No caso de um arranjo em série, os dispositivos têm as suas fontes com um potencial igual a V_{DD} - V_{TH}, exceto para o dispositivo na parte inferior do empilhamento, que tem sua fonte ligada ao nó *terra*. O terminal de fonte dos dispositivos no topo do empilhamento passa mais tempo perto de V_{DD} - V_{TH} e menos tempo conduzindo fortemente do que os outros dispositivos. Isso resulta em um V_{TH} efetivo maior devido ao efeito de corpo, o que resulta em maior resistência efetiva.

O uso do modelo de atraso de Elmore (ELMORE, 1948) permite realizar uma operação de soma direta de PDFs para calcular as resistências equivalentes das redes de transistores, e em seguida a constante de tempo RC. O método de avaliação de forma de onda assintótica (AWE) requer o cálculo das matrizes de condutâncias e de capacitâncias a partir das equações de estado do circuito considerado. A partir das matrizes, os momentos são encontrados e combinados via aproximação de Padé, resultando em modelos de forma reduzida. Em ambos os métodos, é necessário calcular o atraso para os valores de casos (casos fatoriais) de x_1, x_2, x_3 e x_4 (coded ΔV_{TH}) e ajustar uma equação.

Simulações anteriores com transistores NMOS em série revelaram menor desvio de atraso para o caso em que o transistor que possui um sinal transiente em sua entrada está próximo do nó de saída do que para o caso em que o transistor se encontra longe do nó de saída. Para uma arranjo com transistores PMOS em série, maior imunidade à variação das tensões de limiar é obtida quando o transistor de chaveamento está longe do terminal de saída. Os modelos semi-empíricos obtidos para as resistências e utilizados para calcular o desvio do atraso nas redes também mostraram a mesma tendência para a PMOS, mas não para a rede NMOS.

Os resultados revelaram que, em geral, a posição do transistor que chaveia impacta mais fortemente os arranjos em série NMOS do que os arranjos em série PMOS. O

modelo proposto para as resistências dos transistores em uma rede permite investigar como as variações na tensão de limiar de cada dispositivo influenciam as resistências de todos os dispositivos do arranjo. Ao se utilizar as funções de resistência também é possível analisar a variabilidade do desempenho de acordo com o estado e posição dos dispositivos na rede. Apesar do modelo de atraso de Elmore apresentar algumas limitações, ele é plausível de ser utilizado para se realizar a análise.

AVALIAÇÃO DO MÉTODO DE ESTIMATIVA DA VARIABILIDADE DE ATRASO

Neste capítulo, o método de estimativa proposto foi utilizado para avaliar o desvio de atraso de portas implementadas em diferentes topologias e estilos lógicos. O caminho de propagação do sinal foi simplificado a fim de se aplicar as resistências para o modelo de atraso de Elmore (ELMORE, 1948) no cálculo da variação do atraso. Isso foi feito através da substituição das resistências em paralelo por uma resistência equivalente. A técnica de AWE (Pillage 1990) foi também utilizada em alguns casos. As células testadas compreendem uma porta lógica XOR de 2 entradas (XOR2), uma XOR de 4 entradas (XOR4) e um somador completo, que são estruturas usadas em bibliotecas de células para o mapeamento tecnológico.

Porta Lógica XOR de 2 Entradas

A XOR de 2 entradas foi implementada em diferentes topologias, como mostrado na figura. 8.1. O dimensionamento dos transistores foi realizado de forma a equilibrar a capacidade de corrente das diferentes redes através do projeto de dispositivos PMOS com área duas vezes maior do que a área dos dispositivos NMOS. O método proposto é usado para fornecer valores de variação no atraso. Simulações estatísticas Monte Carlo também foram realizadas para efeito de comparação. O objetivo é investigar como o método é confiável em prever a topologia a ser escolhida (ou ignorada) quando uma configuração mais robusta (com menor variação de atraso) é desejada. As configurações consideradas são: (i) uma única porta CMOS complexa, (ii) uma implementação com 4 portas NAND e (iii) uma implementação usando a lógica de transistor de passagem.

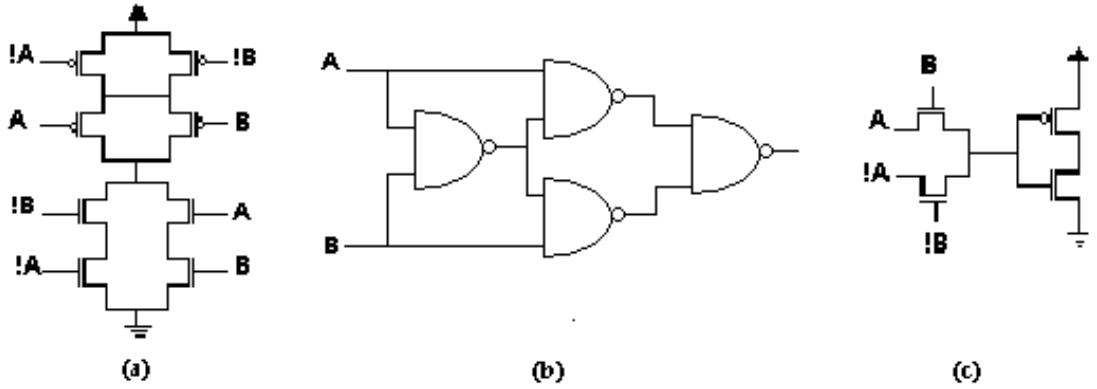


Figura 8.1: Porta lógica XOR de 2 entradas implementada em diferentes estilos lógicos e topologias: (a) porta complexa CMOS; (b) portas básicas CMOS (NAND) e (c) lógica de transistor de passagem (PTL).

O método foi capaz de prever a grande variabilidade no atraso de descida apresentado pelo estilo lógico PTL. Resultados fornecidos pelo método proposto e pelas simulações estatísticas apontaram a configuração com portas NAND como a mais imune às variações de V_{TH} quando ambos os atrasos de subida e descida são considerados.

O maior erro apresentado pelo método foi para o desvio de atraso de descida da porta CMOS complexa, considerando as resistências modeladas como sendo aplicadas ao modelo de atraso de Elmore. No entanto, a técnica AWE também foi testada para este caso e um melhor resultado (0,0388), com um erro de 71,7% foi alcançado.

Porta Lógica XOR de 4 Entradas

Fig. 8.10 apresenta uma XOR de 4 entradas implementada em diferentes estilos lógicos e topologias. Os modelos de resistências foram usados primeiramente no modelo de atraso de Elmore (ELMORE, 1948) e as análises que apresentaram maiores erros também foram realizadas através da técnica de AWE (PILLAGE 1990) .

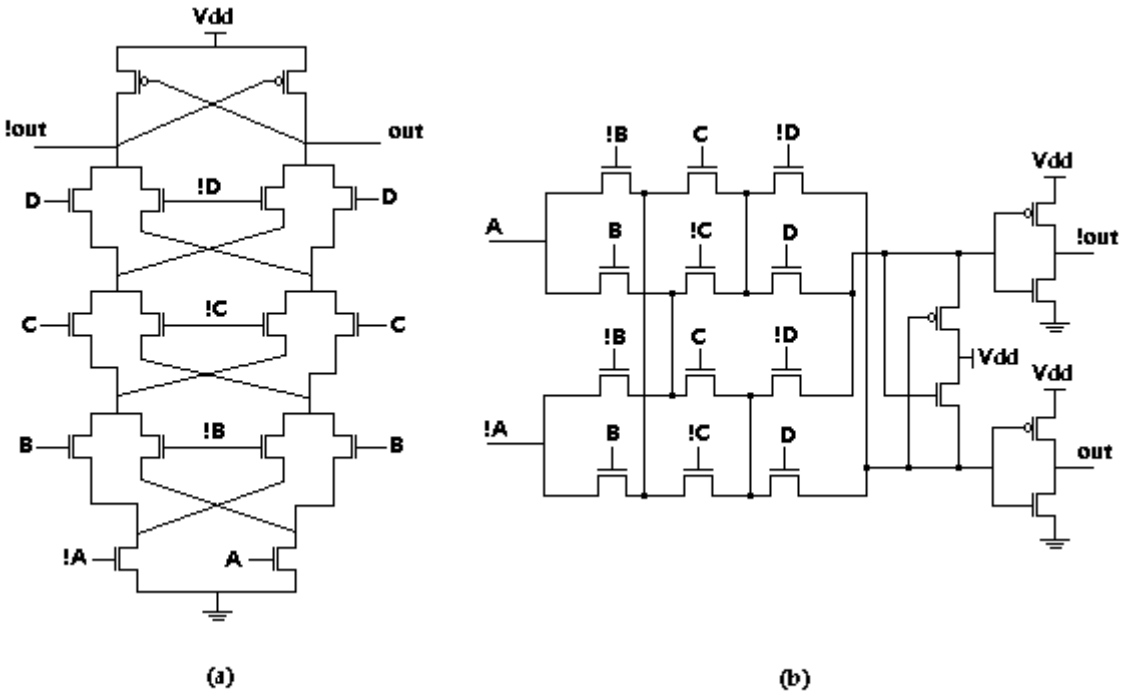


Figure 8.10: 4-input XOR implemented in different logic styles: (a) differential cascode voltage logic (DCVSL) and (b) pass-transistor logic (PTL).

Os melhores resultados (mais próximos dos valores simulados) apresentados pelo método foram encontrados para o estilo DCVSL, considerando-se ambos os atrasos de subida e descida. O método foi capaz de prever a configuração menos robusta entre as que foram utilizadas para implementar a porta XOR de 4 entradas. O maior erro apresentado pelo modelo foi para o desvio de atraso de descida do estilo DPTL considerando-se as resistências aplicadas ao modelo de atraso de Elmore. No entanto, o método foi testado com a técnica AWE para este caso e um melhor resultado (0,1314) foi alcançado, apresentando um erro de apenas 10,3%.

Somador Completo

O somador completo é uma célula lógica com três entradas e duas saídas (Weste 2005).

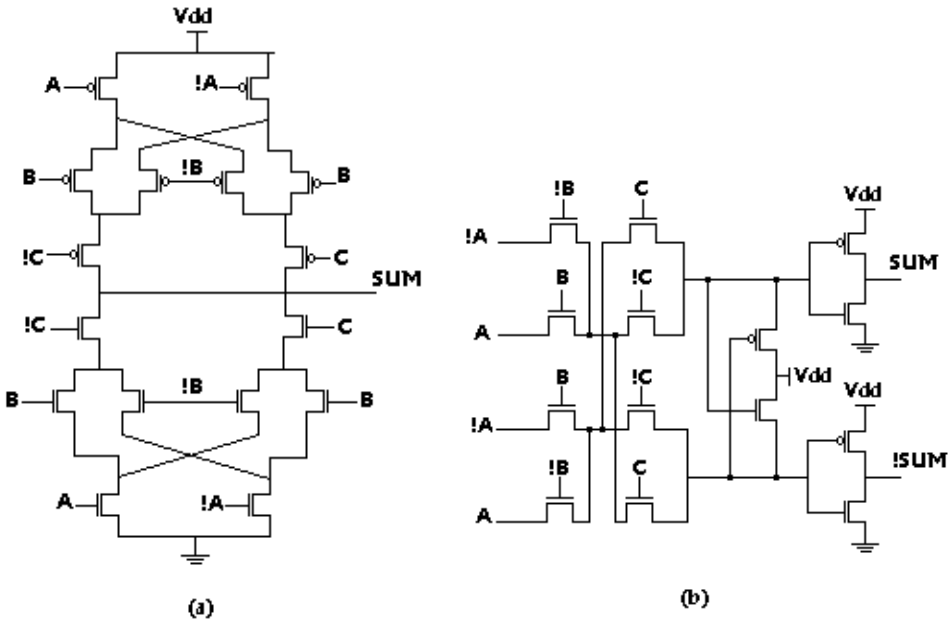


Figura 8.18: Somador completo implementado em diferentes estilos lógicos: (a) porta complexa CMOS; (b) lógica diferencial de transistores de passagem (DPTL).

A rede NMOS da porta CMOS complexa é mais suscetível às variações de tensão de limiar que a rede PMOS, como visto no CAPÍTULO 7. O método proposto está de acordo com os resultados da simulação no se refere ao desvio do atraso de descida, mostrando que a topologia DPTL é a menos robusta dentre as topologias analisadas.

O erro apresentado pelo método quanto ao desvio de atraso de descida do estilo DPTL considerando as resistências aplicadas ao modelo de atraso Elmore foi reduzido para 17,2% quando as resistências foram utilizados com a técnica AWE. Neste caso, o desvio foi 0,1016.

Análise do Tempo de Simulação

A caracterização das portas lógicas a partir de simulações estatísticas Monte Carlo é computacionalmente cara e leva muito tempo para ser completada, o que a torna proibitiva em alguns casos. A abordagem apresentada neste trabalho para caracterizar as células requer simulações que levam apenas alguns segundos, uma vez que apenas uma simulação é transiente e todas as outras são DC.

Método da Equação de Atraso

O aqui chamado Método da Equação de Atraso (DEM) também usa o método dos mínimos quadrados a fim de obter os coeficientes do modelo de primeira ordem (equação) do atraso como uma função das variações de tensão de limiar dos transistores. A diferença desse método em relação ao método de estimativa proposto neste trabalho é que, basicamente, o primeiro exige simulações transientes ao invés de simulações do tipo DC. Nesse sentido, o DEM não utiliza a constante RC da rede a fim de proporcionar o atraso da célula lógica, pois o atraso é a métrica diretamente fornecida pelas simulações transientes.

Variações deste método já estão disponíveis na literatura (OKADA 2003) e é por isso que a análise fornecida pelo método de estimativa proposto é comparado com o DEM neste capítulo.

CONCLUSÃO

Diferentes topologias de inversores apresentam diferentes desvios de atraso quando as tensões de limiar dos transistores variam. A análise preliminar com base em simulações Monte Carlo mostrou que a escolha da topologia e estilo lógico, o número de transistores na rede e a posição dos transistores com sinais transientes no terminal de entrada influenciam a variabilidade de atraso da célula lógica. Quando apenas uma parte do arranjo de transistores é analisada, como o *pull-up* ou *pull-down* em uma configuração CMOS, por exemplo, melhores resultados (desvio de atraso menor) foram alcançados para um maior número de transistores no arranjo.

O método de estimativa da variabilidade proposto foi aplicado principalmente para configurações MOS complementares (CMOS), mas também apresentou resultados importantes para os estilos lógicos DCVSL, PTL e DPTL. Para compor o método, a resistência de cada transistor foi calculada em função das variações de tensão de limiar de todos os dispositivos na rede. Cada topologia requer 2.2^k simulações elétricas DC, o que leva menos de 10 segundos, mesmo para portas lógicas com mais de 10 transistores. A análise de duas regiões diferentes de operação - linear e saturação - a fim de atingir os valores de resistências mostraram que a região linear fornece funções de resistência mais confiáveis.

A análise de redes com transistores em série e paralelo revelou diferenças significativas quando dispositivos N- ou PMOS compõem o arranjo. Redes NMOS têm desvios de atraso que são mais sensíveis às variações na tensão limiar dos transistores. Em geral, um número mais elevado de transistores MOS em uma topologia em série, torna a configuração mais robusta. Em uma rede paralela pura, variações na tensão de limiar de transistores que não estão conduzindo não têm nenhuma influência sobre a variabilidade do atraso.

No caso dos inversores concebidos a partir de diferentes topologias, quando o desvio V_{TH} é alterado de acordo com as dimensões dos transistores (de acordo com o modelo de Pelgrom), torna-se evidente que dispositivos MOS com maior área resultam em menor variabilidade de atraso.

Resultados fornecidos pelas simulações estatísticas, bem como pelo método proposto mostraram que, para uma cadeia de inversores, quanto maior o número de inversores, menores os desvios dos atrasos de subida e descida. Finalmente, a comparação entre diferentes topologias e estilos lógicos usados para se implementar a mesma função lógica - uma XOR de 2 entradas - mostrou que o atraso da configuração com portas NAND sofre menos influência das variações das tensões de limiar dos transistores. A lógica de *transistor de passagem* foi a que apresentou maior desvio de

atraso dentre as topologias analisadas. A portas lógicas também foram analisadas com o método proposto, que forneceu resultados compatíveis com as simulações estatísticas. No caso de uma XOR de 4 entradas, a porta implementada no estilo DCVSL aparece como a configuração mais confiável dentre as analisadas. Para se implementar um somador completo, a porta complexa CMOS é preferível sobre o estilo DPTL, quando o atraso do nó de saída "*SUM*" é analisado.

A fim de se proporcionar a função densidade de probabilidade do atraso de uma topologia, o método proposto exige um tempo de execução que é de cerca de 3.000 vezes menor que o tempo exigido pelas simulações Monte Carlo (10.000 execuções).